Veerasak Kritsanapraphan

Prompt Engineering for Developers

Prompt Engineering

For developers

Veerasak Kritsanapraphan, Software Park Thailand

About myself?

- Graduated from San Francisco State University in Master of Science in Computer Information System, 1997
- PhD. at Chulalongkorn University, research focus on Data Science, Big Data, Mobile Computing and Internet of Thing (IOT)
- Senior Director at Arise by Infinitas (KTB)
- Data Science/AI Independent Consultant

Introduce yourself?

- What is your name/position/role in your organization?
- Tell you a bit about yourself?
- What is your passion?
- What is your expectation for this class?

Massive ChatGPT adoption

- ChatGPT took the Internet by storm
- Just 5 days to 1 million users
- How innovative is it as a technological breakthrough?
- How does OpenAI compare to others?

HOW LONG IT TOOK TOP APPS TO HIT 100M MONTHLY USERS
ChatGPT is estimated to have hit 100M users in January, 2 months after its launch.
Here's how long it took other top apps to reach that:
APP
MONTHS TO REACH 100M GLOBAL MAUS
CHATGPT
2
TIKTOK
9
INSTAGRAM
30
PINTEREST
41
SPOTIFY
55
TELEGRAM
61
UBER
70
GOOGLE TRANSLATE
78
SOURCE: UBS
yahoo! finance

-$200 million revenue in 2023
-Microsoft invested in OpenAI ~$10 billion
-Valuation at ~$30 billion

What is Generative AI, and why is it suddenly everywhere?

Generative AI is a branch of computer science that involves unsupervised and semi-supervised algorithms that enable computers to ***create new content*** *using previously* ***created content, such as text, audio, video, images and code*** *in response to short prompts.*

ChatGPT *(which stands for Chat Generative Pre-Trained Transformer) is a chatbot developed by OpenAI. ChatGPT is built on top of* ***OpenAI's GPT-3.5*** *family of* ***large language models*** *(LLMs) and is fine-tuned with both supervised and reinforcement learning techniques.*

Hype Cycle for Artificial Intelligence, 2023

Plateau will be reached:

○ less than 2 years ● 2 to 5 years ● 5 to 10 years ▲ more than 10 years ⊗ obsolete before plateau

As of July 2023

gartner.com

Gartner

Source: Gartner
© 2023 Gartner, Inc. and/or its affiliates. All rights reserved. 2079794

Gartner Generative AI Framework

Desired Content Type	Non-Sensitive Text	PII/Enterprise IP Included	Enterprise Data and Model Instructions Needed	Model Fine Tuning to Improve Use/ Performance	Custom Model Is Created for Unique Use Case
Enterprise Control Level	No Controls Needed	LLM With Privacy	LLM With Privacy, Policies and Data	LLM w/Privacy, Policies, Data and Added Model Layers	Enterprise Hosted Custom Differentiated Model
Solution Needed	ChatGPT	LLM API Accessed via Application Frame	LLM API With Standard Privacy, Policy and Per Incident Data Injection	Modified LLM With Transfer Learning/Added Layers to Adjust Model Output	Custom Build LLM Using Enterprise Accessible Data
Required/ Recommended Technologies	Open AI Hosted Application	Cloud Instance With LLM APIs	Cloud Instance With LLM APIs, Prompt Engineering, Custom Policies, Indexed Database	Licensed Customizable Model/ Proprietary Model, Data, ML Platform	Custom Build Proprietary Model, Data, ML Platform
Cost	Negligible	$	$$$$	$$$$$$$	$$$$$$$$$$$$

Source: Gartner

© 2023 Gartner, Inc. and/or its affiliates. All rights reserved. 2514783

Gartner

Q&A
Answer questions based on existing knowle...

Grammar correction
Corrects sentences into standard English.

Summarize for a 2nd grader
Translates difficult text into simpler concep...

Natural language to OpenAI API
Create code to call to the OpenAI API usin...

Text to command
Translate text into programmatic commands.

English to other languages
Translates English text into French, Spanish...

Natural language to Stripe API
Create code to call the Stripe API using nat...

SQL translate
Translate natural language to SQL queries.

Parse unstructured data
Create tables from long form text

Classification
Classify items into categories via example.

Python to natural language
Explain a piece of Python code in human un...

Movie to Emoji
Convert movie titles into emoji.

Calculate Time Complexity
Find the time complexity of a function.

Translate programming languages
Translate from one programming language ...

Advanced tweet classifier
Advanced sentiment detection for a piece o...

Explain code
Explain a complicated piece of code.

Keywords
Extract keywords from a block of text.

Factual answering
Guide the model towards factual answering ...

Ad from product description
Turn a product description into ad copy.

Product name generator
Create product names from examples word...

TL;DR summarization
Summarize text by adding a 'tl;dr:' to the en...

Python bug fixer
Find and fix bugs in source code.

Spreadsheet creator
Create spreadsheets of various kinds of dat...

JavaScript helper chatbot
Message-style bot that answers JavaScript ...

ML/AI language model tutor
Bot that answers questions about language...

Science fiction book list maker
Create a list of items for a given topic.

Tweet classifier
Basic sentiment detection for a piece of text.

Airport code extractor
Extract airport codes from text.

SQL request
Create simple SQL queries.

Extract contact information
Extract contact information from a block of ...

JavaScript to Python
Convert simple JavaScript expressions into ...

Friend chat
Emulate a text message conversation.

Mood to color
Turn a text description into a color.

Write a Python docstring
An example of how to create a docstring for ...

Analogy maker
Create analogies. Modified from a communi...

JavaScript one line function
Turn a JavaScript function into a one liner.

Micro horror story creator
Creates two to three sentence short horror ...

Third-person converter
Converts first-person POV to the third-pers...

Notes to summary
Turn meeting notes into a summary.

VR fitness idea generator
Create ideas for fitness and virtual reality g...

ESRB rating
Categorize text based upon ESRB ratings.

Essay outline
Generate an outline for a research topic.

Recipe creator (eat at your own risk)
Create a recipe from a list of ingredients.

Chat
Open ended conversation with an AI assist...

Marv the sarcastic chat bot
Marv is a factual chatbot that is also sarcas...

Turn by turn directions
Convert natural language to turn-by-turn dir...

Restaurant review creator
Turn a few words into a restaurant review.

Create study notes
Provide a topic and get study notes.

Interview questions

ChatGPT Plugins

- Plugins add extra functionality
- Possible to call external APIs for different tasks
- Wolfram Alpha, Internet Browsing, Python Interpreter, Knowledge Retrieval, Shopping, etc.

Shopify's Shop app

https://openai.com/blog/introducing-chatgpt-and-whisper-apis (March 1, 2023)
https://openai.com/blog/chatgpt-plugins (March 23, 2023)

ChatGPT limitations and challenges

- Knowledge up to 2021. Announcement recently is to 2023.
- Issues of factual correctness, bias, and toxicity
- Hallucinations instead of facts
- Confident, but making things up
- Legal issues & copyright
- Ethical and educational challenges

Demo #1

ChatGPT Basics

Creating an account

1) Go to chat.OpenAi.com and register for an account with an email address, or a Google or Microsoft account. You need to create an account on the OpenAI website to log in and access ChatGPT, but it is free.

Creating an account

2) Accept ChatGPT's terms

ChatGPT vs ChatGPT Plus

Basic UI of ChatGPT

1. **New chat & hide sidebar buttons**: On the left-hand side of your screen, you'll see a button for New chat that you can click on to start a new conversation at any time. This is useful because the ChatGPT model remembers what was previously discussed in a conversation and will respond with context, so starting a new chat will start a new conversation without prejudice or context.
2. **Chat history**: The left sidebar also keeps all your previous conversations accessible, in case you need to go back to one, though there is the option for turning off chat history. Here you can edit the title of each chat, share your chat history with others, or delete it.
3. **Account**: Clicking on your e-mail address or name on the bottom left of your screen will give you access to your account information, including settings, the option to log out, and help and FAQ from OpenAI. If you don't have ChatGPT Plus, you'll also see an Upgrade to Plus button here.
4. **Your prompts**: The questions or prompts you give the AI chatbot appear here.
5. **ChatGPT's responses**: Whenever ChatGPT responds to your queries, the logo will appear on the left. The Copy, Thumbs up, and Thumbs down buttons appear to the right of each response. You can copy the text to your clipboard to paste elsewhere and also provide feedback on whether the response was accurate or not. This process helps fine-tune the AI tool
6. **Regenerate response**: If you have issues getting a response in the chat or don't get a satisfactory answer, you can click on Regenerate response to prompt ChatGPT to try again with a new answer for the latest prompt.
7. **Text area**: This is where you enter your prompts and questions; just hit enter to send them.
8. **ChatGPT version**: OpenAI includes a bit of fine print below the text input area, where you can read the disclaimer that this is a "Free Research Preview. ChatGPT may produce inaccurate information about people, places, or facts". There is also the version of the ChatGPT model you're currently using.

Workshop # 1

Getting to know ChatGPT

Instruction

1. Follow the demo to register and login to ChatGPT
2. Familiar with UI
3. Write simple prompts as follows:-
    1. How does a computer store and process information?
    2. Analyze this code and tell me how to fix it

```
public class Test {
    private static void printLength(String str) {
        System.out.println(str.length());
    }
    public static void main(String []args){
        String myString = null;
        printLength(myString);
    }
}
```

    3. Write a sick note for my child who is missing school.

Key Concept

Text Generation Models

- OpenAI's GPT models (like GPT-4 and GPT-3.5) understand natural and formal language.
- These models generate text in response to prompts, which are the inputs provided by users.
- Designing a prompt is how you "program" the model to complete tasks.
- GPT models are versatile, used for content and code generation, summarization, conversation, creative writing, etc.

Assistants

- OpenAI API uses assistants powered by models like GPT-4 for task execution.
- Assistants follow instructions within the model's context window.
- They have access to tools for complex tasks like code execution or file information retrieval.
- Additional details are in the Assistants API Overview.

Embeddings

- Embeddings are vector representations of data (like text) that capture content or meaning.
- Similar data results in closer embeddings, which are useful for several applications including search and clustering.
- OpenAI provides models that convert text strings to embedding vectors.

Foundation of Machine Learning

Traditional Programming vs Machine Learning

Software 1.0

Software 2.0

Types of Machine Learning

Unsupervised Learning

Learn structure of data to generate more data

Supervised Learning

Learn how data maps to labels to recognize or predict

Reinforcement Learning

Learn how to act in an environment to obtain reward

"This product does what it is supposed __"

$\rightarrow$ *cat*

$\rightarrow$ *"Hey Siri"*

Converging on just...

Supervised or Self-supervised Learning

"This product does what it is supposed __" $\rightarrow$ *"to."*

$\rightarrow$ *cat*

$\rightarrow$ *"Hey Siri"*

$\rightarrow$ *next move*

Inputs and outputs are always just numbers

Input

Output

What we see

"Lincoln"

What the machine "sees"

157	153	174	168	150	152	129	151	172	161	155
155	182	163	74	75	62	33	17	110	210	180
180	180	50	14	34	6	10	33	48	106	159
206	109	5	124	131	111	120	204	166	15	56
194	68	137	251	237	239	239	228	227	87	71
172	105	207	233	233	214	220	239	228	98	74
188	88	179	209	185	215	211	158	139	75	20
189	97	165	84	10	168	134	11	31	62	22
199	168	191	193	158	227	178	143	182	106	36
205	174	155	252	236	231	149	178	228	43	95
190	216	116	149	236	187	86	150	79	38	218
190	224	147	108	227	210	127	102	36	101	255

[76, 105, 110, 99, 111, 108, 110]

Why is this hard?

