CSE4077- Recommender Systems

J Component
Final Project Report

FASHION RECOMMENDATION SYSTEM

By

19MIA1014	LOKESH KANNA
19MIA1028	NITHYA SHARMA
19MIA1053	YUVASHREE R
19MIA1083	K NIHARIKA SAMYUKTHA

M.Tech CSE with Specialization in Business Analytics

Submitted to

Dr.A.Bhuvaneswari,
Assistant Professor Senior,
SCOPE, VIT, Chennai

School of Computer Science and Engineering

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that this project report entitled “Fashion Recommendation System” is a bonafide work of **Lokesh Kanna – 19MIA1014, Nithya Sharma – 19MIA1028, Yuvashree R – 19MIA1053, K Niharika Samyuktha – 19MIA1083** who carried out the J-component under my supervision and guidance. The contents of this Project work, in fullor in parts, have neither been taken from any other source nor have been submitted to any other Institute or University for award of any degree or diploma and the same is certified.

Dr.A.Bhuvaneswari,

Assistant Professor Senior,

SCOPE, VIT, Chennai

ABSTRACT

In this project, we have proposed a personalized Fashion Recommender system that generates recommendations for the user based on an input given. Unlike the conventional systems that rely on the user's previous purchases and history, this project aims at using an image of a product given as inputby the user to generate recommendations since many-a-time people see something that they are interested in and tend to look for products that are like that. We have built a fashion recommendation system capable of learning a person's clothing style and preferences by extracting a variety of attributesfrom his/her clothing images. We use convolutional neural networks to process the images from Fashion Product Images Dataset. These attributes are then fed to a similarity model to retrieve closest similar images as recommendations. We have used Model-based recommendation systems that involves building a model based on the dataset. In other words, we extract some information from the dataset, and use that as a "model" to make recommendations without having to use the complete dataset every time. Feature extraction is done using pre-trained models (CNN based architecture) such as VGG, Resnet and Densenet. CNN model is used to identify the closer images and transfer learning model is used to improve accuracy and speed.

ACKNOWLEDGEMENT

We wish to express our sincere thanks and deep sense of gratitude to our project guide, **Dr. A. Bhuvaneswari Assistant** Professor, School of Computer Science Engineering, for his consistent encouragement and valuable guidance offered to us in a pleasant manner throughout the course of the project work.

We express our thanks to our HOD **Sivabalakrishnan M** for his support throughout the course of this project.

We also take this opportunity to thank all the faculty of the school for their support and their wisdom imparted to us throughout the course.

We thank our parents, family, and friends for bearing with us throughout the course of our project and for the opportunity they provided us in undergoing this course in such a prestigious institution.

Lokesh Kanna – 19MIA1014
Nithya Sharma – 19MIA1028
Yuvashree R – 19MIA1053
K Niharika Samyuktha – 19MIA1083

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Computing Science and Engineering

VIT Chennai

Vandalur - Kelambakkam Road, Chennai - 600 127

FALL SEM 22-23

Worklet details

Programme	Int.M.Tech with Specialization in Business Analytics	
Course Name / Code	CSE4077	
Slot	E1+TE1	
Faculty Name	**Bhuvaneswari Anbalagan**	
Component	J – Component	
J Component Title	FASHION RECOMMENDATION SYSTEM	
Team Members Name | Reg. No	19MIA1014	LOKESH KANNA
	19MIA1028	NITHYA SHARMA
	19MIA1053	YUVASHREE R
	19MIA1083	K NIHARIKA SAMYUKTHA

Team Members(s) Contributions – Tentatively planned for implementation:

Worklet Tasks	*Contributor's Names*
Database Collection	Niharika, Nithya
Preprocessing	Lokesh, Yuvashree
Model building	Niharika, Yuvashree, Lokesh, Nithya
Visualization	Yuvashree, Lokesh
Technical Report writing	Lokesh, Niharika
Presentation preparation	Yuvashree, Niharika, Nithya

TABLE OF CONTENTS

1. INTRODUCTION

With the introduction of recommender systems in multiple domains, retail industries are coming forward with investments in the latest technology to improve their business. Fashion has been in existence since centuries and will be prevalent in the coming days as well. The textile and apparel industry have grown tremendously over the last years. With an increase in thestandard of living, peoples' attention gradually moved towards fashion which is concerned to be a popular aesthetic expression. Humans are inevitably drawn towards something that is visually more attractive.This tendency of humans has led to the development of the fashion industry over the course of time. Women are more correlatedwith fashion and style, and they have a larger product base to deal with, making it difficult to make decisions. Moreover, apparel providers need their customers to explore their entire product line so that they can choose what they like the most which is not possible by simply going into a cloth store. Customers no longer have to visit many stores, stand in long queues, or try on garments in dressing rooms as millions of products are now available in online catalogs. However, given the plethora of options available on the e-commerce websites presents new challenges to the customers in identifying their correct outfit.

An effective recommendation system is necessary to properly sort, order, and communicate relevant product material or information to users. Recommender systems help users navigate large collections of products to find items relevant to theirinterests leveraging large amounts of product information and user signals like product views, followed or ignored items, purchases or web-page visits to determine how, when and what to recommend to theircustomers. The main challengein building a fashion recommendation system is that it is a very dynamic industry. It changes very oftenwhen it comes to seasons, festivals, pandemic conditions and many more. To deal withthe aforementioned problems, and given the visual and aesthetic nature of fashion products, there is a growing body of computer vision research addressing tasks like localizing fashion items, determining their category and attributes or establishing the degree of similarity to other products, to name only a few. Although works in the computer vision literature often don’t consider personalization (or recommendation), their predictions and embeddings can be leveraged by recommender systems and combined with past user preferences, thus mitigating sparsity and cold start problems.

