CS 351: Artificial Intelligence

Semester Project: Fashion Recommendation System

Abdul Moiz Javaid (2021016)

Ishika Kirpalani (2021247)

Mahnoor Amir (2021276)

### 1. Introduction:

The Fashion Recommendation System is an AI-powered application designed to provide users with personalized fashion recommendations based on their preferences. It utilizes computer vision and machine learning techniques to analyze images of clothing items and suggest similar items from a database.

### 2. Objectives:

- To create a user-friendly interface for uploading images of clothing items.

- To extract features from the uploaded images using a pre-trained deep learning model.

- To recommend visually similar fashion items based on the extracted features.

- To enhance user experience by providing real-time recommendations.

### 3. Technologies Used:

|  |  |
| --- | --- |
| Python | Programming language used for development. |
| Streamlit | Open-source library for building interactive web applications. |
| TensorFlow | Deep learning framework for building and training neural networks. |
| Scikit-learn | Machine learning library for implementing k-nearest neighbors algorithm. |
| NumPy | Library for numerical computing in Python. |
| PIL (Python Imaging Library) | Library for image processing tasks. |
| Pickle | Module for serializing and deserializing Python objects. |

### 4. Project Structure:

The project consists of the following components:

- **Model:** Utilizes a pre-trained ResNet50 model for feature extraction and a pickled file containing pre-computed image embeddings and filenames.

- **Data:** Contains pre-computed image embeddings and filenames stored in pickle files.

- **Main Script (main.py or app.py):** Streamlit application script for running the recommendation system.

- **Uploads Directory:** Stores uploaded images temporarily.

### 5. Workflow:

1. **User Interface:** The Streamlit application presents a user-friendly interface for uploading images.

2. **Image Upload:** Users upload an image of a clothing item they wish to find similar items for.

3. **Feature Extraction:** The uploaded image is processed to extract its features using the pre-trained ResNet50 model.

Figure 1: File Uploader and Recommendation Display

4. **Recommendation:** Based on the extracted features, the system retrieves visually similar images from the dataset using k-nearest neighbors algorithm.

5. **Display:** The recommended images are displayed to the user in a visually appealing manner.

6. **Feedback:** Users can view and interact with the recommended images to explore further options.

### 6. Future Enhancements:

- Integration with e-commerce platforms for real-time product recommendations.

- Incorporation of user feedback to improve recommendation accuracy.

- Implementation of advanced deep learning models for better feature extraction.

- Deployment on cloud platforms for scalability and accessibility.

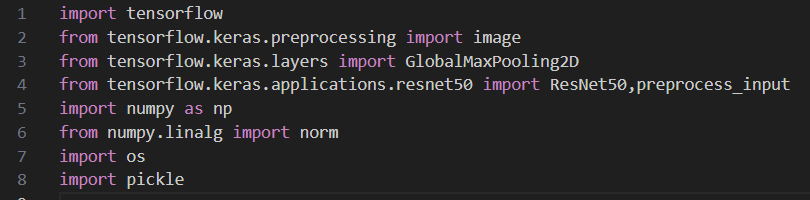
### 7. Conclusion:

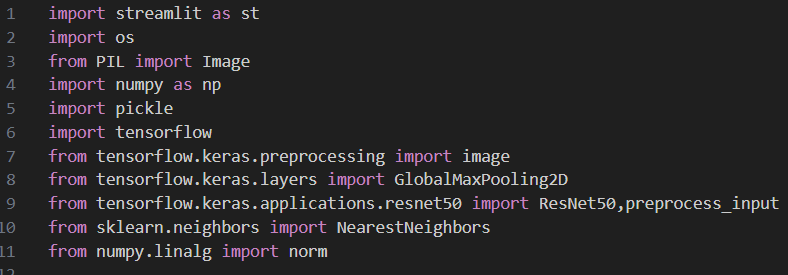
The Fashion Recommendation System project demonstrates the effective use of computer vision and machine learning techniques to provide personalized fashion recommendations to users. With its intuitive interface and accurate recommendations, the system aims to enhance the shopping experience and assist users in discovering new fashion trends.

