The Government of the Russian federation

Federal State Autonomous Educational Institution of Higher Professional Education

National Research University – Higher School of Economics

Faculty of Computer Sciences,

"Introduction to machine learning and data mining"

Final project

«The Recommendation System With Implemented CNN»

Group of students

Kurmasheva Anel

Alkiz Nafe

Sofia Tkhan Tin Le

Table of contents

Problem statement

As introduced by the name of the project, it is about a recommendation system with a CNN model implemented. The dataset used for this model contains the images of different clothes types and their descriptions that are common in a marketplace. The CNN will use the embeddings to identify similar elements, and this information will recommend similar content in RecSys. The prediction of articles will be in the future recommendation system dataset.

There the Recommendation system will start. Two types of recommendation this project will show. First is the simple similarity-based recommendation. It will present how much they are similar in description. Second is the recommendation system made by the collaborative filtering technique. Furthermore, by the idea, it will show rating based on users tastes.

Dataset summary

Fashion Product Images Dataset contains 44000 products of products with multiple category labels, description and high resolution images.

https://www.kaggle.com/paramaggarwal/fashion-product-images-dataset/data

The 'images.csv' is a table (44446 rows × 2 columns) which contains an ID of an item with the link to the corresponding item image which allows to work with images.

The 'styles.csv' is a map to all products. This table contain main features and categories of the images. It's also possible to get the complete metadata in '.json' format. There are 10 columns with the information about:

