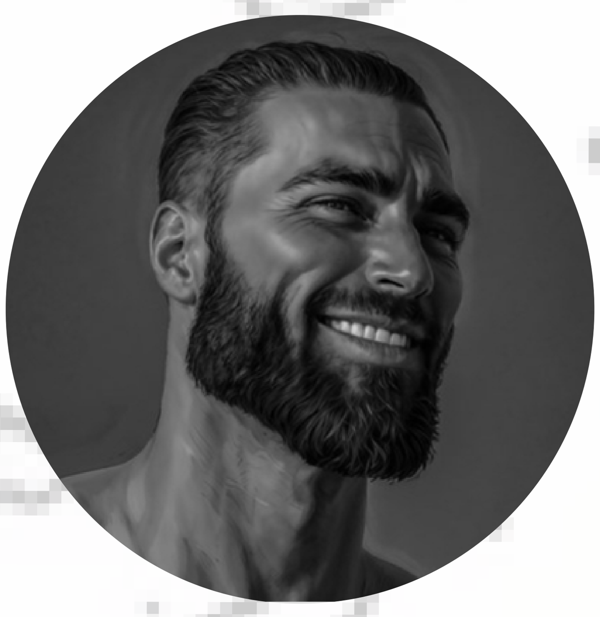
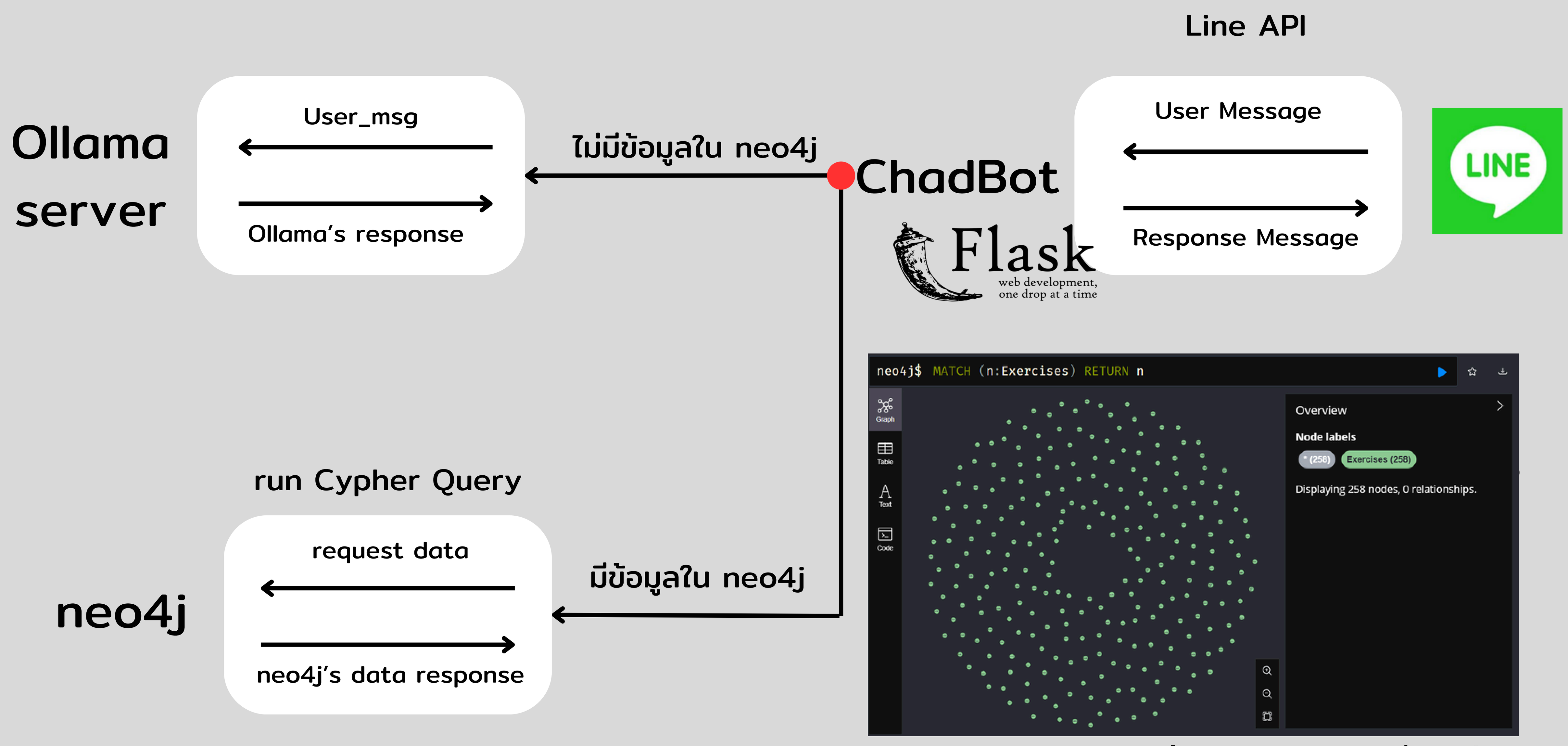


