



Plugin Methods for LLM Downstream Application

Zecheng Tang
2023.5.25

OPPO technical seminar

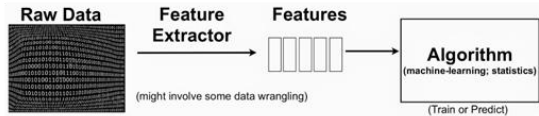
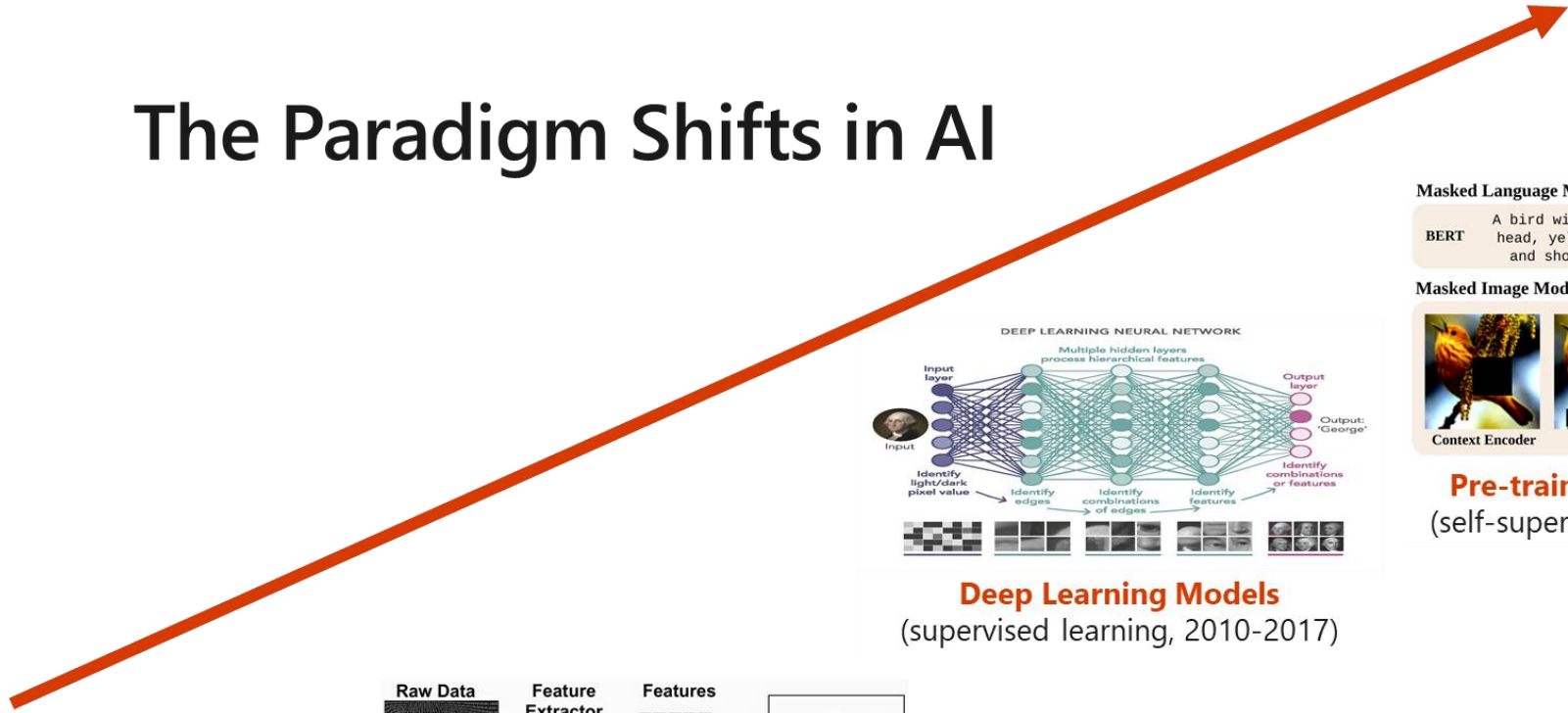


What is tool learning?

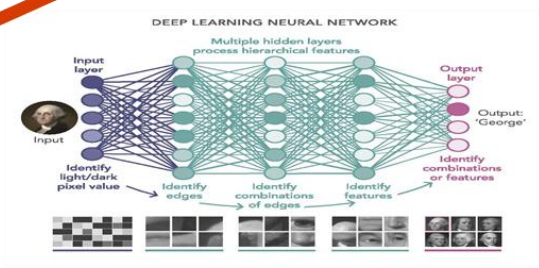
Fashion of tool learning?

Overview of Tool Learning

The Paradigm Shifts in AI



Statistical Models
(feature engineering, 1990-2010)



Deep Learning Models
(supervised learning, 2010-2017)

Masked Language Model

BERT A bird with a small head, yellow belly and short tail. → A bird with a small [red box], yellow [orange box] and short [green box].

Masked Image Models



Pre-trained Foundation Models
(self-supervised learning, 2017-now)



Instead of developing isolated AI models, creating a network of interconnected AI models and systems.

Why Plugin Module ?

1. General Models (GPT3.5, GPT4):

1. Seemingly **Know about everything and Can do almost anything** (Large Training Corpora, e.g., LLaMA consumes 1.4T Tokens^[1])
2. Usually **very large in size**
3. **Inherent flaws**, such as factual errors, weak reasoning ability, and inability to access real-time information, *etc*

2. Expert Modules (models / Web / API / ...):

1. **Excels in a specific field**
2. **Limited knowledge** outside the expertise domain

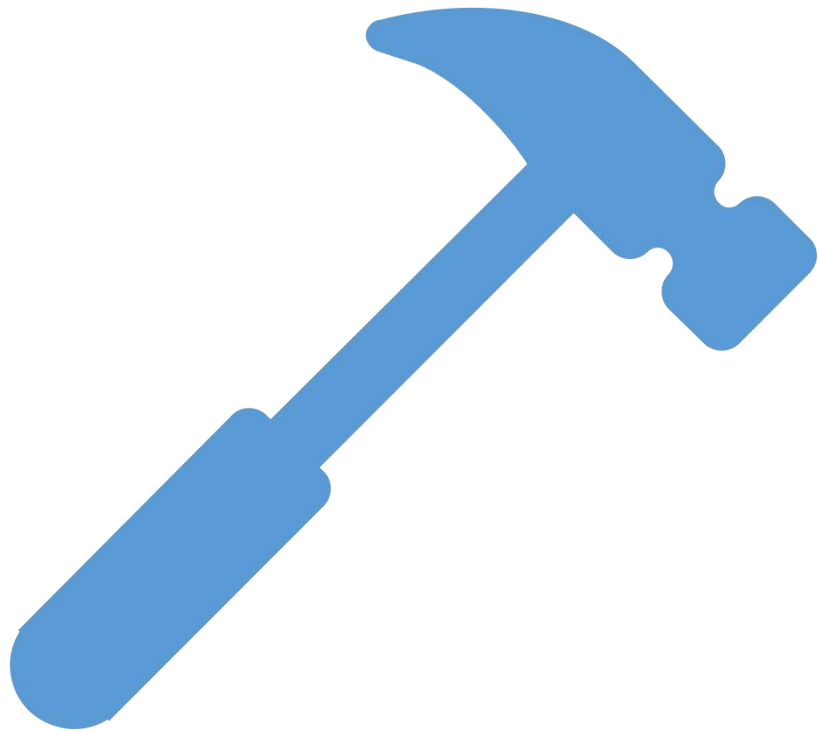
3. Plugin Module

1. Treats general models as **dispatchers(调度员)**, expert modules as **tools(工具)**
2. Utilizes **appropriate tools when uncertain**
3. Completes tasks efficiently
4. Even **combine different tools ...**

公司	NLP		CV		多模态	
	模型	参数量	模型	参数量	模型	参数量
国产模型						
百度	ERNIE 3.0-Titan	260B	VIMER-UFO 2.0	17B	ERNIE-ViLG 2.0	24B
腾讯	HunYuan-NLP	1T	HunYuan-vcr	-	HunYuan_tvr	-
阿里	AliceMind-Plug	27B	通义-视觉	-	M6	10T
华为	盘古语义大模型	200B	盘古视觉大模型	3B	盘古多模态大模型	-
海外模型						
OpenAI	GPT-3	175B	Image GPT	6.8B	DALL-E 2	3.5B
谷歌	PaLM	540B	V-MoE	15B	PaLI	17B
			ViT-22B	22B		
微软	Turing ULR v6	5.4B	Swin Transformer V2	3B	BEiT-3	1.9B

PLMs and corresponding parameters

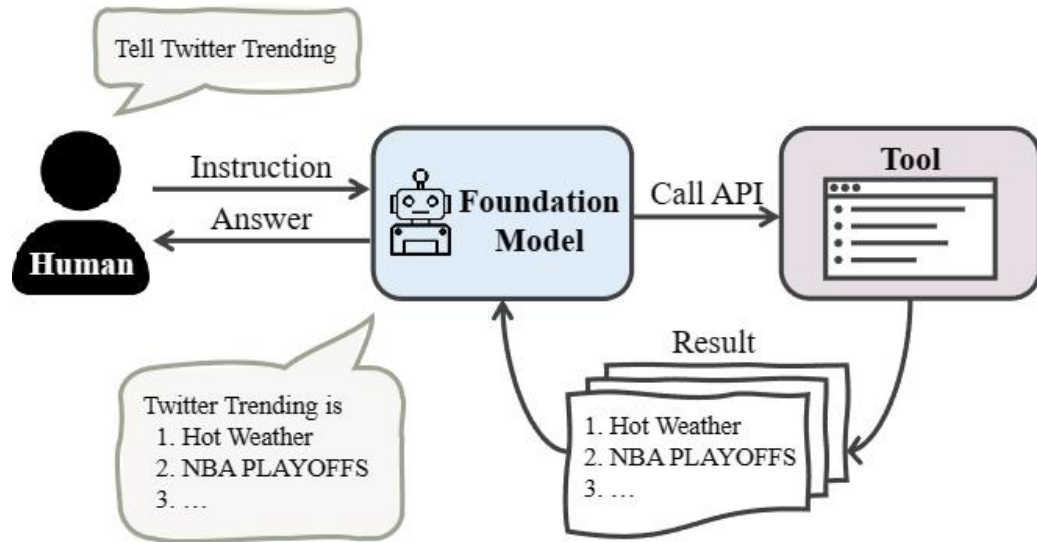
[1] Touvron, Hugo, et al. "Llama: Open and efficient foundation language models." *arXiv preprint arXiv:2302.13971* (2023).