- Infinite variety of inputs can all mean the same thing
- Meaningful differences can be tiny
- Structure of the world is complex

"I loved this movie"

"As good as The Godfather"

"🔥 no cap"

How is it done?

- Many methods for Machine Learning
  - Logistic Regression
  - Support Vector Machines
  - Decision Trees (xgboost)
- But one is dominant
  - Neural Networks (also called Deep Learning)

Inspiration

https://www.the-scientist.com/the-nutshell/what-made-human-brains-so-big-36663

https://medicalxpress.com/news/2018-07-neuron-axons-spindly-theyre-optimizing.html

- Inspired by what we know to be intelligent: the **brain**
- The brain is composed of billions of **neurons**
- Each neuron receives electrical **inputs** and sends an electrical **output**
- The brain itself has high-level inputs and outputs

Formalization

https://www.the-scientist.com/the-nutshell/what-made-human-brains-so-big-36663

https://medicalxpress.com/news/2018-07-neuron-axons-spindly-theyre-optimizing.html

https://www.jessicayung.com/explaining-tensorflow-code-for-a-multilayer-perceptron/

A "perceptron" is a vector of numbers

```
[0.58  0.841 0.835 0.18 ]
```

https://www.jessicayung.com/explaining-tensorflow-code-for-a-multilayer-perceptron/

A "layer" is a matrix of numbers

https://www.jessicayung.com/explaining-tensorflow-code-for-a-multilayer-perceptron/

```
[[0.58  0.841 0.835 0.18 ]
 [0.405 0.813 0.309 0.562]
 [0.422 0.229 0.46  0.152]
 [0.673 0.429 0.441 0.243]
 [0.9   0.744 0.234 0.856]
 [0.971 0.486 0.175 0.248]
 [0.258 0.588 0.478 0.266]
 [0.236 0.496 0.077 0.557]
 [0.413 0.322 0.372 0.741]]
```

The neural network is a set of matrices

Called "parameters" or "weights"

NN operations are just matrix multiplications.
GPUs are really fast at matrix multiplications.

Training

$$loss_{CE} = -\sum(y_i * \log(y'_i))$$

- Data X (e.g. images), labels y (e.g. labels)
- Take a little batch of data x:
  - Use the current model to make a prediction x -> y'
  - Compute loss(y, y')
  - Back-propagate the loss through all the layers of the model
- Repeat until loss stops decreasing

https://www.guru99.com/backpropogation-neural-network.htm

https://developers.google.com/machine-learning/testing-debugging/metrics/interpretic

Dataset Splitting

- Split (X, y) into training (~80%), validation (~10%), and test (~10%) sets
- Validation set is for
  - ensuring that training is not "overfitting"
  - setting hyper-parameters of the model (e.g. number of parameters)
- Test set is for measuring validity of predictions on new data

THIS APPLIES TO YOUR EXPERIMENTATION WITH PROMPTS!

Pre-training:
slow training on a lot of data

Fine-tuning:
fast training on a little data

Model Hubs

- People share pre-trained models!
- 🤗: the most popular Model Hub
  - 180K models
  - 30K datasets

Before ~2020: each task had its own NN architecture

Computer Vision

Convolutional NNs (+ResNets)

Natural Lang. Proc.

Recurrent NNs (+LSTMs)

Speech

Deep Belief Nets (+non-DL)

Translation

Seq2Seq

RL

BC/GAIL

Algorithm 1 Generative adversarial imitation learning

1: **Input:** Expert trajectories $\tau_E \sim \pi_E$, initial policy and discriminator parameters θ_0, w_0
2: **for** $i = 0, 1, 2, \ldots$ **do**
3: Sample trajectories $\tau_i \sim \pi_{\theta_i}$
4: Update the discriminator parameters from w_i to w_{i+1} with the gradient

$$\hat{\mathbb{E}}_{\tau_i}[\nabla_w \log(D_w(s,a))] + \hat{\mathbb{E}}_{\tau_E}[\nabla_w \log(1 - D_w(s,a))] \quad (17)$$

5: Take a policy step from θ_i to θ_{i+1}, using the TRPO rule with cost function $\log(D_{w_{i+1}}(s,a))$. Specifically, take a KL-constrained natural gradient step with

$$\hat{\mathbb{E}}_{\tau_i}[\nabla_\theta \log \pi_\theta(a|s) Q(s,a)] - \lambda \nabla_\theta H(\pi_\theta),$$
$$\text{where } Q(\bar{s},\bar{a}) = \hat{\mathbb{E}}_{\tau_i}[\log(D_{w_{i+1}}(s,a)) \mid s_0 = \bar{s}, a_0 = \bar{a}] \quad (18)$$

6: **end for**

[1] CNN image CC-BY-SA by Aphex34 for Wikipedia https://commons.wikimedia.org/wiki/File:Typical_cnn.png
[2] RNN image CC-BY-SA by GChe for Wikipedia https://commons.wikimedia.org/wiki/File:The_LSTM_Cell.svg

http://lucasb.eyer.be/transformer

Transformer Architecture

Attention is all you need (2017)

- Ground-breaking architecture that set SOTA on first translation and later all other NLP tasks
- For simplicity, can just look at one half of it

Transformer Decoder Overview

- Task is to complete text
  - "It's a blue" -> "sundress"
- Inputs: a sequence of N tokens
  - [It's, a, blue]
- Output:
  - Probability distribution over the next token
- Inference:
  - Sample the next token from the distribution, append it to inputs, run through the model again, sample, append, etc.

Inputs

- Inputs need to be vectors of numbers
- Start with original text:
  - "It's a blue sundress."
- Turn into a sequence of tokens:
  - [<SOS>, It, 's, a, blue, sund, ress, ., <EOS>]
- Turn into vocabulary IDs:
  - [0, 1026, 338, 257, 4171, 37437, 601, 13, 1]
- Each ID can be represented by a one-hot vector
  - e.g. 3 -> [0, 0, 0, 1, 0, 0, 0, 0, ...]

Input Embedding

- One-hot vectors are poor representations of words or tokens
  - e.g. distance between "cat" and "kitten" is the same as
    between "cat" and "tractor"
- Solution: learn an embedding matrix!
  - (The simplest NN layer type)

aardvark	1	0	0	...	0	0	0
black	0	0	...	1	...	0	0
cat	0	0	...	1	...	0	0
duvet	0	0	...	1	...	0	0
zombie	0	0	0	...	0	0	1

VxV

VxE
embedding matrix

aardvark	0.97	0.03	0.15	0.04
black	0.07	0.01	0.20	0.95
cat	0.98	0.98	0.45	0.35
duvet	0.01	0.84	0.12	0.02
zombie	0.74	0.05	0.98	0.93

VxE

47

Attention

- (Ignore "Masked Multi-Head" for now)
- Key insight: for a given token in the output sequence, only one or a few tokens in the input sequence are most important
- Introduced in 2015 for translation tasks

https://lilianweng.github.io/posts/2018-06-24-attention/

Basic self-attention

- Input: sequence of vectors

$$\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_t$$

- Output: sequence of vectors, each one a weighted sum of the input sequence

$$\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_t$$

$$\mathbf{y}_i = \sum_j w_{ij} \mathbf{x}_j$$

- weight is just dot product between input vectors $\quad w'_{ij} = \mathbf{x}_i{}^T \mathbf{x}_j$

- (made to sum to 1) $\quad w_{ij} = \dfrac{\exp w'_{ij}}{\sum_j \exp w'_{ij}}$

http://www.peterbloem.nl/blog/transformers

Basic self-attention

$$\mathbf{y}_i = \sum_j w_{ij} \mathbf{x}_j$$

- Note that every input vector x_i is used in 3 ways:
  - Query x_i
  - Key $\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_t$
  - Value $\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_t$

$$w'_{ij} = \mathbf{x}_i^T \mathbf{x}_j$$

http://www.peterbloem.nl/blog/transformers

Basic self-attention

- Problem: there's no learning involved!
- Solution: project inputs into query, key, value roles
- Learning these matrices = learning attention

$$\mathbf{q}_i = \mathbf{W}_q \mathbf{x}_i \qquad \mathbf{k}_i = \mathbf{W}_k \mathbf{x}_i \qquad \mathbf{v}_i = \mathbf{W}_v \mathbf{x}_i$$

$$w'_{ij} = \mathbf{q}_i^{\mathsf{T}} \mathbf{k}_j$$

$$w_{ij} = \mathrm{softmax}(w'_{ij})$$

$$\mathbf{y}_i = \sum_j w_{ij} \mathbf{v}_j \, .$$

http://lucasb.eyer.be/transformer

http://www.peterbloem.nl/blog/transformers

Multi-head attention

- We can allow different ways of transforming into queries, keys, and values to be learned
- Simply means learning different sets of W_q, W_k, and W_v matrices simultaneously.
  - (Actually implemented as a single matrix, anyway.)

3-headed attention

http://www.peterbloem.nl/blog/transformers

Masking attention

In training:

- Inputs: a sequence of N tokens
  - [It's, a, blue, <BLANK>, <BLANK>, .
- Ground-truth Outputs: a sequence of N tokens
  - [a, blue, sundress, <BLANK>, <BLANK>, ...]
- Actual Outputs: Instead of words, vectors of probabilities over the vocabulary
  - **Crucially: all output probabilities are computed at the same time!**

https://jalammar.github.io/illustrated-

http://www.peterbloem.nl/blog/transformers

Note how you shouldn't see future tokens when predicting

Masked Multi-Head Attention

- Conceptual view:
  - token comes in
  - gets "augmented" with previously-seen tokens that seem relevant (masked self-attention)
  - this happens in several different ways simultaneously (multiple heads)
- NOTE: there's no notion of "position" so far!

Positional Encoding

$$\mathbf{q}_i = \mathbf{W}_q \mathbf{x}_i \qquad \mathbf{k}_i = \mathbf{W}_k \mathbf{x}_i \qquad \mathbf{v}_i = \mathbf{W}_v \mathbf{x}_i$$

$$w'_{ij} = \mathbf{q}_i^{\mathsf{T}} \mathbf{k}_j$$

$$w_{ij} = \mathrm{softmax}(w'_{ij})$$

$$\mathbf{y}_i = \sum_j w_{ij} \mathbf{v}_j \, .$$

- Attention is totally position-invariant!
  - e.g. [this, movie, is, great] is the same as [movie, this, great, is]
- So, let's add position-encoding vectors to embedding vectors
  - It really is that simple

Add

- "Skip connections" aka "residual blocks"
  - output = module(input) + input
  - Allows gradient to flow from the loss function all the way to the first layer
  - (Possible because each module's output is the same shape as its input)

Layer Normalization

- Neural net modules perform best when input vectors have uniform mean and std in each dimension.
- As inputs flow through the network, means and std's get blown out.
- Layer Normalization is a hack to reset things to where we want them in between layers.

output = module(layer_norm(input)) + input

$$y = \frac{x - \mathrm{E}[x]}{\sqrt{\mathrm{Var}[x] + \epsilon}}$$

https://arxiv.org/pdf/1803.08494.pdf

Feed Forward Layer

- Standard Multi-Layer Perceptron with one hidden layer
- Defined $y = W_2(\mathrm{GeLU}\,(W_1 x + b_1)) + b_2$
- Conceptual view:
  - token (augmented with other relevant tokens that it has seen) comes in...
  - ...and "upgrades" its representation

Andrej Karpathy
@karpathy

The Transformer is a magnificient neural network architecture because it is a general-purpose differentiable computer. It is simultaneously:
1) expressive (in the forward pass)
2) optimizable (via backpropagation+gradient descent)
3) efficient (high parallelism compute graph)

11:54 AM · Oct 19, 2022

490 Retweets **39** Quotes **3,670** Likes **1,023** Bookmarks

What are Large Language Models?

