Available online at www.sciencedirect.com

SciVerse ScienceDirect

Physics Procedia 25 (2012) 800 – 807

Physics
Procedia

2012 International Conference on Solid State Devices and Materials Science

AdaBoost for Feature Selection, Classification and Its Relation with SVM*, A Review

Ruihu Wang

Department of Science and Technology Chongqing University of Arts and Sciences
Yongchuan, Chongqing 402160, CHINA

Abstract

In order to clarify the role of AdaBoost algorithm for feature selection, classifier learning and its relation with SVM, this paper provided a brief introduction to the AdaBoost which is used for producing a strong classifier out of weak learners firstly. The original adaptive boosting algorithm and its application in face detection and facial expression recognition are reviewed. In pattern classification domain, support vector machine has been widely used and shows promising performance. However, it is expensive in terms of time-consuming. A sort of cascaded support vector machines architecture is capable of improving the classification accuracy based on AdaBoost boosting algorithm, namely, AdaboostSVM. It applied boosting algorithm to feature selection and classifier learning for support vector machine classification and it has achieved approved performance through some researcher's pioneering work.

Keywords: AdaBoost, Feature selection, Cascaded, Support vector machine.

1. Introduction

The AdaBoost (adaptive boosting) algorithm was proposed by Yoav Freund and Robert Shapire in 1995 for generating a strong classifier from a set of weak classifiers [1,3]. In [2], Y.Freund and R.Schapire illustrated an interesting example, horse-racing gambler, to explain the idea about optimization and solution space search behind. Naturally, the gambler would ask some highly successful

* This work is supported by the Science and Technology Foundations of Chongqing Municipal Education Commission Grant #KJ091216 to Ruihu. Wang and Excellent Science and Technology Program for Overseas Studying Talents of Chongqing Municipal Human Resources and Social Security Bureau Grant #00958023 to Ruihu. Wang, and Key project of Science and Research Foundation of CQWU Grant #Z2009js07 to Ruihu Wang.

doi:10.1016/j.phpro.2012.03.160

expert gamblers for their help before he made his decision on which horse he should bet. Each expert would give him some good suggestions based on his own experience. In terms of pattern classification, these suggestions formed a large pool of classifiers, although they were obviously very rough and inaccurate. The point was that could the individuals' experience be integrated to build up a better classifier for the gambler's betting? From then on, this issue attracted lots of researchers' attention to seek valuable strategies to deal with. Kearns and Valiant [4,5] were the first to pose this question of whether some weak learning algorithm which runs just a little better than random guessing in the PAC model can be "boosted" into an accurate strong learning algorithm.

If we regard each expert's suggestion as a training sample for classifier learning, for a given input pattern $\boldsymbol{x}_i$, each expert classifier $\boldsymbol{k}_j$ can express his opinion, denoted by $\boldsymbol{k}_j$ ($\boldsymbol{x}_i$). Assuming the problem of separating the set of training vectors belonging to two classes , $\boldsymbol{k}_j$ ($\boldsymbol{x}_i$) takes two values only, +1 or -1 respectively, i.e. $\boldsymbol{k}_j (\boldsymbol{x}_i) \in \{-1,+1\}$. The final decision of the committee *K* of experts is made by *sign* $\boldsymbol{C}(\boldsymbol{x}_i)$, the sign of the linear combination of the weighted sum of expert opinions, where

$$C(x_i) = \alpha_1 k_1(x_i) + \alpha_2 k_2(x_i) + \cdots + \alpha_l k_l(x_i) \, , \tag{1}$$

and $k_1, k_2, \cdots, k_l$ denote the l experts. $\alpha_1, \alpha_2, \cdots, \alpha_l$ are the weights the gambler assign to the opinion of each expert in the committee [6]. This idea of combining weak classifier to form a expective strong decision function contributed the emeging of AdaBoost boosting algorithm.

2. Adaboost Algorithm

AdaBoost algorithm creates a set of poor learners by maintaining a collection of weights over training data and adjusts them after each weak learning cycle adaptively. The weights of the training samples which are misclassified by current weak learner will be increased while the weights of the samples which are correctly classified will be decreased [7]. The original AdaBoost algorithm is described in Fig 1.

2.1AdaBoost Algorithm

One of the main ideas of AdaBoost algorithm is to maintain a distribution or set of weights over the training set.The weight of this distribution on training example i on round *t* is denoted $D_t(i)$. Initially, all weights are initialized equally, but on each round, the weights of incorrectly classified examples are increased so that the weak learner is forced to focus on the hard examples in the trading set. The weak learner's job is to find a weak hypothesis $h_t : X \to \{-1,+1\}$ appropriate for the distribution D_t [3].