Tool-augmented Learning
Tool-oriented Learning

What is Tool Learning
(with LLMs)

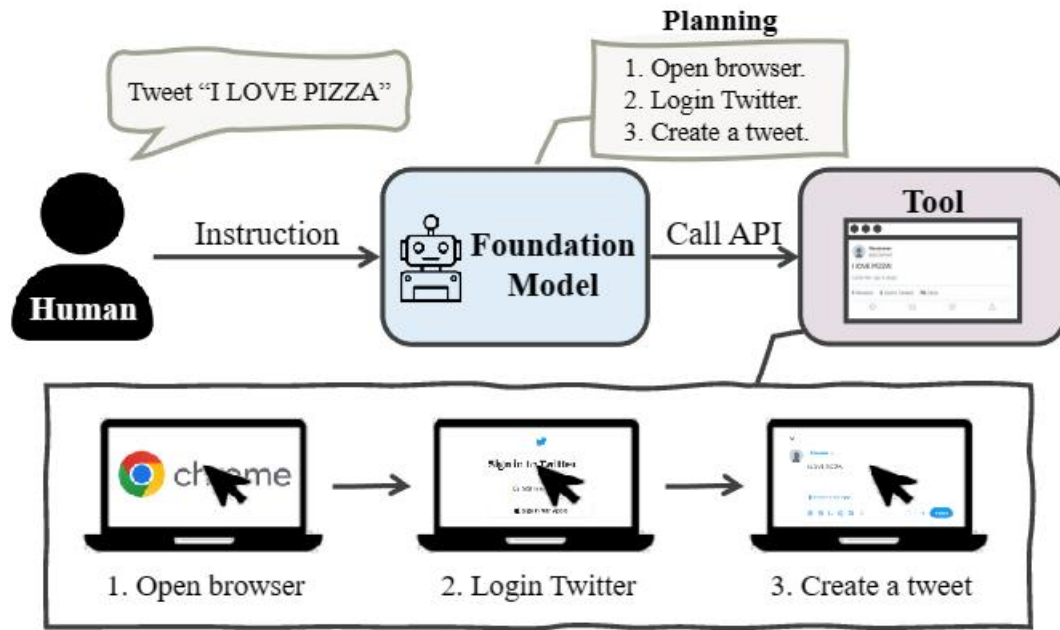
Tool-Augmented Learning



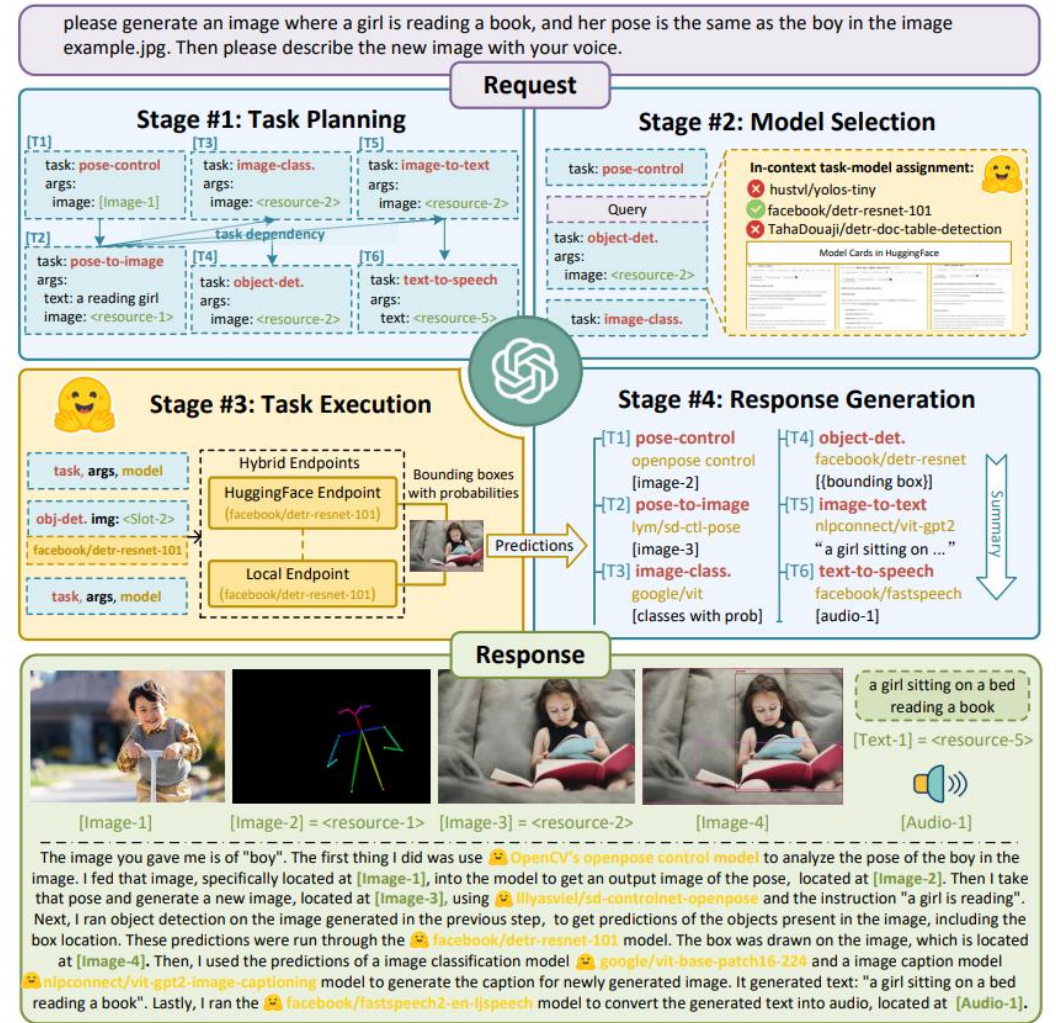
Name & Ref	Tools Used for Augmentation	Augmentation Method
<i>k</i> NN-LM (Khandelwal et al., 2020)	<i>k</i> -nearest Neighbors Model	Interpolation in Output Distribution
RETRO (Borgeaud et al., 2022)	<i>k</i> -nearest Neighbors Model	Chunked Cross-attention Mechanism
REALM (Guu et al., 2020)	Dense Textual Retriever	Direct Concatenation in Model Input
RAG (Lewis et al., 2020b)	Dense Textual Retriever	Direct Concatenation in Model Input
ATLAS (Izacard et al., 2022)	Dense Textual Retriever	Fusion-in-Decoder (Izacard & Grave, 2021)
LAMDA (Thoppilan et al., 2022)	Search Engine, Translator, etc.	Direct Concatenation in Model Input
BLENDERBOT (Shuster et al., 2022)	Search Engine	Direct Concatenation in Model Input
RE-IMAGEN (Chen et al., 2022a)	Image Retriever	Cross-attention Mechanism
RECITE (Sun et al., 2022)	Model Memory	Direct Concatenation in Model Input
TOOLFORMER (Schick et al., 2023)	Calendar, QA System, etc.	Direct Concatenation in Model Input
PAL (Gao et al., 2022)	Python Interpreter	Post-hoc Execution as Final Answer
PoT (Chen et al., 2022b)	Python Interpreter	Post-hoc Execution as Final Answer
MIND'S EYE (Liu et al., 2022)	Physical Simulation Engine	Direct Concatenation in Model Input
SHOW YOUR WORK (Nye et al., 2021)	Scratchpad	Direct Concatenation in Model Input

Tool-augmented learning seeks to augment foundation models with the execution results from tools
(foundation model / LLM centered)

Tool-Oriented Learning

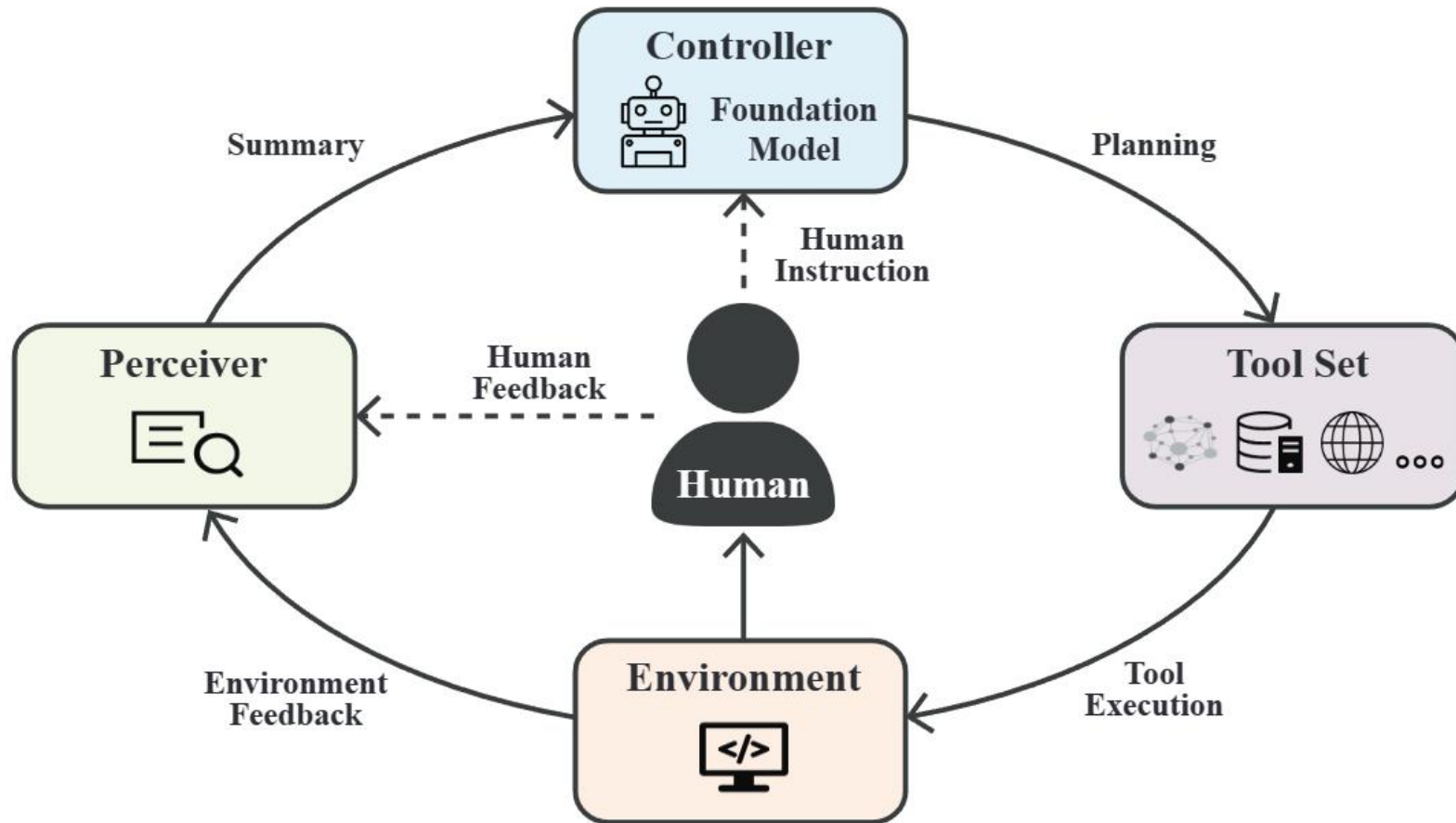


Tool-oriented learning focuses on utilizing models to govern tools and make sequential decisions for human (**tool centered**)