Large, general-purpose language models can be pre-trained and then fine-tuned for specific purposes

Large language models are trained to solve common language problems, like …..

Trained with a relatively small size of field datasets

Characteristics of LLM

- Large
  - Large training dataset
  - Large number of parameters
- General purpose
  - Commonality of human languages
  - Resource restriction
- Pre-trained and fine-tuned

How LLM works?

Interact with LLM using Prompts

Fine-tune it

Healthcare data
Millions of EHRs
Reusable components
Medical Foundation Model
Tasks adoption
Question answering
Chart summarizing
Image analysis/labelling
Risk stratification
Finding similar patients
Human-AI collaboration
Natural language interactions

Three Easy Pieces

BERT

[*] [*] **[sat_]** [*] **[the_]** [*]

[The_] [cat_] **[MASK]** [on_] **[MASK]** [mat_]

T5

Das ist gut.

A storm in Attala caused 6 victims.

This is not toxic.

Translate EN-DE: This is good.

Summarize: state authorities dispatched…

Is this toxic: You look beautiful today!

GPT

[sat_]

[START] [The_] [cat_]

Transformer image source: "Attention Is All You Need" paper

From Lukas Beyer's lecture Intro to Transformers.

BERT (2019)

- *Bidirectional* Encoder Representations from Transformers
- Encoder-only (no attention masking)
- 110M params
- 15% of all words masked out
- Was great, now dated

From Lukas Beyer's lecture Intro to Transformers.

T5: Text-to-Text Transfer Transformer (2020)

- Input and output are both text strings
- Encoder-Decoder architecture
- 11B parameters
- Still could be a good choice for fine-tuning!

Das ist gut.

A storm in Attala caused 6 victims.

This is not toxic.

Translate EN-DE: This is good.

Summarize: state authorities dispatched…

Is this toxic: You look beautiful today!

T5 Training Data

- Unsupervised pre-training on Colossal Clean Crawled Corpus (C4)
  - Start with Common Crawl (over 50TB of compressed data, 10B+ web pages)
  - Filtered down to ~800GB, or ~160B tokens
- Also trained on academic supervised tasks

- We discarded any page with fewer than 5 sentences and only retained lines that contained at least 3 words.
- We removed any page that contained any word on the "List of Dirty, Naughty, Obscene or Otherwise Bad Words".[6]
- Some pages inadvertently contained code. Since the curly bracket "{" appears in many programming languages (such as Javascript, widely used on the web) but not in natural text, we removed any pages that contained a curly bracket.
- To deduplicate the data set, we discarded all but one of any three-sentence span occurring more than once in the data set.

https://paperswithcode.com/dataset/c4

2. **Datasets used for Supervised text-to-text language modeling objective**

- Sentence acceptability judgment
  - CoLA Warstadt et al., 2018
- Sentiment analysis
  - SST-2 Socher et al., 2013
- Paraphrasing/sentence similarity
  - MRPC Dolan and Brockett, 2005
  - STS-B Ceret al., 2017
  - QQP Iyer et al., 2017
- Natural language inference
  - MNLI Williams et al., 2017
  - QNLI Rajpurkar et al.,2016
  - RTE Dagan et al., 2005
  - CB De Marneff et al., 2019
- Sentence completion
  - COPA Roemmele et al., 2011
- Word sense disambiguation
  - WIC Pilehvar and Camacho-Collados, 2018
- Question answering
  - MultiRC Khashabi et al., 2018
  - ReCoRD Zhang et al., 2018
  - BoolQ Clark et al., 2019

https://stanford-cs324.github.io/winter2022/lectures/data/

Here come the

GPTs

What are Generative Pre-trained Transformers?

- **Generative Pre-trained Transformers** (GPT) are a type of deep learning model used to generate human-like text.
- Common uses include
  - answering questions
  - summarizing text
  - translating text to other languages
  - generating code
  - generating blog posts, stories, conversations, and other content types.
- There are **endless applications for GPT models**, and you can even fine-tune them on specific data to create even better results. By using transformers, you will be saving costs on computing, time, and other resources.

GPT-1

In 2018, OpenAI published a paper (Improving Language Understanding by Generative Pre-Training) about using natural language understanding using their GPT-1 language model. This model was a proof-of-concept and was not released publicly.

GPT-2

- The following year, OpenAI published another paper (Language Models are Unsupervised Multitask Learners) about their latest model, GPT-2.
- This time, the model was made available to the machine learning community and found some adoption for text generation tasks.
- GPT-2 could often generate a couple of sentences before breaking down. This was state-of-the-art in 2019.

Model performance on various tasks | GPT-2 paper

GPT-3

- In 2020, OpenAI published another paper (**Language Models are Few-Shot Learners**) about their GPT-3 model. The model had **100 times more parameters than GPT-2** and was trained on an even larger text dataset, resulting in better model performance.
- The model continued to be improved with various iterations known as the GPT-3.5 series, including the **conversation-focused ChatGPT**.
- This version took the world by storm after surprising the world with its ability to generate pages of human-like text.
- ChatGPT became the fastest-growing web application ever, reaching 100 million users in just two months.

Setting	NaturalQS	WebQS	TriviaQA
RAG (Fine-tuned, Open-Domain) [LPP+20]	**44.5**	**45.5**	**68.0**
T5-11B+SSM (Fine-tuned, Closed-Book) [RRS20]	36.6	44.7	60.5
T5-11B (Fine-tuned, Closed-Book)	34.5	37.4	50.1
GPT-3 Zero-Shot	14.6	14.4	64.3
GPT-3 One-Shot	23.0	25.3	**68.0**
GPT-3 Few-Shot	29.9	41.5	**71.2**

Results on three Open-Domain QA tasks | GPT-3 paper

GPT-4

GPT-4 has been developed to improve model "alignment" - the ability to follow user intentions while also making it more truthful and generating **less offensive or dangerous output**.

GPT-4

Performance improvements

- GPT-4 improves on GPT-3.5 models regarding the **factual correctness** of answers. The number of **"hallucinations,"** where the model makes factual or reasoning errors, is lower, with GPT-4 scoring **40% higher** than GPT-3.5 on OpenAI's internal factual performance benchmark.
- It also improves "**steerability**," which is the ability to change its behavior according to user requests. For example, you can command it to write in a different style or tone or voice.
- A further improvement is in the **model's adherence to guardrails. If you ask it to do something illegal or unsavory**, it is better at refusing the request.

Using Visual Inputs in GPT-4

- One major change is that **GPT-4 can use image inputs** (research preview only; not yet available to the public) and text. Users can specify any vision or language task by entering interspersed text and images.
- Examples showcased highlight **GPT-4 correctly interpreting complex imagery such as charts, memes, and screenshots** from academic papers.

Visual inputs: chart reasoning

Sample 2 of 7

User

What is the sum of average daily meat consumption for Georgia and Western Asia? Provide a step-by-step reasoning before providing your answer.

GPT-4

To find the sum of average daily meat consumption for Georgia and Western Asia, we will follow these steps:

1. Identify the average daily meat consumption for Georgia.
2. Identify the average daily meat consumption for Western Asia.
3. Add the values from steps 1 and 2.

Step 1: From the provided chart, we see that the average daily meat consumption for Georgia is 79.84 grams per person per day.
Step 2: From the chart, we see that the average daily meat consumption for Western Asia is 69.62 grams per person per day.
Step 3: Add the values from steps 1 and 2.
79.84 g (Georgia) + 69.62 g (Western Asia) = 149.46 g

So, the sum of average daily meat consumption for Georgia and Western Asia is 149.46 grams per person per day.

GPT-3.5
GPT-4
ChatGPT PLUS
Design a database schema
for an online merch store
Give me ideas
for what to do with my kids' art
Compare business strategies
for transitioning from budget to luxury vs. luxury to bu...
Show me a code snippet
of a website's sticky header
Send a message
ChatGPT may produce inaccurate information about people, places, or facts. ChatGPT September 25 Version

GPT-4 Performance Benchmarks

- OpenAI evaluated GPT-4 by simulating exams designed for humans, such as **the Uniform Bar Examination and LSAT for lawyers, and the SAT for university admission**.
- The results showed that GPT-4 achieved human-level **performance on various professional and academic benchmarks.**

Simulated exams	GPT-4 estimated percentile	GPT-4 (no vision) estimated percentile	GPT-3.5 estimated percentile
Uniform Bar Exam (MBE+MEE+MPT)[1]	298 / 400 ~90th	298 / 400 ~90th	213 / 400 ~10th
LSAT	163 ~88th	161 ~83rd	149 ~40th
SAT Evidence-Based Reading & Writing	710 / 800 ~93rd	710 / 800 ~93rd	670 / 800 ~87th
SAT Math	700 / 800 ~89th	690 / 800 ~89th	590 / 800 ~70th
Graduate Record Examination (GRE) Quantitative	163 / 170 ~80th	157 / 170 ~62nd	147 / 170 ~25th
Graduate Record Examination (GRE) Verbal	169 / 170 ~99th	165 / 170 ~96th	154 / 170 ~63rd
Graduate Record Examination (GRE) Writing	4 / 6 ~54th	4 / 6 ~54th	4 / 6 ~54th
USABO Semifinal Exam 2020	87 / 150 99th - 100th	87 / 150 99th - 100th	43 / 150 31st - 33rd
USNCO Local Section Exam 2022	36 / 60	38 / 60	24 / 60
Medical Knowledge Self-Assessment Program	75 %	75 %	53 %
Codeforces Rating	392 below 5th	392 below 5th	260 below 5th
AP Art History	5 86th - 100th	5 86th - 100th	5 86th - 100th
AP Biology	5 85th - 100th	5 85th - 100th	4 62nd - 85th
AP Calculus BC	4 43rd - 59th	4 43rd - 59th	1 0th - 7th

How LLM works

Model Types

text-to-text

Text-to-text models take a natural language input and product text output. These models are trained to learn the mapping between a pair of texts (e.g. translation from one language to another).

Model Types

Text-to-image

Text-to-image models are relatively new and are trained on a large set of images, each captioned with a short text description. Diffusion is one method used to achieve this.

Model Types

Text-to-video

Text-to-3D

Text-to-video models aim to generate a video representation from text input. The input text can be anything from a single sentence to a full script, the the output is a video that corresponds to the input text. Similarly Text-to-3D models generate three-dimensional objects that correspond to a user's text description (for use in game or other 3D worlds)

Applications
Video generation
Video editing
Game assets

Model Types

Text-to-task

Text-to-task models are trained to perform a specific task or action based on text input. This task can be a wide range of actions such as answering a question, performing a search, making a prediction, or taking some sort of action. For example, a text-to-task model could be trained to navigate web UI or make changes to a doc through the GUI.

Foundation Model

Foundation Model Example : Google Cloud

Foundation Model Examples : Google

AI Application Landscape

Generative AI vs. Large Language Models

Getting on to GPT-3 and GPT-4

Playground
Load a preset...
Save
View code
Share
Write a tagline for an ice cream shop.
Submit
0
Mode
Engine
text-davinci-002
Temperature 0.7
Maximum length 256
Stop sequences
Enter sequence and press Tab
Top P 1
Frequency penalty 0
Presence penalty 0
Best of 1
Inject start text
API components

Here is an overview of the different API components and their functionalities:

Execution Engine: It determines the language model used for execution. Choosing the right engine is the key to determining your model's capabilities and in turn getting the right output.

Response Length: The response length sets a limit on how much text the API includes in its completion. Because OpenAI charges by the length of text generated per API call, response length is a crucial parameter for anyone on a budget. A higher response length will cost more.

