**Nguyễn Đức Toàn – 22110437**

**Trực quan hóa dữ liệu (xây dựng biểu đồ)**

* **Data**
* **books\_clean.jsonl**
* **tiki\_book\_data\_clean.jsonl**
* **templates:**
* **index.html**

<!DOCTYPE html>

<html>

<head>

<title>Trực quan hóa dữ liệu sách</title>

</head>

<body>

<h1>Trực quan hóa dữ liệu so sánh giữa 2 file JSONL</h1>

<div>

<h2>Phân bố giá sách</h2>

<img src="data:image/png;base64,{{ plots['price\_distribution'] }}" alt="Phân bố giá sách" />

</div>

<div>

<h2>Phân bố rating sách</h2>

<img src="data:image/png;base64,{{ plots['rating\_distribution'] }}" alt="Phân bố rating sách" />

</div>

<div>

<h2>Số lượng sách theo rating</h2>

<img src="data:image/png;base64,{{ plots['rating\_count'] }}" alt="Số lượng sách theo rating" />

</div>

</body>

</html>

**app.py**

import json

import pandas as pd

from flask import Flask, render\_template, request

import matplotlib.pyplot as plt

import seaborn as sns

import io

import base64

app = Flask(\_\_name\_\_)

def load\_jsonl(filepath):

data = []

with open(filepath, "r", encoding="utf-8") as f:

for line in f:

try:

data.append(json.loads(line))

except:

pass

return data

def prepare\_data(filepath):

data = load\_jsonl(filepath)

# Chuyển thành DataFrame, giữ các cột title, price, rating

df = pd.DataFrame(data)

df['price'] = pd.to\_numeric(df['price'], errors='coerce')

df['rating'] = pd.to\_numeric(df['rating'], errors='coerce')

return df

def plot\_to\_img(plt):

buf = io.BytesIO()

plt.savefig(buf, format='png', bbox\_inches='tight')

buf.seek(0)

img = base64.b64encode(buf.getvalue()).decode('utf-8')

plt.close()

return img

@app.route("/", methods=["GET", "POST"])

def index():

tiki\_df = prepare\_data("data/tiki\_book\_data\_clean.jsonl")

uk\_df = prepare\_data("data/books\_clean.jsonl")

plots = {}

**# Biểu đồ 1: Phân bố giá sách tiki và uk**

plt.figure(figsize=(10,5))

sns.kdeplot(tiki\_df['price'].dropna(), label='Tiki', fill=True)

sns.kdeplot(uk\_df['price'].dropna(), label='UK', fill=True)

plt.title("Phân bố giá sách")

plt.xlabel("Giá (VNĐ)")

plt.legend()

plots['price\_distribution'] = plot\_to\_img(plt)

**# Biểu đồ 2: Phân bố rating sách tiki và uk**

plt.figure(figsize=(10,5))

sns.kdeplot(tiki\_df['rating'].dropna(), label='Tiki', fill=True)

sns.kdeplot(uk\_df['rating'].dropna(), label='UK', fill=True)

plt.title("Phân bố rating sách")

plt.xlabel("Rating")

plt.legend()

plots['rating\_distribution'] = plot\_to\_img(plt)

**# Biểu đồ 3: Số lượng sách theo rating (countplot)**

plt.figure(figsize=(10,5))

sns.countplot(data=tiki\_df, x='rating', color='blue', alpha=0.5, label='Tiki')

sns.countplot(data=uk\_df, x='rating', color='orange', alpha=0.5, label='UK')

plt.title("Số lượng sách theo rating")

plt.xlabel("Rating")

plt.legend(['Tiki', 'UK'])

plots['rating\_count'] = plot\_to\_img(plt)

return render\_template("index.html", plots=plots)

if \_\_name\_\_ == "\_\_main\_\_":

app.run(debug=True)