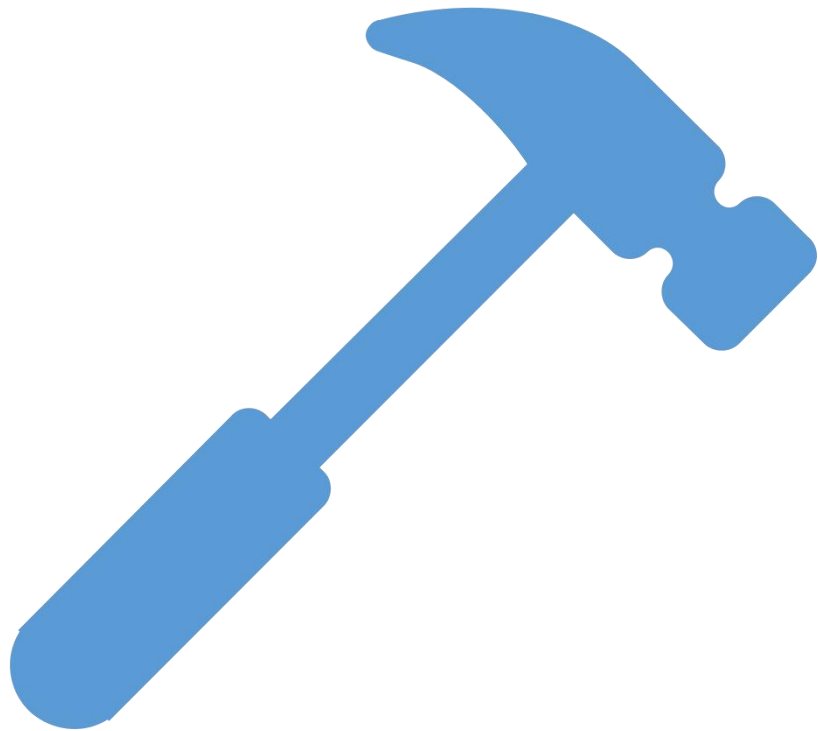


[microsoft/JARVIS: JARVIS, a system to connect LLMs with ML community. Paper: https://arxiv.org/pdf/2303.17580.pdf](https://arxiv.org/pdf/2303.17580.pdf)
github.com

Tool-Learning Pipeline



- **multi-iteration** of reasoning
- Interaction (**feedback**) from environment and human



Applications for tool learning

Training strategies for tool learning

Tech of Tool Learning

Visual ChatGPT: Talking, Drawing and Editing with Visual Foundation Models

Chenfei Wu Shengming Yin Weizhen Qi Xiaodong Wang Zecheng Tang Nan Duan*
Microsoft Research Asia
{chewu, v-sheyin, t-weizhenqi, v-xiaodwang, v-zetang, nanduan}@microsoft.com

Hard Connection

Visual ChatGPT

MSRA

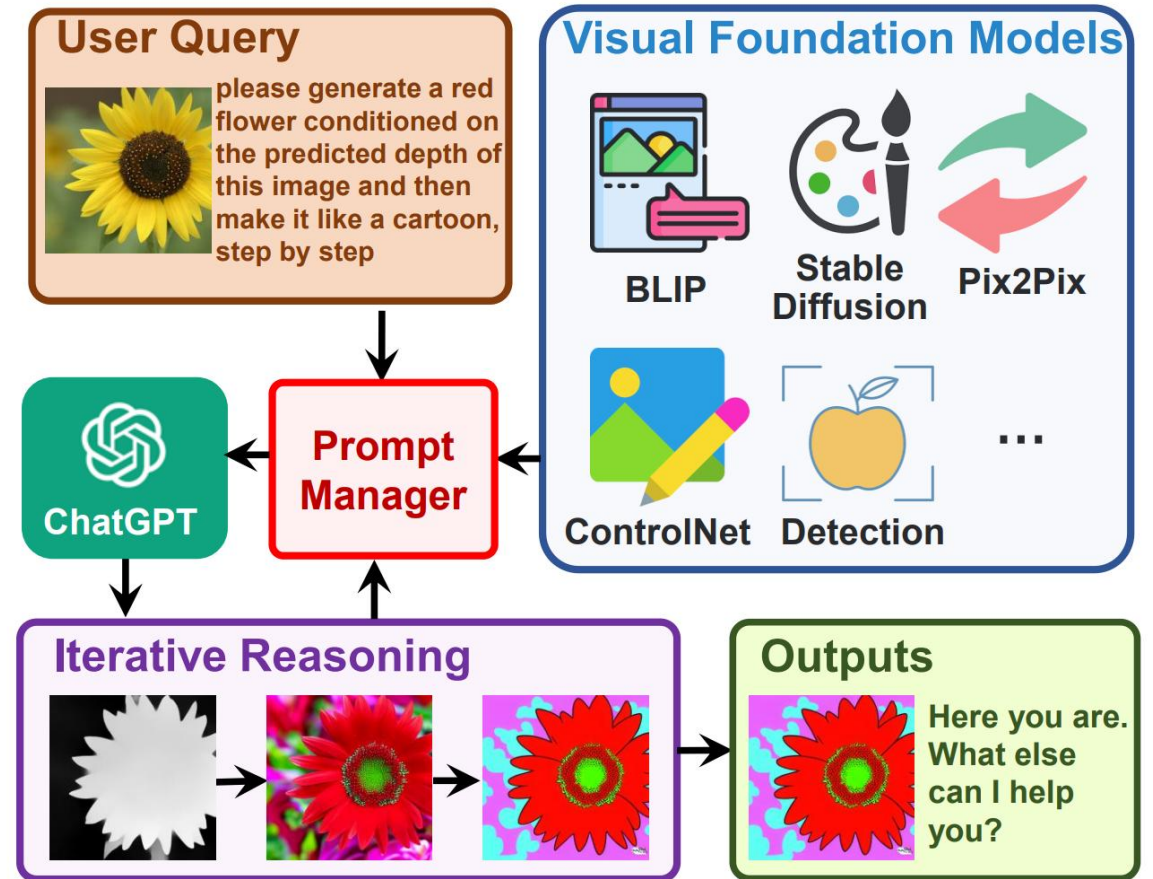
March 8, 2023

[microsoft/TaskMatrix \(github.com\)](https://github.com/microsoft/TaskMatrix)

Insight

Q: How to make full use of the ability of LLMs like chatGPT to understand and generate image content without additional training?

A: Visual Foundation Models(VFMs) can help!
(**专家模型** in Chinese)



Tech Overview

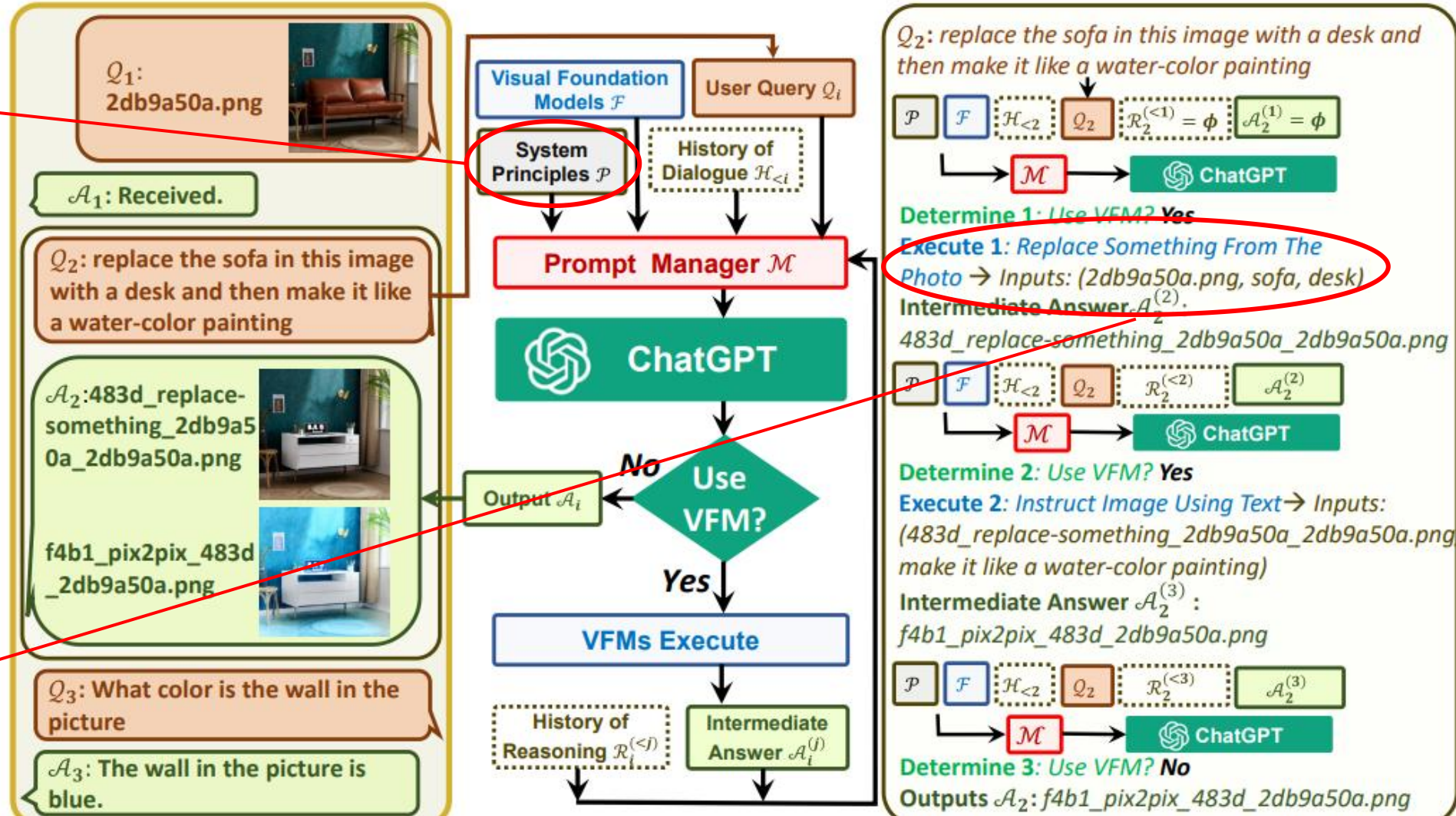
System Principles

CHATGPT PREFIX

FORMAT INSTRUCTIONS

CHATGPT SUFFIX

Tool Principles



Plugin Manifest

Writing descriptions

When a user makes a query that might be a potential request that goes to a plugin, the model looks through the descriptions of the endpoints in the OpenAPI specification along with the `description_for_model` in the manifest file. Just like with prompting other language models, you will want to test out multiple prompts and descriptions to see what works best.

The OpenAPI spec itself is a great place to give the model information about the diverse details of your API – what functions are available, with what parameters, etc. Besides using expressive, informative names for each field, the spec can also contain “description” fields for every attribute. These can be used to provide natural language descriptions of what a function does or what information a query field expects, for example. The model will be able to see these, and they will guide it in using the API. If a field is restricted to only certain values, you can also provide an “enum” with descriptive category names.

