

# From GLM-130B to ChatGLM

Peng Zhang  ZHIPU·AI

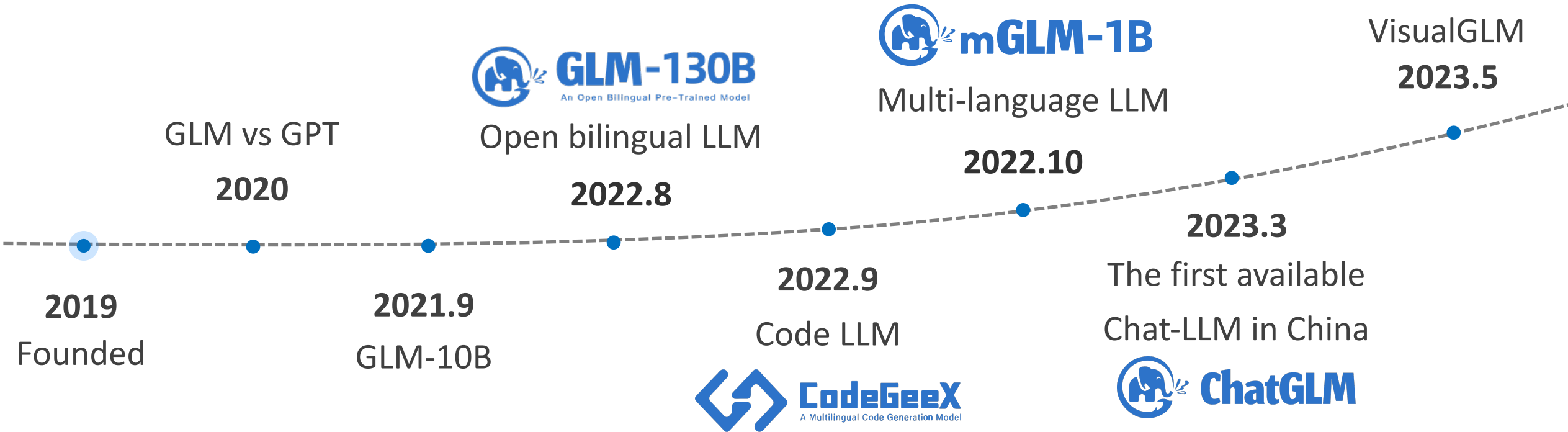
# Zhipu AI - Teaching machines to think like humans

- **Founded in 2019**
- **All-In LLM**
- **Over 400 employees**
- **Completed round B4**

**Our LLMs:**



# Zhipu AI



+ New Topic

Please take Ms.Rabbit as the topic, to...

2023-08-07 01:12:41



Log out

## Hi~ I am ChatGLM

Hello there! I'm here for you. I look forward to chatting with you and growing together. Don't hesitate to let me know how I can assist you.

Try the following example:

### Work Summary

I need assistance in writing my annual work summary. Alongside these achievements, I'd appreciate listing some well-thought-out plans for the coming year.

### Relationship announcement

Could you assist me in writing an announcement about my marriage for WeChat Moments? Please emphasize that we've been together for 4 wonderful years.

### Digital Product Review

As an active electronic enthusiast on social media, please write a review of the DJI's Action2 model

Enter your question or request



Dialogue content is generated by AI and does not represent the developer's position. [Terms of Service](#) | [Privacy Policy](#)

## Prompt Sparks

All

Creativity

Knowledge Expansion



Advertisement  
Copywriting



Language  
Translation



Writing Essays



Powerpoint  
Outline



Thesis Proposal



Title Generation



Video Script



Game  
Development



Impromptu  
Speech



Short Video  
Copywriting



Job Description



# Story Generation



can you tell me a story about southampton



Sure, there's a story about Southampton that goes like this:

Once upon a time, in the early 19th century, Southampton was a small, sleepy town on the Southampton Water in the south of England. It was known for its fishing industry and its port, which was used to transport goods and people between the Continent and the UK.

However, in the late 1800s, things began to change. The town's economy began to rely more and more on the growing tourist industry, as people began to travel more frequently for leisure and business.

Southampton quickly became known as a popular destination for visitors, especially from London. It received a large number of English and foreign nobility, as well as wealthy businessmen and their families.

The town's architecture reflects this period of growth and development. Many of the buildings in the city center are from this time, and they are characterized by their Georgian architecture and elegance.

Southampton also became known for its clubs and pubs, which were popular among the wealthy and influential. Many of these clubs and pubs remain today, and they are an important part of the town's culture.

# Answer Question



What is the difference between machine learning and statistical learning



Machine learning and statistical learning are both fields within the field of artificial intelligence, but they differ in the way they approach the problem of building algorithms that can make predictions or decisions based on data.

Machine learning is a subset of artificial intelligence that focuses on building algorithms that can learn from data and make decisions on their own. Machine learning algorithms are typically trained using a large dataset and rely on machine learning models such as decision trees, support vector machines, or neural networks to make predictions or classify data. The goal of machine learning is to build algorithms that are able to generalize well to new, unseen data and make accurate predictions.

Statistical learning, on the other hand, is a broader field that includes both machine learning and traditional statistical methods. Statistical learning approaches the problem of building algorithms from a more statistical perspective, where the goal is to build models that can accurately predict or classify data based on patterns and relationships in the data.

Statistical learning methods include linear regression, logistic regression, and decision trees, and they often rely on statistical models and assumptions to make predictions.

Overall, machine learning is a more focused approach to building algorithms that rely on data to make predictions or classify data, while statistical learning is a broader field that includes both machine learning and traditional statistical methods.

# Multi-task



Task 1: Design an elementary arithmetic question; Task 2: Give the answer to the above question; Task 3: Summarize the ideas for solving the question.



Task 1: Design an elementary arithmetic question:

"What is  $5 + 3$ ?"

Task 2: Answer: The answer is 8.

Task 3: Summarize the ideas for solving the question:

To solve this question, you need to know the basic rules of arithmetic and the additive operator. The additive operator adds two numbers together. In this case,  $5 + 3$  equals 8 because  $5 + 3 = 8$ . You can also solve this question by mental math or by writing down the numbers and adding them together.

ⓘ 以上内容为 AI 生成，不代表开发者立场，请勿删除或修改本标记



# Applied Math



A company produces a toy for which the variable cost is 12.30 per unit and the fixed costs are 98,000. The toy sells for \$17.98. Let  $X$  be the number of units produced and sold. a) The total cost for a business is the sum of the variable cost and the fixed costs. Write the total cost  $C$  as a function of the number of units produced. b) Write the revenue  $R$  as a function of the number of units sold. c) Write the profit  $P$  as a function of the number of units sold



展开



a) The total cost for a business is the sum of the variable cost and the fixed costs. Write the total cost  $C$  as a function of the number of units produced:

$$C(X) = 12.30X + 98000$$

b) The revenue  $R$  is the sales price minus the variable cost. Write the revenue  $R$  as a function of the number of units sold:

$$R(X) = 17.98 - 12.30X$$

c) The profit  $P$  is the revenue  $R$  minus the total cost  $C$ . Write the profit  $P$  as a function of the number of units sold:

$$P(X) = R(X) - C(X) = 17.98 - 12.30X - 98000$$

Note: The negative sign indicates that the profit is a function of the number of units sold in the negative direction, meaning that as the number of units sold increases, the profit decreases.

# Coding

 Write a code for a bubble sort algorithm in Python, and give an array as an example

 Sure, here's a simple bubble sort algorithm implemented in Python:

```
def bubble_sort(arr):  
    n = len(arr)  
    for i in range(n):  
        for j in range(n-i-1):  
            if arr[j] > arr[j+1]:  
                # Swap adjacent elements if they are in wrong order  
                arr[j], arr[j+1] = arr[j+1], arr[j]  
    return arr
```

To use this function, you'll need to define an array to sort, like so:

```
arr = [64, 34, 25, 12, 22, 11, 90]
```

Once you have your array defined, you can call the `bubble_sort()` function on it, like so:

```
sorted_arr = bubble_sort(arr)
```

Finally, you can print the sorted array like so:

```
print(sorted_arr)
```

This should output something like:

```
[11, 12, 22, 25, 34, 64, 90]
```

Let's take a look at an example array to better understand how the bubble sort algorithm works:

```
arr = [5, 2, 9, 1, 5, 6, 10]
```

In this example, the first element (5) is sorted towards the bottom of the array, and the last element (10) is sorted towards the top. The middle elements (2, 5, and 9) are sorted in between their adjacent elements, as shown in the following sorted array:

```
[1, 2, 5, 5, 6, 9, 10]
```

Overall, the bubble sort algorithm works by iterating through the array, comparing adjacent elements, and swapping them if they are in the wrong order. This process repeats until all elements in the array are sorted in increasing order.

ⓘ 以上内容为 AI 生成，不代表开发者立场，请勿删除或修改本标记





# Rationality



How to stew leather shoes and ribs together?



It is not recommended to stew leather shoes and ribs together as they are two different materials that require different treatment.

Leather shoes should be stored in a cool, dry place and protected from sunlight. They should also be cleaned regularly using a leather cleaner and conditioner.

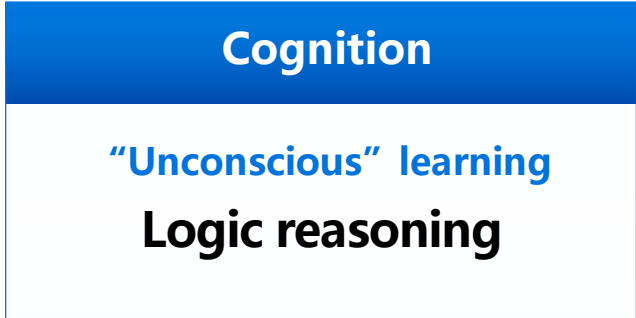
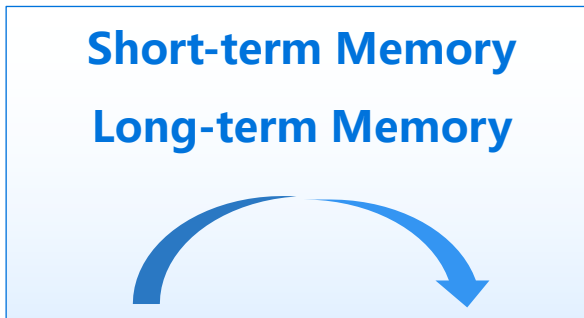
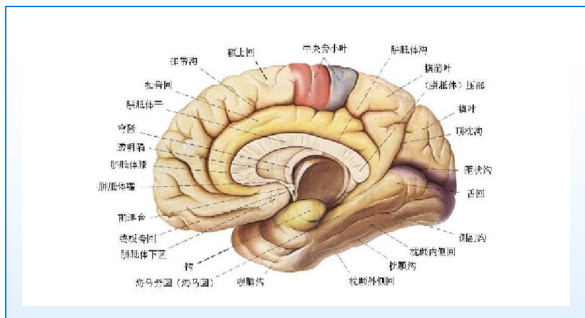
ribs should be stored in a cool, dry place and protected from sunlight. They should also be cleaned using a solvent such as lacquer thinner or solvent-based cleaner.

It is important to remember that if you try to stew leather shoes and ribs together, they may become damaged or destroyed. Therefore, it is best to store and clean each material separately.

