레터논문 (Letter Paper)
방송공학회논문지 제18권 제4호, 2013년 7월 (JBE Vol. 18, No. 4, July 2013)
http://dx.doi.org/10.5909/JBE.2013.18.4.643
ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

샘플 군집화를 이용한 개선된 아다부스트 알고리즘

백 열 민[a], 김 중 근[a], 김 회 율[a]‡

An Improved AdaBoost Algorithm by Clustering Samples

Yeul-Min Baek[a], Joong-Geun Kim[a], and Whoi-Yul Kim[a]‡

요 약

본 논문에서는 아다부스트의 과적합 문제를 해결하기 위해 샘플 군집화를 이용한 개선된 아다부스트 알고리즘을 제안한다. 아다부스트는 다양한 객체 검출 방법에서 좋은 성능을 보이는 방법으로 알려져 있지만 훈련 샘플에 노이즈가 존재하는 경우 과적합 현상이 발생하는 문제가 있다. 이를 해결하기 위해 제안하는 방법은 우선 훈련 샘플의 긍정 샘플을 k-평균 군집화 알고리즘을 이용하여 K개의 군집으로 나눈다. 이후 아다부스트의 약분류기 훈련 시 K개의 군집 중 훈련 오차를 최소화하는 하나의 군집만을 선택하여 사용한다. 이로써, 제안하는 방법은 매 회 반복되는 약분류기의 훈련 시 훈련 샘플들이 과분할 되는 것과 노이즈 샘플이 훈련에 사용되는 것을 방지함으로써 기존 아다부스트의 과적합 현상을 효과적으로 줄여준다. 실험 결과, 제안하는 방법은 다양한 실제 데이터셋에서 기존의 부스팅 기반 방법들에 비해 더 나은 분류 성능 및 일반화 성능을 보여주었다.

Abstract

We present an improved AdaBoost algorithm to avoid overfitting phenomenon. AdaBoost is widely known as one of the best solutions for object detection. However, AdaBoost tends to be overfitting when a training dataset has noisy samples. To avoid the overfitting phenomenon of AdaBoost, the proposed method divides positive samples into K clusters using k-means algorithm, and then uses only one cluster to minimize the training error at each iteration of weak learning. Through this, excessive partitions of samples are prevented. Also, noisy samples are excluded for the training of weak learners so that the overfitting phenomenon is effectively reduced. In our experiment, the proposed method shows better classification and generalization ability than conventional boosting algorithms with various real world datasets.

Keyword : AdaBoost, Overfitting, Clustering, Classifier

Ⅰ. 서 론

아다부스트(AdaBoost)는 다수의 약분류기(weak classifier)들을 선형 결합하여 강분류기(strong classifier)를 만드는 기계 학습 알고리즘으로서 얼굴, 사람, 차량과 같이 다양

a) 한양대학교 전자컴퓨터통신공학과(Dept. of Electronics and Computer Engineering, Hanyang University)
‡ Corresponding Author : 김회율(Whoi-Yul Kim)
E-mail: wykim@hanyang.ac.kr
Tel: +82-2-2281-1759
· Manuscript received June 4, 2013 Revised July 15, 2013 Accepted July 15, 2013

한 객체 검출 문제에 널리 사용되고 있다[1]. 그러나 아다부스트는 훈련 샘플에 노이즈가 존재하는 경우 약분류기의 수가 늘어날수록 과적합(overfitting) 현상이 발생하는 문제점이 있다. 과적합 현상이란 분류기가 훈련 샘플에 특화되어 일반화 성능이 저하되는 현상으로서 작은 훈련 오차에도 불구하고 실제 테스트 오차는 크게 나타나게 된다. 아다부스트는 이전 단계의 약분류기가 오분류한 샘플의 가중치를 증가시켜 다음 약분류기에서 오분류된 샘플을 더 잘 분류하도록 한다. 그러나 이러한 방법이 노이즈 샘플에 대한 가중치를 지나치게 증가시키게 되어 약분류기의 훈련 횟수가 증가할수록 강분류기의 과적합 현상을 야기하고 일반화 성능을 저하시키게 된다[2-7].