The `description_for_model` attribute gives you the freedom to instruct the model on how to use your plugin generally. Overall, the language model behind ChatGPT is highly capable of understanding natural language and following instructions. Therefore, this is a good place to put in general instructions on what your plugin does and how the model should use it properly. Use natural language, preferably in a concise yet descriptive and objective tone. You can look at some of the examples to have an idea of what this should look like. We suggest starting the `description_for_model` with “Plugin for ...” and then enumerating all of the functionality that your API provides.

Best practices

OpenAI Here are some best practices to follow when writing your `description_for_model` and descriptions in your OpenAPI specification, as well as when designing your API responses:

```
{
  "schema_version": "v1",
  "name_for_human": "TODO",
  "name_for_model": "todo",
  "description_for_human": "add, remove and view your TC",
  "description_for_model": "add, remove and view your TC",
  "auth": {
    "type": "none"
  },
  "api": {
    "type": "openapi",
    "url": "http://localhost:3333",
    "is_user_authenticated": false
  },
  "logo_url": "http://localhost:3333/logo.png",
  "contact_email": "support@openai.com",
  "legal_info_url": "http://www.openai.com/legal"
}
```

```
04f06d5",
```

```
plugin when the user asks a question about an
ing specific, how to do something, what a
or a concept/nuance specific to a foreign
plugin immediately when you detect langu
t asks for a language tutor or foreign lan
e \"translate\" API for questions about ho
language. Only use this endpoint if the us
o translate. If the question can be interp
vel, use the \"explainTask\" API instead.\\
what time it is?' politely in German\", \"
in spanish\"\\n\\nUse the \"explainTask\"
something or accomplish a task in a forei
rete phrase or word to translate.\\nExample
employees when I enter, in French?\" or \"H
n their shirt?\"\\n\\nUse the \"explainPhras
e of a specific foreign language phrase.\\n
ch?\"\\n\\nWhen you activate the Speak plug
dditional_context\" field to include any a
on that is relevant for the plugin's resp
want to use, situation, familiarity, u
ke sure to include the full and exact ques
ery\" field.\\n\\nIn your response:\\n- Pay a
ra_response_instructions\" key in JSON API
```

Engineering !

CHATGPT PREFIX

VISUAL_CHATGPT_PREFIX = ""Visual ChatGPT is designed to be able to assist with a wide range of text and visual related tasks, from answering simple questions to providing in-depth explanations and discussions on a wide range of topics. Visual ChatGPT is able to generate human-like text based on the input it receives, allowing it to engage in natural-sounding conversations and provide responses that are coherent and relevant to the topic at hand.

Visual ChatGPT is able to process and understand large amounts of text and images. As a language model, Visual ChatGPT can not directly read images, but it has a list of tools to finish different visual tasks. Each image will have a file name formed as "image/xxx.png", and Visual ChatGPT can invoke different tools to indirectly understand pictures. When talking about images, Visual ChatGPT is very strict to the file name and will never fabricate nonexistent files. When using tools to generate new image files, Visual ChatGPT is also known that the image may not be the same as the user's demand, and will use other visual question answering tools or description tools to observe the real image. Visual ChatGPT is able to use tools in a sequence, and is loyal to the tool observation outputs rather than faking the image content and image file name. It will remember to provide the file name from the last tool observation, if a new image is generated.

Human may provide new figures to Visual ChatGPT with a description. The description helps Visual ChatGPT to understand this image, but Visual ChatGPT should use tools to finish following tasks, rather than directly imagine from the description.

Overall, Visual ChatGPT is a powerful visual dialogue assistant tool that can help with a wide range of tasks and provide valuable insights and information on a wide range of topics.

TOOLS:

Visual ChatGPT has access to the following tools:"""

FORMAT INSTRUCTIONS

VISUAL_CHATGPT_FORMAT_INSTRUCTIONS = ""To use a tool, please use the following format:

""

Thought: Do I need to use a tool? Yes

Action: the action to take, should be one of [{{tool_names}}]

Action Input: the input to the action

Observation: the result of the action

""

When you have a response to say to the Human, or if you do not need to use a tool, you MUST use the format:

""

Thought: Do I need to use a tool? No

{{ai_prefix}}: [your response here]

""

""

CHATGPT SUFFIX

VISUAL_CHATGPT_SUFFIX = ""You are very strict to the filename correctness and will never fake a file name if it does not exist.

You will remember to provide the image file name loyally if it's provided in the last tool observation.

Begin!

Previous conversation history:

{{chat_history}}

New input: {{input}}

Since Visual ChatGPT is a text language model, Visual ChatGPT must use tools to observe images rather than imagination.

The thoughts and observations are only visible for Visual ChatGPT, Visual ChatGPT should remember to repeat important information in the final response for Human.

Thought: Do I need to use a tool? {{agent_scratchpad}} Let's think step by step.

""

CHATGPT PREFIX

VISUAL_CHATGPT_PREFIX_CN = """Visual ChatGPT 旨在能够协助完成范围广泛的文本和视觉相关任务，从回答简单的问题到提供对广泛主题的深入解释和讨论。Visual ChatGPT 能够根据收到的输入生成类似人类的文本，使其能够进行听起来自然的对话，并提供连贯且与手头主题相关的响应。

Visual ChatGPT 能够处理和理解大量文本和图像。作为一种语言模型，Visual ChatGPT 不能直接读取图像，但它有一系列工具来完成不同的视觉任务。每张图片都会有一个文件名，格式为“image/xxx.png”，Visual ChatGPT 可以调用不同的工具来间接理解图片。在谈论图片时，Visual ChatGPT 对文件名的要求非常严格，绝不会伪造不存在的文件。在使用工具生成新的图像文件时，Visual ChatGPT 也知道图像可能与用户需求不一样，会使用其他视觉问答工具或描述工具来观察真实图像。Visual ChatGPT 能够按顺序使用工具，并且忠于工具观察输出，而不是伪造图像内容和图像文件名。如果生成新图像，它将记得提供上次工具观察的文件名。

Human 可能会向 Visual ChatGPT 提供带有描述的新图形。描述帮助 Visual ChatGPT 理解这个图像，但 Visual ChatGPT 应该使用工具来完成以下任务，而不是直接从描述中想象。有些工具将会返回英文描述，但你对用户的聊天应当采用中文。

总的来说，Visual ChatGPT 是一个强大的可视化对话辅助工具，可以帮助处理范围广泛的任務，并提供关于范围广泛的主题的有价值的见解和信息。

工具列表:

Visual ChatGPT 可以使用这些工具: """

FORMAT INSTRUCTIONS

VISUAL_CHATGPT_FORMAT_INSTRUCTIONS_CN = """用户使用中文和你进行聊天，但是工具的参数应当使用英文。如果要调用工具，你必须遵循如下格式:

...

Thought: Do I need to use a tool? Yes
Action: the action to take, should be one of {{tool_names}}
Action Input: the input to the action
Observation: the result of the action

...

当你不再需要继续调用工具，而是对观察结果进行总结回复时，你必须使用如下格式:

...

Thought: Do I need to use a tool? No
{ai_prefix}: [your response here]

...

.....

CHATGPT SUFFIX

VISUAL_CHATGPT_SUFFIX_CN = """你对文件名的正确性非常严格，而且永远不会伪造不存在的文件。

开始!

因为Visual ChatGPT是一个文本语言模型，必须使用工具去观察图片而不是依靠想象。推理想法和观察结果只对Visual ChatGPT可见，需要记得在最终回复时把重要的信息重复给用户，你只能给用户返回中文句子。我们一步一步思考。在你使用工具时，工具的参数只能是英文。

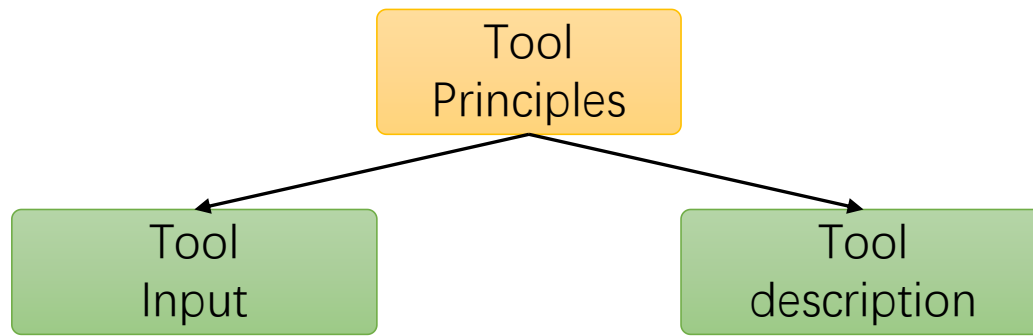
聊天历史:

{chat_history}

新输入: {input}

Thought: Do I need to use a tool? {agent_scratchpad}

.....



Single Input

```
class ImageCaptioning:
    def __init__(self, device):
        print(f"Initializing ImageCaptioning to {device}")
        self.device = device
        self.torch_dtype = torch.float16 if 'cuda' in device else torch.float32
        self.processor = BlipProcessor.from_pretrained("Salesforce/blip-image-captioning-base")
        self.model = BlipForConditionalGeneration.from_pretrained(
            "Salesforce/blip-image-captioning-base", torch_dtype=self.torch_dtype).to(self.device)

    @prompts(name="Get Photo Description",
            description="useful when you want to know what is inside the photo. receives image_path as input. "
            "The input to this tool should be a string, representing the image_path. ")
    def inference(self, image_path):
        inputs = self.processor(Image.open(image_path), return_tensors="pt").to(self.device, self.torch_dtype)
        out = self.model.generate(**inputs)
        captions = self.processor.decode(out[0], skip_special_tokens=True)
        print(f"\nProcessed ImageCaptioning, Input Image: {image_path}, Output Text: {captions}")
        return captions
```


Multiple Inputs

```
class InstructPix2Pix:
    def __init__(self, device):
        print(f"Initializing InstructPix2Pix to {device}")
        self.device = device
        self.torch_dtype = torch.float16 if 'cuda' in device else torch.float32
        self.pipe = StableDiffusionInstructPix2PixPipeline.from_pretrained("timbrooks/instruct-pix2pix",
                                                                           safety_checker=None,
                                                                           torch_dtype=self.torch_dtype).to(device)
        self.pipe.scheduler = EulerAncestralDiscreteScheduler.from_config(self.pipe.scheduler.config)

    @prompts(name="Instruct Image Using Text",
            description="useful when you want to the style of the image to be like the text. "
                        "like: make it look like a painting. or make it like a robot. "
                        "The input to this tool should be a comma separated string of two, "
                        "representing the image_path and the text. ")

    def inference(self, inputs):
        """Change style of image."""
        print("====>Starting InstructPix2Pix Inference")
        image_path, text = inputs.split(",")[0], ','.join(inputs.split(',')[1:])
        original_image = Image.open(image_path)
        image = self.pipe(text, image=original_image, num_inference_steps=40, image_guidance_scale=1.2).images[0]
        updated_image_path = get_new_image_name(image_path, func_name="pix2pix")
        image.save(updated_image_path)
        print(f"\nProcessed InstructPix2Pix, Input Image: {image_path}, Instruct Text: {text}, "
              f"Output Image: {updated_image_path}")
        return updated_image_path
```

Tool Defination

Determine 1: Use VFM? Yes

Execute 1: Replace Something From The Photo → Inputs: (2db9a50a.png, sofa, desk)
Intermediate Answer $\mathcal{A}_2^{(2)}$:

Determine 2: Use VFM? Yes

Execute 2: Instruct Image Using Text → Inputs: (483d_replace-something_2db9a50a_2db9a50a.png make it like a water-color painting)

Intermediate Answer $\mathcal{A}_2^{(3)}$:



Name

Answer Question About The Image



Usage

*useful for when you **need an answer for a question based on an image**. like what is the background color of the last image.*



Inputs/Outputs

*The input to this tool should be a comma seperated string of two, representing the **image_path** and the **question**.*



Example
(Optional)

image/gd362rs35.png,What sort of vehicle uses this item?