Temperature: The temperature controls the randomness of the response, represented as a range from 0 to 1. A lower value of temperature means the API will respond with the first thing that the model sees; a higher value means the model evaluates possible responses that could fit into the context before spitting out the result.

Top P: Top P controls how many random results the model should consider for completion, as suggested by the temperature dial, thus determining the scope of randomness. Top P's range is from 0 to 1. A lower value limits creativity, while a higher value expands its horizons.

Here is an overview of the different API components and their functionalities:

Frequency and Presence Penalty: The frequency penalty decreases the likelihood that the model will repeat the same line verbatim by “punishing” it. The presence penalty increases the likelihood that it will talk about new topics.

Best of: This parameter lets you specify the number of completions (n) to generate on the server-side and returns the best of “n” completions.

Stop Sequence: A stop sequence is a set of characters that signals the API to stop generating completions.

Inject Start & Restart Text: The inject start text and inject restart text parameters allow you to insert text at the beginning or end of the completion, respectively.

Show Probabilities: This option lets you debug the text prompt by showing the probability of tokens that the model can generate for a given input.

Execution Engines

- The OpenAI API offers four different **execution engines that differ in the number of parameters** used, performance capabilities, and price.
- The primary engines in increasing order of their capabilities and size are **Ada** (named after Ada Lovelace), **Babbage** (named after Charles Babbage), **Curie** (named after Madame Marie Curie) and **Davinci** (named after Leonardo da Vinci).

Workshop #2

Getting started with GPT Models

Instruction

1. Clone GitHub repository : https://github.com/vkrit/prompt_engineering_for_developers
2. Open Workshop 2 - Prompt Engineering for Developers
3. Study Codes and Run it

The Role of a Prompt Engineer

A new career path in AI?

- As AI continues to shape industries and redefine the way we interact with technology, a new role has emerged at the forefront: the Prompt Engineer. **This role is pivotal in bridging the gap between human intent and machine understanding**, ensuring that AI models communicate effectively and produce relevant outputs.
- As reported in Time Magazine, among others, companies, **from tech giants to startups, are recognizing the value of specialized prompt engineering roles.** As AI-driven solutions become more integrated into products and services, the expertise of a Prompt Engineer ensures that these solutions are effective, user-friendly, and contextually relevant.

Prompt Engineer & Librarian
Anthropic
Hybrid remote in San Francisco, CA

Estimated $70.2K - $88.8K a year | **Full-time**

- Build up a library of high quality **prompts** or **prompt** chains to accomplish a variety of tasks, with an easy guide to help users search for the one that meets...

Posted 30+ days ago · More...

Prompt Engineer
Usable Machines
Venice, CA 90291

Estimated $105K - $133K a year | **Weekends as needed**

- Document and share best practices for **prompt** engineering with the team.
- The opportunity to work alongside a talented team of **engineers** and innovators.

Posted 30+ days ago · More...

AI Prompt Engineer
Gannett
Remote

$135,000 - $150,000 a year | **Full-time**

- Collaborate with data **engineers** to ensure seamless integration of AI **prompt** models into the LocaliQ DMS infrastructure.
- Job Specification: AI **Prompt** Engineer.

Posted 30+ days ago · More...

View all 2 available locations

Prompt Engineer & Librarian

Anthropic

San Francisco, CA • Hybrid remote

Full-time

You must create an Indeed account before continuing to the company website to apply

Apply on company site

Job details

Here's how the job details align with your job preferences. Manage job preferences anytime in your profile.

Job Type
Full-time

Anthropic's AI technology is amongst the most capable and safe in the world. However, large language models are a new type of intelligence, and the art of instructing them in a way that delivers the best results is still in its infancy — it's a hybrid between programming, instructing, and teaching. You will figure out the best methods of prompting our AI to accomplish a wide range of tasks, then document these methods to build up a library of tools and a set of tutorials that allows others to learn prompt engineering, as well as work with high value partners to directly solve their challenges.

Given that the field of prompt-engineering is arguably less than 2 years old, this position is a bit hard to hire for! As a result, we ask that you share with us a specific prompt engineering project on LLMs that you're proud of in your application! Ideally this project should show off a complex and clever prompting architecture or a systematic evaluation of an LLM's behavior.

About Anthropic
Anthropic is an AI safety and research company that's working to build reliable, interpretable, and steerable AI systems. We want AI to be safe and beneficial for our customers and for society as a whole. Our interdisciplinary team has experience across ML, physics, policy, business and product.

Responsibilities:

- Discover, test, and document best practices for a wide range of tasks relevant to our customers.
- Build up a library of high quality prompts or prompt chains to accomplish a variety of tasks, with an easy guide to help users search for the one that meets their needs.
- Build a set of tutorials and interactive tools that teach the art of prompt engineering to our customers.

Prompt Engineer will have salaries ranging from $50,000 to over $150,000 per year.

Job postings in indeed.com

Skill required for prompt engineers

Technical Skills

- **Understanding of NLP** : Deep knowledge of NLP and algorithm
- **Familiarity with LLMs** : experience with LLMs such as GPT
- **Experimentation and iteration** : A ability to test, refine and optimize prompts based on model outputs.
- **Data Analysis** : Analyzing model responses, identifying patterns and make data-driven decisions.

Skill required for prompt engineers

Non-Technical Skills

- **Communication** : Conveying ideas, collaborate with teams and understand user needs.
- **Subject Matter Expertise** : Domain knowledge is valuable.
- **Language Proficiency** : Language, grammar, and semantics
- **Critical Thinking** : Evaluating model outputs, identifying biases.
- **Creativity** : Thinking outside the box, experimenting with new prompt styles.

How to customize LLMs

Strong Task Performance
Efficient Multitask Serving
Model Tuning
(a.k.a. "Fine-Tuning")
Pre-trained Model
🔥 Tunable 🔥
Input Text
Prompt Tuning
Pre-trained Model
❄ Frozen ❄
Tunable Soft Prompt
Input Text
Prompt Design
(e.g. GPT-3)
Pre-trained Model
❄ Frozen ❄
Engineered Prompt
Input Text

Prompt Engineering

Significance of Prompt in User Interaction

1. Guiding User Expectations:

- **LLM prompts act as a guide for users**, shaping their expectations about the type of information or response they will receive. For example, a prompt that is structured as a question encourages the model to provide an answer, whereas a prompt phrased as a command might lead to the execution of a specific task.
- By framing inputs in certain ways, users can effectively communicate their needs, whether it's **seeking information**, **requesting the creation of content**, or **needing assistance with problem-solving.**

Significance of Prompt in User Interaction

2. Determining Response Relevance and Accuracy:

- The **specificity** and **clarity** of a prompt directly impact the **relevance** and **accuracy** of the LLM's response.
- **Clear**, **specific prompts** enable the model to understand **the user's intent and context**, leading to more **accurate** and **relevant** outputs.
- Conversely, **vague or ambiguous prompts** may result in responses that are **less accurate or relevant**, as the model may struggle to pinpoint the user's exact needs or intentions.

Significance of Prompt in User Interaction

3. Enabling Personalized Interactions:

- LLM prompts can be tailored to elicit **personalized responses**. By including specific details or preferences in a prompt, users can receive responses that are more **aligned with their individual needs** or contexts.
- This aspect of prompting is particularly valuable in scenarios like **education or therapy**, where personalized feedback or guidance can significantly enhance the effectiveness of the interaction.

Significance of Prompt in User Interaction

4. Facilitating Creative and Generative Tasks:

- Prompts play a **crucial role in creative and generative tasks, such as writing, artwork creation, or code generation.** The way a prompt is structured can influence the style, tone, and content of the generated output.
- For instance, in **creative writing**, a prompt can set the scene, define characters, and suggest a plotline, guiding the LLM to generate a story that aligns with these parameters.

Significance of Prompt in User Interaction

5. Enhancing Accessibility and Inclusivity:

- Prompts can be designed to enhance **the accessibility of LLMs for users with diverse needs and abilities.** For instance, prompts that are simple, clear, and free from jargon can make the technology more accessible to individuals with varying levels of expertise or cognitive abilities.
- Additionally, prompts that take into account **different cultural contexts and languages** can make LLM interactions more inclusive and globally applicable.

*Source: **Tweeted by Andrej Karpathy** on June 18, 2020*

Workflow for LLM

What is prompt?

- In the context of Generative AI, a **"prompt" refers to the input given to a generative model that guides or influences the output it generates**. This concept is crucial across various types of generative AI models, such as text-based language models, image generation models, and others.

Basic Prompts

You can already accomplish a lot with **prompts**, but **the quality of the outcomes is determined on how much information you supply**. A prompt can include information such as the instruction or query you are providing to the model, as well as additional features such as inputs or examples.

Here's an example of **a basic prompt**:

Let’s attempt to make it a little better:

Instruction Prompting

One of the simplest prompting methods is just giving instructions

ChatGPT 4

You

Read the following sales email. Remove any personally identifiable information (PII), and replace it with the appropriate placeholder. For example, replace the name "John Doe" with "[Name]".

Hi John,

I'm writing to you because I noticed you recently purchased a new car. I'm a salesperson at a local dealership (Cheap Dealz), and I wanted to let you know that we have a great deal on a new car. If you're interested, please let me know.

Thanks,

Jimmy Smith
Phone: 410-904-9933
Email: jimmysmith@cheapdealz.com

ChatGPT

Certainly, here's the sales email with personally identifiable information removed:

Hi [Name],

I'm writing to you because I noticed you recently purchased a new car. I'm a salesperson at a local dealership ([Dealership Name]), and I wanted to let you know that we have a great deal on a new car. If you're interested, please let me know.

Thanks,

[Salesperson's Name]
Phone: [Phone Number]
Email: [Email Address]

Role Prompting

Another prompting technique is to assign a role to the AI. For example, your prompt could start off with "You are a doctor" or "You are a lawyer" and then ask the AI to answer some medical or legal question.

ChatGPT 4

You

You are a brilliant mathematician who can solve any problem in the world. Attempt to solve the following problem:
What is 100*100/400*56?

ChatGPT

The result of $100 \times 100/400 \times 56$ remains 1400.0.

ChatGPT 4

You

Imagine you are the CEO of a startup that specializes in eco-friendly packaging solutions. Your company is relatively new and is trying to make a mark in a competitive industry. Your task is to develop a strategic plan to increase market share within the next year. Consider factors like marketing, product innovation, and partnerships. What strategies would you propose to achieve this goal?

ChatGPT

As the CEO of a startup specializing in eco-friendly packaging solutions, developing a strategic plan to increase market share within a year would involve a multifaceted approach. Here are some strategies I would propose:

1. **Market Research and Target Audience Identification:**
   - Conduct thorough market research to understand current trends, customer needs, and gaps in the eco-friendly packaging market.
   - Identify target demographics, such as environmentally conscious businesses and consumers, and tailor marketing strategies to these groups.
2. **Brand Positioning and Awareness:**
   - Develop a strong brand identity that resonates with eco-consciousness, quality, and innovation.
   - Increase brand awareness through targeted marketing campaigns, leveraging social media, and content marketing focusing on the importance of sustainable packaging.

Few Shot Prompting

Yet another prompting strategy is few shot prompting, which **is basically just showing the model a few examples (called shots)** of what you want it to do. This technique helps the model to quickly adapt new tasks or type data.

You

Great product, 10/10: positive
Didn't work very well: negative
Super helpful, worth it: positive
It doesn't work at all!:

ChatGPT

Based on the provided examples of few-shot prompting for sentiment analysis, the response for the input "It doesn't work at all!" would be: negative.

You

Task: Write product descriptions.

