An Aggressive Margin-Based Algorithm for Incremental Learning

JuiHsi Fu* and SingLing Lee

National Chung Cheng University
168 University Road, Minhsiung Township,
Chiayi 62162, Taiwan, R.O.C.
{fjh95p,singling}@cs.ccu.edu.tw

Abstract. In incremental learning, the classification model is incrementally updated using the small datasets. Different with existing methods, our approach updates the current classifier according to each sample in the dataset, respectively. The classifier is updated by adjusting more than the margin of each sample. Then the new classifier is generated by carefully analyzing classifier adjustments caused for labeled samples. Additionally the new classifier shall correct prediction mistakes of the previous classifier as many as possible. In details, we formulate simple constrained optimization problems and then the updated classifier is the solution derived using Lagrange multipliers. In our experiments, 13 real-world dataset are used to present the effectiveness of the proposed approach. The experimental results are shown that our update strategy is able to adjust the classifier properly. And it is also shown that the proposed incremental learning approach is suitable to be applied for the requirement of frequently adjusting the existing classifiers.

Keywords: Incremental Learning, Margin-based Approaches, Passive-Aggressive (PA) Algorithm, Period Datasets, Classifier Adjustment.

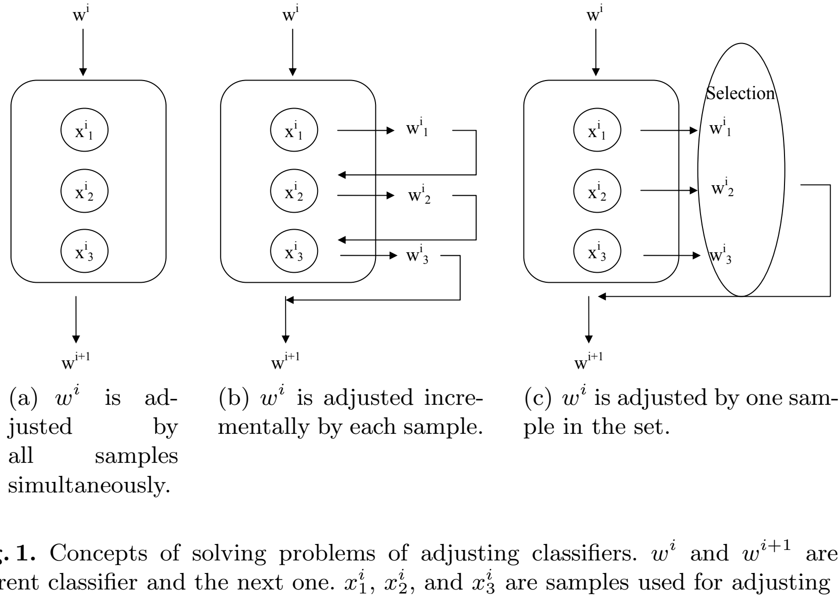
1 Introduction

Requests of analyzing collected period data have been emerged in recent practical applications that includes network traffic analysis [1], anomaly detection [2], and intrusion detection [3]. Generally, those applications are implemented for adjusting classifiers/detectors periodically. Most of incremental learning approaches have been proposed based on decision-tree [4], neural network [5,6], and Support Vector Machines (SVM) [3,7,8,9,10]. Typically they are designed to build the statistic classification model based on the previously seen samples and to correct its prediction mistakes on new labeled samples. While focusing on the sample space, SVM generalizes the separating hyperplane (classifier) based on the whole sample distribution, and maximizes the margins of labeled samples (support vectors). The margin of a sample is a distance between the sample and the separating hyperplane. And SVM is theoretically proven that

* This work is supported by NSC, Taiwan, under grant no. NSC 99-2218-E-492-006.