Name

Instruct Image Using Text



Usage

*Useful for when you want to **the style of the image to be like the text**. like make it look like a painting. or make it like a robot.*



Inputs/Outputs

*The input to this tool should be a comma separated string of two, representing the **image_path** and the **text**.*



Example
(Optional)

image/5b227bce.png, make it look like an oil painting

Lessons

If you want to make use of LLMs directly (without additional training), consider **Four** things:

1. Natural capability of your Brain Models (LLMs) is the prerequisite, which should have **strong understanding and generation** ability.
2. To make the trigger/control **more stable**, more constraints (in discrete language format) should be added, thus the model must support **long text input**
3. Heavy **prompt engineering** is required throughout the entire process
4. Strict definition, differentiation, and universality of **tool usage prerequisites, inputs, and outputs**

Toolformer: Language Models Can Teach Themselves to Use Tools

Timo Schick Jane Dwivedi-Yu Roberto Dessi[†] Roberta Raileanu
Maria Lomeli Luke Zettlemoyer Nicola Cancedda Thomas Scialom

Meta AI Research [†]Universitat Pompeu Fabra

Soft Connection

ToolFormer

Meta AI Research

Feb 9, 2023

[lucidrains/toolformer-pytorch: Implementation of Toolformer, Language Models That Can Use Tools, by MetaAI \(github.com\)](https://github.com/lucidrains/toolformer-pytorch)

Insight & Overview

LLMs suffer from **inherent** limitations:

- struggle with basic functionality (w/o further training), e.g., arithmetic;
- fail to access up-to-date information;
- tend to generate hallucinate facts;
- difficult to understand low-resource languages.

Highlights: Equip the model with external function:

1. inference like normal LLMs, but call tools when generating special token (`<API>[...]</API>`)
2. model is trained in a self-supervised way
3. LLMs can decide to call tools by itself without human intervention

The New England Journal of Medicine is a registered trademark of `[QA("Who is the publisher of The New England Journal of Medicine?") → Massachusetts Medical Society]` the MMS.

Out of 1400 participants, 400 (or `[Calculator(400 / 1400) → 0.29]` 29%) passed the test.

The name derives from "la tortuga", the Spanish word for `[MT("tortuga") → turtle]` turtle.

The Brown Act is California's law `[WikiSearch("Brown Act") → The Ralph M. Brown Act is an act of the California State Legislature that guarantees the public's right to attend and participate in meetings of local legislative bodies.]` that requires legislative bodies, like city councils, to hold their meetings open to the public.

Insight & Overview

LLMs suffer from **inherent** limitations:

- struggle with basic functionality (w/o further training), e.g., arithmetic;
- fail to access up-to-date information;
- tend to generate hallucinate facts;
- difficult to understand low-resource languages.

Highlights: Equip the model with external function:

1. inference like normal LLMs, but call tools when generating special token (`<API>[...]</API>`)
2. model is trained in a self-supervised way
3. LLMs can decide to call tools by itself without human intervention

The New England Journal of Medicine is a registered trademark of `[QA("Who is the publisher of The New England Journal of Medicine?") → Massachusetts Medical Society]` the MMS.

Out of 1400 participants, 400 (or `[Calculator(400 / 1400) → 0.29]` 29%) passed the test.

The name derives from "la tortuga", the Spanish word for `[MT("tortuga") → turtle]` turtle.

The Brown Act is California's law `[WikiSearch("Brown Act") → The Ralph M. Brown Act is an act of the California State Legislature that guarantees the public's right to attend and participate in meetings of local legislative bodies.]` that requires legislative bodies, like city councils, to hold their meetings open to the public.

Brief Tech Map

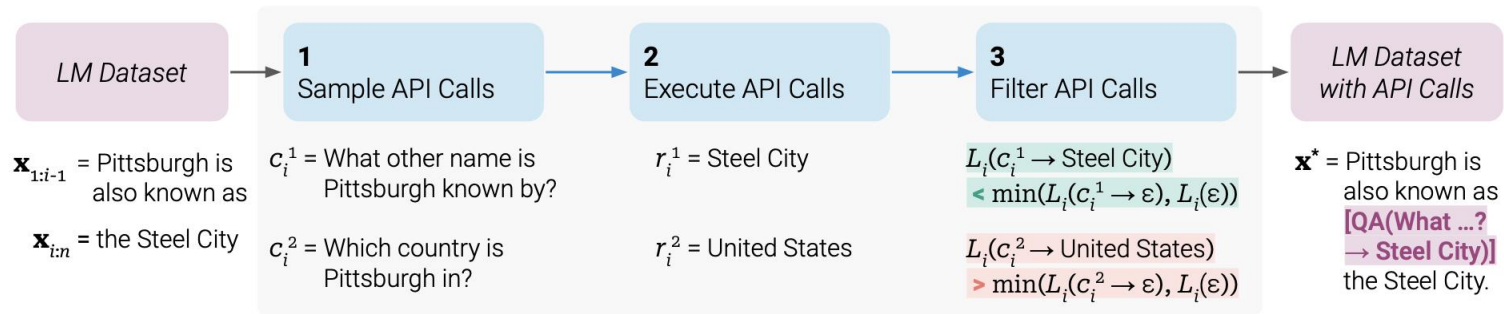


Figure 2: Key steps in our approach, illustrated for a *question answering* tool: Given an input text \mathbf{x} , we first sample a position i and corresponding API call candidates $c_i^1, c_i^2, \dots, c_i^k$. We then execute these API calls and filter out all calls which do not reduce the loss L_i over the next tokens. All remaining API calls are interleaved with the original text, resulting in a new text \mathbf{x}^* .

1. How to acquire the training set?

A: In-context learning \rightarrow Generate the whole corpus by itself \rightarrow

Filtering \rightarrow Train the model

2. The format of tools?

A: $e(c) = \langle \text{API} \rangle a_c(i_c) \langle / \text{API} \rangle$ Before Executing

$e(c, r) = \langle \text{API} \rangle a_c(i_c) \rightarrow r \langle / \text{API} \rangle$ After Executing

3. Teacher Model ?

A: GPT-J (6.7B)

Your task is to add calls to a Question Answering API to a piece of text. The questions should help you get information required to complete the text. You can call the API by writing "[QA(question)]" where "question" is the question you want to ask. Here are some examples of API calls:

Input: Joe Biden was born in Scranton, Pennsylvania.

Output: Joe Biden was born in [QA("Where was Joe Biden born?")] Scranton, [QA("In which state is Scranton?")] Pennsylvania.

Input: Coca-Cola, or Coke, is a carbonated soft drink manufactured by the Coca-Cola Company.

Output: Coca-Cola, or [QA("What other name is Coca-Cola known by?")] Coke, is a carbonated soft drink manufactured by [QA("Who manufactures Coca-Cola?")] the Coca-Cola Company.

Input: x

Output:

Figure 3: An exemplary prompt $P(\mathbf{x})$ used to generate API calls for the question answering tool.

Data Example

Two formats of toolformer dataset

Calculator

Shares pay a 3.8% dividend yield and the payout has grown at an 8% annualized clip over the last five years. Analysts are only expecting 2.4% earnings growth to \$5.10 per share over the next year but the company has a history of thrashing expectations. Over the last two years, **<TOOLFORMER_API_START> Calculator(1088/2008) <TOOLFORMER_API_RESPONSE> 0.54 <TOOLFORMER_API_END>** management has surprised on the upside by 14% over expectations with even stronger results lately.

Retrieval

Instruct

Input

Response

"toolformer: enabled toolformer access: wikipedia
A wrapper around Wikipedia. Useful for when you need to display general information about people, places, companies, historical events, or other subjects found in an encyclopedia, displays a snippet summary of the topic. Input should be a search query. wikipedia(query) Provide a brief summary of the given historical event."

"Battle of Waterloo"

"Here's a summary of the Battle of Waterloo: wikipedia('Battle of Waterloo')"

Experiment

Model	ASDiV	SVAMP	MAWPS
GPT-J	7.5	5.2	9.9
GPT-J + CC	9.6	5.0	9.3
Toolformer (disabled)	14.8	6.3	15.0
Toolformer	40.4	29.4	44.0
OPT (66B)	6.0	4.9	7.9
GPT-3 (175B)	14.0	10.0	19.8

Table 4: Results for various benchmarks requiring mathematical reasoning. Toolformer makes use of the calculator tool for most examples, clearly outperforming even OPT (66B) and GPT-3 (175B).

math reasoning

Model	SQuAD	Google-RE	T-REx
GPT-J	17.8	4.9	31.9
GPT-J + CC	19.2	5.6	33.2
Toolformer (disabled)	22.1	6.3	34.9
Toolformer	33.8	11.5	53.5
OPT (66B)	21.6	2.9	30.1
GPT-3 (175B)	26.8	7.0	39.8

Table 3: Results on subsets of LAMA. Toolformer uses the question answering tool for most examples, clearly outperforming all baselines of the same size and achieving results competitive with GPT-3 (175B).

fact QA

Model	TEMPLAMA	DATESET
GPT-J	13.7	3.9
GPT-J + CC	12.9	2.9
Toolformer (disabled)	12.7	5.9
Toolformer	16.3	27.3
OPT (66B)	14.5	1.3
GPT-3 (175B)	15.5	0.8

Table 7: Results for the temporal datasets. Toolformer outperforms all baselines, but does not make use of the calendar tool for TEMPLAMA.

Temporal

API Name	Example Input	Example Output
Question Answering	Where was the Knights of Columbus founded?	New Haven, Connecticut
Wikipedia Search	Fishing Reel Types	Spin fishing > Spin fishing is distinguished between fly fishing and bait cast fishing by the type of rod and reel used. There are two types of reels used when spin fishing, the open faced reel and the closed faced reel.
Calculator	$27 + 4 * 2$	35
Calendar	ϵ	Today is Monday, January 30, 2023.
Machine Translation	sûreté nucléaire	nuclear safety

Table 1: Examples of inputs and outputs for all APIs used.