1. ID of the item (e.g. 21379, 53759)
2. Year of clothes manufacture (Fig. 1)

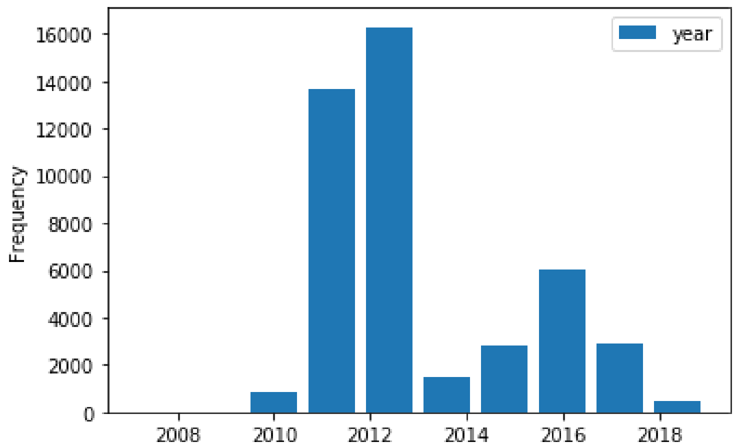

Figure 1. Frequency of occurrence of a product in a certain year

3. Type of clothes (15.9% T Shirts, 7.2% shirts, 6.4% casual shoes etc.) (Fig. 3)

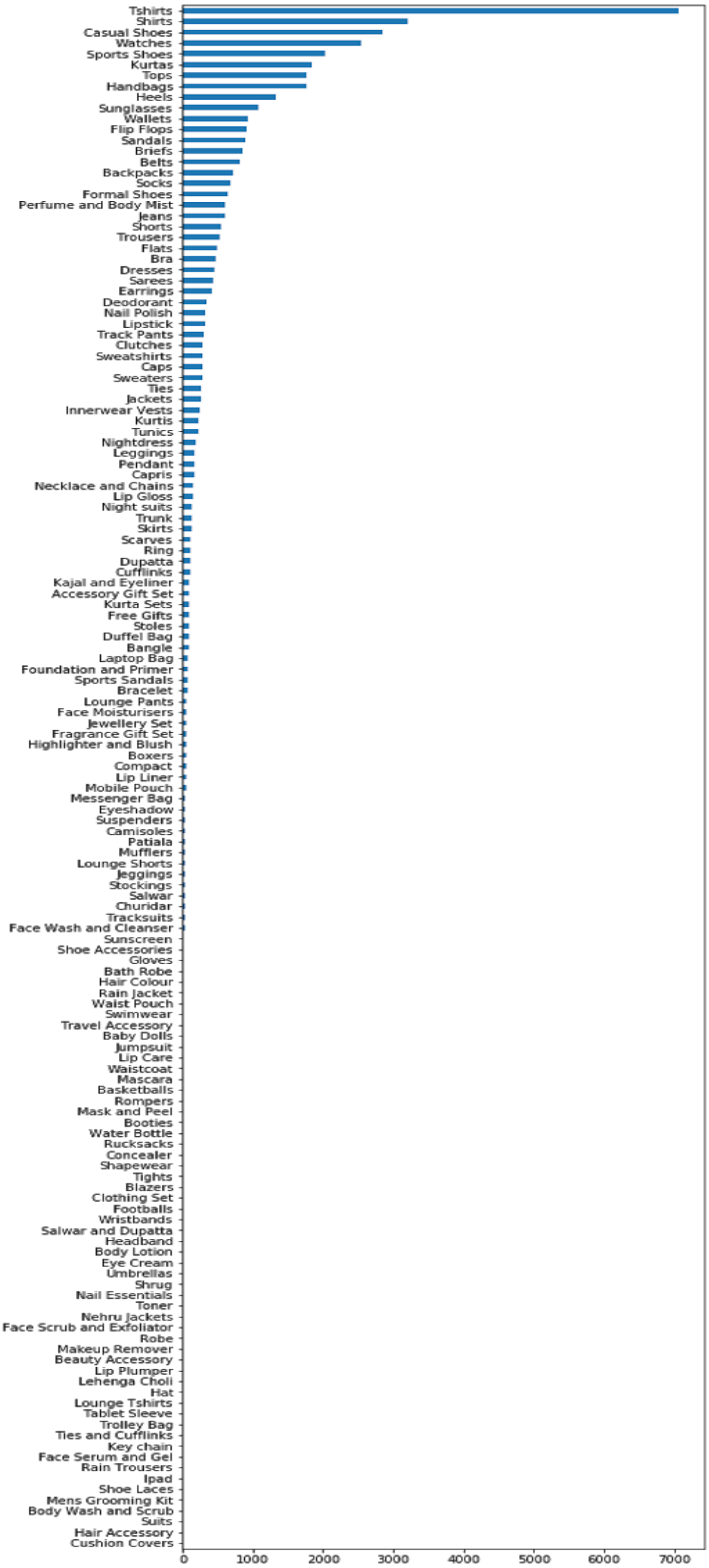

Figure 3. The frequency of the particular type of clothes in this dataset

4. Base color of an item (21.9% black, 12.47% white, 11% blue etc.) (Fig. 5)

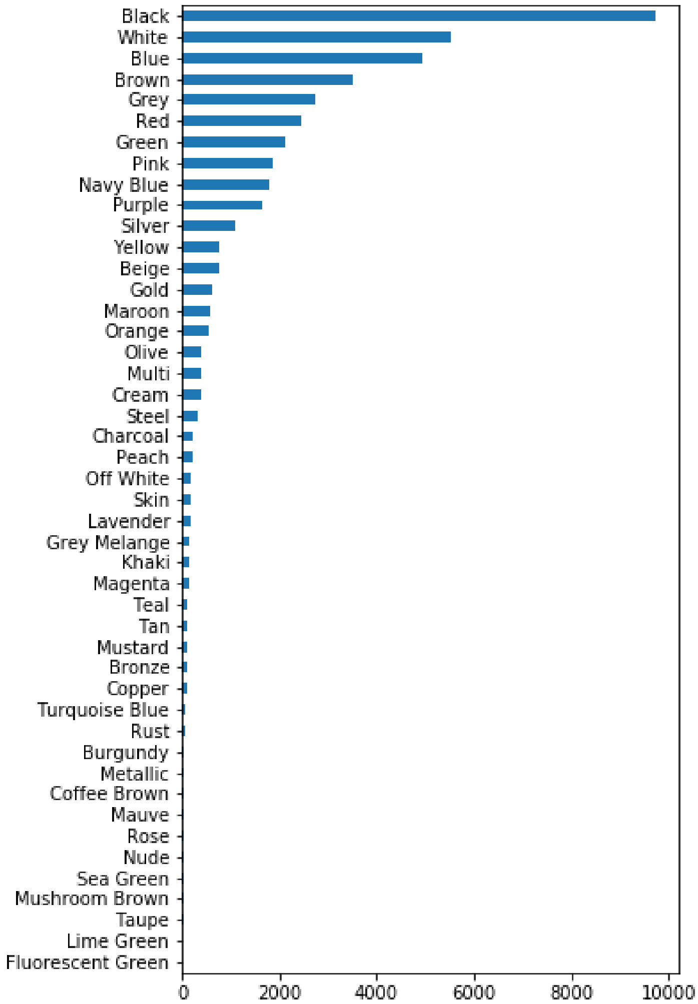

Figure 4. The frequency of the particular color of clothes

5. Gender (49.85% men, 41.93% women, 4.86% unisex etc.) (Fig. 5)

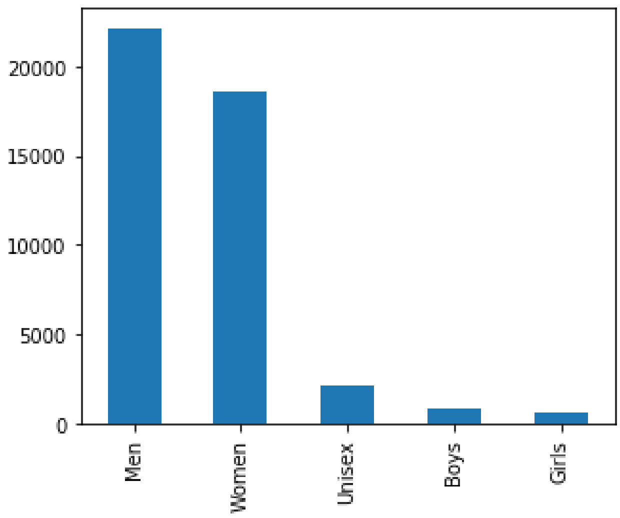

Figure 5. The frequency of the gender affiliation

6. Possible usage (78% casual, 9% sports, 7.2% ethnic etc.) (Fig. 2)

