一般庶民でも
fine-tuneしたい！
～ローカルLLMのGPU節約の手法を解説～

早野康太

自己紹介

- 名前
  - 早野 康太
- お仕事
  - 自然言語モデルの改善
- 今期おすすめ
  - ゾン100
  - 無職転生
  - 呪術廻戦
  - BLEACH
  - ライザ
  - 好きな子がめがねを忘れた

ChatGPTがすごいって話

- ChatGPT
  - 2022年末にOpenAIが公開した大規模言語モデル(LLM)
  - 非常に精度が高い回答を生成できることで話題に
  - 公開後2ヶ月でユーザー数1億人を突破

ChatGPTがすごいって話

大規模言語モデルとはなんですか？

ChatGPT

大規模言語モデルは、自然言語処理（NLP）分野において開発された、巨大なパラメータセットを持つ機械学習モデルです。
これらのモデルは、大量のテキストデータを学習して、人間の言語の特徴やパターンを理解し、自然な文章の生成や言語理解のタスクを行うことができます。

オープンソースのLLM

- ChatGPTは高性能ではあるものの
  モデルがAPI越しでしか利用できない
  - 個人情報や機密情報などを扱いたい場合

- 手元で動かせるオープンソースLLMも最近登場してきている
  - StableLM (StabilityAI)
  - Dolly 2.0 (Databricks)
  - Falcon (TEchnology Innovation Institute)

日本語LLM

- OpenCALM
  - CyberAgent社が開発した日本語LLM
    - サイバーエージェント、最大68億パラメータの日本語LLM（大規模言語モデル）を一般公開 ―オープンなデータで学習した商用利用可能なモデルを提供
    - Huggingfaceで公開
      - cyberagent/open-calm-7b · Hugging Face

実際オープンソースってどうなの？

```python
import torch
from transformers import AutoModelForCausalLM, AutoTokenizer

model = AutoModelForCausalLM.from_pretrained("cyberagent/open-calm-7b", device_map="auto", torch_dtype=torch.float16)
tokenizer = AutoTokenizer.from_pretrained("cyberagent/open-calm-7b")

inputs = tokenizer(
    "大規模言語モデルとはなんですか？",
    return_tensors="pt"
).to(model.device)
with torch.no_grad():
    tokens = model.generate(
        **inputs,
        max_new_tokens=256,
        do_sample=True,
        temperature=0.9,
        top_p=0.75,
        top_k=40,
        num_beams=10,
        repetition_penalty=5.0,
        pad_token_id=tokenizer.pad_token_id,
    )

output = tokenizer.decode(tokens[0], skip_special_tokens=True)
```

実際オープンソースってどうなの？

大規模言語モデルとはなんですか？

OpenCALM

大規模言語モデルとはなんですか?
Q. 自然言語処理で、構文解析ってどうやるんですか?
Q.文章を単語に分割するいい方法はありますか。
Q.日本語の文章をn-gramデータとして扱う際の多い・少ない文字数の構成比率はどのぐらいですか?

実際オープンソースってどうなの？

- タダで利用できるとはいえ
  そのままの状態で使用すると思い通りに答えてくれない
  - データセットを使って
    追加で学習させる必要がある（fine-tuning）

LLMのfine-tuning

- fine-tuning
  - LLMはあらかじめ大規模なデータで学習されているが
    全てのタスクに対応できるわけではない
  - 特定のタスクに適応させるためには
    追加でデータセットを与えて学習させる必要がある