P.-N. Tan et al. (Eds.): PAKDD 2012, Part I, LNAI 7301, pp. 62–73, 2012.

the hyperplane is able to well separate samples with different labels. In [10], an incremental batch SVM approach was designed to update the classifier by solving a constrained optimization problem based on each set of collected samples. An example is illustrated in Fig. 1 (a) where the classifier w^i is adjusted as w^{i+1} depending on the set of samples, $\{x_1^i, x_2^i, x_3^i\}$. This approach should solve a complicated constrained optimization problem since those collected samples are adopted simultaneously. Other approaches [8,9] adjusted SVM classifiers incrementally by identifying each new sample as a support vector or not. Different with [10], in Fig. 1 (b) the classifier w^i is adjusted as w_1^i using first sample x_1^i in the set, and then w_1^i is updated as w_2^i using x_2^i. Thus w^i is incrementally adjusted as w^{i+1} depending on each sample in the set. The advantage of [8,9] is to maintain useful samples that were previously seen as support vectors and to obtain efficient update steps without solving a constrained optimization problem. But in those SVM approaches, the hyperplanes might not be quickly adjusted when encountering diverse sample distribution. In other words, the diverse samples have small chances to be support vectors because the distribution of those samples is significantly different with the distribution of samples in the set. Thus in this paper, our approach is to simplify the constrained optimization problem for update steps and to adapt the diverse sample distribution for classifiers.



(a) w^i is adjusted by all samples simultaneously.

(b) w^i is adjusted incrementally by each sample.

(c) w^i is adjusted by one sample in the set.

Fig. 1. Concepts of solving problems of adjusting classifiers. w^i and w^{i+1} are the current classifier and the next one. x_1^i, x_2^i, and x_3^i are samples used for adjusting w^i.

Rather than training the SVM classifier based on each sample or each set of collected samples, our approach adjusts the current classifier incrementally according one sample in each collected set. Thus for each potential update, we formulate an optimization problem with single constraint. Additionally our updated classifier shall correct prediction mistakes of the previous classifier as many as possible. Compared with [10], we divide a complicated constrained optimization problem into several simpler ones. In other words, the classifier is adjusted as several potential ones depending on different samples. An example is illustrated in Fig. 1 (c). The classifier w^i is adjusted as w_1^i, w_2^i, and w_3^i respectively using x_1^i, x_2^i, and x_3^i. And then the classifier that adjusts the most w^i's mistakes is selected as the next w^{i+1}. In this paper, we are motivated by the simplicity of online Passive-Aggressive (PA) algorithm [11]. One sample's margin is selected as the basis for classifier adjustment. Thus in our approach, while a sample is used for updating and its sign is incorrectly predicted, the classifier adjustment is aggressively achieved within the margin. Additionally the updated classifier shall correct prediction mistakes of the previous classifier as many as possible. In this paper, we formulate a simple constrained optimization problem for each sample and then the candidate updated classifier is the solution derived using Lagrange multipliers. It is noted that, we get a closed form solution for each potential updated classifier. Particularly the selected new classifier, updated by the suitable margin, shall obtain the best classification accuracy on the collected dataset. It is expected that, this selection strategy is able to avoid the new classifier being extremely specific to the previous one. And the updated classifier could flexibly adapt the diverse sample distribution because there is no need for the proposed approach to maintain previously seen samples.

Basically PA has the ability to frequently update the classifiers, but its two straightforward approaches may not be able to achieve impressive results. Firstly, PA update steps are specific to each labeled sample whether it is inconsistent or not. The consequence is that updated classifiers would obtain the unstable prediction ability. Secondly, the other PA approach is to update the classifier respectively using each sample. Then the selected classifier among all updated ones shall have the best classification accuracy on the collected dataset. Compared with our proposed approach, this approach does not actively correct prediction mistakes of the previous classifier. Thus these two approaches do not fully utilize the learning knowledge in each collected dataset. Moreover our approach is similar with re-sampling approaches, like bagging [12], to obtain improved classification accuracy by depending on subsets of the sample set. The major difference is that, we focus on designing efficient update steps for online applications so that a closed form solution for the updated classifier could be obtained.

The rest of our paper is organized as follows. The online PA algorithm is reviewed in Section 2. In Section 3, we detailedly describe the proposed approach and build the mathematic model. Experimental results are presented in Section 4. Finally, we conclude the paper in Section 5.