Prompt with Few-Shot Examples:

Product: Apple iPhone 13
Description: The iPhone 13 features a sleek design, advanced dual-camera system, and A15 Bionic chip for lightning-fast performance. Enjoy brighter photos and up to 19 hours of video playback with this innovative smartphone.

Product: Nike Air Max 270
Description: Nike Air Max 270 combines comfort and style, featuring Nike's biggest heel Air unit yet for a soft ride. Its sleek design and breathable mesh upper make it a perfect choice for both athletes and casual wearers.

New Product for Description :
Product: Bose QuietComfort Headphones

ChatGPT

Based on the structure and style of the provided few-shot examples, here's a product description for the Bose QuietComfort Headphones:

Product: Bose QuietComfort Headphones
Description: Experience unparalleled sound quality with Bose QuietComfort Headphones, designed for superior noise cancellation and audio clarity. These headphones offer a comfortable, over-ear fit and advanced technology that lets you immerse yourself in music, podcasts, or calls. With long-lasting battery life and Bluetooth connectivity, they provide the ultimate listening experience for both music enthusiasts and busy professionals.

Variants of shot prompting

- 0 shot prompting: no examples are shown to the model
- 1 shot prompting: 1 example is shown to the model
- few shot prompting: 2+ examples are shown to the model

0-shot prompting

0-shot prompting is the most basic form of prompting. It is simply showing the model a prompt **without examples and asking it to generate a response**. As such, all of instruction and role prompts that you have seen so far are 0-shot prompts. An additional example of a 0-shot prompt is:

1-shot prompting

One-shot prompting involves providing **a single example** to a large language model, to guide it in performing a specific task. **This single example serves as a reference for the model, demonstrating the format or approach it should take in its response.**

Few-shot prompting

Few-shot prompting is when you show the model **2 or more examples**. All prompts above this variants section have been few-shot prompts. The few-shot analogue of the above two prompts is:

Combining Techniques

As we have seen in the previous pages, prompts can have varying formats and complexity. They can include **context, instructions, and multiple input-output examples**. Combining these different prompting techniques can lead to more powerful prompts.

ChatGPT 4

You

Role: Imagine you are a customer service manager at a leading electronics company. Your goal is to improve the quality of customer interactions.

Instruction: Develop guidelines for effective customer communication. Focus on empathy, clarity, and solution-oriented responses.

Examples:

Customer Complaint: "I recently purchased a laptop from your store, and it's already having battery issues."
Ideal Response: "I'm really sorry to hear about the issues you're facing with your laptop. Let's get this resolved for you. Could you provide me with the purchase details so I can check the warranty status and guide you through the next steps?"

Customer Complaint: "I am frustrated with the delay in the delivery of my order. It was supposed to arrive three days ago."
Ideal Response: "I understand your frustration with the delay, and I apologize for the inconvenience caused. Let me check the status of your order and expedite the delivery process."

Your Task:
Given these scenarios and responses, create a similar response for the following customer complaint:

Customer Complaint: "Your product manual is confusing, and I'm struggling to set up the device."

ChatGPT

In line with the role, instruction, and examples provided, here's an ideal response to the customer complaint about the confusing product manual:

Ideal Response: "I'm sorry to hear that you're finding the product manual confusing and are having difficulties with the setup. We certainly want to make sure that all our customers have a smooth experience with our products. Could you please tell me which part of the setup process is unclear? I'm here to guide you through each step, and we can also provide additional resources like an online tutorial or a step-by-step guide to assist you further."

This response adheres to the guidelines of empathy (acknowledging the customer's struggle), clarity (offering specific assistance), and a solution-oriented approach (proposing additional resources for help).

Parts of a Prompt

- A role
- An instruction/task
- A question
- Context
- Examples (few shot)

Context

Context is any **relevant information** that you want the model to use when **answering the question/performing the instruction**.

You

Background Information: You are the manager of a software development team that recently transitioned to a fully remote work setup. While the team has adapted to this change, you've noticed a decline in overall productivity and collaboration. Your task is to address these challenges and propose strategies to enhance productivity and teamwork in the remote environment.

Specific Elements to Consider:

Communication: How can communication be improved among team members?
Project Management: What tools or practices can be implemented for better project tracking and accountability?
Team Building: Suggest activities or initiatives to foster a sense of team unity and collaboration.
Your Task:
Based on this context, develop a comprehensive plan that includes specific strategies and tools to address the identified issues and improve team productivity and collaboration in a remote work setting.

A “Standard” Prompt

Standard Prompt

```
What is the capital of France?
```

Standard Prompt in QA format

```
Q: What is the capital of France?

A:
```

Few Shot Standard Prompt

```
What is the capital of Spain?
Madrid
What is the capital of Italy?
Rome
What is the capital of France?
```

Few Shot Standard Prompt in QA format

```
Q: What is the capital of Spain?
A: Madrid
Q: What is the capital of Italy?
A: Rome
Q: What is the capital of France?
A:
```

General Tips for Designing Prompts

- Start Simple
- The Instruction
- Specificity
- Avoid Impreciseness
- To do or not to do

Start Simple

- As you get started with designing prompts, you should keep in mind that it is really an **iterative process** that requires a lot of experimentation to get optimal results.
- You can start with **simple prompts and keep adding more elements and context as you aim for better results.** Iterating your prompt along the way is vital for this reason.

The Instruction

- You can design effective prompts for various simple tasks by using commands to instruct the model what you want to achieve, such as "**Write**", "**Classify**", "**Summarize**", "**Translate**", "**Order**", etc.
- Another recommendation is to use some clear separator like "**###**" to separate the instruction and context.

Prompt:

```
### Instruction ###
Translate the text below to Spanish:
Text: "hello!"
```

Output:

```
¡Hola!
```

Specificity

- Be very **specific** about the instruction and task you want the model to perform. **The more descriptive and detailed the prompt is, the better the results.**
- This is particularly important when you have a desired outcome or style of generation you are seeking.

Prompt:

```
Extract the name of places in the following text.
Desired format:
Place: <comma_separated_list_of_company_names>
Input: "Although these developments are encouraging to
researchers, much is still a mystery. “We often have a black box
between the brain and the effect we see in the periphery,” says
Henrique Veiga-Fernandes, a neuroimmunologist at the Champalimaud
Centre for the Unknown in Lisbon. “If we want to use it in the
therapeutic context, we actually need to understand the
mechanism.""
```

Output:

```
Place: Champalimaud Centre for the Unknown, Lisbon
```

Avoid Impreciseness

- Given the tips above about being detailed and improving format, it's easy to fall into the trap of wanting to be too clever about prompts and potentially creating imprecise descriptions.
- It's often **better to be specific and direct.** The analogy here is very similar to effective communication -- the more direct, the more effective the message gets across.

```
Explain the concept prompt engineering. Keep the explanation
short, only a few sentences, and don't be too descriptive.
```

```
Use 2-3 sentences to explain the concept of prompt engineering to
a high school student.
```

To do or not to do?

- Another common tip when designing prompts is to **avoid saying what not to do but say what to do instead.**
- This encourages more specificity and focuses on the details that lead to good responses from the model.

Prompt:

```
The following is an agent that recommends movies to a customer.
DO NOT ASK FOR INTERESTS. DO NOT ASK FOR PERSONAL INFORMATION.
Customer: Please recommend a movie based on my interests.
Agent:
```

Output:

```
Sure, I can recommend a movie based on your interests. What kind
of movie would you like to watch? Do you prefer action, comedy,
romance, or something else?
```

Prompt:

```
The following is an agent that recommends movies to a customer.
The agent is responsible to recommend a movie from the top global
trending movies. It should refrain from asking users for their
preferences and avoid asking for personal information. If the
agent doesn't have a movie to recommend, it should respond
"Sorry, couldn't find a movie to recommend today.".
Customer: Please recommend a movie based on my interests.
Agent:
```

Output:

```
Sorry, I don't have any information about your interests.
However, here's a list of the top global trending movies right
now: [list of movies]. I hope you find something you like!
```

Prompt Practice for each tasks

Text Summarization

One of the standard tasks in natural language generation is **text summarization.**

Text summarization can include many different flavors and domains. In fact, one of the most promising applications of language models is the ability to summarize articles and concepts into quick and easy-to-read summaries.

Prompt:

```
Antibiotics are a type of medication used to treat bacterial
infections. They work by either killing the bacteria or
preventing them from reproducing, allowing the body's immune
system to fight off the infection. Antibiotics are usually taken
orally in the form of pills, capsules, or liquid solutions, or
sometimes administered intravenously. They are not effective
against viral infections, and using them inappropriately can lead
to antibiotic resistance.
Explain the above in one sentence:
```

Output:

```
Antibiotics are medications used to treat bacterial infections by
either killing the bacteria or stopping them from reproducing,
but they are not effective against viruses and overuse can lead
to antibiotic resistance.
```

Workshop # 3

Text Summerization

Instruction

1. Clone GitHub repository : https://github.com/vkrit/prompt_engineering_for_developers
2. Open Workshop 3 - Summarization
3. Study Codes and Run it

Inferring

Inferring, in the context of ChatGPT, refers to the ability of the model to **generate plausible responses or predictions based on the information provided in the conversation**.

It involves understanding the context and making logical deductions or assumptions to generate a coherent and relevant response.

We have experienced a surge in online orders from a specific geographic region. What can you infer about the potential market demand in that area?

Based on the surge in online orders from a specific geographic region, several inferences can be made about the potential market demand in that area:

1. High demand: The surge in online orders suggests that there is a significant demand for the product or service being offered in that specific geographic region. The increased number of orders indicates that customers in that area are interested in purchasing the product.
2. Market growth: The surge in online orders may indicate a growing market in that specific geographic region. It suggests that the population or target audience in that area is receptive to the product or service, leading to increased demand and potential market expansion.
3. Marketing effectiveness: The surge in online orders could imply that the company's marketing efforts in that specific geographic region have been successful. It suggests that the marketing strategies, such as targeted advertising or promotional campaigns, have effectively reached and resonated with the audience in that area.
4. Competitive advantage: The surge in online orders might indicate that the company has a competitive advantage over other businesses operating in that specific geographic region. It suggests that the product or service offered by the company is preferred over alternatives available in the market, leading to higher demand and increased market share.
5. Potential for business expansion: The surge in online orders could signal an opportunity for the company to consider expanding its operations or establishing a physical presence in that specific geographic region. The increased demand indicates a market potential that can be further tapped into for growth and profitability.

It's important to note that these inferences are based on the given information and assumptions, and further analysis and market research would be necessary to validate these inferences and make informed business decisions.

Workshop # 4

Inferring

Instruction

1. Clone GitHub repository : https://github.com/vkrit/prompt_engineering_for_developers
2. Open Workshop 4 - Inferring
3. Study Codes and Run it

Transforming

“transforming" refers to tasks where the model **takes an input and generates a transformed or modified output.** It involves altering the input in some way to produce a desired result.

- Language translation
- Tone Transformation
- Data formatting
- Spell Check and Gramma Check
- Content generation
- Code generation
- Data augmentation

Transforming : Language Translation

Transforming : Data Formatting

Please format the following raw data into a tabular format with columns for 'Name', 'Email', and 'Phone Number':
John Doe, johndoe@email.com, 123-456-7890
Jane Smith, janesmith@email.com, 987-654-3210
Robert Johnson, robertjohnson@email.com, 555-123-4567

Sure! Here's the formatted data in a tabular format:

Name	Email	Phone Number
John Doe	johndoe@email.com	123-456-7890
Jane Smith	janesmith@email.com	987-654-3210
Robert Johnson	robertjohnson@email.com	555-123-4567

In this formatted table, the data is organized into three columns: 'Name', 'Email', and 'Phone Number'. Each row represents an individual entry with corresponding values for each column.