2. LITERATURE SURVEY

Sl no	Title	Author / Journal name / Year	Technique	Result
1	Image-Based Recommendations on Styles and Substitutes	McAuley, Julian & Targett, Christopher & Shi, Qinfeng & Hengel, Anton. (2015).	Content based recommender systems which attempt to model each user's preference toward types of goods	Accuracy of link prediction on subcategories of 'Clothing, Shoes, and Jewelry' with increasing rank K
2	Fashion Recommendation Systems, Models and Methods: A Review	Samit Chakraborty, Md. Saiful Hoque, Naimur Rahman Jeem, Manik Chandra Biswas, Deepayan Bardhan and Edgar Lobaton.	Content-Based Filtering (CBF) Technique and Collaborative Filtering (CF) Technique	KNN showed the maximum accuracy more than 90% in various splits.
3	Size Recommendation System for Fashion E-commerce	G. Mohammed Abdulla, Sumit Borar	Combined both latent and observable features to make the ensemble model	Precision and coverage for both models as per the categories were found.
4	Product Recommendation based on Shared Customer's Behavior	Rodrigues F, Bruno Ferreira (2016), Procedia Computer Science.	Clustering — Association rule mining	Increases 96% the average value of the sales when compared with based recommendation
5	Recommender Systems Using Support Vector Machines	Min SH., Han I. (2005) In: Lowe D., Gaedke M.(eds) Web Engineering. ICWE 2005. Lecture Notes in Computer Science, vol 3579. Springer, Berlin, Heidelberg	Support Vector Machines Genetic Algorithm	Using McNemar's test, Support Vector Machines —Genetic Algorithm model shows better performance than the SVM and Traditional models
6	Collaborative Filtering Recommendation Algorithm Based on Item Clustering and Global Similarity	Wei S, Ye N, Zhang S, Huang X and Zhu J (2012) Fifth International Conference on Business Intelligence and Financial Engineering, Lanzhou.	Clustering K-means	The model can improve the accuracy of the prediction and enhance the recommendation Quality

7	Personal recommendation using deep recurrent neural networks in NetEase	Wu S, Ren W, Yu C, Chen G, Zhang D and Zhu J (2016) IEEE 32nd International Conference on Data Engineering (ICDE), Helsinki, pp.	RNN	It extracts the common purchase patterns and shorten the purchase path for future users, reaching a compression ratio of 0.724127 and an accuracy of 0.331312
8	Fashion Coordinates Recommender System Using Photographs from Fashion Magazines	Tomoharu Iwata Shinji Watanabe Hiroshi Sawada NTT Communication Science Laboratories 2-4 Hikaridai, Seika-cho, Soraku-gun, Kyoto, Japan	They used a probabilistic topic model for learning information about coordinates from visual features in each fashion item region	This method can learn information about coordinates appropriately, and can be used for recommending fashion coordinates., it is valuable to recommend coordinates that match the user's preference and situation, such as business, dating or formal.
9	Personalized fashion recommender system with image based neural networks	M Sridevi1, N ManikyaArun1, M Sheshikala2 and E Sudarshan3 Published under license by IOP Publishing Ltd 9-10 October 2020	Used neural networks to process the images from Deep Fashion dataset and a nearest neighbor backed recommender to generate the final recommendations	The training results show a 98% accuracy of the model with low error, loss and good f-score.
10	Fashion Recommendation System	Aneesh K, P. V. Rohith Kumar, Sai Uday Nagula, Archana Nagelli4	The images are classified based on the outfit type and color using CNN. Inception B3.	The model recommends the most suitable outfit combination using a recommendation algorithm (shows an accuracy of 86%)

3. DATASETS AND TOOLS

<u>Dataset:</u>

Dataset name: Fashion Product Images (Small)

- *images.csv*
- *styles.csv* This dataset contains 10 attributes with simple meaning and which are described as follows:

1. id - That is the serial number
2. gender
3. masterCategory
4. subcategory
5. articleType
6. baseClour
7. season
8. year
9. usage
10. productDisplayName

The image.csv and styles.csv will be combined before further processing and research.

The growing e-commerce industry presents us with a large dataset waiting to be scraped and researched upon. In addition to professionally shot high resolution product images, we also have multiple label attributes describing the product which was manually entered while cataloging. To add to this, we also have descriptive text that comments on the product characteristics.

Each product in this dataset is identified by an ID like 42431. There is a map to all the products in styles.csv. From here, we use the image for this product from images/42431.jpg and the complete metadata from styles/42431.json.

Since the dataset is too large, we started with a smaller (280MB) version here:
https://www.kaggle.com/paramaggarwal/fashion-product-images-small

Sample Data

	id	gender	masterCategory	subCategory	articleType	baseColour	season	year	usage	productDisplayName	image
0	15970	Men	Apparel	Topwear	Shirts	Navy Blue	Fall	2011	Casual	Turtle Check Men Navy Blue Shirt	15970.jpg
1	39386	Men	Apparel	Bottomwear	Jeans	Blue	Summer	2012	Casual	Peter England Men Party Blue Jeans	39386.jpg
2	59263	Women	Accessories	Watches	Watches	Silver	Winter	2016	Casual	Titan Women Silver Watch	59263.jpg
3	21379	Men	Apparel	Bottomwear	Track Pants	Black	Fall	2011	Casual	Manchester United Men Solid Black Track Pants	21379.jpg
4	53759	Men	Apparel	Topwear	Tshirts	Grey	Summer	2012	Casual	Puma Men Grey T-shirt	53759.jpg
5	1855	Men	Apparel	Topwear	Tshirts	Grey	Summer	2011	Casual	Inkfruit Mens Chain Reaction T-shirt	1855.jpg
6	30805	Men	Apparel	Topwear	Shirts	Green	Summer	2012	Ethnic	Fabindia Men Striped Green Shirt	30805.jpg
7	26960	Women	Apparel	Topwear	Shirts	Purple	Summer	2012	Casual	Jealous 21 Women Purple Shirt	26960.jpg
8	29114	Men	Accessories	Socks	Socks	Navy Blue	Summer	2012	Casual	Puma Men Pack of 3 Socks	29114.jpg
9	30039	Men	Accessories	Watches	Watches	Black	Winter	2016	Casual	Skagen Men Black Watch	30039.jpg

TOOLS:

- Google Collaboratory for exploratory data analysis, preprocessing / cleaning, model planning, model building, evaluation and visualization.