LLMのfine-tuning

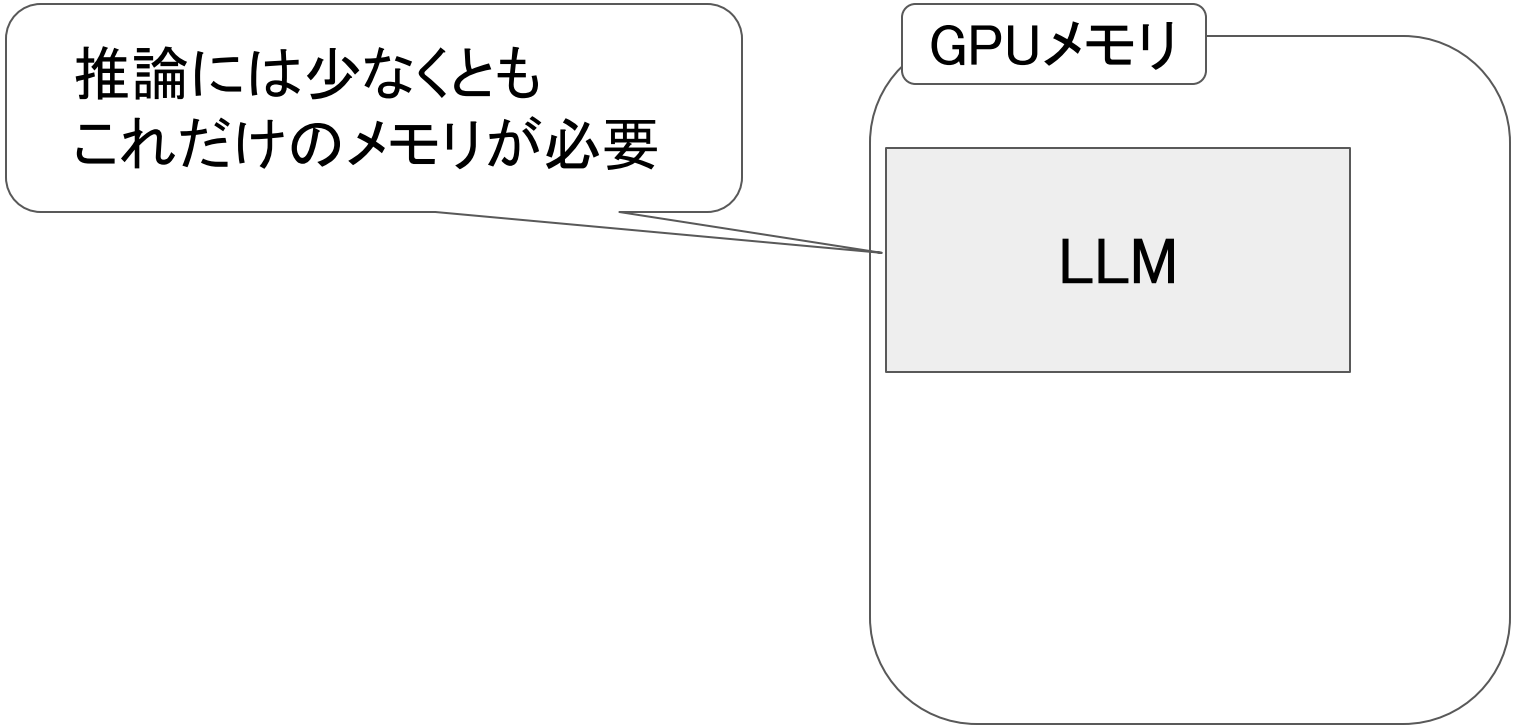
大規模言語モデルとはなんですか？

OpenCALM

質問に対する応答がうまくいかない
→ 質問応答の文章を与えて学習

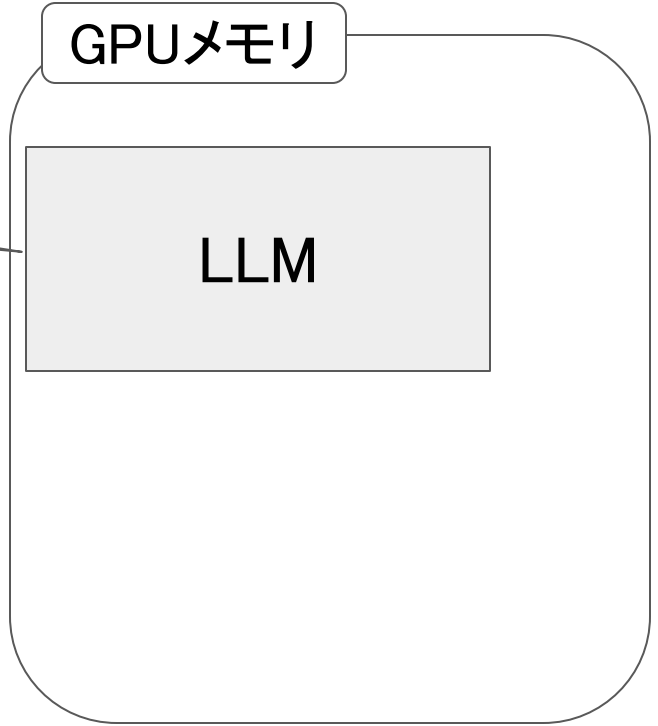
大規模言語モデルとはなんですか?
Q. 自然言語処理で、構文解析ってどうやるんですか?
Q.文章を単語に分割するいい方法はありますか。
Q.日本語の文章をn-gramデータとして扱う際の多い・少ない文字数の構成比率はどのぐらいですか?