Transforming : Content Generation

Act as a customer engagement expert, Generate a blog post on the topic: 'Effective Strategies for Improving Customer Engagement'

Title: Effective Strategies for Improving Customer Engagement

Introduction:
In today's competitive business landscape, customer engagement plays a pivotal role in driving growth and building long-term relationships. Engaged customers are more likely to become loyal advocates, make repeat purchases, and provide valuable feedback. To help businesses thrive, this blog post presents effective strategies for enhancing customer engagement. By implementing these strategies, companies can forge stronger connections with their customers, foster brand loyalty, and drive business success.

1. Personalize the Customer Experience:
   Personalization is key to engaging customers on a deeper level. Utilize customer data and insights to tailor interactions, recommendations, and offers. Leverage personalized emails, targeted promotions, and customized product recommendations to make customers feel valued and understood. By demonstrating that you know their preferences and needs, you enhance their overall experience, leading to increased engagement and loyalty.
2. Leverage the Power of Social Media:
   In today's digital age, social media platforms offer unparalleled opportunities for customer engagement. Establish an active presence on relevant social media channels and engage in meaningful conversations with your audience. Respond promptly to customer inquiries, comments, and feedback. Use social listening tools to monitor brand mentions and gather insights, allowing you to proactively engage with customers, address concerns, and build relationships.
3. Implement a Customer Loyalty Program:

Transforming : Spell Check and Gramma Check

User: "Please check the following paragraph for any spelling or grammar errors: 'I has been working at this company for two years and have gaind valuable experience. I am excited about my futur prospects and cant wait to further develp my skills.'"

Assistant: [Spell Check and Grammar Check]

Assistant: Sure, let's check the paragraph for spelling and grammar errors:

Original: "I has been working at this company for two years and have gaind valuable experience. I am excited about my futur prospects and cant wait to further develp my skills."

Suggested revision: "I have been working at this company for two years and have gained valuable experience. I am excited about my future prospects and can't wait to further develop my skills."

Here's the corrected version of the paragraph. Changes made include:

- "I has" corrected to "I have" (subject-verb agreement)
- "gaind" corrected to "gained" (spelling)
- "futur" corrected to "future" (spelling)
- "cant" corrected to "can't" (contraction)
- "develp" corrected to "develop" (spelling)

I hope this helps! Let me know if there's anything else I can assist you with.

Transforming : Code Generation

Generate Python code to calculate the factorial of a given number.

Certainly! Here's an example of Python code to calculate the factorial of a given number:

python Copy code

```python
def factorial(n):
    if n == 0:
        return 1
    else:
        return n * factorial(n-1)

# Example usage
number = 5
result = factorial(number)
print(f"The factorial of {number} is: {result}")
```

In this code snippet, a recursive function `factorial()` is defined to calculate the factorial of a given number `n`. It checks if the number is 0 and returns 1 (base case). Otherwise, it recursively calls itself with `n-1` and multiplies it with the current `n`. Finally, an example usage is shown, where `number` is set to 5, and the factorial of `number` is calculated and printed.

You can replace the `number` variable with any other positive integer to calculate its factorial using the same code structure.

Generate SQL code to retrieve all customers from the 'Orders' table who have made a purchase in the last month.

Certainly! Here's an example of SQL code to retrieve all customers from the 'Orders' table who have made a purchase in the last month:

sql Copy code

```sql
SELECT DISTINCT CustomerName
FROM Orders
WHERE PurchaseDate >= DATE_SUB(CURRENT_DATE, INTERVAL 1 MONTH);
```

In this SQL code, we use the `SELECT` statement to retrieve the distinct `CustomerName` values from the 'Orders' table. The `WHERE` clause filters the results by checking if the `PurchaseDate` is greater than or equal to the current date minus one month (`DATE_SUB(CURRENT_DATE, INTERVAL 1 MONTH)`). This ensures that only customers who have made a purchase in the last month are included in the result set.

You can adapt this code to match the specific table and column names in your database. Additionally, you can include additional columns in the `SELECT` statement if you need more information about the customers who made purchases in the last month.

Transforming : Data Augmentation

- Data augmentation, in the context of ChatGPT, **refers to the process of generating new or modified data examples** by using the model to create variations of existing data.
- Capacities of ChatGPT includes
  - Data Generation
  - Paraphrasing and Rewriting
  - Sentiment Modification
  - Entity Replacement and Masking
  - Style Transfer

Data Augmentation : Data Generation

Generate a new customer review for a smartphone with negative sentiment.

Certainly! Here's an example of a generated customer review for a smartphone with negative sentiment:

Generated Review: "I recently purchased this smartphone, and I must say, it has been a disappointing experience. The battery life is incredibly poor, barely lasting a few hours even with minimal usage. The camera quality is far from satisfactory, producing grainy and blurry photos. Moreover, the phone tends to lag and freeze frequently, making it frustrating to use. I regret my decision of buying this smartphone and would not recommend it to anyone seeking a reliable and high-performing device."

Please note that the generated review reflects a negative sentiment towards the smartphone and highlights various aspects that the reviewer finds unsatisfactory. The review provides specific criticisms related to the battery life, camera quality, and overall performance of the smartphone.

Data Augmentation : Paraphrasing and Rewriting

Data Augmentation : Sentiment Modification

Modify the sentiment of the following review from positive to negative: 'The product is amazing and exceeded my expectations.

Modified review: 'The product is disappointing and fell short of my expectations.'

In the modified review, the sentiment has been changed from positive to negative. The phrase "amazing and exceeded my expectations" has been transformed to "disappointing and fell short of my expectations," conveying a negative viewpoint about the product's performance and not meeting the reviewer's expectations.

Data Augmentation : Entity Replacement and Masking

Data Augmentation : Style Transfer

Workshop # 5

Tranforming

Instruction

1. Clone GitHub repository : https://github.com/vkrit/prompt_engineering_for_developers
2. Open Workshop 5 - Transforming
3. Study Codes and Run it

Expanding

The term "**expanding**" tasks in the context of ChatGPT refers to the **ability of the model to generate additional information or provide more details on a given topic or query**.

It involves leveraging the model's knowledge and language generation capabilities to provide a broader and more comprehensive response.

Tell me more about renewable energy sources.

Renewable energy sources are forms of energy that are derived from natural processes that are continuously replenished. They offer a sustainable and clean alternative to conventional fossil fuels, helping to mitigate climate change and reduce environmental impact. Here are some key renewable energy sources:

1. Solar Energy: Solar power harnesses the energy of the sun by converting sunlight into electricity using photovoltaic (PV) cells. It is a widely accessible and abundant source of renewable energy.
2. Wind Energy: Wind turbines capture the kinetic energy of the wind and convert it into electricity. Wind farms are established in areas with consistent wind patterns to generate significant amounts of clean power.
3. Hydropower: Hydropower harnesses the energy from flowing or falling water, typically through dams or turbines. It is one of the oldest and most widely used forms of renewable energy.
4. Biomass Energy: Biomass refers to organic matter such as crop residues, wood, or dedicated energy crops. It can be converted into biofuels, biogas, or burned directly to generate heat or electricity.
5. Geothermal Energy: Geothermal power utilizes the Earth's natural heat from deep within its crust. This energy is harnessed through geothermal power plants to generate electricity or for direct heating and cooling purposes.
6. Tidal Energy: Tidal power harnesses the energy from ocean tides and currents. Turbines placed underwater capture the kinetic energy generated by the movement of water during tidal cycles.

These renewable energy sources offer numerous advantages, including reduced greenhouse gas emissions, diversification of energy sources, and the potential for decentralized energy production. They contribute to the transition towards a more sustainable and environmentally friendly energy landscape, promoting a greener future.

Workshop # 6
Expanding

Instruction

1. Clone GitHub repository : https://github.com/vkrit/prompt_engineering_for_developers
2. Open Workshop 6 - Expanding
3. Study Codes and Run it

Chatbot

- The "Chatbot" capability of ChatGPT refers to **its ability to engage in conversational interactions, simulating a chatbot-like behavior**.
- It allows users to have interactive and dynamic conversations with the model, where it can understand and respond to a series of prompts or messages in a conversational manner.

ChatGPT

Chatbot

Adding to the Context

```
messages =
[
    system
    user
    assistant
    user
    assistant
    ...
]
```

System Context

```
context = [
{'role':'system',
 'content':"""
You are OrderBot, an automated service
to collect orders for a pizza restaurant.
You first greet the customer,
then collect the order,
and then ask if it's a pickup or delivery.
You wait to collect the entire order,
then summarize it and check for a final time
if the customer wants to add anything else.
If it's a delivery, you ask for an address.
Finally you collect the payment.
Make sure to clarify all options, extras and sizes
to uniquely identify the item from the menu.
You respond in a short, very conversational
friendly style.
The menu includes
…
"""
}
```

Workshop # 7
Chatbot

Instruction

1. Clone GitHub repository : https://github.com/vkrit/prompt_engineering_for_developers
2. Open Workshop 7 - Chatbot
3. Study Codes and Run it

Reasoning

- Perhaps one of the most difficult tasks for an LLM today is one that requires some form of reasoning. **Reasoning is one of most interesting areas due to the types of complex applications that can emerge from LLMs.**
- There have been some improvements in tasks involving mathematical capabilities. That said, it's important to note that current LLMs struggle to perform reasoning tasks so this requires even more advanced prompt engineering techniques.
- In the context of ChatGPT, **"reasoning" refers to the model's ability to engage in logical reasoning, make deductions, draw conclusions, and provide coherent explanations based on the information provided.** It involves applying logical thinking and inferential abilities to arrive at an informed response.

Reasoning

If it is raining outside, and Jane doesn't have an umbrella, will she get wet?

Yes, if it is raining outside and Jane doesn't have an umbrella, it is likely that she will get wet without any protection from the rain. Without an umbrella or any other means to shield herself from the rain, Jane will be exposed to the falling raindrops, resulting in getting wet.

Reasoning

If a company invests heavily in marketing and has a strong online presence, will it likely attract more customers?

Yes, if a company invests heavily in marketing and has a strong online presence, it is likely to attract more customers. Investing in marketing allows a company to create awareness about its products or services, reach a wider audience, and effectively communicate its value proposition. A strong online presence, including a well-designed website, active social media presence, and digital advertising, enables the company to engage with potential customers and build brand visibility.

With effective marketing strategies and a strong online presence, a company can increase its visibility, generate leads, and capture the attention of its target audience. It allows the company to showcase its offerings, highlight its unique selling points, and differentiate itself from competitors. By consistently reaching out to potential customers through various marketing channels, the company can increase the likelihood of attracting more customers who are interested in its products or services.

However, it's important to note that the success of marketing efforts also depends on various factors such as the quality of products or services, competitive landscape, target audience preferences, and overall market conditions. A comprehensive marketing strategy, tailored to the specific business and its target market, is crucial for maximizing the impact of marketing investments and effectively attracting and retaining customers.

Advance Prompting

Chain-of-Thought Prompting

Chain-of-thought (CoT) prompting enables complex reasoning capabilities through intermediate reasoning steps. You can combine it with few-shot prompting to get better results on more complex tasks that require reasoning before responding.

(a) Few-shot

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?
A: The answer is 11.

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:

(Output) The answer is 8. ✗

(b) Few-shot-CoT

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?
A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. 5 + 6 = 11. The answer is 11.

