Fashion Product Recommendation System Using Deep Learning

Vaibhav Singh
20BCE0468
School of Computer Science and Engineering (SCOPE)
Vellore Institute of Technology

Harsh Wardhan
20BCI0051
School of Computer Science and Engineering (SCOPE)
Vellore Institute of Technology

Mehakdeep Singh
20BCI0319
School of Computer Science and Engineering (SCOPE)
Vellore Institute of Technology

***Abstract* - A artificial intelligence system is being developed that can accurately classify clothes in e-commerce images. This will make it easier for customers to find the clothes they want. This means that the computer was taught how to recognize clothes in pictures, using a data-set of fashion images. The task was to classify the images of clothing apparels, cosmetics, foot wears and other miscellaneous fashion products that were collected from online search engines. In the experiments, we evaluate the impact on classification accuracy of a data set of potential improvements. This includes data augmentation by generating diverse backgrounds and increasing the size of the network using ensembles. We have used ResNET-152 (Residual Neural Network) for the extraction of features of the image. We are looking at how accurate our predictions are, and how efficient our process is. Finally, we present the achieved accuracy rates in the clothes detection task and outline the most successful network configurations for the classification of the clothes. We were able to achieve an accuracy of 97% in our project.**

***Keywords*- augmentation, diverse, ResNET-152, classification, CNN, Artificial Intelligence**

I. INTRODUCTION

Recommendation systems are the techniques that are used to predict the rating one individual will give to an item or social entity. The items can include books, movies, restaurants and things on which individuals have different preferences. These preferences are being predicted using two approaches first content-based approach which involves characteristics of an item and second collaborative filtering approaches which considers user's past behaviour to evaluate its choices.

With the increased use of social networks, people share almost everything in their daily life online nowadays. They share the dinners they had, the movies they watched, the music they listened to, the places they visited, and the outfits they wore. With the rapid expansion of online shopping for fashion products, effective fashion recommendation has become an increasingly important problem. In this work, we study the problem of personalized outfit recommendation, automatically suggesting outfits to users that fit their personal fashion preferences. Unlike existing recommendation systems that usually recommend individual items, we suggest sets of items, which go well with each other, to the users.

This thesis proposes a fashion recommendation system which will recommend clothing images supported the style sort of the provided clothing images. In this work, we focus on the images of upper body as well as the lower body clothing and with human model in the images. We will create our own datasets through web scrapping of different e-commerce websites. We have come up with an idea to build a content-based recommendation system using ResNET-152 (Residual Neural Network) which is a convolutional neural network (CNN) architecture.

II. LITERATURE SURVEY

Fengzi Li, Shashi Kant discusses two potentially challenging problems faced by the ecommerce industry. One relates to the problem faced by sellers while uploading pictures of products on the platform for sale and the consequent manual tagging involved. It gives rise to misclassifications leading to its absence from search results. The other problem concerns with the potential bottleneck in placing orders when a customer may not know the right keywords but has a visual impression of an image. An image-based search algorithm can unleash the true potential of ecommerce by enabling customers to click a picture of an object and search for similar products without the need for typing. We explore machine learning algorithms which can help us solve both these problems. Pre-trained model VGG19 used for classification and autoencoders & cosine similarity for identification. Low resolution images and mislabelling of products can compromise the accuracy.[1]

S. Bhatnagar, D. Ghosal and M. H. Kolekar propose a state-of-the-art model for classification of fashion article images. They trained convolutional neural network based deep learning architectures to classify images in the Fashion-MNIST dataset. They have proposed three different convolutional neural network architectures and used batch normalization and residual skip connections for ease and acceleration of learning process. Increases accuracy by 2% from normal method using 2-layer CNN along Batch Normalization & skip connection. Due to small size images(28x28) algorithm misclassify Top, shirt and coat as distinguishing features only available are rgb images.[2]

A. Alamsyah, M. A. Arya Saputra, & R. A. Masrury re-train the CNN model using top-head fashion accessories dataset, represented by veil, eyeglasses and hat, to recognise the use of those item in each Instagram Image posts gathered as a dataset from 3 representative cities in Indonesia to identify the most popular accessories used in each representative city. Used CNN with object detection framework like TensorFlow to detect top-head fashion products and filter non-human images.Only limited to top-head fashion products. There is no process to give recommendation after identification of product.[3]

Analysing fashion attributes is essential in the fashion design process. Current fashion forecasting firms, such as WGSN utilizes information from all around the world (from fashion shows, visual merchandising, blogs, etc). They gather information by experience, by observation, by media scan, by interviews, and by exposed to new things. Such information analysing process is called abstracting, which recognize similarities or differences across all the garments and collections. Menglin Jia, YichenZhou,Mengyun Shi & Bharath hariharan propose additional pruning mechanism during first stage of training to improve optimization performance. There is data imbalance which make it hard to train model and detect minor categories.[4]

K. Arif Raihan describes that Attributes of pedestrians is a very important factor when it comes to the task of re-identifying pedestrians or distinguishing one pedestrian from another. Different types of clothing on a pedestrian are more recognizable than other attributes. Recognizing multiple attributes together explores the relationship among them and their contribution towards each other's representation. Dropout and batch normalization in CNN to decrease training time and resizing of images to improve accuracy that can detect up to 34 attributes/people at same time. Not accurate for jeans/trouser and improper results in identification of hair type especially short hair.[5]

