Building Text-Based Applications with the ChatGPT API & LangChain

O'REILLY®

How to build LLM apps

Lucas Soares

21-08-2023

Intro

Hi!

- Philosophy, Maths and Cognitive Sciences background
- Machine Learning Engineer

Quick survey to get to
know everyone!

Large Language Models

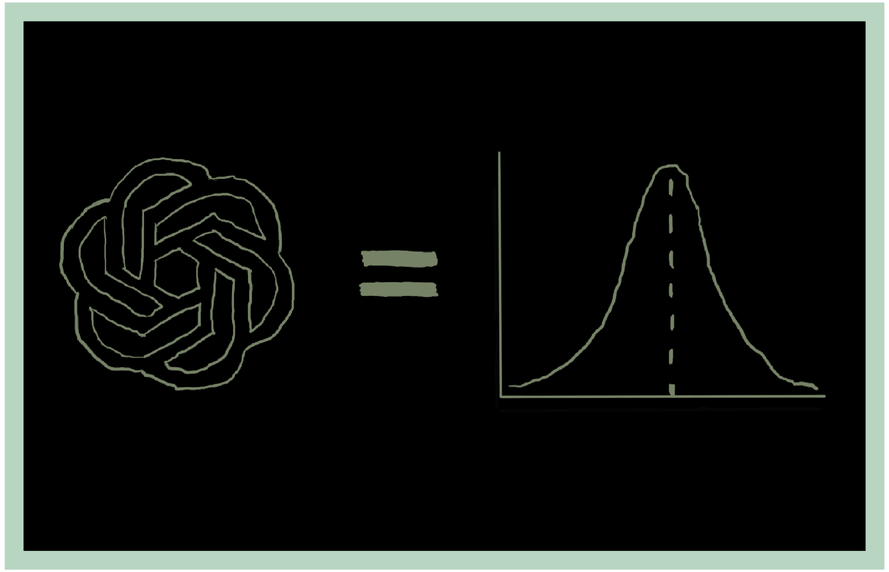
A definition

LLMs are advanced AI systems designed to understand and generate human language.

Large Language Models

A definition

- LLMs are advanced AI systems designed to understand and generate human language.
- They assign probabilities to words based on context, allowing them to predict future words in a sentence.



Large Language Models

As Probability Distributions

At their core, LLMs can be seen as distributions over words.

Large Language Models

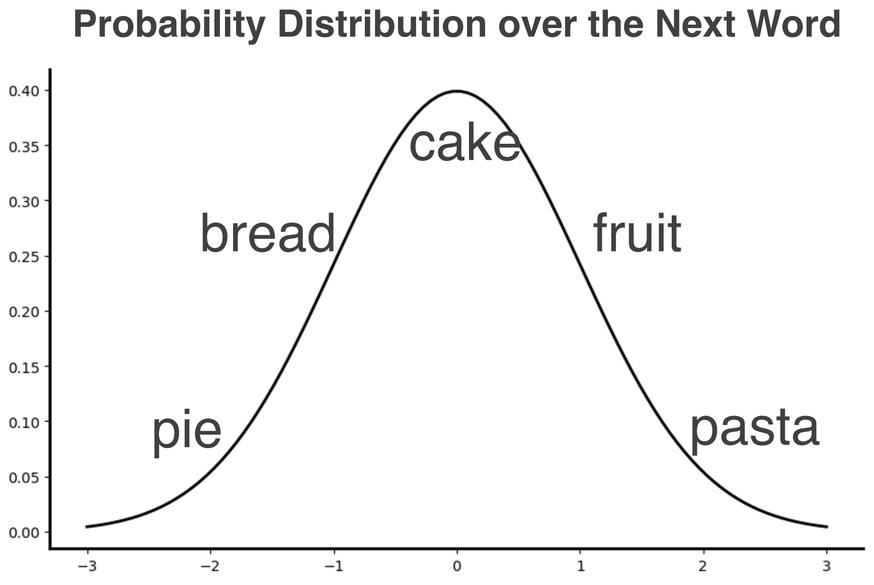
As Probability Distributions

- At their core, LLMs can be seen as distributions over words.
- Use statistical models to capture patterns in text data.

Large Language Models

As Probability Distributions

- At their core, LLMs can be seen as distributions over words.
- Use statistical models to capture patterns in text data.
- They calculate the likelihood of each word occurring given the context.

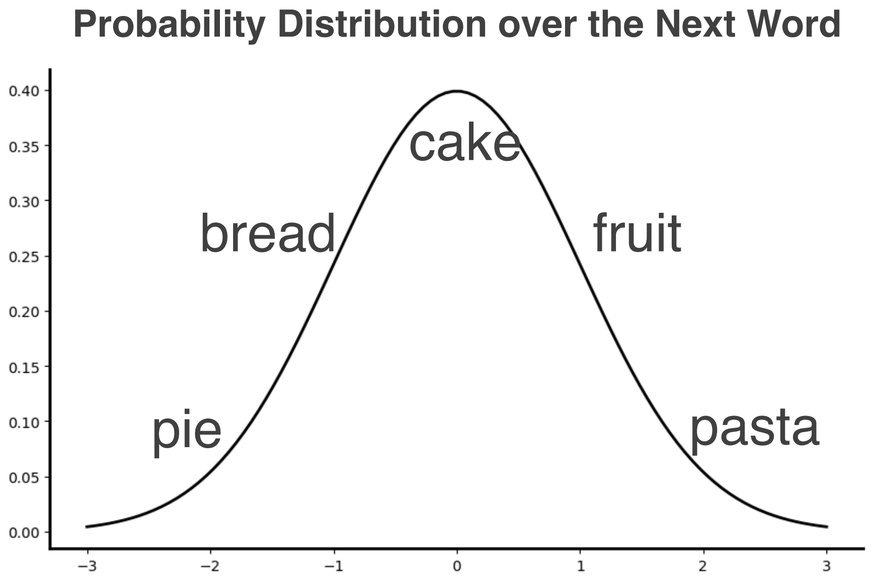


"I love eating…." → ?

Large Language Models

As Probability Distributions

- At their core, LLMs can be seen as distributions over words.
- Use statistical models to capture patterns in text data.
- They calculate the likelihood of each word occurring given the context.



"I love eating…." → ?

Large Language Models

N-gram models

- **Unigram model**: each token/word is independently modeled.

Large Language Models

N-gram models

- **Unigram model**: each token/word is independently modeled.
- Sentence S = *"When the bough breaks, the cradle will fall."*

Unigram Word Probabilities

p = 0.86	p = 0.93	p = 0.22	p = 0.66	p = 0.66	p = 0.37	p = 0.15	p = 0.1
When	the	bough	breaks	the	cradle	will	fall

Large Language Models

N-gram models

```
import random
sentence = "When the bough breaks the craddle will fall"

tokens = sentence.split(" ")
# First, the token_probs list is created with random values
token_probs = [random.random() for _ in tokens]
# Then, the probabilities dictionary is created
# The dictionary has keys of the form 'word1 word2', where 'word1' and
'word2' are consecutive words in the sentence
# The value for each key is the conditional probability of 'word2' given
'word1'
probabilities = {}
for i in range(len(tokens)):
    if tokens[i] in probabilities.keys():
        probabilities[f"{str(tokens[i])+'-2'}"] = round(token_probs[i],2)
    else:
        probabilities[f"{tokens[i]}"] = round(token_probs[i],2)
probabilities
```

Unigram Word Probabilities

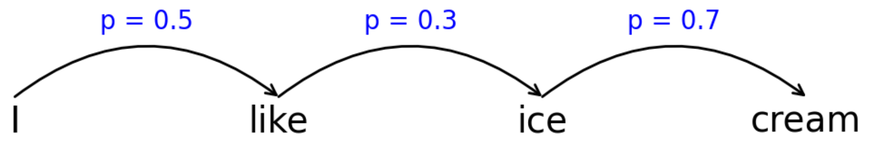
p = 0.86	p = 0.93	p = 0.22	p = 0.66	p = 0.66	p = 0.37	p = 0.15	p = 0.1
When	the	bough	breaks	the	cradle	will	fall

$$P(S) = P(\text{"When"}) \times P(\text{"the"}) \times P(\text{"bough"}) \times P(\text{"breaks"}) \times P(\text{"the"}) \times P(\text{"cradle"}) \times P(\text{"will"}) \times P(\text{"fall"})$$

Large Language Models

N-gram models

- **Bigram model**, the probability of a word depends on its previous word.
- Sentence S = *"I like ice cream."*



The probability of the sentence is the product of the individual probabilities:

$$P(\text{"I like ice cream"}) = P(\text{"I"}) \times P(\text{"like"} \mid \text{"I"}) \times P(\text{"ice"} \mid \text{"like"}) \times P(\text{"cream"} \mid \text{"ice"})$$

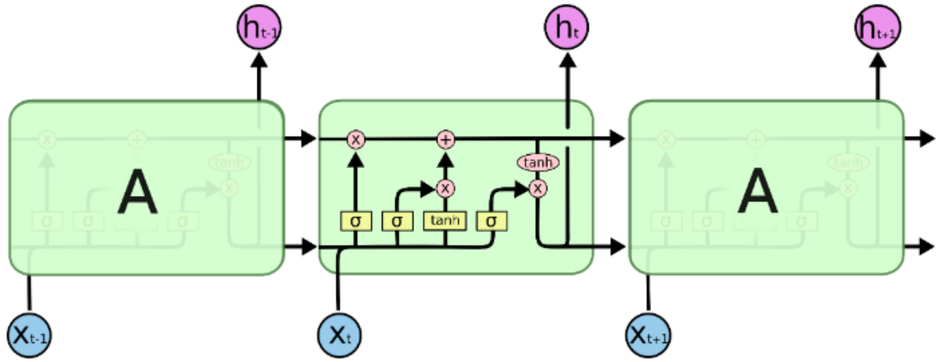
Can't capture long-term relationships

Large Language Models

Sequence to Sequence Models: LSTMs

- **LSTMs**, forgetting mechanism to model longer sentences and maintain context.
  (Hochreiter and Jürgen Schmidhuber 1997)

- Forgetting gates and context maintenance.



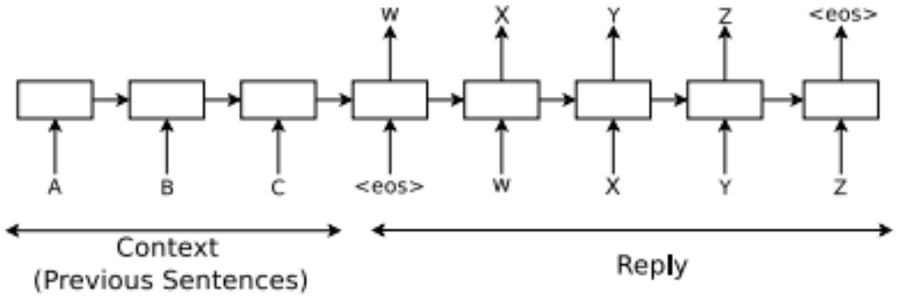
Understanding LSTM Networks by Christopher Olah

The probability of the sentence is now influenced by context.

Large Language Models

Sequence to Sequence Models: LSTMs

- Seq2Seq models require processing input sequentially.
- Struggle with really long complex inputs.



(Vinyals & Le, 2015)

Can’t capture really long contexts!

The complexity of language models increases to account for their ability to consider context when modeling sequences of text.

Essence of LLMs

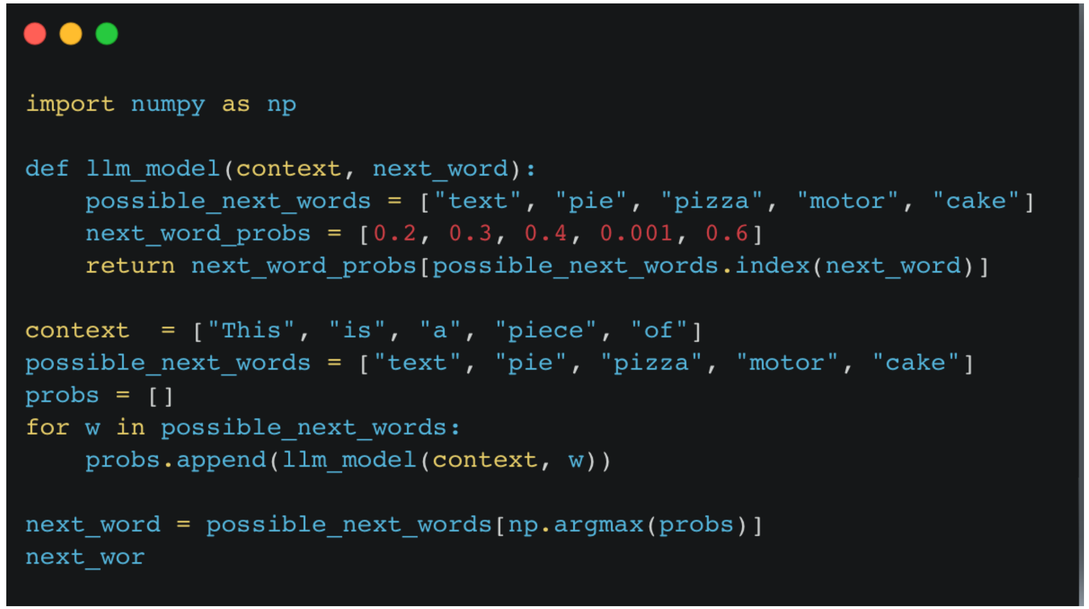
Beyond pattern matching

They capture context, understand dependencies, and predict text based on patterns learned during training.

Ok, but how?