4. METHODOLOGIES AND ALGORITHMS USED:

Tensorflow

Dataflow graphs—structures that depict how data flows across a graph, or a collection of processing nodes—can be created by developers using TensorFlow. A mathematical operation is represented by each node in the graph, and each edge between nodes is a multidimensional data array, or tensor.

One may adopt best practices for data automation, model tracking, performance monitoring, and model retraining with the help of the TensorFlow platform. Success depends on the use of production-level technologies to automate and monitor model training over the course of a good, service, or business process.

Keras

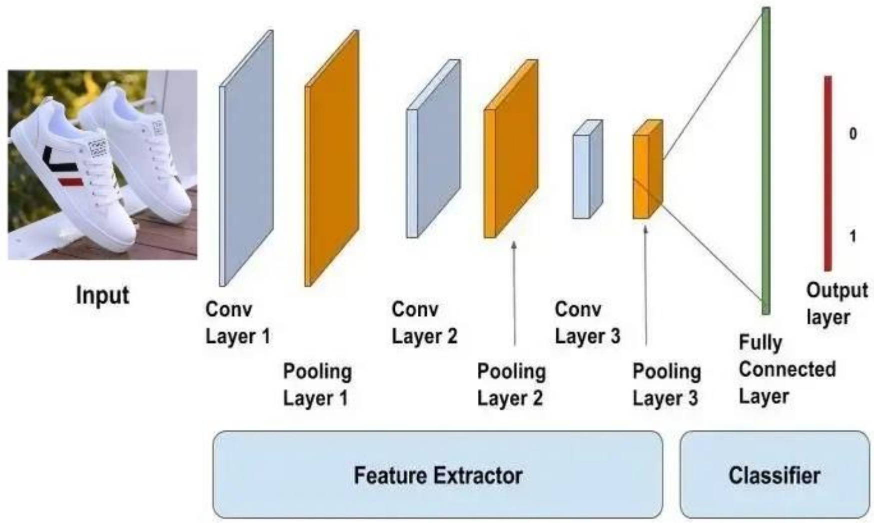
Keras is the high-level API of TensorFlow 2: an approachable, highly-productive interface for solving machine learning problems, with a focus on modern deep learning. It provides essential abstractions and building blocks for developing and shipping machine learning solutions with high iteration velocity.

Feature Extraction

Feature extraction refers to the process of transforming raw data into numerical features that can be processed while preserving the information in the original data set. It yields better results than applying machine learning directly to the raw data.

CNN

A Convolutional Neural Network or CNN is a type of artificial neural network, which is widely used for image/object recognition and classification. Deep Learning thus recognizes objects in an image by using a CNN. CNNs make use of convolution layers that utilize filters to help recognize the important features in an image.



Transfer Learning

Transfer learning is the application of knowledge gained from completing one task to help solve a different, but related, problem. In other words, Transfer Learning is the reuse of a pre-trained model on a new problem. It's currently very popular in deep learning because it can train deep neural networks with comparatively little data. Transfer Learning takes a model trained on a large dataset and transfer its knowledge to a smaller dataset.

1)VGG16

VGG-16 is a convolutional neural network that is 16 layers deep. One can load a pretrained version of the network trained on more than a million images from the ImageNet database. The pretrained network can classify images into 1000 object categories, in this case fashion related objects/images such as clothes, shoes, etc.

2)ResNet-50

ResNet-50 is a convolutional neural network that is 50 layers deep. You can load a pretrained version of the network trained on more than a million images from the ImageNet database. The pretrained network can classify images into 1000 object categories, in this case fashion related objects/images such as clothes, shoes, etc.

3)Densenet121

DenseNet is a convolutional neural network where each layer is connected to all other layers that are deeper in the network, that is, the first layer is connected to the 2nd, 3rd, 4th and so on, the second layer is connected to the 3rd, 4th, 5th and so on.

Embedding

An embedding is a relatively low-dimensional space into which you can translate high-dimensional vectors. Embeddings make it easier to do machine learning on large inputs like sparse vectors representing words. Ideally, an embedding capture some of the semantics of the input by placing semantically similar inputs close together in the embedding space. An embedding can be learned and reused across models.

So, a natural language modelling technique like Word Embedding is used to map words or phrases from a vocabulary to a corresponding vector of real numbers. The vector representation has two important and advantageous properties:

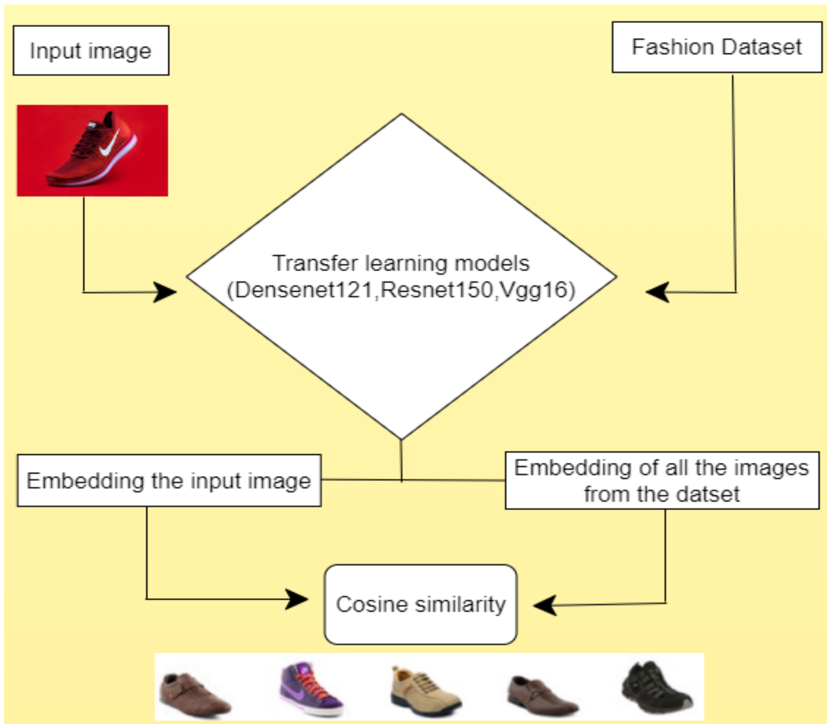
- Dimensionality Reduction—it is a more efficient representation
- Contextual Similarity—it is a more expressive representation

Cosine Similarity

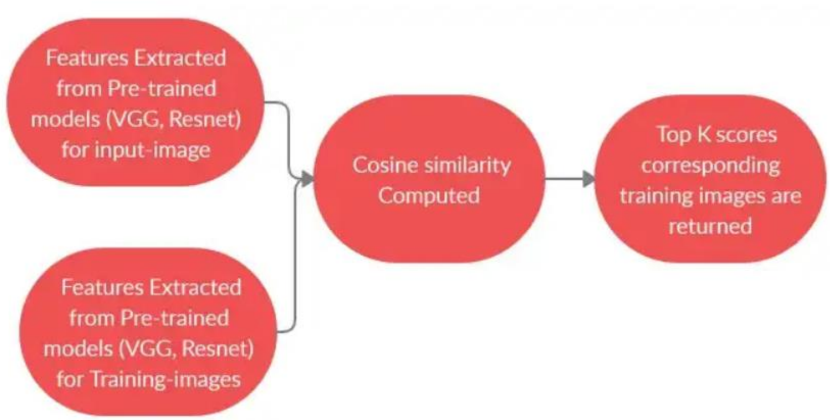
Cosine similarity measures the similarity between two vectors of an inner product space. It is measured by the cosine of the angle between two vectors and determines whether two vectors are pointing in roughly the same direction. It is often used to measure document similarity in text analysis.

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}},$$

5. IMPLEMENTATION

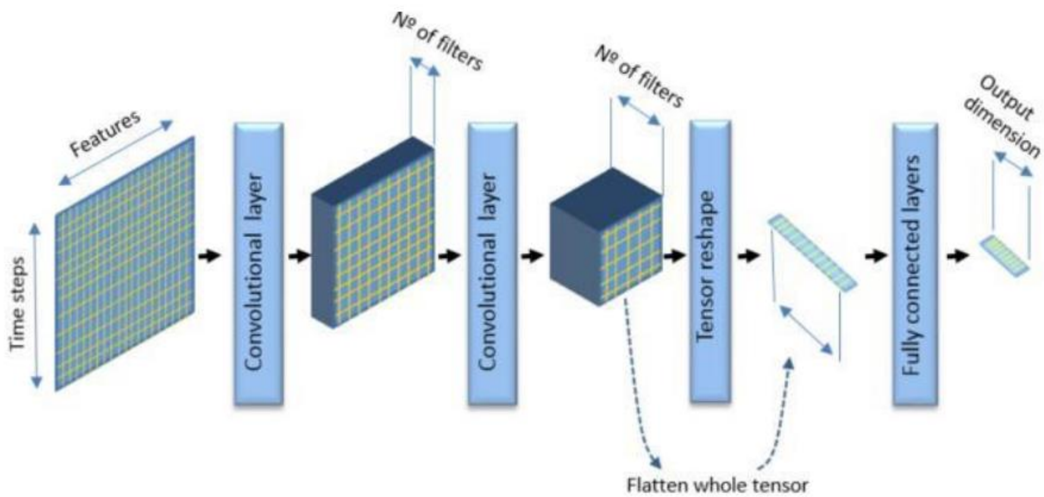


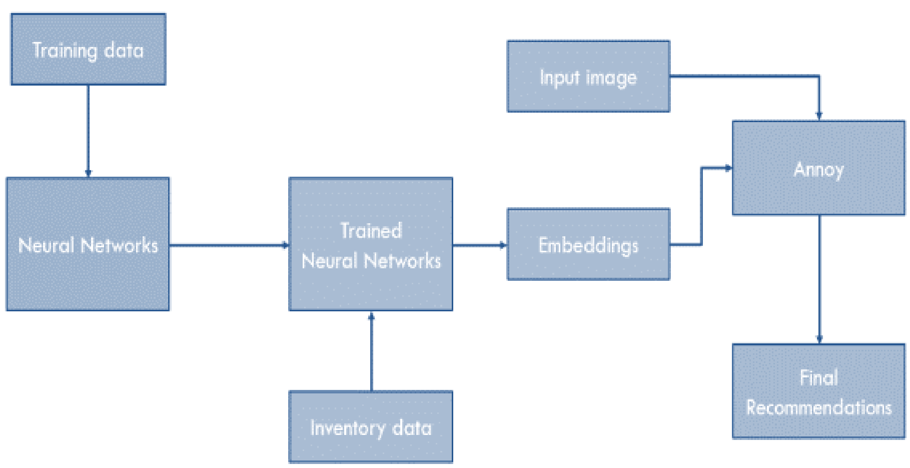
Transfer Learning Models (Densenet121, Resnet150, VGG16) are applied on the Fashion Dataset and the input/test image. After embedding the input/test image and the images in the dataset, we apply cosine similarity and recommend the similar products.



