Injecting Non-Uniformity into AdaBoost for Intensive Production on Noisy Data

Arjun Arora

Assistant Professor: Computer Sc & Engg dept.
Dehradun Institute of Technology
Dehradun, India
mail arjunarora06dit@gmail.com

Prof. R P Arora

Professor: Computer Sc. & Engg dept.
Dehradun Institute of Technology
Dehradun, India

Abstract—**AdaBoost is a known learning algorithm that generates frail classifiers serially and then combines them into a firm one . It shows resistance to over-fitting in minor noise data cases ,experiments[2,3] show that it is considerably sensitive to noisy data. Numerous changes to AdaBoost have been proposed to accord with noisy data out of which Bagging and Random forests have shown their resistance to noisy data which could be explained by their non-uniformity. We study on injecting non-uniformity into AdaBoost and propose a method called TRandom-AdaBoost which concerns with AdaBoost on both low and high noise dataset. We presume that the achievement of the method could be explained by using it as a Random Forest with a fairly distributed sequence of weighted base hypotheses where the weights are formed by the technique used in AdaBoost.**

Keywords- AdaBoost; Random forests ;noise; overfitting; intensive; Artificial Intelligence; Machine Learning

I.   INTRODUCTION

The goal of ensemble learning methods is to construct a strong classifier (or hypothesis) which is a collection of classifiers that generated sequentially. Their effectiveness relies on the strength of the base learning algorithm and the ability to generate distribution based on the training set to simulate the real instance space. AdaBoost [1] is a well-known ensemble classifier learning algorithms. It generates weak classifier in each iteration by calling the weak learning algorithm with a training set weighted based on previous classification errors. It increases the weights of examples that the previous classifier did not classify correctly. Due to its weight update technique, AdaBoost tends to assign much higher weight to noisy instances which are inconsistent with the majority of examples and this is more serious in its later iterations [3]. Several previous studies have focused on exploring the production of ensemble methods in the presence of noise. Opitz and Maclin [2] illustrated the overfitting problem of AdaBoost by a simple experiment. Jiang [8] has studied the theoretical aspects of boosting on noisy data. Oza [9] proposed an approach called AveBoost2 to smooth noise. Kalai and Servedio [12] present a new boosting algorithm and prove that it can attain arbitrary

accuracy when classification noise is present. An algorithm, Smooth Boosting [13], is proven to tolerate a combination of classification and feature noise. The data distribution skew is penalized in the learning process to prevent several hardest examples from spoiling decision boundaries in [7]. Some other works focus on moving the noisy data from the training set [10, 11].

II.   RANDOM-ADABOOST

Bagging [6] works by making bootstrap replicate randomly from the training set and using this as a new sub-training set. Random forests [5] are a combination of tree predictors such that each tree depends on the values of a random vector sampled independently and identically distributed for all trees in the forest. Bagging and Random forests have shown their intensity to noise which could be explained partially by the non-uniformity in the weak classifier generating procedure. In this paper we purpose the idea of non-uniformity into AdaBoost and propose a new algorithm, which we call TRandom-AdaBoost. Experiments show that the proposed method is superior to AdaBoost on both low and high noise dataset and is also resist to overfit. We state from an algorithm called Random-AdaBoost, in which the sub-optimal classifier is selected instead of the best one as in the original AdaBoost. The algorithm is proposed below.

- Given Training set (x_m , y_m), m=1,2,….,m. $x_i \in X$

  $y_i \in \{ -1, +1\}$

  Set the Non-uniformity Level r.

  Start with distribution $D_1(i)= 1/m$;

- For t= 1,….T:

  Train weak learner using distribution D_t and get a hypothesis set $\{h_t\}$

  Get a hypothesis randomly from the best r percent of the set: $h_t : X \rightarrow R$

  Choose $\alpha_t \in R$.

Update:

$$D_{t+1}(i) = \frac{D_t(i)\exp\mathbb{C}(-\alpha_t y_i h_t(x_i)}{Z_t}$$

Where Z_t is a normalization factor (chosen such that D_{t+1} will be distribution).

