Content-based Clothing Recommender System using Deep Neural Network

Narges Yarahmadi Gharaei
Faculty of Computer Engineering,
K. N. Toosi university of Technology
Tehran, Iran
yarahmadi@email.kntu.ac.ir

Chitra Dadkhah
Faculty of Computer Engineering,
K. N. Toosi university of Technology
Tehran, Iran
dadkhah@kntu.ac.ir

Lorence Daryoush
Amirkabir Robotic Research Institute(ARRI)
Amirkabir University of Technology
(TehranPolytechnic)
Tehran, Iran
Lorence@aut.ac.ir

2021 26th International Computer Conference, Computer Society of Iran (CSICC) | 978-1-6654-1241-4/20/$31.00 ©2021 IEEE | DOI: 10.1109/CSICC52343.2021.9420544

***Abstract*—A recommender system primary purpose is to provide a series of item suggestions on a topic to its user. Deep learning is used in many fields and solved difficult and complex problems with large volumes of data. Deep learning can also be used in referral systems. Today, online shopping systems are looking for a method that can recommend items according to the user preference and interest in order to increase their sales. Clothing sales systems offer a set of recommendation based on the needs and interests of the users. Today, due to the current situation caused by the Coronavirus, the majority of tasks are done online. In this paper, we propose a content-based clothing recommender system using deep neural network. In content-based systems, product features are required for prediction of unobserved items ratings. In our proposed system by using a deep neural network, the cloth category is obtained and the need to manually extract the product features is eliminated by producing the required features with a large and useful volume. The advantage of this system is that it uses the same network to specify gender as a feature in making suggestions then shows the results to the user. Different machine learning algorithms are tested and analyzed with and without considering demographic information such as gender. The experimental results show that the loss of our proposed system is lower than the other related systems and solves the cold start problem for new items. Our proposed system also recommends novel, relevant and unexpected items.**

Keywords—Clothing, Recommender System, Deep learning, Demographic, Feature Extraction, Cold start, Content, Coronavirus

I. Introduction

Recommender systems are used in a variety of fields, including music, movies, and books. The main purpose of these systems is helping the users reach their needs faster and it is also useful in online sales systems. Recommender systems are divided into 4 general categories as follows [1].

1) Collaborative Filtering Recommender Systems (CFRS) which find users with close behaviors and tastes. If one of the two users with similar behavior has bought a product and has given it a positive rating, the system recommends this product to the other user guessing with high probability that this user will like this product as well. One of the problems that these systems face is the cold start problem in which when there is a new user/item in the system with no background so no neighbor can be found for user/item.
2) Content-based Recommender Systems (CBRS) unlike the CFRS are not interested in the similarity between users. These systems basically deal with the resemblance and closeness of the items according to the data given to them. Based on target user's history, the items they have bought and the similarity of them with the items available in the system, offers a list of recommendation to the user. The cold start problem is also present.
3) Knowledge-based recommender system is appropriate for special items that are not included in daily basket of online shopping such as car, apartment. In this kind of systems user's preferences will be fed to system to make a recommendation based on user requirement. For instance in case of purchasing a car, information like car mileage, price, and color will be defined by users to filter database according to user desire. This system has no cold start problem.
4) Hybrid recommender systems are combination of different recommender systems; the use of Hybrid recommender systems has become much more prevalent due to their high accuracy.

Machine learning (ML) is a portion of artificial intelligence that makes system learned from data without explicitly programming. Among the application of machine learning, we can mention business [2], advertising [3] and medicine [4].

Deep learning the most common method to model a problem, is a branch of ML. it has the ability to solve many challenging problems in machine learning [5]. It can be used in variety of fields such as text processing, image processing, audio and signal processing and is also applicable in recommender systems. CBRS needs its products features and extracts them in different ways, which will be explained in the next section.

In this paper, we use deep neural networks to identify the product category and gender and exploit the same network to extract features.

The structure of the paper is as follows: we referred the related works in Section II and explained the problem in Section III. The dataset used in our system is introduced in Section IV. Section V is dedicated to explaining the strategy and the used deep neural networks. Section VI shows the experimental results. Finally Section VII presents the conclusion.