LLMs as pattern matchers

- LLMs are akin to probabilistic programs.
- They predict outcomes based on probabilities learned from training data.



```python
import numpy as np

def llm_model(context, next_word):
    possible_next_words = ["text", "pie", "pizza", "motor", "cake"]
    next_word_probs = [0.2, 0.3, 0.4, 0.001, 0.6]
    return next_word_probs[possible_next_words.index(next_word)]

context  = ["This", "is", "a", "piece", "of"]
possible_next_words = ["text", "pie", "pizza", "motor", "cake"]
probs = []
for w in possible_next_words:
    probs.append(llm_model(context, w))

next_word = possible_next_words[np.argmax(probs)]
next_wor
```


Context/Past
LM
Future

How LLMs work

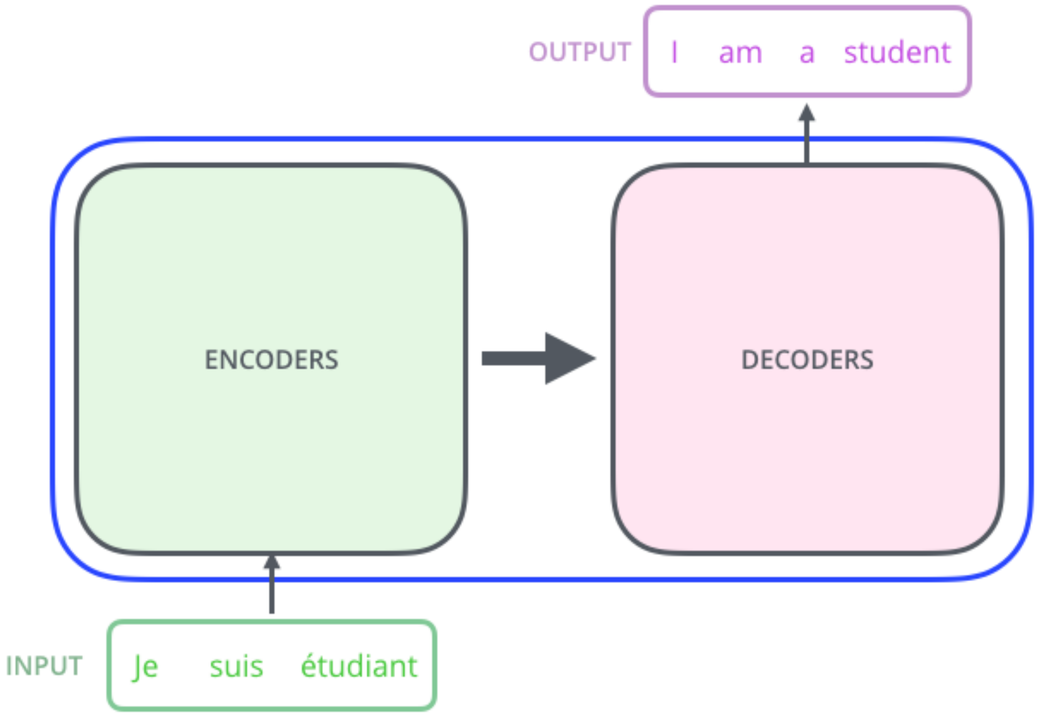
Transformers Architecture

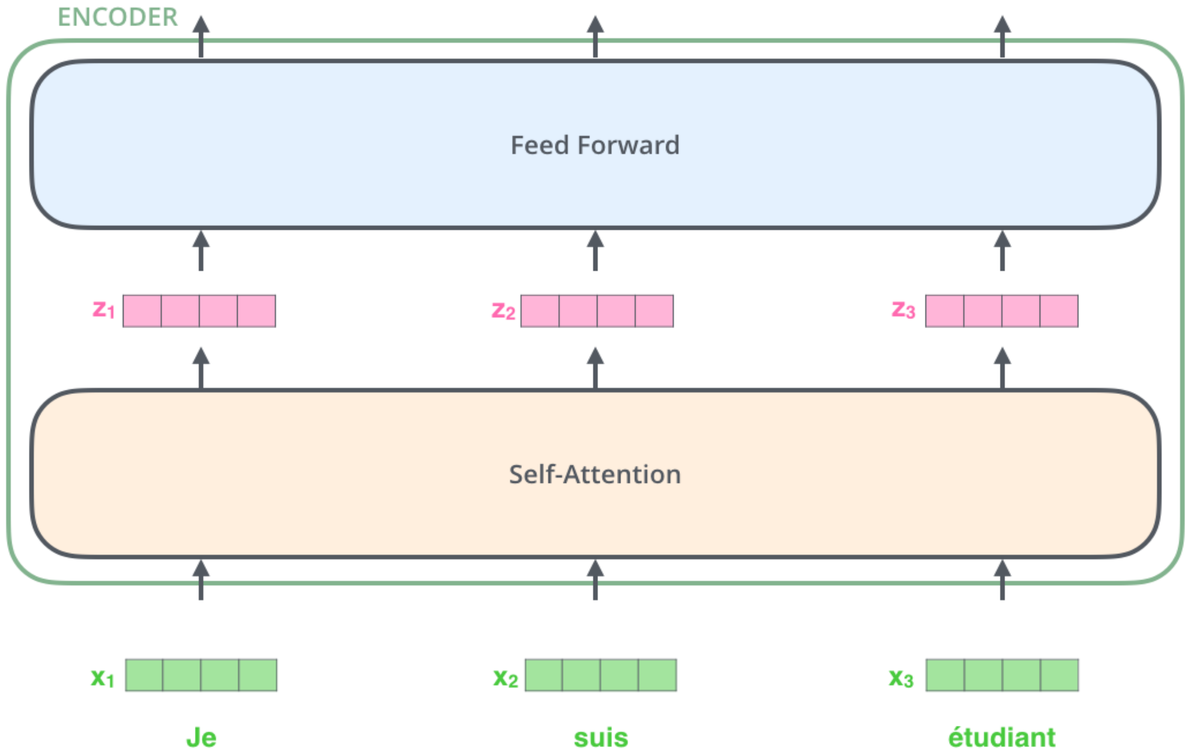
- Traditional sequential models struggle with context.
- Transformers use attention mechanisms to capture global dependencies, enabling contextual understanding.
- The attention mechanism allows Transformers to focus on different parts of input simultaneously.
- Transformers can understand and predict based on long-range dependencies.

(Vaswani et al 2017)

How LLMs work

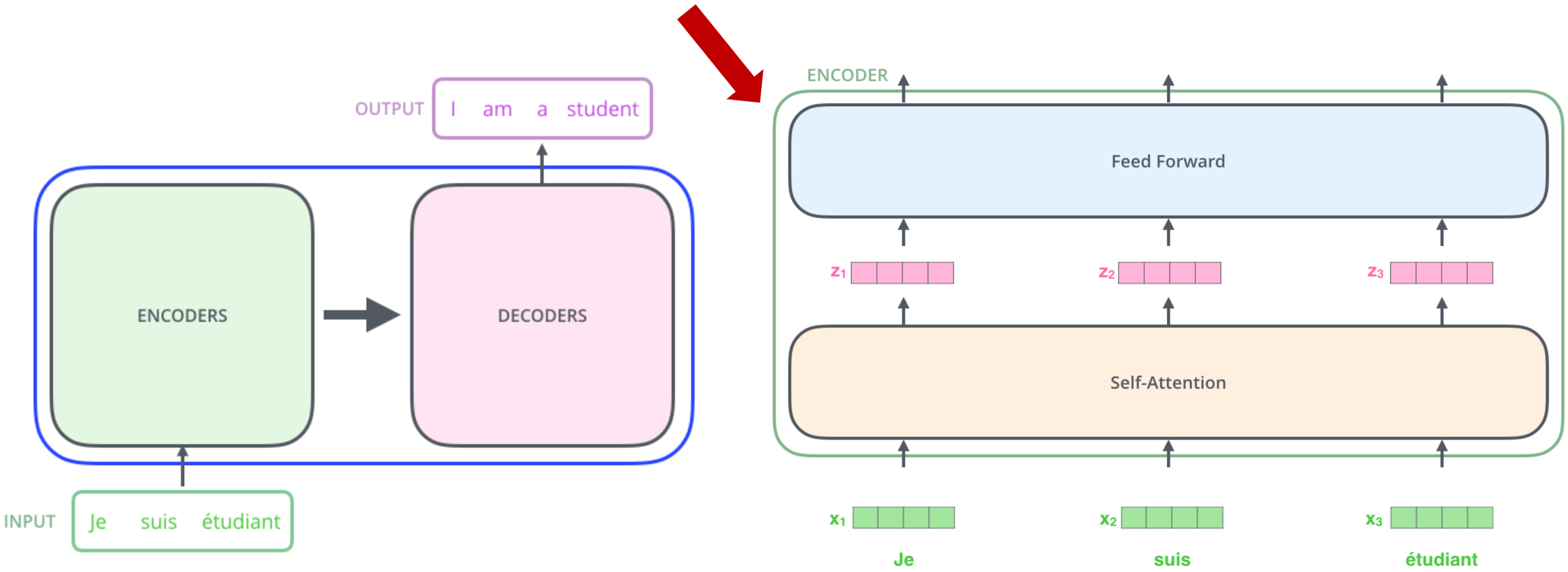
Transformers Architecture





How LLMs work

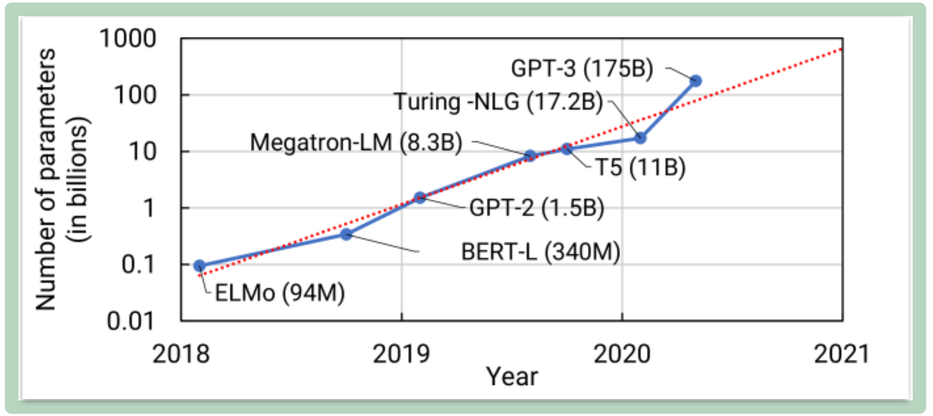
Transformers Architecture



Inputs are processed in parallel!

Why “Large” Language Models?

- Large → Big number of parameters



Benefits of Large Language Models

Multi-task, fine tuning, scalability

- **Multi-task Capability:** LLMs find applications in content generation, question answering, translation, tutoring, and personal assistants.
- **Fine-tunning:** LLMs can usually be fine tuned with a relatively small amount of data, making them adaptable to a wide range of tasks.
- **Scalability**: LLMs demonstrate excellent scalability to very large capacity networks and huge datasets.

Applications of Large Language Models

- Content generation
- Q&A
- Translation
- Tutoring
- Personal Assistants

Applications of LLMs

Content generation Example - Demo

Applications of LLMs

Translation Example - Demo

Limitations and Ethical Considerations

LLMs are far from perfect

- **Knowledge Limit:** LLMs have a cutoff point for their knowledge.
- **Understanding Limit**: LLMs do not understand text in the same way humans do. They don't have beliefs or desires; they simply predict what comes next based on their training.
- **Misuse:** LLMs can hallucinate and produce false or harmful content**.**
- **Reproducibility**: Unpredictability of LLM behavior (Watkins 2023**).**
- **Data Privacy and Bias:** Ethical considerations should extend to the acquisition of data for training additional models. Models may have biases; their use should be transparent and biases mitigated (Watkins 2023**).**

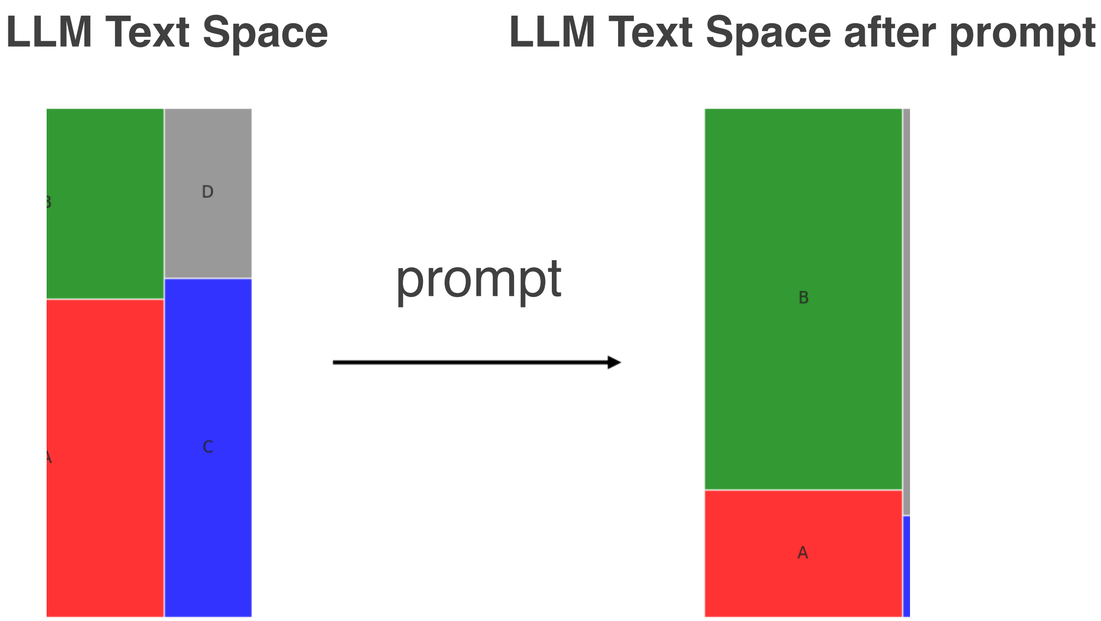
Questions and Discussion

Introduction to prompt engineering and the ChatGPT API

1. Prompt basics
2. Introduction to the ChatGPT API
3. Prompt engineering guide
4. Best practices for writing effective prompts

Prompt Basics

- A prompt is a piece of text that conveys to the LLM the user’s intention.
- Question → Instruction → Behavior
- It constrains the space of possibilities in the LLM's text space



Components of a prompt

Task, input, context, style

- **Task description:** where you describe what you want
- **Input data:** data the model has not seen to illustrate what you need
- **Context information:** background info on what you are requesting, the data you are providing etc
- **Prompt style:** Its about how you ask the thing you want to the model

Introduction to the ChatGPT API

Where does ChatGPT fit in this chaotic LLM universe?