- Output the final hypothesis:

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

ALGORITHM I. RANDOM ADABOOST

The main difference between Random-AdaBoost and the original AdaBoost is the new parameter r which is used to denote non-uniformity level. In each iteration a hypothesis is selected randomly from the best r percent of the hypothesis set. Later, this parameter will be discussed jointly with noise level and the number of the iterations.

Dietterich [3] purposed a method Randomization to grow the tree where at each node the split is selected at random from among the K best splits. This method was used to improve the production of the decision tree algorithm C4.5 on noisy data. Their idea is similar to ours but they didn't purpose the idea of non-uniformity into AdaBoost.

III. EXPERIMENTS

In this part, experiments are implemented on dataset "ionosphere" which is a binary problem with 351 samples and 34 input variables. The Non-uniformity Level, noise level and the number of iterations are discussed jointly. As our goal in this paper is to explore how this simple technique works, stump is used as the weak learning algorithm. The dataset is divided into training set and test set randomly on each running of the algorithm and test errors with fixed parameters are averaged on 100 running of the algorithm. The noisy data is generated by reversing the labels of the instances randomly under the noise level.

A. *Experiments on Noisy Data*

Figure I show the test error of AdaBoost and Random-AdaBoost on noisy data where non-uniformity level is fixed to 30. When the iteration number is small, AdaBoost is generally better than Random-Adaboost. This is easily understood that in such case the strength of the selected weak classifiers play a big part in classification ability and ones selected by AdaBoost are stronger than Random-Adaboost. As the number of iteration increases, the test error of AdaBoost tends to decreases in low noise cases but increases in high noise cases. Random-Adaboost shows similar trend but the test error decreases faster and finally domains AdaBoost on each noise level.
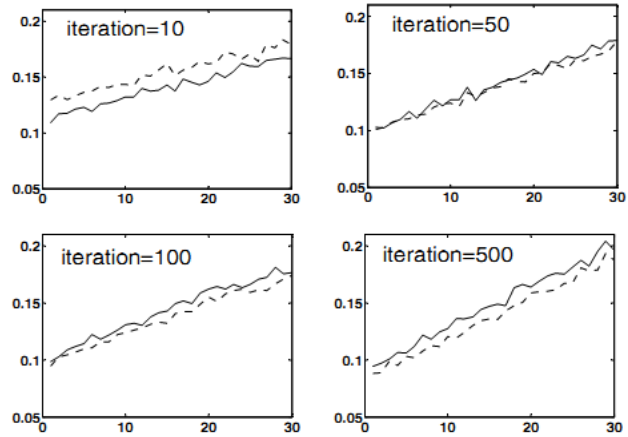

Figure .1   Test error of AdaBoost and Random-AdaBoost, indicated by solid line and dashed line respectively ,on different iteration numbers. The horizontal axis indicates noise level which varies from 0 to 30 and the vertical axis indicates the test error .The randomness level is fixed to 30

B. *TRandom-AdaBoost*

Random forests [5], proposed by Breiman, are a combination of tree predictors in which a random selection of features is used to split each node. Similar to random forests, Random-AdaBoost selects weak classifier totally in random when non-uniformity level is up to 100. But to the different, Random-AdaBoost still remains the weighted ensemble technique to generate a sequence of weights $w_1, w_2, \ldots, w_t$ on the hypotheses.
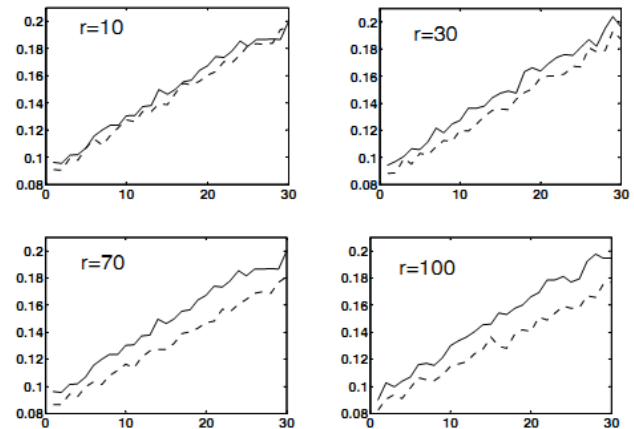

