A Supervised and Unsupervised Approach for DeepFashion Category Prediction

Abhinav Gupta
MS Computer Science
New York University
New York , NY 10002
abhinav.gupta@nyu.edu

Anuj Menta
MS Computing, Entrepreneurship and Innovation
New York University
New York , NY 10002
menta@nyu.edu

Abstract

Data can be classified by two methods, supervised and unsupervised. In this paper, we explore various approaches to classify the Deep Fashion dataset. We present present two models, one which is supervised and one which is unsupervised. Our supervised model provides better results than the state of the art models proposed till now. We also present, so far unexplored, a self supervised approach to classify the DeepFashion dataset to handle the major problem of data labelling. The autoencoder proposed does a lossy compression using fully connected CNNs, combined with PCA and KMeans is used for category prediction.

1 Introduction

Category prediction for clothes, or classification of clothes has been gaining a lot of traction in the recent times. A lot of research is being put into the classification of clothing items, retrieval etc. This is majorly due to their value and the never ending demand for clothes in the industry. Clothing is an ever growing market with a database growing at a faster pace.
Recognition systems for clothes faces a real world challenges. One of them is the vast variety of styles that clothes have to offer. With fashion changing rapidly, a t-shirt can look very different from piece to piece, brand to brand, and further, culture to culture. Moreover, the clothing items wear and tear, and change in appearance over time, for example, color fades over time, or the general wear tear. Furthermore, they tend to be perceived differently based on the angle of the image taken, for example, a t-shirt will look very different in a selfie than a full body picture. This is because of, there are changes in style, and changes in camera angle more often than not fails to capture general features of the clothing item.
These challenges make this problem, a particularly hard one to solve, even so that it is sometimes hard for humans as well to identify and classify the piece of clothing. Also, with the ever growing data at our disposal today, there is a need for an unsupervised classifier to help label and classify the data.
In this paper, we explore and propose a supervised and an unsupervised method for classifying the the images from the DeepFashion dataset. We trained a supervised classifier to predict the classes. For this we train a resnet34 and a resnet50 to use the labels to classify the images into 50 categories. Train an self-supervised model on the same dataset to predict (or) cluster the images to assign categories. To achieve this we used an autoencoder for a lossy compression and reconstruction of the image and then filtered the encoder feature map using PCA to then classify the dataset using KMeans clustering.

1.1 Dataset and problem

The dataset for the DeepFashion[1] dataset contains over 800,000 images. We chose this dataset sure to the vast variety of images it offers. This data set is richly annotated with massive attributes[3,4],

clothing landmarks and correspondence of images taken under different scenarios including store, street snapshot, and consumer classified by type, texture and design.
There are four different categories of data and corresponding problems which could be solved using this data:

- Category and Attribute Prediction
- In-shop Clothes Retrieval
- Consumer-to-shop Clothes Retrieval[9]
- Fashion Landmark Detection

Category and Attribute Prediction Benchmark has a dataset of 50 categores and 1000 attributes of clothes. This has a large collection of categories and attributes with good number of images available. Therefore, choose this part of the dataset for our supervised and unsupervised classification of fashion images.

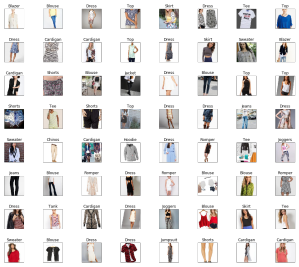

Figure 1: Brief look at the dataset

According to the paper [], Clothing Category classification is considered a fine-grained image classification task, since, most of the clothing looks visually very similar and have high intra-class and low-inter class variance.

