



# COMP 336 I Natural Language Processing

## Lecture I: Introduction

Spring 2024

# Logistics

- Location: KB 132
- Meetings: Tuesday 9:30 am - 10:20 am and Friday 9:30 am - 11:20 am
- Instructor: Tao Yu (<https://taoyds.github.io/>)
- Office hours: Wednesday 4 - 5 pm @IDS

# Logistics

Course website: <https://taoyds.github.io/courses/comp3361>

- We will maintain the website for schedule, lecture slides, reading lists, grading policies, etc
- Only submit your reports on Moodle.

# Logistics

Slack: [https://join.slack.com/t/slack-fdv4728/shared\\_invite/zt-2asgddr0h-6wIXbRndwKhBw2IX2~ZrjQ](https://join.slack.com/t/slack-fdv4728/shared_invite/zt-2asgddr0h-6wIXbRndwKhBw2IX2~ZrjQ)

- We will use Slack as the primary mode of communication. DM me on Slack instead of emails.
  - Answer any questions about lectures, assignments, grading, and so on
  - Share random thoughts, highlight interesting papers, brag about cool finding there.
- Join Slack via the invitation link above.

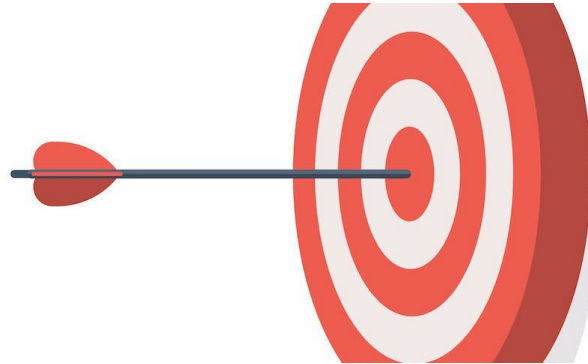


# Course prerequisites

- COMP3314 or COMP3340; and MATH1853
- Familiarity with deep learning and machine learning
- Familiarity with Python programming
- Helpful: exposure to AI assistants such as ChatGPT

# Course goals

- Understand core techniques and modern advances in NLP, especially in the era of large language models.
- Design, implement, and test NLP systems based on large language models.



# Components and grading

- **Assignments: 40%**
  - ~2 assignments, 20% for each
- **Course project: 30%**
  - More guidelines will be announced soon
- **In-class exam: 25%**
- **Class participation: 5%**

# Policy on ChatGPT, Copilot, and other AI assistants

- This course emphasizes understanding the capabilities and limitations of these AI systems, and there's no better way to do that than by using them! **Collaboration with these systems is allowed**, treating them as collaborators in the problem-solving process. However, **Using them to substantially complete assignments will be considered a violation of the Honor Code.**



# Class readings

- Readings from textbook chapters, blogs, tutorials, and papers will be posted on the course website.
- You may find it useful to do these readings before lecture as preparation or after lecture to review, but you are not expected to know everything discussed in the textbook if it isn't covered in lecture.
- Paper readings are intended to supplement the course material if you are interested in diving deeper on particular topics.

# Topics and schedule (tentative)

- Introduction and NLP model basics
- Large language models (LLMs)
- NLP applications
- Advanced LLM topics

# Introduction and NLP model basics

- Word embeddings
- Text Classification and Language Modeling
- Sequence-to-Sequence, Attention, Transformers

# Large language models (LLMs)

- LLM pretraining
- LLM Prompting, in-context learning
- LLM evaluation, data, and benchmarking
- Instruction tuning for LLMs
- LLM alignment/RLHF

