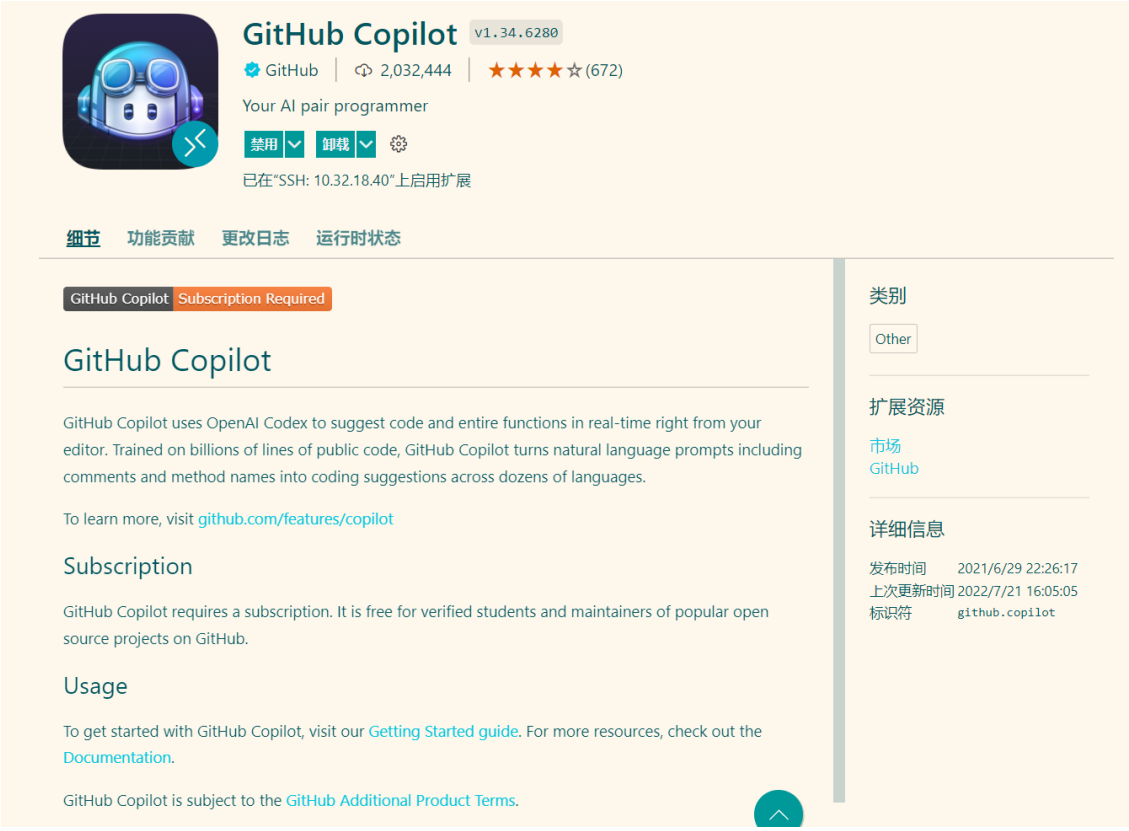
代码补全的预训练模型实践

作为对比的Benchmark:

- GitHub Copilot



1、PyCodeGPT(结论：暂未验证可用性)

地址：https://github.com/microsoft/PyCodeGPT

预训练模型已下载：

- PyCodeGPT-110M

Evaluation

1. Install requirements (python 3.7)

```
$ pip install -r requirements.txt
```

1. Install HumanEval

- Note that you can successfully evaluate your model after uncommenting 58th line of `human-eval/human_eval/execution.py`

```
$ git clone https://github.com/openai/human-eval
$ pip install -e human-eval
```

1. Run `eval_human_eval.py` to generate programs
   - Arguments
     - `model_name_or_path` : Path to the model checkpoint to be evaluated.
     - `output_dir` : Path to save generated programs
     - `num_completions` : The number of program to be generated
     - `temperature` : Temperature for sampling
     - `top_p` : p value for nucleus sampling
     - `max_new_tokens` : Maximum number of generated token
   - Example usage