Features Extraction using pre-trained models

To use pre-trained deep learning models as feature extractors, the very first step is to remove its final output layer as we do not intend to use these models as classifiers. With this step, we are left with the output of a convolutional layer that is pooled and reduced using a global average pooling layer followed by a flatten layer to get a linear feature vector for an image.





The feature vectors are generated for test image and all training images and cosine similarity is computed and based on the top K scores; the corresponding top K images are returned as a recommendation.

- **VGG16 (CODE)**

```
#VGG16

from tensorflow.keras.applications import VGG16

vgg=VGG16(include_top=False, weights='imagenet',input_shape=(img_width, img_height, chnl))

vgg.trainable=False

model1 = keras.Sequential([vgg,GlobalMaxPooling2D()])

model1.summary()
```

- **DENSENET121 (CODE)**

```
#DenseNet121

densenet = DenseNet121(include_top=False, weights='imagenet', input_shape=(img_width, img_height, chnl))

densenet.trainable = False

model2 = keras.Sequential([densenet,GlobalMaxPooling2D()])

model2.summary()
```

- **RESNET50 (CODE)**

```
#Resnet50

from tensorflow.keras.applications.resnet50 import ResNet50

resnet = ResNet50(weights='imagenet', include_top=False, input_shape = (img_width, img_height, chnl))

resnet.trainable = False

model3 = keras.Sequential([resnet,GlobalMaxPooling2D()])

model3.summary()
```

6. CODE

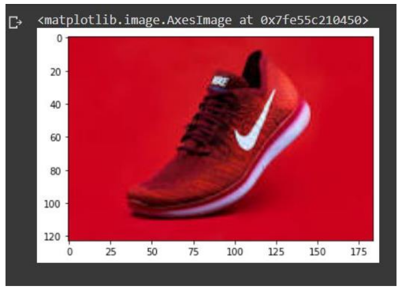
```
#COSINESIMILARITY
def get_recommendations(x, df, cosine_sim):
#idx = indices[index]
# Get the pairwsie similarity scores of all clothes with that one
    sim_scores = list(enumerate(cosine_sim[x]))
    # Sort the clothes based on the similarity scores
    sim_scores = sorted(sim_scores, key=lambda x: x[1], reverse=True)
    # Get the scores of the 5 most similar clothes
    sim_scores = sim_scores[0:5]
    print(sim_scores)
    # Get the clothes indices
    cloth_indices = [i[0] for i in sim_scores]
    # Return the top 5 most similar products
     return df['image'].iloc[cloth_indices]

#PRINTING THE OUTPUT (TOP5 SIMILAR OBJECTS AS THE INPUT FOR EACH
ALGORITHM)
chosen_img_indx = 0
recommendation = get_recommendations(chosen_img_indx, df, vgg_norm)
recommendation_list = recommendation.to_list()
#recommended images
plt.figure(figsize=(20,20))
j=0
for i in recommendation_list:
    plt.subplot(6, 10, j+1)
    cloth_img =  mpimg.imread(DATASET_PATH + 'images/'+ i)
    plt.imshow(cloth_img)
    plt.axis("off")
    j+=1
plt.title("Vgg16 recommended images",loc='left')
```

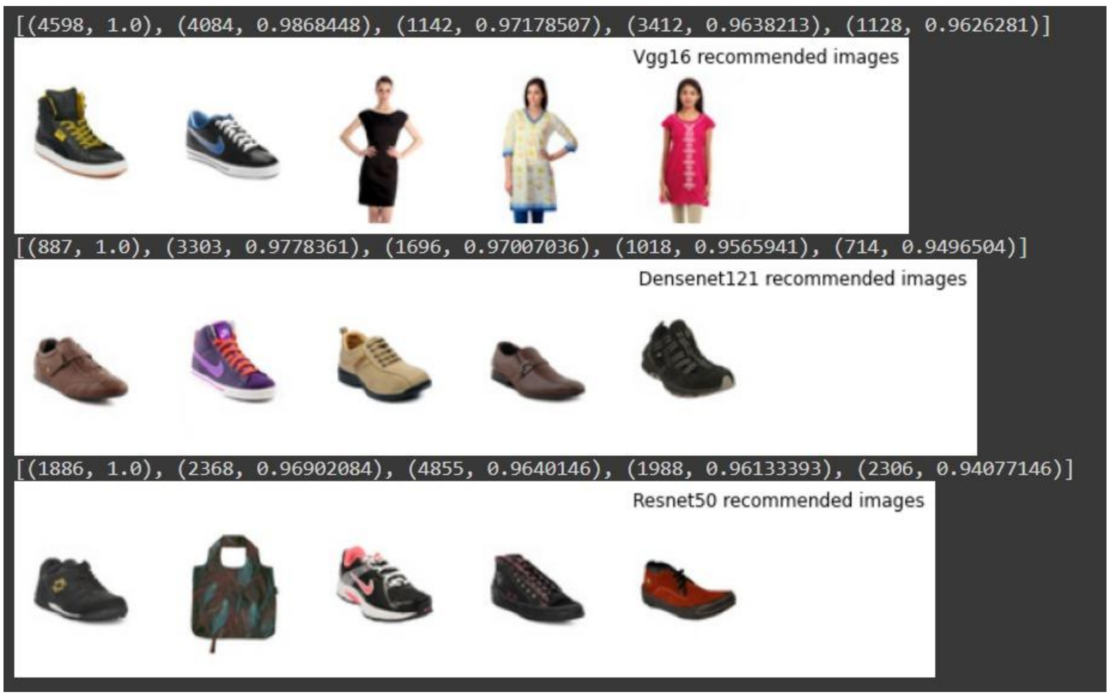
```
plt.subplots_adjust(wspace=-0.5, hspace=1)
plt.show()
recommendation = get_recommendations(chosen_img_indx, df, dn_norm)
recommendation_list = recommendation.to_list()
#recommended images
plt.figure(figsize=(20,20))
j=0
for i in recommendation_list:
    plt.subplot(6, 10, j+1)
    cloth_img =  mpimg.imread(DATASET_PATH + 'images/'+ i)
    plt.imshow(cloth_img)
    plt.axis("off")
    j+=1
plt.title(" Densenet121 recommended images",loc='left')
plt.subplots_adjust(wspace=-0.5, hspace=1)
plt.show()
recommendation = get_recommendations(chosen_img_indx, df, rn_norm)
recommendation_list = recommendation.to_list()
#recommended images
plt.figure(figsize=(20,20))
j=0
for i in recommendation_list:
    plt.subplot(6, 10, j+1)
    cloth_img =  mpimg.imread(DATASET_PATH + 'images/'+ i)
    plt.imshow(cloth_img)
    plt.axis("off")
    j+=1
plt.title("Resnet50 recommended images",loc='left')
plt.subplots_adjust(wspace=-0.5, hspace=1)
plt.show()
```

