Landscaping of Random Forests through Controlled Deforestation

Kausik Das IMKAUSIKDAS@GMAIL.COM
Abhijit Guha Roy ABHI4SSJ@GMAIL.COM
Jyotirmoy Chatterjee JCHATTERJEE@SMST.IITKGP.ERNET.IN
Debdoot Sheet DEBDOOT@EE.IITKGP.ERNET.IN
Indian Institute of Technology Kharagpur, WB, 721302

Abstract

Random forest (RF) is an ensemble learner constructed using a set of decision trees, where each tree is trained using randomly bootstrapped samples and aggregated to provide a decision. While the generalization error is reduced by increasing the number of trees in a RF, it substantially increases the testing time complexity, inhibiting its fast deployment in practical applications. In this paper, we propose a post-training optimization technique termed landscaping of RF for reducing computational complexity by compensating for trees associated with similar decision boundary. This allows faster deployment of the RF without compromising its performance. Landscaping is achieved through a two stage mechanism: (i) computation of decision similarity between all pairs of trees in the RF, and (ii) deletion of the computationally expensive tree in the RF with decision bias compensation for the removed tree. Performance of the proposed methodology was evaluated using three publicly available datasets. The RF performance before and after landscaping over the datasets was observed to have an error of 0.1084 ± 0.03 and 0.1087 ± 0.03, respectively, while testing times of the RF before landscaping was 2.5508 ± 0.08 sec. and 0.9066 ± 0.19 sec. after landscaping with $32 - 76\%$ reduction in execution time. These results strongly substantiates our claim of achieving deployment speedup without compromising the decision quality with landscaping of RF through controlled deforestation.

Appearing in Proceedings of the 2^{nd} Indian Workshop on Machine Learning, IIT Kanpur, India, 2016.

1. Introduction

Random forest (RF) is an ensemble learner which consists of a set of independently learned decision trees and is popularly used to solve classification and regression tasks across machine vision viz. image classification (Bosch et al., 2007), segmentation (Yu et al., 2011), etc. The salient feature of RF which provides it an edge compared to other shallow learning systems (Fernández-Delgado et al., 2014) is its ability to generalize over the feature space using limited training samples. Despite the increase in performance, increased number of trees in the RF restricts its deployment in real-time application due to increased computational complexity. In this paper, we present and evaluate a method for post-training refinement of a RF by removing trees with redundant contribution to the decision boundary (termed as *controlled deforestation*), while retaining the remaining ones to form a new *landscaped* RF (LRF) for fast deployment with reduced computational complexity while preserving decision making performance.

Related Work Generally for solving complex real world problems, trees with higher depths are required (Gall & Lempitsky, 2013; Dollár & Zitnick, 2013). Since the computational complexity at deployment is dependent on depth of the tree, there have been some approaches to reduce the tree size via pruning. Decision jungles is an alternative where merging of nodes with similar class *posteriori* distribution is used to reduce the number of nodes to be evaluated thus leading to reduced computational complexity without compromising on the decision boundary. An alternative route has been through removal of trivial trees in the RF viz. by removing most correlated trees, or sequential incremental selection of bag of trees to form subforest (Bernard et al., 2009).

Challenge Considering a fully grown RF with N trees, the number of possible sub-forests to search from is $\sum_{i=1}^{N} \binom{N}{i} = 2^N - 1$. Considering a RF with 50 trees,

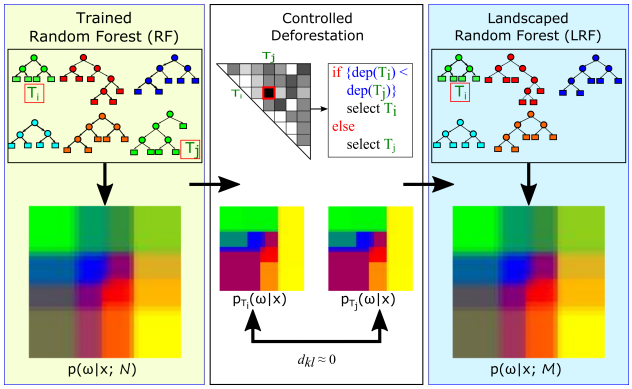

Figure 1. The overall process of the landscaping of RF is illustrated in the figure above. The trained RF is used to create a similarity matrix (d_{kl}) for controlled deforestation. The tree pair is selected corresponding to lowest score. This corresponds to functionally similar trees as shown. The tree with higher depth is deleted. This process is repeated to get the landscaped RF without hampering the overall performance as shown.

this equates to searching over 1.13×10^{15} possible subforests to find an optimal configuration. The challenge posed is to select an optimal configuration of a smaller sized RF with substantially lower search time complexity, preferably lesser then the current solutions which depend on an order exponentially on the number of trees in the RF.