ChadBot Exercises

ที่มาและความสำคัญ

เนื่องจากปัจจุบันเริ่มมีคนหลายคนสนใจหันมาดูแลสุขภาพมากขึ้น แต่ไม่รู้ว่าควรเริ่มจากตรงไหน ต้องทำอย่างไรบ้าง เพราะไม่เคยมีประสบการณ์มาก่อน จึงเกิดเป็น ChadBot ที่คอยตอบคำถามและให้คำปรึกษาเกี่ยวกับการออกกำลังกายเพื่อการดูแลสุขภาพได้อย่างถูกต้องและมีประสิทธิภาพ

โครงสร้าง ChatBot



หมายเหตุ : รูปเหล่านี้เป็นเพียงแค่ส่วนหนึ่งเท่านั้น

การตั้งค่าเริ่มต้นและการเริ่มต้นการเชื่อมต่อ

Python Library

- faiss
- requests
- pyngrok import ngrok
- flask import Flask, request
- linebot import LineBotApi, WebhookHandler
- linebot.models import TextSendMessage
- sentence_transformers import SentenceTransformer
- neo4j import GraphDatabase
- json

NLP Model

- sentence-transformers/distiluse-base-multilingual-cased-v2

Configuration

Config ngrok

```
port = "5000"
ngrok.set_auth_token("<AUTH_TOKEN>")
public_url = ngrok.connect(port).public_url
# Open a ngrok tunnel to the HTTP server
print(f"ngrok tunnel {public_url} -> http://127.0.0.1:{port}")
```

Config Line

```
@app.route("/chat", methods=['POST'])
def linebot():
    body = request.get_data(as_text=True)
    try: # check if request from LINE
        json_data = json.loads(body)
        access_token = '<ACCESS_TOKEN>'
        secret = '<SECRET>'
        line_bot_api = LineBotApi(access_token)
        handler = WebhookHandler(secret)
        signature = request.headers['X-Line-Signature']
        handler.handle(body, signature)
        msg = json_data['events'][0]['message']['text']
        tk = json_data['events'][0]['replyToken']
```

Config Flask

```
app = Flask(__name__)
app.config["BASE_URL"] = public_url
app.config['JSON_AS_ASCII'] = False
```

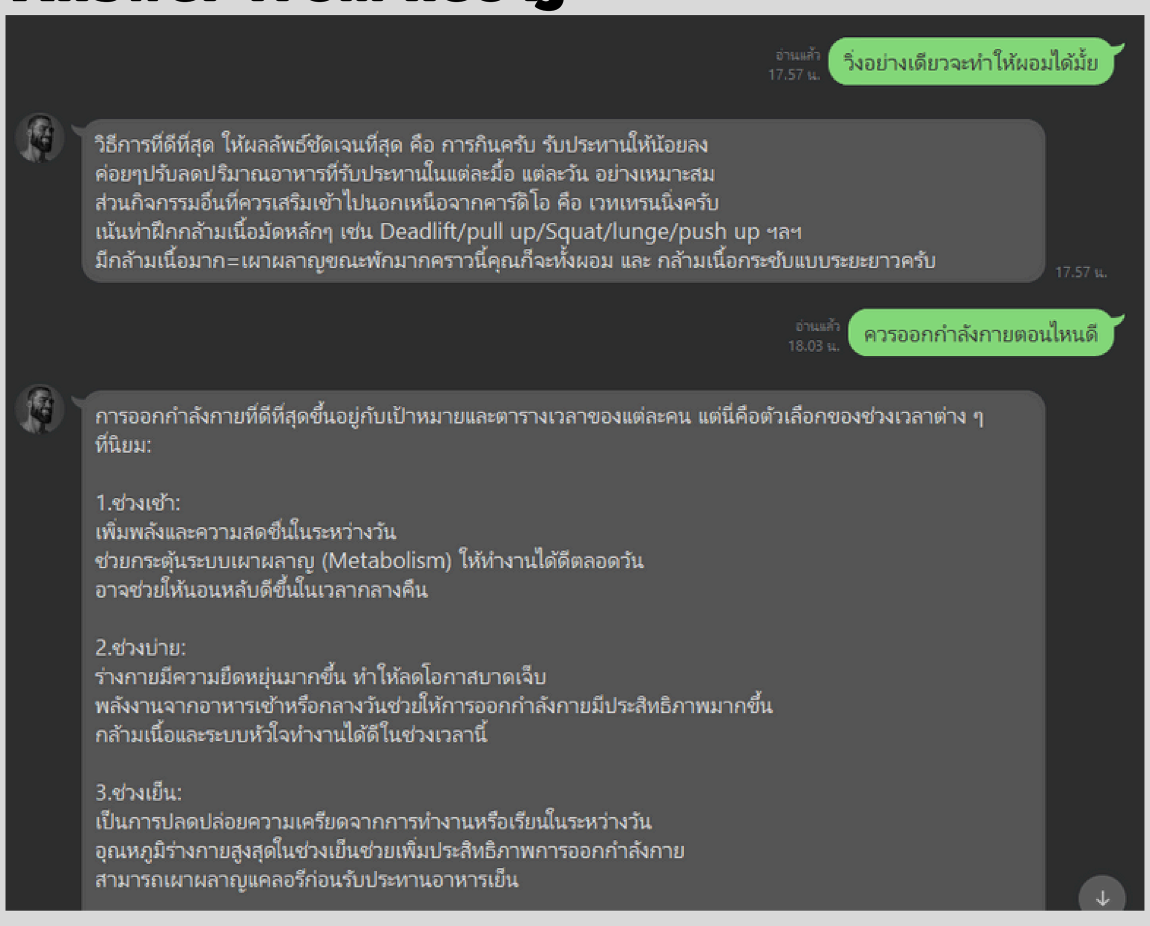
```
if __name__ == '__main__':
    app.run(port=5000)
```

