华南理工大学

South China University of Technology

The Experiment Report of Machine Learning

SCHOOL: SCHOOL OF SOFTWARE ENGINEERING

SUBJECT: SOFTWARE ENGINEERING

Author:
Jin Cao
Xianlin Hu
PinChaoHuang

Student ID：
201530611111
201530781104
201530611760

Supervisor:
Mingkui Tan

Grade:
Undergraduate

December 21, 2017

Face Classification based on AdaBoost Algorithm

Abstract—In this experiment, we use Adaboost algorithm and decision tree to solve the face classification problem, which determine whether a picture contains a human face or not.

I. INTRODUCTION

In the field of machine learning, face recognition is an important issue. In this experiment, we use Adaboost algorithm with decision tree as the weak classifier, and try to realize the goal of face classification. In this experiment, we expect the classification accuracy to reach 0.9.

II. METHODS AND THEORY

Adaboost is an iterative algorithm. Its core idea is to train different classifiers (weak classifiers) on the same training set, and then combine these weak classifiers to form a stronger final classifier (strong classifier). The algorithm itself is achieved by changing the data distribution, which determines the weight of each sample based on the correct classification of each sample in each training set and the accuracy of the last overall classification. The new data set with modified weights is sent to the lower classifier for training. Finally, the classifier obtained by each training is finally fused together to be the final decision classifier.

Input: $D = \{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n)\}$, **where** $\mathbf{x}_i \in X, y_i \in \{-1, 1\}$
Initialize: Sample distribution w_m
Base learner: $\mathcal{L}$
$w_1(i) = \frac{1}{n}$
for m=1,2,...,M **do**
  $h_m(x) = \mathcal{L}(D, w_m)$
  $\epsilon_m = \sum_{i=1}^{n} w_m(i)\mathbb{I}(h_m(\mathbf{x}_i) \neq y_i)$
  if $\epsilon_m > 0.5$ **then**
    break
  end
  $\alpha_m = \frac{1}{2}\log\frac{1-\epsilon_m}{\epsilon_m}$
  $w_{m+1}(i) = \frac{w_m(i)}{z_m} e^{-\alpha_m y_i h_m(\mathbf{x}_i)}$ **,where** $i = 1, 2, ..., n$ **and**
  $z_m = \sum_{i=1}^{n} w_m(i) e^{-\alpha_m y_i h_m(\mathbf{x}_i)}$
end
Output: $H(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m h_m(\mathbf{x})$

Fig1. The implementation of Adaboost

III. EXPERIMENT

A. Dataset

We use 1000 pictures, of which 500 are human face RGB images, the other 500 is a non-face RGB images.

B. Implementation

1.Read data set data. The images are supposed to converted into a size of 24 * 24 grayscale, the number and the proportion of the positive and negative samples is not limited, the data set label is not limited.

2.Processing data set data to extract NPD features. Extract features using the NPDFeature class in feature.py. (Tip: Because the time of the pretreatment is relatively long, it can be pretreated with pickle function library dump () save the data in the cache, then may be used load () function reads the characteristic data from cache.)

3.The data set is divisded into training set and calidation set, this experiment does not divide the test set.

4.Write all AdaboostClassifier functions based on the reserved interface in ensemble.py. The following is the guide of fit function in the AdaboostClassifier class:

4.1 Initialize training set weights , each training sample is given the same weight.

4.2Training a base classifier , which can be sklearn.tree library DecisionTreeClassifier (note that the training time you need to pass the weight as a parameter).

4.3 Calculate the classification error rate of the base classifier on the training set.

4.4 Calculate the parameter according to the classification error rate .

4.5 Update training set weights .

4.6 Repeat steps 4.2-4.6 above for iteration, the number of iterations is based on the number of classifiers.

5.Predict and verify the accuracy on the validation set using the method in AdaboostClassifier and use classification_report () of the sklearn.metrics library function writes predicted result to report.txt .

Number of base learner	8
Max_depth of decision tree	3

Table1. Parameters

	precision	recall	f1-score	support
face	0.94	0.97	0.96	150
nonface	0.94	0.94	0.96	150
Avg/total	0.96	0.96	0.96	300

Table2.Result

IV. CONCLUSION

In this experiment, we learn to use Adaboost algorithm with decision tree to solve the face classification problem. Through adjusting the parameters, including the max depth of decision tree and the number of base learner, we get a higher classification accuracy, and 0.96 is the highest accuracy.