2 Online Passive-Aggressive Algorithm

In online learning, each training sample is discarded after it is used to update the classifiers. Some research works like the Perceptron algorithm [13,14,15] and margin-based approaches [16,17] have been proven to be effective in a board range of applications. Additionally it is worth noting the Passive-Aggressive

(PA) Algorithm [11] is a margin-based online learning approach that could be applied for various prediction tasks. PA uses linear predictors for label prediction of each incoming sample. And each update step of PA is executed depending on the margin of the labeled sample. The objective of PA update is to adjust the previous classifier as less as possible while the condition of classifier adjustment is satisfied. At the round t, let w^t be the vector of weights, x^t be the sample, $y^t \in \{+1, -1\}$ be x^t's true label, and the term $y^t(w^t \cdot x^t)$ be the signed margin. The new classifier w^{t+1} is the solution to the following constrained optimization problem,

$$w^{t+1} = argmin_{w \in R^n} \frac{1}{2}||w - w^t||^2 \quad s.t. \quad l(w, (x^t, y^t)) = 0, \tag{1}$$

where $l(w, (x^t, y^t))$ is the hinge loss of w's prediction on x^t.

$$l(w, (x, y)) = \begin{cases} 0, & y(w \cdot x) \geq 1 \\ 1 - y(w \cdot x), & \text{otherwise} \end{cases} \tag{2}$$

Typically whenever the loss is zero, PA is *passive* and $w^{t+1} = w^t$ means no classifier adjustment. And while the loss is positive (less than 1), w^t is *aggressively* updated by adjusting more than the margin, $y^t(w^t \cdot x^t)$, and then the constrain $l(w^{t+1}, (x^t, y^t)) = 0$ can be satisfied. Then the Lagrangian of the optimization problem in Eq. (1) is defined as Eq. (3).

$$L(w, \tau) = \frac{1}{2}||w - w^t||^2 + \tau(1 - y^t(w \cdot x^t)) \tag{3}$$

Let the partial derivation of l with respect to w be zero and then let the deviation of τ with respect to τ be zero, we have

$$w = w^t + \tau y^t x^t$$
$$\tau = \frac{1 - y^t(w^t \cdot x^t)}{||x^t||^2}$$

Ultimately the PA update is performed by solving the constrained optimization problem in Eq. (1). And it is theoretically shown that the aggressive update strategy of PA modifies the weight vector as less as possible. The effectiveness of PA in solving problems of classification and regression is formally analyzed in [11]. Based on this well-defined learning model of PA, several online algorithms [18,19] have been proposed for adding confidence information and handling non-separable data.

3 Incremental Passive-Aggressive Learning Algorithm

While each set of labeled period samples comes, the existing classifier shall be periodically updated for adapting the latest sample distribution. In this paper, we propose an incremental learning algorithm, named Incremental Passive-Aggressive (IPA). It adjusts the current classifier incrementally using one sample in each collected set. For each potential sample, there are two update steps in IPA: 1) to correct prediction mistakes of the current classifier, and 2) to aggressively update the current classifier by adjusting more than the margin. At last, the error minimization classifier on the collected dataset is selected as the next classifier. Before formulating the model of the proposed approach, we define some notations. Given the labeled dataset K^t collected at the round t, there are $|K^t|$ sample-label pairs, $\{(x_1, y_1), ..., (x_{|K^t|}, y_{|K^t|})\}$. w^t is the classifier at the round t, the vector of weights. When using each labeled sample $x_k \in K^t$, the updated classifier w^{t+1} shall correct mistakes of the previous classifier w^t as many as possible and w^t shall be adjusted as less as possible. Aggressively, if x_k obtains the incorrect predicted sign from w^t, then the adjustment for w^t should be achieved within more than x_k's margin. Thus these update steps to w^t are formulated as the constrained optimization problem,

$$\begin{aligned} f(w^t, (x_k, y_k), K^t) &= argmin_{\overline{w} \in R^n} \{\frac{1}{2} ||\overline{w} - w^t||^2 \\ &+ C_0 \sum_{x_i \in K^t, x_i \neq x_k} l(\overline{w}, (x_i, y_i))\} \\ &s.t.\ l(\overline{w}, (x_k, y_k)) = 0, \end{aligned} \tag{4}$$

where C_0 is a constant to control the tradeoff between the classifier deviation and the corrected prediction mistakes, and $l(\overline{w}, (x_i, y_i))$ is the hinge loss function.

