

# 日々是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。

2024年5月24日 金曜日

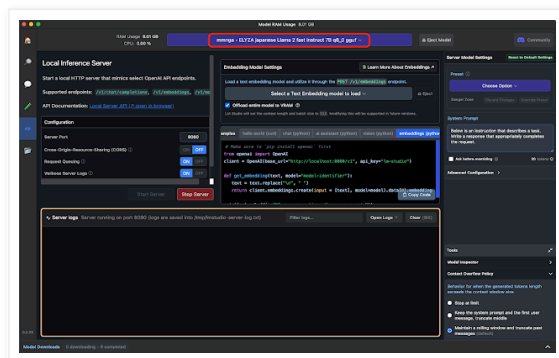
## DBMS\_VECTOR\_CHAIN.UTL\_TO\_SUMMARYと UTL\_TO\_GENERATE\_TEXTの動作を確認する

Oracle Database 23aiに追加されたPL/SQLパッケージDBMS\_VECTOR\_CHAINには、ドキュメントからサマリーを生成するUTL\_TO\_SUMMARYおよび文章を生成するUTL\_TO\_GENERATE\_TEXTという関数があります。

3rd PartyのプロバイダがCohereの場合は、CohereのAPIにsummarizeがあるので（ただし、すでにLegacyの扱い）それを呼び出すことは想定できるのですが、OpenAIのAPIにサマリーの生成はありません。

今回はLM StudioのOpenAI互換のChat Completions APIを、ファンクションUTL\_TO\_SUMMARYとUTL\_TO\_GENERATE\_TEXTから呼び出し、APIとしてどのようなリクエストが送信されているかを確認してみます。

LM Studioでは、モデルとしてELYZA japanese Llama 2 fast instruct 7B q8\_0 ggufを読み込みました。



SQLコマンドから以下のコードを実行します。

```
declare
    l_text clob;
    l_response clob;
begin
    l_text := q'~
吾輩わがはいは猫である。名前はまだ無い。
どこで生れたかとうと見当けんとうがつかぬ。何でも薄暗いじめじめした所でニャーニャー泣いていた事だけは記憶している。吾輩はここで
~';

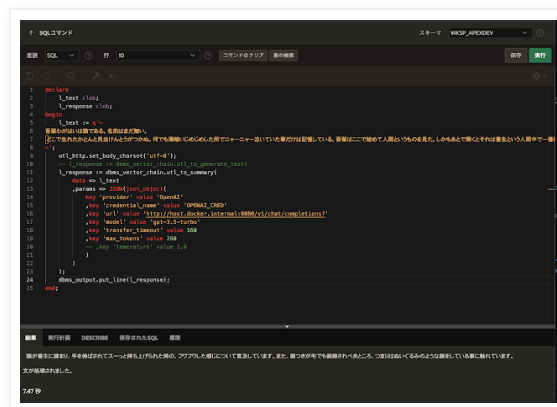
    utl_http.set_body_charset('utf-8');
    -- l_response := dbms_vector_chain.utl_to_generate_text(
```

```

l_response := dbms_vector_chain.utl_to_summary(
  data => l_text
  ,params => JSON(json_object(
    key 'provider' value 'OpenAI'
    ,key 'credential_name' value 'OPENAI_CRED'
    ,key 'url' value 'http://host.docker.internal:8080/v1/chat/completions?'
    ,key 'model' value 'gpt-3.5-turbo'
    ,key 'transfer_timeout' value 160
    ,key 'max_tokens' value 260
    -- ,key 'temerature' value 1.0
  )
)
);
dbms_output.put_line(l_response);
end;
```

utl-generate-text-summary.sql hosted with ❤ by GitHub

[view raw](#)



LM Studioに出力されたリクエストは以下でした。

プロバイダがOpenAIの場合は、roleが**system**のプロンプトとして**Generate a summary**を渡すことにより、後続のroleがuserのメッセージのサマリーを生成しているようです。

[2024-05-24 16:48:32.526] [INFO] Received POST request to /v1/chat/completions? with body: {

```

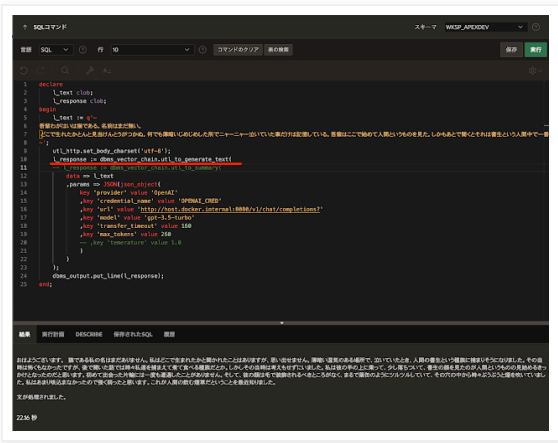
"max_tokens": 260,
"model": "gpt-3.5-turbo",
"messages": [
  {
    "role": "system",
    "content": "Generate a summary"
  },
  {
    "role": "user",
```

"content": "\n吾輩わがはいは猫である。名前はまだ無い。どこで生れたかとうと見当けんとうがつかぬ。何でも薄暗いじめじめした所でニャーニャー泣いていた事だけは記憶している。吾輩はここで始めて人間というものを見た。しかもあとで聞くとそれは書生という人間中で一番獰悪どうあくな種族であったそうだ。この書生というのは時々我々を捕つかまえて煮にて食うという話である。しかしその当時は何という考もなかったから別段恐いとも思わなかった。ただ彼の掌てのひらに載せられてスーと持ち上げられた時何だかフワフワした感じがあったばかりである。掌の上で少し落ちついて書生の顔を見たのがいわゆる人間とい

