CS676A Project
Report

Learning with Relative Attributes

Submitted in partial fulfillment of
the requirements for the course CS676A

April 15, 2016

Submitted by

13683 Shubham Jain
13788 Vikas Jain

Under the guidance of
Prof Vinay P. Namboodiri

Department of Computer Science and Engineering
Indian Institute of Technology Kanpur
Kanpur, U.P. , India – 208 016

July 2015

Contents

1 Introduction

Human nameable attributes are more intuitive to understand, e.g. open natural scene, white young face. Due to their simplicity, human nameable attributes are used for recognition tasks. But sometimes, it becomes difficult to assign categorical labels to the images i.e binary labels – yes or no if attribute is present or not respectively. Relative attributes[1] are more understandable and intuitive, e.g. a person in a image is taller than some other person in other image and shorter than an another person in another image. In this project, we aim to learn relative attributes associated with face images namely Masculine, white, young, chubby, visible-forehead, bushy-eyebrows, narrow-eyes, pointy-nose, big-lips and round-face. We used PubFig dataset[5] for the task. We extracted features using Convolutional Neural Network(VGG16 model)[4]. We implemented and trained Ranknet model to predict absolute ranking of attributes give the image features. We also explored zero shot learning for unseen classes by building probabilistic model of each seen and unseen class using attribute ranking for each image of seen class and relative description of unseen classes with seen classes respectively.

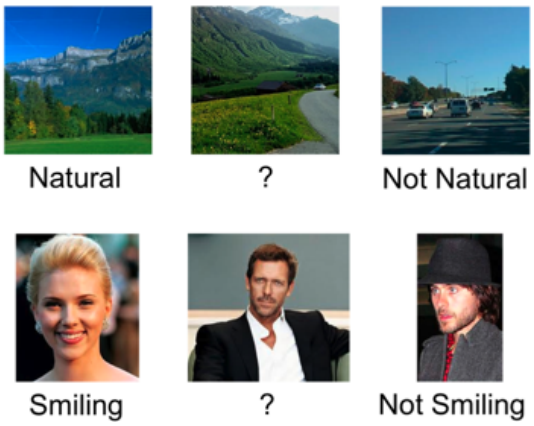


Figure 1: Relative Attribute [1]

2 Methodology

The main idea in a simplified version is as follows: Find a good feature representation of the image which somehow incorporates the strength of features not just their presence. Now using these features learn a model which when given a feature representation of a new image tries to predict a ranking for the image for that particular attribute. We will now describe the feature representation that we used and the ranking models.

2.1 Feature representation

Parikh et. al. [1] in their paper used GIST features of the images for the learning task. Instead of using handcrafted features, We used features extracted from Convolutional Neural Network. The VGG16 [4] model for extracting features from the images. We used learned model of VGG16 trained on ImageNet dataset[6].

The last layer(softmax) from the network is removed and the output of layer before it(FC-4096) is used as feature vector of each image. The dimension of each feature vector is 4096.

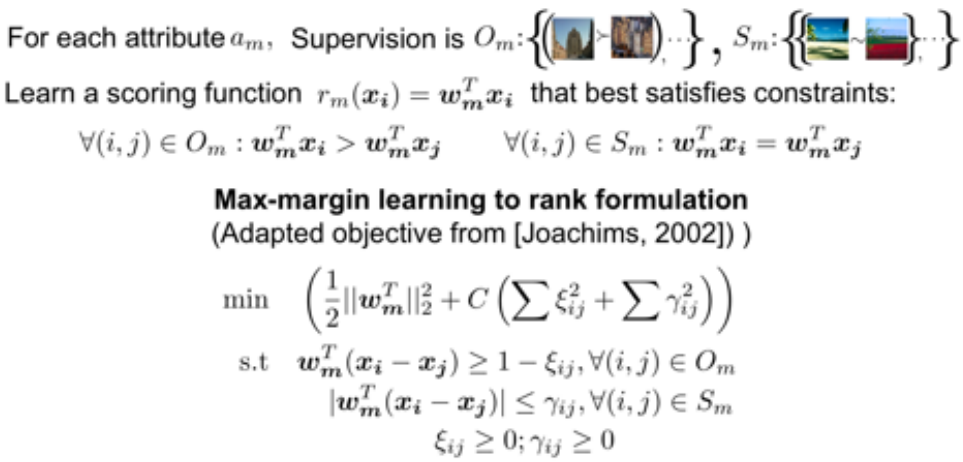
2.2 Ranking Methods

For ranking we have compared two methods. One is based on an approximation algorithm to solve the problem via Support Vector Machines (SVM). The latter is based on neural networks.