II. RELATED WORKS

Here are some of the recent developments in content-based recommender systems based on machine learning.

Younus et al in [6] proposed Content-Based Image Retrieval (CBIR) method aim to retrieve images accurately from large image databases similar to the query image based on the similarity between image features. A new hybrid method has been proposed for image clustering using the combination of the Particle Swarm Optimization (PSO) with k-means clustering algorithms. Their method uses the color and texture images as visual features to represent the images.

Authorized licensed use limited to: East Carolina University. Downloaded on June 21,2021 at 01:25:16 UTC from IEEE Xplore. Restrictions apply.

Their proposed method is based on four feature extractions for measuring the similarity, which are color histogram, color moment, co-occurrence matrices, and wavelet moment.

Ali et al [7] uses SIFT feature extraction algorithm for feature extraction, which gives the key point in an image for implementing the features that CBIR needs. They use the optimization technique BFOA (Bacteria Foraging Optimization Algorithm) to reduce the complexity, cost, energy, and Time consumption because of not being valuable the SIFT image feature algorithm.

There have been a lot of researches and works in content based recommender systems, but in a lot of them deep learning is not used, for example the researches in [8, 9, 10] so they are not included in our review.

Yun-Rou Lin et al in [11] presented (appropriate) clothing with proper size recommendation by considering factors such as gender, body height and clothing features. Online texture modeling was implemented to produce variety in the texture of clothing so their proposed system can create logical and diverse options for consumers. They used CNN (Convolutional Neural Network) for gender recognition, Dlib for face detection [12], and InceptionV3 (GoogleNet) for clothing attributes recognition. Their attributes contain five clothing categories with eight dimensions.

Batuhan AŞIROĞLU et al in [13], developed a recommender system by taking just a single photo of the user with their scalable embedded system. Their recommendation is made regardless of shopping history of the user. The utilized hardware is a Linux based system and a low-cost Raspberry Pi Zero W. They developed two inceptions based CNN, one for the prediction part and one for feed-forward NN as a recommender system. The prediction part exploits from Haar-Cascade of OPENCV python library for user's face gender detection. The personal information extraction is done by inception based on deep CNN from the processed image. The last step is recommending the best-fitted cloth to the user by another CNN. Their cloth dataset contains 132 samples.

Yufan Wen et al in [14],, constructed a knowledge graph of the user, a knowledge graph of clothing, and a knowledge graph of context. They used the Apriori algorithm to capture the correlation between clothing attributes and context attributes. They generated recommendation results directly by taking into account the top-N algorithm and searching and matching the established knowledge graph which is based on user's requirements.

III. MOTIVATION

Today, due to the current situation caused by the Coronavirus, the majority of tasks are done online. Online clothes shopping systems are one of them. Although they have excited before but because of the current situation a lot more attentions have been turned towards them. Therefore, the users of these systems have increased compared to the past so now we are dealing with big data. LUISAVIAROMA, Fashionphile, Runway Catalog, and Amazon are examples of these systems.

So, the main challenge of the sale industries is creating a system to deal with big data. Users usually get confused when they face a wide range of options. Therefore, there is a need for a system that immediately offers the right target product based on the user's requests and tastes. Sales systems also need a system that improves the speed, quantity, and quality of their sales. Therefore, clothing recommender systems bring mutual benefit for both customers and the merchant systems. So, we provided a recommender system that meets all these goals.

Content-Base recommender systems require product features. In a sense, due to the fact that the data volume of these products is considerably high, entering the features of these products manually into system is extremely time-consuming. One of the most important benefits of using deep learning is its ability to execute feature extraction. Another benefit of it is the elimination of the need for data labeling and maximum utilization of unstructured data. So these reasons encouraged us to use deep learning.

We have presented a recommender system that eliminates the need to provide the product features manually. Our system extracts its required features with high volume from just one product image using a Deep Neural Network (DNN). These features are more explained in section V. So, for each new product in the system, the feature extraction process is performed only by presenting an image of the product. Also our system has solved the cold start problem for new items.