Furthermore, after w^t is updated using every sample $x_k \in K^t$ according to Eq. (4), those updated classifiers, $\{f(w^t, (x_k, y_k), K^t) : 1 \leq k \leq |K^t|\}$, are the candidates for the new classifier. In order to avoid the new classifier being extremely specific to the current classifier, the selection strategy is to find the proper classifier which has the most accurate classification performance on K^t. When more than one updated classifiers have the highest classification accuracy, we select the updated classifier which has the smallest difference with w^t. Hence the new classifier w^{t+1}, selected among the candidate set of the updated classifiers, is the solution to the optimization problem,

$$w^{t+1} = argmin_{w \in \{f(w^t, (x_k, y_k), K^t)\ :\ 1 \leq k \leq |K^t|\}} C \sum_{x_i \in K^t} l(w, (x_i, y_i)) + ||w - w_t||, \tag{5}$$

where C is a large constant in order to select w strongly depending on the errors.

To solve the problem in Eq. (4), let $C_0 = 1$ and κ^t, the subset of $|K^t|$, be the set of samples of which predicted labels are incorrectly decided by w^t. While the loss of each sample in κ^t is positive (less than 1), the Lagrangian of the constrained optimization problem is defined as Eq. (6):

$$L(\overline{w}, \tau) = \frac{1}{2} ||\overline{w} - w^t||^2 + \sum_{x_i \in \kappa^t, x_i \neq x_k} (1 - y_i(\overline{w} \cdot x_i)) + \tau(1 - y_k(\overline{w} \cdot x_k)) \tag{6}$$

Let the partial derivation of l with respect to $\overline{w}$ be zero,

$$\begin{aligned}\nabla_{\overline{w}} L(\overline{w}, \tau) &= \overline{w} - w^t - \sum_{x_i \in \kappa^t, x_i \neq x_k} y_i x_i - \tau y_k x_k \\ => \overline{w} &= w^t + \sum_{x_i \in \kappa^t, x_i \neq x_k} y_i x_i + \tau y_k x_k \end{aligned} \tag{7}$$

Then substituting Eq. (7) into Eq. (6), we have

$$\begin{aligned} L(\tau) = &\frac{1}{2} || \sum_{x_i \in \kappa^t} y_i x_i + \tau y_k x_k ||^2 \\ &+ \sum_{x_i \in \kappa^t, x_i \neq x_k} (1 - y_i((w^t + \sum_{x_i \in \kappa^t, x_i \neq x_k} y_i x_i + \tau y_k x_k) \cdot x_i)) \\ &+ \tau (1 - y_k((w^t + \sum_{x_i \in \kappa^t, x_i \neq x_k} y_i x_i + \tau y_k x_k) \cdot x_k)) \end{aligned} \tag{8}$$

At last let the deviation of Eq. (8) with respect to τ be zero,

$$\begin{aligned} 0 &= -\tau y_k^2 ||x_k||^2 + (1 - y_k(w^t \cdot x_k)) - y_k x_k \sum_{x_i \in \kappa^t, x_i \neq x_k} y_i x_i \\ => \tau &= \frac{1 - y_k(w^t \cdot x_k) - y_k x_k \sum_{x_i \in \kappa^t, x_i \neq x_k} y_i x_i}{||x_k||^2} \end{aligned} \tag{9}$$

Ultimately, each update of the proposed incremental learning algorithm is performed by solving the constrained optimization in Eq. (4) and the updated classifier is determined by solving Eq. (5). It is theoretically presented in Eq. (7) and (9) that the update to the current classifier w^t is performed by correcting its prediction mistakes κ^t, and by adjusting it within the margin when the sample is incorrectly predicted. Overall the proposed algorithm is presented in Algorithm 1. At each round t, the dataset K^t is collected to update the current classifier w^t. And the samples of which predicted labels are incorrectly assigned by w^t are identified as κ^t, at line 4-5. Then for each sample $x_k \in K^t$, the current classifier w^t is individually updated as the candidate classifier $\overline{w_k}$ according to Eq. (7) and (9), at line 7-8. At last, the classifier $\overline{w_k}$ is selected as w^{t+1} if it gains the least prediction errors on K^t, at line 10. Particularly at the first round, w^1 is initialized as $(0, ..., 0)$ and its prediction result is always positive. Thus the w^1 is adjusted as the first updated classifier w^2 depending on the false positive sample that could cause the minimum $||w^2 - w^1||$. Moreover in addition to minimizing the classifier deviation, we correct mistakes of the previous classifier. In terms of convergence, each classifier is adjusted as small as possible. Also it is expected that, our approach is able to adaptively enhance the degree of adjusting classifiers when encountering diverse sample distribution that would cause significant prediction losses.