7. EXPERIMENTAL RESULTS:

INPUT TEST IMAGE:



TOP K RECOMMENDATIONS EACH MODEL GENERATED:



From the above top 5 recommendations for the test image provided, Densenet121 model produced images of more relevant and accurate recommendations.

VGG16

Accuracy

number of examples correctly predicted / total number of examples
$ACC = (TP + TN)/(TP + FP + FN + TN)$

```
results = {}
metric = "ACC"
results[metric] = (TP + TN) / (TP + TN + FP + FN)
print(f"{metric} is {results[metric]: .3f}")
```

```
ACC is  0.455
```

True Positive Rate

number of samples actually and predicted as `Positive` / total number of samples actually `Positive`
Also called **Sensitivity or Recall**.
$TPR = TP/P = TP/(TP + FN)$

```
# Sensitivity or Recall
metric = "TPR"
results[metric] = TP / (TP + FN)
print(f"{metric} is {results[metric]: .3f}")
```

```
TPR is  0.400
```

Positive Predictive Value

number of samples actually and predicted as `Positive` / total number of samples predicted as `Positive`
Also called **Precision**.
$PPV = TP/(TP + FP)$

```
# Precision
metric = "PPV"
results[metric] = TP / (TP + FP)
print(f"{metric} is {results[metric]: .3f}")
```

```
PPV is  0.400
```

ResNest 50

```
results = {}
metric = "ACC"
results[metric] = (TP + TN) / (TP + TN + FP + FN)
print(f"{metric} is {results[metric]: .3f}")
```

```
ACC is  0.833
```

```
# Sensitivity or Recall
metric = "TPR"
results[metric] = TP / (TP + FN)
print(f"{metric} is {results[metric]: .3f}")
```

```
TPR is  0.800
```

```
# Precision
metric = "PPV"
results[metric] = TP / (TP + FP)
print(f"{metric} is {results[metric]: .3f}")
```

```
PPV is  1.000
```

Densenest121

```
#Accuracy
results = {}
metric = "ACC"
results[metric] = (TP + TN) / (TP + TN + FP + FN)
print(f"{metric} is {results[metric]: .3f}")
```

```
ACC is  1.000
```

```
[ ] # Sensitivity or Recall
metric = "TPR"
results[metric] = TP / (TP + FN)
print(f"{metric} is {results[metric]: .3f}")
```

```
TPR is  1.000
```

```
# Precision
metric = "PPV"
results[metric] = TP / (TP + FP)
print(f"{metric} is {results[metric]: .3f}")
```

```
PPV is  1.000
```

From the performance metrics calculated and by inferencing the top 5 recommendations for the given input image, **Densenet121 model** gives us the most desirable result.

8. VISUALIZATION

Visualization of Latent Space of Contents

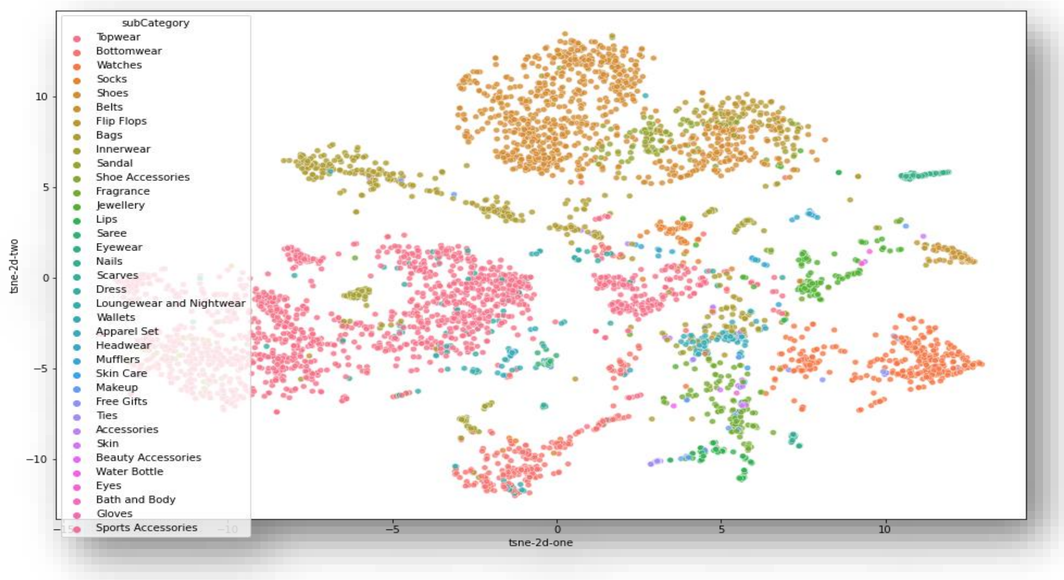
Latent Space is a lower dimensional manifold of the high dimensional images where we expect all the instances of the dataset to lie in proximity.