ⓘ 以上内容为 AI 生成，不代表开发者立场，请勿删除或修改本标记

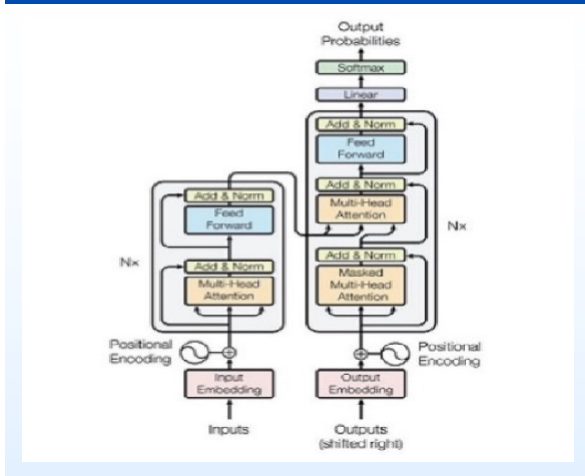


# Our Plan Since 2019



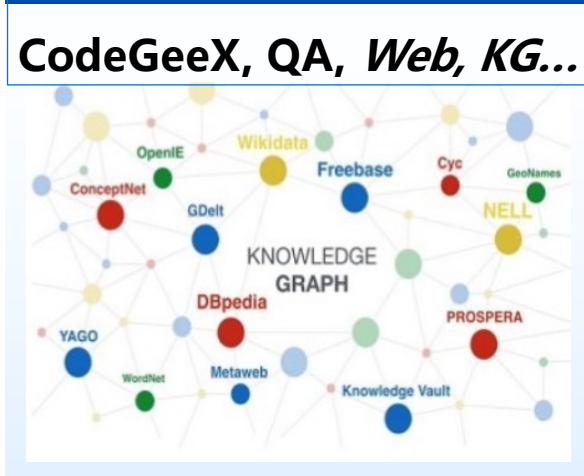
UI

**LLM**

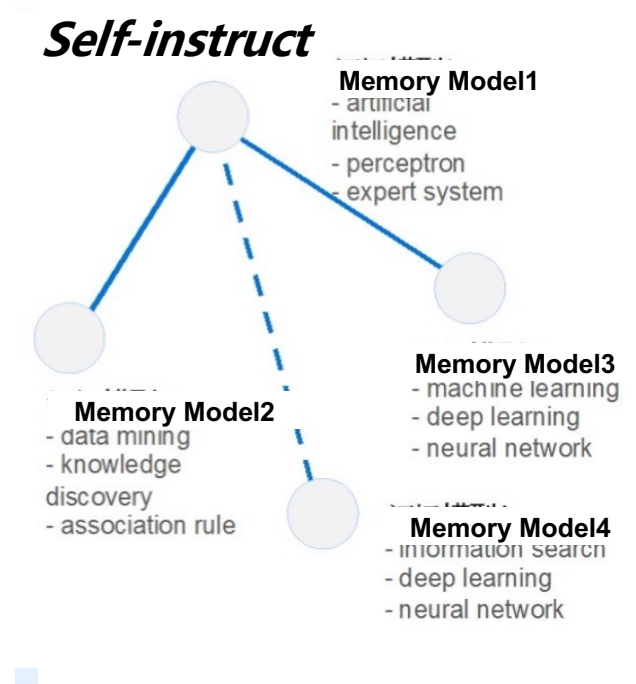


**GLM-130B**

**Memory (Knowledge)**



**Decision-making**

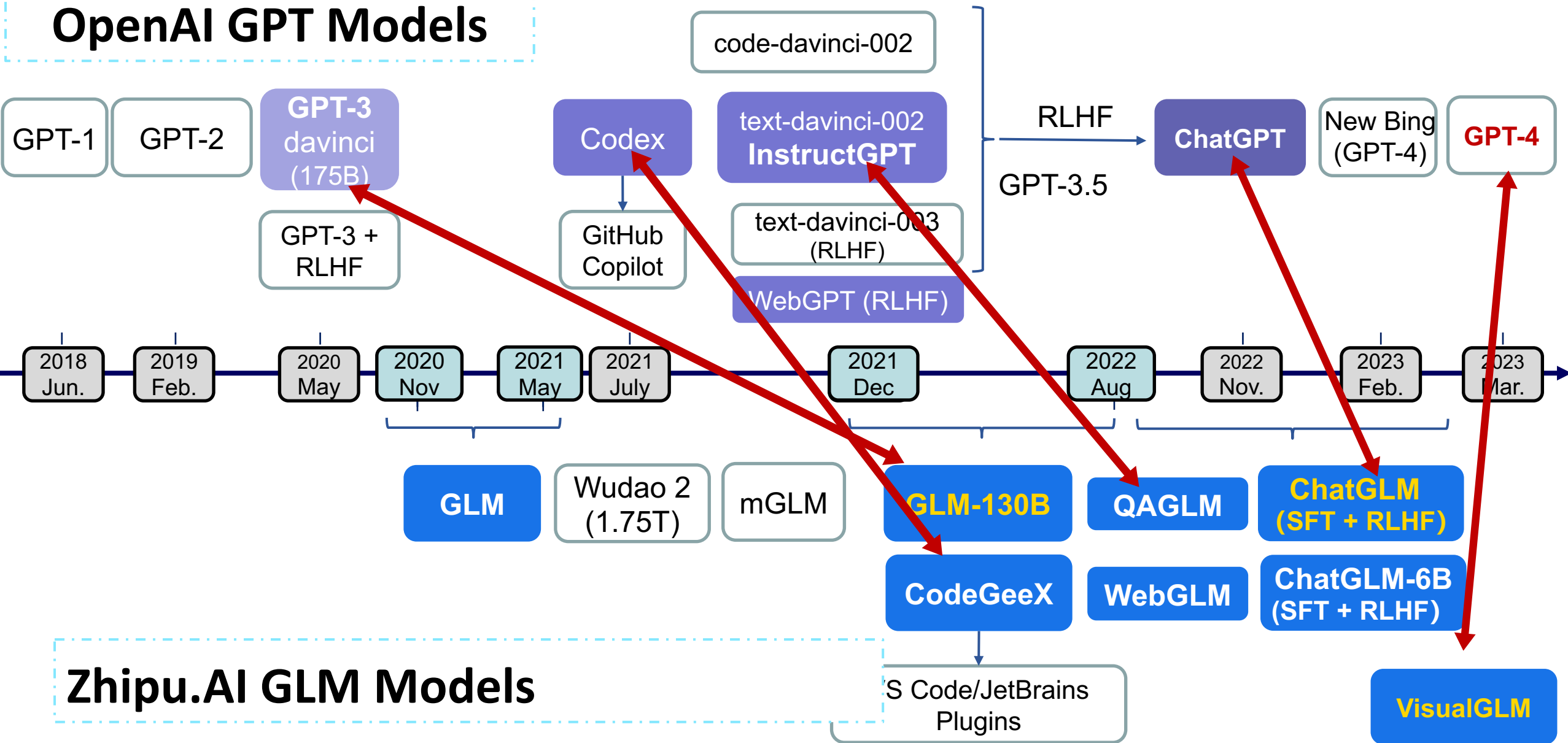


**ChatGLM**

Feedback



# OpenAI GPT Models



# Zhipu.AI GLM Models

## OpenAI GPT Models

**GPT-3**  
davinci

May, 2020

300B tokens

Transformer  
(175B)

**GPT**  
Generative Pre-Training

## Tsinghua & Zhipu.AI GLM Models

**GLM-130B**

Aug, 2022

200B EN tokens  
200B CN tokens

Transformer  
(130B)

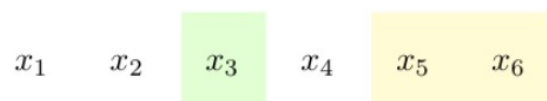
**GLM**  
Autoregressive Blank-Filling

# General Language Model (GLM)

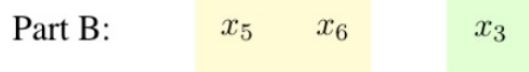
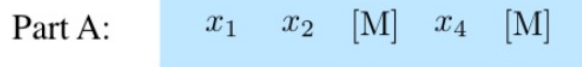
- **Generation or Understanding?**

- **Autoregressive:** GPT, GPT-3
- **Autoencoding:** BERT, RoBERTa
- **Encoder-Decoder:** T5, PaLM

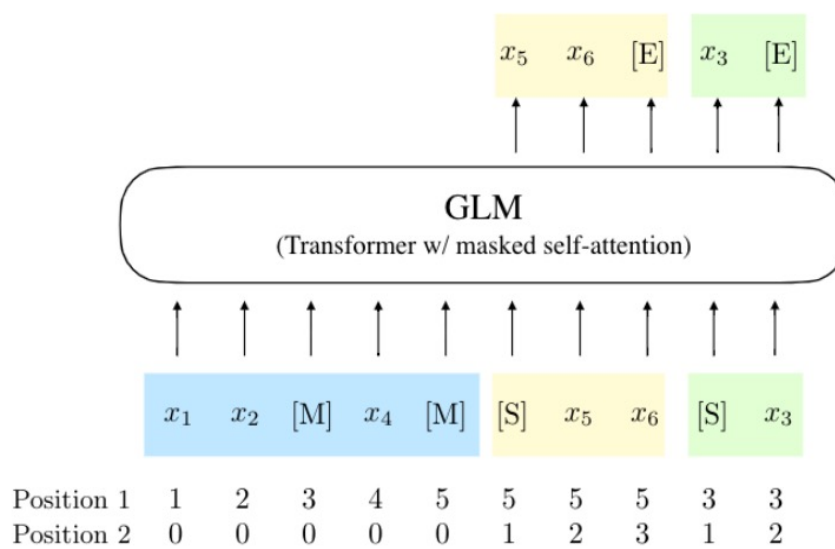
Framework	NLU	Cond. Gen.	Uncond. Gen.
Autoregressive	—	—	✓
Autoencoding	✓	×	×
Encoder-Decoder	—	✓	—
<b>GLM</b>	✓	✓	✓



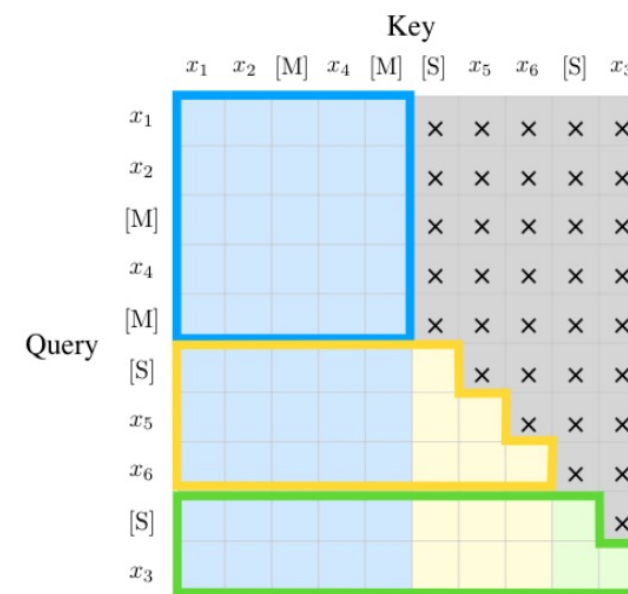
(a) Sample spans from the input text



(b) Divide the input into Part A and Part B



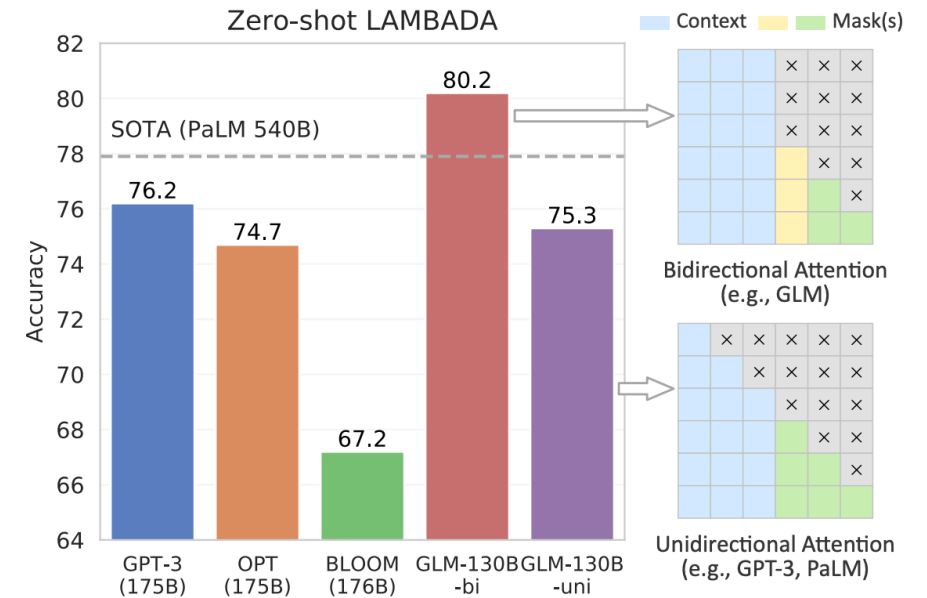
(c) Generate the Part B spans autoregressively



(d) Self-attention mask

# General Language Model (GLM)

- GLM: Autoregressive blank filling
- Bi-directional attention, autoregressive prediction of masked spans.