Online fashion market is constantly growing, and an algorithm capable of identifying garments can help companies in the clothing sales sector to understand the profile of potential buyers and focus on sales targeting specific niches, as well as developing campaigns based on the taste of customers and improve user experience. Artificial Intelligence approaches able to understand and label humans' clothes are necessary, and can be used to improve sales, or better understanding users. Allison Steffens, Anita Maria compares the original method using SVM to new CNN model called cnn-dropout-3 with increase of approximately 10% in accuracy. Runtime can vary from hardware to hardware. Images sizes are very small than can affect identification of products.[6]

Recommender systems become widely used on the web, especially in e-commerce. In online fashion platforms, personalization becomes prevalent as user's decisions are driven not only by textual features but also the aesthetic features of products. B.Suleiman and B.Yaqub propose a novel machine learning approach that makes personalized fashion recommendations based on aesthetic and descriptive features of fashion products. Aesthetic-aware fashion recommendation using images and titles of product using BDN &Bows models. Although accuracy is improved from traditional CNN model but time for searching is increased [7]

H.Tuinhof, C.Pirker, &M.Haltmeier develop a two-stage deep learning framework that recommends fashion images based on other input images of similar style. For that purpose, a neural network classifier is used as a data-driven, visually-aware feature extractor. The latter then serves as input for similarity-based recommendations using a ranking algorithm. Composes trained-CNN classifier and modified k-NN algorithm for effective & simple classification model. Model is trained only based on texture and type of cloth.[8].

Fashion retail has a large and ever-increasing popularity and relevance, allowing customers to buy anytime finding the best offers and providing satisfactory experiences in the shops. Consequently, Customer Relationship Management solutions have been enhanced by means of several technologies to better understand the behaviour and requirements of customers, engaging and influencing them to improve their shopping experience, as well as increasing the retailers' profitability. Bellini P., Palesi L.A.I., Nesi P Used K-means (each cluster is represented by the geometric centre of the data points belonging the cluster, supposing the feature on some numerical space. Small changes involve recalculating all distances between items or customers.[9]

Clothing we wear reveals our personal style - wealth, occupation, religion, location and socio-identity. Shopper's aesthetic preferences thus influence purchasing decision in a lifestyle marketplace. Given the image of a fashion item, recommending complementary matches is a challenge. This tutorial O. Sonie, M. Chelliah, and S. Sural discusses various techniques for fashion recommendation which in turn enhance conventional data mining approaches like collaborative filtering and matrix factorization. Multilayer perceptron (MLP) model has been applied here for recommendation systems including for fashion clothing and makeup recommendations. MLP include too many parameters because it is fully connected. [10]

W. Li and B. Xu suggested with the rapid growth of fashion e-commerce, fashion recommendation has become a main digital marketing tool that is built on customer reviews and ratings. Online review is a powerful source for understanding users' shopping experiences, preferences and feedbacks on product/item performances, and thus is useful for enhancing personalized recommendations for future purchases. W. Li and B. Xu has constructed this model with two independent paths to process user/item reviews simultaneously, and each path has a convolutional neural network (CNN). CNN do not encode the position and orientation of object. Lots of training data is required.[11]

Fashion has a great impact in everyday life and therefore, people pay close attention to the way they dress. Fashion item recommendation is typically a manual, curated process, where experts recommend items and trends to large populations. However, there is increasing use of automated, personalized recommendation systems, which have valuable applications in e-commerce websites. M. A. Stefani, V. Stefanis and J. Garofalakis propose a collaborative fashion recommendation system, called CFRS. Apart from classic features, we propose a new metric, called trend score. Trend score shows how trendy a product is and is calculated taking into account the ratings provided by CFRS users (fashion experts and registered users). This paper proposes metric called trend score, which suggests the users the most trending cloth items at the moment. It fails to suggest personalized clothing items to the users.[12]

With the rapid development of online shopping, interpretable personalized fashion recommendation using image has attracted increasing attention in recent years. The current work has been able to capture the user's preferences for visible features and provide visual explanations. However, Q. Wu, P. Zhao and Z. Cui ignored the invisible features, such as the material and quality of the clothes, and failed to offer textual explanations. To this end, we propose a Visual and Textual Jointly Enhanced Interpretable (VTJEI) model for fashion recommendations based on the product image and historical review. In this model the word vector of each word has been fused with context after being processed by convolutional neural network (CNN). If the user's review has lots of noise it can lead to inaccurate extraction of user's preferences [13]

Today's online shopping is very prosperous, and fashion shopping online has also expanded rapidly. Effective fashion recommendation has become an increasingly important issue. In the past, most of the research by Y. Hsieh and Y. -M. Li focused on personal recommendation, however personality change with the environment in addition to time and space such that this trait never stagnate. Fashion reflects not only a person's character, but also his/her person's current personality influenced by the larger environment. The purpose of this study is to design a recommendation mechanism, which could infer fashion trends derived from social commerce platform and accurately recommend appropriate clothing for the users.

This module uses Latent Dirichlet Allocation (LDA). The system is based on fashion trend analysis module. Products not mentioned in requirement list and not matching with the dataset cannot be processed.[14]

Production recommendation by K. Kawattikul systems allow users to review other information that are relating to the product that they are interested in. The fundament of this problem in computer and technology perspective is to how extract information from the product that can be used for matching the related products. This work presents a technique that integrates information from production images and the description of the product (text format) to match a set of products collected in a data based. The matching will be used as the product recommendation system. This project is basically based on multilayer CNN for image processing and recommendation. Inefficient use of data, and difficulty in engineering application.[15]

The paper by Lavinia De Divitiis, Federico Becattini has presented an approach based on the combination of color/shape feature disentanglement and the usage of external memory modules to store pairing modalities between top and bottom fashion items. The most similar approach to ours is the one of De Divitiis. The authors propose to use a Memory Augmented Neural Network (MANN) as the central part of their garment recommendation system to pair compatible clothing items. The MANN is populated with a memory writing controller, trained to store a non-redundant subset of samples, which is then used to propose a ranked list of suitable bottoms to complement a given top. There is a common controller loss to train such memory modules as issues arise from uneven data distributions.[16]