Algorithm 1. Incremental PA Learning Algorithm

```
input : $C_0$
Initialize: $w^1 = (0, ..., 0)$, $C = 10,000$ ;
for $t$=1,2,... do
    receive the collected labeled dataset $K^t$ ;
    predict $\widehat{y_x}$=sign($w^t \cdot x_k$) for each $x_k \in K^t$ ;
    collect $\kappa^t = \{x_k | x_k \in K^t \ and \ y_x \neq \widehat{y_x}\}$ ;
    for each $x_k \in K^t$ do
        set $\tau_k = \frac{1 - y_k(w^t \cdot x_k) - y_k x_k \sum_{x_i \in \kappa^t, x_i \neq x_k} y_i x_i}{||x_k||^2}$ ;
        update $\overline{w_k} = w^t + \sum_{x_i \in \kappa^t, x_i \neq x_k} y_i x_i + \tau y_k x_k$ ;
    end
    select $w^{t+1} = argmin_{w \in \{\overline{w_k} \,:\, 1 \leq k \leq |K^t|\}} C \sum_{x_i \in K^t} l(w, (x_i, y_i)) + ||w - w^t||$ ;
end
```

4 Experiments

In this section, our experiments are designed to present the performance of our approach in classification accuracy while the classifier is incrementally updated by several small training sets. To present the effectiveness of updating classifiers in our approach, we also implement the online PA and an incremental batch SVM [9]. Additionally in order to show the effectiveness of correcting mistakes of the previous classifier in eq. (4), the performance of our approach with $C_0 = 0$ is also compared in following experiments. In terms of evaluating classification accuracy of a classifier, we would like to significantly present classification results of samples in two different classes. We use the measurement of micro-average accuracy to average the classification accuracies that are calculated in two classes, respectively. For consistence, the summations of loss errors in the eq. (4) and (5) are also revised as (1 - micro-average accuracy).

Table 1 presents 13 real-world data collections from 4 different sources used in our experiments. The *multi-domain sentiment dataset* [1] contains product reviews downloaded from Amazon.com from four product types (domains): Kitchen, Books, DVDs, and Electronics. Each domain has several thousand reviews, but the exact number varies by domain. In this experiment, only Books, DVDs are used for evaluating performance of those learning approaches. From the second data source, the dataset at *ECML/PKDD-2006 discovery challenge*[2] is used to decide whether received emails are spam or non-spam. Especially there are over 10,000 features in those three datasets, Books, DVDs, and Emails. But it is difficult to analyze performance of the SVM classifiers implemented in Matlab [9] because the execution is time consuming on those high dimensional datasets. Thus we randomly select a part of documents, as presented in Tab. 1, in following experiments. From the third data source, *Spamming Bots* [20] is the set of response codes of the sent emails, collected in National Chung Cheng

[1] Sentiment. http://www.cs.jhu.edu/ mdredze/datasets/sentiment/

[2] ECML/PKDD-2006. http://www.ecmlpkdd2006.org/challenge.html

Table 1. 10 real-world datasets: sizes of the classes and the size of feature dimensions

Dataset	Source	Class(size)	Class(size)	Dimensions
DVDs	Sentiment	positive(292)	negative(300)	1488
Books	Sentiment	positive(289)	negative(287)	1548
Emails	ECML/PKDD	spam(210)	non-spam(445)	1034
Connectionist Bench	UCI	1(111)	2(97)	60
Ionosphere	UCI	b(126)	g(225)	34
German	UCI	Good(700)	Bad(300)	23
Australian Credit Approval	UCI	0(383)	1(307)	14
Statlog (Heart)	UCI	1(150)	2(120)	13
yeast	UCI	CYT(463)	ME1(44)	9
abalone	UCI	10(634)	4(57)	8
Pima Indians Diabetes	UCI	0(500)	1(268)	8
ecoli	UCI	cp(143)	im(77)	8
Spamming Bots	CCU	normal(1560)	spamming(150)	5

University (CCU). It is used to analyze the behavior of each email sender and then to detect the spamming bots. At last the other datasets are the benchmarks in the *UCI repository* [3]. While we evaluate classification performance of learning approaches, we randomly divide each dataset into 10 subsets, and one of subsets is received at each round. In other words, one subset is used for initially training the classifier and deciding the value of C_0 in eq. (4) by obtaining the highest classification accuracy on the first subset. Then others are received at each of 9 rounds. The classification accuracy at each round is measured by classification results of the classifier updated at previous rounds. To reduce variability in experimental results, we arrange 10 subset-round permutations on each dataset and average those 10 classification accuracies at each round.