# NLP applications

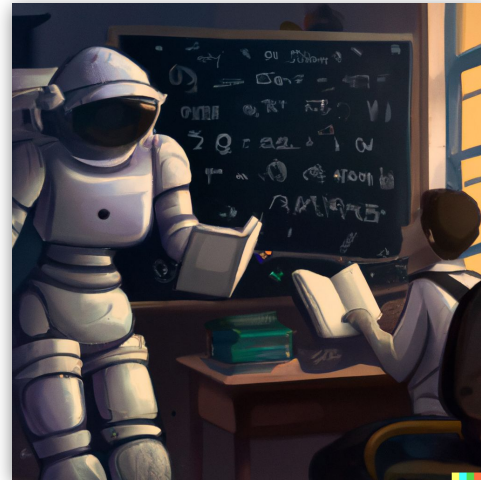
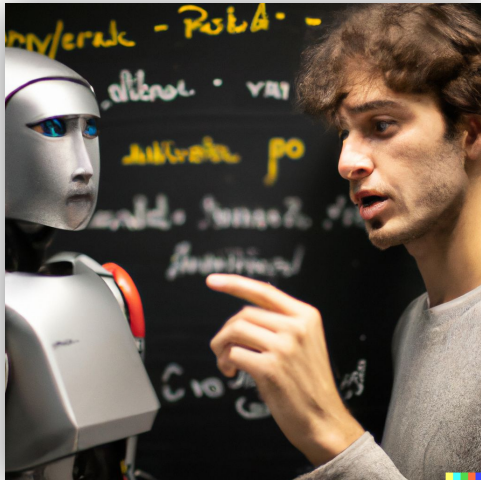
- Question answering, reasoning
- Text generation
- Semantic parsing, code generation
- LM agent, language grounding

# Advanced LLM topics

- Robustness, interpretability, explainability of LLMs
- Bias, toxicity, and privacy in LLMs
- Parameter-efficient LM tuning
- Efficient LLM methods and Infrastructure
- Multimodal LM, language in robotics, and embodied interaction

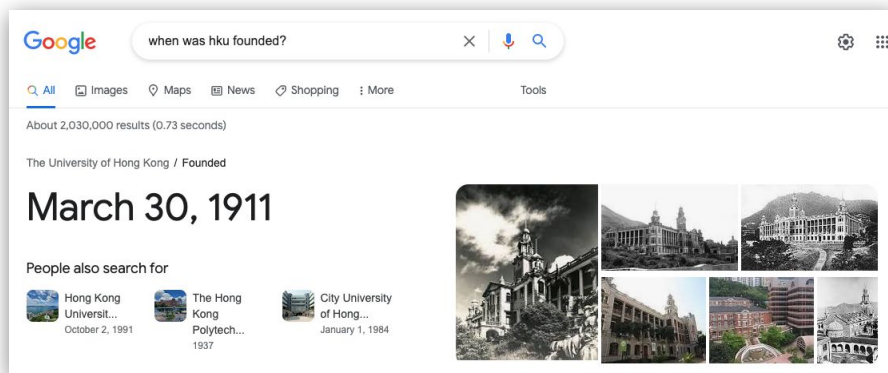
# What is NLP? Wait, what is language?

- **Language** is the abstraction of the real world!
- **Natural Language Processing (NLP)** aims to teach computers human languages a computational perspective.