The paper by Zilin Yang, Zhuo Su, Yang Yang, Ge Lin proposed a novel clothing recommendation model based on the Siamese network and Bayesian personalized ranking, which recommends clothing items that meet user's personalized preference. First, the authors construct a clothing recommendation system to recommend suitable clothing according to user's personal preference and consumption level. Second, they propose a novel fashion clothing framework from recommendation to generation to provide clothing collocation suggestions out of the dataset for users to refer in their future purchase. Third, experiments on three large real-world datasets proves the validity and feasibility of our method The model fails to recommend the user based on their price preference to an extent.[17]

Pierfrancesco Bellini, L.A.I. Palesi, Paolo Nesi, Gianni Pantaleo recommend system in the context of fashion retail has been proposed and described, relying on a multi-level clustering approach of items and users profiles in online and physical stores. The proposed solution relies on mining techniques, allowing to predict the purchase behaviour of newly acquired customers, thus solving the cold start problems which is typical of the systems at the state of the art. Problem in generating recommendations for newly acquired customers[18]

The key design elements of this system is a shared model across tasks and a custom training algorithm, using a neural network where most of the layers are shared across the tasks. Angelo Cardoso, Fabio Daolio, Saul Vargas's paper aiming to show how customer experience can be improved by enriching product/content data, this paper contributes: (1) the description of a real use case where augmenting product information enables better personalisation; (2) a system for consolidating product attributes that deals with missing labels in a multi-task setting and at scale; (3) a hybrid recommender system approach using vector composition of content-based and collaborative components. The quality of the image features can be enhanced by first detecting which area of the picture contains the relevant product.[19]

This paper by Xu Chen, Hanxiong Chen, Hongteng Xu proposes a novel neural architecture for fashion recommendation based on both image region-level features and user review information. This paper proposes a novel neural architecture for fashion recommendation based on both image region-level features and user review information. Their basic intuition is that: for a fashion image, not all the regions are equally important for the users, i.e., people usually care about a few parts of the fashion image. To model such human sense, the paper proposes to learn an attention model over many pre-segmented image regions, based on which we can understand where a user is really interested in on the image, and correspondingly, represent the image in a more accurate manner.[20]

The proposed approach uses a CNN classifier to extract features that are used as input for similarity recommendations. Hessel Tuinhof, Clemens Pirker, Markus Haltmeier develops a two-stage deep learning framework that recommends fashion images based on other input images of similar style. For that purpose, a neural network classifier is used as a data-driven, visually-aware feature extractor. The latter then serves as input for similarity-based recommendations using a ranking algorithm. This approach is tested on the publicly available Fashion dataset. Initialization strategies using transfer learning from larger product databases are presented. It is not trained on gender classification tasks.[21]

The paper by Xingchen Li, Xiang Wang, Xiangnan He has used Hierarchical Fashion Neural Network (HFGN), to solve the task of personalized outfit recommendation, which requires the recommended targets not only to have a nice compatibility but also meet user's personal taste. The paper proposes a Hierarchical Fashion Graph Neural Network (HFGN) to obtain more expressive representations for users and outfits. Benefiting from the message propagation rules, the representations can be updated by the neighbour embeddings iteratively. Different from the existing methods which only consider item-level semantics for outfits, they incorporate outfit-level semantics into the representations for outfits. Compared to separately considering compatibility matching and personalized recommendation, they regard the compatibility information as a passing message in the graph and encode this information into item and outfit representations. [22]

This system by S.F. Tsarouchis, Argyrios S. V., I.P. Bountouridis, A.Karafyllis, P.A. Mitkas consists of an interactive environment where a user utilizes different modules responsible for a) data collection from online sources, b) knowledge extraction, c) clustering, and d) trend/product recommendation. This paper is an extension of our previous work, which introduced a digital assistant for fashion designers. Science4Fashion System is a combination of a clothing recommendation and a design assistant system that takes simple and easy-to-fill high-level configuration instructions and automates the aforementioned menial and time-consuming tasks Problem in generating recommendations for newly acquired customers.[23]

A gradient boosting-based method is used to learn the nonlinear functions that map the feature vectors from the feature space to a latent space. The paper by Yang Hu, Xi Yi, Larry S. Davis proposes a functional tensor factorization method to model the interactions between user and fashion items. To effectively utilize the multi-modal features of the fashion items, they use a gradient boosting-based method to learn nonlinear functions to map the feature vectors from the feature space into some low dimensional latent space Problem in generating recommendations for newly acquired customers.[24]

This paper by Wang-Cheng Kang; Chen Fang; Zhaowen Wang; Julian McAuley seeks to combine two lines of work, by training image representations specifically for the purpose of fashion recommendation. In other words, the paper seeks to use image content (at the pixel level) to build recommender systems. This follows the recent trend of incorporating representation learning techniques into recommender systems, and methodologically is most similar to the work of where comparative judgments between images are modelled using a certain type of Siamese network. The paper adapts the popular formulation from Bayesian Personalized Ranking (BPR) to include image content via a Siamese net and show significant improvements over BPR itself, as well as extensions of BPR that make use of pre-trained representations.[25]

III. PROBLEM STATEMENT

To develop a computationally cost-effective and accurate classification model for e-commercial diverse fashion products.

Objective

To train a model in such a way that it gives an optimal result in terms of accuracy and speed The objective is also to minimize the computation Enable the sites to also filter out the choices based on the user's preferences.