うものの見始めはじめてであろう。この時妙なものだと思った感じが今でも残っている。第一毛をもって装飾されべきはずの顔がつるつるしてまるで薬缶やかんだ。その後ご猫にもだいが逢あったがこんな片輪かたわには一度も出会えてくわした事がない。のみならず顔の真中があまりに突起している。そうしてその穴の中から時々ぶうぶうと煙けむりを吹く。どうも咽むせばくて実に弱った。これが人間の飲む煙草たばこというものである事はようやくこの頃知った。\\n"

```
}
]
}
```

コードに含まれるdbms\_vector\_chain.utl\_to\_summaryの行をコメントアウトし、代わりにdbms\_vector\_chain.utl\_to\_generate\_textを呼び出してみます。



utl\_to\_generate\_textの呼び出しでは、roleがsystemのプロンプトはYou are a helpful assistantになっています。

[2024-05-24 16:53:18.893] [INFO] Received POST request to /v1/chat/completions? with body: {

```
"max_tokens": 260,
"model": "gpt-3.5-turbo",
"messages": [
  {
    "role": "system",
    "content": "You are a helpful assistant"
  },
  {
    "role": "user",
```

"content": "\\n吾輩わがはいは猫である。名前はまだ無い。\\n どこで生れたかとうと見当けんとうがつかぬ。何でも薄暗いじめじめした所でニャーニャー泣いていた事だけは記憶している。吾輩はここで始めて人間というものを見た。しかもあとで聞くとそれは書生という人間中で一番獰悪どうあくな種族であったそうだ。この書生というのは時々我々を捕つかまえて煮にて食うという話である。しかしその当時は何という考もなかったから別段恐しいとも思わなかった。ただ彼の掌てのひらに載せられてスーと持ち上げられた時何だかフワフワした感じがあったばかりである。掌の上で少し落ちついて書生の顔を見たのがいわゆる人間というものの見始めはじめてであろう。この時妙なものだと思った感じが今でも残っている。第一毛をもって装飾されべきはずの顔がつるつるしてまるで薬缶やかんだ。その後ご猫にもだいが逢あったがこんな片輪かたわには一度も出会えてくわした事がない。のみならず顔の真中があまりに突起している。そうしてその穴の中から時々ぶうぶうと煙けむりを吹く。どうも咽むせばくて実に弱った。これが人間の飲む煙草たばこというものである事はようやくこの頃知った。\\n"

```
}
]
}
```

Cohere、GoogleAIといった他の3rd Partyのプロバイダについても、指定したプロバイダに合わせたリクエストになっている模様です。

OpenAIのChat Completions APIを呼び出すにあたって、systemプロンプトが固定というのは、扱いにくいと思います。

以下は、`apex_web_service.make_rest_request`を使って、同じリクエスト（systemプロンプトがGenerate a summary）を発行するコードの例です。

```
declare
    l_text clob;
    l_request_json json_object_t;
    l_request      clob;
    l_messages     json_array_t;
    l_message      json_object_t;
    l_response     clob;
    l_response_json json_object_t;
    l_choices      json_array_t;
    l_object       json_object_t;

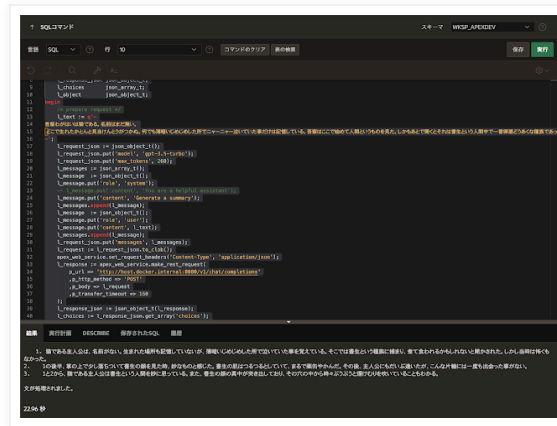
begin
    /* prepare request */
    l_text := q'~
吾輩わがはいは猫である。名前はまだ無い。
どこで生れたかとうと見当けんとうがつかぬ。何でも薄暗いじめじめした所でニャーニャー泣いていた事だけは記憶している。吾輩はここで
~';

    l_request_json := json_object_t();
    l_request_json.put('model', 'gpt-3.5-turbo');
    l_request_json.put('max_tokens', 260);
    l_messages := json_array_t();
    l_message := json_object_t();
    l_message.put('role', 'system');
    -- l_message.put('content', 'You are a helpful assistant');
    l_message.put('content', 'Generate a summary');
    l_messages.append(l_message);
    l_message := json_object_t();
    l_message.put('role', 'user');
    l_message.put('content', l_text);
    l_messages.append(l_message);
    l_request_json.put('messages', l_messages);
    l_request := l_request_json.to_clob();
    apex_web_service.set_request_headers('Content-Type', 'application/json');
    l_response := apex_web_service.make_rest_request(
        p_url => 'http://host.docker.internal:8080/v1/chat/completions'
        ,p_http_method => 'POST'
        ,p_body => l_request
        ,p_transfer_timeout => 160
    );
    l_response_json := json_object_t(l_response);
```

```
l_choices := l_response_json.get_array('choices');
l_object := treat(l_choices.get(0) as json_object_t);
l_message := l_object.get_object('message');
dbms_output.put_line(l_message.get_string('content'));
end;
```

openai-generate.sql hosted with ❤ by GitHub

[view raw](#)



完

Yuji N. 時刻: 17:10

共有



ホーム



[ウェブ バージョンを表示](#)

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](#)

Powered by Blogger.