Given: $(x_1, y_1), \cdots, (x_n, y_n)$ where $x_i \in X, y_i \in Y = \{-1, +1)$,

Algorithm: Adaptive Boosting (AdaBoost)

```
Initialize $D_1(i) = \frac{1}{m}$.
For t = 1,···, T:
    Train weak learner using distribution $D_t$.
    Get weak hypothesis $h_t : X \to \{-1,+1\}$ with error
        $\varepsilon_t = \Pr_{i \sim D_t}[h_t(x_i) \neq y_i]$.
    Choose $\alpha_t = \frac{1}{2}\ln\left(\frac{1-\varepsilon_t}{\varepsilon_t}\right)$.
```

Update

$$D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \begin{cases} e^{-\alpha_t} & if\ h_t(x_i) = y_i \\ e^{\alpha_t} & if\ h_t(x_i) \neq y_i \end{cases}$$

$$= \frac{D_t(i) e^{-\alpha_t y_i h_t(x_i)}}{Z_t}$$

where Z_t is a normalization factor.

Output the final hypothesis:

$$H(x) = sign\left(\sum_{t=1}^{T} \alpha_t h_t(x) \right).$$

Fig. 1 The boosting algorithm AdaBoost.

2.2AdaBoost For Feature Extraction and Selection

AdaBoost is one of the most promising, fast convergence, and easy to be implemented machine learning algorithm. It requires no prior knowledge about the weak learner and can be easily combined with other method to find weak hypothesis, such as support vector machine. In Table 1, two main biometric recognition applications: Face Detection and Facial Expression Recognition are presented which associated with AdaBoost algorithm for feature extraction, feature selection and classifier learning.

Table 1 adaboost for face detection and facial expression recognition

Application	Feature Extraction	Feature Selection and Classifier Learning
Face Detection	Rectangle Feature[19], PCA[21,25],LDA[26] Haar-like Rectangle Feature[23] Gabor Wavelet[24]	AdaBoost[19,21,23]
Facial Expression Recognition	LBP[17], Gabor Filter[22,24], Rectangle Feature[18], PCA[20,22]	PCA, AdaBoost[17,18,19,20,22] BP Neutral Network[20]

3. Feature Extraction

Any pattern classification and recognition problem can be regarded as machine learning and intelligent human computer interaction ultimately. The goal of machine intelligence system is to learn a classification function from a given feature set and a training set of positive and negative samples. In its original form, described in Fig 1, the AdaBoost learning algorithm is used to boost the classification performance out of some weak learners. This rough feature set has to be selected and refined before being submitted to classifier learning. Feature selection is an optimization process to reduce a large set of original rough features to a relatively smaller feature subset which containing only significant to improve the classification accuracy fast and effectively.

3.1Feature Dimensionality Curse

An automated object recognition must solve two basic problems: feature extraction and classifier design. Among different kinds of feature extraction methods, Gabor filter is a quite useful tool to computer vision and image analysis because it has optimal localization properties in both spatial analysis and frequency domain. The application of Gabor wavelet for face recognition was put forward by Lades et al. since Dynamic Link Architecture (DLA) was proposed in 1993 [27]. It has been found that Gabor wavelet-based features are relatively robust to illumination changes and head movement due to multiple resolution and multiple orientation filtering [28]. Unfortunately, it requires expensive computational costs to implement.

A 2D Gabor filter $\phi(k,x)$ is defined as a Gaussian low-pass filter modulated by a plane wave.

$$\phi(k,x)=\frac{|k|^2}{\sigma^2}\exp\left(-\frac{|k|^2|x|^2}{2\sigma^2}\right)\left(\exp\left(ik^Tx\right)-\exp\left(-\frac{\sigma^2}{2}\right)\right) \tag{2}$$

where ***x*** represents the spatial localization and the wave vector $\boldsymbol{k}=\left(k\cos\theta,k\sin\theta\right)^T$ represents the translation and orientation of the tuned filter in the frequency domain [29]. The Gabor wavelet outputs are generated by convolving the region of interest images with the bank of 40 Gabor filters, 5 frequencies and six orientations. For facial expression database JAFFE, in which the face image is sized 256×256, there are over 2,600,000 features. In [28], the filters in their approach were modulated to three frequencies $k\in\{\pi/4,\pi/8,\pi/16\}$ and six orientations $\theta\in\{\pi/6,\pi/3,\pi/2,2\pi/3,5\pi/6,\pi\}$. After Gabor filtering, altogether 18 filter outputs per region of interest. With their normalized image sizes of 150×100 pixels for the eye region, the filtering operation produces 270,000 features per image. They used PCA to reduce this large number of features. In [30,31], AdaBoost is also used for facial expression feature extraction and selection.