Importance of Idea

Basically, the motive of this model is to help the online fashion retailers by identifying and classifying the images. By training this model, it will enable the sites to also filter out the choices based on the user's preferences. This way it can also make the site more user friendly and thus benefiting the e-commerce fashion-oriented sites. Filtering out is very essential for these sites in order to provide the user with the product of their wish and needs, hence our model will make sure that the user is able to do so and that too efficiently.

IV. SOFTWARE REQUIREMENTS

We intend on using Python as the implementation language. The main advantage of using Python in ML/DL tasks is its various libraries which facilitate image manipulation, model training, plotting various plots and graphs 8 of metrics. The libraries and software in use are:

Tensorflow: The core open-source library to help you develop and train ML models. It is developed and maintained by Google. It is easy to use and allows a number of operations which help generate powerful deep learning models

Keras: It is an API built on top of Tensorflow to ease its use and perform various operations with fewer lines of code.

Numpy: It is a library for matrix operations and scientific computations. NumPy brings the computational power of languages like C and Fortran to Python, a language much easier to learn and use. With this power comes simplicity: a solution in NumPy is often clear and elegant.

Pandas: It is a fast, powerful, flexible and easy to use open-source data analysis and manipulation tool, built on top of the Python programming language. We are going to use it for dataset manipulation.

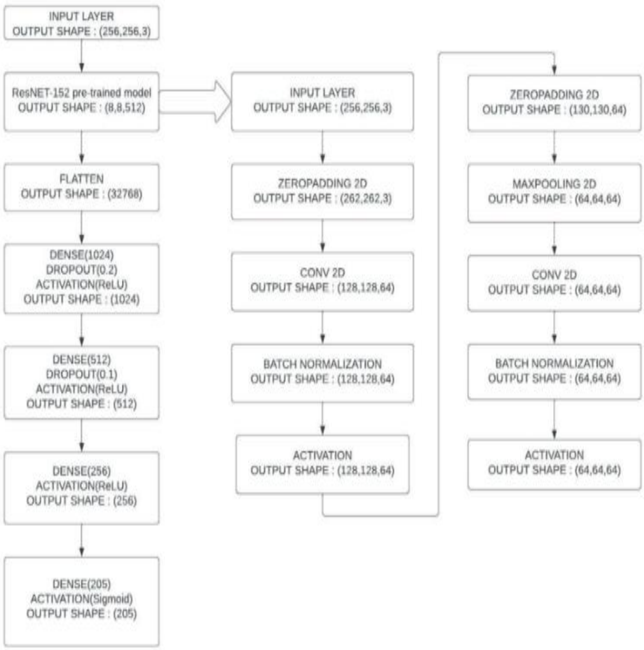
OpenCV: It is an open-source computer vision and image processing library with a vast potential. It was originally written in C++ but a Python version is also available.

Matplotlib: It is a comprehensive library for creating static, animated, and interactive visualizations in Python. It is easy to use and can be used to plot various graphs, diagrams and visualize data.

Google Colab: Colaboratory, or "Colab" for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education. More technically, Colab is a hosted Jupyter notebook service that requires no setup to use, while providing free access to computing resources including GPUs. It is free to use for everyone.

V. METHODOLOGY

Our framework is general in terms of the choice of image features. We have used ResNET-152 (Residual Neural Network) for the extraction of features of the image. Results indicate that features extracted by RNN with many layers perform significantly better than the traditional hand-crafted features. Features extracted by RNN which are pre-trained on a large image dataset are also effective on other vision tasks.



An original image is being passed through a Convolutional 2D Batch Norm layer and the features of the same have been extracted. Next step we do is to perform Pooling so that the image gets down-sampled and then repeat the same process as mentioned above. Finally, we Flatten the features and pass it through a Fully Connected Neural Network and get the end classification of the image. The end classification of the image which we got has to be checked with our predictions we made earlier, whether it was correctly predicted or not.

VI. IMPLEMENTATION

Dataset Description

The dataset consists of images of fashion products like clothes, accessories, etc and a .csv file with the labels related to each image. There are more than 9 44.4k images with each having dimensions of (256, 256, 3). This dataset is available on Kaggle with variable dimensions. The original dataset is much larger due to large dimensions of images.

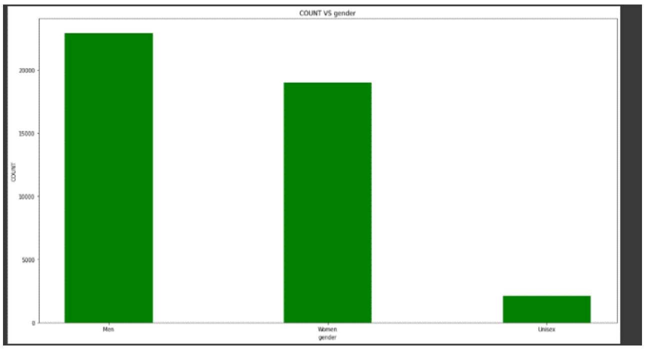
	id	gender	master	sub	article	color	season	year	usage	name
0	/content/fashion_small/resized_images/15970.jpg	Men	Apparel	Topwear	Shirts	Navy Blue	Fall	2011.0	Casual	Turtle Check Men Navy Blue Shirt
1	/content/fashion_small/resized_images/39386.jpg	Men	Apparel	Bottomwear	Jeans	Blue	Summer	2012.0	Casual	Peter England Men Party Blue Jeans
2	/content/fashion_small/resized_images/59263.jpg	Women	Accessories	Watches	Watches	Silver	Winter	2016.0	Casual	Titan Women Silver Watch
3	/content/fashion_small/resized_images/21379.jpg	Men	Apparel	Bottomwear	Track Pants	Black	Fall	2011.0	Casual	Manchester United Men Solid Black Track Pants
4	/content/fashion_small/resized_images/53759.jpg	Men	Apparel	Topwear	Tshirts	Grey	Summer	2012.0	Casual	Puma Men Grey T-shirt