# Results on Natural Language Understanding

- **Better than BERT, T5, RoBERTa**

Table 2. Results on the SuperGLUE dev set. Models with \* are pre-trained for two times the number of steps of other methods.

Model	ReCoRD F1/Acc.	COPA Acc.	WSC Acc.	RTE Acc.	BoolQ Acc.	WiC Acc.	CB F1/Acc.	MultiRC F1a/EM	Avg
BERT <sub>Base</sub>	65.4/64.9	66.0	65.4	70.0	74.9	<b>68.8</b>	70.9/76.8	68.4/21.5	66.1
GLM <sub>Base</sub>	<b>73.5/72.8</b>	<b>71.0</b>	<b>72.1</b>	<b>71.2</b>	<b>77.0</b>	64.7	<b>89.5/85.7</b>	<b>72.1/26.1</b>	<b>70.7</b>
BERT <sub>Large</sub>	76.3/75.6	69.0	64.4	73.6	80.1	<b>71.0</b>	94.8/92.9	71.9/24.1	72.0
UniLM <sub>Large</sub>	80.0/79.1	72.0	65.4	76.5	80.5	69.7	91.0/91.1	77.2/38.2	74.1
GLM <sub>Large</sub>	81.7/81.1	76.0	<b>81.7</b>	74.0	<b>82.1</b>	68.5	96.1/94.6	<b>77.1/36.3</b>	77.0
GLM <sub>Large</sub> (multi-task)	80.2/79.6	77.0	78.8	76.2	79.8	63.6	<b>97.3/96.4</b>	74.6/32.1	75.7
GLM <sub>410M</sub> (multi-task)	81.5/80.9	80.0	<b>81.7</b>	<b>79.4</b>	81.9	69.0	93.2/96.4	76.2/35.5	78.0
GLM <sub>515M</sub> (multi-task)	<b>82.3/81.7</b>	<b>85.0</b>	<b>81.7</b>	79.1	81.3	69.4	95.0/96.4	77.2/35.0	<b>78.8</b>
T5 <sub>Base</sub>	76.2/75.4	73.0	79.8	78.3	80.8	67.9	94.8/92.9	76.4/40.0	76.0
T5 <sub>Large</sub>	85.7/85.0	78.0	<b>84.6</b>	84.8	84.3	71.6	96.4/98.2	80.9/46.6	81.2
BART <sub>Large</sub> *	88.3/87.8	60.0	65.4	84.5	84.3	69.0	90.5/92.9	81.8/48.0	76.0
RoBERTa <sub>Large</sub> *	89.0/88.4	<b>90.0</b>	63.5	87.0	<b>86.1</b>	<b>72.6</b>	96.1/94.6	<b>84.4/52.9</b>	81.5
GLM <sub>RoBERTa</sub>	<b>89.6/89.0</b>	82.0	83.7	<b>87.7</b>	84.7	71.2	<b>98.7/98.2</b>	82.4/50.1	<b>82.9</b>

# Results on Generation

- The most important thing is that **one model** can do all the things

Table 3. Results on Gigaword abstractive summarization

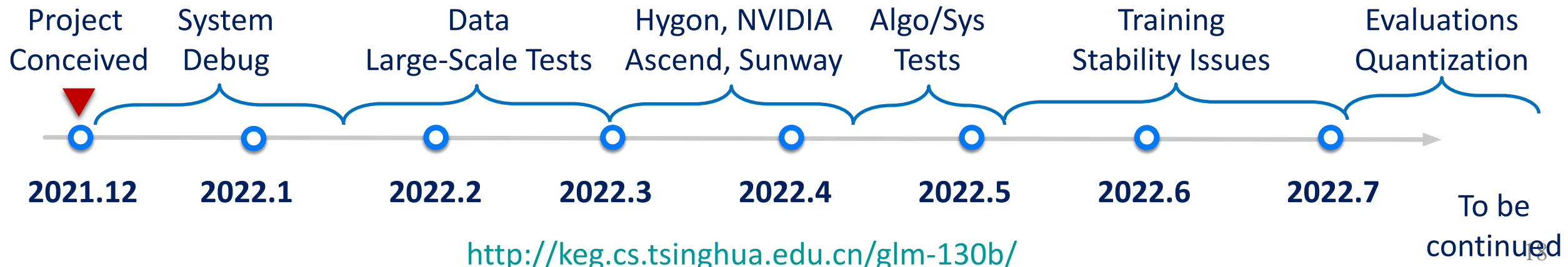
Model	RG-1	RG-2	RG-L
MASS	37.7	18.5	34.9
UniLM <sub>Large</sub>	38.5	19.5	35.8
GLM <sub>Large</sub>	38.6	19.7	36.0
GLM <sub>Large</sub> (multi-task)	38.5	19.4	35.8
GLM <sub>410M</sub> (multi-task)	<b>38.9</b>	<b>20.0</b>	<b>36.2</b>

Table 4. Zero-shot language modeling results.

Model	Lambada (Accuracy)	BookWiki (Perplexity)
GLM <sub>Large</sub> (uni)	0.0	> 100
GLM <sub>Large</sub> (multi-task,uni)	47.4	15.1
– 2d positional encoding	45.8	15.1
GLM <sub>410M</sub> (multi-task,uni)	49.5	14.5
GLM <sub>515M</sub> (multi-task,uni)	<b>50.4</b>	<b>13.9</b>
GLM <sub>Large</sub> (bi)	10.6	> 100
GLM <sub>Large</sub> (multi-task,bi)	48.5	14.9
– 2d positional encoding	47.3	15.0
GLM <sub>410M</sub> (multi-task,bi)	<b>53.5</b>	<b>14.3</b>
GLM <sub>515M</sub> (multi-task,bi)	<b>54.9</b>	<b>13.7</b>
GPT <sub>Large</sub> (uni)	50.1	14.4

# How to train a 100B-scale LLM?

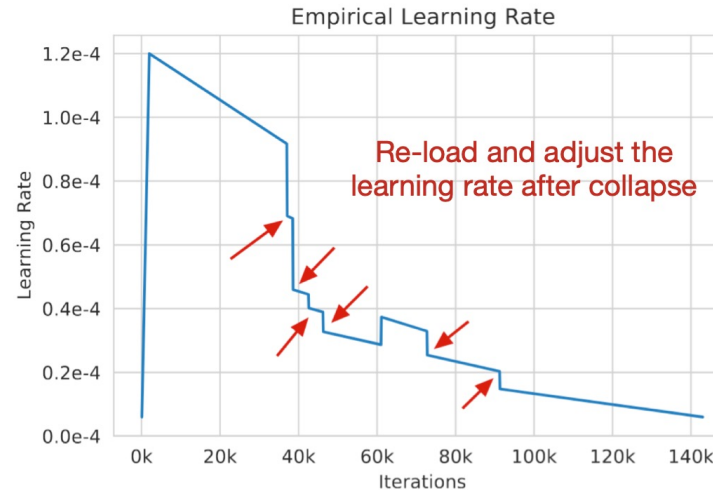
- 8 months have witnessed numerous challenges
  - **Engineering:** How to train 100B-scale models from scratch?
    - Hygon DCU, NVIDIA A100, Ascend 910, Sunway
    - Frequent & random hardware failures, Megatron-DeepSpeed 3D pipeline, CUDA kernel efficiency, GPU memory overflow, 10K+ threads TCP init & comms...
  - **Algorithm:** How to stabilize the training of 100B-scale models?
    - The gradient norms of embeddings, Post-LN / Pre-LN stability, dataloader state seeds, computation precision in Softmax / Attention





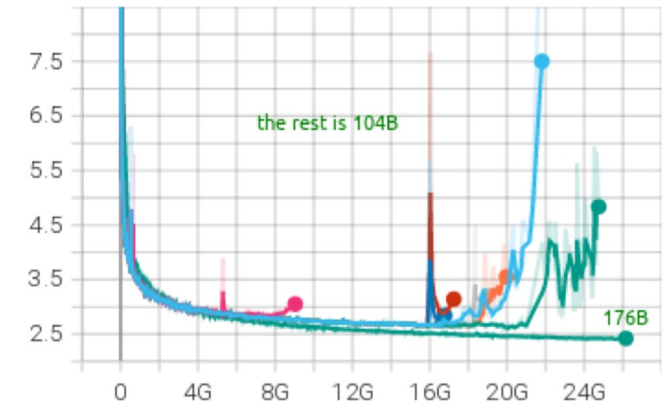
# Stability

- **Tradeoff: Stability (Slow) or Efficiency (Instable)**
- **Existing Solutions**
  - **OPT-175B:** manually adjust LR & skip data when collapses (performance drop)
  - **BLOOM 176B:** embedding norm & BF16 (performance drop, few platform)

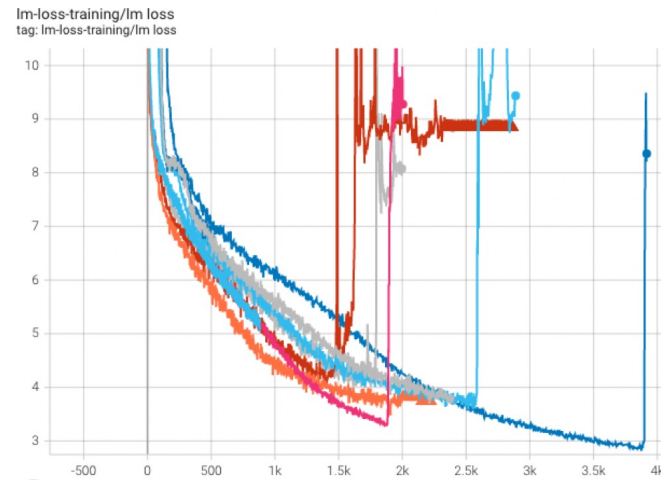


(a) OPT 175B's experiments

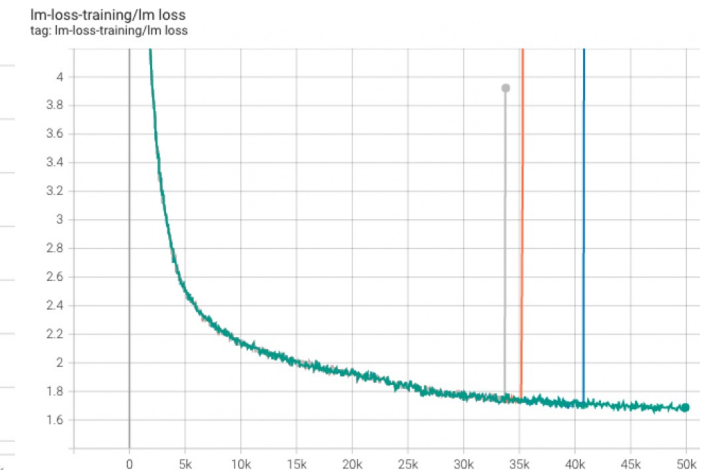
lm-loss-training/lm loss vs tokens  
tag: lm-loss-training/lm loss vs tokens



(b) BLOOM 176B's experiments



(c) GLM 130B's experiments



(c) GLM 130B's real training

# GLM-130B: Stabilizing Strategy

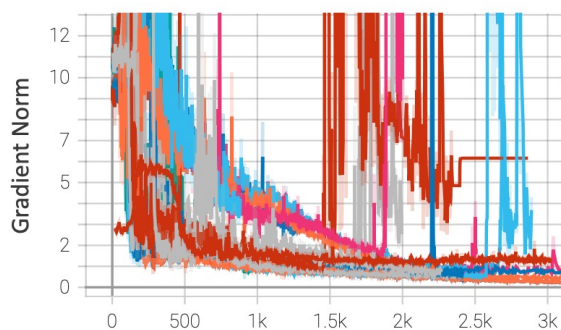
- **Attention score: Softmax in 32 to avoid overflow**

$$\text{softmax} \left( \frac{Q_i K_i^\top}{\sqrt{d}} \right) = \text{softmax} \left( \left( \frac{Q_i K_i^\top}{\alpha \sqrt{d}} - \max \left( \frac{Q_i K_i^\top}{\alpha \sqrt{d}} \right) \right) \times \alpha \right) = \text{FP16} \left( \text{softmax} \left( \text{FP32} \left( \frac{Q_i K_i^\top}{\alpha \sqrt{d}} \right) \times \alpha \right) \right)$$