For recommendation, it is better to identify the gender of the product so that the clothes we offer are suitable for them. The gender recognition section of our system is also performed by a DNN. The systems in [13] performed gender identification solely on the user's face image. Gender of the items in their system are already available and they do not perform diagnostic operations. These systems when compared to the proposed system in [13] have the following differences.

1. Our system does not need the face of the model to recognize the gender. The procedure is done based on the style and the type of clothing.
2. We have eliminated the need of defining the gender attribute for the product.

Our proposed system also uses the same DNN for product gender recognition and feature extraction and it also recommends novel, relevant and unexpected items.

IV. DATASET

The utilized data set used in the gender detection, feature extraction and suggestion is Fashion Product Images (Small) from Kaggle site. This database contains 44,000 images of different types of products. 50% of the images in this database are related to men's products, 42% belongs to women and the remaining 8% belongs to Unisex, Girls, and Boys. 48% of images are related to Apparel, 52% belong to Accessories and the remaining 26% belong to other categories. In this paper, we have only used images related to Apparel. The total number of categories in this database is 143, which we have only used the categories mentioned in the table I. Therefore, we have worked on 14932 images. Fig. *1* and Fig. *2* display data scatter based on article type and gender and the numbers written in the pie chart is their percentage. More detailed information about this database is available on the Kaggle website at https://www.kaggle.com/paramaggarwal/fashion-product-images-small.

Authorized licensed use limited to: East Carolina University. Downloaded on June 21,2021 at 01:25:16 UTC from IEEE Xplore. Restrictions apply.

TABLE I. ARTICLE TYPES IN DATASET

Article type	counts
Waistcoat	15
Stocking	32
Skirts	128
Jackets	258
Track pants	304
Dresses	464
Trousers	530
Shorts	547
Jeans	608
Tops	1761
Shirts	3215
T-shirts	7060

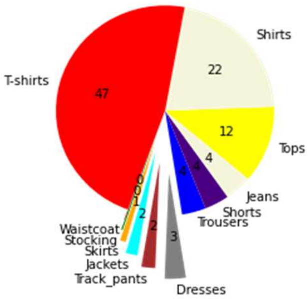

Fig. 1. Pie chart of data scatter rate based on article type.

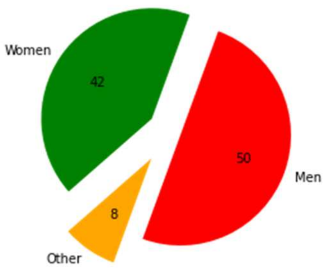

Fig. 2. Pie chart of data scatter rate based on gender type.

V. PROPOSED METHOD

Our proposed system uses a deep convolution neural network to detect categories, product gender, and feature extraction from an image. A Convolutional Neural Network (CNN, or ConvNet) is a class of deep neural networks, mostly used to analyze images and videos. CNNs are a version of fully connected multilayer perceptron. That means each neuron in each layer is connected to all neurons in the next layer. Each convolution layer convolves the input by a specific filter and passes its result to the next layer [15]. As images have high dimensions, every pixel of an image is considered as a feature, so when we want to pass it to a multilayer perceptron the volume of computations and parameters grows rapidly. CNNs are an efficient solution for reducing the number of parameters without losing quality on the models. Since our data is an image, we also use CNNs. We have implemented our design model with Keras for all three purposes as shown in the Fig. 3.

The names of the methods used in different layers in the model is written on the guide on the right side of the figure according to their color used in the schematic. This network has 13 layers. The kernel size for all layers is 4*4. Activation function of all layers except the last one is RELU. The last layer activation function is SIGMOID. This network is only trained once for learning the gender of the products.

Input of this network is images and its labels are based on the gender category of dataset and we call the trained model GenderModel. Then, to identify the clothing article type, we train this network by images as input and articles types as labels, and this time we save the model weights in a separate file from the previous model named ArticleTypeModel. The training is done with 50 epochs. Our loss function is Categorical_focal_loss with alpha 0.25 and gamma equals to 2. We minimized the error by Nadam [16] optimization algorithm. The mentioned alpha and gamma value gave us the best output among the various tests. Our metric for evaluation is accuracy.