### 8. Code:

* **Imports:**

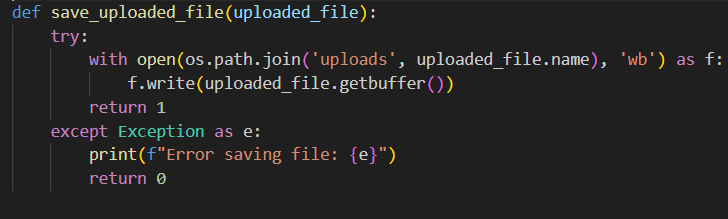
- The script imports necessary libraries and modules such as Streamlit, PIL (Python Imaging Library), NumPy, TensorFlow, Pickle, and others.



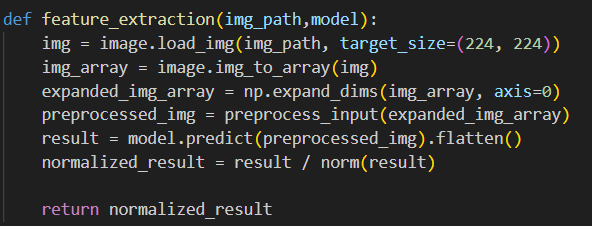


* **Functions:**

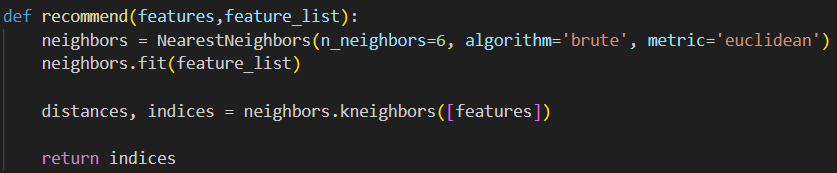
- save\_uploaded\_file: Saves the uploaded file to a directory.



- feature\_extraction: Extracts features from an image using the pre-trained ResNet50 model.

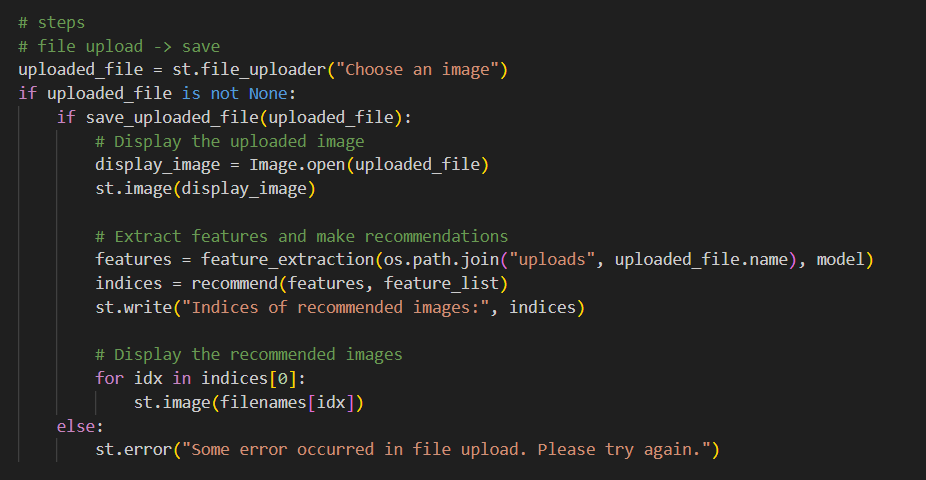


- recommend: Recommends similar images based on the extracted features using k-nearest neighbors algorithm.



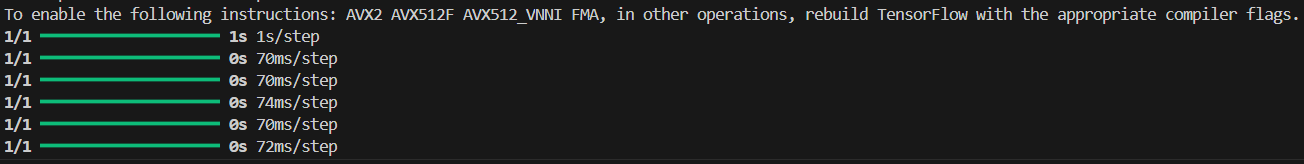
* **Error Handling:**

- Handles errors during file upload and displays an error message if an issue occurs.

