Recommend Matching Outfits and Accessories Using AI in Fashion

A Project Report
Presented to
The Faculty of the College of Engineering

San Jose State University
In Partial Fulfillment
of the Requirements for the Degree

Master of Science in Software Engineering
&
Master of Science in Computer Engineering

By:
Krishna Priya Gajula (krishnapriya.gajula@sjsu.edu) | December 2019
Krutika Mude (krutika.mude@sjsu.edu) | December 2019
Vaidehi Naik (vaidehi.naik@sjsu.edu) | December 2019
Varsha Katariya (varshaamrutlal.katariya@sjsu.edu) | December 2019

APPROVED

DocuSigned by:

Magdalini Erinaki

87141F0A65F14AC...

Dr. Magdalini Erinaki, Project Advisor

ABSTRACT

Recommend Matching Outfits and Accessories Using AI in Fashion

By Krisha Priya Gajula, Krutika Mude, Vaidehi Naik, Varsha Katariya

Artificial Intelligence (AI) is becoming more prevalent in many fields such as banking, e-commerce, and the fashion industry. AI has continued to influence the consumer–experience in fashion industries. For instance, Amazon’s online shopping is using a recommender system that suggests products that are similar to products selected by the customer. Bluestone’s online shopping uses visual search which suggests products to customers that are similar to images provided by customers. These suggestions are given to customers by using AI. In current shopping systems, customers must cognitively think about coordinating different pieces of their outfits. Often it is tedious to complete an outfit and sometimes customers end up making wrong choices. Consequently, online merchants have to deal with replacements and refund procedures of products returned by customers. To avoid inconvenience for both merchants and customers and to reduce transaction activities, it is important to have shopping portals that make smart recommendations using AI. In this project, an online shopping platform is designed to suggest accessories and outfits that match items selected by customers. Our system will use a deep learning model that is trained with redefined and analyzed images from trending fashion ideas. In order to make recommendations, the system will match images of items selected by customers with those that are processed based on fashion trends. Use of such a model can help in saving time and effort in customers’ decision-making process and reducing transactional burdens of merchants. Another advantage is increase in resvenue, if customer end up buying complete sets instead of one clothing item.

Acknowledgments

The authors are deeply indebted to Professor Magdalini Erinaki for her invaluable comments and assistance in the preparation of this study.

Recommend Matching Outfits and Accessories Using AI in Fashion

Krutika Mude, KrishnaPriya Gajula, Vaidehi Naik, Varsha Katariya
Computer Engineering Department
San José State University (SJSU)
San José, CA, USA
Email: krutika.mude@sjsu.edu, krishnapriya.gajula@sjsu.edu, vaidehi.naik@sjsu.edu,
varshaamrutlal.katariya@sjsu.edu

***Abstract*—Artificial Intelligence (AI) is becoming more prevalent in many fields such as banking, e-commerce, and the fashion industry. AI has continued to influence the consumer–experience in fashion industries. For instance, Amazon's online shopping is using a recommender system that suggests products that are similar to products selected by the customer. Bluestone's online shopping uses visual search which suggests products to customers that are similar to images provided by customers. These suggestions are given to customers by using AI. In current shopping systems, customers must cognitively think about coordinating different pieces of their outfits. Often it is tedious to complete an outfit and sometimes customers end up making wrong choices. Consequently, online merchants have to deal with replacements and refund procedures of products returned by customers. To avoid inconvenience for both merchants and customers and to reduce transaction activities, it is important to have shopping portals that make smart recommendations using AI. In this project, an online shopping platform is designed to suggest accessories and outfits that match items selected by customers. Our system will use a deep learning model that is trained with redefined and analyzed images from trending fashion ideas. In order to make recommendations, the system will match images of items selected by customers with those that are processed based on fashion trends. Use of such a model can help in saving time and effort in customers' decision-making process and reducing transactional burdens of merchants. Another advantage is increase in revenue, if customer end up buying complete sets instead of one clothing item.**

***Index Terms*—Apparel Matching, Keras, Image Preprocessing, Color Matching, Region of Interest Extraction, Image Features Extraction**