- The ChatGPT API allows us to use OpenAI's chat models to generate dynamic, contextually-aware responses.
- Required parameters: model, messages.

```python
import openai

openai.ChatCompletion.create(
  model="gpt-3.5-turbo",
  messages=[
        {"role": "system", "content": "You are a helpful assistant."},
        {"role": "user", "content": "Who won the world series in 2020?"},
        {"role": "assistant", "content": "The Los Angeles Dodgers won the World Series in 2020."},
        {"role": "user", "content": "Where was it played?"}
    ]
```

2

Introduction to the ChatGPT API

Where does ChatGPT fit in this chaotic LLM universe?

```
!pip install openai

openai.api_key = os.getenv("OPENAI_API_KEY")

import openai

openai.ChatCompletion.create(model="gpt-3.5-turbo",
                             messages=
                             [
                                 {"role": "system", "content": "You are a savy guru with knowledge
about existence and the secrets of life."},
                                 {"role": "user", "content": "What is the meaning of life?"}
                             ],
                             max_tokens=100,
                             temperature=0.9,
                             n = 1)
```

Introduction to the ChatGPT API

Where does ChatGPT fit in this chaotic LLM universe?

```
!pip install openai

openai.api_key = os.getenv("OPENAI_API_KEY")

import openai

openai.ChatCompletion.create(model="gpt-3.5-turbo",
                             messages=
                             [
                                 {"role": "system", "content": "You are a savy guru with knowledge
about existence and the secrets of life."},
                                 {"role": "user", "content": "What is the meaning of life?"}
                             ],
                             max_tokens=100,
                             temperature=0.9,
                             n = 1)
```

Introduction to the ChatGPT API

Where does ChatGPT fit in this chaotic LLM universe?

```
!pip install openai

openai.api_key = os.getenv("OPENAI_API_KEY")

import openai

openai.ChatCompletion.create(model="gpt-3.5-turbo",
                             messages=
                             [
                                 {"role": "system", "content": "You are a savy guru with knowledge
about existence and the secrets of life."},
                                 {"role": "user", "content": "What is the meaning of life?"}
                             ],
                             max_tokens=100,
                             temperature=0.9,
                             n = 1)
```

Introduction to the ChatGPT API

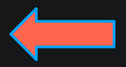
Where does ChatGPT fit in this chaotic LLM universe?

```
!pip install openai

openai.api_key = os.getenv("OPENAI_API_KEY")

import openai

openai.ChatCompletion.create(model="gpt-3.5-turbo",
                             messages=
                             [
                                 {"role": "system", "content": "You are a savy guru with knowledge
about existence and the secrets of life."},
                                 {"role": "user", "content": "What is the meaning of life?"}
                             ],
                             max_tokens=100,
                             temperature=0.9,
                             n = 1)
```

Introduction to the ChatGPT API

Where does ChatGPT fit in this chaotic LLM universe?

```
!pip install openai

openai.api_key = os.getenv("OPENAI_API_KEY")

import openai

openai.ChatCompletion.create(model="gpt-3.5-turbo",
                             messages=
                             [
                                 {"role": "system", "content": "You are a savy guru with knowledge
about existence and the secrets of life."},
                                 {"role": "user", "content": "What is the meaning of life?"}
                             ],
                             max_tokens=100,
                             temperature=0.9,
                             n = 1)
```



Introduction to the ChatGPT API

Where does ChatGPT fit in this chaotic LLM universe?

```
!pip install openai

openai.api_key = os.getenv("OPENAI_API_KEY")

import openai

openai.ChatCompletion.create(model="gpt-3.5-turbo",
                             messages=
                             [
                                 {"role": "system", "content": "You are a savy guru with knowledge
about existence and the secrets of life."},
                                 {"role": "user", "content": "What is the meaning of life?"}
                             ],
                             max_tokens=100,
                             temperature=0.9,
                             n = 1)
```

Introduction to the ChatGPT API

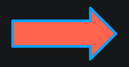
Where does ChatGPT fit in this chaotic LLM universe?

```
!pip install openai

openai.api_key = os.getenv("OPENAI_API_KEY")

import openai

openai.ChatCompletion.create(model="gpt-3.5-turbo",
                             messages=
                             [
                                 {"role": "system", "content": "You are a savy guru with knowledge
about existence and the secrets of life."},
                                 {"role": "user", "content": "What is the meaning of life?"}
                             ],
                             max_tokens=100,
                             temperature=0.9,
                             n = 1)
```

Introduction to the ChatGPT API

Where does ChatGPT fit in this chaotic LLM universe?

```
!pip install openai

openai.api_key = os.getenv("OPENAI_API_KEY")

import openai

openai.ChatCompletion.create(model="gpt-3.5-turbo",
                             messages=
                             [
                                 {"role": "system", "content": "You are a savy guru with knowledge
about existence and the secrets of life."},
                                 {"role": "user", "content": "What is the meaning of life?"}
                             ],
                             max_tokens=100,
                             temperature=0.9,
                             n = 1)
```

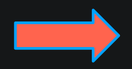


2

Introduction to the ChatGPT API

Where does ChatGPT fit in this chaotic LLM universe?

```
!pip install openai

openai.api_key = os.getenv("OPENAI_API_KEY")

import openai

openai.ChatCompletion.create(model="gpt-3.5-turbo",
                             messages=
                             [
                                 {"role": "system", "content": "You are a savy guru with knowledge
about existence and the secrets of life."},
                                 {"role": "user", "content": "What is the meaning of life?"}
                             ],
                             max_tokens=100,
                             temperature=0.9,
                             n = 1)
```



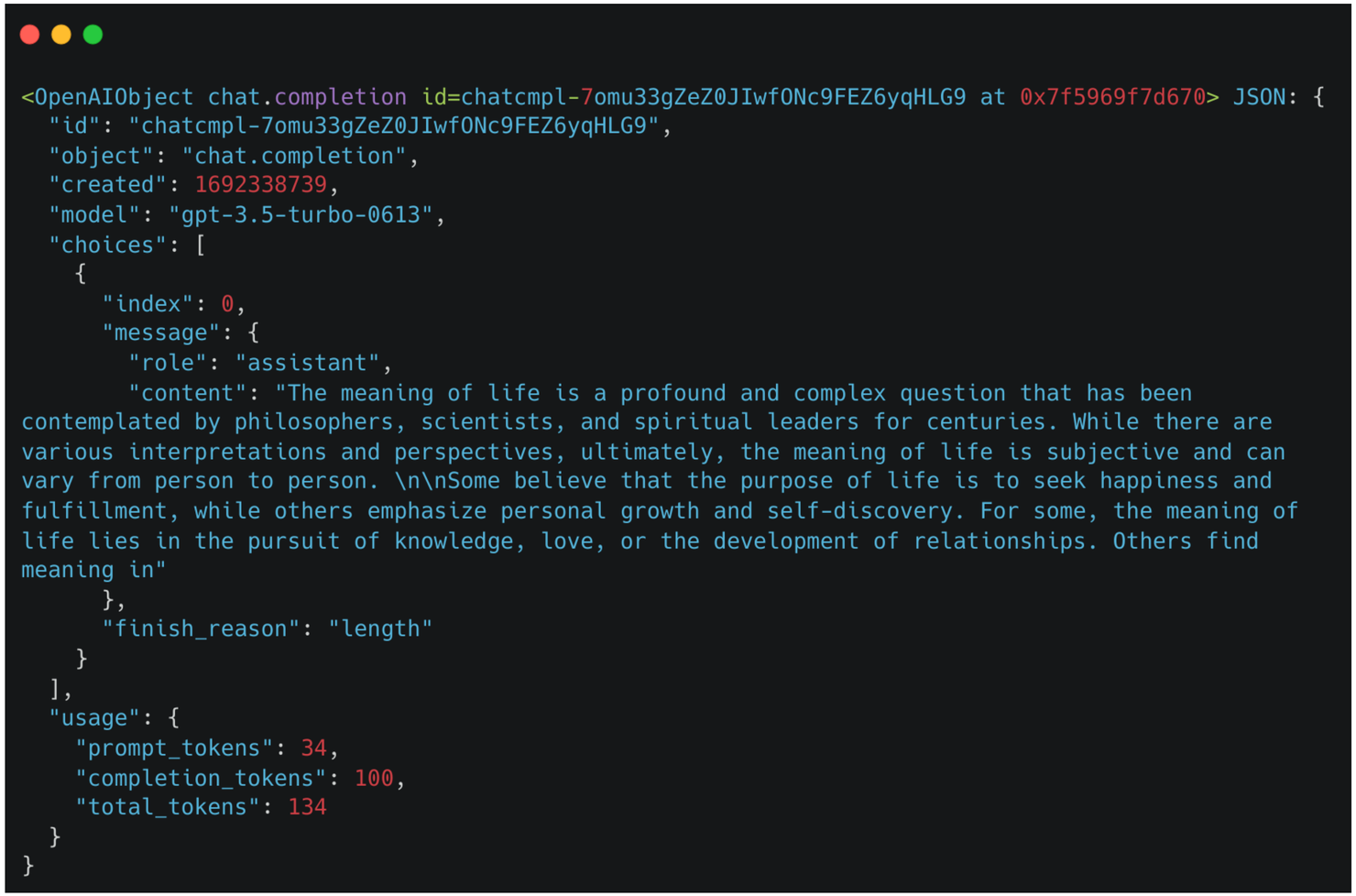
Introduction to the ChatGPT API

Where does ChatGPT fit in this chaotic LLM universe?

```
!pip install openai

openai.api_key = os.getenv("OPENAI_API_KEY")

import openai

openai.ChatCompletion.create(model="gpt-3.5-turbo",
                             messages=
                             [
                                 {"role": "system", "content": "You are a savy guru with knowledge
about existence and the secrets of life."},
                                 {"role": "user", "content": "What is the meaning of life?"}
                             ],
                             max_tokens=100,
                             temperature=0.9,
                             n = 1)
```

Introduction to the ChatGPT API

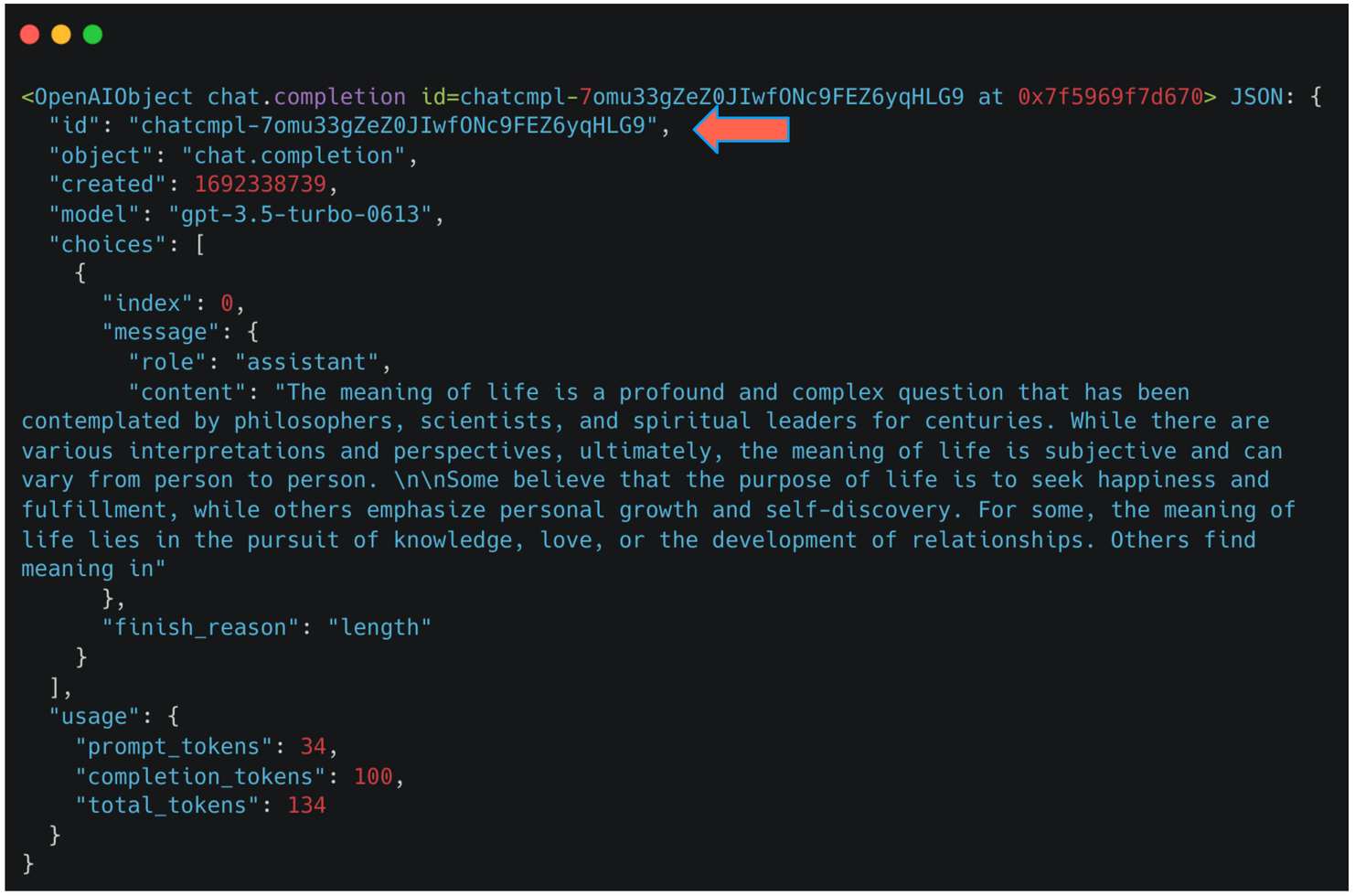
A breakdown of the response



```
<OpenAIObject chat.completion id=chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9 at 0x7f5969f7d670> JSON: {
  "id": "chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9",
  "object": "chat.completion",
  "created": 1692338739,
  "model": "gpt-3.5-turbo-0613",
  "choices": [
    {
      "index": 0,
      "message": {
        "role": "assistant",
        "content": "The meaning of life is a profound and complex question that has been
contemplated by philosophers, scientists, and spiritual leaders for centuries. While there are
various interpretations and perspectives, ultimately, the meaning of life is subjective and can
vary from person to person. \n\nSome believe that the purpose of life is to seek happiness and
fulfillment, while others emphasize personal growth and self-discovery. For some, the meaning of
life lies in the pursuit of knowledge, love, or the development of relationships. Others find
meaning in"
      },
      "finish_reason": "length"
    }
  ],
  "usage": {
    "prompt_tokens": 34,
    "completion_tokens": 100,
    "total_tokens": 134
  }
}
```