# About NLP: teaching computers human languages

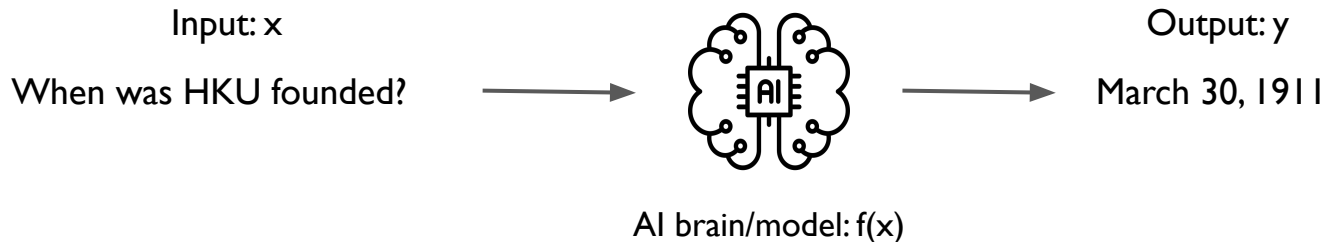
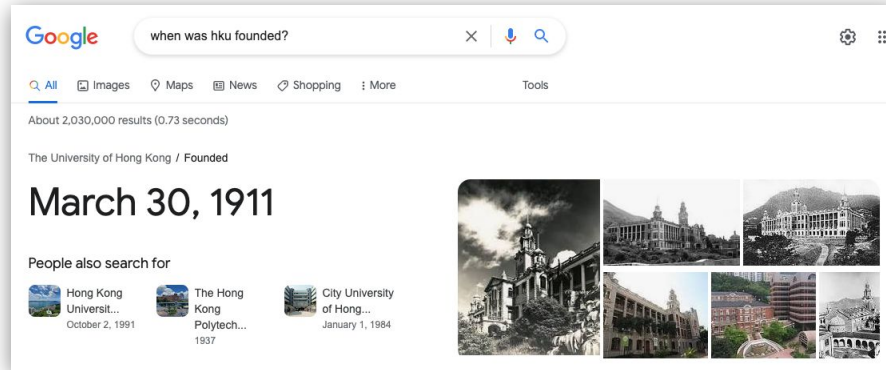
- NLP in real world applications
  - Q&A / IR - Google search





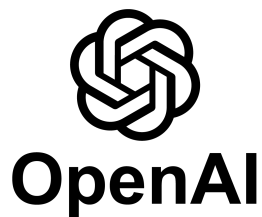
# About NLP: teaching computers human languages

- NLP in real world applications
  - Q&A / IR - Google search



# About NLP: teaching computers human languages

- NLP in real world applications
  - Q&A / IR - Google search
  - Dialogs - Apple Siri / Amazon Alexa



# About NLP: teaching computers human languages

- NLP in real world applications
  - Q&A / IR - Google search
  - Dialogs - Apple Siri / Amazon Alexa
  - Grammar checking (Grammarly), summarization, sentiment analysis ...



Welcome to the Grammarly Editor, the best place to write what's important.

Red underlines mean that Grammarly has spotted a mistake in your writing. You'll sea one if you mispell something. (See what we did there?) You'll also see an underline, if you misuse a punctuation mark. If you're worry about typos or grammatical errors that could effect your credibility, Grammarly will helps you fix those to. Click any of Grammarly's suggested corrections to apply them to your text, or open a brief explanation to learn

#### • SPELLING

sea → see

The word **sea** doesn't seem to fit this context. Consider replacing it with a different one.

[? Learn more](#)



# What ChatGPT can do?

The screenshot displays the 'Examples' page on the OpenAI website. At the top, there are navigation links for 'Overview', 'Documentation', and 'Examples'. The main heading is 'Examples', followed by the subtitle 'Explore what's possible with some example applications'. Below this is a search bar and a dropdown menu for 'All categories'. The page is filled with a grid of application examples, each with a colored icon, a title, and a brief description.

Category	Description
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 ...
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 line...
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.

# Q&A example with ChatGPT

## Q&A

Answers Generation Conversation

Answer questions based on existing knowledge.

**Prompt**

I am a highly intelligent question answering bot. If you ask me a question that is rooted in truth, I will give you the answer. If you ask me a question that is nonsense, trickery, or has no clear answer, I will respond with "Unknown".

Q: What is human life expectancy in the United States?  
A: Human life expectancy in the United States is 78 years.

Q: Who was president of the United States in 1955?  
A: Dwight D. Eisenhower was president of the United States in 1955.

Q: Which party did he belong to?  
A: He belonged to the Republican Party.

Q: What is the square root of banana?  
A: Unknown

Q: How does a telescope work?  
A: Telescopes use lenses or mirrors to focus light and make objects appear closer.

Q: Where were the 1992 Olympics held?  
A: The 1992 Olympics were held in Barcelona, Spain.