I. Introduction

With the surge of online shopping, many people prefer purchasing clothes online rather than visiting the store personally. These sites show a huge catalog of a variety of outfits. Only the clothes similar to the one user is currently viewing are recommended in those online shopping portals. To get matching apparel on the selected clothing, customers must cognitively think about coordinating different pieces of their outfits. Searching a matching outfit for a selected clothing item in such a huge catalog can get very tedious for the user. Many people find choosing and purchasing new clothing items very difficult because they require knowledge about the combination of fashion items. Sometimes customers may end up making the wrong matching choices. Consequently, they return the items and online merchants have to deal with replacements and refund procedures of products returned by the customers. Apparel has a high return rate in the online retail industry. According to Mulpuru et al.,[1] 22% of apparel sales get returned on average. One of the main reasons for these returns is that the item ordered is not a good match for the other item they wish to wear. To reduce returns due to this reason, a system is required that can suggest the outfits that go really well with the selected item. In this project, an online shopping platform is designed to suggest accessories and outfits that match items selected by customers. Our system is using a deep learning model that is trained with redefined and analyzed images from trending fashion ideas. In order to make recommendations, the system will match images of items selected by customers with those that are processed based on fashion trends and analysis of customers' past purchases. Use of such a model can suggest fashion item combination as per the current trend. It also helps in saving time and effort in customers' decision-making process and reducing transactional burdens of merchants. Another advantage is increase in revenue, if customer end up buying complete sets instead of one clothing item. The motivation for our project is from the paper by Abeer Hamdy, Noha Kareem, and Khaled Nagaty [2]. This paper is an android based mobile platform which provides the customer a mean for autonomous clothes matching with color identification.

II. Project Architecture

The architecture of this deep learning-based application can be divided into two parts. One part is the deep learning model to categorize the apparels. This deep learning model is combined with the color matching module to choose the opposite matching color apparel and accessories. This information is then used by the rule-based module for appropriate apparel recommendation. The other part is the online portal, where the user will select an apparel from online shopping portal. This portal also displays all the apparel and accessories recommendations for the apparel selected by the user. The images in the shopping portal act as test data for the deep learning model.

The image which is selected by the user is pre-processed and then the type, pattern and color of the apparel is predicted using Region of Interest 1 (ROI1), Region of Interest 2 (ROI2)

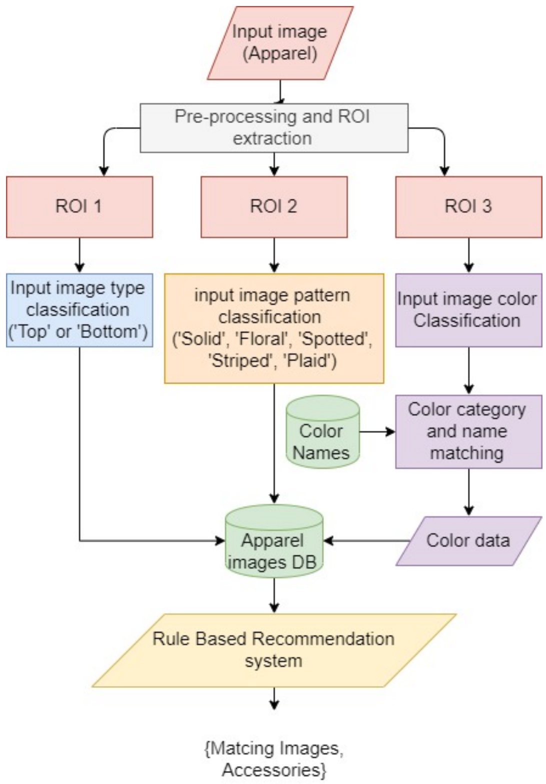


Fig. 1. Project Architecture.

and Region of Interest 3 (ROI3) classifiers. ROI1 classifier determines the apparel outline such as 'Top' or 'Bottom'. The ROI2 classifier predicts the most prominent color from the apparel in the image. And finally, the ROI3 classifier detects the pattern of the apparel such as 'Solid', 'Floral', 'Plaid', 'Striped' and 'Spotted'. A color wheel (Fig. 6) is used to get analogous and complementary colors to dominant color in the apparel. The three outputs of ROI1, ROI2 and ROI3 classifiers are used as inputs to the rule-based recommendation module. This rule-based module determines matching apparels and accessories for the apparel in input image selected by the user on the shopping portal.

III. System Design

In this project, an online shopping platform is designed to suggest accessories and outfits that match items, selected by customers. Our system is using a deep learning model that is trained with refined and analyzed images from trending fashion ideas. In order to make recommendations, the system will match images of items selected by customers with those that are processed based on fashion trends. Use of such a model can help in saving time and effort in customers' decision-making process and reducing transactional burdens of merchants.

In any Machine Learning based application, quality of data plays a key role in determining the accuracy of model. For choosing the dataset for this project, we studied Polyvore [3], Deep-fashion and StreetStyle datasets[4]. The images in Polyvore and Deep-Fashion datasets follow constraints like fixed background, clear lighting and correct angle of the object of interest. The images in StreetStyle dataset contains both, the images with constraints and real-life images without any of the above constraints. Training a model with both high and low quality images improves testing results. For this reason we used Streetstyle cornell dataset. The dataset contains 27,000 images in which 70 percent of the data is used for training and 30 percent is used for testing. The dataset contains apparels of different types with different patterns and colors. It also contains accessories.