Two models created by this network are used to obtain the features of the items. We have considered the output obtained by the weights of the 12th layer as extracting features layer in this network. The size of the resulting vector by both networks is (14932, 68). Therefore, we have merged the two vectors of the extracted features together and used them as image features. So the size of the vectors becomes (14932, 136) by considering feature extraction on the network, it can be pointed out that, this vector is a black box. Considering the high dimension of the obtained vector, the features we have achieved are certainly better than the features extracted by

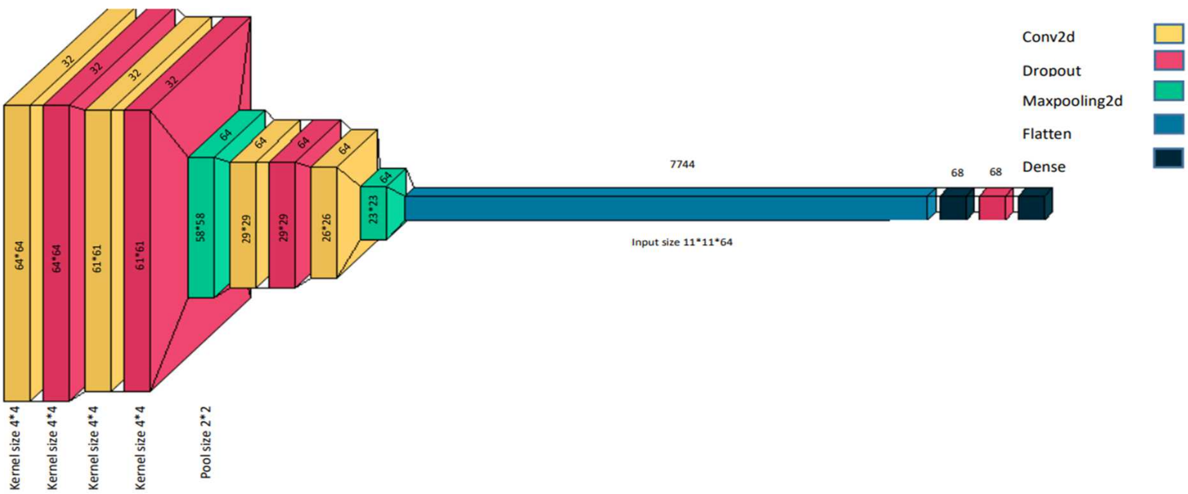

Fig. 3. Fully convolutional neural network schema in our proposed system

Authorized licensed use limited to: East Carolina University. Downloaded on June 21,2021 at 01:25:16 UTC from IEEE Xplore. Restrictions apply.

humans. Each feature that we have some information about it in dataset can be included in the training phase. We just tested 2 explicit feature (gender and article type) and the result was satisfying. Including more explicit features like brand and quality are guessed to achieve better results. Although the model may extract these features from the image in spite of involving them explicitly.

Weights are already set by the training phases on the available data. After adding new data to our data set the training process will be continued by the previous weights so less time is spent for training the new data.

The similarity between the extracted feature vectors of the images is calculated using the cosine similarity as Eq. 1. To receive a recommendation from the system, it is enough to obtain the feature vector of the user image by both models and then combining them, using the cosine similarity, find and arrange the products with the most similarity and present the Top-N of the similar products.

$$\cos\theta = \frac{\vec{a}.\vec{b}}{\|\vec{a}\|\,\|\vec{b}\|} = \frac{\sum_1^n a_i * b_i}{\sqrt{\sum_1^n a_i^2} * \sqrt{\sum_1^n b_i^2}} \quad (1)$$

VI. RESULTS AND EXPERIMENTS

The experimental result has been divided into two parts: prediction and recommendation accuracy.

A. *Prediction accuracy*

We have implemented the explained algorithms and ResNet-50 [17] in python on the dataset. 80% of data is used for train and 20% for test. As we have showed in table II loss of ResNet-50 model for gender detection is 0.136 but for our GenderModel is 0.0334. In table III loss of ResNet-50 for Article type detection is 0.1216 and loss of our model is about 0.0334. Also we have presented the confusion matrix and a test of all implementations in Fig. 4 and Fig. 5. According to Fig. *1*, we have realized that we have a lot of heterogeneity in terms of numbers in different categories of data. Therefore, we cannot expect high accuracy from the algorithm.