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:

(Output) The juggler can juggle 16 balls. Half of the balls are golf balls. So there are 16 / 2 = 8 golf balls. Half of the golf balls are blue. So there are 8 / 2 = 4 blue golf balls. The answer is 4. ✓

(c) Zero-shot

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: The answer (arabic numerals) is

(Output) 8 ✗

(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. ✓

Automatic Chain-of-Thought (Auto-CoT)

- When applying chain-of-thought prompting with demonstrations, the process involves hand-crafting effective and diverse examples.
- This manual effort could lead to suboptimal solutions. Researchers propose an approach to eliminate manual efforts **by leveraging LLMs with "Let's think step by step" prompt to generate reasoning chains** for demonstrations one by one.
- Auto-CoT, which samples questions with diversity and generates reasoning chains to construct the demonstrations.

Auto-CoT consists of two main stages:

- Stage 1): question clustering: partition questions of a given dataset into a few clusters
- Stage 2): demonstration sampling: select a representative question from each cluster and generate its reasoning chain using Zero-Shot-CoT with simple heuristics

Auto Demos One by One
Q: While shopping for music online, Zoe bought 3 …
Q: A chef needs to cook 9 potatoes. He has already…
Clustering
1
k
LLM
Demo Construction
1
Q: While shopping for music online … A: Let's …
Sampling by Selection Criteria
k
Q: A chef needs to cook 9 potatoes ... A: Let's …
Q: While shopping for music online, Zoe bought 3 country albums and 5 pop albums. Each album came with a lyric sheet and had 3 songs. How many songs did Zoe buy total?
A: Let's think step by step. Zoe bought 3 country albums. Each album has 3 songs. So she bought 3*3=9 songs from the country albums. Zoe bought 5 pop albums. Each album has 3 songs. So she bought 5*3=15 songs from the pop albums. Zoe bought 9+15=24 songs in total. The answer is 24.
…
Q: A chef needs to cook 9 potatoes. He has already cooked 7. If each potato takes 3 minutes to cook, how long will it take him to cook the rest?
A: Let's think step by step. The chef has already cooked 7 potatoes. That means it has taken him 7 * 3 minutes to cook those 7 potatoes. That means it will take him 3 more minutes to cook each of the remaining 2 potatoes …
Q: A pet store had 64 puppies. In one day they sold 28 of them and put the rest into cages with 4 in each cage. How many cages did they use?
A: Let's think step by step.
Test Question
LLM
In-Context Reasoning
The pet store had 64 puppies. They sold 28 of them. That means they have 36 puppies left. They put the rest into cages with 4 in each cage. That means they have 9 cages. The answer is 9.

You

Scenario: You are a marketing manager tasked with allocating the annual budget for various marketing campaigns. Your total budget is $500,000. You need to decide how much to allocate to online advertising, event sponsorships, and print media. The decision should be based on the return on investment (ROI) from last year's campaigns.

Last Year's Data (Provided to the Model):

Online advertising cost $150,000 and generated an ROI of 20%.
Event sponsorships cost $200,000 with an ROI of 15%.
Print media cost $100,000 with an ROI of 5%.
Auto-Chain of Thought Prompt:

Analyze Last Year's ROI: Calculate the actual returns from each category last year.
Compare ROI: Assess which categories yielded the highest ROI.
Budget Allocation Based on ROI: Propose a new budget allocation for this year, focusing on maximizing overall ROI.
Consider Market Trends: Adjust the proposed allocation considering current market trends and potential changes from last year.
Final Recommendation: Provide a reasoned allocation of the $500,000 budget across the three marketing channels.

ChatGPT

To allocate the annual marketing budget based on the provided scenario and Auto-Chain of Thought approach, let's go through the steps:

1. **Analyze Last Year's ROI:**
   - Online advertising: $150,000 with a 20% ROI = $150,000 * 20% = $30,000 return.
   - Event sponsorships: $200,000 with a 15% ROI = $200,000 * 15% = $30,000 return.
   - Print media: $100,000 with a 5% ROI = $100,000 * 5% = $5,000 return.
2. **Compare ROI:**
   - Online advertising yielded a 20% ROI, the highest among the three.
   - Event sponsorships also performed well with a 15% ROI.
   - Print media had the lowest ROI at 5%.
3. **Budget Allocation Based on ROI:**
   - Given the higher ROI, a larger portion of the budget should be allocated to online advertising.
   - Event sponsorships should also receive a significant allocation but slightly less than online advertising.
   - Allocate a smaller portion to print media due to its lower ROI.
4. **Consider Market Trends:**
   - With increasing digital consumption, the effectiveness of online advertising is likely to grow.
   - Event sponsorships may vary based on the current year's events and market conditions.
   - Print media, generally on the decline, should be allocated less budget unless specific strategic opportunities are identified.
5. **Final Recommendation:**
   - Online Advertising: Allocate $250,000 considering its high ROI and growing digital trend.
   - Event Sponsorships: Allocate $200,000, maintaining a strong presence but slightly less than digital.
   - Print Media: Allocate $50,000, reflecting its lower ROI and market trends.

This allocation focuses on maximizing overall ROI while considering market trends and the performance of each marketing channel from the previous year. The allocation can be adjusted as per real-time market analysis and specific campaign goals.

Self-Consistency

Self-consistency aims "to replace the naive greedy decoding used in chain-of-thought prompting". **The idea is to sample multiple, diverse reasoning paths through few-shot CoT, and use the generations to select the most consistent answer.** This helps to boost the performance of CoT prompting on tasks involving arithmetic and commonsense reasoning.

Prompt:

```
When I was 6 my sister was half my age. Now
I'm 70 how old is my sister?
```

Output:

```
35
```

The output is wrong! How may we improve this with self-consistency?

```
Q: There are 15 trees in the grove. Grove workers will plant trees in the grove today.
After they are done,
there will be 21 trees. How many trees did the grove workers plant today?
A: We start with 15 trees. Later we have 21 trees. The difference must be the number of
trees they planted.
So, they must have planted 21 - 15 = 6 trees. The answer is 6.
Q: If there are 3 cars in the parking lot and 2 more cars arrive, how many cars are in the
parking lot?
A: There are 3 cars in the parking lot already. 2 more arrive. Now there are 3 + 2 = 5
cars. The answer is 5.
Q: Leah had 32 chocolates and her sister had 42. If they ate 35, how many pieces do they
have left in total?
A: Leah had 32 chocolates and Leah's sister had 42. That means there were originally 32 +
42 = 74
chocolates. 35 have been eaten. So in total they still have 74 - 35 = 39 chocolates. The
answer is 39.
Q: Jason had 20 lollipops. He gave Denny some lollipops. Now Jason has 12 lollipops. How
many lollipops
did Jason give to Denny?
A: Jason had 20 lollipops. Since he only has 12 now, he must have given the rest to Denny.
The number of
lollipops he has given to Denny must have been 20 - 12 = 8 lollipops. The answer is 8.
Q: Shawn has five toys. For Christmas, he got two toys each from his mom and dad. How many
toys does
he have now?
A: He has 5 toys. He got 2 from mom, so after that he has 5 + 2 = 7 toys. Then he got 2
more from dad, so
in total he has 7 + 2 = 9 toys. The answer is 9.
Q: There were nine computers in the server room. Five more computers were installed each
day, from
monday to thursday. How many computers are now in the server room?
A: There are 4 days from monday to thursday. 5 computers were added each day. That means
in total 4 * 5 =
20 computers were added. There were 9 computers in the beginning, so now there are 9 + 20
= 29 computers.
The answer is 29.
Q: Michael had 58 golf balls. On tuesday, he lost 23 golf balls. On wednesday, he lost 2
more. How many
golf balls did he have at the end of wednesday?
A: Michael initially had 58 balls. He lost 23 on Tuesday, so after that he has 58 - 23 =
35 balls. On
Wednesday he lost 2 more so now he has 35 - 2 = 33 balls. The answer is 33.
Q: Olivia has $23. She bought five bagels for $3 each. How much money does she have left?
A: She bought 5 bagels for $3 each. This means she spent 5
Q: When I was 6 my sister was half my age. Now I'm 70 how old is my sister?
A:
```

```
When I was 6 my sister was half my age, so she was 3. Now I am 70, so she is 70 - 3 = 67.
The answer is 67.
```

Generated Knowledge Prompting

LLMs continue to be improved and one popular technique includes the ability to **incorporate knowledge or information** to help the model make more accurate predictions.

Prompt:

```
Part of golf is trying to get a higher point total than others. Yes or No?
```

Output:

```
Yes.
```

Prompt:

```
Question: Part of golf is trying to get a higher point total than others. Yes or No?
Knowledge: The objective of golf is to play a set of holes in the least number of strokes.
A round of golf typically consists of 18 holes. Each hole is played once in the round on a
standard golf course. Each stroke is counted as one point, and the total number of strokes
is used to determine the winner of the game.
Explain and Answer:
```

Answer 1 (confidence very high):

```
No, the objective of golf is not to get a higher point total than others. Rather, the
objective is to play a set of holes in the least number of strokes. The total number of
strokes is used to determine the winner of the game, not the total number of points.
```

Tree of Thoughts (ToT)

- **Tree of Thoughts (ToT)**, a framework that generalizes over chain-of-thought prompting and encourages exploration over thoughts that serve as intermediate steps for general problem solving with language models.
- ToT maintains a tree of thoughts, where thoughts represent coherent language sequences that serve as intermediate steps toward solving a problem. **This approach enables an LM to self-evaluate the progress intermediate thoughts make towards solving a problem through a deliberate reasoning process.** The LM ability to generate and evaluate thoughts is then combined with search algorithms (e.g., breadth-first search and depth-first search) to enable systematic exploration of thoughts with lookahead and backtracking.

(a) Input-Output Prompting (IO)

(c) Chain of Thought Prompting (CoT)

(c) Self Consistency with CoT (CoT-SC)

(d) Tree of Thoughts (ToT)

You

Scenario: My company is considering launching a new product.

Initial Consideration: Market Research

Branch A: If market research shows strong demand,
- Consider production scale.
- Assess marketing strategies.

Branch B: If market research shows weak demand,
- Evaluate the need for product modification.
- Consider alternative markets or niches.

Production Scale Assessment

Branch A1: For large-scale production,
- Explore supply chain readiness.
- Calculate the required investment.

Branch B1: For small-scale production,
- Consider local suppliers.
- Plan for gradual scaling based on demand.

Marketing Strategies

Branch A2: For a mass-market approach,
- Develop widespread advertising campaigns.
- Partner with major retailers.

Branch B2: For a niche market approach,
- Target specific customer segments.
- Use specialized marketing channels.

Final Decision:

- Based on the outcomes of the above branches, decide on whether to proceed with the product launch.
- If proceeding, select the appropriate scale of production and marketing strategy based on the initial market research and subsequent assessments.

ChatGPT

Based on the Tree of Thought structure provided for your company's new product launch, let's walk through the decision-making process:

1. **Market Research Outcome:**
   - **Scenario 1 (Branch A):** If market research indicates strong demand,
     - **Production Scale:** Prepare for large-scale production by assessing supply chain readiness and calculating necessary investments (Branch A1).
     - **Marketing Strategy:** Develop a mass-market approach with widespread advertising campaigns and partnerships with major retailers (Branch A2).
   - **Scenario 2 (Branch B):** If market research shows weak demand,
     - **Product Modification:** Evaluate the need for modifying the product to better meet market needs (Branch B).
     - **Alternative Markets:** Explore niche markets or specific customer segments that might be more receptive to the product (Branch B2).
2. **Assessment of Production Scale (Based on Scenario):**
   - **Large-Scale Production (Branch A1):** If opting for large-scale production,
     - Check the readiness of the supply chain to handle large volumes.
     - Calculate the financial investment required to support this scale.
   - **Small-Scale Production (Branch B1):** If considering small-scale production,
     - Look into local suppliers for more manageable production runs.
     - Plan for gradual scaling based on market demand and reception.
3. **Marketing Strategies (Based on Initial Research and Production Scale):**
   - **Mass-Market Approach (Branch A2):** If the product has broad appeal,
     - Initiate widespread advertising campaigns across various media.
     - Form partnerships with major retailers to maximize reach.
   - **Niche Market Approach (Branch B2):** If focusing on specific segments,
     - Develop targeted marketing strategies tailored to these segments.
     - Use specialized marketing channels that resonate with the niche market.
4. **Final Decision:**
   - **Proceed with Launch:** If the assessment supports the product's market potential, proceed with the launch. Choose the scale of production and marketing strategy based on the comprehensive analysis from the branches.
   - **Reconsider or Delay Launch:** If the assessments reveal significant risks or insufficient market potential, reconsider the launch. Explore ways to refine the product or strategy before proceeding.