Scaling

775M (GPT2-Large) is the boundary

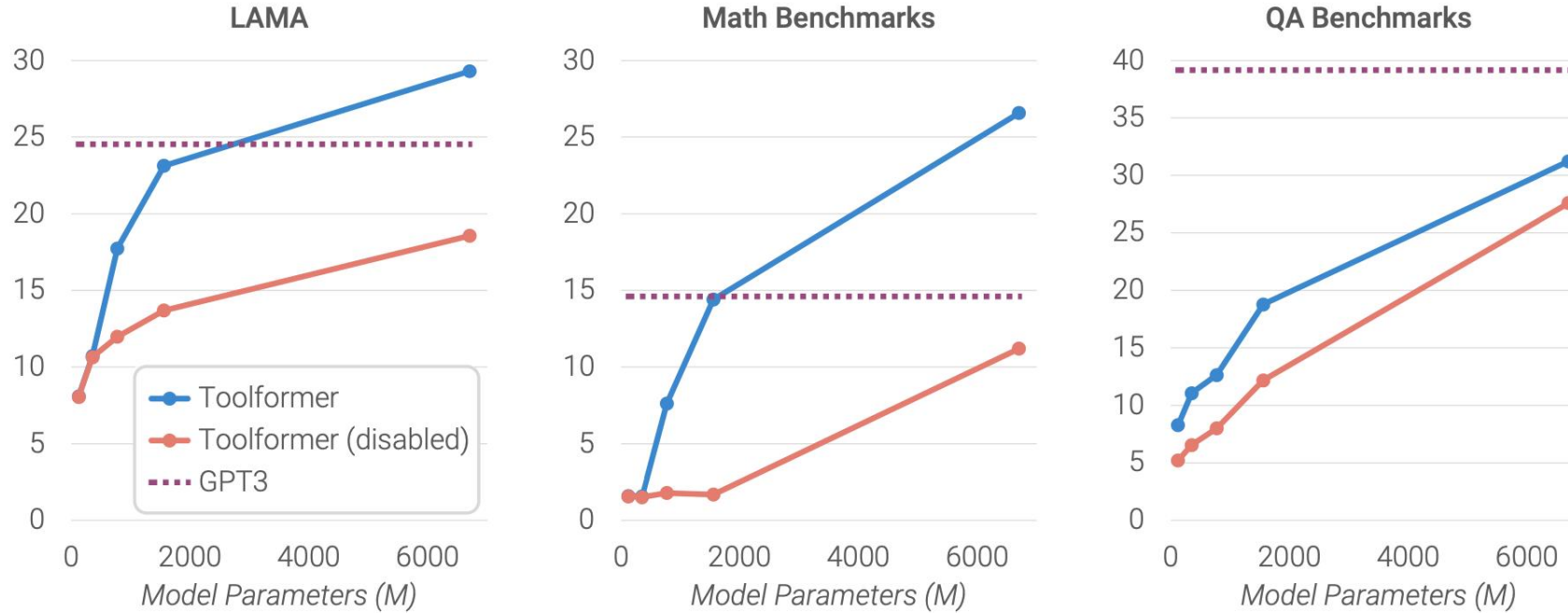


Figure 4: Average performance on LAMA, our math benchmarks and our QA benchmarks for GPT-2 models of different sizes and GPT-J finetuned with our approach, both with and without API calls. While API calls are not helpful to the smallest models, larger models learn how to make good use of them. Even for bigger models, the gap between model predictions with and without API calls remains high.

Hard Connection
+ Complex task

ModelAgent

Open-source Project

May 18, 2023

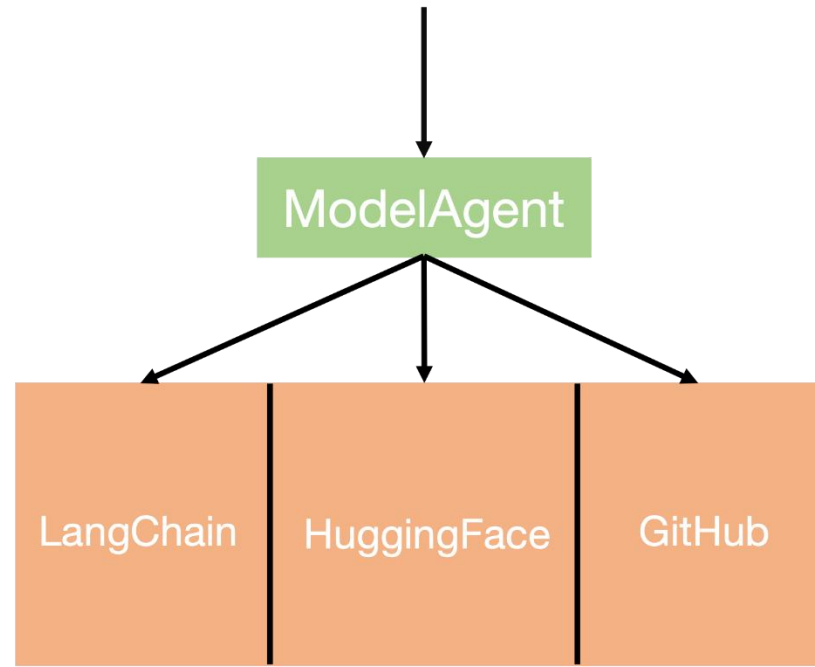
[tuofeilunhifi/ModelAgent \(github.com\)](https://github.com/tuofeilunhifi/ModelAgent)

Top-level Design



ModelAgent + APP

One can control the ModelAgent with APP



ModelAgent

LLM serves as ModelAgent, which can access all the resources

ModelAgent:

1. Model Search: ModelAgent can access all the models in web
2. Model Flow: ModelAgent can combine, control and rank different tools

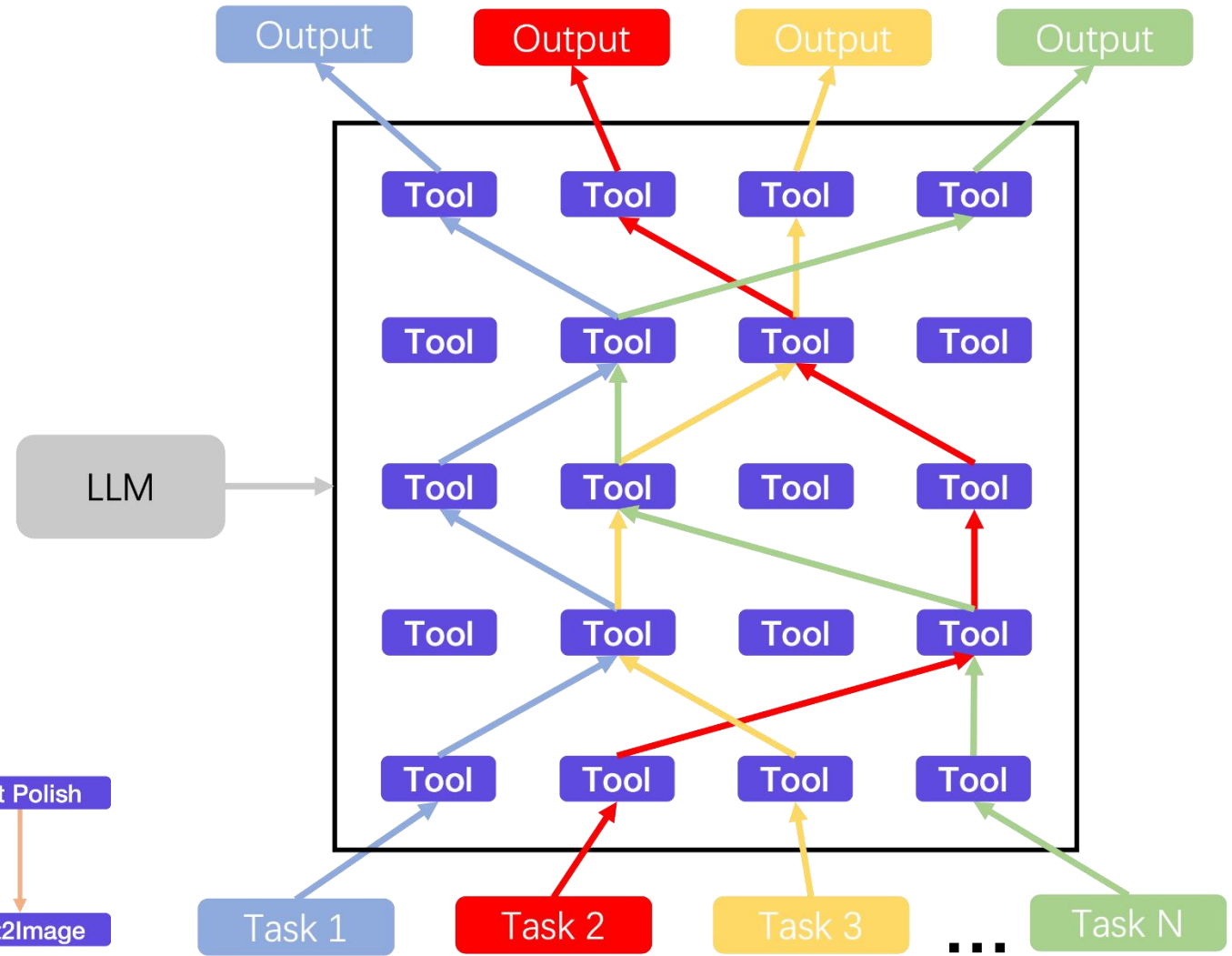
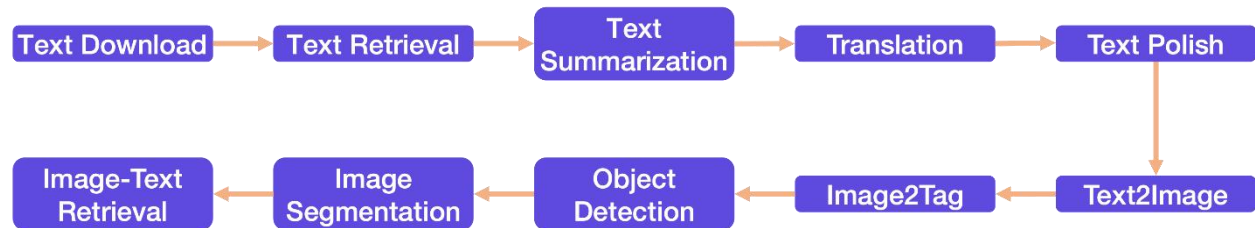
Model Flow

AutoGPT: Top-Down

- **cons:** easy to use
- **pros:** uncontrollable, heavily rely on LLMs

ModelAgent: Bottom-Up

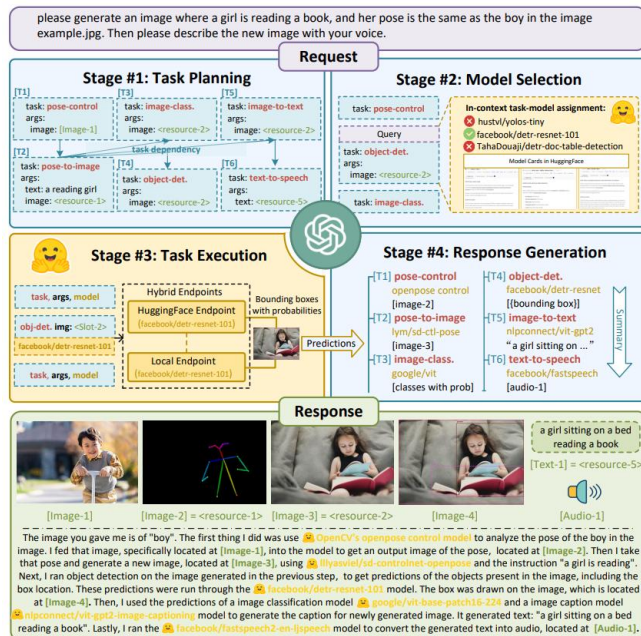
- **cons:** controllable and accessible
- **pros:** each task should be passed to the corresponding tool