The elapsed time for training the ArticleTypeModel is 652 seconds and the GenderModel is 735 seconds for training data by our proposed method. As you can see, the training time is acceptable.

TABLE II. COMPARING LOSS AND ACCURACY OF MODELS IN GENDER DETECTION

	Model name	Loss	Accuracy
1	ResNet-50	0.136	86.24%
2	Our Keras model	0.0334	83.29%

TABLE III. COMPARING LOSS AND ACCURACY OF MODELS IN ARTICLE DETECTION

	Model name	Loss	Accuracy
1	ResNet-50	0.1216	68.26%
2	Our Keras model	0.0334	66%

B. *Recommendation precision*

In each row of, some examples of recommended images according to user's purchase history can be seen. The images that have the original label are the images of the purchased product by the user and the rest of images are the top-K items recommended by our proposed system. As it is shown, the recommended clothes are very similar to the original selected clothes by the user. Also, among them, there are several unexpected recommendations which have some similar features and some very different ones. For testing the accuracy of our recommendations, we have used the ground-truth evaluation, so the feedbacks were given by real users and they were analyzed as it is shown in Table IV. The recommendation precision of our proposed system is about 73.7% which means the users liked 73.7% of our recommendations. According to the recommended items in Fig. 6, you can see the degree of novelty with the cosine similarity metric in table IV. As shown in Fig. 6 by considering that the similarity of these 10-Top recommended item don't equal to 1, it can be concluded that the system never recommends the items in user's Purchase history. By using datasets containing more clothes in each category, the recommendation accuracy will increase. The average time required to assign recommendations per user is 0.07 second.

TABLE IV. THE PRESICION OF RECOMMENDATION

User ID	Number of recommends items	Liked items by user	Precision
1	500	358	71.6 %
2	500	313	62.6 %
3	500	354	70.8 %
4	520	418	80.38 %
5	410	295	71.90 %
total	**2430**	**1793**	**73.7 %**

VII. CONCLUSION

Nowadays introducing new products to prevent the user from getting bored is a challenge. In this article, we have presented a method to recommend clothes. Our proposed system solves the cold start problem for new items by considering their features. Our proposed system uses the DNN for extracting the features from the image of the items and our system is evaluated using feedbacks from real users using precision metrics and the results show that the system is both efficient and precise with 73.7% precision.

Recommending serendipitous item is one of the benefits of our proposed system. Also, it has high speed due to its simplicity. In the future, we intend to compare our system with other content base clothing recommendation algorithms which doesn't use deep learning methods and measure and improve the performance of our proposed method. The source code is available at https://colab.research.google.com/drive/1_fhBCR1jbg3cnoh4Afmz7iZaK6xC8Bbo?usp=sharing.

Authorized licensed use limited to: East Carolina University. Downloaded on June 21,2021 at 01:25:16 UTC from IEEE Xplore. Restrictions apply.

TABLE V. COMPARING THE NOVELTY OF RECOMMENDED ITEMS

Original images / Recommended items	13770	52940	32210	11559	59589
Item 0	0.999468	0.999129	0.999672	0.999389	0.999493
Item 1	0.999462	0.999065	0.999578	0.999373	0.999417
Item 2	0.999409	0.999050	0.999459	0.999379	0.999280
Item 3	0.999377	0.999029	0.999432	0.999364	0.999233
Item 4	0.999377	0.998996	0.999422	0.999274	0.999217
Item 5	0.999363	0.998978	0.999422	0.999273	0.999217
Item 6	0.999327	0.998696	0.999422	0.999249	0.999180
Item 7	0.999308	0.998693	0.999363	0.999162	0.999128
Item 8	0.999292	0.999620	0.999310	0.999162	0.999128
Item 9	0.999273	0.998618	0.999308	0.999131	0.999120
Average	**0.999366**	**0.998987**	**0.999439**	**0.999276**	**0.999241**

REFERENCES

[1] C. C. Aggarwal, Recommender Systems, Springer, 2016.