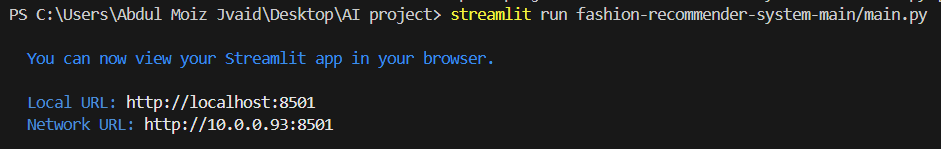


### 9. Outputs:

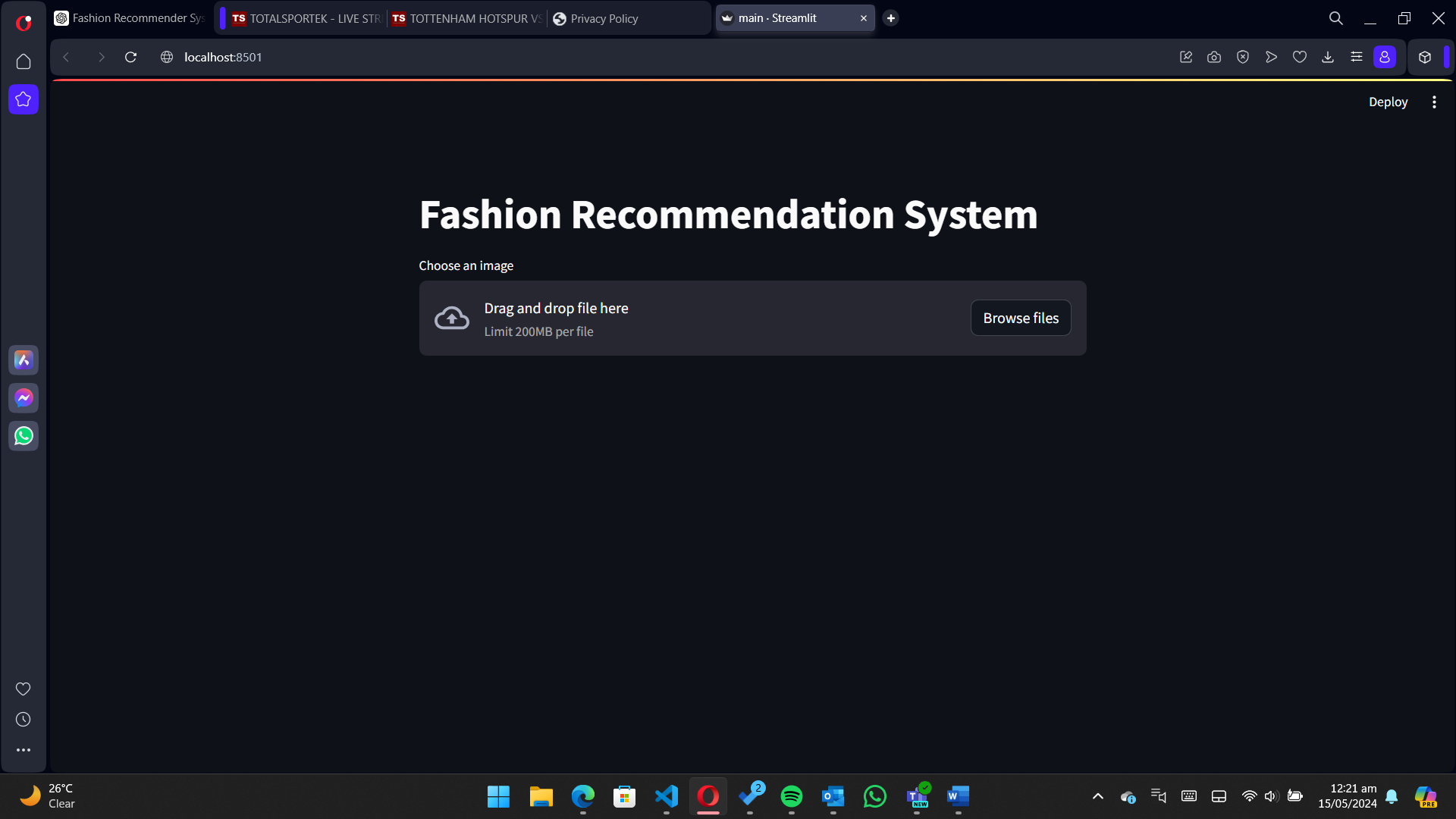
Images being preprocessed and images are being converted to numerical values using ResNet50:



Starting up the local website:



Website from streamlit:



After uploading image from the learned dataset we get 5 recommended:

