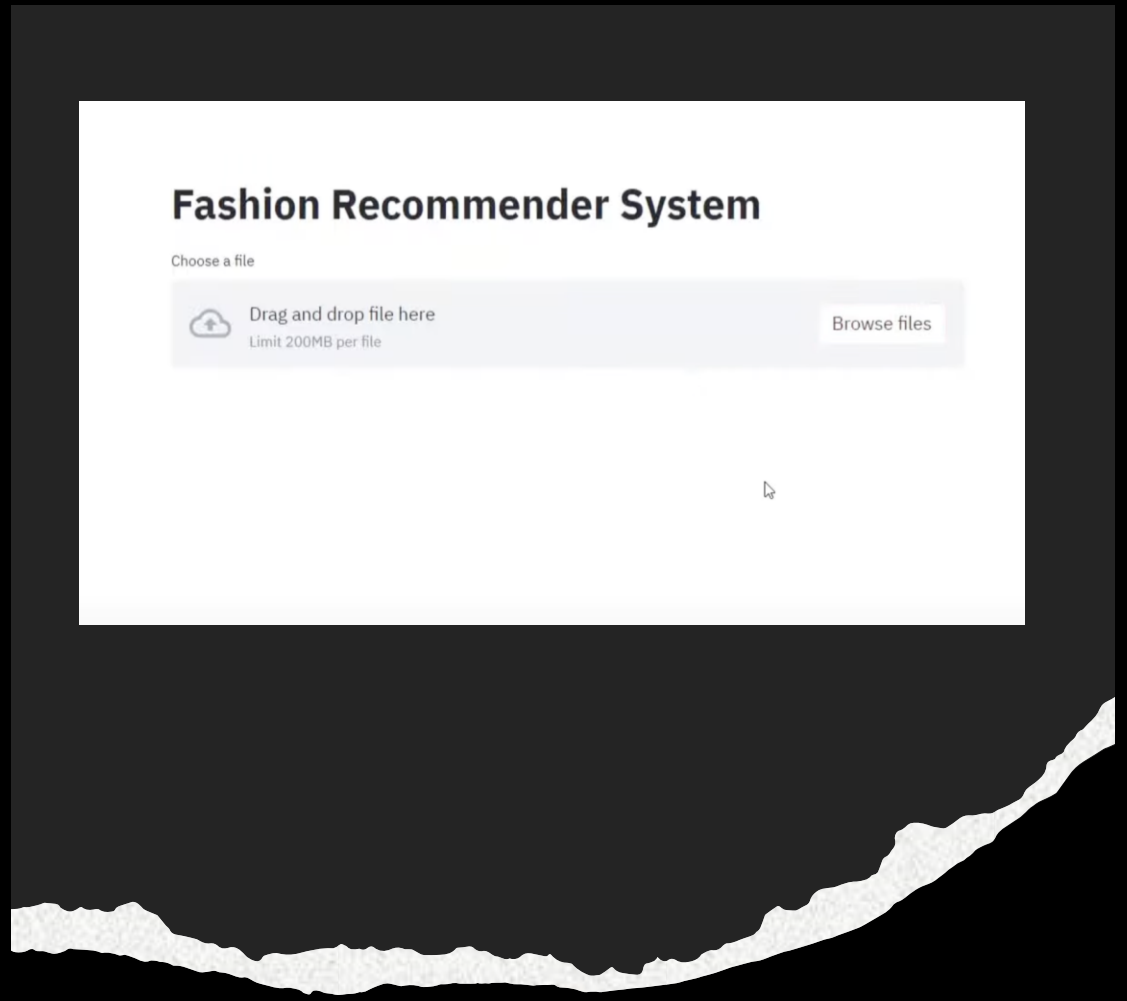
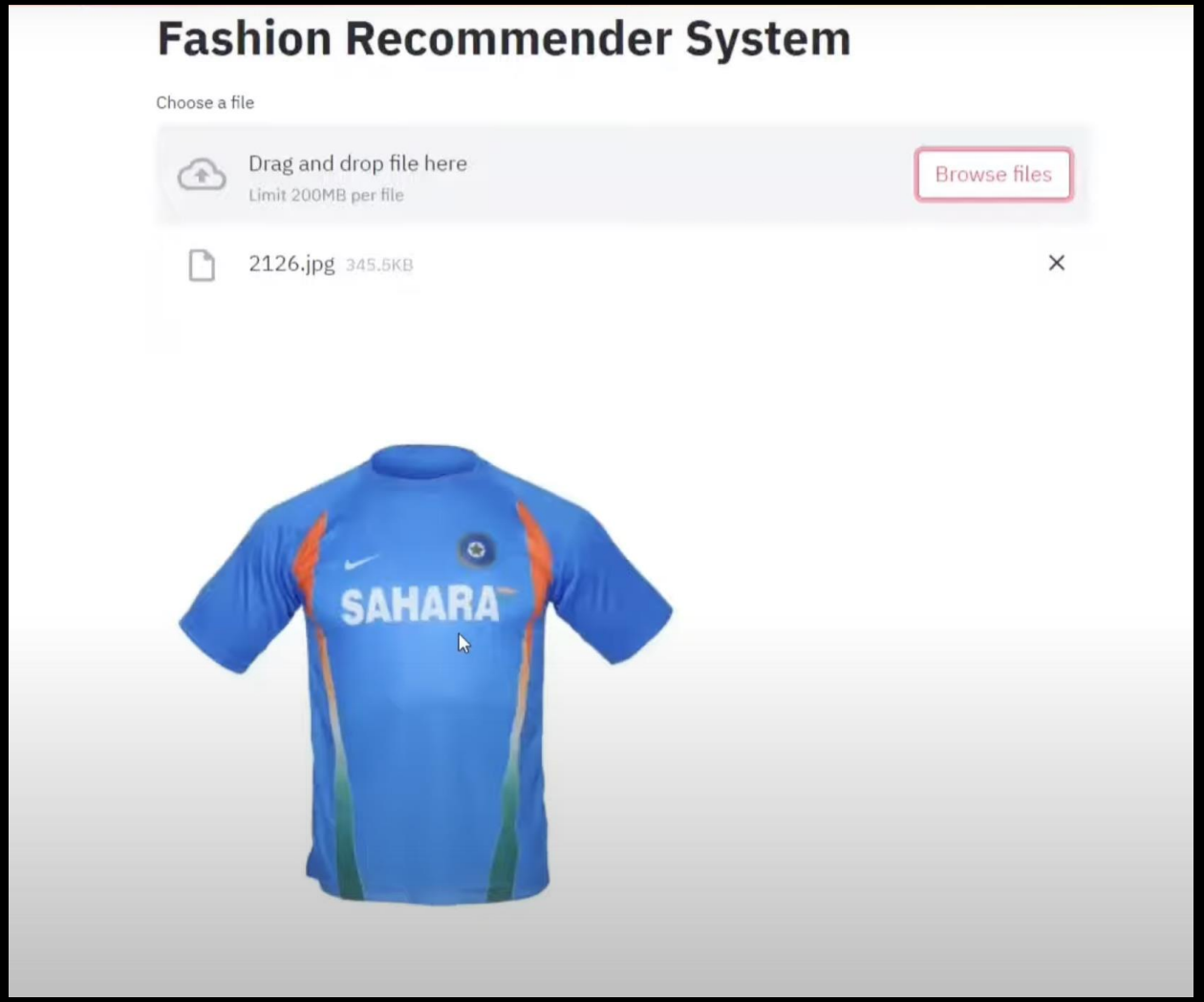
Deep Learning based Fashion Recommender System using ResNet50