1.2 Understanding the data

Existing datasets for clothes recognition have very varied sizes and annotations. The previous datasets were labeled with limited number of attributes, bounding boxes, or consumer-to-shop pair correspondences. DeepFashion contains 800K images, which are annotated with 50 categories, 1, 000 attributes, clothing landmarks (each image has 4 to 8 landmarks), and over 300K image pairs. Some other datasets in the vision community were dedicated to the tasks of clothes segmentation, parsing and fashion modeling, while DeepFashion focuses on clothes recognition and retrieval. It is the largest and most comprehensive clothes dataset to date.

```
print(data.classes)
print (len(data.classes),data.c)
>>> ['Blouse', 'Blazer', 'Button-Down', 'Bomber', 'Anorak', 'Tee', 'Tank', 'Top',
    'Sweater', 'Flannel', 'Hoodie', 'Cardigan', 'Jacket', 'Henley', 'Poncho',
    'Jersey', 'Turtleneck', 'Parka', 'Peacoat', 'Halter', 'Skirt', 'Shorts',
    'Jeans', 'Joggers', 'Sweatpants', 'Jeggings', 'Cutoffs', 'Sweatshorts',
    'Leggings', 'Culottes', 'Chinos', 'Trunks', 'Sarong', 'Gauchos', 'Jodhpurs',
    'Capris', 'Dress', 'Romper', 'Coat', 'Kimono', 'Jumpsuit', 'Robe', 'Caftan',
    'Kaftan', 'Coverup', 'Onesie']
(46, 46)
```

The data consists of 50 classes although the data is a bit skewed with three classes accounting for about 133,000 images which makes the problem even challenging. Clothing Category classification is considered a fine-grained image classification task, since, most of the clothing looks visually very similar and have high intra-class and low-inter class variance.

1.3 Existing work

The original paper published in 2016 built FashionNet[1], while combining it with different methodologies to run a set of experiments. WTBI[3,6] concatenated multi-layer perceptron (MLP) on top of the pre-trained ImageNet[8] models. They only implement the category-independent metric network of WTBI, which handles all clothing categories in a single network. DARN[7] adopted an attribute-regularized two-stream CNN. One stream handles shop images, while the other handles street images. Note that for category classification and attribute prediction, only one stream of DARN is used.

There have been numerous attempts at using this dataset to build several statistical models which solve different problems. The model proposed and tested as a part of this project is about 4.5% better than the best accuracy of the original paper.

Some other datasets in the vision community were dedicated to the tasks of clothes segmentation, parsing [11, 12, 13, 14, 15] and fashion modeling [16, 17], while DeepFashion focuses on clothes recognition and retrieval.

2 Methodology

We started by using a Resnet34 which to use supervised learning to classify images into the right categories. We initially assumed the accuracy metric to be Top1 accuracy and nothing else.

The results for the loss values of the model is as follows along with the plot of training and validation loss.

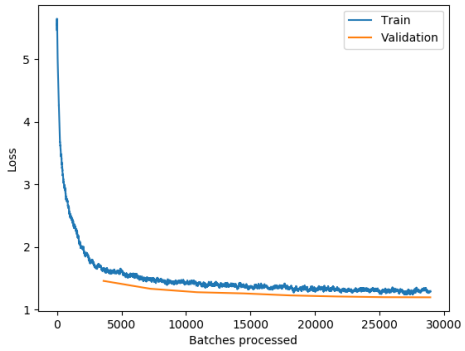

Figure 2: The plot of validation loss and training loss

Training results			
#Epoch	Training Loss	Validation Loss	Accuracy
1	1.647003	1.460895	0.568018
2	1.486313	1.335597	0.605134
3	1.410295	1.280500	0.621886
4	1.395943	1.261057	0.625949
5	1.349992	1.229344	0.634350
6	1.352072	1.212104	0.640833
7	1.297580	1.201270	0.643202
8	1.300530	1.199177	0.642718

Their Accuracy ranges for Category Classification are shown 82.58 for top-3 and 90.17 for top-5. Since we calculated the Top 1 accuracy to begin with, the Top 3 and Top 5 accuracy are as follows. The Top 3 and Top 5 assumed here are the usual definitions.

	Category		Texture		Fabric		Shape		Part		Style		All	
	top-3	top-5	top-3	top-5	top-3	top-5	top-3	top-5	top-3	top-5	top-3	top-5	top-3	top-5
WTBI [3]	43.73	66.26	24.21	32.65	25.38	36.06	23.39	31.26	26.31	33.24	49.85	58.68	27.46	35.37
DARN [10]	59.48	79.58	36.15	48.15	36.64	48.52	35.89	46.93	39.17	50.14	66.11	71.36	42.35	51.95
FashionNet+100	47.38	70.57	28.05	36.59	29.12	40.58	28.51	36.51	31.65	38.53	53.92	62.47	31.58	39.06
FashionNet+500	57.44	77.39	34.73	46.35	34.47	46.60	33.61	44.57	38.48	49.01	63.48	67.94	38.94	49.71
FashionNet+Joints [34]	72.30	81.52	35.92	48.73	38.21	49.04	37.59	47.73	40.21	51.81	64.91	73.14	43.14	52.33
FashionNet+Poselets [34]	75.34	84.87	36.85	49.11	38.88	49.48	38.19	47.09	41.60	52.85	64.84	73.03	43.57	52.65
FashionNet (Ours)	**82.58**	**90.17**	**37.46**	**49.52**	**39.30**	**49.84**	**39.47**	**48.59**	**44.13**	**54.02**	**66.43**	**73.16**	**45.52**	**54.61**