Since our latent space is not two-dimensional, we will use TSNE to reduce dimensionality, so we can use some visualizations to analyze the spread of the dataset. One is to look at the neighborhoods of different classes in the latent 2D plane

You can see different clusters generated by embeddings, which reinforces that features make sense. In the proceeding plots of different transfer learning model embedding, it is possible to observe the separation by Category, and in more detail by subcategory.

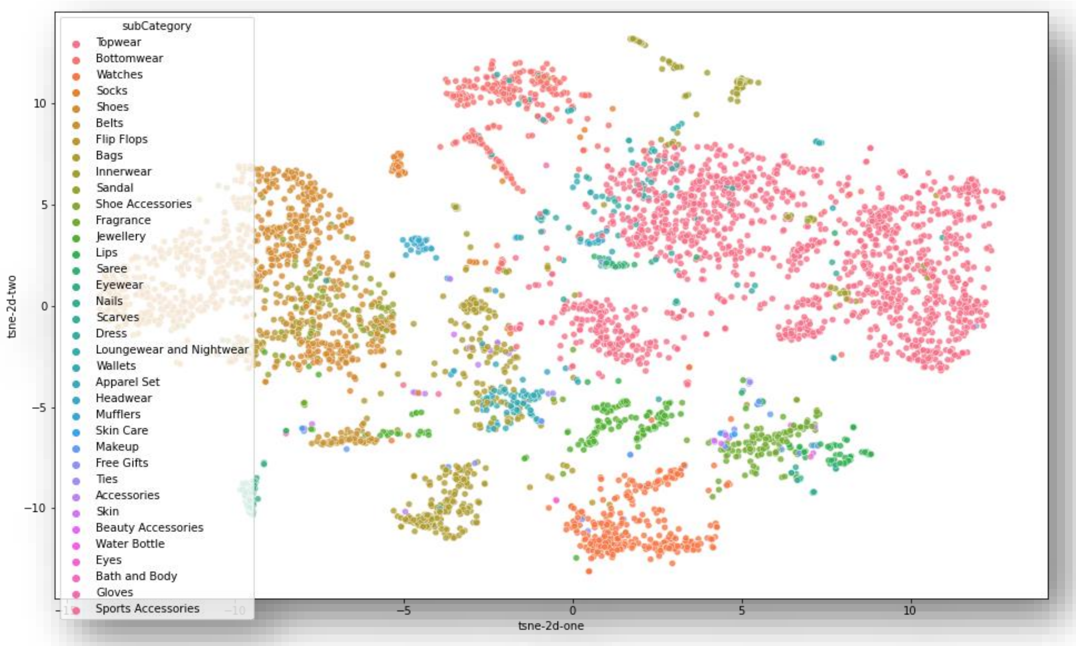
Visual Analysis of the results from Vgg16 model

Here we have done clustering-based classification chart for the Master Categories: "Apparel", "Accessories"," Footwear", "Personal Care" and "Free Items". From the visualization it is clearly depicted that clothes and footwear have the most unique value and more collection of products to recommend. Vgg16 has provided least relevant results compared to other models.



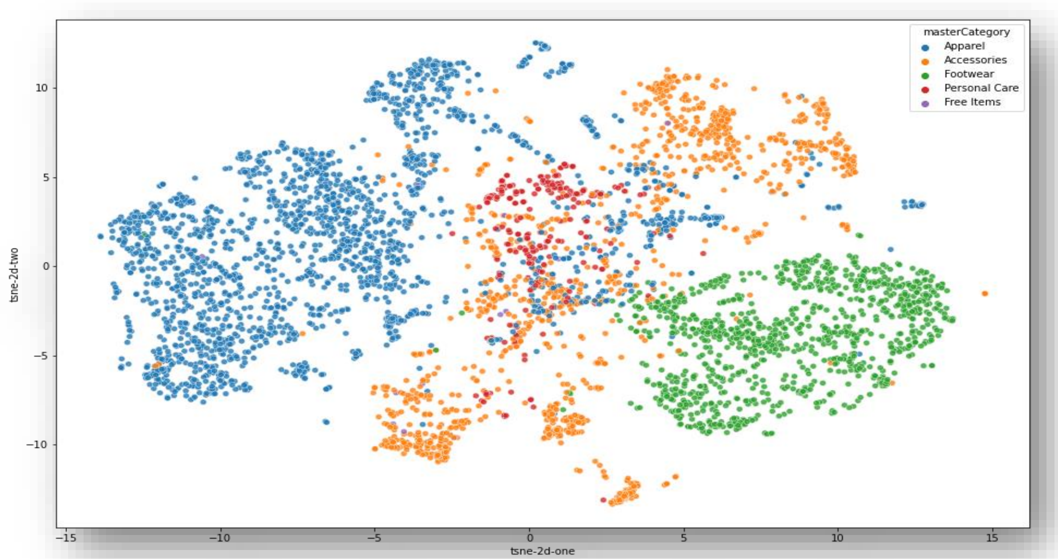
__Visual Analysis of the results from Densenet121 model__