In this project, the methodology used in [2] is used as the baseline approach. It proposes two Region of Interest (ROI) methods to extract required information from the images. ROI1 is used to determine whether the apparel in the image is a top or bottom and ROI2 is used to extract the dominant color in the apparel. ROI1 and ROI2 has been developed as per the paper. Additionally, ROI3 has been implemented that is used to detect whether the apparel is plain or has a pattern. If any pattern is present on the apparel, then it detects the type of the pattern on it like floral, plaid, spots, stripes. The outputs of ROI1, ROI2, and ROI3 are fed to the rule based system which provides recommendations for the given input.

The steps followed in implementation of this project are as follows:

Step 1: Importing Data set and File organization

The images in the Streetstyle dataset are organized into different folders based on the labels on the images. File organization for training the deep learning model is one of the important step to categorize the test data accurately. The organization of the folders is as follows. The main folder holds four folders, these four folders are created depending on color, pattern, cloth types and accessories type from the images. Each of these folders have sub-folders structure as shown in Fig.2. This kind of organization of images is done for both test and train data.

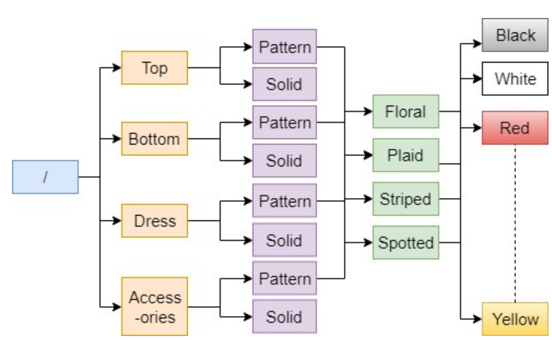


Fig. 2. Folder Organization

Step 2: Preprocessing of images

The data is prepared for the machine learning model by resizing all the images to be of the same size. To detect the region of the interest in the image, bounding boxes are created. Bounding boxes are of rectangular shape and are drawn around the apparels inside the image.

Step 3: ROI1 (Region of Interest 1)
ROI1 step focuses on determining whether the given apparel is top or a bottom. A classifier is built to classify type of the input image as 'Top' or 'Bottom'. This classifier is a Convolution Neural Network (CNN) using Keras open source neural network library.

Step 4: ROI2 (Region of Interest 2)
In ROI2 step, the dominant color in the apparel is extracted to classify the image by color. The CNN classifier used in this step classifies the input image using the extracted prominent color in classes such as 'Blue', 'Pink', 'More than 1 color', etc. For the construction of the classifier 'Keras' an open-source neural network library is used.

Step 5: ROI3 (Region of Interest 3)
The pattern detection and classification for the apparels is being made in ROI3. A classifier is built to classify pattern of the input image such as 'Solid', 'Spotted', 'Striped', 'Plaid' and 'Floral' etc. This classifier is a bulit using Keras open-source neural network library.

Step 6: Rule Based System
Output of ROI1, ROI2 and ROI3 are given to rule based recommendation system. The rule based system is built to find matching clothes. The input for the rule based system is basically the output of deep-learning modules, that is the predicted type, color and pattern of the apparel. For example, if the predicted output from the deep learning model is 'Shirt','Red','Solid'. The recommendation algorithm looks for the matching clothes images in opposite type of database that is 'Bottom'. And, then the recommendation algorithm uses color wheel to get its matching colors for Red using complementary and analogous colors. After finding the complementary and analogous colors, the recommendation system looks for the apparels in that particular color. And finally,it starts looking in a folder which is opposite to solid like floral, spotted or stripped. The architecture of rule based system is shown in Fig. 3.

Step 7: Full Stack Application To demonstrate a project, A full-stack application using React JS as front end, Flask as backend. The ROI1, ROI2 and ROI3 classifiers are pickled into h5 files and extracted in flask back-end.

1. Convolution Neural Network:
CNN is a type of Neural Network that is based on the structure of visual cortex in an animal and has great application in computer vision domain. A CNN consists of one input layer, one output layer and multiple hidden layers. The hidden layers of a CNN typically consist of a series of convolution layers in which convolution operations are performed by moving a small size filter across the image. The convolution operations include multiplication and sum of feature values in the image, which after reaching certain count of epochs starts detecting features in the image [5]. As we move further with this process, features that are more complex are detected. An image contains a lot of pixels. To learn the features from the images,