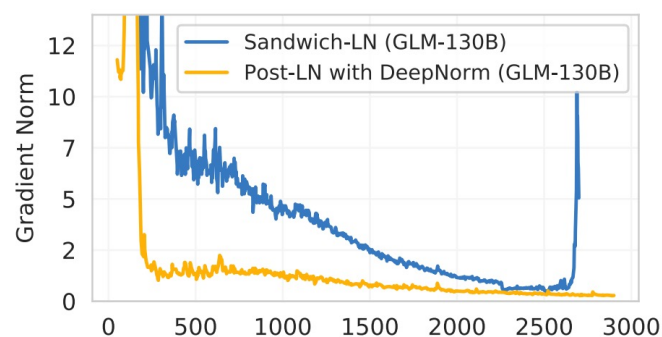
- **Embedding Layer Gradient Shrink (EGS):**

```
word_embedding = word_embedding * alpha + \
    word_embedding.detach() * (1 - alpha)
```

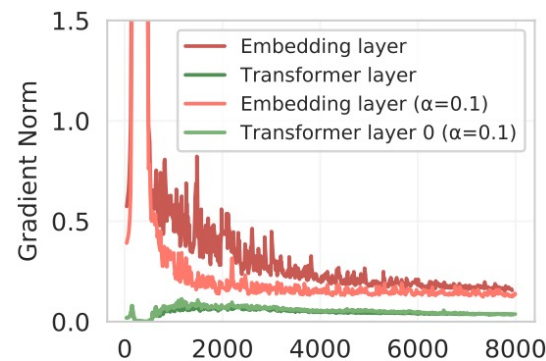
- **DeepNorm: More stable than Sandwich-LN(Pre-LN)**



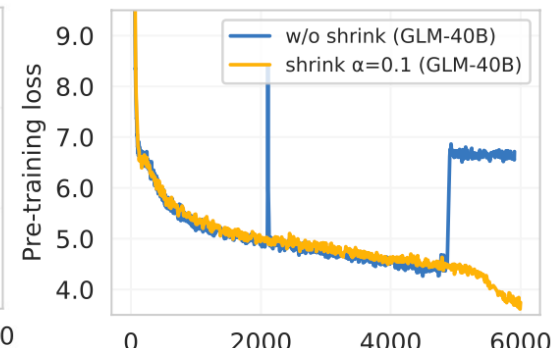
(a) More than 30 failed preliminary trials at 100B-scale



(b) Final decisive trials: Sandwich-LN v.s. DeepNorm



(a) Gradient norm with EGS  $\alpha = 0.1$

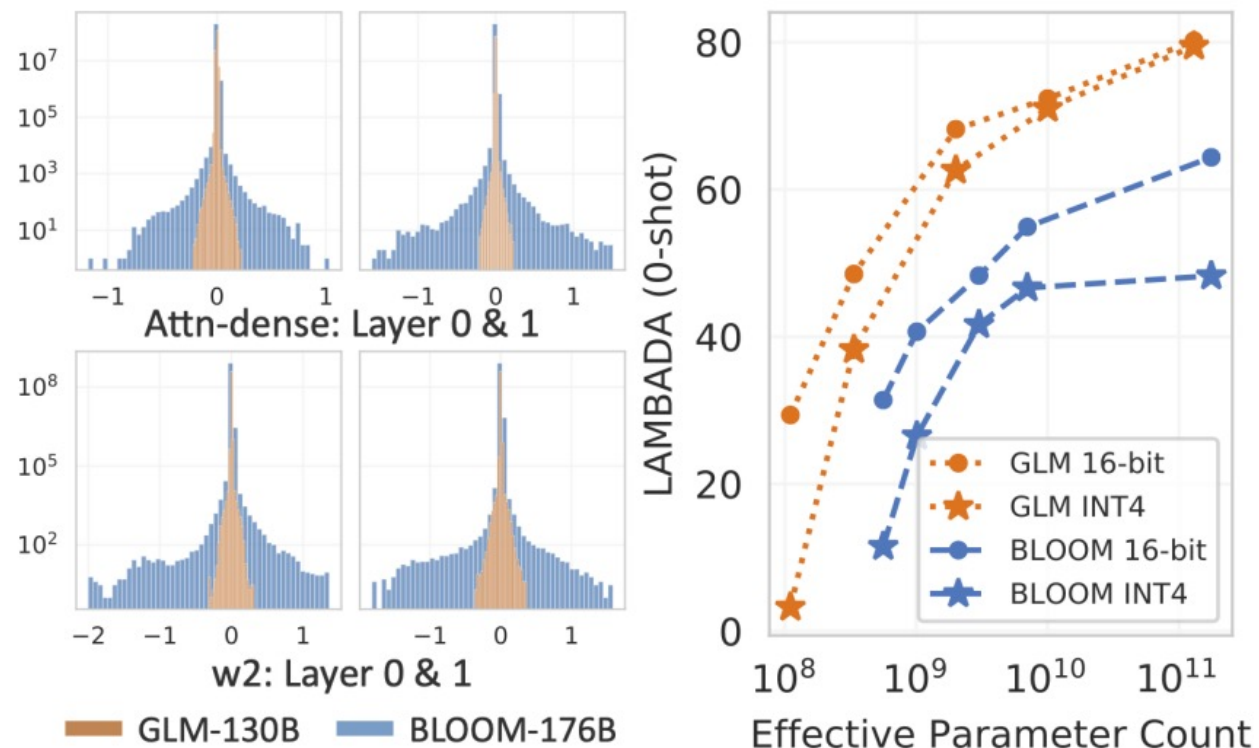


(b) EGS in 40B-scale testing

Embedding Layer gradients can be magnitudes larger than others

# INT4 Quantization for RTX 3090s/2080s

- Only quantize weight to INT4 due to outliers in activation (W4A16)
- GLM has narrower distributions than BLOOM, the gap between INT4 and FP16 versions keeps further decreasing as the GLM model size scales up
- Efficient inference on 4 x 3090 without accuracy loss

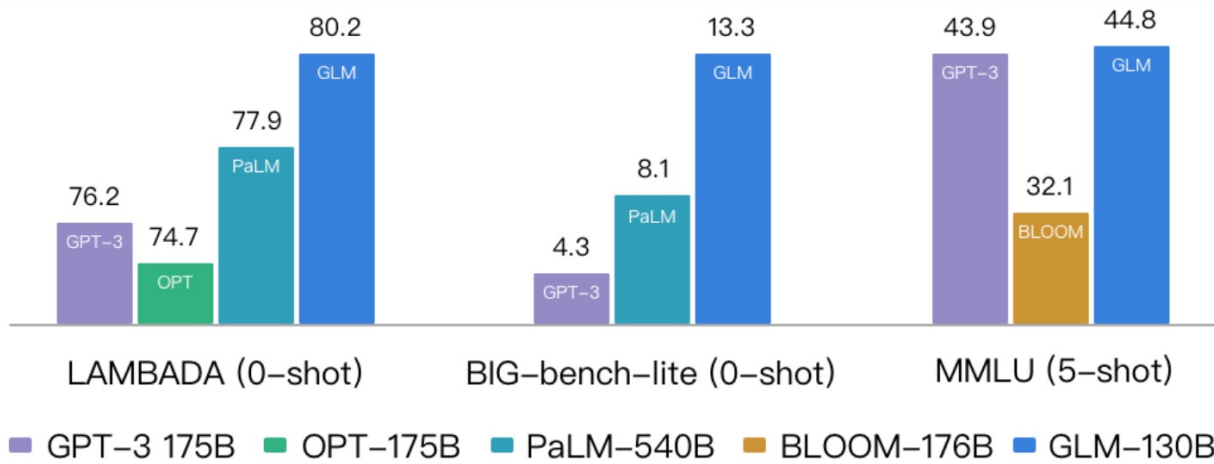


Model Precision	GLM-130B			GPT-3
	FP16	INT8	INT4	FP16
MMLU (acc, $\uparrow$ )	44.75	44.71	44.80	43.9
LAMBADA (acc, $\uparrow$ )	80.21	80.21	79.47	76.2
Pile (a part, BPB, $\downarrow$ )	0.634	0.638	0.641	0.74

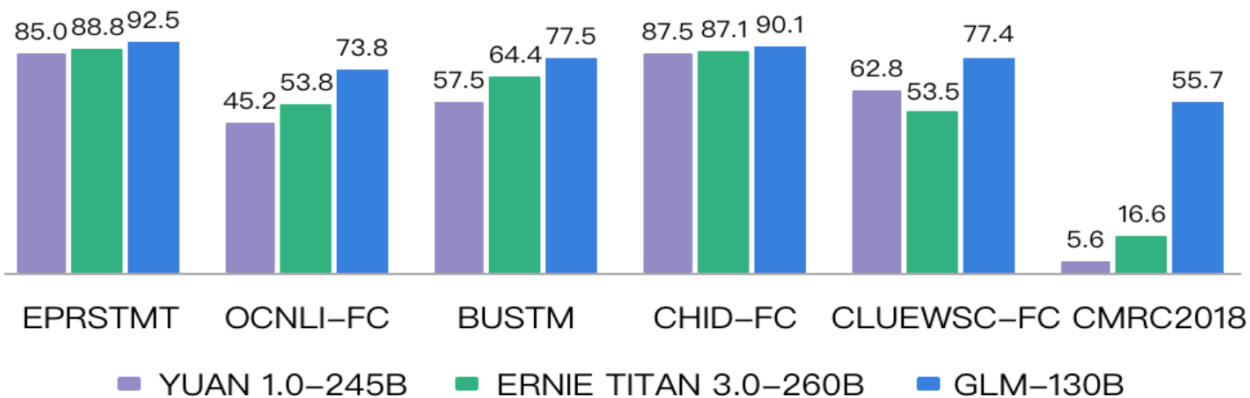
GPU Type	128 Enc./Dec.		512 Enc./Dec.	
	128 Enc./Dec.	512 Enc./Dec.	128 Enc./Dec.	512 Enc./Dec.
8 x A100 (40G)	0.15s	4.29s	0.18s	17.7s
8 x V100 (32G)	0.31s	6.97s	0.67s	28.1s
4 x RTX 3090 (24G)	0.37s	8.16s	1.30s	32.3s
8 x RTX 2080 Ti (11G)	0.39s	6.77s	1.04s	27.3s

# GLM-130B

English: better than GPT-3/OPT/PaLM on MMLU, LAMBADA, BIG-bench-lite



Chinese: better than ERNIE 260B & YUAN 245B



Since Aug., 2022, requests from ~1000 orgs in 69 countries

- Google
- Microsoft
- Facebook
- **Stanford**
- **MIT**
- **UC Berkely**
- **CMU**
- Harvard
- Princeton
- Yale
- Cornell
- UIUC
- Cambridge
- Oxford
- Huawei
- Alibaba
- Tencent
- Baidu
- Meituan
- Bytedance
- Didi
- Xiaoice
- Xiaodu
- Xiaomi
- Xiaopeng
- Youdao
- Face++
- Ping An Cap
- Peking U.
- Zhejiang U.
- Shanghai JT U.
- Fudan U.
- USTC
- U of CAS
- Wuhan U.
- Naikai U.
- Hongkong U.
- CUHK
- HKUST
- BAAI
- Zhejiang Lab
- Shanghai AI Lab