LLMをローカルで動かす



LLMをローカルで動かす

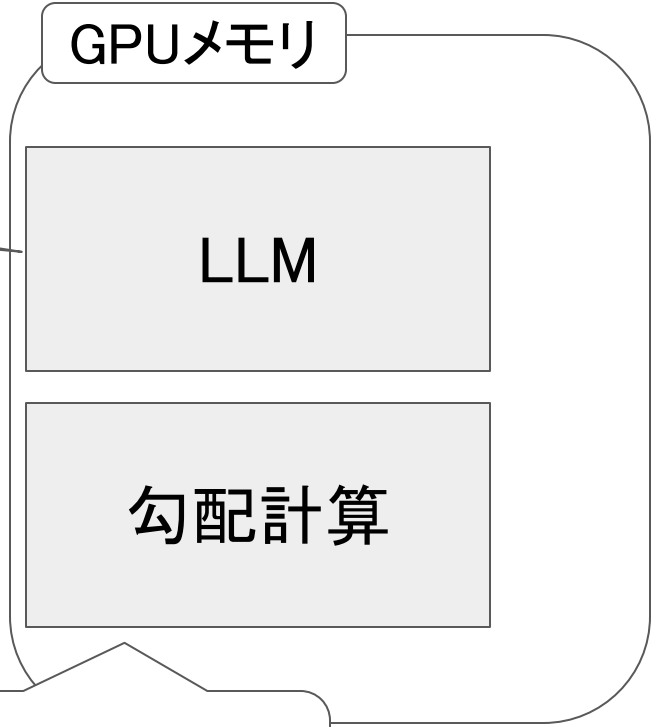
推論には少なくとも
これだけのメモリが必要



- LLMのサイズ
  ≒モデルのパラメータ数 × バイト数
  - OpenCALM (68億パラメータ)
    - float32
      → 68 × 10^9 × 4bytes = 28 GB
    - (参考)

LLMをローカルで動かす

推論には少なくとも
これだけのメモリが必要



- LLMのサイズ
  ≒モデルのパラメータ数 × バイト数
  - OpenCALM (68億パラメータ)
    - float32
      → 68 × 10^9 × 4bytes = 28 GB
    - (参考)

LLMをローカルで動かす

推論には少なくとも
これだけのメモリが必要

- LLMのサイズ
  ≒モデルのパラメータ数 × バイト数
  - OpenCALM (68億パラメータ)
    - float32
      → 68×10^9×4bytes = 28
    - (参考)

fine-tuneで

（GPU代が）

出典：新テニスの王子様© 許斐 剛／集英社

fine-tuningは
庶民には無理なのか？

メモリ消費を削減する方向性

- モデル自体のサイズを減らす
  - 量子化 (quantization)
    - モデルのパラメータ計算に
      使うビット数を減らす

- 学習するパラメータ数を減らす
  - LoRA (Low-Rank Adaptation)
    - モデル本体のパラメータを凍結して
      新たに学習するパラメータを挿入する

量子化（quantization）

- データタイプを変換すれば
  パラメータの保持に必要なメモリ使用量を削減できる
  - 4 byte FP32 → 2 byte FP16なら半分になる
  - ただし、値を丸めることで
    モデルの性能が低下する可能性はある
- transformersのライブラリでサポートされている

```
# pip install transformers accelerate bitsandbytes
from transformers import AutoModelForCausalLM, AutoTokenizer

model_id = "bigscience/bloom-1b7"

tokenizer = AutoTokenizer.from_pretrained(model_id)
model = AutoModelForCausalLM.from_pretrained(model_id, device_map="auto", load_in_4bit=True)
```