Q: How many squigs are in a bonk?  
A: Unknown

Q: Where is the Valley of Kings?  
A:

## Sample response

The Valley of Kings is located in Luxor, Egypt.

## API request

```
python  Copy

1 import os
2 import openai
3
4 openai.api_key = os.getenv("OPENAI_API_KEY")
5
6 response = openai.Completion.create(
7     engine="text-davinci-002",
8     prompt="I am a highly intelligent question answering bot",
9     temperature=0,
10    max_tokens=100,
11    top_p=1,
12    frequency_penalty=0.0,
13    presence_penalty=0.0,
14    stop=["\n"]
15 )
```

# More examples with ChatGPT



## TL;DR summarization

Transformation Generation

Summarize text by adding a 'tl;dr:' to the end of a text passage. It shows that the API understands how to perform a number of tasks with no instructions.

### Prompt

A neutron star is the collapsed core of a massive supergiant star, which had a total mass of between 10 and 25 solar masses, possibly more if the star was especially metal-rich.[1] Neutron stars are the smallest and densest stellar objects, excluding black holes and hypothetical white holes, quark stars, and strange stars.[2] Neutron stars have a radius on the order of 10 kilometres (6.2 mi) and a mass of about 1.4 solar masses. [3] They result from the supernova explosion of a massive star, combined with gravitational collapse, that compresses the core past white dwarf star density to that of atomic nuclei.

Tl;dr

### Sample response

A neutron star is the collapsed core of a massive supergiant star. These ultra-dense objects are incredibly fascinating due to their strange properties and their potential for phenomena such as extreme gravitational forces and a strong magnetic field.



## Interview questions

Generation

Create interview questions.


### Prompt

Create a list of 8 questions for my interview with a science fiction author:

### Sample response

1. What inspired you to write science fiction?
2. What themes or topics do you explore in your stories?
3. What are some of the unique elements that make science fiction stand out from other genres?
4. What challenges do you face when writing science fiction?
5. What are some of the most important aspects of a good science fiction story?
6. What do you think are the biggest challenges facing science fiction today?
7. What do you think the future of science fiction holds for readers and writers?
8. What advice would you give to aspiring science fiction authors?

# Examples with ChatGPT



## SQL request

Transformation Generation Translation

Create simple SQL queries.

**Prompt**

Create a SQL request to find all users who live in California and have over 1000 credits:

**Sample response**

```
SELECT * FROM users WHERE state='CA' AND credits > 1000;
```



## Python bug fixer

Code Generation

There's a number of ways of structuring the prompt for checking for bugs. Here we add a comment suggesting that source code is buggy, and then ask codex to generate a fixed code.

**Prompt**

```
##### Fix bugs in the below function

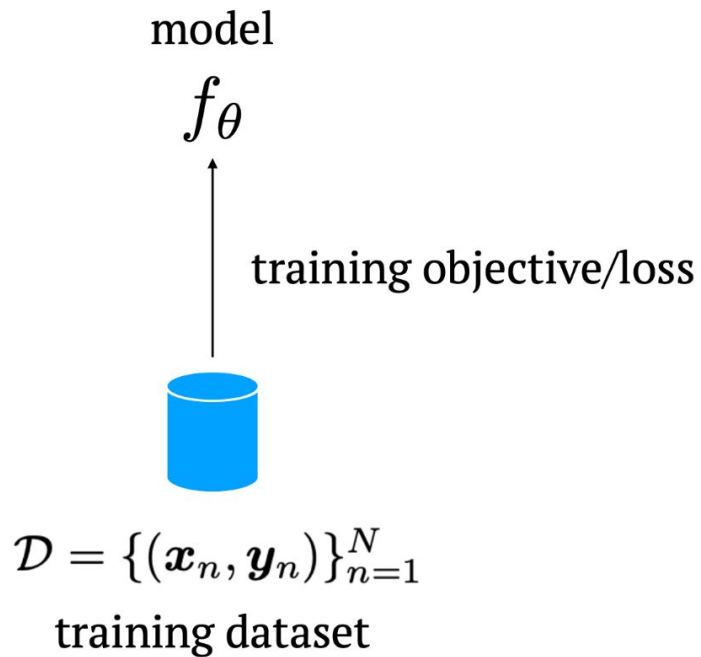
### Buggy Python
import Random
a = random.randint(1,12)
b = random.randint(1,12)
for i in range(10):
    question = "What is "+a+" x "+b+"? "
    answer = input(question)
    if answer = a*b
        print (Well done!)
    else:
        print("No.")

### Fixed Python
```