Introduction to the ChatGPT API

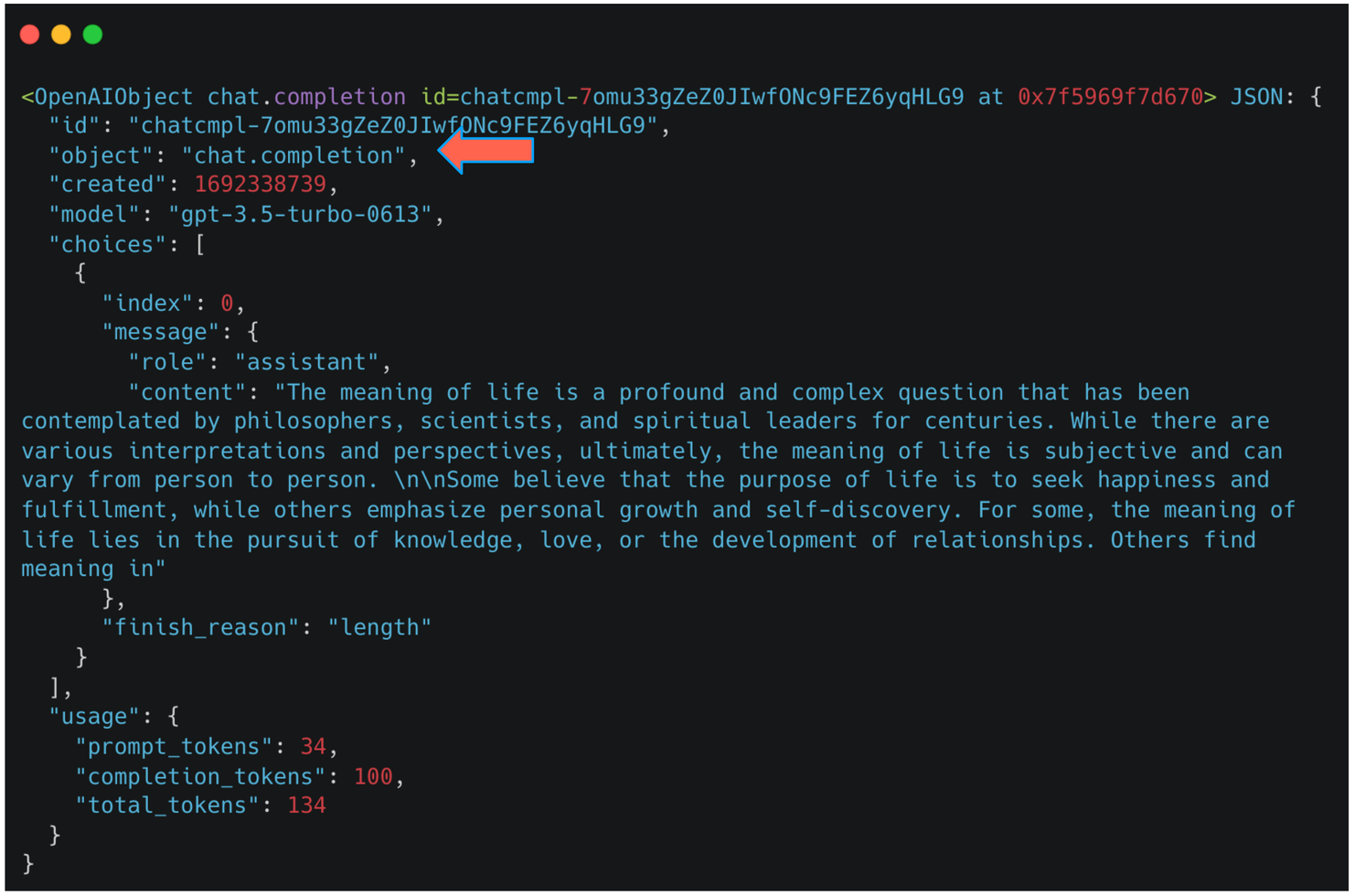
A breakdown of the response



```
<OpenAIObject chat.completion id=chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9 at 0x7f5969f7d670> JSON: {
  "id": "chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9",
  "object": "chat.completion",
  "created": 1692338739,
  "model": "gpt-3.5-turbo-0613",
  "choices": [
    {
      "index": 0,
      "message": {
        "role": "assistant",
        "content": "The meaning of life is a profound and complex question that has been
contemplated by philosophers, scientists, and spiritual leaders for centuries. While there are
various interpretations and perspectives, ultimately, the meaning of life is subjective and can
vary from person to person. \n\nSome believe that the purpose of life is to seek happiness and
fulfillment, while others emphasize personal growth and self-discovery. For some, the meaning of
life lies in the pursuit of knowledge, love, or the development of relationships. Others find
meaning in"
      },
      "finish_reason": "length"
    }
  ],
  "usage": {
    "prompt_tokens": 34,
    "completion_tokens": 100,
    "total_tokens": 134
  }
}
```

Introduction to the ChatGPT API

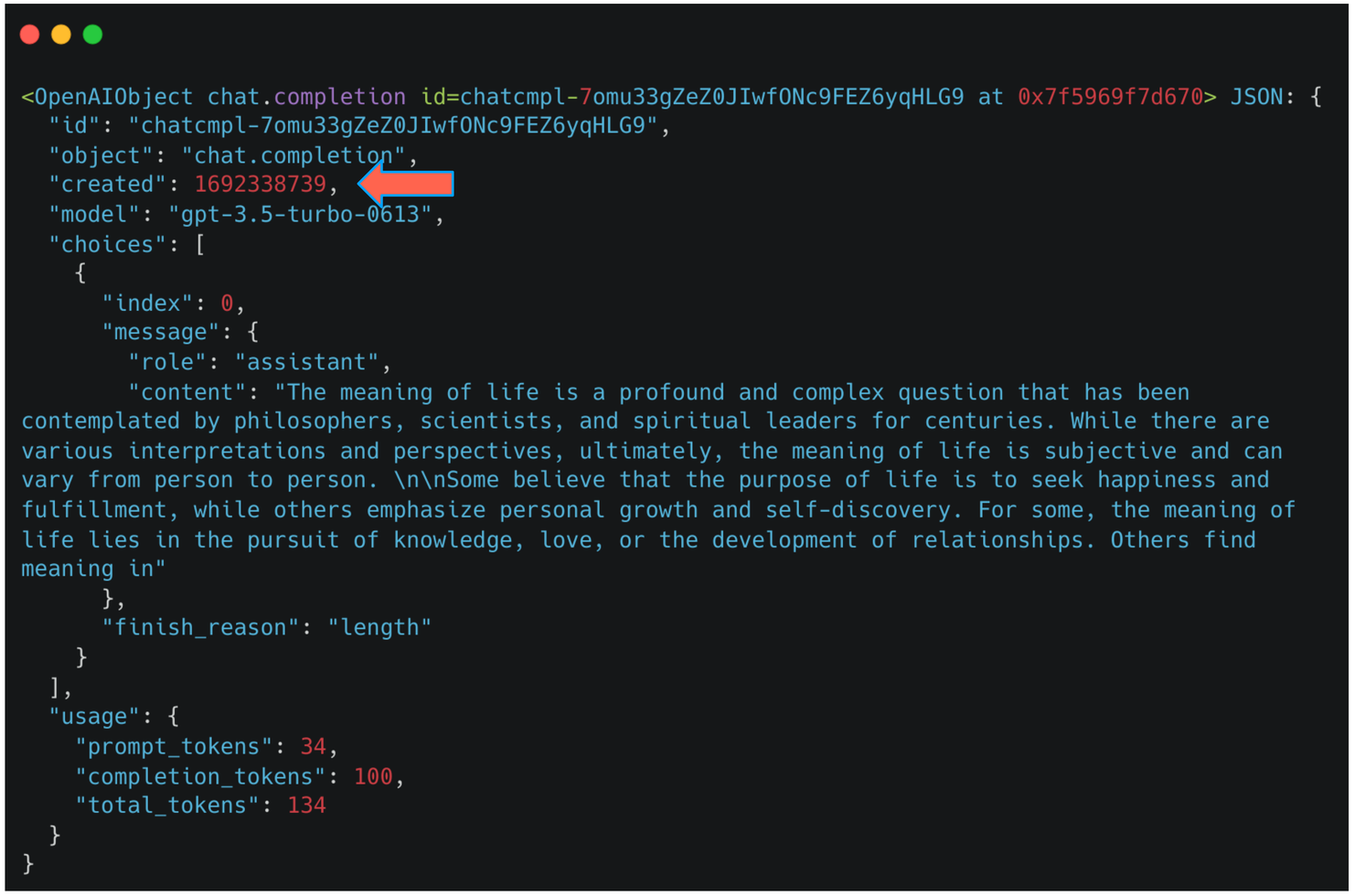
A breakdown of the response



```
<OpenAIObject chat.completion id=chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9 at 0x7f5969f7d670> JSON: {
  "id": "chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9",
  "object": "chat.completion",
  "created": 1692338739,
  "model": "gpt-3.5-turbo-0613",
  "choices": [
    {
      "index": 0,
      "message": {
        "role": "assistant",
        "content": "The meaning of life is a profound and complex question that has been
contemplated by philosophers, scientists, and spiritual leaders for centuries. While there are
various interpretations and perspectives, ultimately, the meaning of life is subjective and can
vary from person to person. \n\nSome believe that the purpose of life is to seek happiness and
fulfillment, while others emphasize personal growth and self-discovery. For some, the meaning of
life lies in the pursuit of knowledge, love, or the development of relationships. Others find
meaning in"
      },
      "finish_reason": "length"
    }
  ],
  "usage": {
    "prompt_tokens": 34,
    "completion_tokens": 100,
    "total_tokens": 134
  }
}
```

Introduction to the ChatGPT API

A breakdown of the response



```
<OpenAIObject chat.completion id=chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9 at 0x7f5969f7d670> JSON: {
  "id": "chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9",
  "object": "chat.completion",
  "created": 1692338739,
  "model": "gpt-3.5-turbo-0613",
  "choices": [
    {
      "index": 0,
      "message": {
        "role": "assistant",
        "content": "The meaning of life is a profound and complex question that has been
contemplated by philosophers, scientists, and spiritual leaders for centuries. While there are
various interpretations and perspectives, ultimately, the meaning of life is subjective and can
vary from person to person. \n\nSome believe that the purpose of life is to seek happiness and
fulfillment, while others emphasize personal growth and self-discovery. For some, the meaning of
life lies in the pursuit of knowledge, love, or the development of relationships. Others find
meaning in"
      },
      "finish_reason": "length"
    }
  ],
  "usage": {
    "prompt_tokens": 34,
    "completion_tokens": 100,
    "total_tokens": 134
  }
}
```

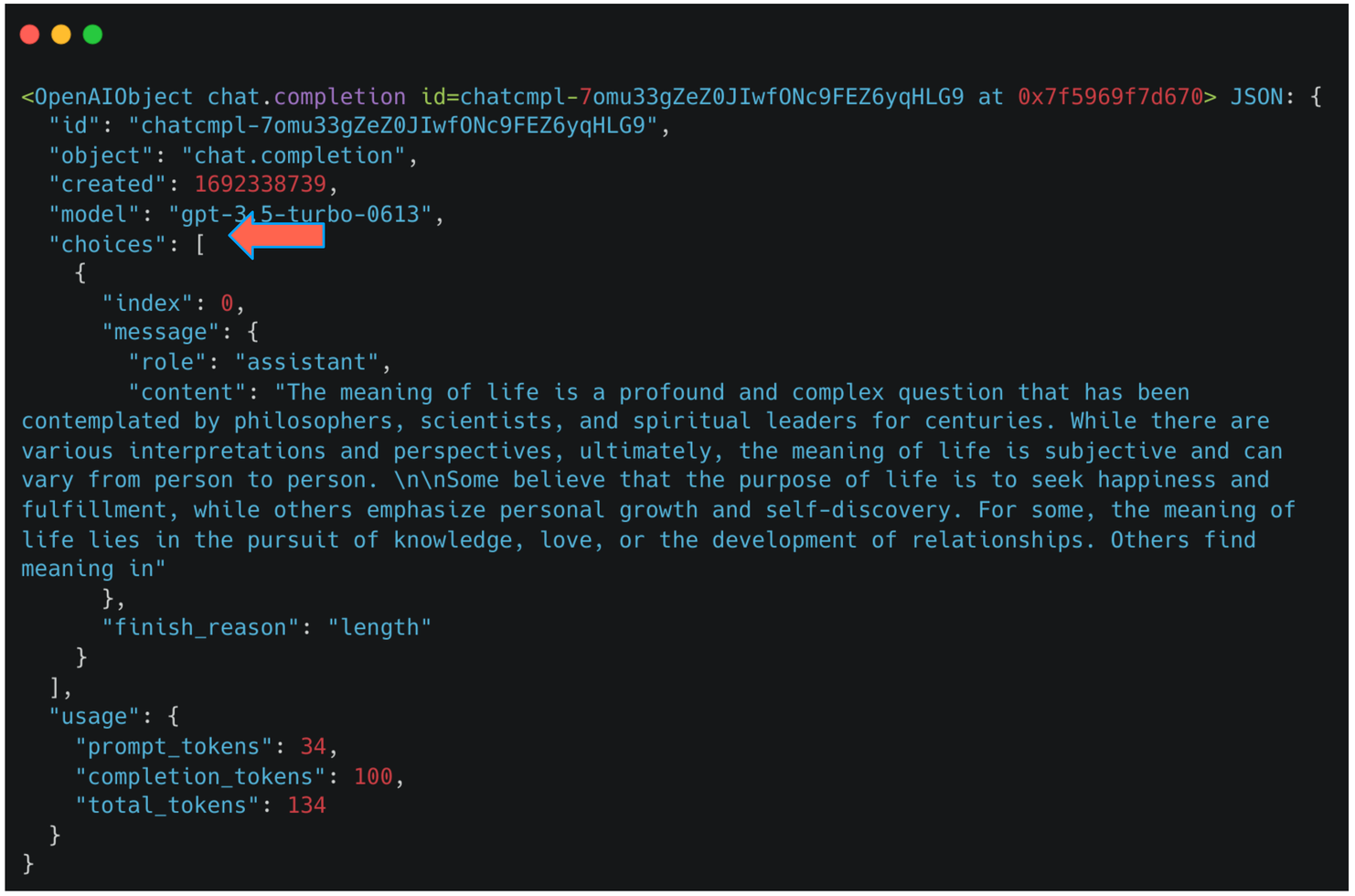
Introduction to the ChatGPT API

A breakdown of the response

```
<OpenAIObject chat.completion id=chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9 at 0x7f5969f7d670> JSON: {
  "id": "chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9",
  "object": "chat.completion",
  "created": 1692338739,
  "model": "gpt-3.5-turbo-0613",
  "choices": [
    {
      "index": 0,
      "message": {
        "role": "assistant",
        "content": "The meaning of life is a profound and complex question that has been
contemplated by philosophers, scientists, and spiritual leaders for centuries. While there are
various interpretations and perspectives, ultimately, the meaning of life is subjective and can
vary from person to person. \n\nSome believe that the purpose of life is to seek happiness and
fulfillment, while others emphasize personal growth and self-discovery. For some, the meaning of
life lies in the pursuit of knowledge, love, or the development of relationships. Others find
meaning in"
      },
      "finish_reason": "length"
    }
  ],
  "usage": {
    "prompt_tokens": 34,
    "completion_tokens": 100,
    "total_tokens": 134
  }
}
```