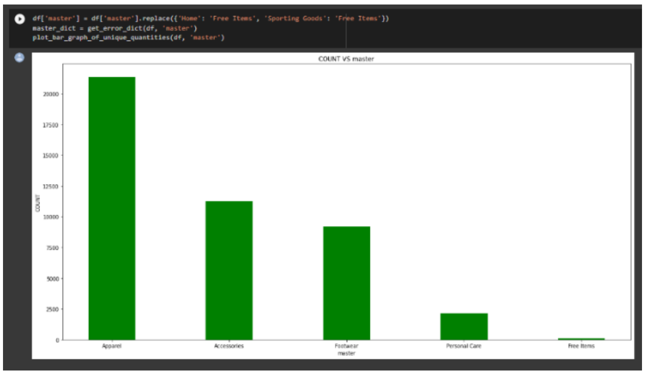
PSEUDOCODE

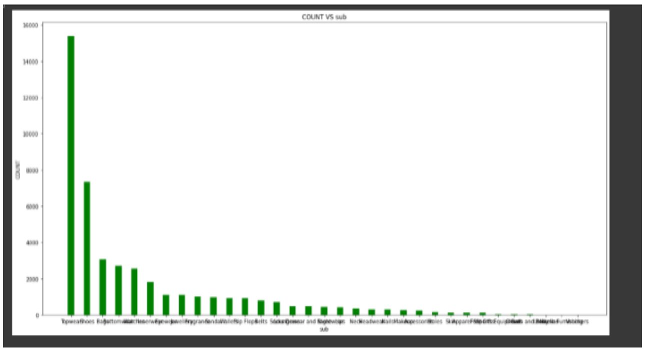
Connect Gdrive.
Connect to Kaggle.
Import files from Kaggle.
Remove unnecessary files and folders.
Import libraries. Read .csv file.
Convert ID column to full path.
Remove irrelevant columns.
Check for missing values.
Create a separate test dataset which consists of missing data.
Display images with corresponding labels from train dataset.
Replace duplicate labels and plot count of each unique label in a column.
Encode target columns.
Create Data Pipeline.
Create Model Architecture.
Create custom loss and metrics.
Compile model. Train model by passing training data through data pipeline & into the model.
Save trained model.
Plot losses and metrics.
Test model on testing data and visualize the results