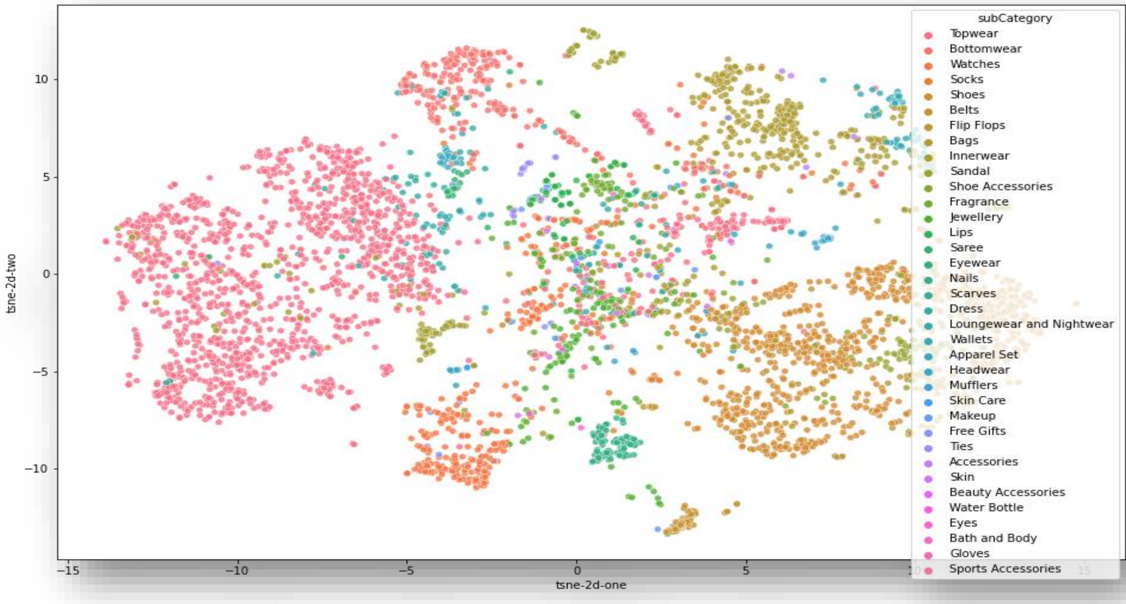
This model is so far the best pre trained model for recommending all categories of product. From the results the top 5 recommendation is noted to be spot on for every test input and the model is accurate and consistent.



Visual Analysis of the results from Resnet50 model

This model is near accurate and provides better clustering results than Vgg16 model but both Resnet and Vgg16 were not constant over every other product category. i.e., the result in Vgg16 was 40 to 50 percent accurate, densenet121 marked a 100 percent accuracy rate and Resnet50 provides 90 percent accurate results.





INFERENCES FROM VISUALIZATION:

As we can see in the visualization of the latent space that the similar object has formed clusters while different objects are farther from each other in the latent space. For example, various types of top wears, watches, shoes or bags form respective clusters. While visually dissimilar objects example: top wears vs shoes have the highest distance in latent space. One thing is to be noted that since t-SNE embedding is stochastic, the results may appear slightly different every time it is re-run.

We have visualized the latent space using t-SNE embedding. Then we have embedded the data into Latent Space and visualized the results.

9. CONCLUSION:

In our project, we tried recommending top 5 similar product recommender system that is driven by data, visually related and an effective recommendation system for generating fashion product images. The proposed approach extracts the features of the image using CNN classifier, for instance allowing the customers to upload any random fashion image from any Ecommerce website and later generating similar images to the uploaded image based on the features and texture of the input image. The similarity measured used in this project is Cosine Similarity measure. The top 5 recommendations are extracted from the database and their images are displayed. The concept of Transfer learning is used to overcome the issues of the small size Fashion dataset.

We tried multiple approaches and obtained different types of feature representations. Accuracy comparisons of different techniques were applied.

In the Pre-trained models, the direct feature representations were used and similarity was computed with all training images irrespective of the class the images belong to. The pre-trained models were trained on Image-Net where they may have learned broader and generic image feature representation

Densenet121 model performed really well compared to the other models we approached.

We plan to work on this project further to develop much better recommendation system. There can be two major extensions in this project such as

- Using transfer learning and fine tuning VGG and Resnet to adapt the model to fashion dataset specifically
- Using Generative adversarial neural networks for getting accurate recommendation

It is imperative that such research goes forward to facilitate greater recommendation accuracy and improve the overall experience of fashion exploration for direct and indirect consumers alike.

10. GITHUB REPOSITORY:

https://github.com/Lokesh-kanna-18/J-Comp-Recommender-System

REFERENCES

1. Ricci, Francesco; Rokach, Lior; Shapira, Bracha (2022). "Recommender Systems: Techniques, Applications, and Challenges". In Ricci, Francesco; Rokach, Lior; Shapira, Bracha (eds.). Recommender Systems Handbook (3 ed.). New York:

2. Aggarwal, Charu C. (2016). Recommender Systems: The Textbook. Springer.

3. Blanda, Stephanie (May 25, 2015). "Online Recommender Systems – How Does a Website Know What I Want?". American Mathematical Society. Retrieved October 31, 2016.

4. Kang, Wang-Cheng; McAuley, Julian (2018). "Self-Attentive Sequential Recommendation"

5.https://www.researchgate.net/publication/278734421_ImageBased_Recommendations_on_Styles_and_Substitutes

6.https://www.researchgate.net/publication/353485380_Fashion_Recommendation_Systems_Models_and_Methods_A_Review

7. https://www.researchgate.net/publication/360087895_Fashion_recommendation_system_using_CNN

8. https://www.ijraset.com/best-journal/fashion-recommendation-system

9.https://www.globalscientificjournal.com/researchpaper/Research_Paper_On_Recommendation_System.pdf