Figure 2.   Comparison of AdaBoost and Random-AdaBoost, indicated by solid line and dashed line respectively ,on different randomness level (indicated by r). The horizontal axis indicates noise level and the vertical axis indicates the test error. The iteration number is fixed to 500.
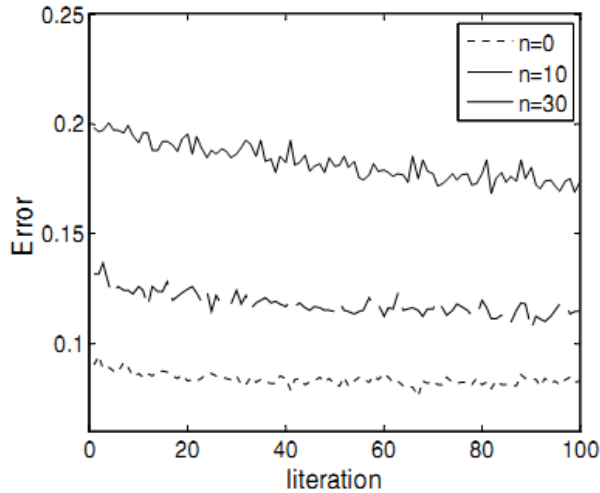
Figure 3. Test error of Random-AdaBoost on Randomness Level from 0 to 100 with different noise levels n=0,10,30, indicated by dotted line, dashed line and solid line respectively. Iteration number is fixed to 500

As Non-uniformity Level increases, the ability of Random-AdaBoost to focus on the examples incorrectly classified is getting down and when non-uniformity level is 100, the weight update technique used in the original AdaBoost no longer works and totally random classifiers are selected instead of the hardest ones. As the strength of the weak classifier decreases we were expect that the test error of the finally combined strong classifier to increase. But to our surprise, the test error gets even lower in such cases as indicated by figure 2 where the gap between AdaBoost and Random-AdaBoost becomes larger as r increases on every noise levels and when the non-uniformity level equals to 100, the Random-AdaBoost domains AdaBoost. This could be revealed more clearly in figure 3 where the test error generally decreases on different noise levels as the Non-uniformity level increases to 100. Notice that When random level equals zero, it is identical to the original AdaBoost.

A totally random version of Random-AdaBoost is proposed below which we call Trandom-AdaBoost. To explain its success, we conjecture that in such case, r=100, the decrease of the test error could be explained by regarding Random-Adaboost as an Random Forest with an ergodic sequence of weighted base hypotheses whose average converges to an optimal function. The weights of the hypotheses w_1, w_2,…, w_t are generated based on the distribution alternating technique used by the original AdaBoost.

- Given Training set (x_m, y_m), m=1,2,…..,m. $x_i \epsilon X$ $y_i \epsilon \{-1,+1\}$
  Start with distribution $D_1(i)= 1/m$;
- For t= 1,….,T:
  A hypothesis is selected at random: h_t: X→R. The error rate of h_t is compared based on D_t . Choose $\alpha_t \epsilon R$ based on the error rate.

Update:

$$D_{t+1}(i) = \frac{D_t(i)\exp(-\alpha_t y_i h_t(x_i))}{Z_t}$$

Where Z_t is a normalization factor (chosen so that D_{t+1} will be a distribution).