Introduction to the ChatGPT API

A breakdown of the response



```
<OpenAIObject chat.completion id=chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9 at 0x7f5969f7d670> JSON: {
  "id": "chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9",
  "object": "chat.completion",
  "created": 1692338739,
  "model": "gpt-3.5-turbo-0613",
  "choices": [
    {
      "index": 0,
      "message": {
        "role": "assistant",
        "content": "The meaning of life is a profound and complex question that has been
contemplated by philosophers, scientists, and spiritual leaders for centuries. While there are
various interpretations and perspectives, ultimately, the meaning of life is subjective and can
vary from person to person. \n\nSome believe that the purpose of life is to seek happiness and
fulfillment, while others emphasize personal growth and self-discovery. For some, the meaning of
life lies in the pursuit of knowledge, love, or the development of relationships. Others find
meaning in"
      },
      "finish_reason": "length"
    }
  ],
  "usage": {
    "prompt_tokens": 34,
    "completion_tokens": 100,
    "total_tokens": 134
  }
}
```

Introduction to the ChatGPT API

A breakdown of the response

```
<OpenAIObject chat.completion id=chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9 at 0x7f5969f7d670> JSON: {
  "id": "chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9",
  "object": "chat.completion",
  "created": 1692338739,
  "model": "gpt-3.5-turbo-0613",
  "choices": [
    {
      "index": 0,
      "message": {
        "role": "assistant",
        "content": "The meaning of life is a profound and complex question that has been
contemplated by philosophers, scientists, and spiritual leaders for centuries. While there are
various interpretations and perspectives, ultimately, the meaning of life is subjective and can
vary from person to person. \n\nSome believe that the purpose of life is to seek happiness and
fulfillment, while others emphasize personal growth and self-discovery. For some, the meaning of
life lies in the pursuit of knowledge, love, or the development of relationships. Others find
meaning in"
      },
      "finish_reason": "length"
    }
  ],
  "usage": {
    "prompt_tokens": 34,
    "completion_tokens": 100,
    "total_tokens": 134
  }
}
```

Introduction to the ChatGPT API

A breakdown of the response

```
<OpenAIObject chat.completion id=chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9 at 0x7f5969f7d670> JSON: {
  "id": "chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9",
  "object": "chat.completion",
  "created": 1692338739,
  "model": "gpt-3.5-turbo-0613",
  "choices": [
    {
      "index": 0,
      "message": {
        "role": "assistant",
        "content": "The meaning of life is a profound and complex question that has been
contemplated by philosophers, scientists, and spiritual leaders for centuries. While there are
various interpretations and perspectives, ultimately, the meaning of life is subjective and can
vary from person to person. \n\nSome believe that the purpose of life is to seek happiness and
fulfillment, while others emphasize personal growth and self-discovery. For some, the meaning of
life lies in the pursuit of knowledge, love, or the development of relationships. Others find
meaning in"
      },
      "finish_reason": "length"
    }
  ],
  "usage": {
    "prompt_tokens": 34,
    "completion_tokens": 100,
    "total_tokens": 134
  }
}
```

Introduction to the ChatGPT API

A breakdown of the response


```
<OpenAIObject chat.completion id=chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9 at 0x7f5969f7d670> JSON: {
  "id": "chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9",
  "object": "chat.completion",
  "created": 1692338739,
  "model": "gpt-3.5-turbo-0613",
  "choices": [
    {
      "index": 0,
      "message": {
        "role": "assistant",
        "content": "The meaning of life is a profound and complex question that has been
contemplated by philosophers, scientists, and spiritual leaders for centuries. While there are
various interpretations and perspectives, ultimately, the meaning of life is subjective and can
vary from person to person. \n\nSome believe that the purpose of life is to seek happiness and
fulfillment, while others emphasize personal growth and self-discovery. For some, the meaning of
life lies in the pursuit of knowledge, love, or the development of relationships. Others find
meaning in"
      },
      "finish_reason": "length"
    }
  ],
  "usage": {
    "prompt_tokens": 34,
    "completion_tokens": 100,
    "total_tokens": 134
  }
}
```

Introduction to the ChatGPT API

A breakdown of the response

<OpenAIObject chat.completion id=chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9 at 0x7f5969f7d670> JSON: {
  "id": "chatcmpl-7omu33gZeZ0JIwfONc9FEZ6yqHLG9",
  "object": "chat.completion",
  "created": 1692338739,
  "model": "gpt-3.5-turbo-0613",
  "choices": [
    {
      "index": 0,
      "message": {
        "role": "assistant",
        "content": "The meaning of life is a profound and complex question that has been contemplated by philosophers, scientists, and spiritual leaders for centuries. While there are various interpretations and perspectives, ultimately, the meaning of life is subjective and can vary from person to person. \n\nSome believe that the purpose of life is to seek happiness and fulfillment, while others emphasize personal growth and self-discovery. For some, the meaning of life lies in the pursuit of knowledge, love, or the development of relationships. Others find meaning in"
      },
      "finish_reason": "length"
    }
  ],
  "usage": {
    "prompt_tokens": 34,
    "completion_tokens": 100,
    "total_tokens": 134
  }
}

2

Introduction to the ChatGPT API

A breakdown of the response

Notebook demo

Prompt Engineering Guide

What is prompt engineering?

- **Prompt engineering:** discipline for engineering prompts
- Means by which LLMs can be programmed through prompting.
- Process of creating a prompting function that results in the most effective performance on the downstream task.

The basic goal of prompt engineering is designing appropriate inputs for prompting methods.

Prompt Engineering Techniques

A simplified guide of prompting techniques

- Zero-shot Prompting
- Few-shot Prompting
- Chain-of-Thought
- Self-Consistency
- Generate Knowledge
- Tree of thoughts (ToT)

Zero-shot Prompting

- Zero-shot prompting is when you solve the task without showing any examples of what a solution might look like

- One can use this as the first try at a model to see what kind of tasks LLM can already solve out of the box

Zero-shot Prompting

Example

Classify the sentiment in this sentence as negative or positive:
Text: I will go to a vacation.
Sentiment:

Few-shot Prompting

Provide information in the form of examples to the LLM

Few-shot Prompting: technique where you show a few examples of what a solution might look like.

Few-shot Prompting

Example

A "whatpu" is a small, furry animal native to Tanzania.

An example of a sentence that uses the word whatpu is: We were traveling in Africa and we saw these very cute whatpus.

To do a "farduddle" means to jump up and down really fast.

An example of a sentence that uses the word farduddle is:

Example taken from (Brown et al. 2020)

Chain-of-Thought

Induce step-by-step reasoning and planning

Chain-of-thought (CoT) enables complex reasoning capabilities through intermediate reasoning steps (Wei et al. 2022).

Chain-of-Thought

Example

Q: I have one sister and one brother. I am 20 years of age. My sister is 5 years older and my brother 2 years younger than my sister. How old is my brother?

A: If I am 20 years of age and my sister is 5 years older, my sister is 20+5=25 years old. If my brother is 2 years younger than my sister, my brother is 25-2=23 years old. The answer is 23 years old.

Q: I have 2 friends, Jack and Sally. Jack is 2 years older than Sally. Sally is 5 years younger than me. I am 17 years old. How old is Jack?

There are many more prompt engineering techniques that grow in complexity, such as:

- Self-Consistency
- Generate Knowledge
- ToT
- Retrieval Augmented Generation (RAG)
- Automatic Prompt Engineer
- Active Prompt
- Directional Stimulus Prompting
- React Prompting
- Mulitmodal CoT
- Graph Prompting

A Framework for Building Good Prompts

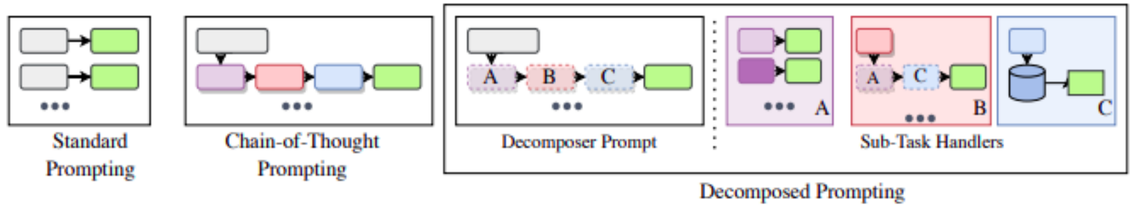
- Operate on Structured Text

```
<python 3 shebang>

<module docstring>

<imports>


<do not include email dunder>

<initialize dotenv>
<set key using OPENAI_API KEY env var>

def complete(prompt: str, **openai_kwargs) -> str:
    <one-line docstring; no params>
    <use default kwargs: model=text-davinci-003, top_p=0.7,
max_tokens=512, 'engine' parameter is deprecated>
    <get completion>
    <strip whitespace before returning>

<as script, demo using prompt "English: Hello\nFrench:">
```

@goodside

A Framework for Building Good Prompts

- Operate on Structured Text
- Introduce decomposition



QC: Concatenate the first letter of every word in "Jack Ryan" using spaces
Q1: [split] What are the words in "Jack Ryan"?
#1: ["Jack". "Ryan"]
Q2: (foreach) [str_pos] What is the first letter of #1?
#2: ["J", "R"]
Q3: [merge] Concatenate #2 with spaces
#3: "J R"
Q4: [EOQ]
...
decomp

Q: What are the words in "Elon Musk Tesla"?
A: ["Elon", "Musk", "Tesla"]

Q: What are the letters in "C++"?
A: ["C", "+", "+"]
...
split

Q: Concatenate ["n", "i", "e"]
A: "nie"

Q: Concatenate ["n", "i", "c", "e"] using spaces
A: "n i c e"
...
merge

Khot et al 2023

A Framework for Building Good Prompts

- Operate on Structured Text
- Introduce decomposition

Self-Ask

GPT-3

Question: Who lived longer, Theodor Haecker or Harry Vaughan Watkins?
Are follow up questions needed here: Yes.
Follow up: How old was Theodor Haecker when he died?
Intermediate answer: Theodor Haecker was 65 years old when he died.
Follow up: How old was Harry Vaughan Watkins when he died?
Intermediate answer: Harry Vaughan Watkins was 69 years old when he died.
So the final answer is: Harry Vaughan Watkins

Question: Who was president of the U.S. when superconductivity was discovered?
Are follow up questions needed here: Yes.
Follow up: When was superconductivity discovered?
Intermediate answer: Superconductivity was discovered in 1911.
Follow up: Who was president of the U.S. in 1911?
Intermediate answer: William Howard Taft.
So the final answer is: William Howard Taft.

Press et al 2023

A Framework for Building Good Prompts

- Operate on Structured Text
- Introduce decomposition

(d) Zero-shot-CoT (Ours)

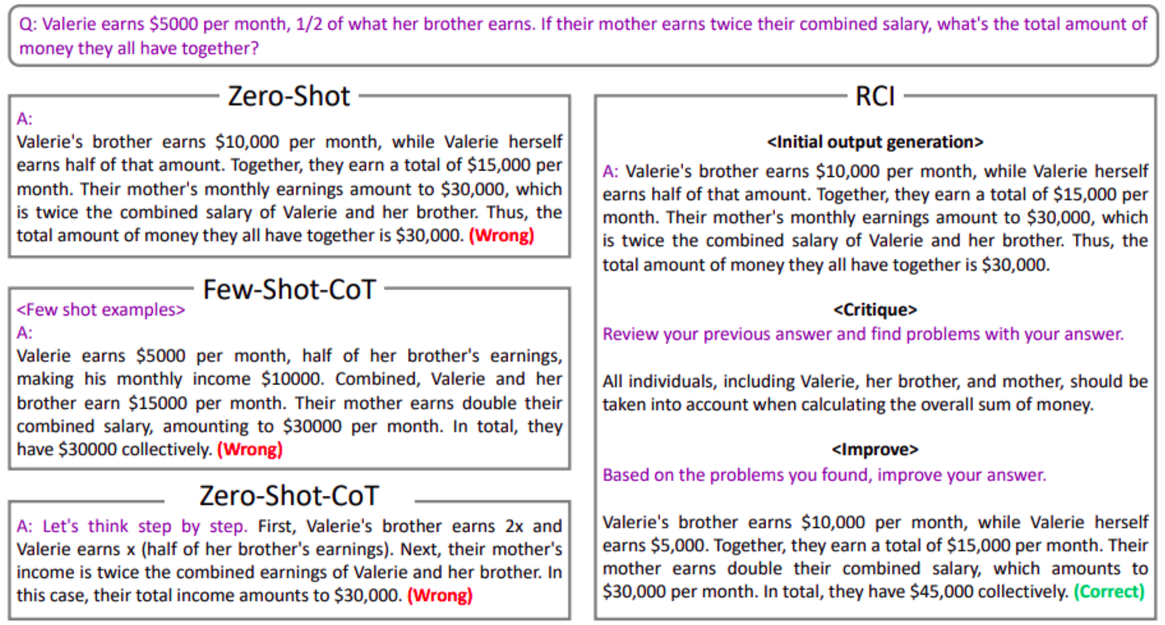
Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A: ***Let's think step by step.***

(Output) There are 16 balls in total. Half of the balls are golf balls. That means that there are 8 golf balls. Half of the golf balls are blue. That means that there are 4 blue golf balls. ✔