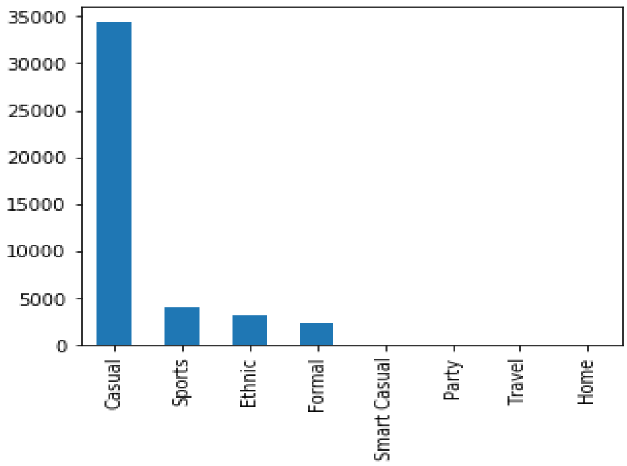

Figure 2. Frequency of occurrence of a product according to the possible usage

7. For what season is this item suitable (48.35% for Summer, 25.74% for Fall, 19.18% for Winter and 6.71% for Spring)
8. Master category of the item (48.1% apparel, 25.37% accessories, 20.75% footwear etc.)
9. Subcategory of the item (34.67% topwear, 16.52% shoes, 6.87% bags etc.)
10. Product display name (e.g. "Turtle Check Men Navy Blue Shirt", "Peter England Men Party Blue Jeans")

Methodology

CNN method

At CNN, the dataset was split into two parts, a train of 80% and a test of 20%. This proportion was chosen due to the fact that many positions do not have enough examples, and if they were split in half, the prediction would be much worse. Based on it, the type of clothing was predicted.