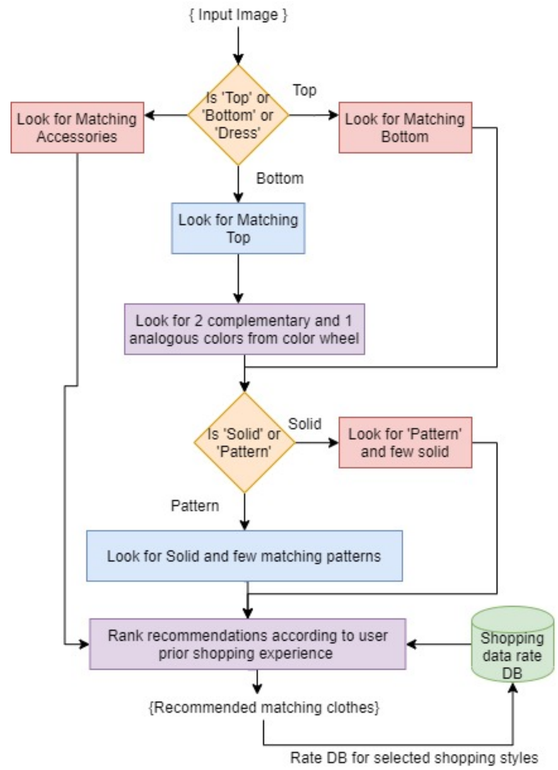


Fig. 3. Rule Based System Architecture

pooling layer reduces the number of parameters/dimensions of the data by combining the outputs of neuron clusters at one layer into a single neuron in the next layer [6]. This in turn reduces the number of computations. Due to this feature of CNN, the problem of overfitting is also avoided. Convolutional networks may include local or global pooling layers that may perform a max or an average pooling. Max pooling uses the maximum value and Average pooling uses average value from each of a cluster of neurons at the prior layer. Another important component of a neural network is an activation function. An activation function is specified to decide whether a neuron in the neural network should be activated or not by calculating weighted sum and adding bias with it. In this project, a CNN model is built with ReLU activation function and Max-pooling for pooling operation.

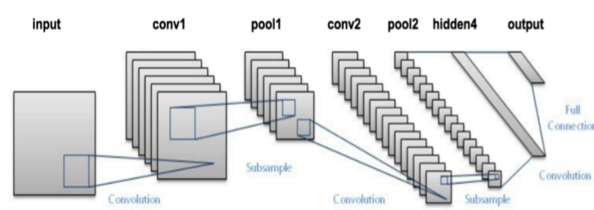


Fig. 4. Typical CNN Architecture [7]

2. Color Wheel
On studying fashion industry we observed that certain combination of colors go well together and there is always an mathematical pattern in which colors match with each other. This can be represented by color wheel. In this project we

are using analogous and complementary color of given color to select and recommend matching clothes along with other factors. A colour wheel is used to represent 12 most distinct colours Red, Orange, Yellow, Chartreuse Green, Green, Spring Green, Cyan, Azure, Blue, Violet, Magenta, Rose (Fig. 5).

Fig. 5. Color Wheel

The most dominant color in the apparel is obtained from the image and located on the color wheel using its color code. The above mentioned colors present within 45 degrees on both side of the color point(CP1) are considered for analogous matching. For complementary color matching, the color point (CP2) at 180 degrees from CP1 is taken and the colors from above list present in 45 degrees vicinity on both sides of CP2 are considered. To check which colors are in the vicinity, the distance is calculated between CP1 and all the colors in the above list. Every color is composed of RGB values and can be represented in RGB code e.g. Red is (255,0,0), Orange is (255,165,0). The distance between two colors is calculated by using formula mentioned in Fig. 7.

$$colour_1 = (R_1, G_1, B_1)$$
$$colour_2 = (R_2, G_2, B_2)$$
$$difference = \sqrt{(R_2 - R_1)^2 + (G_2 - G_1)^2 + (B_2 - B_1)^2}$$

Fig. 6. Color difference formula

3. Keras

Keras is written in Python and is an API for high-level neural networks. It is proficient enough to run on the top of tensor flow. The prefered library for deep learning will be Keras as it has easy framework and it is user friendly. It also supports the collaboration of recurrent and convolutional networks.

4. TensorFlow

Tensor flow is used in machine learning to build and deploy machine learning applications. It is an open source platform which provides a variety of libraries and resources to easily develop machine learning algorithms. Tensor flow provides range of APIs compatible with Python and C++ which helps in developing and deploying deep learning applications.

IV. Evaluation Metrics and Results

For any machine learning project, evaluation of the algorithm is one of the most essential parts. The model may give satisfactory results for a few evaluation metrics but may perform poorly for other metrics. In this project, different methods are used to process the images containing apparel and obtain features of the apparel, classify the apparel as top or bottom, identify patterns and find colors. Later, these outcomes are used to recommend matching outfits.

Evaluation metrics for classifiers:

During our study, it's been noticed that the classifier performance was better on StreetStyle dataset compare to Polyvore and Deep-fashion data-sets. The Streetstyle dataset contains images with both high quality and low quality images. With the use of such kind dataset the performance of the classifier improved compared to other data-sets. The Streetstyle data-set contains 27,000 images, in which we used 70 percent of data for training and 30 percent for testing.The data set contains labeled images of different types of clothes with colors and patterns.