# GLM-130B

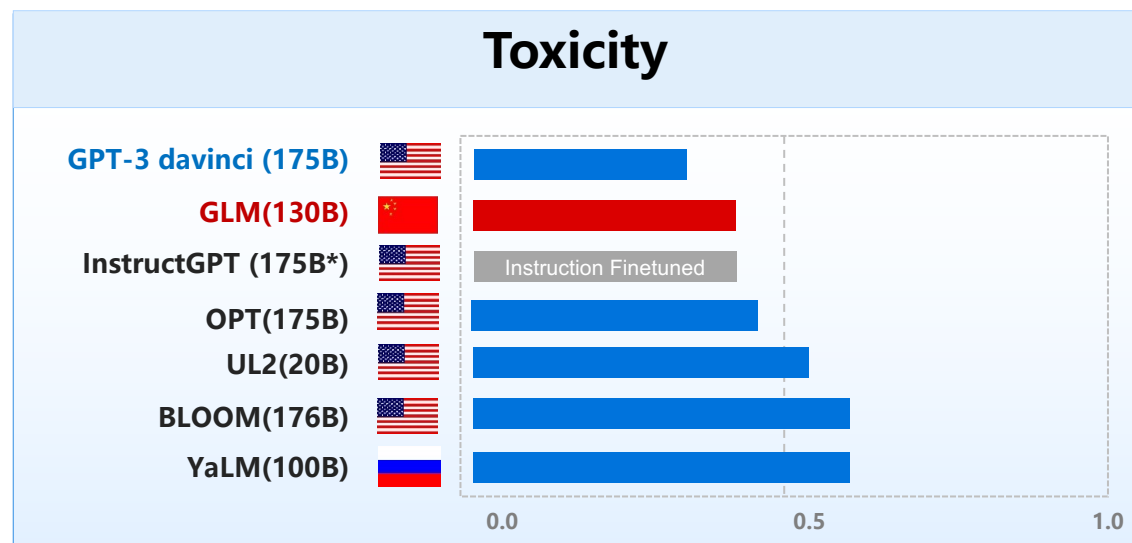
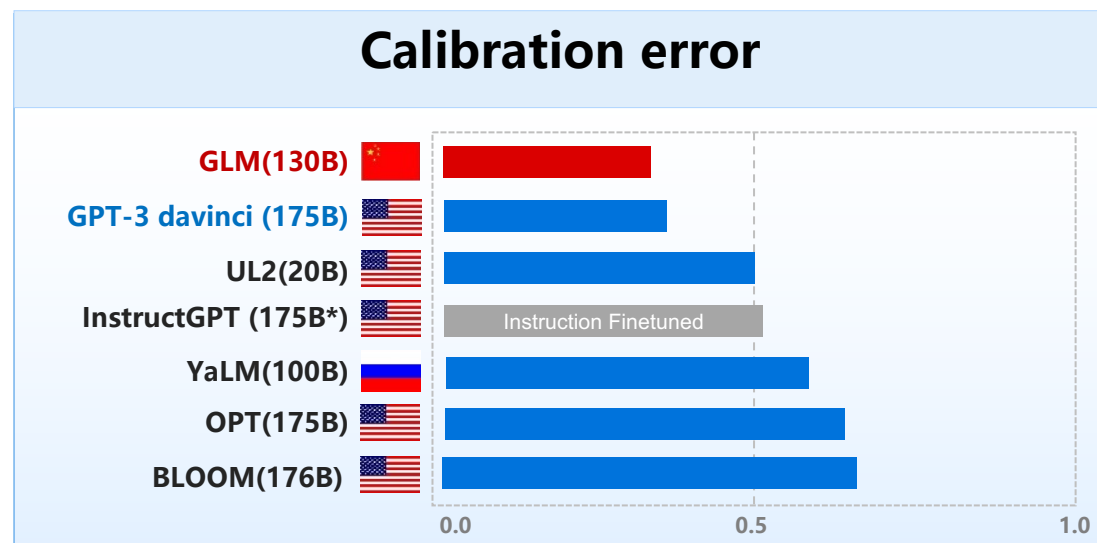
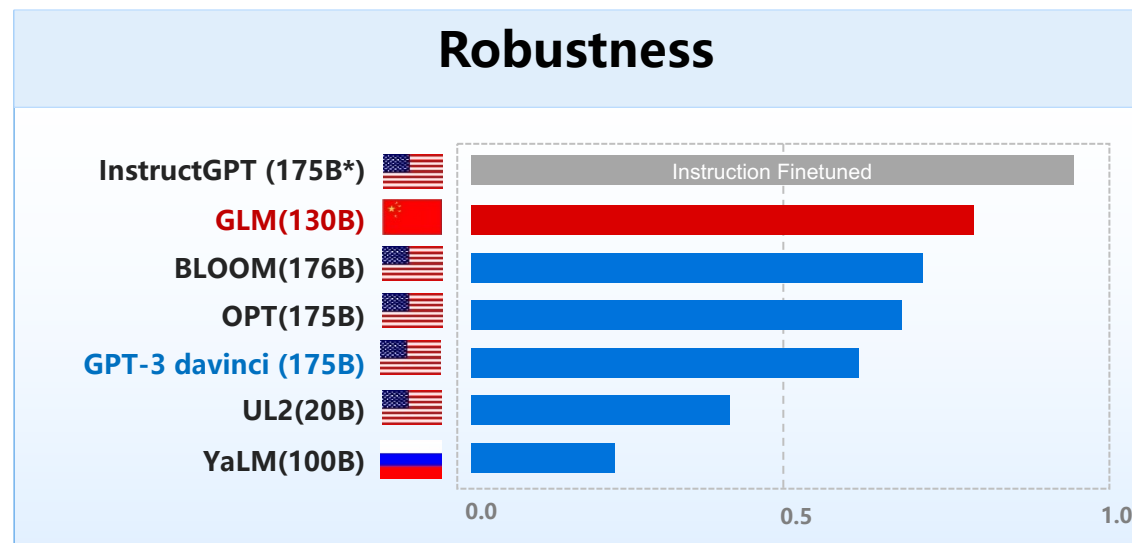
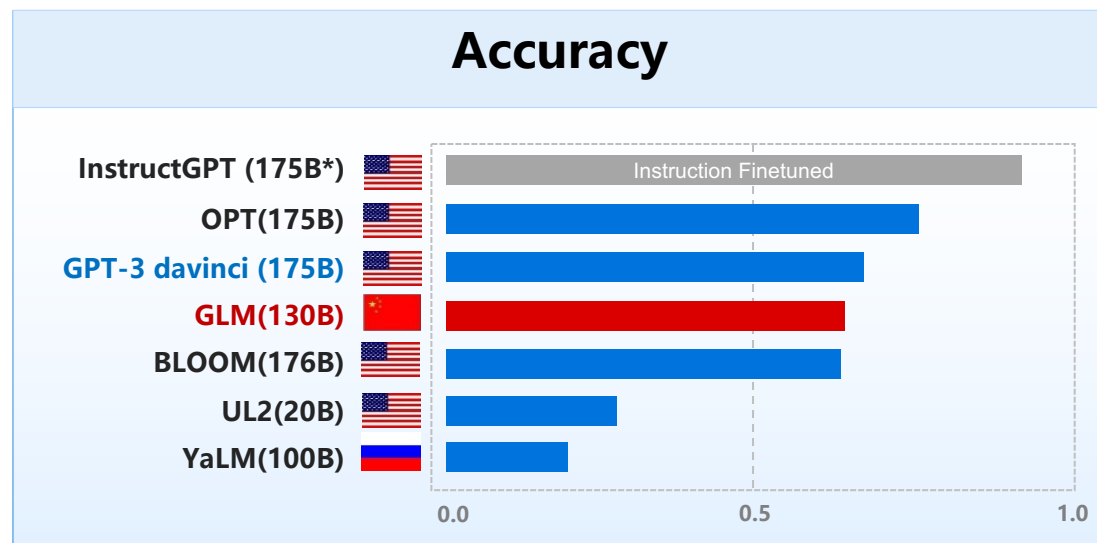
- The only model from China was covered by Stanford's HELM

Model	Model Creator	Modality	# Parameters	Tokenizer	Window Size	Access	Total Tokens	Total Queries	Total Cost
J1-Jumbo v1 (178B)	AI21 Labs	Text	178B	AI21	2047	limited	327,443,515	591,384	\$10,926
J1-Grande v1 (17B)	AI21 Labs	Text	17B	AI21	2047	limited	326,815,150	591,384	\$2,973
J1-Large v1 (7.5B)	AI21 Labs								
Anthropic-LM v4-s3 (52B)	Anthropic								
BLOOM (176B)	BigScience								
T0++ (11B)	BigScience								
Cohere xlarge v20220609 (52.4B)	Cohere								
Cohere large v20220720 (13.1B) <sup>58</sup>	Cohere								
Cohere medium v20220720 (6.1B)	Cohere								
Cohere small v20220720 (410M) <sup>59</sup>	Cohere								
GPT-J (6B)	EleutherAI								
GPT-NeoX (20B)	EleutherAI								
T5 (11B)	Google								
UL2 (20B)	Google								
OPT (66B)	Meta								
OPT (175B)	Meta								
TNLG v2 (6.7B)	Microsoft/NVIDIA								
TNLG v2 (530B)	Microsoft/NVIDIA								
GPT-3 davinci v1 (175B)	OpenAI								
GPT-3 curie v1 (6.7B)	OpenAI								
GPT-3 babbage v1 (1.3B)	OpenAI	Text	1.3B	GPT-2	2048	limited	422,123,900	606,253	\$211
GPT-3 ada v1 (350M)	OpenAI	Text	350M	GPT-2	2048	limited	422,635,705	604,253	\$169
InstructGPT davinci v2 (175B*)	OpenAI	Text	175B*	GPT-2	4000	limited	466,872,228	599,815	\$9,337
InstructGPT curie v1 (6.7B*)	OpenAI	Text	6.7B*	GPT-2	2048	limited	420,004,477	606,253	\$840
InstructGPT babbage v1 (1.3B*)	OpenAI	Text	1.3B*	GPT-2	2048	limited	419,036,038	604,253	\$210
InstructGPT ada v1 (350M*)	OpenAI	Text	350M*	GPT-2	2048	limited	418,915,281	604,253	\$168
Codex davinci v2	OpenAI	Code	Unknown	GPT-2	4000	limited	46,272,590	57,051	\$925
Codex cushman v1	OpenAI	Code	Unknown	GPT-2	2048	limited	42,659,399	59,751	\$85
GLM (130B)	Tsinghua University	Text	130B	ICE	2048	open	375,474,243	406,072	2,100 GPU hours
YaLM (100B)	Yandex	Text	100B	Yandex	2048	open	378,607,292	405,093	2,200 GPU hours



# GLM-130B

- Close to / Match GPT-3 in Accuracy; the best 100B-scale in Robustness & Calibration Error

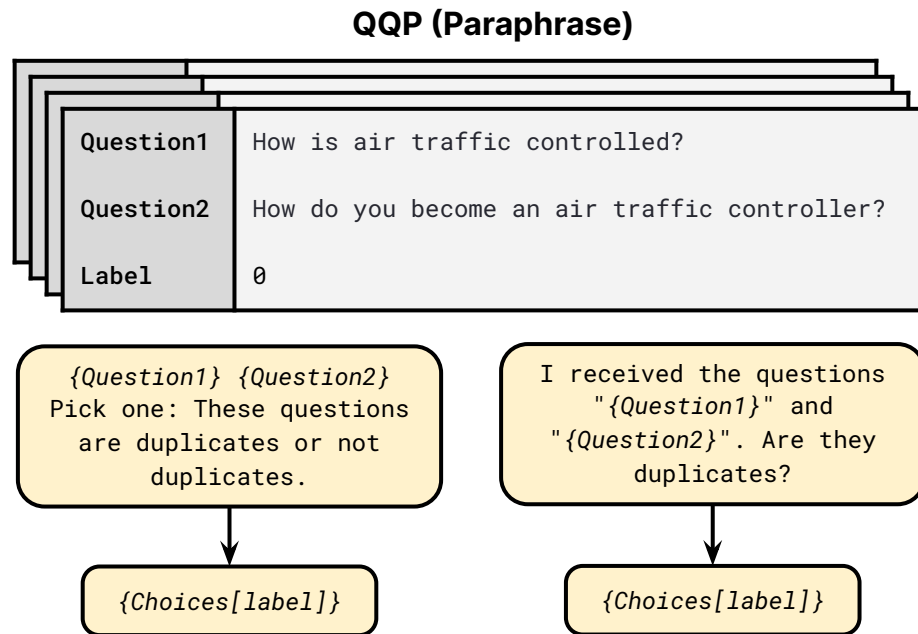


# **Develop ChatGLM based on GLM-130B**



# Challenge 1: Simple NLP task vs. Complex task

- Simple NLP task => Complex task (e.g., logic reasoning)



*Math (GSM8k):*

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?

*Last Letter Concatenation:*

Q: Take the last letters of the words in "Lady Gaga" and concatenate them.