Table 2. Performance of category classification and attribute prediction.

Figure 3: Results from the original research paper

Top 3 and Top 5 accuracies as follows:

Type	Acc
Top 3	.8513
Top 5	.9201

The results we have obtained so far are already better than the original paper but it was a few years ago. The best way to understand what went wrong would be to visualize the confusion matrix of the results. For example, the model is the most confused between a Rompter and Dress accounting for 663 instances whereas it is confused between A 'Top' and a 'Blouse', and a 'Tee' and a 'Blouse' close to 1100 instances. One example where the model is very confused is the following.

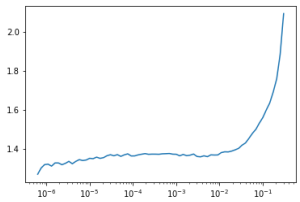

Figure 4: Plot of the learning rate

Figure 5: An example of a image which could be classified into two categories

Taking top-k accuracy does make sense, Instead of asking the model to find only 1 item, let's ask the model to find top 3 items or top 5 items from the image. It will surely help on images which have multiple items together.

Since our model is working as we expect it to, we will unfreeze our model and train some more. From Figure 4, the optimal learning rate range where the losses were low, is from 1e-6 to 1e-4, We used it to train an unfreeze learner.

Training results			
#Epoch	Training Loss	Validation Loss	Accuracy
1	1.300257	1.158340	0.653592
2	1.253208	1.134286	0.661786
3	1.173708	1.099417	0.672573
4	1.161334	1.077408	0.679938
5	1.107785	1.06841	0.682548
6	1.092696	1.059060	0.686317
7	1.110213	1.054054	0.687423
8	1.077311	1.053436	0.687527

Top 3 and Top 5 accuracies as follows:

Type	Acc
Top 3	.8942
Top 5	.9461

Our Unfreeze models are doing better than our quick trained model. Finding Optimal learning rate and unfreezing resent34 architecture surely works better than most of the models and the top-5 accuracy is 2.5% more than the original paper and the top-3 accuracy 3.4% more than the original paper

3 Experiments

3.1 Data Filtering

Data in machine learning is considered as the new oil, and different methods are utilized to collect, store and analyze the ML data. However, this data needs to be refined before it can be used further. One of the biggest challenges when it comes to utilizing Machine Learning data is Data Cleaning.

Although data cleaning may not be mentioned too often, it is very critical for the success of Machine Learning applications. The algorithms that you may use can be powerful, but without the relevant or right data training, your system may fail to yield ideal results. We experimented with the data in several ways to help with our problem statement.

- The fashion data set is not evenly distributed over 50 classes. We see that there are certain classes have as few images as 0 for certain classes and over 8000 images for another. Therefore, we broke down the classes to 14 classes in total, each having atleast 1000 images.
- The fashion dataset has varied images. Some have humans wearing clothes, some with clothing, some with humans and white background, and some with complex background. Therefore, we first BBox[6] the images as given by the dataset to get the focus region. This will help with the unsupervised classification and training as Bounding Boxes will remove the extra information and the clothers.

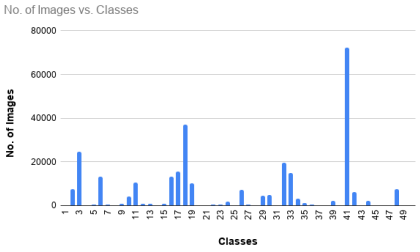

Figure 6: Before filtering Data

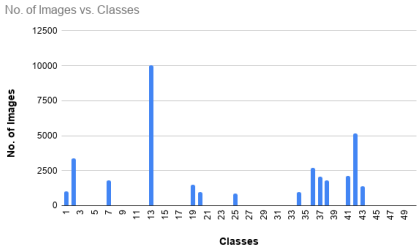

Figure 7: After Filtering Data

3.2 AutoEncoders

Autoencoder are generally used to find hidden correlations between the images so that they can be deconstructed and then constructed back to the original image. For the problem of unsupervised classification, we use Kmeans. However, the complete image is too large an input for Kmeans to cluster together efficiently. Therefore, we use Autoencoders. Autoencoders provide an excellent way to find the hidden correlations in the image. We use a an autoencoder for lossy image compression and then use the learned feature map of the image to classify the images using Kmeans. We modelled a novel autoencoder using 6 fully connected CNNs layers for the encoder and 5 fully connected layers for the decoder.
We tried fine-tuning our autoencoders in several ways :

- Fine tuning the output layers of the Encoder We tried to reduce the dimensions of the image to as low as possible. However, as expected, we lost a lot of features as we lowered the dimensions. We found the optimal dimension that we could construct the image with to be

16x16.

- We further increased the number of feature channels while reducing the dimensions of the image. This helped use further lower the dimensions as we now stored the important feature information in the feature channels and loosing comparatively lesser amount of data. Our model currently brings the image to 32 input channels which are then scaled down back to 3 channels as given in the input. $Figure\ 8$ shows the encoder architecture.

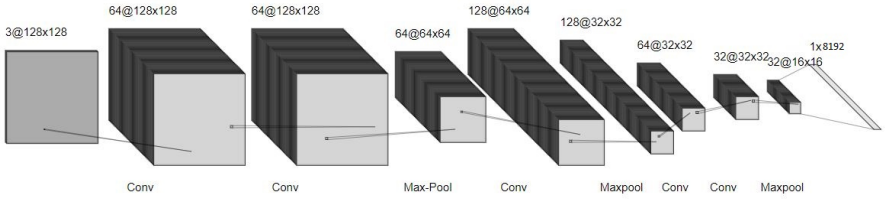

Figure 8: Encoder Architecture

Below are examples of few that were compressed and reconstructed. We see that a fair amount of detail, which could easily identify the piece of clothing. We therefore store effective information in a lower dimension space to help Kmeans classify the data better. We resize our image to a size of

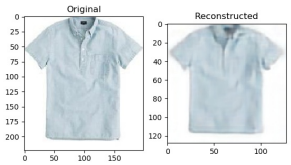

Figure 9: Blue t-shirt

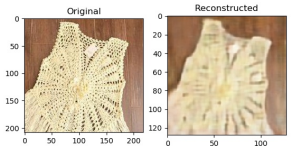

Figure 10: White dress

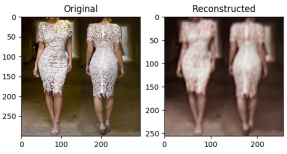

Figure 11: Full white dress

Figure 12: Blue Jacket

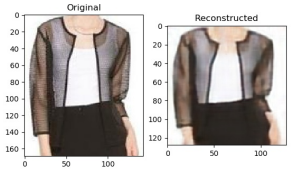

Figure 13: Net Jacket

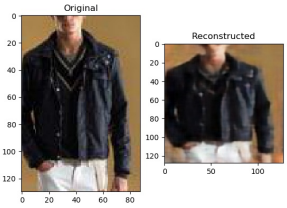

Figure 14: Black Jacket

$128x128x3$ before processing it. Our auto encoder scaled down the image to $16x16x32$ and then upscales the image back to $128x128x3$.

3.3 Kmeans and PCA

3.3.1 PCA

Reducing the dimensions were severely affecting the loss and was strongly affecting the reconstruction and classification. We therefore, further added a layer of PCA get get rid of further unwanted values to provide to kmeans for better classification.
We used PCA provided by the $sklearn$ package. We tried several values changing the $n\_components$ value. This therefore, further helped us bring down the dimensions by only keeping what was necessary. PCA helped to bring the dimension down to $32 * 4$ maintaining a cumulative co-variance of 0.91.