본 논문에서는 이러한 아다부스트의 과적합 현상을 줄여 노이즈가 많은 데이터셋에서도 일반화 성능을 확보함으로써 분류 성능을 높인 개선된 아다부스트 알고리즘을 제안한다. 제안하는 방법은 샘플 군집화를 이용한 군집별 약분류기를 구성함으로써 기존의 아다부스트 개선 방법들과 다르게 가중치 갱신을 강제로 조정하는 것에 기인하는 기본 아다부스트의 성능 제약이 발생하지 않는다. 또한, 과적합을 발생시키는 노이즈 샘플들을 검출하는 것에 기반 하지 않기 때문에 노이즈 샘플의 검출에 기반 하여 샘플의 일부를 제거하거나 가중치를 작게 하는 방식들처럼 훈련 샘플의 정보를 훼손하거나 완벽하지 못한 노이즈 샘플 검출로 인한 문제점이 발생하지 않는다.

표 1. 제안하는 개선된 아다부스트의 의사 코드
Table 1. Pseudo code of proposed improved AdaBoost

Input: A set of training samples, $\{(\mathbf{x}_1,y_1),\cdots,(\mathbf{x}_m,y_m)\}$ where $y_i\in\{-1,+1\}$
A set of positive training samples, $S_P=\{(\mathbf{x}_i,y_i)|y_i=+1\}$
A set of negative training samples, $S_N=\{(\mathbf{x}_i,y_i)|y_i=-1\}$

1. **Divide** S_P into K clusters using k-means clustering algorithm,
2. Initialize $D_1(i) = 1/m$
3. **Do for** $t = 1,\cdots,T$
   a) Train weak learner using weight distribution $D_t(i)/L_t^k$ and clustered training samples $S_P^k \cup S_N^k, k = 1,\cdots,K$ where L_t^k is the normalization constant, and $\sum_{i\in S_P^k\cup S_P^k} D_t(i)=1$
   b) Calculate the training error of $h_k : \varepsilon_k = \sum_{i=1}^{m} D_t(i),\ y_i \neq h_k(\mathbf{x}_i)$,
   c) Calculate $k' = \arg\min_k(\varepsilon_k)$
   d) Get weak classifier $h_t = h_{k'}, \varepsilon_t = \varepsilon_{k'}$ and choose $\alpha_t = \frac{1}{2} ln\left(\frac{1-\varepsilon_t}{\varepsilon_t}\right)$
   e) if $\varepsilon_t > 0.5$ or $\varepsilon_t = 0$ then Generate uniformly distributed weights $D_t(i) = 1/m$, continue;
   Update weight distribution $D_{t+1}(i)= \frac{D_t(i)\exp(-\alpha_t y_i h_t(\mathbf{x}_i))}{Z_t}$ where Z_t is a normalization constant, and $\sum D_t(i)=1$.

Output: final strong classifier, $H(\mathbf{x})= \mathrm{sign}\left(\sum_{t=1}^{T}\alpha_t h_t(\mathbf{x})\right)$

II. 제안하는 방법

대부분의 기존 방법들이 아다부스트의 과적합 문제를 훈련 샘플의 지나치게 큰 가중치 문제와 노이즈 샘플의 검출 및 제거와 같이 샘플의 관점에서 고려한다. 그러나 샘플이 아닌 분류기의 관점에서 보면 아다부스트 훈련 시 과적합이 발생하는 이유는 약분류기의 트레이닝 횟수가 반복될수록 노이즈 샘플들을 과분할하는 분류 경계를 생성하기 때문이다. 이는 약분류기의 분류 성능이 단어 그대로 약하기 때문에 약분류기가 일반 샘플들과 노이즈 샘플들을 동시에 잘 분류하는 분류 경계를 생성하지 못하는 것에 기인한다. 따라서 두 개의 클래스를 가지는 이진 분류 문제에 있어, 훈련 샘플들 중 긍정 샘플들을 부정 샘플들과 선형적으로 잘 분류될 정도의 군집들로 분할한 후 약분류기가 하나의 군집만을 분류하게 하면 각 군집을 잘 분할하는 분류 경계만이 생성되어 약분류기의 훈련 횟수가 증가하여도 노이즈 샘플들이 과분할 되는 것을 방지 할 수 있다. 또한 이렇게 각각의 군집들을 분할하는 약분류기들의 결합으로도 일종의 부분 기반 분류기(part-based classifier)와 같은 형태를 가지게 되어 강분류기의 형성이 가능하다. 그리고 노이즈 샘플들이 다수 포함된 군집을 분류하는 약 분류기는 일반 샘플들이 다수 포함된 다른 군집들을 분류하는 약분류기들에 비해 더 큰 훈련 오차를 가질 확률이 더 높다. 이는 많은 경우에 노이즈 샘플들이 일반 샘플들과 상이한 분포를 가지고 있기 때문이다.