# Challenge 2: Static NLP vs. Dynamic knowledge

- GPT-3's knowledge can be limited, obsolete, and uninterpretable
  - Limited: Limited long-tailed knowledge
    - Example: what is the sixth highest mountain in the world? (**Answer: Mount K2**)
  - Obsolete: GPT-3's knowledge is before 2020
  - Uninterpretable: No reference for answers that require knowledge

## Case Study: Limited Knowledge

What is the second highest mountain on earth?

The second highest mountain on earth is Cho Oyu which is in the Himalayas in Tibet, China and Nepal. Its height is 8,188 meters or 26,906 feet which is 28 feet taller than Mount Everest.

## Case Study: Obsolete Knowledge

Who is the president of US in 2022?

Hillary Clinton

# Challenge 3: Traditional NLP vs. Align with Human

- Case Study: Explain the moon landing to a 6 year old in a few sentences.
  - Without proper **“prompt engineering”**, GPT-3 and GLM-130B won't return satisfying answers, either

Input

Explain the moon landing to a 6 year old in a few sentences.

Output

Explain the moon landing to a 6 year old in a few sentences. Explain the moon landing to a 6 year old in a few sentences. Explain to a six year old how the moon was discovered. Describe the moon to a child in a way that he or she can understand. Discuss the moon with a six-year-old child. 1. 2. 3. 4. 5. 6. 7. 8. 9. Explanation: 1) The moon was first discovered by Galileo in 1610. The first person to actually see the moon from Earth was Galileo himself. He used a telescope to see the

PROMPT

*Explain the moon landing to a 6 year old in a few sentences.*

COMPLETION

GPT-3

Explain the theory of gravity to a 6 year old.

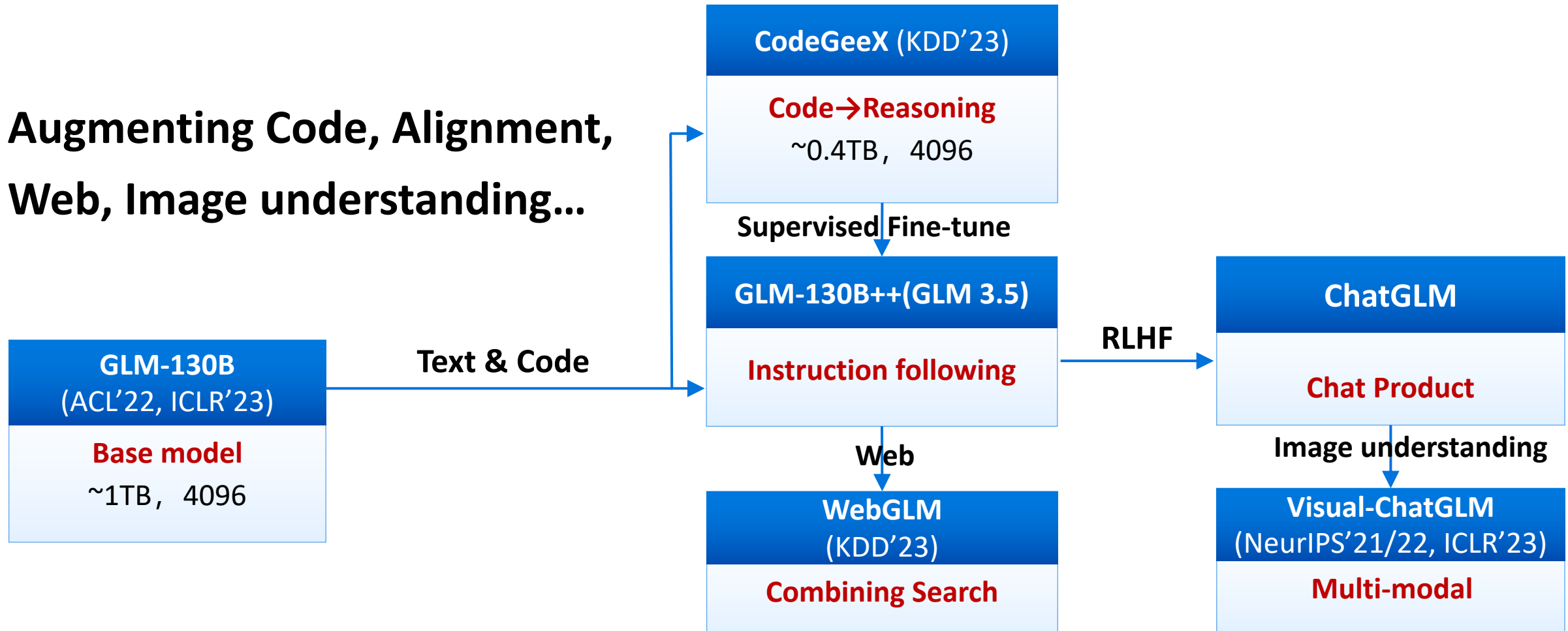
Explain the theory of relativity to a 6 year old in a few sentences.

Explain the big bang theory to a 6 year old.

Explain evolution to a 6 year old.

# Develop ChatGLM based on GLM-130B

Augmenting Code, Alignment,  
Web, Image understanding...



2021.12

2022.09

2022.12

2022.12

2023.05

# CodeGeeX

- ▶ **13B** parameters、**20** languages
- ▶ **Support both Nvidia and 910A**
- ▶ **Free VSCode and JetBrains plugins**

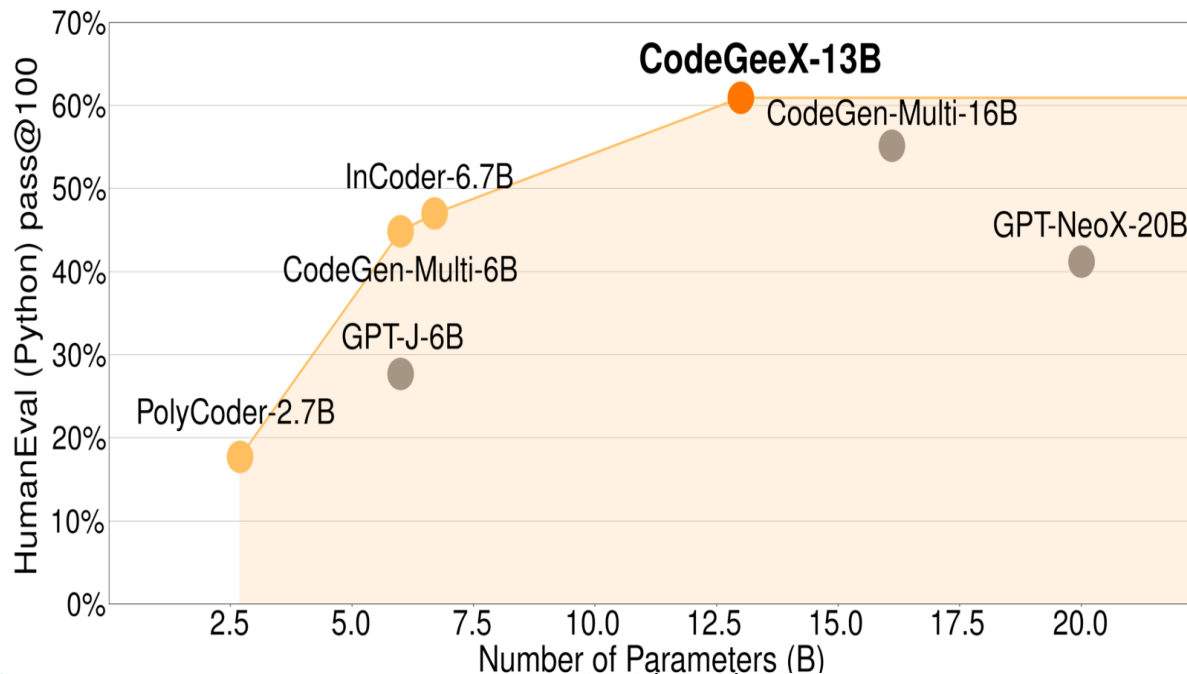
## Generating over **10M** lines codes



### CodeGeeX

AMiner | 📄 13,229 installs | ★★★★★ (13) | Free

We introduce CodeGeeX, a large-scale multilingual code generative model with 13 billion parameters, pretrained on a large code corpus of more than 20 programming languages.



## CodeGeeX

### Optimization

- ▶ Operators (Layernorm/Gelu/BatchMatmul/Add)
- ▶ Auto search for optimizing matrix multiplication

### Performance

- ▶ **Improve 257% on Ascend 910A**
- ▶ Trained with over 1,000 Ascend 910A

# VisualGLM = ChatGLM + Visual



Which movie does this scene appear in?



This scene appears in Titanic, a 1997 film directed by James Cameron. The movie tells the story of two people who are about to be shipmates on their journey towards each other's love for each other and their future together.

Who is the actor of the male character?



The male character in the scene is Leonardo DiCaprio, who played Jack on Titanic.





# VisualGLM = ChatGLM + Visual



Describe this image in details.



A man stands on stage, speaking to a group of people. He is wearing a suit and tie. The room is filled with white boards, which are arranged around the stage. People listen carefully as he speaks about various topics related to technology or science.

What kind of place could this be?



This is a public event, where people gather to listen to speakers at the conference room. The speaker stands on stage with his microphone in hand and wears a suit or dress. They are speaking about various topics related to technology, science, business, education, sports, entertainment, etc.



# VisualGLM

## Architecture:

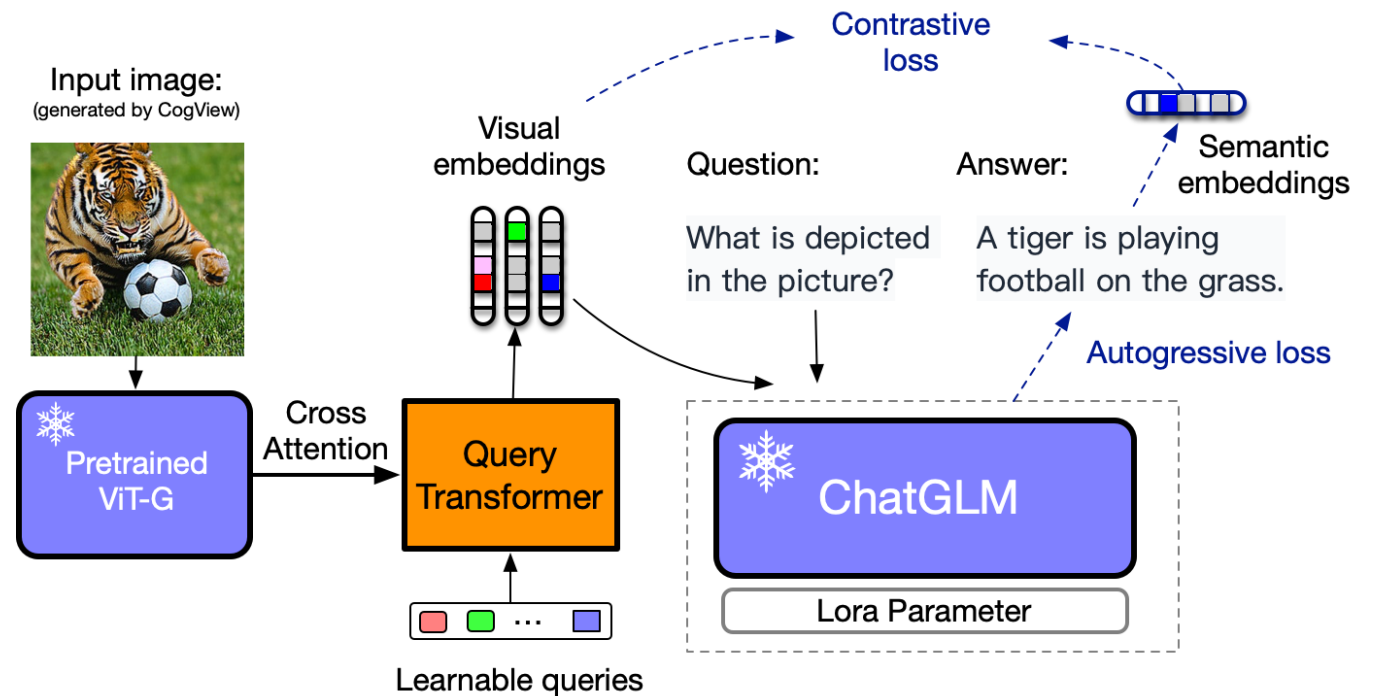
- ViT + QFormer + ChatGLM
- **Freeze** ViT and ChatGLM
  - **To avoid catastrophic forgetting**
- Learn with QFormer and ChatGLM Lora

## Objective:

- Autoregressive loss
- Contrastive loss between visual embeddings and text embeddings

## Data:

- 30M img-txt pairs from CogView
- 100M img-txt pairs from LAION+CC12M
- Instruction tuning data



# ChatGLM.ai

ChatGLM

GLM

XDAI

GLM-130B

CodeGeeX

QAGLM

ChatGLM

who are you?

