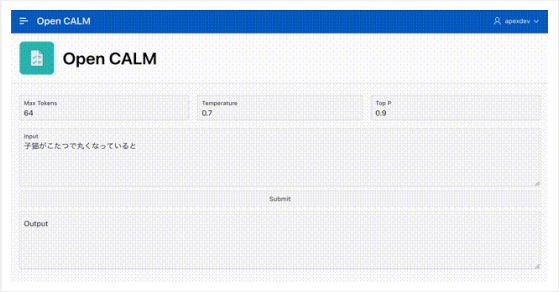
日日是Oracle APEX

Oracle APEXを使った作業をしていて、気の付いたところを忘れないようにメモをとります。

2023年7月26日水曜日

Always FreeのAmpere A1のインスタンスでOpenCALM-1Bを実行する

サイバーエージェント社が一般公開しているOpenCALM-1Bを、Oracle CloudのAmpere A1のインスタンスで実行してみました。Llama2を動かしたこちらの記事と同じインスタンスを使用しています。OpenCALM-3B、OpenCALM-7Bはメモリが足りないせいか実行できませんでした。



Ampere A1のインスタンスで実行するサーバーのコードは以下になります。

```
import logging
import sys
import os
import json

import torch
from transformers import AutoModelForCausalLM, AutoTokenizer
from flask import Flask, request

# ログレベルと出力先の設定
logging.basicConfig(stream=sys.stdout, level=logging.INFO, force=True)
logging.getLogger().addHandler(logging.StreamHandler(stream=sys.stdout))

# Prepare model and tokenizer
model = AutoModelForCausalLM.from_pretrained("cyberagent/open-calm-1b", device_map="auto")
tokenizer = AutoTokenizer.from_pretrained("cyberagent/open-calm-1b")

# /generateの呼び出しに対する処理
app = Flask(__name__)
@app.route('/generate', methods=['POST'])
def generate():
    if request.method == 'POST':
```

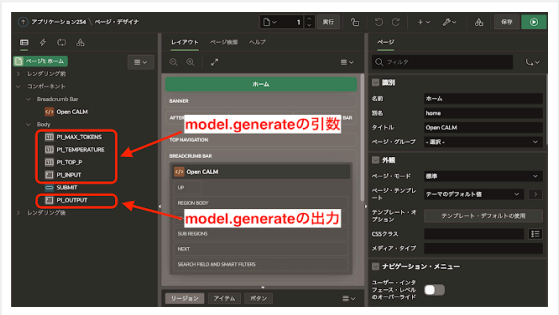
```
        text = request.json['input']
        max_tokens = request.json['max_tokens']
        top_p = request.json['top_p']
        temperature = request.json['temperature']
        print("received text", text)
        inputs = tokenizer(text, return_tensors="pt").to(model.device)
        with torch.no_grad():
            tokens = model.generate(
                **inputs,
                max_new_tokens=max_tokens,
                do_sample=True,
                temperature=temperature,
                top_p=top_p,
                repetition_penalty=1.05,
                pad_token_id=tokenizer.pad_token_id,
        )
        output = tokenizer.decode(tokens[0], skip_special_tokens=True)
        # output_json = json.dumps(output, ensure_ascii=False)
        return output

if __name__ == "__main__":
        app.run(host='0.0.0.0', port=8443, ssl_context=('./certs/fullchain.pem', './certs/privk
```

open-calm-server.py hosted with ❤️ by GitHub view raw

APEXアプリケーションのエクスポートを以下に置きました。

https://github.com/ujnak/apexapps/blob/master/exports/open-calm.zip

model.generateの引数となるページ・アイテムとして、**P1_MAX_TOKENS**、**P1_TEMPERATURE**、**P1_TOP_P**、**P1_INPUT**を作成しています。model.generateの出力はページ・アイテム**P1_OUTPUT**に設定します。

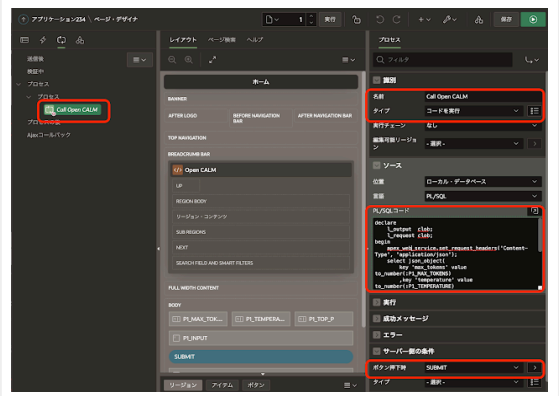


ボタン**SUBMIT**を押した時に、以下のコードを実行します。

```
declare
    l_output  clob;
    l_request clob;
begin
    apex_web_service.set_request_headers('Content-Type', 'application/json');
```

```
    select json_object(
        key 'max_tokens' value to_number(:P1_MAX_TOKENS)
        ,key 'temperature' value to_number(:P1_TEMPERATURE)
        ,key 'top_p' value to_number(:P1_TOP_P)
        ,key 'input' value :P1_INPUT
    ) into l_request from dual;
    l_output := apex_web_service.make_rest_request(
        p_url => :G_SERVER || '/generate'
        ,p_http_method => 'POST'
        ,p_body => l_request
    );
    :P1_OUTPUT := replace(l_output, '\n', chr(10));
end;
```

call-open-calm.sql hosted with ❤️ by GitHub view raw



呼び出すサーバーは**アプリケーション定義に**、置換文字列**G_SERVER**の置換値として設定します。

入力を変えて出力を確認する作業を手軽に行なうために、ユーザー・インターフェースを作ってみました。

Oracle APEXのアプリケーション作成の参考になると幸いです。

完

Yuji N. 時刻: 12:16

共有

自己紹介

Yuji N.

日本オラクル株式会社に勤務していて、Oracle APEXのGroundbreaker Advocateを拝命しました。
こちらの記事につきましては、免責事項の参照をお願いいたします。

詳細プロフィールを表示