이러한 기본 착상들을 바탕으로 제안하는 개선된 아다부스트 알고리즘의 의사 코드는 표 1과 같다. 제안하는 방법은 우선 그림 1(a)와 같이 긍정 샘플들을 k-평균 군집화 알

고리즘을 이용하여 K개의군집으로 분할한다. 이 후 각 군집별로 하나의 긍정 샘플 군집과 부정 샘플 전체를 나누는 약분류기 후보들을 구한다. 이 후 그림 1(b)와 같이 훈련 오차를 최소로 만드는 군집의 해당 약분류기 후보를 현재 훈련 단계의 약분류기로 결정한다. 약분류기의 결정 후 샘플들의 가중치 변경은 기본적인 아다부스트 알고리즘과 동일하게 이루어진다. 이렇게 가장 작은 훈련 오차를 가지는 군집들로부터 생성된 약분류기를 계속 추가하게 되며 그림 1의 가운데 부분에 존재하는 노이즈 샘플들로 구성된 군집의 경우 지속적으로 큰 훈련 오차를 가지게 되기 때문에 약분류기로 선택되지 못하게 된다. 따라서, 군집별로 구해진 분류 경계들만 사용함으로 인해 훈련 샘플들의 과분할이 방지되며 아다부스트의 과적합을 발생시키는 노이즈 샘플들은 가장 작은 훈련 오차를 가지는 군집들만 사용하는 약분류기 선택 방식에 의해 훈련 과정에서 제외되게 된다. 이로써 제안하는 개선된 아다부스트 알고리즘은 과적합 현상을 효과적으로 방지 할 수 있다. 그러나 매 훈련 반복 단계 마다 K개의 군집 수 만큼의 약분류기 훈련 과정이 수행되기 때문에 전체 훈련 시간이 크게 증가한다. 또한, K의 값에 따라 알고리즘 성능이 달라질 수 있다. 따라서 최적의 성능 도출을 위해서는 적절한 K의 값을 결정하는 것이 중요하다. 이를 위해 훈련 샘플에 대한 사전 정보가 활용되어질 수 있다. 예를 들어, 훈련 샘플이 조명이나 형태에 따라 일정 개의 군집으로 분할되는 것이 예상 가능하다면 이를 적정 K의 결정을 위한 초기값 등으로 활용할 수 있다.

III. 실험 결과

본 논문에서는 제안하는 방법의 성능을 검증하기 위해 UCI-repository로부터 다양한 특징값 차원과 샘플 수를 가지는 9개의 실제 데이터셋을 사용하여 기존의 부스팅 알고리즘들과 성능을 비교하였다[8]. 비교를 위한 알고리즘들로는 GML AdaBoost Matlab Toolbox에서 제공하는 리얼 아다부스트(Real AdaBoost), 젠틀 아다부스트(Gentle AdaBoost), 그리고 모디스트 아다부스트(Modest AdaBoost)가 사용되었다[9]. 리얼 아다부스트는 최초로 제안된 이진 아다부스트의 일반화 버전으로 기본적인 아다부스트