Kojima et al 2023

A Framework for Building Good Prompts

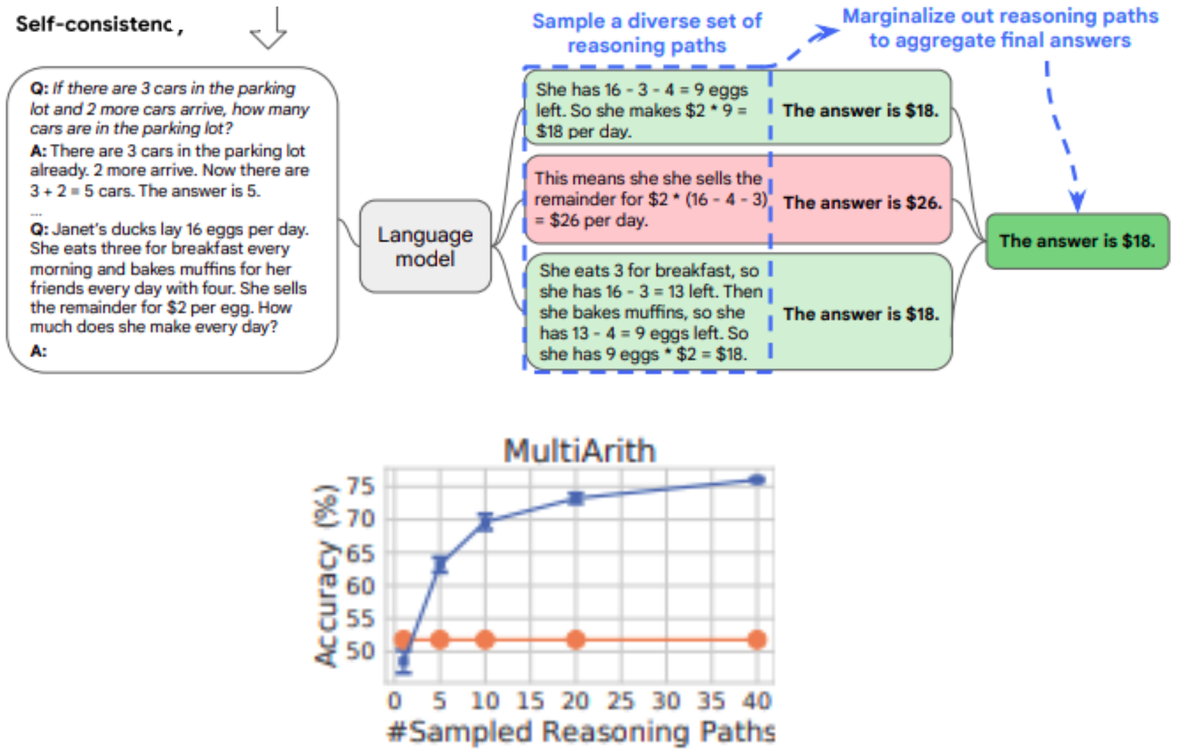
- Operate on Structured Text
- Introduce decomposition
- Self-criticism



Kim, Baldi & McAleer 2023

A Framework for Building Good Prompts

- Operate on Structured Text
- Introduce decomposition
- Self-criticism
- Ensembling

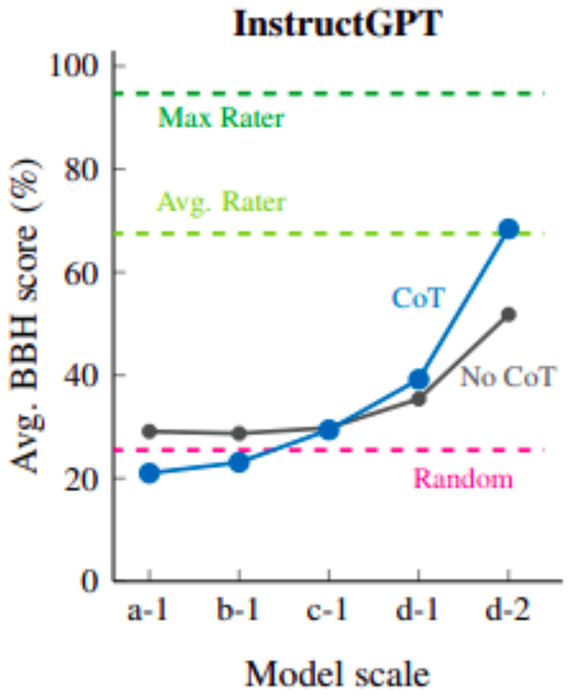


Wang et al 2023

A Framework for Building Good Prompts

- Operate on Structured Text
- Introduce decomposition
- Self-criticism
- Ensembling

Combine for better performance!

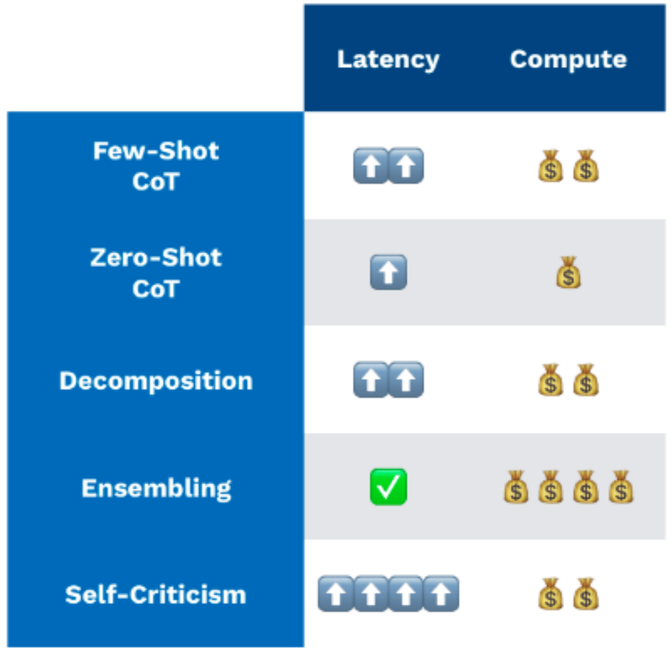


Suzgun et al 2022

A Framework for Building Good Prompts

- Operate on Structured Text
- Introduce decomposition
- Self-criticism
- Ensembling

Be aware of quality-cost tradeoff!



FSDL LLM Bootcamp 2023

Exercise / Lab

- Getting started with prompt engineering using the ChatGPT API
- Prompt engineering for text summarization and question answering
- Extracting dates from unstructured data

Building Blocks of LLM Apps

2 core concepts

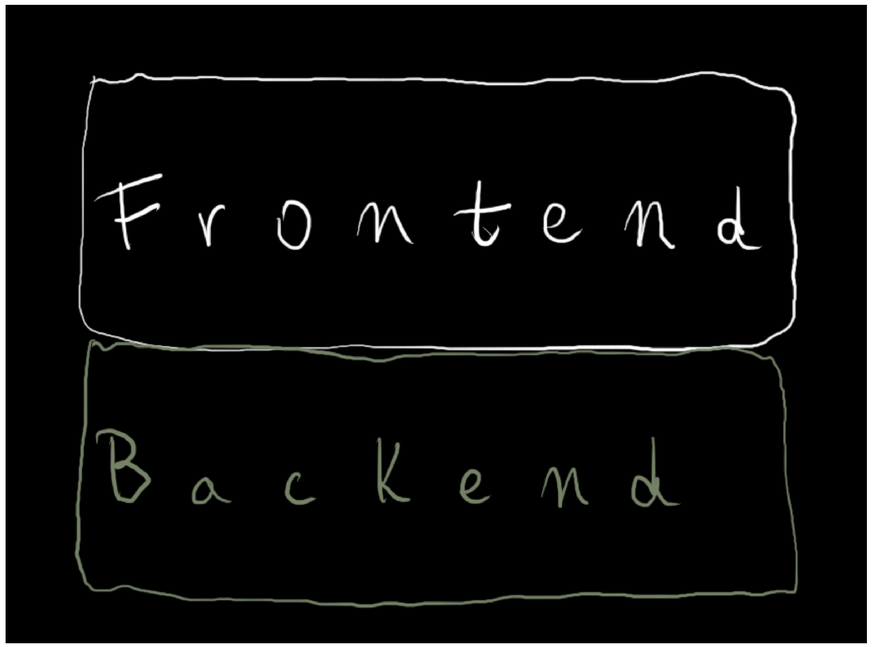
- **Prompt**: text input from user
- **Interface**: UI user interacts with to access the LLM

The LLM App will join both these concepts into one environment that is well suited to solve the task/ problem at hand.

LLM App

LLM app components

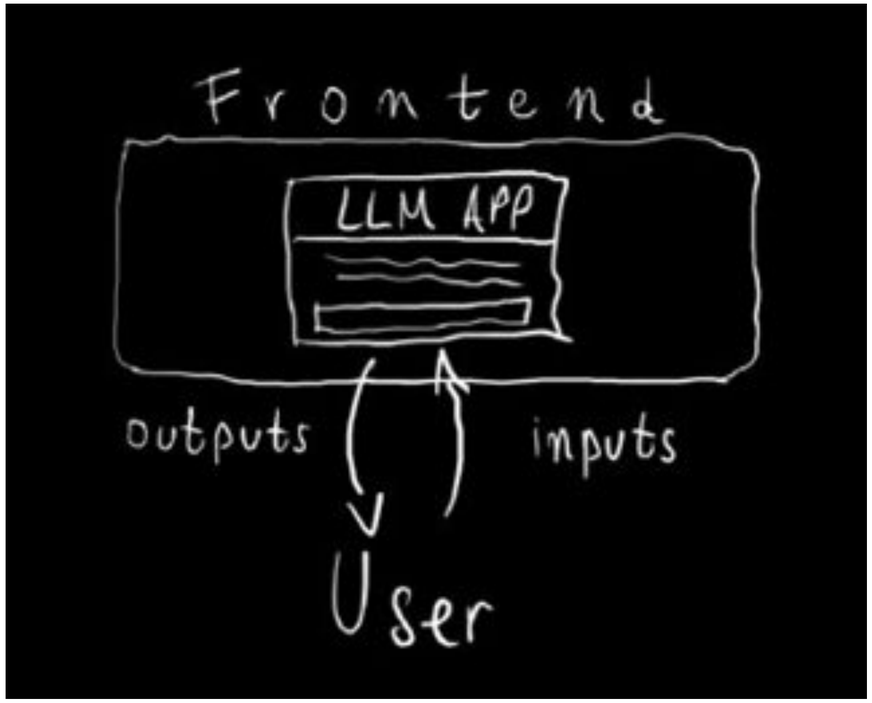
- Frontend → prompt/interface
- Backend → LLM model



LLM App

LLM app components

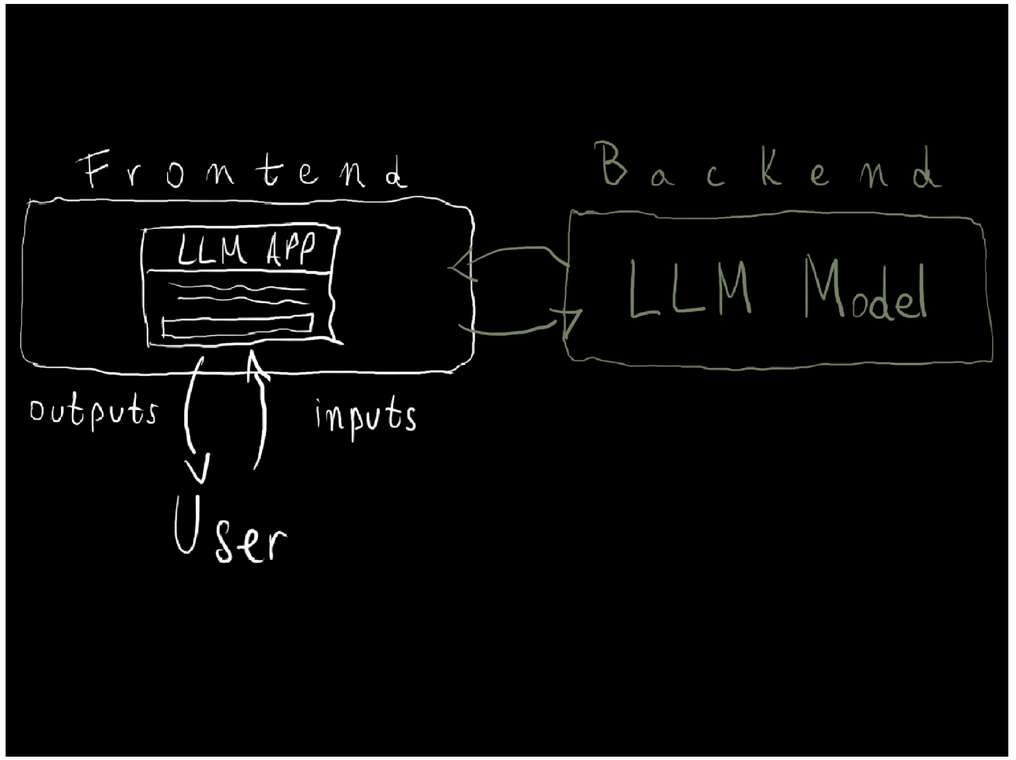
- **Frontend** → **prompt/interface**
- Backend → LLM model



LLM App

LLM app components

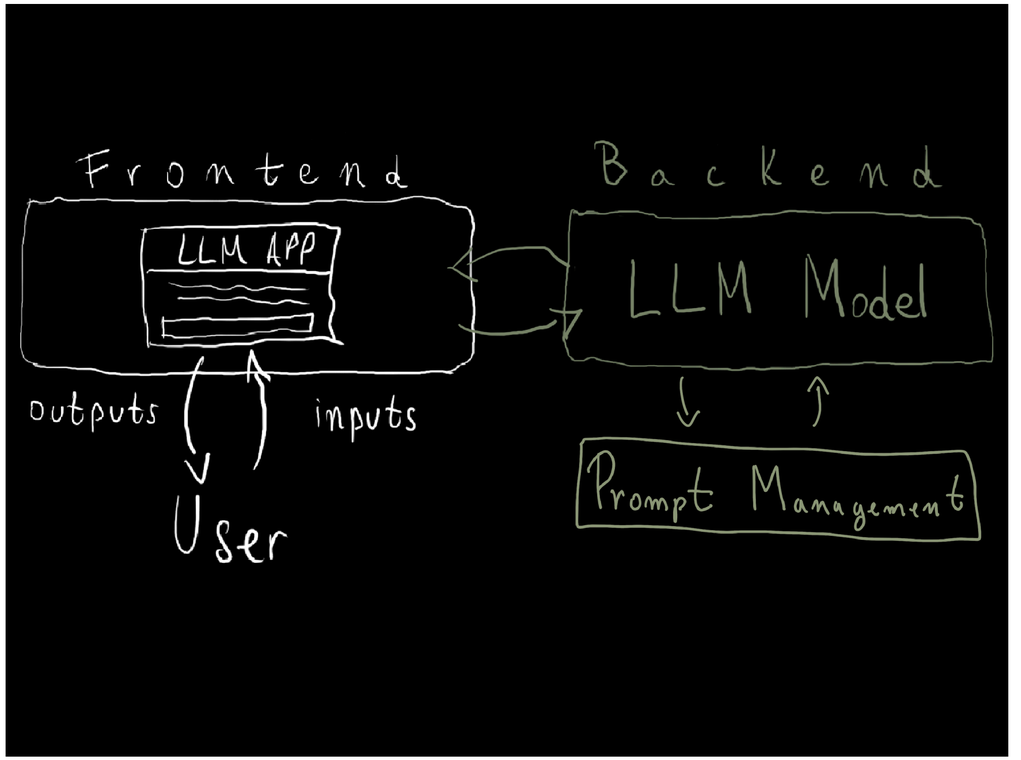
- Frontend → prompt/interface
- **Backend → LLM model**