2.2.1 RankSVM

The idea was to learn given a query image and user preferences which are incorporated to retrieve more relevant images in the search results. It uses a set of comparisons between certain images with respect to an attribute. The exact equation is NP-Hard and an approximate solution is obtained by using negative

slack variables. The mathematical description is as follows:



Rank SVM formalization[1]

2.2.2 RankNet [2]

It is a Neural network with two hidden layers and back-propagation can be extended to train on these pairs. The first hidden layer had 2048 neurons and the second hidden layer had 512 neurons. The model trains on pairs of examples to learn a ranking function that maps to the real numbers. It uses a natural probabilistic cost function on pairs of examples. The network as described in the paper is as follows :

For the ith training sample, denote the outputs of net by o_i, the targets by t_i, let the transfer function of each node in the jth layer of nodes be g_j, and let the cost function be $\sum_{i=1}^{q} f(o_i, t_i)$ If α_k are the parameters of the model, then a gradient descent step amounts to $\partial\alpha_k$ = -$\eta_k \frac{\partial f}{\partial \alpha_k}$, where the η_k are positive learning rates. The net embodies the function

$$o_i = g^3(\sum_j w_{ij}^{32} g^2(\sum_k w_{jk}^{21} x_k + b_j^2) + b_i^3) \equiv g_i^3 \tag{1}$$

where for the weights w and offsets b, the upper indices index the node layer, and the lower indices index the nodes within each corresponding layer. The cost function becomes a function of the

difference of the outputs of two consecutive training samples: f ($o_2 - o_1$). The update equations (back propagation) are as follows (f' $\equiv$ f' ($o_2 - o_1$)):

$$\frac{\partial f}{\partial b^3} = f'({g'}_2^3 - {g'}_1^3) \equiv \triangle_2^3 - \triangle_1^3 \tag{2}$$

$$\frac{\partial f}{\partial w_m^{32}} = \triangle_2^3 g_{2m}^2 - \triangle_1^3 g_{1m}^2 \tag{3}$$

$$\frac{\partial f}{\partial b_m^2} = \triangle_2^3 w_m^{32} {g'}_{2m}^2 - \triangle_1^3 w_m^{32} {g'}_{1m}^2 \tag{4}$$

$$\frac{\partial f}{\partial w_m^{21}} = \triangle_{2m}^2 g_{1n}^1 - \triangle_{1m}^2 g_{1m}^2 \tag{5}$$

2.3 Zero Shot Learning

Once the ranking of each attribute of the classes are learnt. The feature vector of each image of each class is represented in the attribute space. Using the features vector of each image of a particular class, the parameters of normal distribution is found to obtain a probabilistic model of the class.

The probabilistic model for the unseen class is found using the relative description of unseen class with the seen classes. The method to find the Normal Distribution parameters is same as given in [1]

Once the probabilistic model of seen and unseen classes are obtained, for a query image, fist the rank of attributes for the images is found and we obtain a vector $\bar{x}$ in the attribute space. The image is labelled with the class with maximum probability among all seen and unseen classes.

$$c^* = \max_{j \in 1,\ldots,N} P(\bar{x}_i | \mu_i, \Sigma_i) \tag{6}$$

3 Dataset

We used PubFig[5] Dataset for this project. The dataset consist of face images of celebrity. The images from 8 classes were taken for the learning tasks. Around 100 images from each class were used, out of which 85 were used for training and rest for testing.

For zero shot learning, we took images from 3 different classes. Around 30 images of each unseen class were used for prediction the class.

4 Code Used

- **Feature Extraction** - From [4] `https://gist.github.com/baraldilorenzo/07d7802847aaad0a35d3`
- **RankNet** - Adapted from `https://github.com/shiba24/learning2rank`
- **RankNet** - From [1]

5 Results

5.1 Correctly ordered pairs:

The result obtained for the learned relative ranking per attribute in the 8 classes are shown in Table 1.

Attribute	**Acc.**	**Attribute**	**Acc.**
Masculine	80.35	Bushy-eyebrows	80.81
Young	79.14	Pointy-nose	79.25
Chubby	69.68	Big-lips	88.25
Forehead	66.48	Round-face	83.66

Table 1: Accuracy of learned attributes

5.2 Class Prediction:

The results of predicted class by formulating Normal Distribution of each class are shown in Table 2.

Class	**RankNet Acc.**	**RankSVM Acc.**
1	0.80	0.67
2	0.91	0.36
3	1.00	0.75
4	0.89	0.56
5	0.75	0.67
6	0.83	0.58
7	0.85	0.77
8	0.92	0.85
9(Unseen)	0.03	–
10(Unseen)	0.00	–
11(Unseen)	0.10	–

Table 2:

6 Conclusions

It can be inferred that feature vectors extracted from Convolutional Neural Network and RankNet model used for learning relative attributes are performing better than the techniques used in [1] of GIST features and RankSVM model.

However, for zero shot learning task, while the prediction of seen classes are better for RankNet as compared to RankSVM, it performs significantly poor in case of unseen classes.

7 Scope and Future Work

For zero shot we can instead do unsupervised step like clustering instead of using the comparison of attributes but that is one thing that the paper wanted to incorporate so that it kind of

comes out in a natural way. Also, we need to find a better way to fins out the parameters of the unseen classes as the one that the paper used we found not to work so well.

References

[1] Parikh, Devi, and Kristen Grauman. "Relative attributes." Computer Vision (ICCV), 2011 IEEE International Conference on. IEEE, 2011.

[2] Burges, Chris, et al. "Learning to rank using gradient descent." Proceedings of the 22nd international conference on Machine learning. ACM, 2005.

[3] Joachims, Thorsten. "Optimizing search engines using clickthrough data." Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2002.

[4] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).

[5] "Attribute and Simile Classifiers for Face Verification," Neeraj Kumar, Alexander C. Berg, Peter N. Belhumeur, and Shree K. Nayar, International Conference on Computer Vision (ICCV), 2009.

[6] Deng, Jia, et al. "Imagenet: A large-scale hierarchical image database." Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, 2009.