3.2AdaBoost for Feature Selection

Algorithm :AdaBoost for Feature Selection

```
Given example images: $(x_1,y_1),\cdots,(x_n,y_n)$ where $x_i\in X, y_i\in Y=\{0,1)$ for negative and positive examples respectively
Initialize weights $w_{1,i}=\frac{1}{2m},\frac{1}{2l}$ for $y_i=0,1$ respectively, where m and l are the number of negatives and positives examples respectively.
For t = 1,··, T:
    Normalize the weights,
        $w_{t,i}=\frac{w_{t,i}}{\sum_{j=1}^{n}w_{t,j}}$.
    For each feature, j, train a classifier $h_j$ which is restricted to using a single feature. The error is evaluated with respect to $w_t$, $\varepsilon_t=\sum_i w_i\left|h_j(x_i)-y_i\right|$
    Choose the classifier $h_t$, with the lowest error $\varepsilon_t$
    Update the weights:
        $w_{t+1,i}=w_{t,i}\beta_t^{1-e_i}$
```

where $e_i = 0$ if example x_i is classified correctly, $e_i = 1$ otherwise, and

$$\beta_t = \frac{\varepsilon_t}{1-\varepsilon_t}.$$

The final strong classifier is:

$$h(x) = \begin{cases} 1 & \sum_{t=1}^{T} \alpha_t h_t(x) \ge \frac{1}{2}\sum_{t=1}^{T} \alpha_t \\ 0 & otherwise \end{cases}$$

where $\alpha_t = \log\frac{1}{\beta_t}$.

Fig. 2 The AdaBoost algorithm for classifier learning

4. Support Vector Machine

Support vector machine was developed by Vapnik from the theory of Structural Risk Minimization. However, the classification performance of the practically implemented is often far from the theoretically expected. In order to improve the the classification performance of the real SVM, some researchers attempt to employ ensemble methods, such as conventional Bagging and AdaBoost [14]. However, in [15,16], AdaBoost algorithm are not always expected to improve the performance of SVMs, and even they worsen the performance particularly. This fact is SVM is essentailly a stable and strong classifier.

Considering the problem of classifying a set of training vectors belonging to two separate classes,

$$T = \{(x_1, y_1), \cdots, (x_l, y_l)\} \subseteq X \times Y \tag{3}$$

where

$$x_i \in X \subset R^n, y_i \in Y = \{-1, +1\}, i = 1, 2, \cdots, l.$$

SVM can be trained by solving the following optimization problem:

$$\min_w \Phi(w, \xi) = \frac{1}{2}\|w\|^2 + C\sum_i \xi_i \tag{4}$$

$$\text{subject to } y_i(\langle w, \phi(x_i)\rangle + b) \ge 1 - \xi_i, i = 1, 2, \cdots l \tag{5}$$

where $\xi_i > 0$ is the i-th slack variable and *C* is the regularization parameter.

The above optimization problem can be solved in its dual form:

$$\alpha^* = \arg\max_\alpha \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l} \alpha_i \alpha_j y_i y_j K\langle x_i, x_j\rangle - \sum_{k=1}^{l} \alpha_k \tag{6}$$

where $K\langle x_i, x_j\rangle$ is the kernel function performing the nonlinear mapping into feature space. The most frequently used kernel are Radius Basis Functions (RBF):

$$k(x_i, x_j) = e^{-\frac{\|x_i - x_j\|^2}{2\sigma^2}} \tag{7}$$

There are two parameters in SVM-RBF, i.e. the regularization parameter C, and the Gaussian width σ .