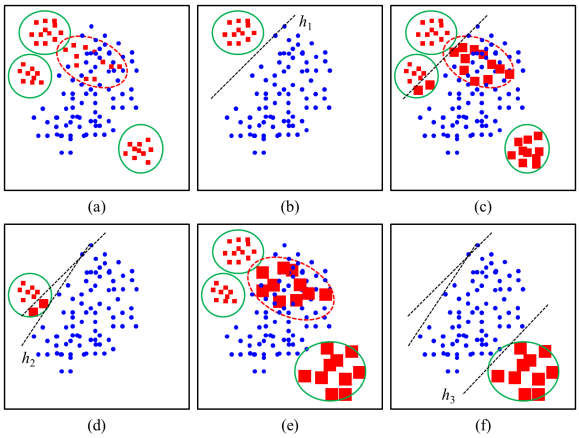

그림 1 . 제안하는 방법의 흐름도 (a) 주어진 훈련 샘플과 긍정 샘플 (붉은색 사각형)의 군집화, (b) 훈련 오차를 최소화하는 하나의 클러스터를 선택하여 첫 번째 약 분류기 훈련, (c) 샘플들의 가중치 분포 갱신, (d-f) 연속되는 부스팅 훈련과정에서 약 분류기 훈련 및 샘플 가중치 분포 갱신
Fig. 1. The flow of proposed method. (a) given training samples and clustering positive samples (red square), (b) weak learning with the selected cluster to minimize the training error at first boosting round, (c) update weight distribution of samples, (d-f) weak learning and update weight distribution at consecutive boosting rounds.

알고리즘으로 간주된다[1]. 그리고 젠틀 아다부스트와 모디스트 아다부스트는 기본 아다부스트의 과적합 현상을 제거하기 위해 개발된 알고리즘들이다[2,3]. 제안하는 방법을 비롯하여 비교를 위한 모든 부스팅 알고리즘들은 동일하게 결정 그루터기(decision stump)를 약분류기로 사용하였다. 제안하는 방법은 모든 실험에서 긍정 샘플을 10개의 군집으로 분할하였다. 또한, 노이즈 샘플의 양에 따른 알고리즘 성능 검증을 위해 앞서 선택된 데이터 셋들에 노이즈를 10%와 20% 추가한 데이터셋을 구성하였다. 따라서 총 27개의 실제 데이터셋들이 실험에 사용되었다. 노이즈의 추가 방법은 원래의 데이터셋에서 해당 비율의 샘플을 무작위로 추출하여 클래스 라벨을 반대로 바꾸어 주는 방식으로 수행하였다. 각 데이터셋의 샘플들 중 무작위로 추출한 절반은 훈련용으로 사용하고 나머지 절반은 테스트용으로 사용하여 훈련 오차 및 테스트 오차를 측정하였으며 이러한 과정을 100회 반복하여 매회 얻어진 훈련 오차와 테스트 오차를 평균한 결과를 성능 검증에 사용하였다. 이렇게 얻어진 평균 테스트 오차는 표 2와 같다.

표 2에서 보여지는 바와 같이 제안하는 방법은 노이즈가

없는 경우, 전체 데이터셋들 중 4개의 데이터셋에서 가장 좋은 성능을 보이며 모디스트 아다부스트는 3개, 젠틀 아다부스트는 2개의 데이터셋에서 가장 좋은 성능을 나타낸다. 따라서 노이즈가 추가되지 않은 실제 데이터셋들에서 제안하는 방법이 과적합 현상을 제거하기 위한 기존의 부스팅 방법들보다 약간 더 나은 성능을 나타냄을 확인할 수 있다. 노이즈 샘플들이 많이 존재하여 과적합 현상이 발생하기 쉬운 노이즈 추가 데이터셋들에 대해서는 제안하는 방법이 기존의 방법들에 비해 훨씬 더 나은 성능을 보임을 확인할 수 있었다. 노이즈가 10% 추가된 경우, 제안하는 방법은 6개의 데이터셋에서 가장 좋은 성능을 보였으며 모디스트 아다부스트는 3개의 데이터셋에서 가장 좋은 성능을 보였다. 노이즈가 20% 추가된 경우에는 제안하는 방법이 7개의 데이터셋에서 가장 좋은 성능을 보인 반면 모디스트 아다부스트는 단지 2개의 데이터셋에서 가장 좋은 성능을 보였다.