- Output the final hypothesis:

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

ALGORITHM II. TRandom-AdaBoost

C. Over fitting

The TRandom-AdaBoost also shows its intensity to overfit on different noise levels. In figure 4, the algorithm runs up to 1000 iterations. As the noise level increases, the test error generally becomes larger. With low noise data (n=0,10), the test error decreases as the iteration number increases. As noise level increases TRandom-AdaBoost seems to overfit a little bit. When n=40, the lowest error is obtained within 100 iterations, after that point, the curve goes upward slightly. But the test error doesn't seem to increase after 500 iterations.
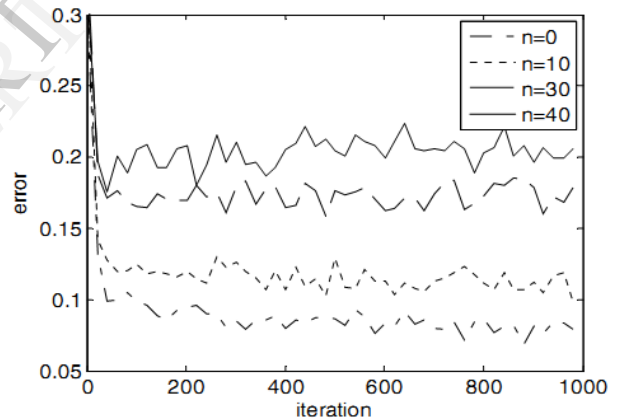

Figure 4. Test error of TRandom-AdaBoost on different noise levels as iteration number increases

IV. CONCLUSION

In this paper, we studied on injecting non-uniformity into AdaBoost and proposed an algorithm which we call TRandom-AdaBoost. Experiments show that the proposed method is superior to AdaBoost on both low and high noisy data and is also resist to overfit. Notice that the TRandom-AdaBoost is also extremely fast as only the test error of the selected hypothesis is need to be calculated instead of all the hypothesis generated by the weak learner. Also more experiments should be implemented, we conjecture that the success of the proposed method could be explained by regarding it as a Random Forest with a sequence of weighted base hypotheses whose average converges to an optimal

function and the weights is generated by the technique used by AdaBoost.

REFERENCES

[1] Y. Freund, R. E. Schapire, Experiments with a new boosting algorithm, Machine Learning: Proceedings of the Thirteenth International Conference, pp. 148–156, 1996.

[2] D. Opitz, R. Maclin, Popular ensemble methods: An empirical study, Journal of Artificial Intelligence Research, vol. 211, pp. 169-198, 1999.

[3] T. Dietterich, An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting and randomization, Machine Learning, vol. 40, pp. 139-158, 2000.

[4] P. Melville, N. Shah, L. Mihalkova, R. Mooney, Experiments on Ensembles with Missing and Noisy Data, Multiple Classifier Systems, vol. 3077, pp. 293-302, 2004.

[5] L. Breiman, Random forests, Machine Learning, vol. 45, page. 5-32, October 2001

[6] L. Breiman, Bagging predictors, Machine Learning, vol.40, page. 123-140, 1996.

[7] Y. J. Sun, S. Todorovic, J .Li, Reducing the overfitting of AdaBoost by controlling its data distribution skewness, International journal of pattern recognition and artifical intelligence, vol.20, pp. 1093-1116, Novemble 2006.

[8] W. Jiang, Some Theoritical aspects of Boosting in the presence of noisy data, Eighteenth Proceeedings of International Conference on Machine Learning, July, 2001.

[9] N. Oza, AveBoost2: Boosting for Noisy Data, Fifth International Workshop on Multiple Classifier Systems, pp.31–40, 2004.

[10] S. Verbaeten, A. Assche, Ensemble methods for noise elimination in classification problems, Department of Computer Science, K.U.Leuven, Report CW 358, Leuven, Belgium, 2004.

[11] V. Wheway, Using Boosting to Detect Noisy Data, PRICAI Workshops, pp.123-132, 2000.

[12] Kalai, R. A. Servedio, Boosting in the presence of noise, Proceedings of the thirty-Fifth Annual ACM Symposium on Theory of Computing, 2003.

[13] R. A. Servedio, Smooth boosting and learning with malicious noise, The Journal of Machine Learning Research, vol. 4, pp. 633–648, 2003.