A Deep Learning based Fashion Recommender System using ResNet50



Introduction

- **Objective**
- Develop a user-friendly Fashion Recommender System using ResNet50 to enhance the online shopping experience.
- **Benefits:**
- **Personalized Recommendations:** Provide users with personalized fashion suggestions based on their preferences.
- **Increased User Engagement:** Encourage users to explore a wider range of fashion items, boosting engagement.
- **Enhanced Shopping Experience:** Simplify the decision-making process for users, making online fashion shopping more enjoyable.
- **Improved Customer Satisfaction:** Increase user satisfaction by offering relevant and appealing fashion choices.

Background

- **Why Recommender Systems?**

  - Enhance user experience in online fashion shopping.
  - Increase user engagement and satisfaction.
  - Streamline fashion discovery for effortless and enjoyable online shopping.
  - **Increase user engagement and satisfaction:**
  - Elevate user interaction and happiness through personalized fashion recommendations.

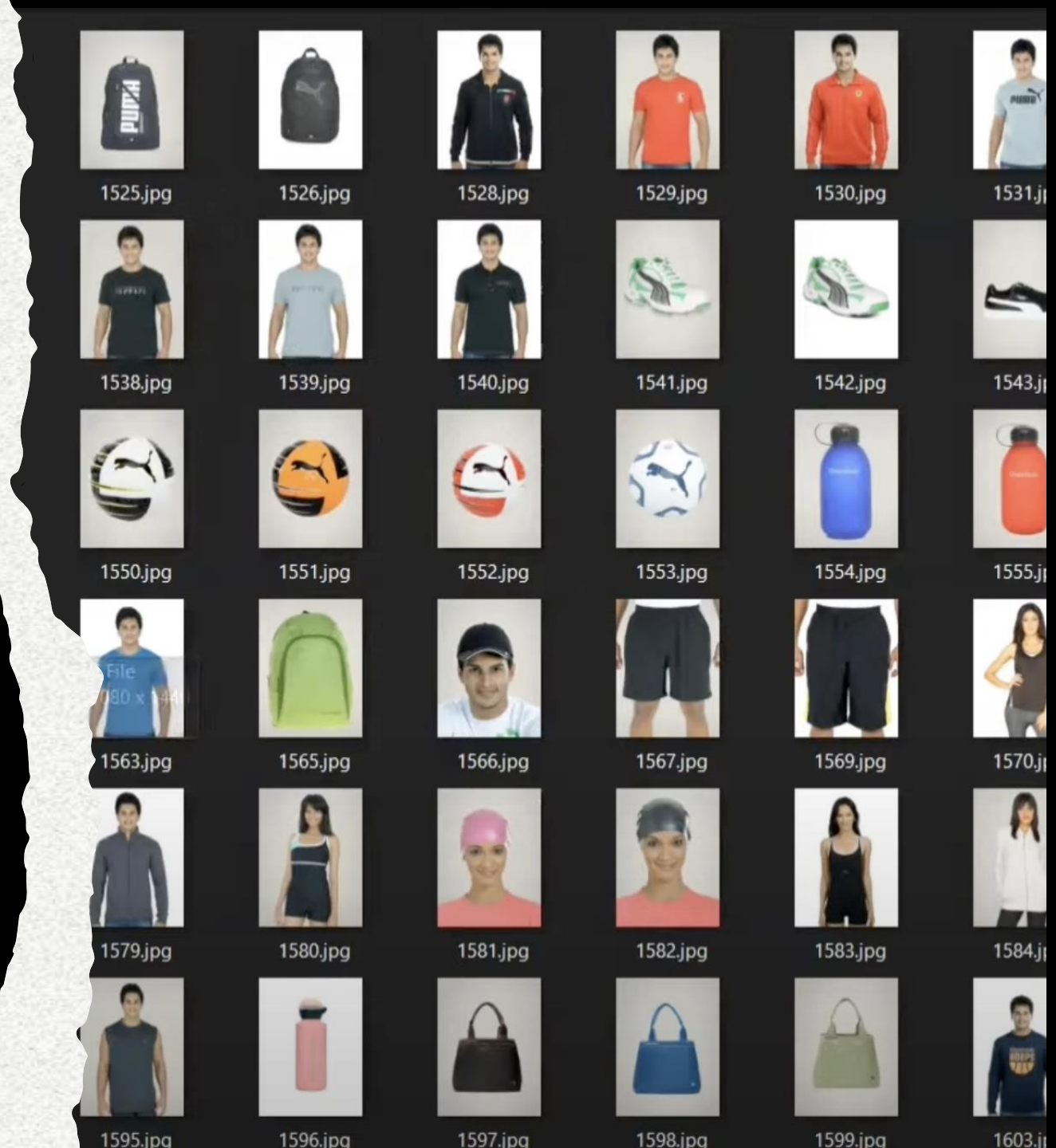
Architecture Overview

Components

- **Training Script (app.py)**
- **Testing Script (test.py)**
- **Main Application (main.py)**