At first these experiments, except on *Diabetes* in Fig. 2, are demonstrated that the proposed IPA has better performance than IPA with $C_0 = 0$. That means, in addition to minimizing the classifier deviation, it is effective in eq. (4) to correct mistakes for updating the previous classifier. And on *Diabetes*, correction of mistakes to the classifier could not improve the classification accuracy on latter samples. It seems, on *Diabetes* previous learning knowledge is not useful for latter label prediction. Secondly on *Australian*, *Ionosphere*, *Bots*, and *10+4* in Fig. 3-4, it is presented that the online PA method can not obtain the remarkable classification performance since its update strategy is specific to each labeled sample. That means, the online PA method tends to be updated by inconsistent samples. Furthermore, except experimental results on *Australian* and *Ionosphere* in Fig. 3, it is shown that our approach obtains the best (or similar) classification accuracy in comparison with other approaches. We update the classifier by carefully analyzing classifier adjustment caused for the labeled dataset. Then the remarkable classification accuracy is obtained at each round after the classifier is incrementally updated on most of datasets. Also it is shown

[3] UCI Repository. http://archive.ics.uci.edu/ml/

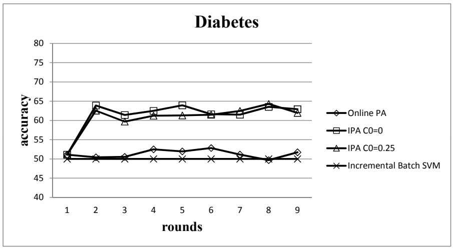


Fig. 2. Classification results of incremental learning approaches on *Diabetes*

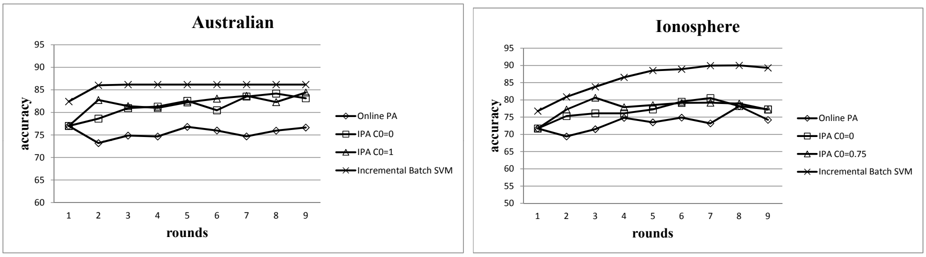


Fig. 3. Classification results of incremental learning approaches on *Australian* and *Ionosphere*

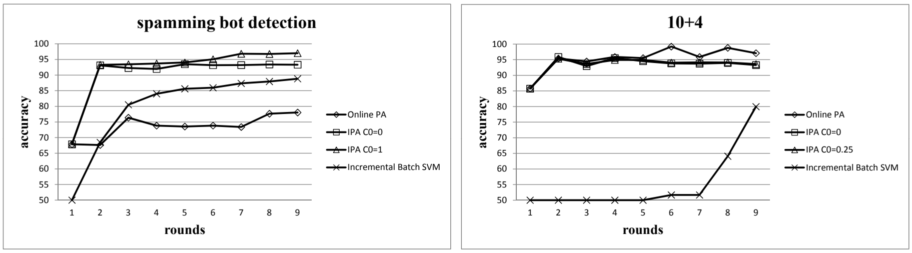


Fig. 4. Classification results of incremental learning approaches on *bot* and *10+4*

that our approach has the ability to adapt the diverse sample distribution for classifiers because we obtain better performance in accuracy than the SVM approach of which support vectors are maintained as informative samples. Mention to the performance on *Australian* and *Ionosphere*, it seems ambiguous or noise samples exist so that the approaches (PA and IPA) to incrementally update the classifier by one sample do not have impressive results. In this case, collected samples in the set might be simultaneously used for updating classifiers, like the incremental batch SVM, to filter out misleading or noise samples.