The distribution of data based on the type of the apparel in shown in Fig.7

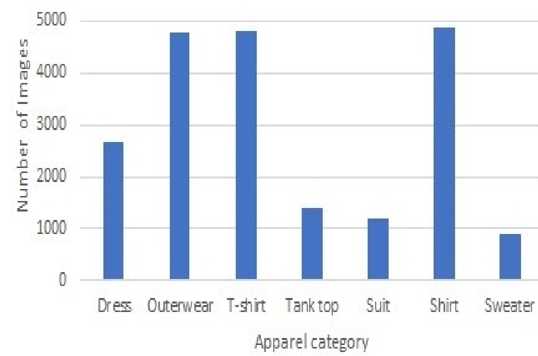


Fig. 7. Frequency distribution of data based on type of the clothes

In Fig.8 the distribution of data is shown based on the color on the apparel.

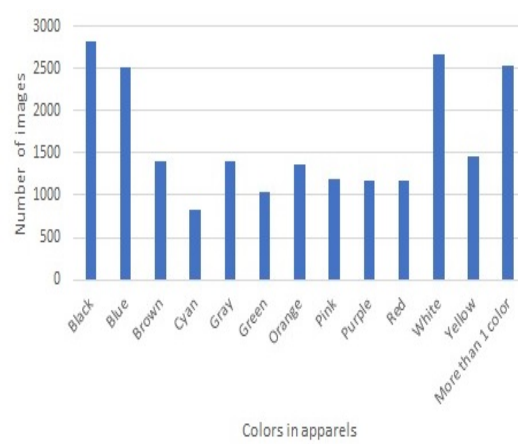


Fig. 8. Frequency distribution of based on color on the apparel

The distribution of data based on pattern on the clothes in shown in Fig.9

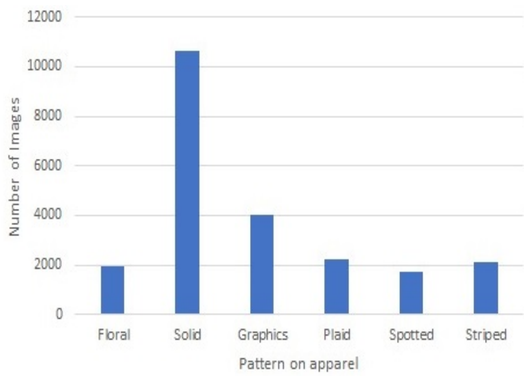


Fig. 9. Frequency distribution of data based on pattern on the apparel

Accuracy is used as evaluation metrics for all the 3 classifiers of ROI1, ROI2, ROI3.

1. ROI1 :

No. of Epochs	No. of Sub Epochs	Accuracy
50	10	0.7134
100	50	0.7497
300	100	0.7867
500	250	0.8681
1000	500	0.9344

TABLE I
CLASSIFIER PERFORMANCE METRICES - ROI1

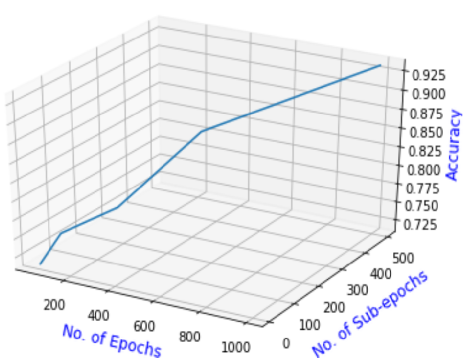


Fig. 10. Classifier Performance Graph - ROI1

2. ROI2 :

No. of Epochs	No. of Sub Epochs	Accuracy
50	10	0.5798
100	50	0.5936
300	100	0.6049
500	250	0.6412
1000	500	0.7243

TABLE II
CLASSIFIER PERFORMANCE METRICES - ROI2

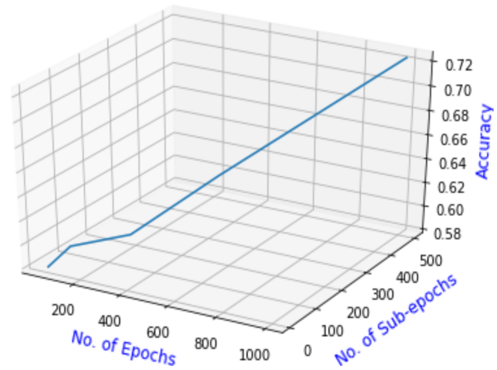


Fig. 11. Classifier Performance Graph - ROI2

3. ROI3 :

No. of Epochs	No. of Sub Epochs	Accuracy
50	10	0.7912
100	50	0.8489
300	100	0.8867
500	250	0.9188
1000	500	0.9256