Config neo4j with faiss

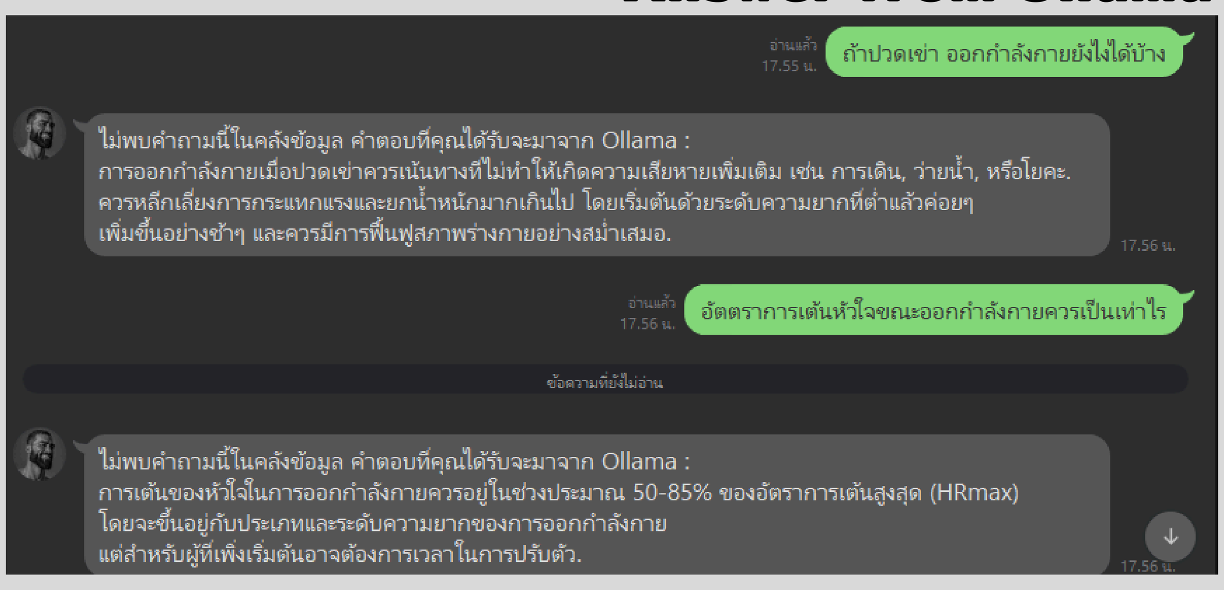
```
cypher_query = '''
MATCH (n) WHERE (n:Exercises OR n:Greeting OR n:Grateful OR n:Goodbye) RETURN n.name as name, n.msg_reply as reply;
'''
input_corpus = []
results = run_query(cypher_query)
for record in results:
    input_corpus.append(record['name'])
input_corpus = list(set(input_corpus))
print(input_corpus)

# Encode the input corpus into vectors using the sentence transformer model
input_vecs = model.encode(input_corpus, convert_to_numpy=True, normalize_embeddings=True)

# Initialize FAISS index
d = input_vecs.shape[1]  # Dimension of vectors
index = faiss.IndexFlatL2(d)  # L2 distance index (cosine similarity can be used with normalization)
index.add(input_vecs)  # Add vectors to FAISS index
```

ตัวอย่างการทำงาน

Answer from neo4j



Answer from Ollama



โอกาสในการพัฒนา

- เพิ่มข้อมูลในการตอบเกี่ยวกับโภชนาการ
- มีการเก็บประวัติการแชทของผู้ใช้เพื่อเพิ่มประสิทธิภาพในการตอบ/ให้ข้อมูลแก่ผู้ใช้
- พัฒนาเพิ่มประสิทธิภาพความเร็วในการตอบให้เร็วขึ้น

Python Code

https://github.com/Poadkee/ChadBot/tree/exercise