

Ứng Dụng Gợi Ý Phim với Flask

Tổng Quan Ứng Dụng

Ứng dụng này sử dụng Flask để xây dựng hệ thống gợi ý phim dựa trên mô hình đã được huấn luyện trước đó. Người dùng có thể tìm kiếm phim, xem gợi ý các phim liên quan, và truy cập các tài liệu liên quan.

Giải Thích Mã Nguồn

1 Cấu Hình và Tải Thư Viện

```
from flask import Flask, render_template, request, send_from_directory
import pandas as pd
import pickle
import os
import requests
from dotenv import load_dotenv

# Load API Key từ file .env
load_dotenv()
api_key = os.getenv("API_KEY")

app = Flask(__name__)
```

- **Flask:** Dùng để tạo ứng dụng web.
 - **dotenv:** Tải API Key từ file `.env`.
 - **pickle:** Tải dữ liệu mô hình đã lưu.
-

2 Tải Dữ Liệu

```
movies = pickle.load(open('model/movie_list.pkl', 'rb'))
similarity = pickle.load(open('model/similarity.pkl', 'rb'))

df_ratings = pd.read_csv('data/data.csv', encoding='utf-8')
df_ratings['vote_average'] = pd.to_numeric(df_ratings['vote_average'], errors='coerce')
df_ratings['vote_count'] = pd.to_numeric(df_ratings['vote_count'], errors='coerce')
df_ratings['runtime'] = pd.to_numeric(df_ratings['runtime'], errors='coerce')
```

- **Dữ liệu:**
 - `movie_list.pkl`: Danh sách phim.
 - `similarity.pkl`: Ma trận tương đồng giữa các phim.
-

3 Hàm Lấy Poster Phim

```
def fetch_poster(movie_id):
    if movie_id in poster_cache:
        return poster_cache[movie_id]

    url = f"https://api.themoviedb.org/3/movie/{movie_id}?api_key={api_key}&language=en-US"
    response = requests.get(url).json()
    poster_path = response.get('poster_path')
    poster_url = f"https://image.tmdb.org/t/p/w500/{poster_path}" if poster_path else "https://image.tmdb.org/t/p/w500/default.jpg"
    poster_cache[movie_id] = poster_url

    return poster_url
```

- **Mục Đích:** Lấy poster phim từ API TMDB.
-

4 Hàm Gợi Ý Phim

```
def recommend(movie):
    try:
        index = movies[movies['title'] == movie].index[0]
    except IndexError:
        return []

    distances = sorted(list(enumerate(similarity[index])), reverse=True, key=lambda x: x[1])
    recommended_movies = []

    for i in distances[1:6]:
        movie_id = movies.iloc[i[0]].id
        movie_title = movies.iloc[i[0]].title
        rating_info = df_ratings[df_ratings['id'] == movie_id].iloc[0] if movie_id in df_ratings.index else None

        recommended_movies.append({
            'title': movie_title,
            'poster_path': fetch_poster(movie_id),
            'vote_average': rating_info['vote_average'] if rating_info is not None else "N/A",
            'vote_count': rating_info['vote_count'] if rating_info is not None else "N/A",
            'runtime': rating_info['runtime'] if rating_info is not None else "N/A"
        })

    return recommended_movies
```

- **Mục Đích:** Trả về danh sách 5 phim gợi ý dựa trên phim người dùng chọn.
-

5 Các Route Chính

```
@app.route('/')
def home():
    return render_template('index.html', movie_list=movies['title'].tolist(), recommendation=recommendations)

@app.route('/recommend', methods=['POST'])
def recommend_movies():
    movie_name = request.form['movie_name']
    recommendations = recommend(movie_name)
    return render_template('index.html', movie_list=movies['title'].tolist(), recommendation=recommendations)

@app.route('/members')
def members():
    team_members = [
        {"MSSV": "22110158", "Hovaten": "Tran Chau Phu"},
        {"MSSV": "22110170", "Hovaten": "Ho Minh Quan"},
        {"MSSV": "22110123", "Hovaten": "Le Nguyen Duc Nam"},
        {"MSSV": "22110124", "Hovaten": "Le Thi Kim Nga"},
        {"MSSV": "22110155", "Hovaten": "Tran Nguyen Thanh Phong"},
    ]
    return render_template('members.html', members=team_members)

@app.route('/documents')
def documents():
    files = os.listdir('./static/doc')
    return render_template('documents.html', files=files)

@app.route('/doc/<filename>')
def serve_pdf(filename):
    return send_from_directory('./static/doc', filename)
```

- `/`: Trang chủ.
- `/recommend`: Trang gợi ý phim.
- `/members`: Danh sách thành viên.
- `/documents`: Tài liệu dự án.

Công Nghệ Sử Dụng

- Flask
- Pandas
- Pickle
- TMDB API

Tệp Quan Trọng

- `movie_list.pkl`
- `similarity.pkl`

Liên Hệ

- Email: `phutc04@gmail.com`

Ứng Dụng Đã Sẵn Sàng!