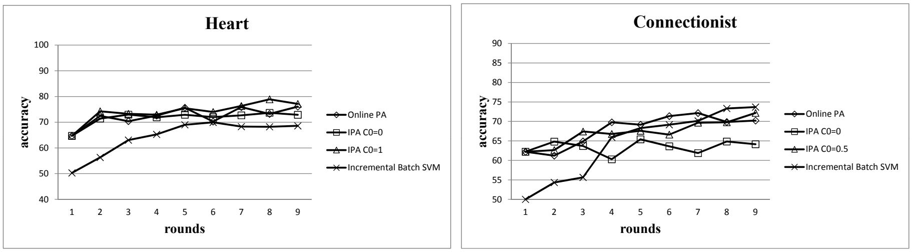


Fig. 5. Classification results of incremental learning approaches on *heart* and *Connectionist*

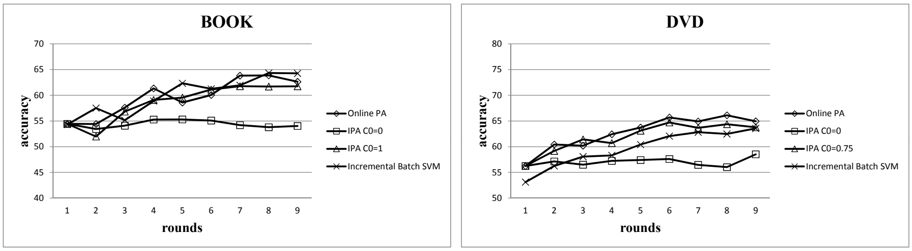


Fig. 6. Classification results of incremental learning approaches on *BOOK* and *DVD*

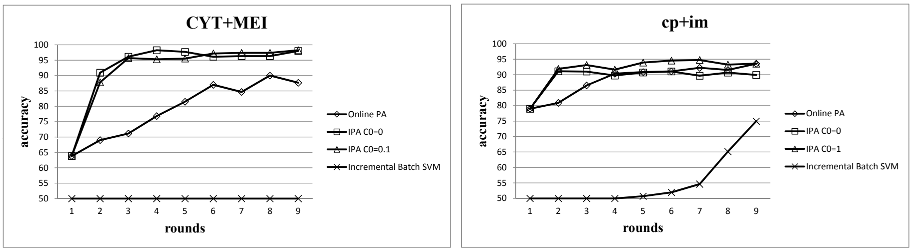


Fig. 7. Classification results of incremental learning approaches on *CYT+ME1* and *cp+im*

Interestingly on *CYT+MEI*, *cp+im*, *German*, and *Emails* in Fig. 7-8, the incremental batch SVM approach has biased results. It is observed that, in estimating performance of the classifier, it focuses on non-weighting estimated errors, instead of average weights for errors on two respective classes. Still on those datasets, proposed IPA has the practical ability to obtain the best classification accuracy. Hence, our approach to update classifiers is not affected by biased classification results.

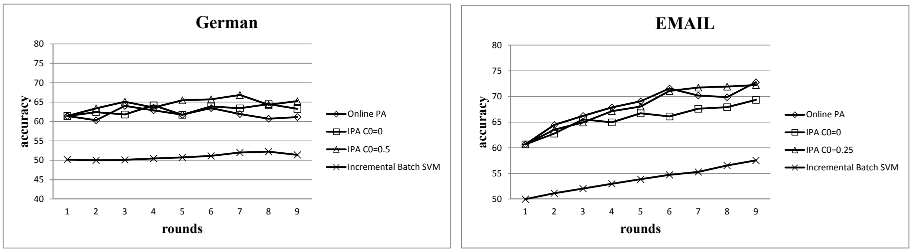


Fig. 8. Classification results of incremental learning approaches on *German* and *Emails*

5 Conclusion

In this paper, we propose an efficient incremental learning approach to deal with the practical requirement of frequently updating classifiers. Our approach is proposed to adjust the classifier incrementally using one sample in each collected set. That is, the classifier is aggressively updated by adjusting more than the margin of a sample, and its prediction mistakes are corrected as more as possible. For each potential update step, we get a closed form solution for the updated classifier through solving a simple constrained optimization problem. At last the selected classifier shall have the least prediction errors on the collected dataset. Our experimental results are presented that, when updating a classifier, it is effective to correct its prediction mistakes, in addition to minimizing the classifier deviation. And it is also shown that our approach has the ability to adapt the diverse sample distribution for classifiers. Except several datasets that consist of some misleading or noise samples, the classifier that is incrementally adjusted by our approach is able to gain remarkable classification accuracy. Therefore it is presented that the proposed approach is suitable to be applied for effectively adjusting the existing classifiers using periodically collected datasets.