[2] C. Apte, "The role of machine learning in business optimization," *In ICML,* 2010.

[3] Q. Cui, F.-S. Ba, B. Gao and T.-Y. Liu, "Global Optimization for Advertisement Selection in Sponsored Search," *Journal of Computer Science and Technology 30,* vol. no. 2, pp. 295-310, 2015.

[4] A. Rajkomar, J. Dean and I. Kohane, "Machine Learning in Medicine," *New England Journal of Medicine 380,* vol. no. 14, pp. 1347-1358, 2019.

[5] Y. LeCun, Y. Bengio and G. Hinton, Deep learning, Nature, 2015.

[6] Y. Zeyad Safaa , M. Dzulkifli , S. Tanzila , Mohammed Hazim Alkawaz, R. Amjad , Mznah Al-Rodhaan and Abdullah Al-Dhelaan, "Content-based image retrieval using PSO and k-means clustering algorithm.," *Arabian Journal of Geosciences 8,* vol. no. 8, pp. 6211-6224, 2015.

[7] A. Ali and S. Sharma, "Content based image retrieval using feature extraction with machine learning," *2017 International Conference on Intelligent Computing and Control Systems (ICICCS),* pp. 1048-1053, 2017.

[8] P. K. Gopalan, L. Charlin and D. Blei, "Content-based recommendations with Poisson factorization.," *Advances in Neural Information Processing Systems,* p. 3176–3184, 2014.

[9] A. Pujahari and V. Padmanabhan, "An Approach to Content Based Recommender Systems Using Decision List Based Classification with k-DNF Rule Set," in *IEEE*, Bhubaneswar, India, 22-24 Dec. 2014.

[10] L. Yu-Chu, Y. Kawakita, E. Suzuki and H. Ichikawa, "Personalized Clothing-Recommendation System Based on a Modified Bayesian Network," in *IEEE*, Izmir, Turkey, 16-20 July 2012.

[11] Y.-R. Lin, W.-H. Su, C.-H. Lin, B.-F. Wu, C.-H. Lin, H.-Y. Yang and M.-Y. Chen, "Clothing Recommendation System based on Visual Information Analytics," in *2019 International Automatic Control Conference (CACS)*, Keelung, Taiwan, Taiwan, 05 March 2020.

[12] D. E. King, "Dlib-ml: A Machine Learning Toolkit," *Journal of Machine Learning Research,* vol. 10, pp. 1755-1758, December 2009.

[13] B. AŞIROĞLU, M. İ. ATALAY, A. BALKAYA, E. TÜZÜNKAN, M. Dağtekin and T. ENSARİ, "Smart Clothing Recommendation System with Deep Learning," in *IEEE*, Ankara, Turkey, Turkey, 11-13 Oct. 2019.

[14] Y. Wen, X. Liu and B. Xu, "Personalized Clothing Recommendation Based on Knowledge Graph," in *IEEE*, Shanghai, China, 06 September 2018.

[15] K. O'Shea and R. Nash, "An introduction to convolutional neural networks.," *arXiv preprint arXiv:1511.08458,* 2015.

[16] T. Dozat, "Incorporating Nesterov Momentum into Adam," *ICLR Workshop,* 2016.

[17] K. He, S. R. X. Zhang and J. Sun, "Deep residual learning for image," *The IEEE Conference on Computer Vision and Pattern,* June 2016.