TABLE III
CLASSIFIER PERFORMANCE METRICES - ROI3

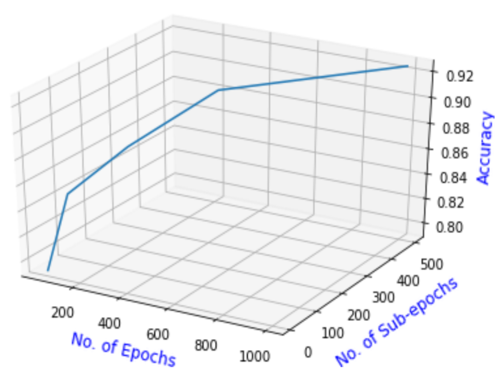


Fig. 12. Classifier Performance Graph - ROI3

The formula used for performance metrics:

Metric	Formula (Percentage)
Accuracy	$\frac{100 * \, correctly\ classified\ items}{total\ test\ items}$

The true positives refer to apparels correctly classified as positives; total positives refer to all apparels originally categorized as positives. The same naming convention is applied to true negatives and total negatives with respect to the negative category.

Fig. 13. Sample input image

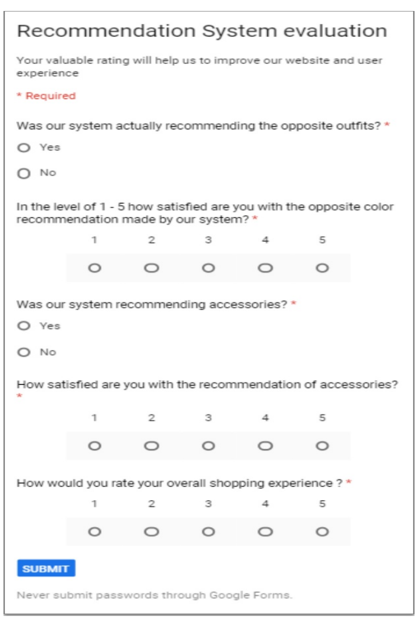
The sample image shown in Fig. 13. is send as input to our deep-learning model.

In Fig.14 an output prediction of the classifier for the sample input image Fig.13. is shown. The prediction scores of each of the classes is given by the deep learning model as shown in the in Fig.11. The highest confidence among all the classes is considered. By considering scores which are above the threshold the final output is obtained as "Solid Brown Pant". As this project is regarding Fashion and shopping

```
for image: 0: 40.1298755073334 , 155.04560834169388, 1

confidence scores for type of clothes
[('n045576468','jeans','0.89479302'),('n04447861','skirt','0.43458328'),
 ('n03062245','pants','0.928747234')]

confidence scores for color
[('n023456745','red','0.73632998'),('n0390567','orange','0.515864367'),
 ('n043456745','brown','0.81496437')]

confidence scores for pattern
[('n03766468','solid','0.91374201')]

pants ,brown, solid
```

Fig. 14. Confidence for a prediction of sample input (Fig. 13) image

website experience, the clothing choice and preferences varies with individuals. Hence, we recorded 100 responses from 100 individuals about their experience with the shopping website and recommendation system.We made a google form for recording the responses. Fig.15 is the sample of our google form which we used for recording the responses.



Recommendation System evaluation

Your valuable rating will help us to improve our website and user experience

* Required

Was our system actually recommending the opposite outfits? *

Yes

No

In the level of 1 - 5 how satisfied are you with the opposite color recommendation made by our system? *

1 2 3 4 5

Was our system recommending accessories? *

Yes

No

How satisfied are you with the recommendation of accessories? *

1 2 3 4 5

How would you rate your overall shopping experience ? *

1 2 3 4 5

SUBMIT

Never submit passwords through Google Forms.

Fig. 15. Google Form

Below are the response metrics:

1. Overall rating :

100 individuals were asked to rate their overall experience of using the recommendation system. Fig 16. represents the bar graph for 100 rating where on x- axis we have ratings and on y-axis we have percentage of individuals.

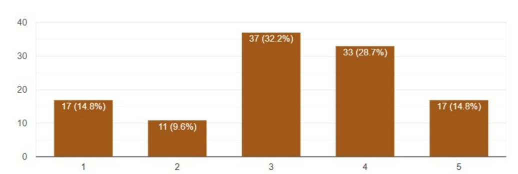


Fig. 16. Percentage of overall rating for 100 individuals

2. Percentage of the working state of recommendation module for accessories:

In the survey the 100 individuals were asked if the recommendation module for accessories is working or not. The percentage of yes and no responses are showed in a pie chart in Fig.17.