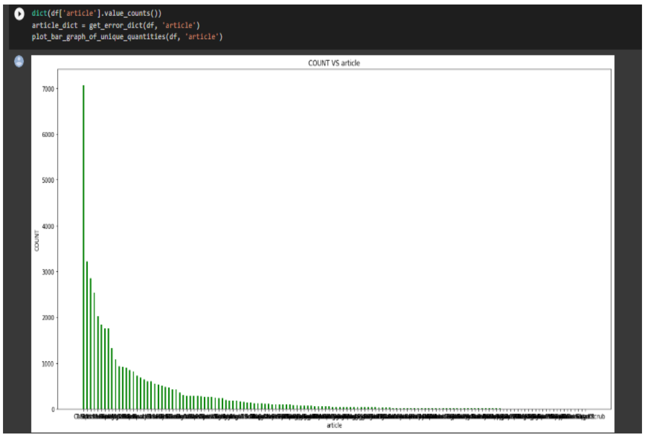
OUTPUT

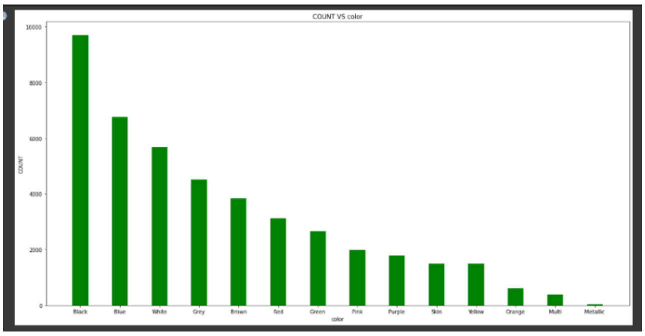
COUNT vs. LABELS

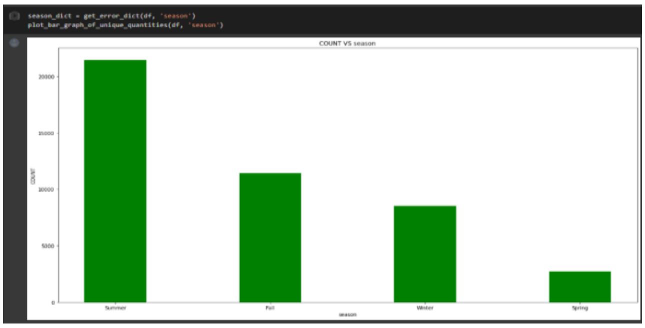












WRAPPER CLASS

```python
def loss_wrapper(y_true, y_pred):
    def gender_loss(y_true, y_pred):
        return K.categorical_crossentropy(y_true[:, 0:3], y_pred[:, 0:3])

    def master_loss(y_true, y_pred):
        return K.categorical_crossentropy(y_true[:, 3:8], y_pred[:, 3:8])

    def sub_loss(y_true, y_pred):
        return K.categorical_crossentropy(y_true[:, 8:41], y_pred[:, 8:41])

    def article_loss(y_true, y_pred):
        return K.categorical_crossentropy(y_true[:, 41:183], y_pred[:, 41:183])

    def color_loss(y_true, y_pred):
        return K.categorical_crossentropy(y_true[:, 183:197], y_pred[:, 183:197])

    def season_loss(y_true, y_pred):
        return K.categorical_crossentropy(y_true[:, 197:201], y_pred[:, 197:201])

    def usage_loss(y_true, y_pred):
        return K.categorical_crossentropy(y_true[:, 201:], y_pred[:, 201:])

    loss = gender_loss(y_true, y_pred) + master_loss(y_true, y_pred) + sub_loss(y_true, y_pred) + a
    return loss
```

```
model.compile(optimizer=ks.optimizers.Adam(learning_rate=1e-4),
              loss=loss_wrapper,
              metrics=[gender_accuracy,
                       master_accuracy,
                       sub_accuracy,
                       article_accuracy,
                       color_accuracy,
                       season_accuracy,
                       usage_accuracy])
e = 5
es = ks.callbacks.EarlyStopping(monitor='loss', mode='min', restore_best_weights=True, patience=5)
mc = ks.callbacks.ModelCheckpoint('weights.h5', monitor='loss', mode='min', save_best_only=True)
rlrop = ks.callbacks.ReduceLROnPlateau(monitor='loss', factor=0.2, patience=3, mode='min')
fit = model.fit(train_gen, epochs=e, callbacks=[es, mc, rlrop])

Epoch 1/5
1377/1377 [==============================] - 1371s 978ms/step - loss: 4.5410 - gender_accuracy: 0.8815 - master_accuracy: 0.9762 - sub_accuracy: 0.8868 - article_accuracy: 0.7099 - color_accuracy: 0.4768 -
Epoch 2/5
1377/1377 [==============================] - 1350s 980ms/step - loss: 2.2818 - gender_accuracy: 0.9362 - master_accuracy: 0.9948 - sub_accuracy: 0.9729 - article_accuracy: 0.8904 - color_accuracy: 0.7365 -
Epoch 3/5
1377/1377 [==============================] - 1355s 984ms/step - loss: 1.5456 - gender_accuracy: 0.9577 - master_accuracy: 0.9965 - sub_accuracy: 0.9855 - article_accuracy: 0.9328 - color_accuracy: 0.8076 -
Epoch 4/5
1377/1377 [==============================] - 1353s 982ms/step - loss: 1.0561 - gender_accuracy: 0.9694 - master_accuracy: 0.9970 - sub_accuracy: 0.9907 - article_accuracy: 0.9592 - color_accuracy: 0.8669 -
Epoch 5/5
1377/1377 [==============================] - 1352s 982ms/step - loss: 0.7262 - gender_accuracy: 0.9809 - master_accuracy: 0.9981 - sub_accuracy: 0.9944 - article_accuracy: 0.9722 - color_accuracy: 0.9056 -
```

Confusion matrix per sub categories.

```
gen = CustomDataGenerator(dataframe=df,
                          x_col='id',
                          y_col=y,
                          batch_size=32,
                          img_h=h,
                          img_w=w,
                          img_c=c,
                          unique_dict=unique_value_dict)
y_pred, y_true = gen.__getitem__(0)
y_pred = model.predict(y_pred)
cm = ConfusionMatrix(y_true, y_pred)
cm.gender_cm()

<tf.Tensor: shape=(3, 3), dtype=int32, numpy=
array([[20,  0,  1],
       [ 0,  1,  0],
       [ 0,  0, 10]], dtype=int32)>
```

```
cm.master_cm()

<tf.Tensor: shape=(5, 5), dtype=int32, numpy=
array([[11,  0,  0,  0,  0],
       [ 0, 14,  0,  0,  0],
       [ 0,  0,  5,  0,  0],
       [ 0,  0,  0,  0,  0],
       [ 0,  0,  0,  0,  2]], dtype=int32)>
```

```
cm.master_cm()

<tf.Tensor: shape=(5, 5), dtype=int32, numpy=
array([[11,  0,  0,  0,  0],
       [ 0, 14,  0,  0,  0],
       [ 0,  0,  5,  0,  0],
       [ 0,  0,  0,  0,  0],
       [ 0,  0,  0,  0,  2]], dtype=int32)>

cm.sub_cm()

<tf.Tensor: shape=(33, 33), dtype=int32, numpy=
array([[0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 3, ..., 0, 0, 0],
       ...,
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 4]], dtype=int32)>

cm.article_cm()

<tf.Tensor: shape=(140, 140), dtype=int32, numpy=
array([[0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       ...,
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 4]], dtype=int32)>
```

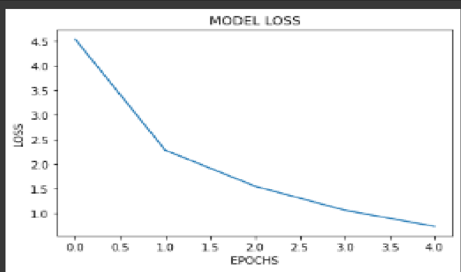
```
cm.color_cm()

<tf.Tensor: shape=(13, 13), dtype=int32, numpy=
array([[10,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0],
       [ 0,  7,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0],
       [ 0,  0,  3,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0],
       [ 0,  0,  0,  2,  0,  0,  0,  0,  0,  0,  0,  0,  0],
       [ 0,  0,  0,  0,  4,  0,  0,  0,  0,  0,  0,  0,  0],
       [ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0],
       [ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0],
       [ 0,  0,  0,  0,  0,  0,  0,  1,  0,  0,  0,  0,  0],
       [ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0],
       [ 0,  0,  0,  0,  0,  0,  0,  0,  0,  2,  0,  0,  0],
       [ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0],
       [ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  1,  0],
       [ 0,  1,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  1]], dtype=int32)>
```

```
cm.season_cm()

<tf.Tensor: shape=(4, 4), dtype=int32, numpy=
array([[ 5,  0,  3,  0],
       [ 0,  2,  0,  0],
       [ 0,  0, 15,  0],
       [ 0,  0,  1,  6]], dtype=int32)>

cm.usage_cm()

<tf.Tensor: shape=(4, 4), dtype=int32, numpy=
array([[26,  0,  0,  0],
       [ 0,  1,  0,  0],
       [ 0,  0,  3,  0],
       [ 1,  0,  0,  1]], dtype=int32)>

from google.colab import files
files.download('/content/model.hdf5')
files.download('/content/weights.h5')
```