All items of clothing items were digitized, and all images were converted to gray scale and resize to 1 * 28 * 28. This approach can significantly reduce the size of memory consumed by 3 times and accelerated learning.

Data preparation for recommendation system

For recommendation systems, there was a division of dataset into two groups according to the "sex" variable obtained by CNN. The first group contained clothes for men, and the second group contained clothes for women. The purpose was not to recommend women's closing to men visa versa.

All categorical variables of the dataset converted to dummy variables (all values are in numerical value 1 or 0) except the "year" of product collection. The CNN model tested through separate recommendations for data with real articles and data with predicted by neural network articles.

Since the data did not contain real users and their object ratings, we generated them separately by creating three types of behaviour based on the similarity of objects using the Euclidean metrics. A random favourite product generated for 23 users based on which similarity ratings and ratings from 50-250 random objects created. After this stage of preparation, the data is ready for collaborative filtering.

Recommendation system

A simple recommender system was built based on the similarity of objects. The similarity of the objects uses the Euclidean metric. The result was then normalised to range [0, 1]. According to this metric, a logical interpretation is the following: the smaller the value of the metric, the more similar the objects are. So, the ideal similarity is '0', and the maximal difference is '1'.

The collaborative filtering uses with the synthetic data. This recommendation is item-based, and output rating is a 5-point scale (5 - excellent, 1 - very bad). The similarity of objects is also calculated based on the Euclidean metric. Then the rating prediction used the kNN (k-nearest neighbours) method.

Experiment setup and results

CNN description

In deep learning, a convolutional neural network (CNN, or ConvNet) is a class of deep neural networks, most commonly applied to analyzing visual imagery. They are also known as shift invariant or space invariant artificial neural networks (SIANN), based on their shared-weights architecture and translation invariance characteristics. They have applications in image and video recognition, recommender systems, image classification, medical image analysis, natural language processing, and financial time series.

This part briefly describes the architecture of the LeNet5 network that we used to recognize clothes (Fig. 6). The architecture consists of 5 layers.

1. First layer

Input: Tensor 1 * 1 * 28 * 28

Output: Tensor 1 * 6 * 14 * 14

This layer consists of Conv2d that has a core of size 5 and padding equal to 2. The first layer converts the input image (Fig. 7) to 1 * 6 * 28 * 28 and after that AveragePool application reduces dimension of given tensor to 1 * 6 * 14 * 14.