According to Yang and Honvar [8], the choice of feature used to represent patterns that are presented to a classifier has great impact on several pattern recognition properties, including the accuracy of the learned classification algorithm, the time need for learning a classification function, and the number of examples needed for learning, the cost associated with the features. In addition to feature selection, C.Huang and C. Wang [9] suggested that proper parameters setting can also improve the SVM classification accuracy. The parameters include penalty parameter *C* and the kernel function parameter σ for RBF, which should be optimized before training. In order to achieve optimal feature subset selection and SVM-RBF parameters, Hsu and Lin [10] proposed a Grid algorithm to find the best *C* and *sigma* for RBF kernel. However their method has expensive computational complexity and does not perform well. Genetic algorithm is an another alternative tool, which has the potential to generate both the optimal feature subset and SVM parameters at the same time. Huang and Wang [9] conducted some experiments on UCI database using GA-based approach. Their result has better accuracy performance with fewer features than grid algorithm. Compared to Genetic Algorithm, Particle Swarm Optimization has no evolution operators such as crossover and mutation. There are few parameters to adjust. It works well in a wide variety of applications with slight variations [11].

5. AdaBoostSVM

The classification performance of Support Vector Machine is affected by its parameters. For SVM-RBF, the parameters are Gaussian width σ and regularization parameter *C*. SVM-RBF classifier's performance largely rely on the σ value if a roughly suitable *C* is choosen [12]. For a given *C*, the performance of SVM-RBF can be changed by simply adjusting the value of σ . Increasing the value often reduces the complexity of learner model, and lowering the classification performance and vice versus So when the SVM-RBF is used as weak classifier for AdaBoost, a relatively large σ value is preferred, which brings a SVMRBF with relatively weak learn ability [13].

Algorithm: Varation of σ AdaBoost SVM-RBF

```
Given: a set of training samples labeled
    $T = (x_1, y_1), \cdots, (x_n, y_n)$ ,
    where $x_i \in X, y_i \in Y = \{-1, +1)$
Initialize the weight value of training samples: $w_1(i) = \frac{1}{n}, n = 1, 2, \cdots, n$ ·
For t = 1,··, T:
    Using SVM-RBF to train weak learner $C_t$ on the weighted training
    sample set and select training sample subset $d_t$ of $C_t$ , $d_t \subseteq T$ .
    Calculating the standard variation $\sigma$
    Using $d_t$ and $\sigma$ to the trained weak learner $C_t$ to get $h_t$
    Calculating training error of $C_t$ :
        $\varepsilon_t = \sum_{i=1}^{n} w_t(i), y_i \neq h_t(x_i)$
    Set weight of the weak classifier: $C_t$ : $\alpha_t = \frac{1}{2} \ln\left(\frac{1-\varepsilon_t}{\varepsilon_t}\right)$
    Update weight value of training samples:
```

$$w_{t+1}(i) = \frac{w_t(i)e^{-\alpha_t y_i h_t(x_i)}}{Z_t} = \frac{w_t(i)}{Z_t} \times \begin{cases} e^{-\alpha_t} & y_i = h_t(x_i) \\ e^{\alpha_t} & y_i \neq h_t(x_i) \end{cases}$$

where Z_t is a normalization factor, and $\sum_{i=1}^{n} w_i^{t+1} = 1$

Output the final hypothesis:

$$H(x) = sign\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right).$$

Fig 3. Varation of σ AdaBoost SVM-RBF Algorithm

6. Conclusion and Future Work

In this paper we first reviewed some fundamental background knowledge about AdaBoost Algorithm briefly. As a productive machine learning method, AdaBoost has been widely used in many kinds of real application. Not only feature extraction but also feature selection, AdaBoost shows promising and satisfied performance. Other than AdaBoost, there is another effective and easily to be implemented optimization algorithm is Particle Swarm Optimization (PSO). We will conduct experiments on this two kinds of optimization methods for facial expression recognition to provide a real time, fast, spontaneous emotional human computer interaction in intelligent biometric surveillance system.

References

[1] Y.Freund, R.Shapire. A decision-theoretic generalization of on-line learning and application to boosting. Proceedings of the Second European Conference on Computaional Learning Theory, 1995:23-27.

[2] .Freund, R.Shapire. A decision-theoretic generalization of on-line learning and application to boosting. Journal of Computer and System Sciences, 1997(55):119-139.

[3] Y.Freund, R.Shapire. A Short Introduction to Boosting. Journal of Japanese Society for Artificial Intelligence, 14(5):771-780, 1999.

[4] Michael Kearns, Leslie G.Valiant. Learning Boolean formulae or finite automata is as hard as factoring. Technical Report TR-14-88, Harvard University Aiken Computation Laboratory, 1988.

[5] Michael Kearns, Leslie G.Valiant. Cryptographic limitation on learning Boolean formulae and finite automata, Journal of the Association for Computing Machinery, 41(1):67-95,1994.