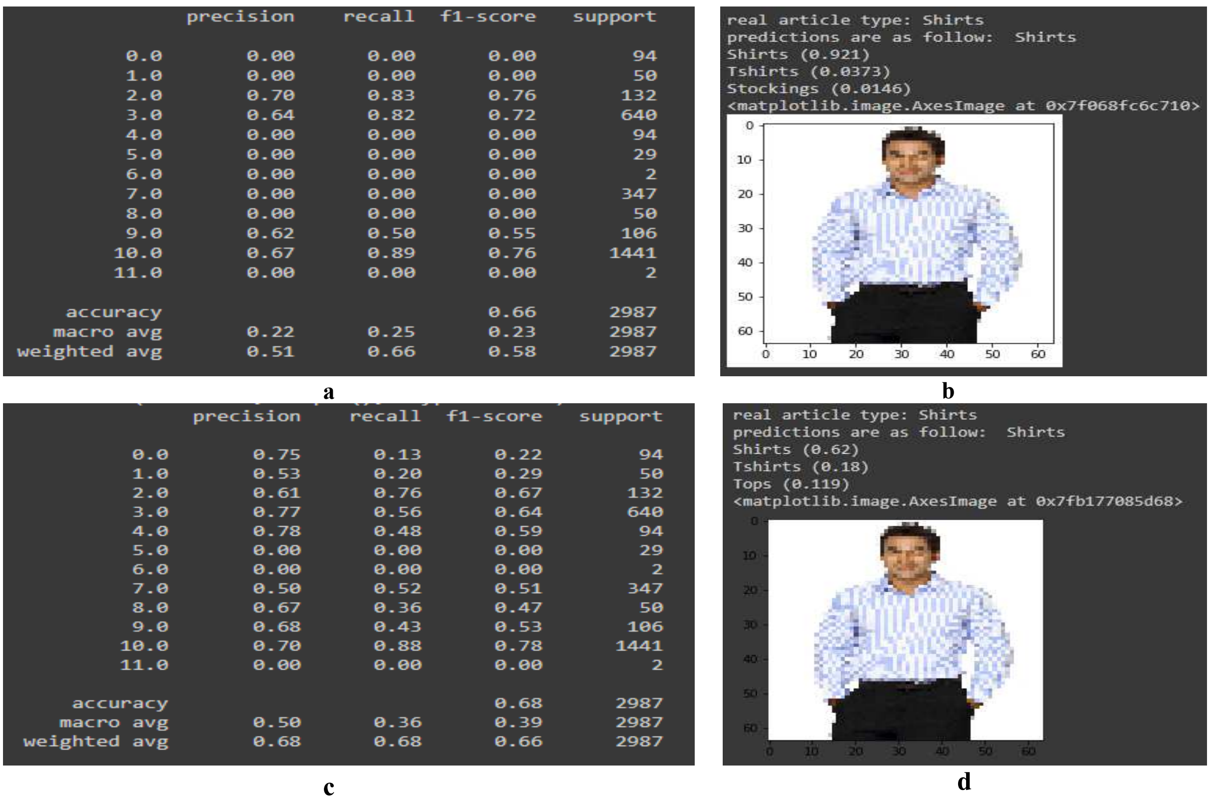

Fig. 4. a) Confusion matrix of our keras model for article type b) Our keras ArticleTypeModel test c) Confusion matrix of ResNet-50 for article type detection d) ResNet-50 artice type detection model test

Authorized licensed use limited to: East Carolina University. Downloaded on June 21,2021 at 01:25:16 UTC from IEEE Xplore. Restrictions apply.

	precision	recall	f1-score	support
0.0	0.00	0.00	0.00	146
1.0	0.00	0.00	0.00	87
2.0	0.80	1.00	0.89	1952
3.0	0.00	0.00	0.00	19
4.0	0.98	0.69	0.81	783
accuracy			0.83	2987
macro avg	0.36	0.34	0.34	2987
weighted avg	0.78	0.83	0.79	2987

a

b

	precision	recall	f1-score	support
0.0	0.57	0.66	0.61	146
1.0	0.56	0.21	0.30	87
2.0	0.87	0.97	0.92	1952
3.0	0.00	0.00	0.00	19
4.0	0.92	0.73	0.82	783
accuracy			0.86	2987
macro avg	0.59	0.51	0.53	2987
weighted avg	0.86	0.86	0.85	2987

c

d

Fig. 5. a) Confusion matrix of our keras model for gender detection b) Our keras GenderModel test c) Confusion matrix of ResNet-50 model for gender detection d) ResNet-50 gender detection model test

Fig. 6. Top-K recommendation based on original user clothing

Authorized licensed use limited to: East Carolina University. Downloaded on June 21,2021 at 01:25:16 UTC from IEEE Xplore. Restrictions apply.