Than Activation Features

$$g(x) = \frac{e^x}{1 + e^x}$$

2. Second layer

Input: Tensor 1 * 6 * 14 * 14

Output: Tensor 1 * 16 * 5 * 5

3. Third layer

Input: 1 * 5 * 5 * 16

Output: 120

Third layer converts 2D tensor to 1D and Than activation function

4. Fourth layer

Input: 120

Output: 84

Works as a filter Than activation function

5. Fifth layer

Input: 84

Output: 142

Works as a Softmax

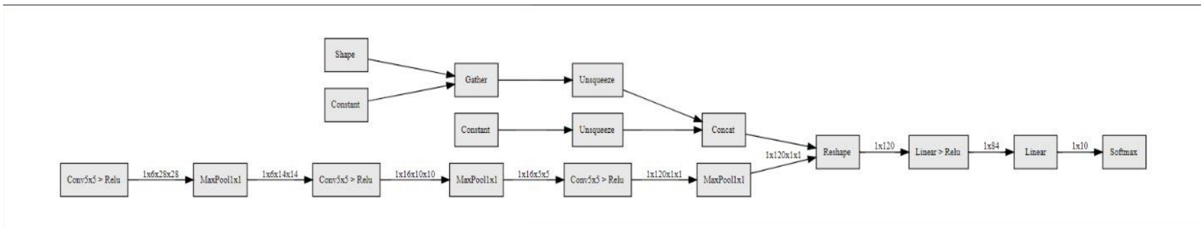

Figure 6. LeNet5 Architecture

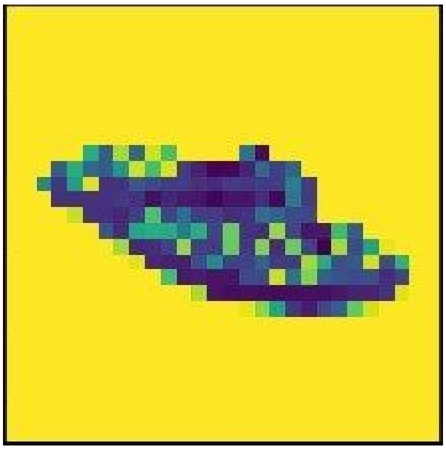

Figure 7. An example of the input image

CNN metrics

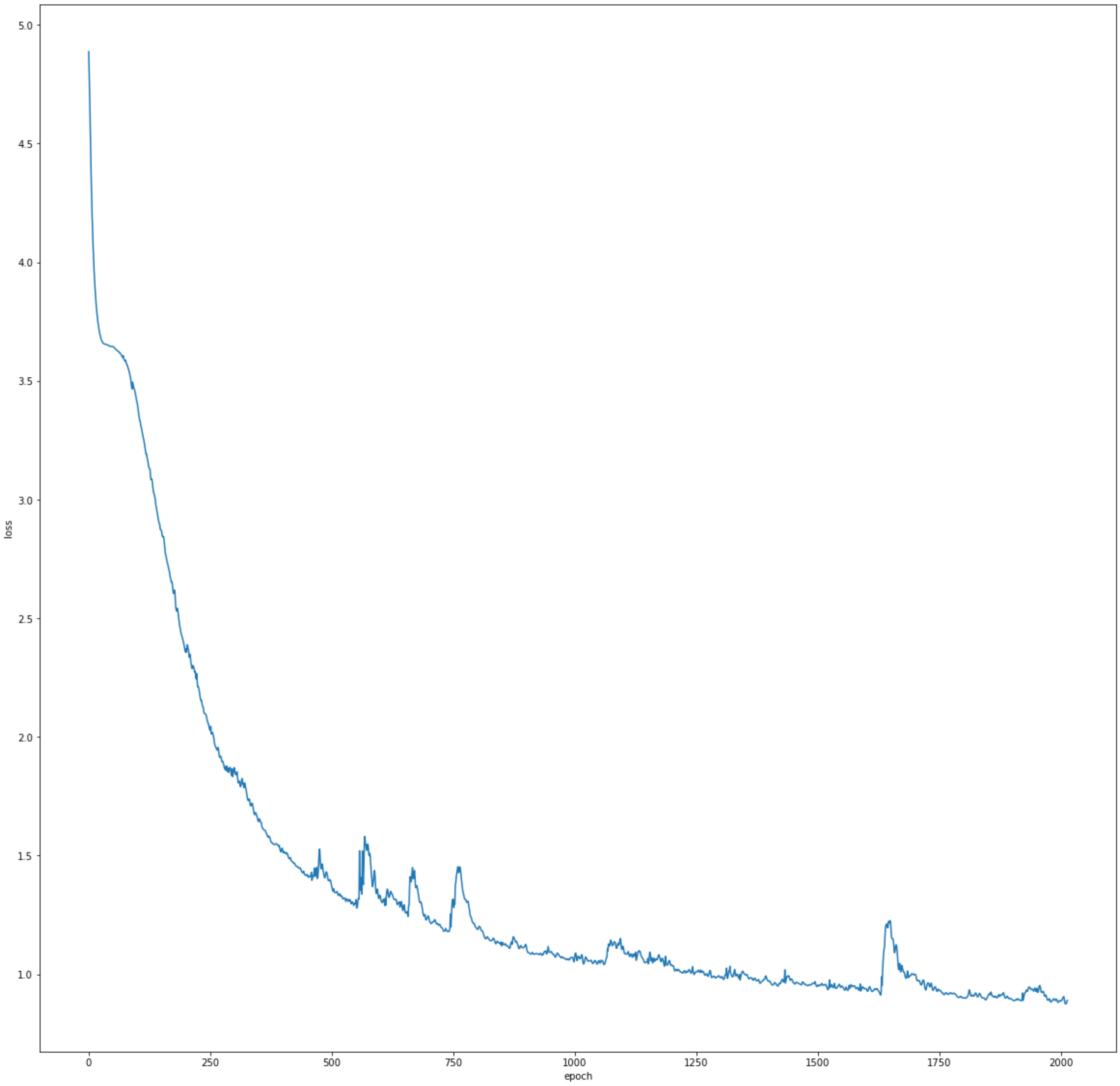

Figure 8. Graph of the Loss Function for 2000 epochs

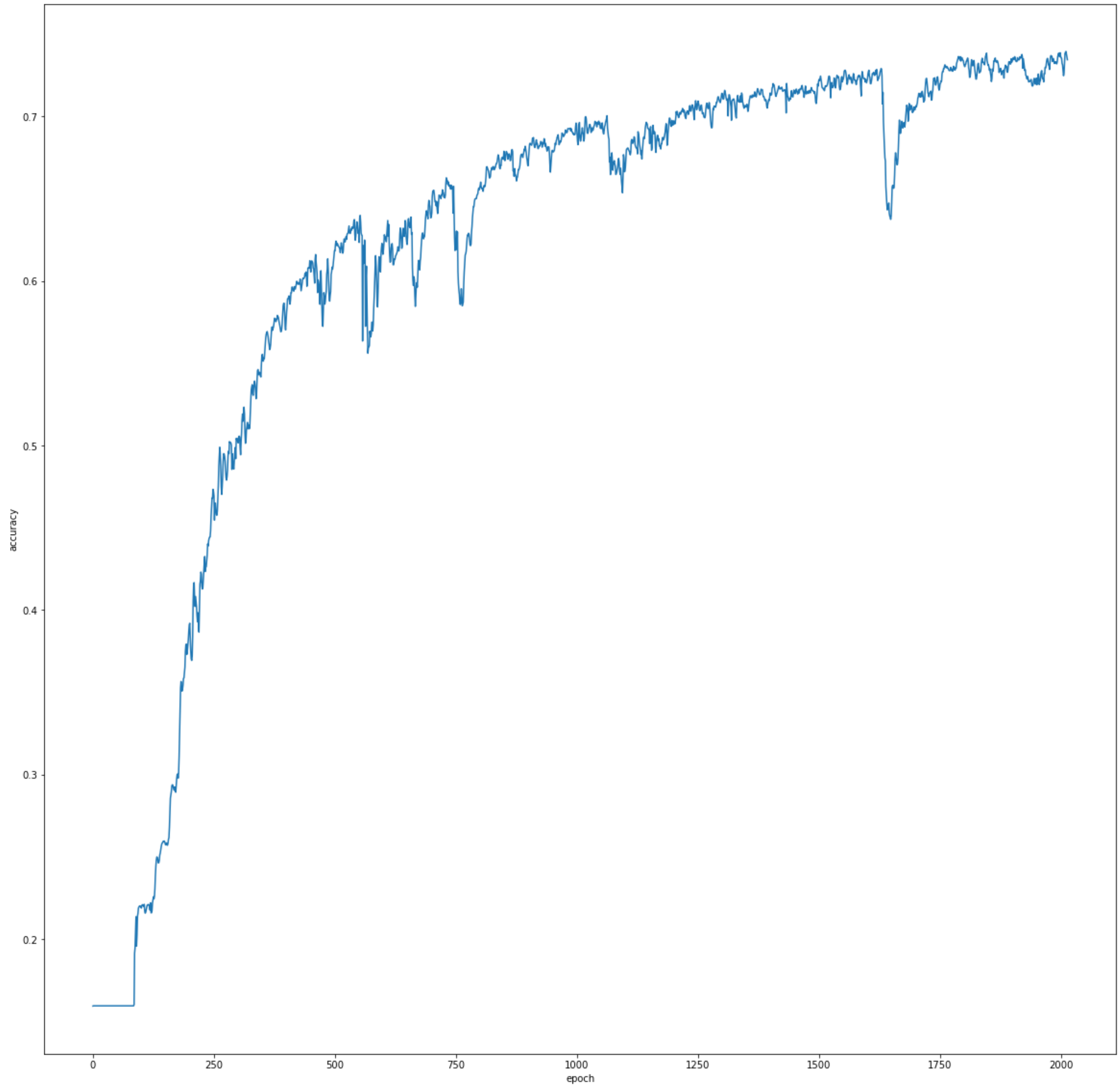

Figure 9. Graph of the accuracy for 2000 epochs

Best accuracy	Best loss
0.7395	0.5982

Simple recommendation system results

The result of simple recommendation was a list of normalised similarities over all dataset. Then ascendingly sorted list can be interpreted as rating list, where higher items are more preferable over lower ones. For proper check of results between real and generated by CNN data we take products from the same places in rating and check also their similarity mark.