3.3.2 Kmeans

KMeans is a clustering algorithm which divides observations into k clusters where k is the number of classes, which is 50 for the deep fashion dataset. We utilised the inbuilt kmeans provided by the $sklearn$ package. We took the feature map extracted from the encoder output and passed it to KMeans to classify the images. Figure x shows result output of cluster of t-shirts together and Figure y shows the result output of the cluster of crop-tops together as filtered by the proposed unsupervised classification model.

Figure 15: T-shirt Cluster

Figure 16: Crop-top Cluster

4 Conclusion

We propose the two models and evaluated existing models for classification of DeepFashion dataset. The supervised model has the top 5 accuracy of 94% which is better than the existing state of the 92% and the paper which has the accuracy of 90%. The supervised approach used a ResNet34 using the unfreeze models. We then extended our domain to unsupervised learning. For this, we used an Autoencoder to compress the image to reduce the dimensions of the image while increasing the feature channels. The feature map obtained from the encoder was then filtered using PCA to removed unwanted features which was then classified using KMeans.

5 Challenges and Future Work

The project has helped us learn some of the very important concepts and also allow us to play around with some of the cutting-edge models out there. A few things we could still improve on would be to use the cropped images for supervised learning instead of just using the image to predict the category. This would not only address the categories which the model is most confused on, it would also help the model focus on the part which it needs to classify. One example of the same is an image where both the top and the skirt are visible(full-body image) and the model incorrectly predicted the category. One of the challenges we faced were around building an accuracy metric for the unsupervised classification. We tried a bunch of different metrics but wasn't satisfactory enough. It is challenging since there is no right answer for the accuracy metric at hand and they are vary very much from dataset to dataset.

References

[1] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016

[2] L. Bossard, M. Dantone, C. Leistner, C. Wengert, T. Quack, and L. Van Gool. Apparel classification with style. In ACCV, pages 321–335. 2012.

[3] Y. Kalantidis, L. Kennedy, and L.-J. Li. Getting the look: clothing recognition and segmentation for automatic product suggestions in everyday photos. In ICMR, pages 105–112, 2013.

[4] H. Chen, A. Gallagher, and B. Girod. Describing clothing by semantic attributes. In ECCV, pages 609–623. 2012.

[5] Q. Chen, J. Huang, R. Feris, L. M. Brown, J. Dong, and S. Yan. Deep domain adaptation for describing people based on fine-grained clothing attributes. In CVPR, pages 5315– 5324, 2015.

[6] M. H. Kiapour, X. Han, S. Lazebnik, A. C. Berg, and T. L. Berg. Where to buy it: Matching street clothing photos in online shops. In ICCV, 2015

[7] J. Huang, R. S. Feris, Q. Chen, and S. Yan. Cross-domain image retrieval with a dual attribute-aware ranking network. In ICCV, 2015.

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database

[9] S. Liu, Z. Song, G. Liu, C. Xu, H. Lu, and S. Yan. Street-toshop: Cross-scenario clothing retrieval via parts alignment and auxiliary set. In CVPR, pages 3330–3337, 2012

[10] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In CVPR, pages 1385–1392, 2011. Fig2

[11] K. Yamaguchi, M. H. Kiapour, L. E. Ortiz, and T. L. Berg. Parsing clothing in fashion photographs. In CVPR, pages 3570–3577, 2012.

[12] K. Yamaguchi, M. H. Kiapour, and T. Berg. Paper doll parsing: Retrieving similar styles to parse clothing items. In ICCV, pages 3519–3526, 2013

[13] E. Simo-Serra, S. Fidler, F. Moreno-Noguer, and R. Urtasun. A high performance crf model for clothes parsing. In ACCV, 2014.

[14] S. Liu, J. Feng, C. Domokos, H. Xu, J. Huang, Z. Hu, and S. Yan. Fashion parsing with weak color-category labels. TMM, 16(1):253–265, 2014.

[15] W. Yang, P. Luo, and L. Lin. Clothing co-parsing by joint image segmentation and labeling. In CVPR, pages 3182– 3189, 2014

[16] E. Simo-Serra, S. Fidler, F. Moreno-Noguer, and R. Urtasun. Neuroaesthetics in fashion: Modeling the perception of beauty. In CVPR, 2015.

[17] K. Yamaguchi, T. L. Berg, and L. E. Ortiz. Chic or social: Visual popularity analysis in online fashion networks. In ACM MM, pages 773–776, 2014.