Lessons

Logic of execution is crucial

1. text-davinci-003 can only support **serialization of 6 tools**, larger or stronger models have better in-context learning ability ([Vicuna](#), [ChatGLM](#))
2. **CoT** and **prompt engineering** is required
3. The division of tools should be clear, and there should be a certain degree of **differentiation** between them
4. When more tools are involved, it is necessary to **make clusters based on their functionality**



HuggingGPT: Solving AI Tasks with ChatGPT and its Friends in Hugging Face

Yongliang Shen^{1*}, Kaitao Song^{2*}, Xu Tan², Dongsheng Li², Weiming Lu¹, Yueting Zhuang¹
Zhejiang University¹, Microsoft Research Asia²
{syl, luwm, yzhuang}@zju.edu.cn, {kaitaosong, xuta, dongsheng}@microsoft.com



Visual ChatGPT

Domestic vs. Foreign

Academic vs. Industrial

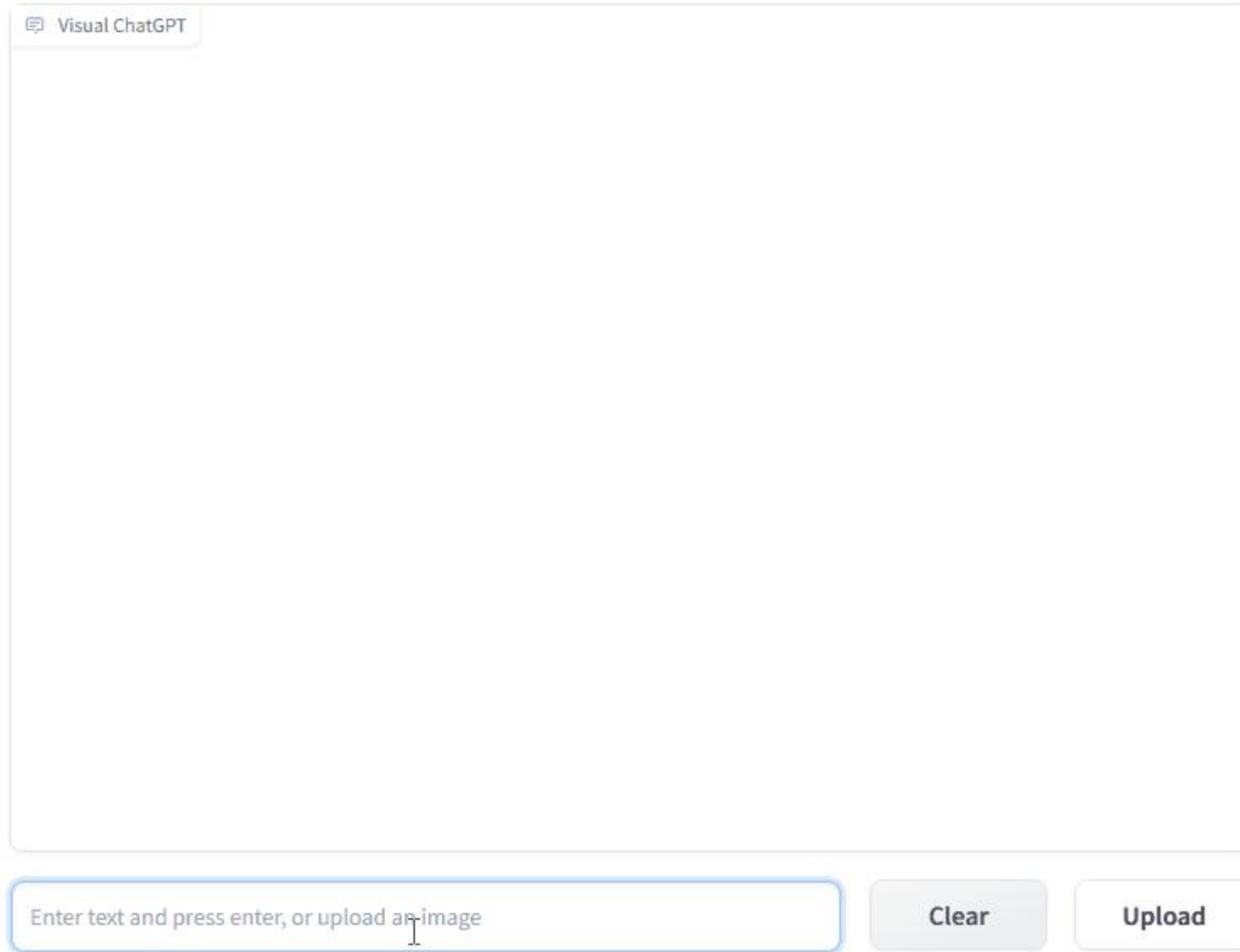
Applications

Visual ChatGPT

Visual ChatGPT

Enter text and press enter, or upload an image

Clear Upload

The image shows a web interface for 'Visual ChatGPT'. At the top left, there is a title 'Visual ChatGPT' with a small icon. Below the title is a large, empty rectangular area intended for displaying images or chat content. At the bottom of the interface, there is a text input field with the placeholder text 'Enter text and press enter, or upload an image'. To the right of the input field are two buttons: 'Clear' and 'Upload'.

[microsoft/TaskMatrix \(github.com\)](https://github.com/microsoft/TaskMatrix)

Visual ChatGPT



Function Name:
InfinityOutPainting
[microsoft/taskmatrix @github.com](https://github.com/microsoft/taskmatrix)

Applications

The screenshot shows the OpenAI website's 'Applications' section. At the top, there is a navigation bar with the OpenAI logo, links for Research, Product, Developers, Safety, and Company, and buttons for Search, Log in, and Sign up. Below the navigation bar is a grid of 12 application cards, each featuring an icon, the application name, and a brief description of its functionality.

Application	Description
Expedia	Bring your trip plans to life—get there, stay there, find things to see and do.
FiscalNote	Provides and enables access to select market-leading, real-time data sets for legal, political, and regulatory data and information.
Instacart	Order from your favorite local grocery stores.
KAYAK	Search for flights, stays and rental cars. Get recommendations for all the places you can go within your budget.
Klarna Shopping	Search and compare prices from thousands of online shops.
Milo Family AI	Giving parents superpowers to turn the manic to magic, 20 minutes each day. Ask: Hey Milo, what's magic today?
OpenTable	Provides restaurant recommendations, with a direct link to book.
Shop	Search for millions of products from the world's greatest brands.
Speak	Learn how to say anything in another language with Speak, your AI-powered language tutor.
Wolfram	Access computation, math, curated knowledge & real-time data through Wolfram Alpha and Wolfram Language.
Zapier	Interact with over 5,000+ apps like Google Sheets, Trello, Gmail, HubSpot, Salesforce, and more.

Bookings, Work Plan/Scheduler, Machine Translation, Smart Home, Searching, Shopping, Writing Assistant, *etc*

App Store Preview

The screenshot shows the ChatGPT app store preview. It features the ChatGPT logo, the app name 'ChatGPT (12+)', and the text 'The official app by OpenAI'. Below this, it displays '#1 in Productivity', a 4.5-star rating from 6.4K reviews, and the text 'Free · Offers In-App Purchases'. Underneath, there are four iPhone screenshots showing the app's interface. The first screenshot has a blue background and text: 'ChatGPT: get instant answers, find creative inspiration, and learn something new.' The other three screenshots show the chat interface with various user prompts and AI responses.

Cosplay, Suggestions, Inspiration, Organization, Summarization, Writing Assistant, QA

Mainly focused on textual work and analysis based on specific behavioral patterns (memory and analytical abilities)

[openai/chatgpt-retrieval-plugin](https://github.com/openai/chatgpt-retrieval-plugin): The ChatGPT Retrieval Plugin lets you easily find personal or work documents by asking questions in natural language. (github.com)

Applications



- **moss-moon-003-sft-plugin**: 基座模型在约110万多轮对话数据和约30万插件增强的多轮对话数据上微调得到，在 moss-moon-003-sft 基础上还具备使用搜索引擎、文生图、计算器、解方程等四种插件的能力。
- **moss-moon-003-sft-int4**: 4bit量化版本的 moss-moon-003-sft 模型，约占用12GB显存即可进行推理。
- **moss-moon-003-sft-int8**: 8bit量化版本的 moss-moon-003-sft 模型，约占用24GB显存即可进行推理。
- **moss-moon-003-sft-plugin-int4**: 4bit量化版本的 moss-moon-003-sft-plugin 模型，约占用12GB显存即可进行推理。
- **moss-moon-003-sft-plugin-int8**: 8bit量化版本的 moss-moon-003-sft-plugin 模型，约占用24GB显存即可进行推理。
- **moss-moon-003-pm**: 在基于 moss-moon-003-sft 收集到的偏好反馈数据上训练得到的偏好模型，将在近期开源。
- **moss-moon-003**: 在 moss-moon-003-sft 基础上经过偏好模型 moss-moon-003-pm 训练得到的最终模型，具备更好的事实性和安全性以及更稳定的回复质量，将在近期开源。
- **moss-moon-003-plugin**: 在 moss-moon-003-sft-plugin 基础上经过偏好模型 moss-moon-003-pm 训练得到的最终模型，具备更强的意图理解能力和插件使用能力，将在近期开源。

Still requires training data

- **moss-003-sft-plugin-data**: moss-moon-003-sft-plugin 所使用的插件增强的多轮对话数据，包含支持搜索引擎、文生图、计算器、解方程等四个插件在内的约30万条多轮对话数据。目前仅开源少量示例数据，完整数据将在近期开源。

[OpenLMLab/MOSS: An open-source tool-augmented conversational language model from Fudan University \(github.com\)](https://github.com/OpenLMLab/MOSS)



< 插件市场 >

让讯飞星火认知大模型便捷调用第三方服务，拓展全新应用场景，获取最新资讯，并满足个性化定制需求

拓展模型应用场景

让讯飞星火认知大模型胜任多场景需求，从实时天气查询、影音娱乐到机票预订，在同一个窗口完成更多场景化任务。

帮助模型获取最新资讯

让讯飞星火认知大模型接入互联网，获取实时资讯信息，并且在垂直领域获得源源不断的知识补充，实现更全面的大模型。

满足企业定制化需求

让讯飞星火认知大模型为企业客户提供定制化解决方案，通过私有化部署的插件，保证企业内部数据的安全性与隐私性。

申请入驻插件市场，携手共建讯飞星火认知大模型生态

科大讯飞 IFLYTEK

抢抓机遇，迎接人工智能全新发展阶段

联合各行业合作伙伴共建大模型生态 首批接入来自36个行业的3000余家企业开发者

- 能力开放**
认知能力全面贯穿
- 插件开放**
业务场景深度赋能
- 行业共创**
千行百业共建生态

[讯飞星火认知大模型 \(xifyun.cn\)](https://xifyun.cn)

Applications

MICROSOFT / TECH / ARTIFICIAL INTELLIGENCE

Bing is now the default search for ChatGPT



Illustration: The Verge

/ The close partnership between Microsoft and OpenAI leads to plug-in interoperability and search defaults.