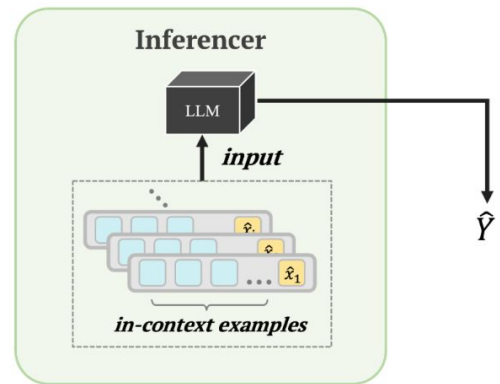
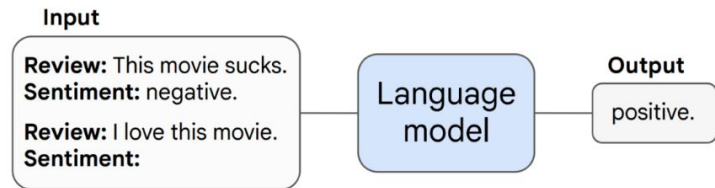
**Sample response**

```
import random
a = random.randint(1,12)
b = random.randint(1,12)
for i in range(10):
    question = "What is "+str(a)+" x "+str(b)+"? "
    answer = input(question)
    if answer == str(a*b):
        print ("Well done!")
    else:
```

# New learning paradigm: in-context learning



Supervised Learning



In-Context Learning



# Few-shot in-context learning

- **Few-shot:** In addition to the task description, the model sees **a few examples** of the task.
- **No fine-tuning,** GPT-3 doesn't update their parameters!

1	Translate English to French:	← task description
2	sea otter => loutre de mer	← examples
3	peppermint => menthe poivrée	←
4	plush girafe => girafe peluche	←
5	cheese => .....	← prompt

**Q&A**  
Answers Generation Conversation

Answer questions based on existing knowledge.

**Prompt**

I am a highly intelligent question answering bot. If you ask me a question that is rooted in truth, I will give you the answer. If you ask me a question that is nonsense, trickery, or has no clear answer, I will respond with "Unknown". ← *task description*

Q: What is human life expectancy in the United States?  
A: Human life expectancy in the United States is 78 years. ← *task examples*

Q: Who was president of the United States in 1955?  
A: Dwight D. Eisenhower was president of the United States in 1955.

Q: Which party did he belong to?  
A: He belonged to the Republican Party.

Q: What is the square root of banana?  
A: Unknown

Q: How does a telescope work?  
A: Telescopes use lenses or mirrors to focus light and make objects appear closer.

Q: Where were the 1992 Olympics held?  
A: The 1992 Olympics were held in Barcelona, Spain.

Q: How many squigs are in a bonk?  
A: Unknown

Q: Where is the Valley of Kings?  
A:

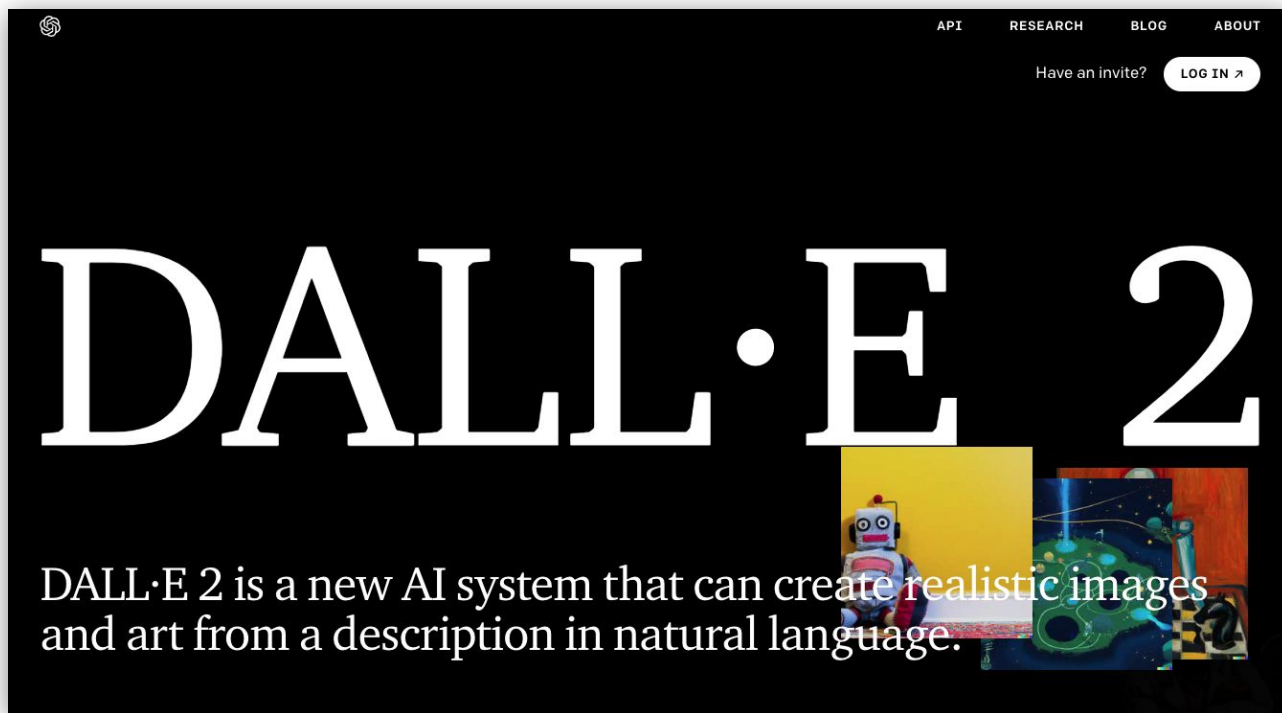
**Sample response**

The Valley of Kings is located in Luxor, Egypt. ← *GPT-3 outputs*

# About NLP: teaching computers human languages

- NLP in real world applications
  - Q&A / IR - Google search
  - Dialogs - Apple Siri / Amazon Alexa
  - Grammar checking (Grammarly), summarization, sentiment analysis ...
  - Text to images: image creation from a text description - OpenAI's DALLÉ-2

# DALLE-2 demo: text to images



The image shows the landing page for DALL·E 2. At the top left is the OpenAI logo. The top right navigation bar includes links for API, RESEARCH, BLOG, and ABOUT. Below the navigation is a "Have an invite?" prompt with a "LOG IN" button. The main heading "DALL·E 2" is displayed in large white serif font. Below the heading is a collage of three AI-generated images: a robot on a yellow background, a green alien planet, and a chess knight on a checkered board. A text block below the images reads: "DALL·E 2 is a new AI system that can create realistic images and art from a description in natural language."

API RESEARCH BLOG ABOUT

Have an invite? [LOG IN ↗](#)

# DALL·E 2

DALL·E 2 is a new AI system that can create realistic images and art from a description in natural language.

# DALLE-2 demo: text to images

## TEXT DESCRIPTION

An astronaut Teddy bears A bowl of soup

mixing sparkling chemicals as mad scientists shopping for groceries working on new AI research

as kids' crayon art on the moon in the 1980s underwater with 1990s technology



## DALL-E 2



# Language models are powerful, but they still suffer from

- Lack of interpretability
- Inconsistency
- Limited scalability
- Restricted capabilities
- ...