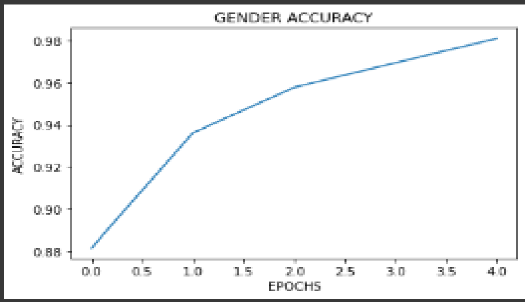
Plotting metrics and loss

PLOTTING METICS AND LOSS

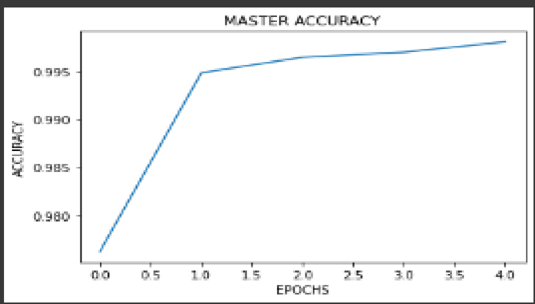
```
plt.plot(fit.history['loss'])
plt.title('MODEL LOSS')
plt.ylabel('LOSS')
plt.xlabel('EPOCHS')
plt.show()
```



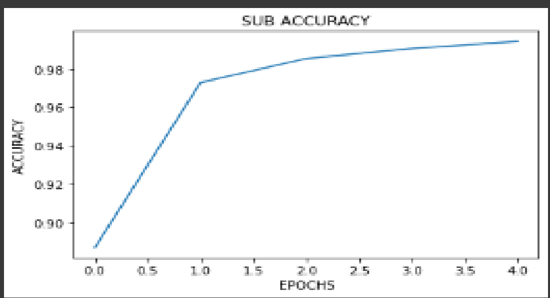

```
plt.plot(fit.history['gender_accuracy'])
plt.title('GENDER ACCURACY')
plt.ylabel('ACCURACY')
plt.xlabel('EPOCHS')
plt.show()
```



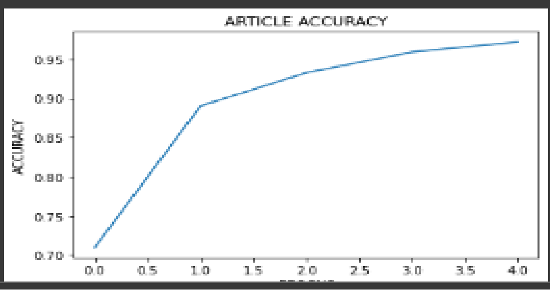

```
plt.plot(fit.history['master_accuracy'])
plt.title('MASTER ACCURACY')
plt.ylabel('ACCURACY')
plt.xlabel('EPOCHS')
plt.show()
```




```
plt.plot(fit.history['sub_accuracy'])
plt.title('SUB ACCURACY')
plt.ylabel('ACCURACY')
plt.xlabel('EPOCHS')
plt.show()
```




```
plt.plot(fit.history['article_accuracy'])
plt.title('ARTICLE ACCURACY')
plt.ylabel('ACCURACY')
plt.xlabel('EPOCHS')
plt.show()
```

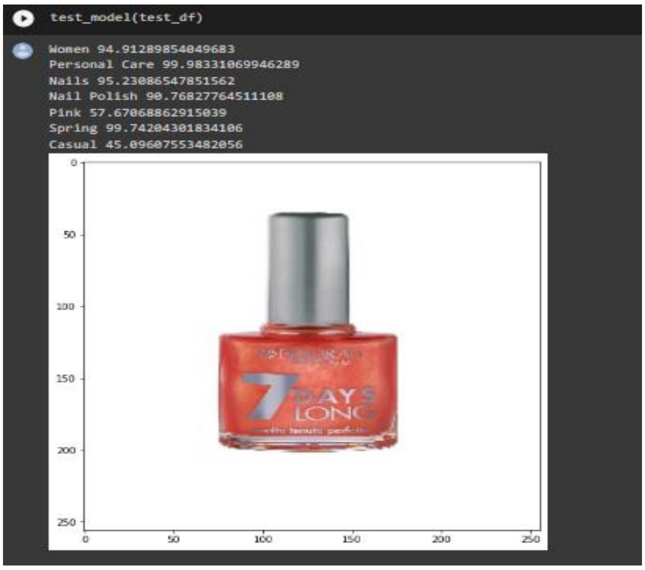


Results of testing: recommending products.

```
Women 98.51027727127075
Personal Care 97.78048992156982
Lips 95.49520015716553
Lipstick 79.23669815063477
Pink 99.94427561759949
Spring 99.84682202339172
Casual 35.64586341381073
```


```
test_model(test_df)
```

```
Men 11.988189071416855
Personal Care 82.9792320728302
Fragrance 21.750417351722717
Perfume and Body Mist 1.590869016945362
White 46.20000422000885
Spring 96.75827622413635
Casual 39.42565321922302
```



VII. RESULT ANALYSIS

Accuracy results for each category of classification provides us with an understanding of how efficient our model is as compared to the established traditional models. Our proposed model gives a loss of 0.7262 and gender accuracy of 98.09%, master accuracy of 99.81%, sub accuracy of 99.44%, article accuracy of 97.22%, color accuracy of 90.56%, season accuracy of 91.26%, usage accuracy of 97.38% after being trained for 5 epochs. These results are obtained by having the initial learning rate of 0.001.