Approach In this paper we proposed a scheme for removal of redundant set of trees in the RF through the process of *controlled deforestation*. It is a two stage process based on (i) similarity scoring between the pairs of trees in a RF based on the statistical distance of their *posteriori* decision boundary, and (ii) sequential removal of the computationally expensive redundant tree in a similar pair along with compensation of the bias associated with this process as summarized in Fig. 1. The rest of the paper is organized accordingly with problem statement introduced in Sec. 2 and exposition to the solution detailed in Sec. 3. The experiments with results and discussions are presented in Sec. 4. Finally, Sec. 5 concludes the work.

2. Problem Statement

Let us consider a data set represented as $\mathcal{Q} = \{(\mathbf{X}_1, Y_1), (\mathbf{X}_2, Y_2), \ldots, (\mathbf{X}_L, Y_L)\}$, where $\mathbf{X}_l \in \mathbb{R}^D$ is a D-dimensional feature vector representing the l^{th} observation sample and $Y_l \in \Omega = \{\omega_1, \omega_2, \ldots, \omega_K\}$ is the class label associated with this observation. Let $\mathbf{x} \in \mathbb{R}^D$ be a random variable (r.v.), such that given a value of $\mathbf{x}$, the decision making task is defined as computing the *posteriori* probability $p(\omega_k|\mathbf{x}) = \mathcal{H}(\omega_k|\mathbf{x}; \{\mathcal{Q}\}_{\text{train}}, N)$ using a RF $\mathcal{H}(\cdot)$. Here the set $\{\mathcal{Q}\}_{\text{train}}$ consists of a few samples drawn randomly from $\mathcal{Q}$ such that $|\{\mathcal{Q}\}_{\text{train}}| \leq |\mathcal{Q}|$ and is generally referred to as the training set, $|\cdot|$ is the set cardinality operator; N is the number of trees in the RF such that $\mathcal{H}(\cdot) = \{\mathcal{T}_1(\cdot), \mathcal{T}_2(\cdot), \ldots, \mathcal{T}_N(\cdot)\}$ and the *posteriori* estimate of the RF is computed as

$$p(\omega_k|\mathbf{x}) = \frac{1}{N} \sum_{i=1}^{N} p_{\mathcal{T}_i}(\omega_k|\mathbf{x}) \tag{1}$$

where $p_{\mathcal{T}_i}(\omega_k|\mathbf{x})$ is the *posteriori* estimate of $\mathcal{T}_i(\cdot)$. The process of *controlled deforestation* (CD) is formally defined as a transformation

$$\mathcal{H}(\omega_k|\mathbf{x}; \{\mathcal{Q}\}_{\text{train}}, N) \overset{\text{CD}}{\rightarrow} \mathcal{H}_{LRF}(\omega_k|\mathbf{x}; \{\mathcal{Q}\}_{\text{train}}, N \rightarrow M) \tag{2}$$

where $\mathcal{H}_{LRF}(\cdot)$ is the *landscaped* RF (LRF) obtained from the trained RF $\mathcal{H}(\cdot)$, consists of M trees such that $M \leq N$, and CD is performed under the constraint that

$$\begin{aligned} p(\omega_k|\mathbf{x}) &= \mathcal{H}(\omega_k|\mathbf{x}; \{\mathcal{Q}\}_{\text{train}}, N) \\ &\simeq \mathcal{H}_{LRF}(\omega_k|\mathbf{x}; \{\mathcal{Q}\}_{\text{train}}, N \rightarrow M) \end{aligned} \tag{3}$$

and our approach for achieving this is subsequently described.

3. Exposition to the Solution

We achieve this objective of reducing the computational complexity of a RF through a two stage process: (i) *Controlled deforestation*: similarity scoring between the pairs of trees in a RF based on the statistical distance of their *posteriori* decision boundary, and (ii) *Landscaping of random forests*: sequential removal of the computationally expensive redundant tree in a similar pair along with compensation of the bias associated with this process.

An RF model $\mathcal{H}(\cdot) = \{\mathcal{T}_1(\cdot), \mathcal{T}_2(\cdot), \ldots, \mathcal{T}_N(\cdot)\}$ is formally constructed to have N number of trees. The number of trees is generally empirically defined at the start of the process. The *posteriori* estimate of each tree in the RF is represented as

$$p_{\mathcal{T}_i}(\omega_k|\mathbf{x}) = \mathcal{T}_i(\omega_k|\mathbf{x}; \mathcal{Q}_i, \mathbf{\Theta}_i) \tag{4}$$

where $\mathcal{Q}_i \subset \{\mathcal{Q}\}_{\text{train}}$ is a set of samples bootstrapped from the training set $\{\mathcal{Q}\}_{\text{train}}$ such that:

1. $\cup\{\mathcal{Q}_i\} = \{\mathcal{Q}\}_{\text{train}}$,
2. $\mathcal{Q}_i \cap \mathcal{Q}_j \neq \emptyset$, and

3. $\{(\mathbf{x}, y)\}$ in $\mathcal{Q}_i$ are independently drawn and identically distributed in $\{\mathcal{Q}\}_{\text{train}}$.