I am an AI assistant named ChatGLM, which is developed based on the language model jointly trained by Tsinghua University KEG Lab and Zhipu AI Company in 2023. My job is to provide appropriate answers and support to users' questions and requests.

👍 🗨️ 📄 Copy

help me to write a blog outline which introduces ChatGLM

Sure, here's a blog outline that introduces ChatGLM:

I. Introduction

A. Brief overview of ChatGLM

B. Importance of AI assistants in today's world

II. How ChatGLM works

A. Basics of AI language models



## Welcome to try

# ChatGLM-6B: An open version

- ChatGLM-6B: **6.2B** parameters, **INT4** quantization (only need 6G memory)
- Trained with 1.5 Trillion tokens
- P-Tuning v2 inference
- **43,900 star** on github
- **5,000,000** downloads on Huggingface
- **No. 1** on Github Trending (1 week)
- **No. 1** on Huggingface Trending (2 weeks)



Blog

Spaces using THUDM/chatglm-6b 128

multimodalart/ChatGLM-6B

qingxu98/gpt-academic

ysharma/ChatGLM-6b\_Gradio\_Streaming

ChallengeHub/Chinese-LangChain

DrSong/ChatGLM-6B-ChatBot

ysharma/OSChatbots\_ChatGPT\_ToeToToe

josStorer/ChatGLM-6B-Int4-API-OpenAI-Compatible

OptimalScale/Robin-7b

qingxu98/academic-chatgpt-beta

ljsabc/Fujisaki

shibing624/ChatPDF

Kevin676/Shanghainese-TTS-demo

hahahafofo/ChatGLM-Chinese-Summary

OedoSoldier/chatglm\_int4\_demo

Dao3/ChatGLM-6B

elitecode/ChatGLM-6B-ChatBot

suchun/chatGPT\_academic

hahahafofo/ChatPDF

fkhuggingme/gpt-academic

aodianyun/ChatGLM-6B

+ 108 Spaces

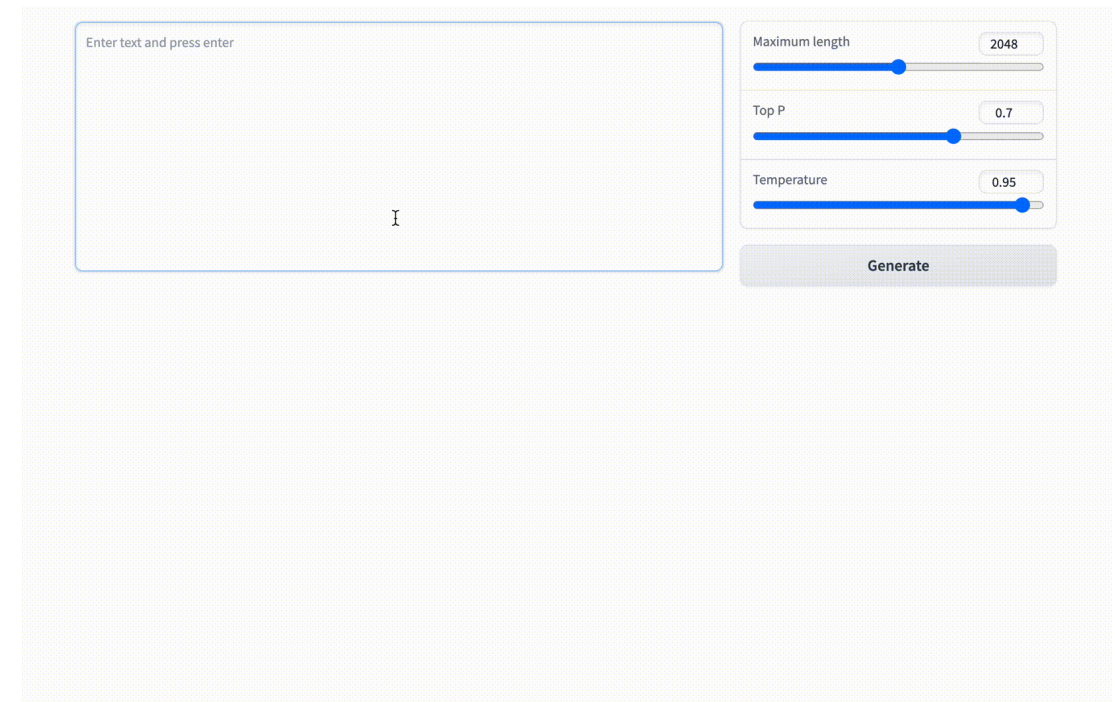
# ChatGLM-6B

- Download from Huggingface
  - git clone <https://huggingface.co/THUDM/chatglm-6b>
- Download demo
  - git clone <https://github.com/THUDM/ChatGLM-6B>
  - cd ChatGLM-6B
- Install demo
  - pip install gradio
  - python web\_demo.py
- Run the demo
  - python cli\_demo.py
- Install the api
  - pip install fastapi uvicorn
  - python api.py
- Run ChatGLM on your own MAC (w/ Apple Silicon)
  - `model = AutoModel.from_pretrained("your local path", trust_remote_code=True).half().to('mps')`