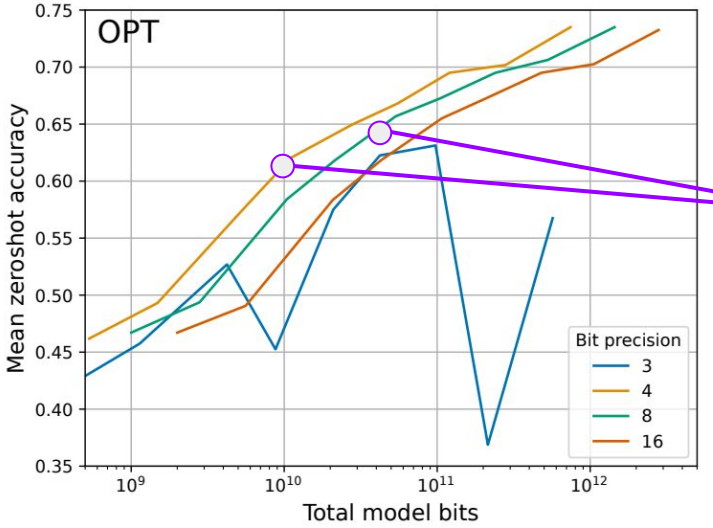
量子化 (quantization)

- LLM.int8(): 8-bit Matrix Multiplication for Transformers at Scale
  - GPUメモリにロードできるモデルサイズの違い

			Largest Model that can be run	
Class	Hardware	GPU Memory	8-bit	16-bit
Enterprise	8x A100	80 GB	**OPT-175B / BLOOM**	**OPT-175B / BLOOM**
Enterprise	8x A100	40 GB	**OPT-175B / BLOOM**	OPT-66B
Academic server	8x RTX 3090	24 GB	**OPT-175B / BLOOM**	OPT-66B
Academic desktop	4x RTX 3090	24 GB	**OPT-66B**	OPT-30B
Paid Cloud	Colab Pro	15 GB	**OPT-13B**	GPT-J-6B
Free Cloud	Colab	12 GB	**T0/T5-11B**	GPT-2 1.3B

量子化 (quantization)

- The case for 4-bit precision: k-bit Inference Scaling Laws
  - n-bitで量子化した際の総モデルビット数とaccuracyの関係
  - パラメータ数を固定したとき4-bitと8-bitで総モデルビット数が2倍違うことに注意



だいたい同じパラメータ数

LoRA (Low-Rank Adaptation)

- LoRA: Low-Rank Adaptation of Large Language Models
  - 元々のモデルのパラメータを更新せず
    追加のパラメータの値を導入して学習する
- Huggingfaceのpeftライブラリでサポートされている

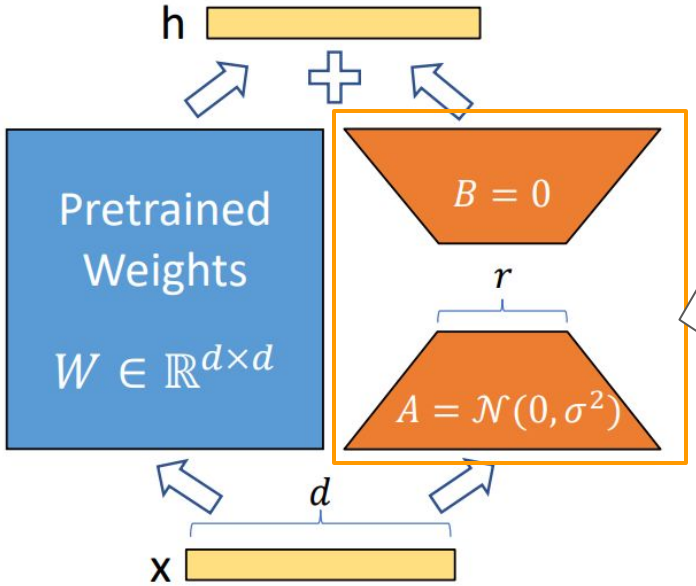
```
from transformers import AutoModelForSeq2SeqLM
from peft import get_peft_config, get_peft_model, LoraConfig, TaskType
model_name_or_path = "bigscience/mt0-large"
tokenizer_name_or_path = "bigscience/mt0-large"

peft_config = LoraConfig(
    task_type=TaskType.SEQ_2_SEQ_LM, inference_mode=False, r=8, lora_alpha=32, lora_dropout=0.1
)

model = AutoModelForSeq2SeqLM.from_pretrained(model_name_or_path)
model = get_peft_model(model, peft_config)
model.print_trainable_parameters()
```

LoRA (Low-Rank Adaptation)

- LoRA: Low-Rank Adaptation of Large Language Models
  - 元々のモデルのパラメータを更新せず
    追加のパラメータの値を導入して学習する