LLM App

LLM app components

- Frontend → prompt/interface
- **Backend** → **LLM model** + **prompt management**



Different Levels of an LLM App

Level 1: Calling the API

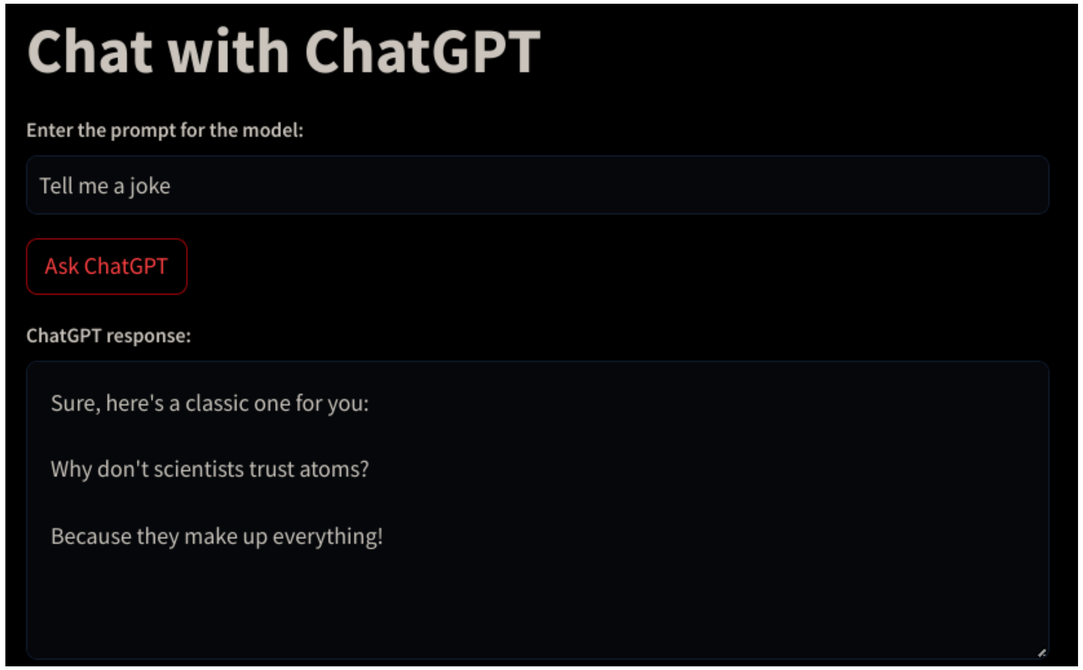
- **Level 1** - Calling the API
- Prototyping level

```python
import openai

def llm_model(prompt_question):
    response = openai.ChatCompletion.create(
        model="gpt-3.5-turbo",
        messages=[{"role": "system", "content": "You are a helpful research and\
            programming assistant"},
                  {"role": "user", "content": prompt_question}]
    )

    return response["choices"][0]["message"]["content"]

prompt = "Give me 5 exercises to practice calculus"
response = llm_model(prompt)
response
# Output
# '1. Differentiate the following functions:
# \n\na) f(x) = 3x^2 + 4x^3 - 2x\nb) g(x) = sin(x) + cos(x)\nc)
# h(x) = 2e^x - 4ln(x)\n\n2.
# ......
# ......
```

Different Levels of an LLM App

Level 2: LLM API call + UI

- **Level 2** - LLM API call + UI
- Giving the user specialized freedom

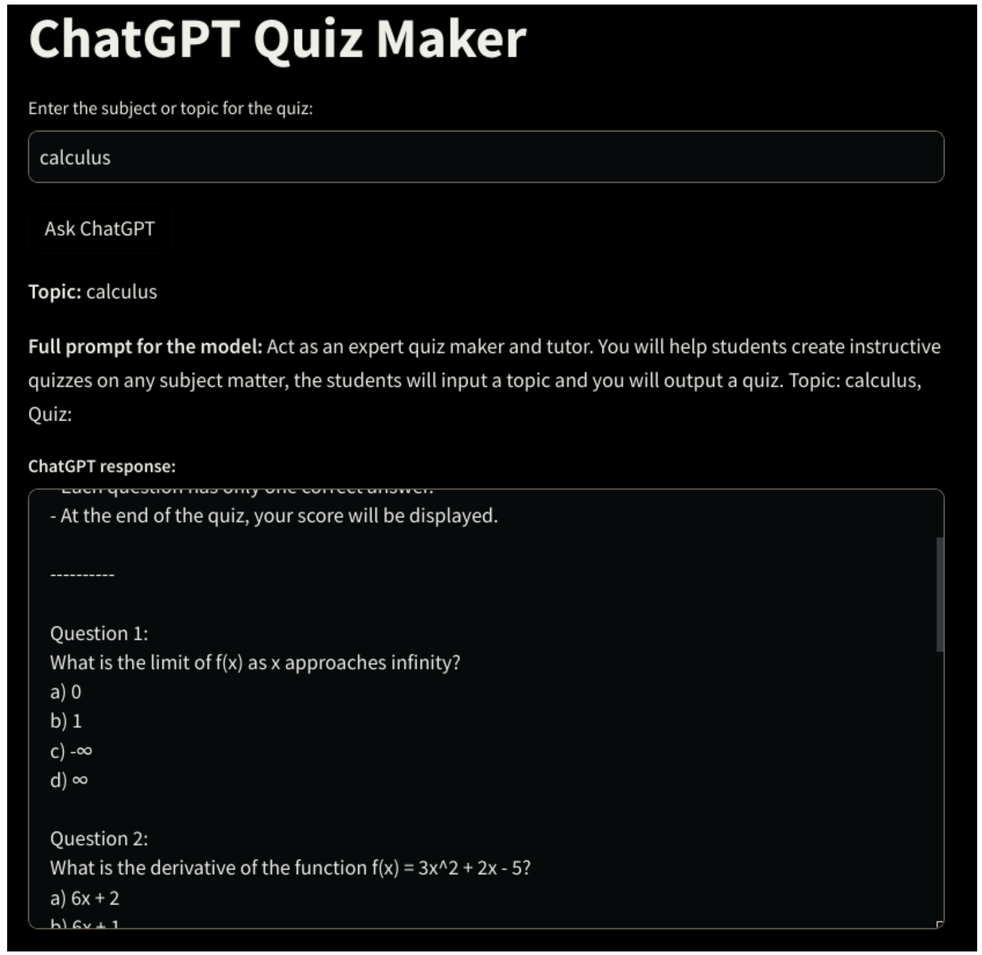


Different Levels of an LLM App

Level 3: Prompt Management

- **Level 3** - Prompt Management
- **Pre-prompting:** specifying desirable behaviors

"Act as an expert researcher and learning assistant and you will help students create instructive quizzes on any subject matter"

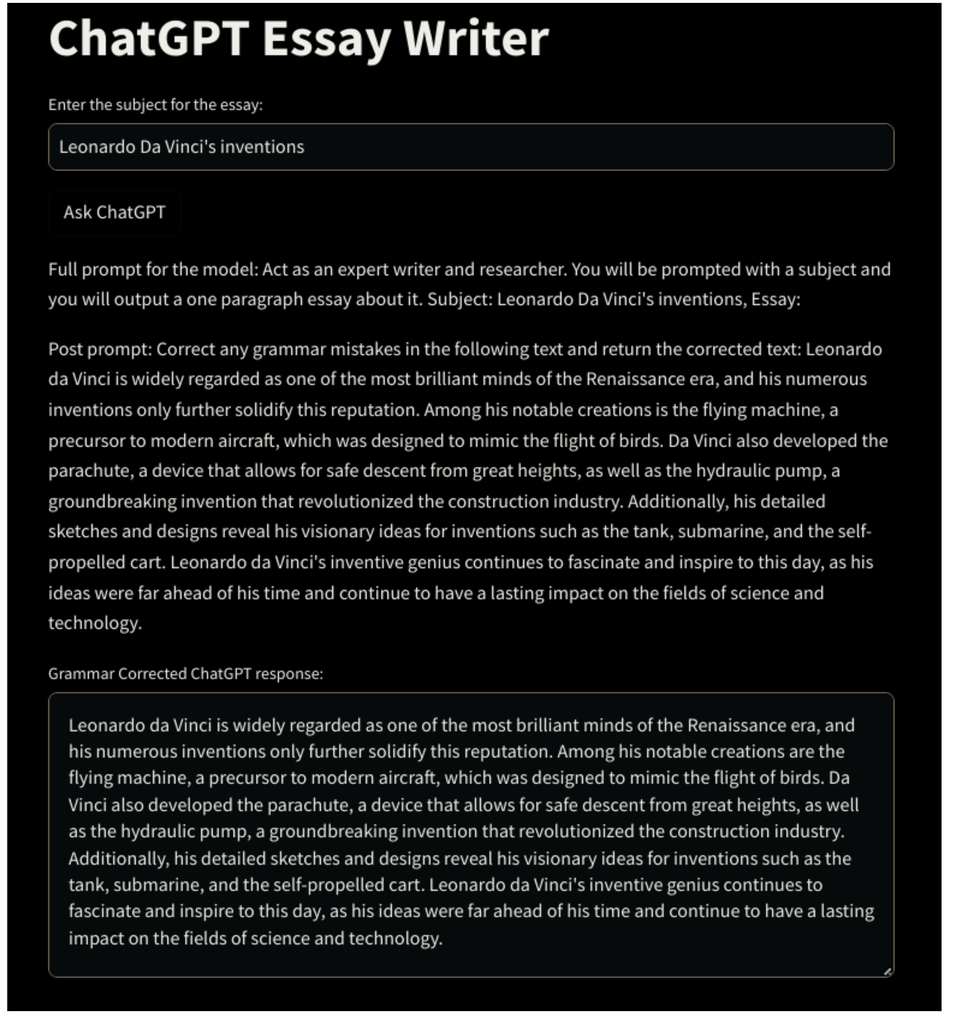


Different Levels of an LLM App

Level 3: Prompt Management

- **Level 3** - Prompt Management
- **Post-prompting:** aligning the model's response with the app's purpose

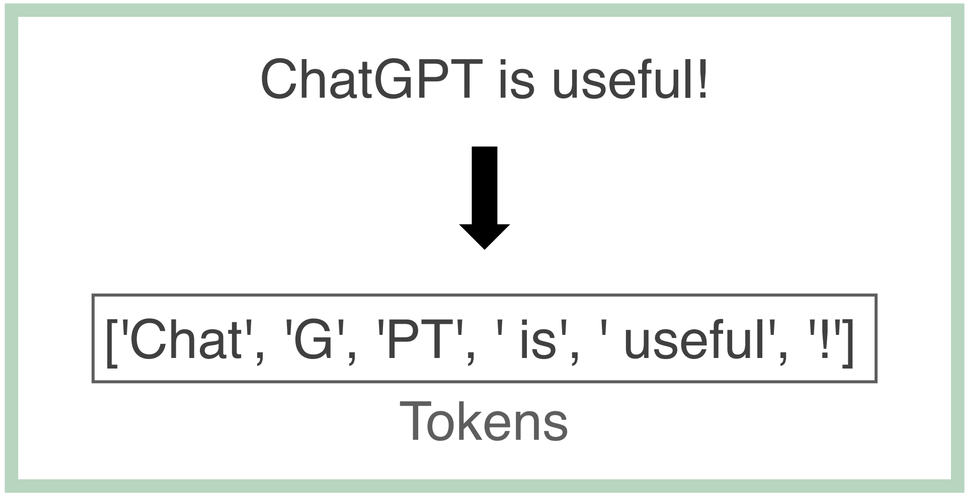
"Correct any grammar mistakes in the following text and return the corrected text"



Different Levels of an LLM App

Level 4: Accounting for costs

- **Level 4** - Accounting for costs
- Token management



Different Levels of an LLM App

Level 4: Accounting for costs

- **Cost:** Your API call cost is calculated per token.
- **Time:** The duration of your API call is influenced by the number of tokens as writing more tokens takes more time.
- **Functionality:** An API call can only function if the total tokens used are below the model's maximum limit (4096 for gpt-3.5-turbo).

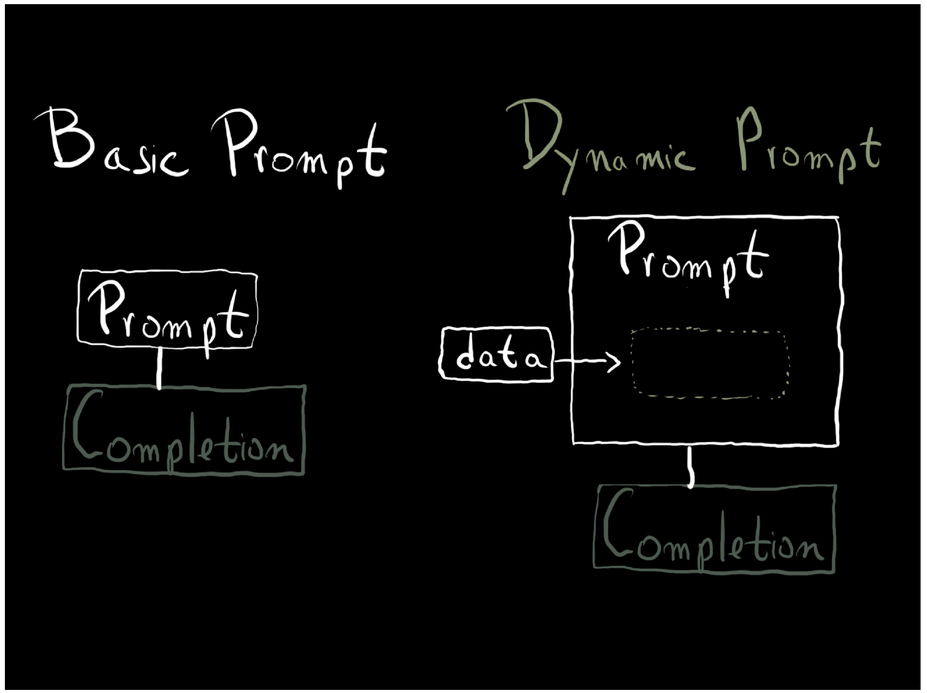
Both the input and output tokens count toward these quantities.