Item ID 12369 	Real male data		Predicted male data	
Similarity	Similarity metric value	Recommended item ID	Similarity metric value	Recommended item ID
Ideal (0.0-percentile)	0.0	9248 	0.0	9248 
	0.0	9253 	0.0	9253 
Good (0.25-percentile)	0.324	17501 	0.324	17511 

	0.324	17502	0.324	17512 
Normal (0.5-percentile)	0.374	14605 	0.374	13939 
	0.374	14606 	0.374	13945 
Bad (0.75-percentile)	0.458	34234 	0.458	34234 
	0.458	34235 	0.458	34235 

Very bad (1.0-percentile)	1.0	50764 	1.0	50764 
	1.0	50765 	1.0	50765 

Table 1. Resulting recommendations based on the real and predicted male data

Item ID 20099 	Real female data		Predicted female data	
Similarity	Similarity metric value	Recommended item ID	Similarity metric value	Recommended item ID
Ideal (0.0-percentile)	0.0	11790 	0.0	11790 

	0.0	11794 	0.0	11794 
Good (0.005-percentile)	0.447	11545 	0.447	11545 
	0.447	11547 	0.447	11547 
Normal (0.01-percentile)	0.632	11247 	0.632	11247 
	0.632	11370 	0.632	11370 

Bad (0.2-percentile)	0.774	27428 	0.774	27428 
	0.774	27434 	0.774	27434 
Very bad (1.0-percentile)	1.0	45366 	1.0	45366 
	1.0	45367 	1.0	45367 

Table 2. Resulting recommendations based on the real and predicted female data

Discussion

In this project, we attempted to build a recommendation system based on available labelled data containing images of clothes from online stores.

We are fully aware that collaborative filtering should be created based on real user data, thus making non-trivial recommendations since the mathematical similarity of products generated user tastes, recommendations exactly not give the accurate magic taste prediction result. However, even though the obtained data were synthetic, the method we tested provided some results for further consideration.

Conclusion

In conclusion, the main achievement of our project is a successful attempt to combine CNN with a recommendation system. It revealed that CNN gives a similar result, which allows us to present in the future a product that can take a picture of the clothes they like as input and receive a recommendation based only on the picture data.