- パラメータA, Bを学習する
- 元々のモデルより
  はるかにパラメータ数が少ない

LoRA (Low-Rank Adaptation)

- GitHub - huggingface/peft: 🤗 PEFT: State-of-the-art Parameter-Efficient Fine-Tuning.
  - LoRAの有無によるメモリ消費の違い

Model	Full Finetuning	PEFT-LoRA PyTorch	PEFT-LoRA DeepSpeed with CPU Offloading
bigscience/T0_3B (3B params)	47.14GB GPU / 2.96GB CPU	14.4GB GPU / 2.96GB CPU	9.8GB GPU / 17.8GB CPU
bigscience/mt0-xxl (12B params)	OOM GPU	56GB GPU / 3GB CPU	22GB GPU / 52GB CPU
bigscience/bloomz-7b1 (7B params)	OOM GPU	32GB GPU / 3.8GB CPU	18.1GB GPU / 35GB CPU

LoRA試してみた

- コード
  - tloen/alpaca-lora: Instruct-tune LLaMA on consumer hardware
    - ほとんどそのままの形でスクリプト実行可能
- base model
  - OpenCALM (CyberAgent社, 68億パラメータ)
- ハイパラ
  - epochs: 3
  - lr: 0.0003
  - batch size: 32
  - lora_r: 8
  - lora_alpha: 16

フルfine-tuningなら28 GB以上かかるところを10数 GB程度のメモリ消費で実行できた

LoRA試してみた

- データセット
  - kunishou/databricks-dolly-15k-ja · Datasets at Hugging Face
    - instruction-tuning用のデータセットを日本語に翻訳したもの
    - “指示: {instruction} 入力: {input}”のように
      テンプレートにinput, instructionを当てはめた文を入力として
      学習させる

input	instruction	output
ヴァージン・オーストラリア航空（Virgin Australia Airlines Pty Ltd）は...	ヴァージン・オーストラリア航空はいつから運航を開始したのですか？	ヴァージン・オーストラリア航空は、2000年8月31日にヴァージン・ブルー航空として、2機の航空機で単一路線の運航を開始しました。

fine-tuning結果

大規模言語モデルとはなんですか？

OpenCALM LoRA fine-tuned

大規模な言語モデルは、自然言語処理の分野で最も強力なアルゴリズムの1つです。このアルゴリズムは、非構造化テキストから情報を抽出し、コンピューターが理解できる形式に変換するために使用されます。

fine-tuning結果

おすすめの中華料理店を教えてください

OpenCALM

【銀座・日比谷・有楽町】で誕生日・記念日祝いに人気のレストランランキング!
絶品モンブランや横浜中華街の小籠包など!こだわり餃子のお店まとめ

fine-tuning結果

おすすめの中華料理店を教えてください

OpenCALM LoRA fine-tuned

中華料理屋さんはたくさんあるので、どこに行くか迷いますよね。そんなときに便利なのが、Google Mapのストリートビューです。今いる場所の周辺を360度見渡せるので、行きたい場所がすぐに見つかりますよ。

fine-tuning結果

東大に受かるにはどうすればいいですか？

OpenCALM

僕は今高校3年生で、来年には大学受験を控えています。ですが、今まで全くと言っていいほど受験勉強をしてこなかったので、志望校に合格できるかとても不安です。そこで、皆さんにお聞きしたいことがあります。国語・数学・英語・社会・理科の5教科のうち、どの教科の偏差値を上げればよいと思いますか?回答よろしくお願いしますm(_ _)m

fine-tuning結果

東大に受かるにはどうすればいいですか？

OpenCALM LoRA fine-tuned

東京大学の入試問題は難問揃いですが、基本的なことをおさえていれば解ける問題も多く出題されています。まずは教科書レベルの問題を完璧にしていきましょう。その上で、過去問演習を通じて時間配分や解き方の手順を身に付けていくのが良いでしょう。

まとめ

- LLMのメモリ効率化技術
  - 量子化
    - パラメータのビット数を減らすことで
      メモリ上のLLMのサイズを減らす
  - LoRA
    - 学習するパラメータ数を減らすことで
      fine-tuning時のメモリ消費を減らす
- 量子化やLoRAを使えば
  fine-tuningのメモリ消費を数分の一に抑えられる
- OpenCALMはfine-tuningでだいぶいい感じになる