References

1. Sena, G.G., Belzarena, P.: Early traffic classification using support vector machines. In: 5th International Latin American Networking Conference, pp. 60–66. ACM, New York (2009)
2. Robertson, W.K., Maggi, F., Kruegel, C., Vigna, G.: Effective Anomaly Detection with Scarce Training Data. In: The Network and Distributed System Security Symposium. ISOC (2010)
3. Du, H., Teng, S., Yang, M., Zhu, Q.: Intrusion Detection System Based on Improved SVM Incremental Learning. In: International Conference on Artificial Intelligence and Computational intelligence, pp. 23–28. IEEE Press (2009)
4. Utgoff, P.E.: Incremental Induction of Decision Trees. J. Machine Learning 4, 161–186 (1989)
5. Mohamed, S., Rubin, D., Marwala, T.: Incremental Learning for Classification of Protein Sequences. In: International Joint Conference on Neural Networks, pp. 19–24. IEEE Press (2007)

6. Chen, Z., Huang, L., Murphey, Y.L.: Incremental Learning for Text Document Classification. In: International Joint Conference on Neural NetWorks, pp. 2592–2597. IEEE Press (2007)
7. Ruping, S.: Incremental Learning with Support Vector Machines. In: International Conference on Data Mining, pp. 641–642. IEEE Press (2001)
8. Xiao, R., Wang, J., Zhang, F.: An Approach to Incremental SVM Learning Algorithm. In: International Conference on Tools with Artificial Intelligence, pp. 268–273. IEEE Press (2000)
9. Cauwenberghs, G., Poggio, T.: Incremental and Decremental Support Vector Machine Learning. In: Neural Information Processing Systems, vol. 13. MIT Press, Cambridge (2001)
10. Liu, Y., He, Q., Chen, Q.: Incremental Batch Learning with Support Vector Machines. In: 5th World Congress on Intelligent Control and Automation, pp. 1857–1861. IEEE Press (2004)
11. Crammer, K., Dekel, O., Keshet, J., Shwartz, S.S., Singer, Y.: Online Passive-Aggressive Algorithms. J. Machine Learning Research 7, 551–585 (2006)
12. Zhu, X.: Lazy Bagging for Classifying Imbalanced Data. In: 7th IEEE International Conference on Data Mining, pp. 763–768 (2007)
13. Freund, Y., Schapire, R.E.: Large Margin Classification Using the Perceptron Algorithm. J. Machine Learning 37, 277–296 (1999)
14. Ng, H.T., Goh, W.B., Low, K.L.: Feature selection, perceptron learning, and a usability case study for text categorization. In: International Conference on Research and Development in Information Retrieval, pp. 67–73. ACM, New York (1997)
15. Cesa-Bianchi, N., Conconi, A., Gentile, C.: A Second-Order Perceptron Algorithm. J. Computing 34(3), 640–668 (2005)
16. Wang, S., San, Y., Wang, S.: An Online Modeling Method Based on Support Vector Machine. In: International Conference on COmputer Science and Software Engineering, pp. 98–101. IEEE Press (2008)
17. Sculley, D., Wachman, G.M.: Relaxed Online SVMs for spam filtering. In: International Conference on Research and Development in Information Retrieval, pp. 415–422. ACM, New York (2007)
18. Dredze, M., Crammer, K., Pereira, F.: Confidence-Weighted Linear Classification. In: International Conference on Machine Learning, pp. 264–271. ACM, New York (2008)
19. Crammer, K., Kulesza, A., Dredze, M.: Adaptive Regularization of Weight Vectors. In: Neural Information Processing Systems. MIT Press, Cambridge (2009)
20. Lin, P., Yen, T., Fu, J., Yu, C.: Analyzing Anomalous Spamming Activities in a Campus Network. In: TANET (2011)