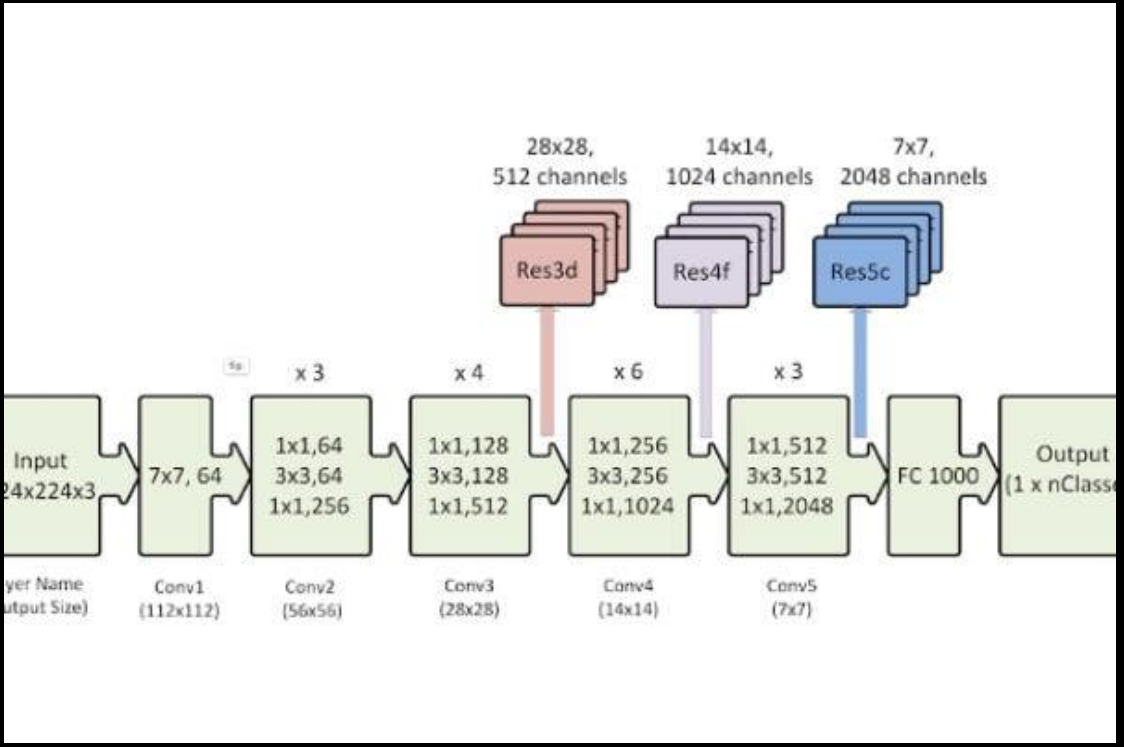
- **Dataset :** Fashion Product Images (Small)
- **Kaggle Dataset Link**

- **Google Colaboratory for Processing.**



Model Architecture

- **ResNet50**
- Pre-trained on ImageNet.
- Features extraction using GlobalMaxPooling2D.



- **Deep Representation:**
- **Skip Connections:**
- **State-of-the-Art Performance:**
- **Transfer Learning Capability:**
- **Global Receptive Field:**
- **Computational Complexity:**
- **Overfitting Tendency:**
- **Large Model Size:**
- **Training Time:**
- **Longer training times compared to shallower architectures, impacting rapid prototyping.**
- **Not Interpretable:**

- # Training Script (app.py)
- **Code Overview**
  - **Load pre-trained ResNet50.**
  - **Extract features from fashion images.**
  - **Save features and filenames using pickle.**
- # Testing Script (test.py)
- **Code Overview**
  - **Load pre-trained ResNet50.**
  - **Extract features from a test image.**
  - **Use Nearest Neighbors to find similar fashion items.**
- # Main Application (main.py)
- **Code Overview**
  - **Utilize Streamlit for the user interface.**
  - **Upload an image for recommendation.**
  - **Extract features and find similar items.**

- **Step-by-Step Demonstration**
  Upload an image.
  Extract features.
  Display recommended fashion items.

Fashion Recommender System

Choose a file

Drag and drop file here

Limit 200MB per file

Browse files

Uploaded Image.

Fashion Recommender System

Choose a file

Drag and drop file here
Limit 200MB per file

Browse files

2126.jpg 345.5KB

Conclusion

Key Takeaways

- **Deep Learning-based Recommender System.**
- **Utilizing ResNet50 for feature extraction.**
- **Streamlit for a user-friendly interface.**

Potential Improvements

Incorporate user feedback for personalized recommendations.

Fine-tune the model for specific fashion domains.

Thank You!

Contact Information
Akshat Chauhan
akshatchauhan777@gmail.com
JIMS Engineering Management Technical Campus