VIII. CONCLUSION

Using ResNET-152 Architecture, we were able to achieve an accuracy of above 90% in all of the classification categories and an outstanding accuracy of 99.81% in MASTER class. We can clearly see how Batch Normalization and Pooling help improve the overall accuracy and significantly reduce the training time. These are an improvement on the various other deep learning models proposed by reputed research papers for the fashion products classification purpose.

IX. FUTURE WORK

Identifying an article type in the fashion industry is a crucial task. This model can serve as a building block for a service which shows related fashion 15 articles based on a given image. This work can also be extended to image and video indexing. It can also be very useful in improving seller experience in listing their merchandise on the platform. sellers can add pictures in their products and automated image-to-text machine learning algorithms can generate suitable tags to label them. This can lessen the inaccuracies in labelling products which sometimes have an effect on the call for adversely as the goods aren't rendered correctly inside the search results. Furthermore, we can step up the proposed project, for instance, adding more attributes depending on which our model can predict the dressing code (like is it formal or informal in terms of percentage) of the person whose picture is being taken as an input by the program. Hence, the output will be that how formally dressed the person is in terms of percentage, the higher the percentage, the more formally dressed the person is. This can help regulate the decency in the dressing style of the students in schools and colleges, and also of the employees in offices.

IX. REFERENCES

1. Fengzi Li, Shashi Kant, "Neural Networks for Fashion Image Classification and Visual Search," 2020
2. S. Bhatnagar, D. Ghosal and M. H. Kolekar, "Classification of fashion article images using convolutional neural networks," IEEE 2018

3. Alamsyah, M. A. Arya Saputra, & R. A. Masrury, “Object Detection Using Convolutional Neural Network To Identify Popular Fashion Product,” 2019
4. Menglin Jia, Yichen Zhou,Mengyun Shi & Bharath hariharan “A Deep-Learning-Based Fashion Attributes Detection Model,” 2018
5. K. Arif Raihan , “Recognition of Pedestrian Clothing Attributes from Far View Images using Convolutional Neural Network” IEEE 2019
6. Allison Steffens, Anita Maria “Classifying garments from Fashion-MNIST Dataset through CNNs” 2021
7. B.Suleiman and B.Yaqub “Aesthetic-aware recommender system for Online Fashion Products” 2021
8. H.Tuinhof, C.Pirker, & M.Haltmeier “Image Based fashion product Recommendation with Deep learning” 2018
9. Bellini, P., Palesi, L.A.I., Nesi, P. *et al.”*Multi Clustering Recommendation System for Fashion Retail*“ 2022*
10. O. Sonie, M. Chelliah, and S. Sural, “Personalised Fashion Recommendation using Deep Learning,”2019
11. W. Li and B. Xu, "Aspect-Based Fashion Recommendation With Attention Mechanism," 2020
12. M. A. Stefani, V. Stefanis and J. Garofalakis, "CFRS: A Trends-Driven Collaborative Fashion Recommendation System," 2019
13. Q. Wu, P. Zhao and Z. Cui, "Visual and Textual Jointly Enhanced Interpretable Fashion Recommendation" 2020
14. Y. Hsieh and Y. -M. Li, "Fashion Recommendation with Social Intelligence on Personality and Trends" 2019
15. K. Kawattikul, "Product Recommendation using Image and Text Processing" 2018
16. Lavinia De Divitiis, Federico Becattini “Disentangling Features for Fashion Recommendation” 2022
17. Zilin Yang, Zhuo Su, Yang Yang, Ge Lin “From Recommendation to Generation: A Novel Fashion Clothing Advising Framework” 2018
18. Pierfrancesco Bellini, L.A.I. Palesi, Paolo Nesi, Gianni Pantaleo “Multi Clustering Recommendation System for Fashion Retail” 2021
19. Angelo Cardoso, Fabio Daolio, Saul Vargas: “Product Characterisation towards Personalisation” 2018
20. Xu Chen, Hanxiong Chen, Hongteng Xu: “Personalized Fashion Recommendation with Visual Explanations” 2019
21. Hessel Tuinhof, Clemens Pirker, Markus Haltmeier: “Image-Based Fashion Product Recommendation with Deep Learning” 2019
22. Xingchen Li, Xiang Wang, Xiangnan He: “Hierarchical Fashion Graph Network for Personalized Outfit Recommendation” 2020
23. S.F. Tsarouchis, Argyrios S. V., I.P. Bountouridis, A.Karafyllis, P.A. Mitkas “Science4Fashion: An Autonomous Recommendation System for Fashion Designers” 2021
24. Yang Hu, Xi Yi , Larry S. Davis: “Collaborative Fashion Recommendation: A Functional Tensor Factorization Approach” 2016
25. Wang-Cheng Kang; Chen Fang; Zhaowen Wang; Julian McAuley: “Visually-Aware Fashion Recommendation and Design with Generative Image Models”, 2017