III. 결 론

본 논문에서는 기존 아다부스트의 과적합 현상을 해결하기 위한 개선된 아다부스트 알고리즘을 제안하였다. 제안하는 방법은 훈련 샘플들을 군집화 하고 약분류기의 훈련 시 전체 훈련 오차를 최소화 하는 하나의 군집만을 사용함으로써 과적합 현상을 효과적으로 방지한다. 다양한 실제 데이터 셋을 이용한 실험 결과, 제안하는 방법은 기존의 방법들과 달리 기본 아다부스트의 성능을 제약하거나, 훈련 샘플의 정보를 훼손하지 않으면서 효과적으로 과적합 현상을 방지하기 때문에 기존의 부스팅 방법들에 비해 더 나은 분류 성능과 일반화 성능을 보여주었다.

참 고 문 헌

[1] R. E. Schapire and Y. Singer, “Improved boosting algorithms using confidence-rated predictions,” Machine Learning, vol. 37, no. 3 pp. 297-336, 1999.

[2] J. Friedman, T. Hastie, and R. Tibshirani, “Additive logistic regression: A statistical view of boosting,” The Annals of Statistics, vol. 38, no. 2, pp. 337-374, 2000.

[3] A. Vezhnevets and V. Vezhnevets, “Modest AdaBoost - Teaching AdaBoost to Generalize Better,” Graphicon, vol. 12, no. 5, pp. 987-997, 2005.

[4] S. Merler, B. Caprile, and C. Furlanello, “Bias-Variance Control via Hard Points Shaving,” International Journal of Pattern Recognition and Artificial Intelligence, vol. 18, no. 5, pp. 891-903, 2004.

[5] D.-S. Kim, Y.-M. Baek, and W.-Y. Kim, “Reducing Overfitting of AdaBoost by Clustering-based Pruning of Hard Examples,” Proceedings of the 7th International Conference on Ubiquitos Information Management and Communication, no. 90, 2013.

[6] A. Vezhnevets and O. Barinova, “Avoiding boosting overfitting by removing confusing samples,” Proceedings of European Conference on Machine Learning, pp. 430-441, 2007.

[7] J. Cao, S. Kwong, and R. Wang, “A noise-detection based AdaBoost algorithm for mislabeled data,” Pattern Recognition, vol. 45, no. 12, pp. 4451-4465, 2012.

[8] A. Frank, and A. Asuncion, UCI Machine Learning Repository, <http://archieve.ics.uci.edu/ml/>, 2013.

[9] A. Vezhnevets, GML Matlab Toolbox, Technical Manual, Graphics and Media Lab., Computer Science Department, Moscow state University.

표 2. 약분류기 훈련 반복 1000회 후 UCI 데이터셋들의 평균 테스트 오차 (%)
Table 2. Average test error (%) of UCI datasets after 1000 training iterations
(G: 젠틀 아다부스트, M: 모디스트 아다부스트, R: 리얼 아다부스트, Pro: 제안하는 방법, 굵은 숫자는 해당 데이터셋에서 가장 작은 테스트 오차를 나타냄.)

Noise	0%				10%				20%			
Dataset	G	M	R	Pro	G	M	R	Pro	G	M	R	Pro
Aust	17.46	14.60	17.78	**14.49**	26.15	22.42	26.47	**20.61**	31.35	28.55	31.36	**25.65**
Breast	4.52	**3.79**	4.31	7.04	14.64	**10.86**	14.62	13.38	24.96	**18.53**	25.36	20.05
German	**28.44**	29.01	28.66	36.40	35.22	**31.95**	35.20	37.95	40.49	**36.41**	40.44	39.63
Haber	33.91	**25.79**	34.30	32.61	40.28	**35.02**	41.64	39.94	43.22	39.47	44.30	**37.52**
Heart	23.59	19.26	23.39	**17.63**	37.60	35.00	36.81	**26.96**	43.90	42.53	43.21	**32.89**
Iono	9.43	**8.73**	8.61	10.49	18.39	18.49	18.17	**16.47**	24.36	23.84	23.80	**19.56**
Pima	28.22	25.42	28.61	**23.93**	35.86	35.59	35.66	**21.13**	40.30	37.48	39.86	**33.92**
Sonar	20.82	22.93	20.97	**14.73**	32.28	30.22	32.13	**21.23**	41.28	39.56	41.02	**23.17**
Wdbc	**3.49**	4.37	**3.49**	5.43	17.03	16.70	16.57	**14.85**	27.38	26.07	26.94	**21.13**