Multimodal Prompting

- Multimodal prompts involve using **more than one mode of communication or sensory channel** to convey information or instructions.
- This means combining **text, images, audio, video, or interactive elements** to create a more engaging and effective way of communication or task execution.

Visual Math and Text Reasoning

What is "a" in the first equation?
What is y in the last equation?

5 + a = 7	40 + y = 45
10 + y = 15	39 + a = 41
20 + a = 22	24 + y = 29
48 + a = 50	15 + a =17
10 + a = 12	34 + y = 39

The value of "a" in the first equation "5 + a = 7" is 2.

The value of "y" in the last equation "34 + y = 39" is 5.

Visual-Conditioned Joke/Meme

What is funny about it?

The humor in the comic may come from the situation, the characters, or the dialogue. The person is feeling unwell and is asked if they are pregnant or have the coronavirus, which is a current and relevant topic. The humor may come from the absurdity of the situation or the unexpectedness of the question.

Spatial/Coordinate Understanding

Where is the frisbee?
What is the relative position of the frisbee and the man?

The frisbee is located at coordinates (63,119,133,177).

The frisbee is located above the man in the image. The man is falling on the ground while the frisbee is in the air. The frisbee is also closer to the top of the image than the man..

Visual Planning and Prediction

What should I do after kneading the dough, before dividing it into pieces? For how long?

After kneading the dough, you should put it in a warm place before dividing it into pieces, according to the recipe in the image.

You should put the dough in a warm place for 1 hour.

Multi-Image Reasoning

How much in total did I spend on travel?
How much in total did I pay for taxes?

You spent $1299.90 on airfare and $43.83 on an Uber ride for a total of $1343.73 on travel.

You paid $323.23 in total for taxes. This includes $321.90 for taxes and fees for the flight purchase and $1.33 for sales tax for the Windmill Family Restaurant.

Multi-Hop Document Understanding

Excluding the regular cab, what is the average total fueling cost?

The average total fueling cost excluding the Ford F150 (Regular Cab) is $76.55. This is calculated by taking the sum of the total fueling cost of Toyota RAV4 and Honda Civic and dividing it by 2. ($82.56 + $70.55) / 2 = $76.55.

Open-World Concept Understanding

What is the brand of these drink? Which one is unsweetened?

The brand of these drinks is "BETTER THAN MILK". The unsweetened one is "BETTER THAN MILK organic OAT DRINK".

Who are the celebrities in this image?

Keira Knightley and Natalie Portman are the celebrities in this image.

Is it safe to eat this mushroom?

Morel mushrooms are generally considered safe to eat when properly cooked.

Video Summarization/Event Localization

Can you breakdown this video tutorial to step-by-step instructions, with start and end time?

Here's a step-by-step breakdown of the tutorial with start and end time.

··· Add Lettuce Add Tomato Add Bacon ···

Workshop # 8
Advanced Prompting

Use cases of Prompt Engineering

In different domain

Prompt Engineering in Different Domain

Prompt engineering can be employed in various domains to shape the behavior of models and elicit desired responses. Here are some use cases of prompt engineering in different domains:

Customer Service:

Prompt: "You are a customer service representative. A customer reports a product defect. Respond to the customer."

Use case: Prompt engineering can guide the model to provide empathetic and helpful responses, ensuring the customer's concern is addressed appropriately.

Prompt Engineering in Different Domain

Technical Support:

Prompt: "You are a technical support agent. A customer is experiencing network connectivity issues. Troubleshoot and assist the customer."

Use case: Prompt engineering helps the model generate detailed and accurate troubleshooting steps or requests for necessary information to diagnose and resolve the network connectivity problem.

Legal Advice:

Prompt: "You are a lawyer. A client seeks advice regarding a breach of contract. Provide legal guidance."

Use case: Prompt engineering can ensure the model provides accurate legal information, explains legal concepts clearly, and offers advice based on relevant laws and regulations.

Prompt Engineering in Different Domain

Sales and Marketing:

Prompt: "You are a sales representative. Convince a potential customer to purchase a product. Highlight its key features and benefits."

Use case: Prompt engineering can guide the model to emphasize the product's unique selling points, address potential customer concerns, and provide persuasive arguments to drive sales.

Content Creation:

Prompt: "Write a blog post about healthy eating. Provide tips and recipes for a balanced diet."

Use case: Prompt engineering helps the model generate informative and engaging content by suggesting specific tips, recipes, or sections to include in the blog post.

Prompt Engineering in Different Domain

Language Learning:

Prompt: "You are a language tutor. Teach a beginner-level lesson on basic greetings in French."

Use case: Prompt engineering guides the model to provide step-by-step instructions, useful phrases, and pronunciation tips for teaching basic French greetings to language learners.

Medical Diagnosis:

Prompt: "You are a doctor. Evaluate a patient's symptoms of fever, cough, and fatigue. Provide a diagnosis and recommended treatment."

Use case: Prompt engineering helps the model generate logical and accurate diagnoses based on the presented symptoms and provide appropriate treatment recommendations.

Fine-tuning ChatGPT

Fine-tuning GPT-3

- Fine-tuning lets you get more out of the models available through the API by providing:
  - Higher quality results than prompt design
  - Ability to train on more examples than can fit in a prompt
  - Token savings due to shorter prompts
  - Lower latency requests
- GPT-3 has been pre-trained on a vast amount of text from the open internet. When given a prompt with just a few examples, it can often intuit what task you are trying to perform and generate a plausible completion. This is often called "few-shot learning."
- Fine-tuning improves on few-shot learning by training on many more examples than can fit in the prompt, letting you achieve better results on a wide number of tasks. Once a model has been fine-tuned, you won't need to provide examples in the prompt anymore. This saves costs and enables lower-latency requests.

GradientFlow.com

Fine-Tuning Language Models: A Simple Playbook

Select a pre-trained model

AI21labs ANTHROP\C
co:here Google
Hugging Face OpenAI
Meta PowerML

... and more. Check model license.

Define the task

- classification
- entity extraction
- translation
- other

Prepare Data

- gather, pre-process, label
- split new data:
  - training set
  - validation set
  - test data

Choose a fine-tuning strategy

- fine tune full model
- or modify only last layers; other layers are frozen
- or use other strategy

Configure the model

Set up
- model architecture
- hyperparameters
- optimizer

You can use pre-trained model as starting point.

Fine-tune the model

Train model on your specific tasks/data
- run through multiple epochs
- adjust hyperparameters

Monitor loss/accuracy on validation set.

Evaluate

Use the validation data set to evaluate the performance of the model

Iterate and improve

Repeat previous steps until you achieve satisfactory results.

Test the model

Use model on held-out test set to confirm its performance on unseen data

Deploy or use the model

Fine-tuning GPT-3

- At a high level, fine-tuning involves the following steps:
  - Prepare and upload training data
  - Train a new fine-tuned model
  - Use your fine-tuned model

Installation

We recommend using our OpenAI command-line interface (CLI). To install this, run

```
pip install --upgrade openai
```

(The following instructions work for version **0.9.4** and up. Additionally, the OpenAI CLI requires python 3.)

Set your `OPENAI_API_KEY` environment variable by adding the following line into your shell initialization script (e.g. .bashrc, zshrc, etc.) or running it in the command line before the fine-tuning command:

```
export OPENAI_API_KEY="<OPENAI_API_KEY>"
```

Prepare training data

Training data is how you teach GPT-3 what you'd like it to say.

Your data must be a JSONL document, where each line is a prompt-completion pair corresponding to a training example. You can use our CLI data preparation tool to easily convert your data into this file format.

```
{"prompt": "<prompt text>", "completion": "<idea ne
{"prompt": "<prompt text>", "completion": "<ideal gene
{"prompt": "<prompt text>", "completion": "<ideal gene
...
```

CLI data preparation tool

We developed a tool which validates, gives suggestions and reformats your data:

```
openai tools fine_tunes.prepare_data -f <LOCAL_FILE>
```

This tool accepts different formats, with the only requirement that they contain a prompt and a completion column/key. You can pass a **CSV, TSV, XLSX, JSON** or **JSONL** file, and it will save the output into a JSONL file ready for fine-tuning, after guiding you through the process of suggested changes.

Create a fine-tuned model

The following assumes you've already prepared training data following the above instructions.

Start your fine-tuning job using the OpenAI CLI:

```
openai api fine_tunes.create -t <TRAIN_FILE_ID_OR_PATH> -m <BASE_MODEL>
```

Where `BASE_MODEL` is the name of the base model you're starting from (ada, babbage, curie, or davinci). You can customize your fine-tuned model's name using the suffix parameter.

Running the above command does several things:

1. Uploads the file using the files API (or uses an already-uploaded file)
2. Creates a fine-tune job
3. Streams events until the job is done (this often takes minutes, but can take hours if there are many jobs in the queue or your dataset is large)

After you've started a fine-tune job, it may take some time to complete. Your job may be queued behind other jobs on our system, and training our model can take minutes or hours depending on the model and dataset size. If the event stream is interrupted for any reason, you can resume it by running:

```
openai api fine_tunes.follow -i <YOUR_FINE_TUNE_JOB_ID>
```

When the job is done, it should display the name of the fine-tuned model.

In addition to creating a fine-tune job, you can also list existing jobs, retrieve the status of a job, or cancel a job.

```
# List all created fine-tunes
openai api fine_tunes.list

# Retrieve the state of a fine-tune. The resulting object includes
# job status (which can be one of pending, running, succeeded, or failed)
# and other information
openai api fine_tunes.get -i <YOUR_FINE_TUNE_JOB_ID>

# Cancel a job
openai api fine_tunes.cancel -i <YOUR_FINE_TUNE_JOB_ID>
```

Use a fine-tuned model

When a job has succeeded, the `fine_tuned_model` field will be populated with the name of the model. You may now specify this model as a parameter to our Completions API, and make requests to it using the Playground.

After your job first completes, it may take several minutes for your model to become ready to handle requests. If completion requests to your model time out, it is likely because your model is still being loaded. If this happens, try again in a few minutes.

You can start making requests by passing the model name as the `model` parameter of a completion request:

OpenAI CLI:

```
openai api completions.create -m <FINE_TUNED_MODEL> -p <YOUR_PROMPT>
```

cURL:

```
curl https://api.openai.com/v1/completions \
  -H "Authorization: Bearer $OPENAI_API_KEY" \
  -H "Content-Type: application/json" \
  -d '{"prompt": YOUR_PROMPT, "model": FINE_TUNED_MODEL}'
```

Python:

```
import openai
openai.Completion.create(
    model=FINE_TUNED_MODEL,
    prompt=YOUR_PROMPT)
```

Workshop # 9
Fine-tuning GPT

Workshop #9

1. Clone GitHub repository : https://github.com/vkrit/prompt_engineering_for_developers
2. Open Workshop 9 - Fine tuning
3. Study Codes and Run it

veerasak.kritsanapraphan@gmail.com