Refer to the usage field in the API response:
for example, response['usage']['total_tokens']

Langchain for LLM App Development

From static prompts to dynamic prompts

- Prompt management workflows that require dynamic prompts
- This dynamics requirement leads to the need for creating certain types of abstractions



Langchain for LLM App Development

Langchain framework

- Langchain is a framework that facilitates the creation and management of dynamic prompts and chaining between prompts.
- **Main features:** components and off-the-shelf-chains.

Langchain for LLM App Development

Langchain components

- **Models:** abstractions over the LLM APIs like the ChatGPT API.
- **Prompts**: “Prompt Templates” abstraction, output parsers.
- **Indexes**: Abstraction for interacting with documents (document loaders, text splitters, vector stores, retrievers).
- **Chains**: Combination of prompt, LLM and output parsing

Agents are outside the scope of this live-training

Langchain for LLM App Development

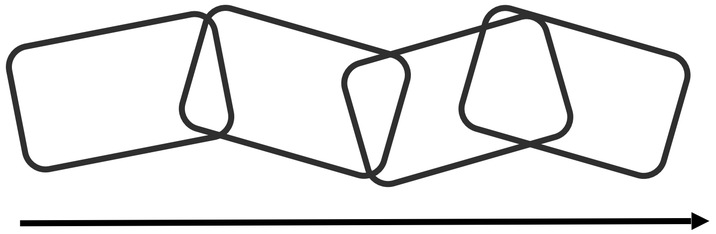
Models, Prompts and Output Parsers

Notebook Demo

Langchain for LLM App Development

LLMChain & Sequential Chains

LLMChain & Sequential Chains

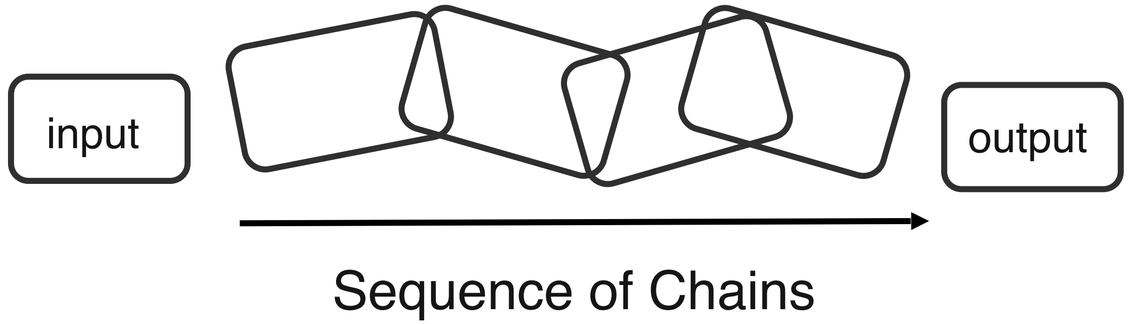


Chains of Prompts

Langchain for LLM App Development

LLMChain & Sequential Chains

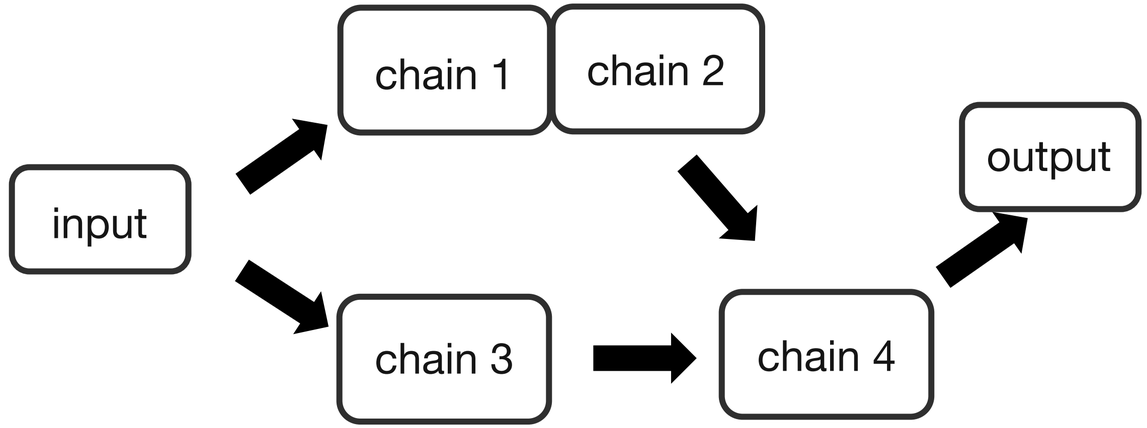
- LLMChain & Sequential Chains
- **SimpleSequentialChain: single input/output**



Langchain for LLM App Development

LLMChain & Sequential Chains

- LLMChain & Sequential Chains
- SimpleSequentialChain: single input/output
- **SequentialChain: multiple inputs/ outputs**

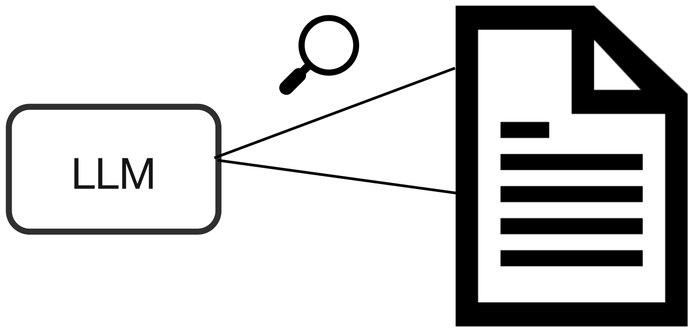


Notebook demo

Langchain for LLM App Development

Langchain with Documents

- LLMs have a limited context length



Langchain for LLM App Development

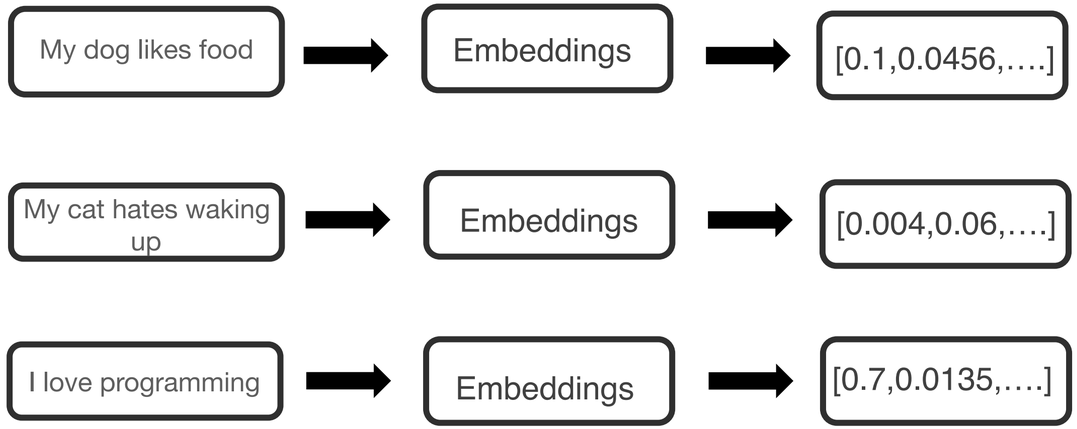
Langchain with Documents

- LLMs have a limited context length
- **Embeddings**: capture content and meaning

Langchain for LLM App Development

Langchain with Documents

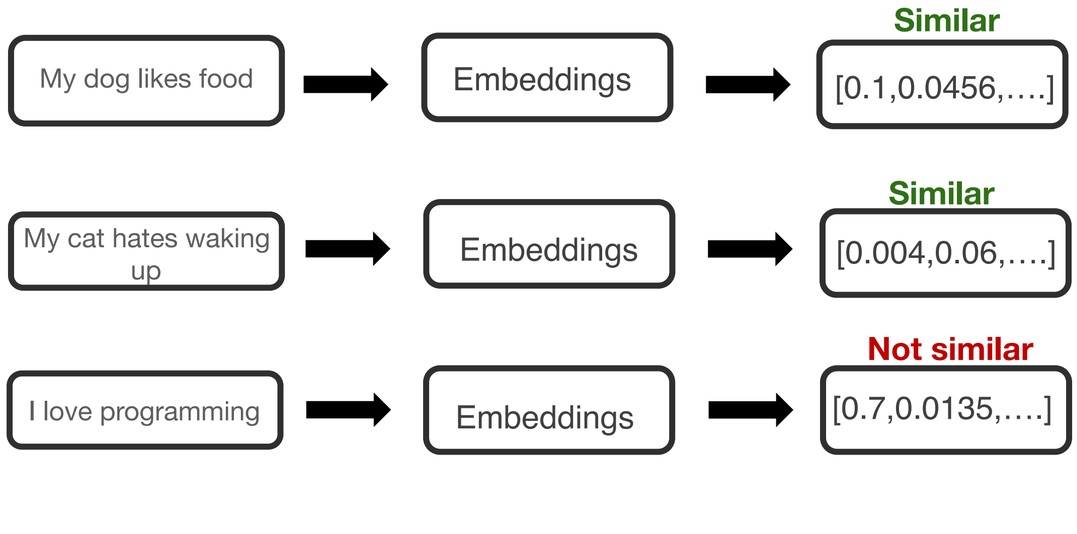
- LLMs have a limited context length
- **Embeddings**: capture content and meaning



Langchain for LLM App Development

Langchain with Documents

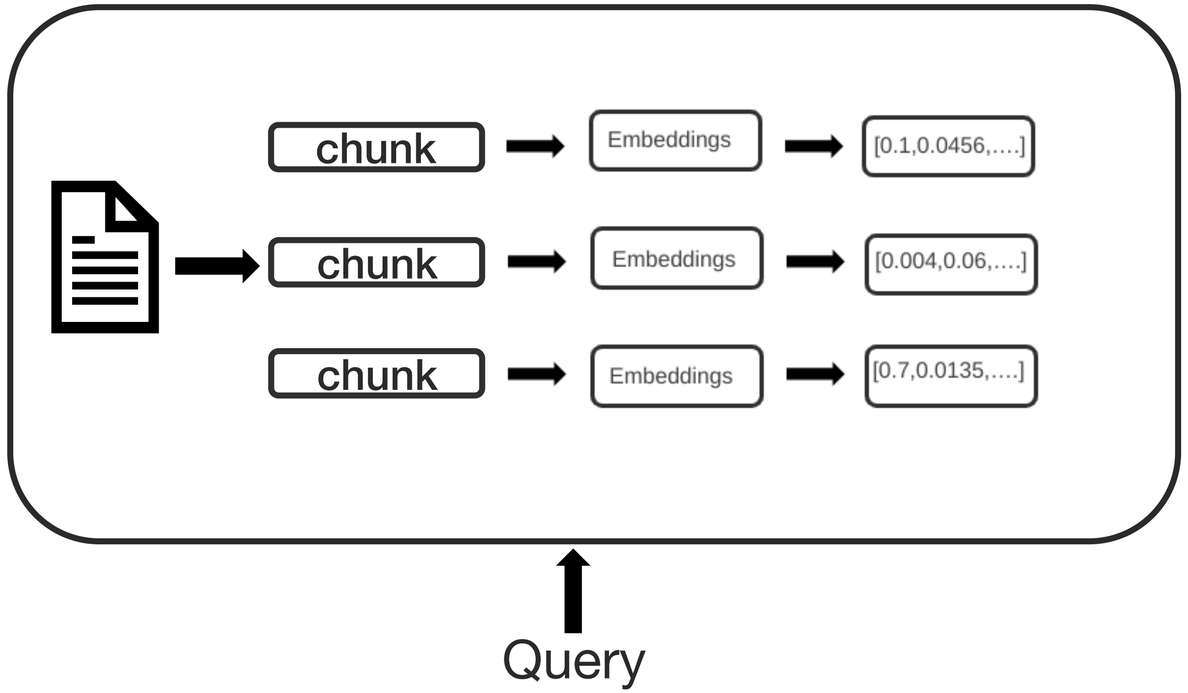
- LLMs have a limited context length
- **Embeddings**: capture content and meaning



Langchain for LLM App Development

Langchain with Documents

- LLMs have a limited context length
- Embeddings: capture content and meaning
- **Vector DBs**



Langchain for LLM App Development

Langchain with Documents

- LLMs have a limited context length
- Embeddings: capture content and meaning
- Vector DBs

Chunking methods for large docs

- Map_reduce
- Refine
- Map_rerank

Notebook demo

Langchain for LLM App Development

Evaluation with Langchain

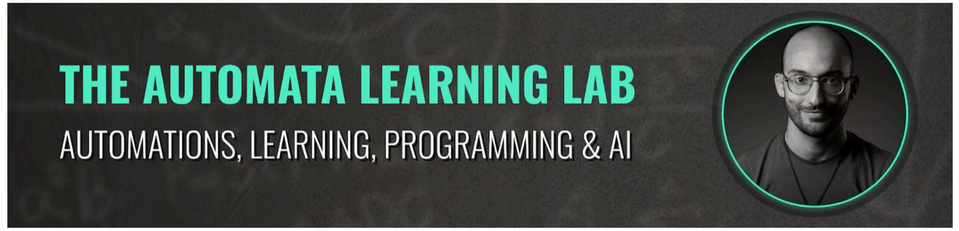
Notebook demo

Thank you for your attention!

Find me on:

YouTube

@automatalearninglab



Medium

@lucas-soares

Lucas Soares