     ```
     $ python eval_human_eval.py \
         --model_name_or_path PyCodeGPT-110M/ \
         --output_dir results/ \
         --num_completions 100 \
         --temperature 0.2 \
         --top_p 0.95 \
         --max_new_tokens 100 \
         --gpu_device 0
     ```

2. Evaluate functional correctness

   ```
   $ evaluate_functional_correctness <samples_path>
   # Example
   $ evaluate_functional_correctness
   results/human_eval.t0.2.p0.95.l100.n100.samples.jsonl
   ```

2、GPT-NEO(结论：不可用)

地址：https://github.com/EleutherAI/gpt-neo

预训练模型已下载：

- GPT-NEO，体积较大。

说明：

- 使用Mesh-tensorflow库的模型并行GPT-2和GPT-3式模型的实现。

此处的预训练模型为在The Pile数据集上进行训练的模型，无法直接进行代码补全的推理使用。

3、CodeGen(结论：可直接使用)

地址：https://github.com/salesforce/CodeGen

HuggingFace

The model is available on the HuggingFace Hub with a Colab demo here.

> 说明：
>
> CODEGEN-NL模型为在ThePile数据集上训练的模型；
>
> CODEGEN-MULTI模型为在BigQuery数据集上训练的多语言模型；
>
> CODEGEN-MONO模型为在单一数据集BIGPYTHON上训练的单语言模型；

Model	pass@k [%]		
	$k=1$	$k=10$	$k=100$
GPT-Neo 350M	0.85	2.55	5.95
GPT-Neo 2.7B	6.41	11.27	21.37
GPT-J 6B	11.62	15.74	27.74
Codex 300M	13.17	20.37	36.27
Codex 2.5B	21.36	35.42	59.50
Codex 12B	28.81	46.81	72.31
CodeGen-NL 350M	2.12	4.10	7.38
CodeGen-NL 2.7B	6.70	14.15	22.84
CodeGen-NL 6.1B	10.43	18.36	29.85
CodeGen-Multi 350M	6.67	10.61	16.84
CodeGen-Multi 2.7B	14.51	24.67	38.56
CodeGen-Multi 6.1B	18.16	28.71	44.85
CodeGen-Mono 350M	12.76	23.11	35.19
CodeGen-Mono 2.7B	23.70	36.64	57.01
CodeGen-Mono 6.1B	26.13	42.29	65.82
CodeGen-Mono 16.1B	**29.28**	**49.86**	**75.00**

- codegen-350M-nl
- codegen-350M-multi
- codegen-350M-mono
- codegen-2B-nl
- codegen-2B-multi
- codegen-2B-mono
- codegen-6B-nl
- codegen-6B-multi
- codegen-6B-mono

- codegen-16B-nl
- codegen-16B-multi
- codegen-16B-mono

```
import torch
from transformers import AutoTokenizer, AutoModelForCausalLM
tokenizer = AutoTokenizer.from_pretrained("Salesforce/codegen-2B-mono")
model = AutoModelForCausalLM.from_pretrained("Salesforce/codegen-2B-mono")
inputs = tokenizer("# this function prints hello world", return_tensors="pt").to(0)
sample = model.generate(**inputs, max_length=128)
print(tokenizer.decode(sample[0], truncate_before_pattern=[r"\n\n^#", "^'''",
"\n\n\n"]))
```

Colab

This Google Colab notebook allows for sampling from the CodeGen models.

4、InCoder(结论：可直接使用)

Github地址：https://github.com/dpfried/incoder/blob/main/README.md

官网地址：https://sites.google.com/view/incoder-code-models

Demo：https://huggingface.co/spaces/facebook/incoder-demo

Models

You can obtain the models from HuggingFace's hub:

- 6.7B parameter model: facebook/incoder-6B
- 1.3B parameter model: facebook/incoder-1B