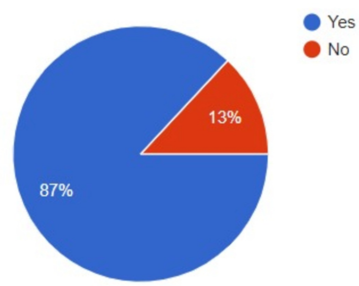


Fig. 17. Recommendation Evaluation for Accessories

3. Ratings for matching accessories recommendation module:

The rating for the accessories recommendation module is recorded from the 100 individuals. These ratings are represented in a bar graph in Fig.18. The x-axis represents the range of rantings and the y-axis represents the percentage of individuals.

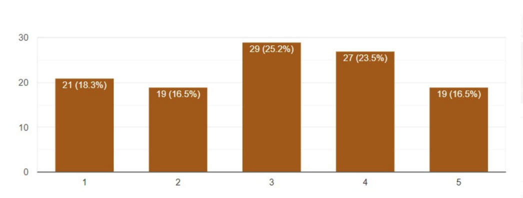


Fig. 18. Percentage of accessories recommendation rating for 100 individuals

4. Percentage of the working state of the matching outfit module :

The percentage of yes and no responses for the working state of matching outfit module is showed in a pie chart in Fig.19.

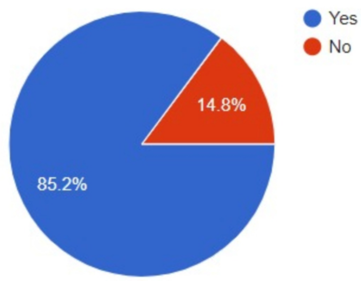


Fig. 19. Opposite color recommendation Evaluation

5. Ratings for the opposite color recommendation module :

The rating for the matching clothes recommendation module is recorded from the 100 individuals. These ratings are represented in a bar graph in Fig.20. The x-axis represents the range of rantings and the y-axis represents the percentage of individuals.

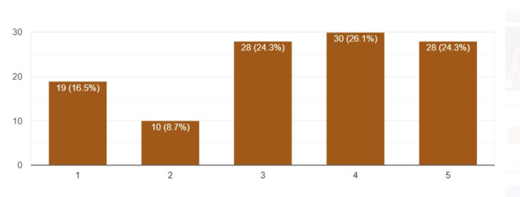


Fig. 20. Percentage of opposite color recommendation rating for 100 individuals

Results: Below image shows the final recommendations made by the recommendation system.

Fig. 21. Final Recommendations for the given input

V. Related Works

The paper by Abeer Hamdy, Noha Kareem, and Khaled Nagaty [2], proposes a mobile application namely “Android Clothing Stylist (ACS)” for clothing coordination, which helps people to seek fashion advices. The ACS matches an input apparel image captured by the mobile camera against an apparel image database which is served on the device in order to generate suggestions. A total feature of 192 images has been used (84 tops and 58 bottoms). These images have the properties like minimal details in the background, a single image in the foreground, convenient lighting to support optimal color reflection and the apparel is not folded. They used a database to store images of around 10 color categories with names, representing over 581 shades of colors in total. The classification methods used are two level SVM, and K-means. The test results were based on a sample set of 10 colors, half of which did not exist in the color database. The most dominant color in the apparel’s material was considered in the outfit matching result. For every matching color the system searches for matches through the querying the closet for apparels with the needed matching color category outline and thus the matching outfit is obtained.

Pongsate Tangseng, Kota Yamaguchi and Takayuki Okatani [8] demonstrated a system that provides a clothing recommendation and grading based on closet of items of the users which have different sizes and variety of clothes in the closet. For this, they have developed a deep neural network system that takes these varying inputs and predicts a score. The amount of data collected for their research was gathered from Polyvore dataset which consists of 644,192 unique items. Their model achieved an accuracy and precision of 84% for the grading the quality of the item/outfit.

Takuma Nakamura and Ryosuke Goto [9] proposed a model which keeps style as the basis for selecting each fashion item. Unlike old methodologies of outfit generation which relies on global information of an outfit, in this research paper they proposed style as a key feature of the outfit. The model was successful in generating outfits more efficiently and easily without any additional information. The model in this research was generated from two human generated outfit datasets.

Another effective method discussed in the domain for fashion items matching is predicting the visual compatibility between the items. The paper [3] talks about using visual compatibility in fashion recommendation. A joint modal for learning visual-semantic embedding and the compatibility relationships among apparels and accessories is proposed in it. Fashion items are taken as a sequence and an item is considered in a step. For each item the prediction of all compatible items with it are made. It sequentially predicts compatible items for next item conditioned on the previous one. This is achieved by training a Bidirectional LSTM model. As a regularization for training LSTM modal, the attribute and category information is injected by learning the visual-semantic space by regressing image features to their semantic representation [3]. This modal not only gives the recommendations for the outfits in the sequence but also determines the compatibility of a given outfit.