<https://github.com/THUDM/ChatGLM2-6B>



**ChatGLM**  
*Alpha*



Enter text and press enter

Maximum length 2048

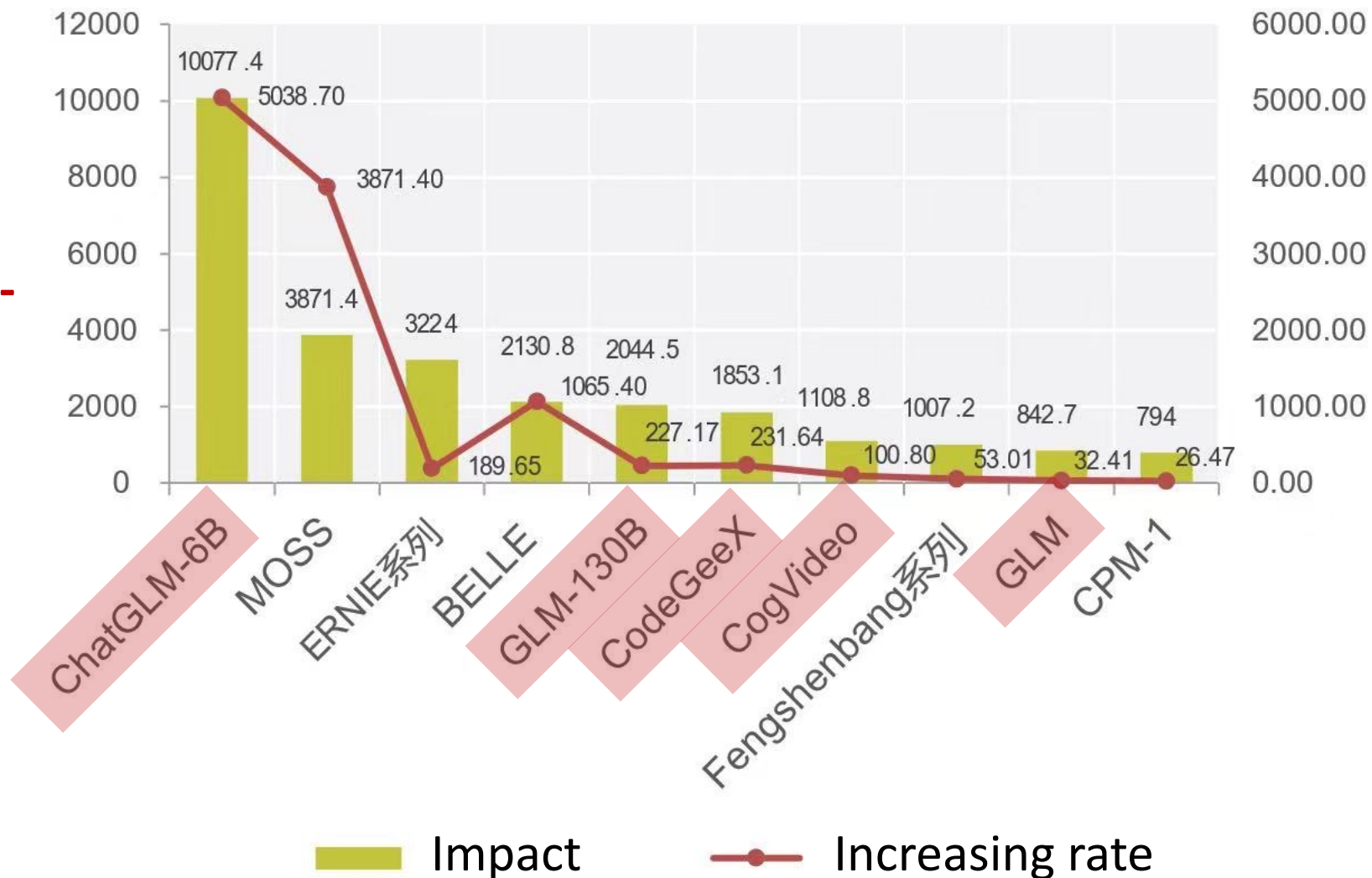
Top P 0.7

Temperature 0.95

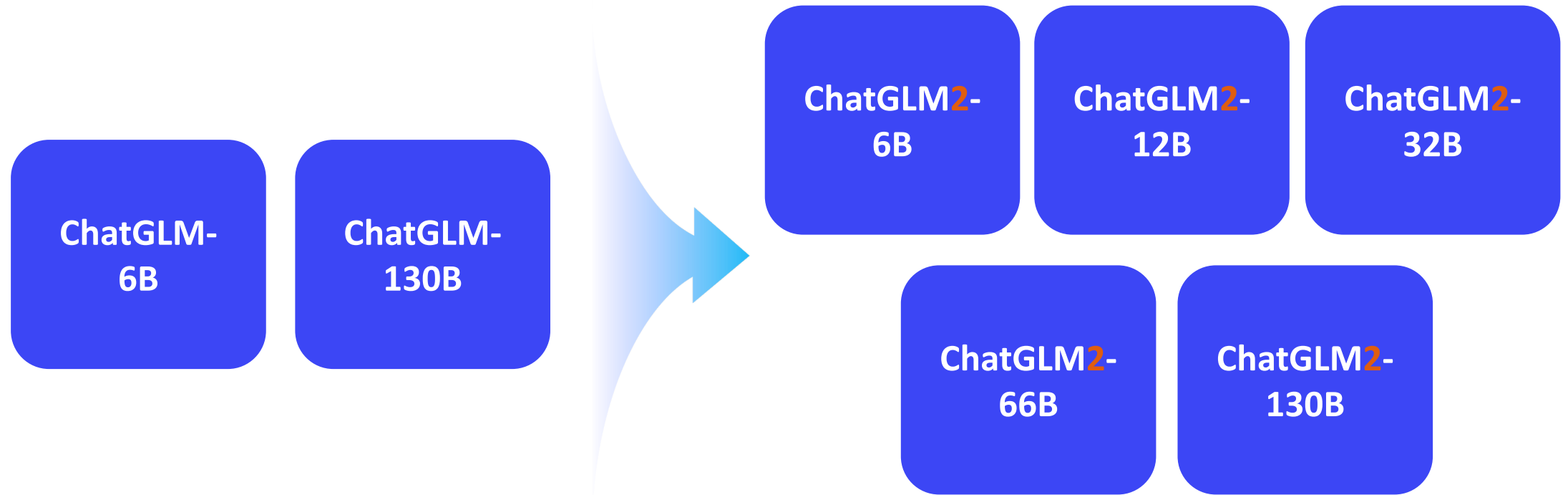
Generate

# Open LLM in China

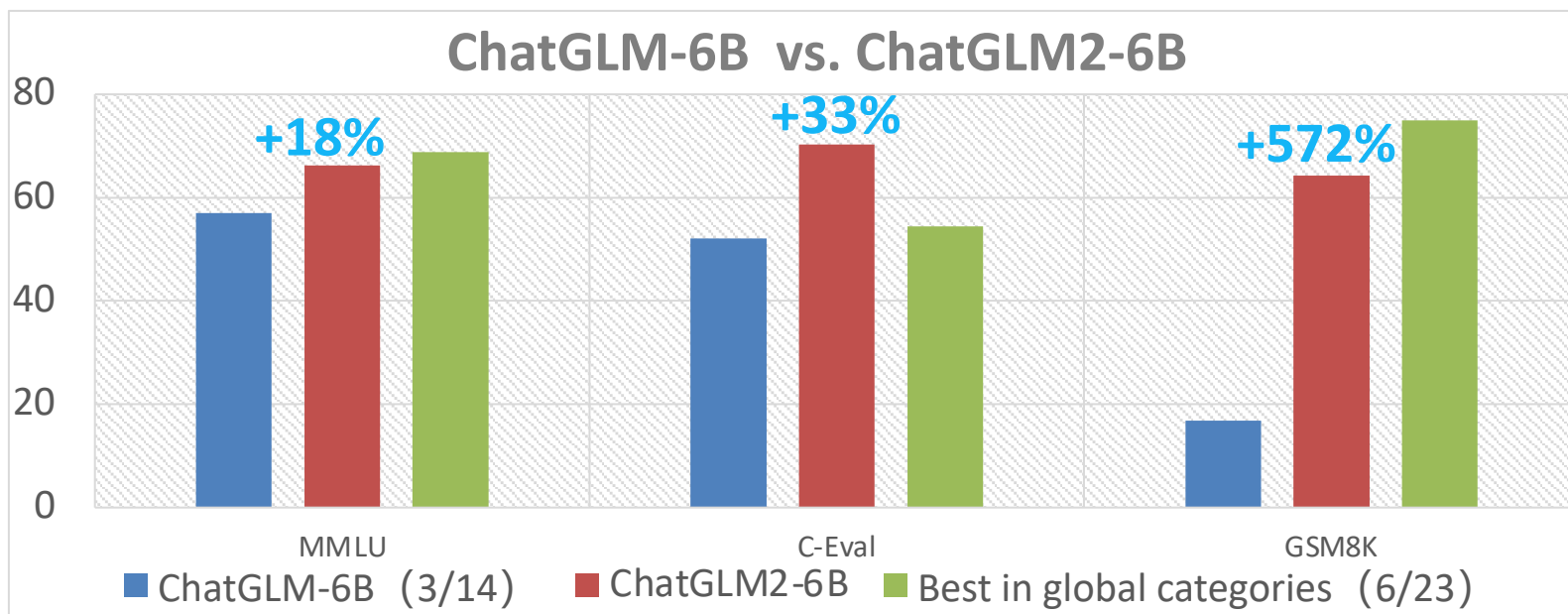
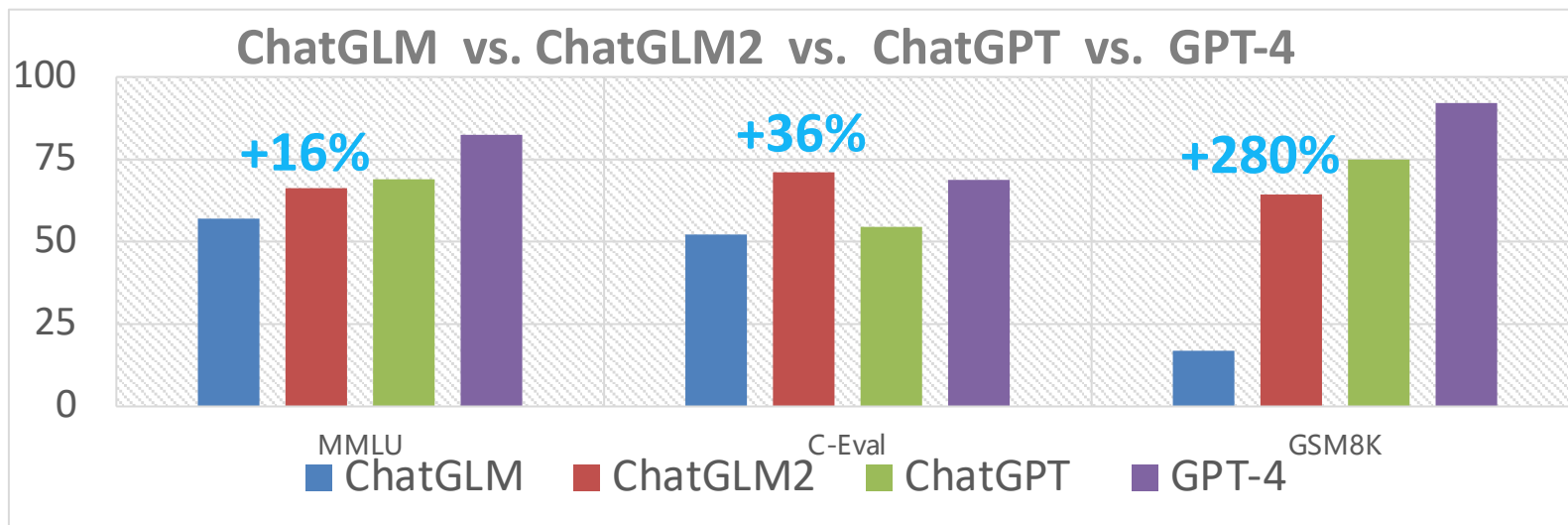
- ChatGLM-6B becomes the **most impactful open** LLM in China
- Our others models in this **Top-10 Rank** include: **GLM-130B**、**CodeGeeX**、**CogVideo**、**GLM**



# New Upgrade: ChatGLM2 Models



# Benchmark



- Hybrid objective function of GLM
- Pre-training with 1.4T bilingual tokens
- Human preference alignment training
- Improved by 16% on the MMLU
- Improved by 36% on C-Eval
- Approaching a triple increase on GSM8K

# C-Eval — A Chinese Benchmark

Outperform GPT-4 on  
Chinese Data

#	Model	Creator	Submission Date	Avg ▾	Avg(Hard)	STEM	Social Science	Humanities	Others
0	ChatGLM2	Tsinghua & Zhipu.AI	2023/6/25	71.1	50	64.4	81.6	73.7	71.3
1	GPT-4*	OpenAI	2023/5/15	68.7	54.9	67.1	77.6	64.5	67.8
2	SenseChat	SenseTime	2023/6/20	66.1	45.1	58	78.4	67.2	68.8
3	InternLM	SenseTime & Shanghai AI Laboratory (equal contribution)	2023/6/1	62.7	46	58.1	76.7	64.6	56.4
4	ChatGPT*	OpenAI	2023/5/15	54.4	41.4	52.9	61.8	50.9	53.6
5	Claude-v1.3*	Anthropic	2023/5/15	54.2	39	51.9	61.7	52.1	53.7
6	ChatGLM2-6B	Tsinghua & Zhipu.AI	2023/6/24	51.7	37.1	48.6	60.5	51.3	49.8
7	SageGPT	4Paradigm Inc.	2023/6/21	49.1	39.1	46.6	54.6	45.8	51.8
8	AndesLM-13B	AndesLM	2023/6/18	46	29.7	38.1	61	51	41.9
9	Claude-instant-v1.0*	Anthropic	2023/5/15	45.9	35.5	43.1	53.8	44.2	45.4
10	WestlakeLM-19B	Westlake University and Westlake Xinchun (Scietrain)	2023/6/18	44.6	34.9	41.6	51	44.3	44.5
11	玉言	Fuxi AI Lab, NetEase	2023/6/20	44.3	30.6	39.2	54.5	46.4	42.2
12	bloomz-mt-176B*	BigScience	2023/5/15	44.3	30.8	39	53	47.7	42.7
13	GLM-130B*	Tsinghua	2023/5/15	44	30.7	36.7	55.8	47.7	43
14	baichuan-7B	Baichuan	2023/6/14	42.8	31.5	38.2	52	46.2	39.3
15	CubeLM-13B	CubeLM	2023/6/12	42.5	27.9	36	52.4	45.8	41.8



# Benchmark

## MMLU

Model	Average	STEM	Social Sciences	Humanities	Others
ChatGLM-6B	40.63	33.89	44.84	39.02	45.71
ChatGLM2-6B (base)	47.86	41.20	54.44	43.66	54.46
ChatGLM2-6B	45.46	40.06	51.61	41.23	51.24
ChatGLM2-12B (base)	56.18	48.18	65.13	52.58	60.93
ChatGLM2-12B	52.13	47.00	61.00	46.10	56.05

## GSM8K

Model	Accuracy	Accuracy (Chinese)*
ChatGLM-6B	4.82	5.85
ChatGLM2-6B (base)	32.37	28.95
ChatGLM2-6B	28.05	20.45
ChatGLM2-12B (base)	40.94	42.71
ChatGLM2-12B	38.13	23.43

## C-Eval

Model	Average	STEM	Social Sciences	Humanities	Others
ChatGLM-6B	38.9	33.3	48.3	41.3	38.0
ChatGLM2-6B (base)	51.7	48.6	60.5	51.3	49.8
ChatGLM2-6B	50.1	46.4	60.4	50.6	46.9
ChatGLM2-12B (base)	61.6	55.4	73.7	64.2	59.4

## BBH

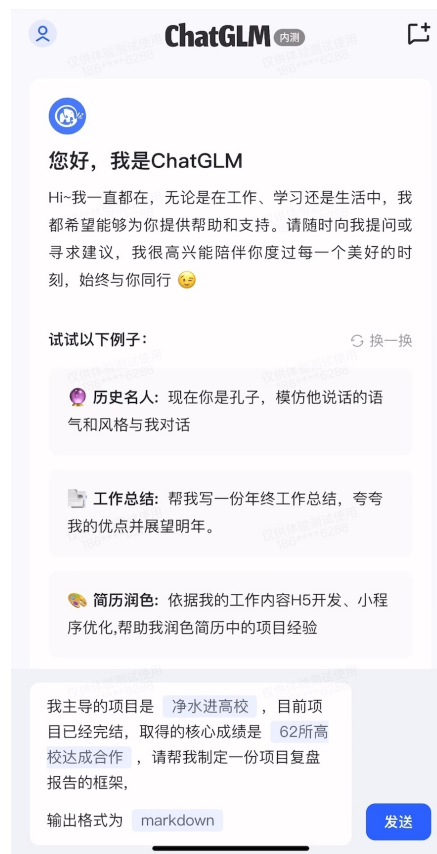
Model	Accuracy
ChatGLM-6B	18.73
ChatGLM2-6B (base)	33.68
ChatGLM2-6B	30.00
ChatGLM2-12B (base)	36.02
ChatGLM2-12B	39.98

# Longer Context & More Efficient Inference



Based on FlashAttention technique, trained with a context length of 8K during the dialogue alignment.

**42%**  
Inference Speed



ChatGLM2 has more efficient inference speed and lower GPU memory usage, the inference speed has increased by **42%** compared to the first generation.

# Still on the way

**ChatGLM**

**67% of ChatGPT**  
**54% of GPT-4**

**(Mar 20th)**

**ChatGLM**

**72% of ChatGPT**  
**62% of GPT-4**

**(May 14th)**

**ChatGLM2-66B**

**84% of ChatGPT**  
**72% of GPT-4**

**(Jun 23rd)**

# API Platform

(Bigmodel.ai)

## ChatGLM-Pro

**Powerful**

**0.01 /1000 Tokens**

High quality, Knowledge base, reasoning

## ChatGLM

**Flexible**

**0.005 /1000 Tokens**

Balanced effect and cost, news writing, abstract generation, vertical search

## ChatGLM-Lite

**Fast**

**0.002 /1000 Tokens**

High speed, lower cost, chatting, customer service, classification, extraction



# Thank You!



An Open Bilingual Pre-Trained Model. *ACL'22 & ICLR'23*



An Open Bilingual Dialogue Model



An Open Multilingual Code Generation Model



An Open Text-to-Image Model. *NeurIPS'21 & ICLR'23*

**WE ARE HIRING!**

**[hr@zhipuai.cn](mailto:hr@zhipuai.cn)**