[6] R. Rojias, AdaBoost and the Super Bowl of Classifiers A Tutorial Introduction to Adaptive Boostring, Technical Report, 2009.

[7] X.Li,L.Wang,E,Sung. A Study of AdaBoost with SVM Based Weak Learners, Proceedings of International Joint Conference on Neural Network, 2005.

[8] J. Yang, V. Honvar. Feature subset selection using a genetic algorithm. IEEE Intelligent System and their Application, 13(2),44-49,1998.

[9] Cheng Lung Huang, Chieh Jen Wang. A GA-based feature selection and parameters optimization for support vector machines. Expert Systems with Applications. 31(2006) 231-240.

[10] Hsu,C.W., Chang,C.C., Lin,C.J.(2003). A pratical guide to support vector classification. Available at: http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf

[11] R.Wang., Z.Hu, L.Chen, J.Xiong. An Approach on Feature Selection and Parameters Optimization of Cascaded SVM with Particle Swarm Optimization Algorithm, accepted by the 3rd International Workshop on Computer Science and Engineering (WCSE), 2010

[12] G.Valentini, T.Dietterrich. Bias-variance analysis of support vector machines for the development of svm-based ensemble methods. Journal of Machine Learning Research. 2004(5):725-775.

[13] X.Li, L.Wang, E.Sung. A Study of AdaBoost with SVM Based Weak Learners. Proceedings of International Conference on Neural Networks, 2005:196-200.

[14] Kim, S.Pang,M.Je. Constructing support vector machine ensemble. Pattern Recognition, 2005(36):2757-2767

[15] I.Buciu. Demonstrating the stability of support vector machines for classification. Signal Processing. 2006(86): 2364-2380.

[16] W.Jeevani. Performance Degradation in Boosting. In conf. MCS 2001:multiple classifier systems,11-21.

[17] Z.Ying,X.Fang. Combining LBP and AdaBoost for Facial Expression Recognition. ICSP 2008 Proceedings: 1461-1464.

[18] S.Jung,D.Kim,K.An,M.Chung. Efficient Rectangle Feature Extraction for Real-time Facial Expression Recognition based on AdaBoost.

[19] Paul Viola, Michael Jones. Rapid Object Detection using a Boosted Cascade of Simple Features. Conference of Computer Vision and Pattern Recognition, 2001

[20] G.Yang,Z.Wang,J.Ren. Facial expression recognition based on adaboost algorithm. Chinese Journal of Computer Application. 25(4):946-948,2005

[21] W.Zhu,S.Luo. Face detection based on cascaded boosting algorithm. Chinese Journal of Computer Application. 25(9):2128-2130,2005

[22] J.Ren,G.Yang. Facial Expression Recognition based on Gabor Transform and AdaBoost Algorithm. Chinese Journal of MicroComputer Information. 23(3-1):290-292,2007

[23] J.Ruan,J.Yin. Multi-pose Face Detection Using Facial Features and AdaBoost Algorithm. The Second International Workshop on Computer Science and Engineering, 31-34, 2009

[24] L.Shen,L.Bai. A review on Gabor wavelets for face recognition. Pattern Anal Applic, (2006) 9:273-292.

[25] M.Turk, A.Pentland. Eigenfaces for recognition. J Cogn Neurosci, 3(1):71-86,1991.

[26] P.Belhumeur,J.Hespanha,D.Kriegman. Eigenfaces vs. Fisherfaces: recognition using class specific linear projection. IEEE Trans PAMI, 19(7):711-720, 1997

[27] M. Lades et al. Distortion invariant object recognition in the Dynamic Link Architecture. IEEE Trans Comput, 42(3):300-311,1993

[28] M.Grimm,D.Dastidar,K.Kroschel. Recognizing Emotions in Spontaneous Facial Expression. Internation Conference on Intelligent System and Computing, 2006.

[29] M.Lyons, J.Budynek,A.Plantey and S.Akamatsu. Classifying Facial Attributes Using a 2-D Gabor Wavelet and Discrimination Analysis. Proceedings of the 4th Int. Conf. on Automatic Face and Gesture Recognition,202-207,2000

[30] H.Deng, J.Zhu, M.Lyu, I.King. Two-stage Multi-class AdaBoost for Facial Expression Recognition. Proceedings of International Joint Conference on Neural Networks, 2007.

[31] Y.Wang, H.AI, B.Wu, C. Huang. Real Time Facial Expression Recognition with AdaBoost. Proceedings of the 17th International Conference on Pattern Recognition, 2004