In [10], Context-Aware Visual Compatibility model is proposed for apparel and accessories matching and recommendation. To predict the compatibility between two or more clothes and accessories, the visual features, also referred as context, are considered. A certain set of apparels and accessories, when worn together, are visually pleasing, they are added to the set of compatible items. This set is extended as more compatible items are added. It is represented in the form of graph. Many other combinations of matching items can be deduced from this graph. To achieve this, a Graph neural networks is used that gives product recommendation and compatibility prediction. The approach of predicting visual-compatibility, when embedded with our current approach would give not only metric based learning results based on pairwise comparison, but also context aware based learning results. This helps in ensuring that the recommendations are aesthetically compatible as well as visually appealing. This method can be incorporated in future work of this project.

VI. Conclusions

The advancement of AI in online shopping is becoming sophisticated. The customers are busy in their day to day life and want someone to coordinate their outlook according to the latest trend in the fashion world. An online shopping portal that would coordinate clothes according to the customers previous selections and to the current trend of fashion world would be great help to the customers who seek for fashion advice. The online shopping portal provides an exclusive purpose that comprises of 3 ROI approaches. All the three ROI models uses Convolution neural network which is implemented using keras.The ROI1 approach classifies if the input into a top or bottom.In ROI2 approach the most dominant color is determined.The ROI3 approach is used to determine the pattern for the apparel.After predicting the type,color and pattern, using the color wheel the color complementary and analogous to the most dominant color is determined. After obtaining the opposite color using rule based system the quires are made in that color database with opposite type of the apparel.Finally by querying we obtained our recommendations.

An extension to this can be a mobile shopping application, where the user can scan his apparel and the application would suggest the matching outfit based on the current trend in that region. These suggestions can be bought by the user directly through the application. Another extension can be merging the method used in [3] and [10] with current approach to obtain aesthetically compatible and visually pleasing recommendations.

References

[1] S. Mulpuru, "Sucharita mulpuru, 'think-tank: The opportunity in online return'." [Online]. Available: https://wwd.com/business-news/business-features/sucharita-mulpuru-think-tank-returns-10939339/

[2] A. Hamdy, N. Kareem, and K. Nagaty, "Apparel coordination based on contour and dominant colours matching," in *2016 SAI Computing Conference (SAI)*, July 2016, pp. 394–402.

[3] X. Han, Z. Wu, Y. Jiang, and L. S. Davis, "Learning fashion compatibility with bidirectional lstms," *CoRR*, vol. abs/1707.05691, 2017. [Online]. Available: http://arxiv.org/abs/1707.05691

[4] K. Matzen, K. Bala, and N. Snavely, "StreetStyle: Exploring world-wide clothing styles from millions of photos," *arXiv preprint arXiv:1706.01869*, 2017.

[5] Wikipedia contributors, "Convolutional neural network — Wikipedia, the free encyclopedia," 2019, [Online; accessed 19-November-2019]. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Convolutional_neural_network&oldid=926243824

[6] ujjwalkarn, "An intuitive explanation of convolutional neural networks," 2016. [Online]. Available: "https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/"

[7] D. Gupta, "Architecture of convolutional neural networks (cnns) demystified." [Online]. Available: https://www.analyticsvidhya.com/blog/2017/06/architecture-of-convolutional-neural-networks-simplified-demystified/

[8] P. Tangseng, K. Yamaguchi, and T. Okatani, "Recommending outfits from personal closet," in *2017 IEEE International Conference on Computer Vision Workshops, ICCV Workshops 2017, Venice, Italy, October 22-29, 2017*, 2017, pp. 2275–2279. [Online]. Available: https://doi.org/10.1109/ICCVW.2017.267

[9] T. Nakamura and R. Goto, "Outfit generation and style extraction via bidirectional LSTM and autoencoder," *CoRR*, vol. abs/1807.03133, 2018. [Online]. Available: http://arxiv.org/abs/1807.03133

[10] G. Cucurull, P. Taslakian, and D. Vázquez, "Context-aware visual compatibility prediction," *CoRR*, vol. abs/1902.03646, 2019. [Online]. Available: http://arxiv.org/abs/1902.03646

Krutika Mude is a Graduate Student in San Jose State University. She is Software Engineering student currently pursuing dual specialization in Enterprise Software Technologies and Data Science.

KrishnaPriya Gajula is a Graduate Student in San Jose State University. She is Computer Engineering student currently pursuing specialization in Data Science.

Vaidehi Naik is a Graduate Student in San Jose State University. She is Computer Engineering student currently pursuing specialization in Data Science.

Varsha Amrutlal Katariya is a Graduate Student in San Jose State University. She is Computer Engineering student currently pursuing specialization in Data Science.