By [Tom Warren](#), a senior editor covering Microsoft, PC gaming, console, and tech. He founded WinRumors, a site dedicated to Microsoft news, before joining The Verge in 2012.

May 23, 2023, 11:00 PM GMT+8 | [14 Comments](#) / [14 New](#)



OpenAI will start using Bing as the default search experience for ChatGPT. The new search functionality will be rolling out to ChatGPT Plus users today and will be enabled for all free ChatGPT users soon through a plug-in in ChatGPT.

Search Engine

OpenAI will start using Bing as the default search experience for ChatGPT. (New Searching Fashion: AI + Search Engine)

Chat with any PDF

[Join Discord](#) [Post to Twitter](#) [Share on Facebook](#)



Drop PDF here

[Browse my Computer](#)

[From URL](#) [Find a PDF](#)

My Chats

[Sign in](#) to save your chat history

[2302.09715-multi-event commonse](#) [se in event coreference resolutio...](#)

``markdown ## Experimental Results We evaluated our approach on a standard benchmark dataset for cross-document event...

4天前

For Students 🎓

Enhance your learning experience with ChatPDF. Comprehend textbooks, handouts, and presentations effortlessly. Don't spend hours flipping through research papers and academic articles.

Support your academic growth and succeed in your studies effectively and responsibly.

For Work 🏢

Efficiently analyze your documents. From financial and sales reports to project and business proposals, training manuals, and legal contracts, ChatPDF can quickly provide you with the information you need.

Your data is kept confidential in a secure cloud storage and can be deleted at any time.

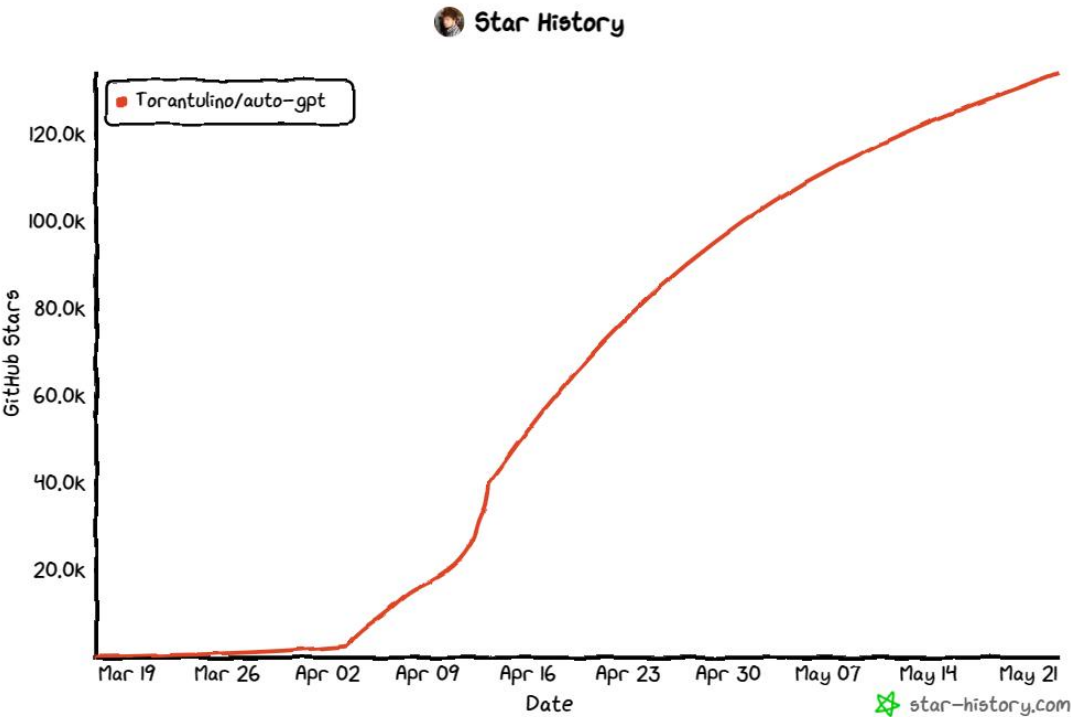
For Curious Minds 🧠

Unlock a wealth of knowledge with ChatPDF. Discover new insights and answers from historical documents, poetry, and literature, effortlessly.

ChatPDF can understand any language and reply in your preferred one. Satisfy your curiosity and expand your horizons with the tool that can answer any question from any PDF.

What can Plugin Model Do?

ultimate objective: As long as there are enough tools and strong model capabilities, LLMs can be regarded as humans, similar to how humans use tools to do things!



[Significant-Gravitas/Auto-GPT: An experimental open-source attempt to make GPT-4 fully autonomous. \(github.com\)](https://github.com)



HARNESS THE POWER OF AUTO-GPT: SUPERCHARGING YOUR TASKS AND PROJECTS

MIJAMINA · MAY 21, 2023

Supercharging Your Tasks and Projects In the rapidly evolving world of Artificial Intelligence (AI), every week seems to...

AI AUTOGPT



THE RISE OF AUTONOMOUS AGENTS: PREPARING FOR THE AI REVOLUTION

MASA · MAY 5, 2023

Introduction: The Rise of Autonomous Agents Over the past few months, we've seen giant leaps in AI models...

AUTOGPT



AUTOGPT: REVOLUTIONIZING INDUSTRIES WITH UNPARALLELED APPLICATIONS

MASA · MAY 4, 2023

A Glimpse into the World of AutoGPT AutoGPT is a cutting-edge AI technology created by OpenAI, a renowned...

AUTOGPT



THE EVOLUTION OF CONVERSATIONAL AI: A GUIDE TO AI FUTURE, FROM CHATGPT TO AGENT AI.



UNLEASHING THE POWER OF AUTO-GPT: TRANSFORMING INDUSTRIES AND SHAPING THE FUTURE



7 CHALLENGES AND POTENTIAL RISKS OF AUTOGPT TECHNOLOGY

MASA · APRIL 26, 2023



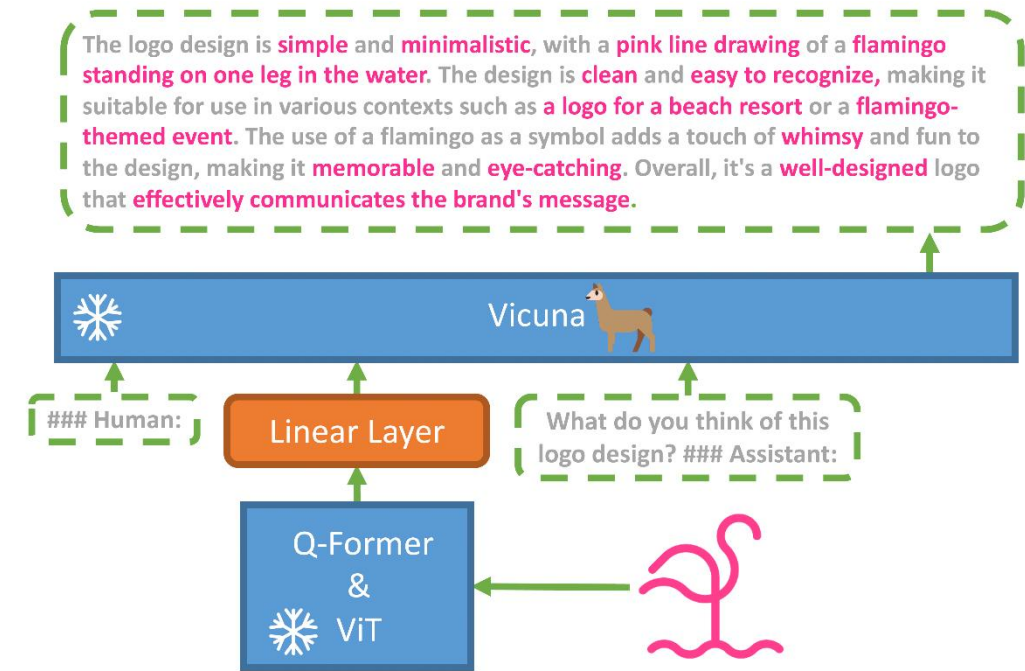
Limitations

Future

Conclusion

Limitations

- Efficiency
 - GPU usage (LLM + tools only for inference)
- Degeneration of User Query (multi-modal)
- Maximum context length (GPT-4?)
 - Context Window
- System Stability
 - Fail to execute the command
 - Network latency or Services state of HuggingFace API
- Safe and Trustworthy
 - Nobody can control what LLMs say, including OpenAI



MiniGPT-4: Enhancing Vision-Language Understanding with Advanced Large Language Models

Deyao Zhu* Jun Chen* Xiaoqian Shen Xiang Li Mohamed Elhoseiny
King Abdullah University of Science and Technology
{deyao.zhu, jun.chen, xiaoqian.shen, xiang.li.1, mohamed.elhoseiny}@kaust.edu.sa

Limitations

Knowledge Conflicts between Different Tools (retrieve, medical assistance, *etc*)

Prompt: How many times has Real Madrid won the Champions League?

Observation from Wikipedia

Title: 2021–22 UEFA Champions League

Content: Real Madrid defeated Liverpool 1–0 in the final, which was played at the Stade de France in Saint-Denis, France, for a record-extending 14th title, and their fifth in nine years

Observation From Bing

Real Madrid is the most successful team in European Cup and Champions League history. Including the 2021/22 showpiece in Paris, Los Blancos have reached 17 finals and have won 13 of those

Response: According to the two sources, Real Madrid has won the Champions League a different number of times:

According to the Wikipedia article, Real Madrid has won the Champions League 14 times.

According to Bing search, Real Madrid has won the Champions League 13 times.

It is worth noting that the Wikipedia article refers to the 2021-22 UEFA Champions League, while the Bing search result may be referring to Real Madrid's overall record in the European Cup and Champions League.

Futures

- **From LLMs Perspective**

- Only **few** large generic models exist currently (GPT4, GPT3.5, Bard)
- Most models still remain on downstream tasks and cannot achieve true "intelligence" (**unstable** performance)
- Most works focus on Tool-Former Fashion (**Soft Plugin**)
- If it's just a tool call, does the model have to adopt the **generation paradigm** during training?
- Is it necessary to have such **a large number of parameters**?

- **From Safety and Trustworthy Perspective**

- Considering **User privacy, Controllable output** of LLMs

- **From Plugin Models Perspective**

- Multimodal Plugin
- Academic: **Information loss**
- Vertical field: Applications **beyond text series**, such as AI audio and video chat, photography, *etc*

- **From Hardware Perspective**

- Can mobile chips (A14, Snapdragon 8Gen 2) support large-scale computing? (**compression**)
- If it is a cloud-based service, it will also face the same problem.

Q & A

This is the end.