The vector $\mathbf{\Theta}_i$ defines the ordered set of split functions at each non-leaf node in $\mathcal{T}_i$; $\mathbf{\Theta}_i$ and $\mathbf{\Theta}_j$ are not necessarily related since they are obtained by training $\mathcal{T}_i$ and $\mathcal{T}_j$ on $\mathcal{Q}_i$ and $\mathcal{Q}_j$ respectively (**?**).

3.1. Controlled Deforestation of Random Forests

CD which has been formally defined in (2) involves identification of redundant trees in the RF. Let the r.v. $\mathbf{x} = \{x^{(1)}, x^{(2)}, \ldots, x^{(d)}, \ldots, x^{(D)}\} \in \mathcal{Q}$ have a dynamic range in $\mathcal{Q}$ for each entry such that $x^{(d)} \in [x^{(d)}_{\min}, x^{(d)}_{\max}]$ and thus $\mathbf{x}$ is any random location in the space $\mathcal{S} \subset \mathbb{R}^D$ where $\mathcal{S} = [x^{(1)}_{\min}, x^{(1)}_{\max}] \times [x^{(2)}_{\min}, x^{(2)}_{\max}] \times \cdots \times [x^{(D)}_{\min}, x^{(D)}_{\max}]$. The *posteriori* decision boundary of $\mathcal{T}_i$ is thus defined as

$$\mathcal{P}(\mathcal{T}_i) \supset \{p_{\mathcal{T}_i}(\omega_k|\mathbf{x}) \forall \mathbf{x} \in \mathcal{S}, \omega_k \in \Omega\} \tag{5}$$

Accordingly, the statistical distance between a pair of trees $\mathcal{T}_i(\cdot)$ and $\mathcal{T}_j(\cdot)$ s.t. $(\mathcal{T}_i(\cdot), \mathcal{T}_j(\cdot)) \in \mathcal{H}(\cdot)$ and $i \neq j$ can be obtained as

$$\begin{aligned} D\left(\mathcal{T}_i(\cdot), \mathcal{T}_j(\cdot)\right) &= d_{JS}\left(\mathcal{P}(\mathcal{T}_i), \mathcal{P}(\mathcal{T}_j)\right) \\ &= \sum_{\omega_k \in \Omega} d_{JS}\left(p_{\mathcal{T}_i}(\omega_k|\mathbf{x}), p_{\mathcal{T}_j}(\omega_k|\mathbf{x})\right) \end{aligned} \tag{6}$$

where, $d_{JS}(\cdot)$ is the Jensen-Shannon statistical distance metric (Jianhua, 1991), and $D\left(\mathcal{T}_i(\cdot), \mathcal{T}_j(\cdot)\right)$ are the $\binom{N}{i}$ number of valid entries in the matrix $\mathbf{D}$ with $|\mathbf{D}| = N \times N$.

This pairwise distance indicated in $\mathbf{D}$ is the key to identification of redundant trees associated with similar decision boundary in the RF. The method of CD employs identification of such pairwise redundancy, associated with minimum bin value in $\mathbf{D}$, for substituting the pair by a single tree for computational complexity reduction for creating the LRF as detailed next.

3.2. Landscaping of Random Forests

In order to implement the transformation in (2), the tree (say $\mathcal{T}_i(\cdot)$) associated with a lower depth is preserved, in a pair of trees $(\mathcal{T}_i(\cdot), \mathcal{T}_j(\cdot))$ identified with the minimum bin entry in $\mathbf{D}$, and the tree in the pair with higher depth (say $\mathcal{T}_j(\cdot)$) is removed. This is on account of time complexity in prediction through a tree being proportional to its depth (Moshkov & Mikhail, 2005).

However, removal of a tree from the RF modifies the nature of its decision boundary, and thus to preserve the constraint in (3) we introduce the following bias compensation

$$\begin{aligned} \sum_{\forall \mathcal{T}_n \neq \mathcal{T}_i, \mathcal{T}_j} & p_{\mathcal{T}_n}(\omega_k|\mathbf{x}) + p_{\mathcal{T}_i}(\omega_k|\mathbf{x}) + p_{\mathcal{T}_j}(\omega_k|\mathbf{x}) \\ & \simeq \sum_{\forall \mathcal{T}_n \neq \mathcal{T}_i, \mathcal{T}_j} p_{\mathcal{T}_n}(\omega_k|\mathbf{x}) + 2p_{\mathcal{T}_i}(\omega_k|\mathbf{x}) \end{aligned} \tag{7}$$

where the left sided term is $\mathcal{H}(\omega_k|\mathbf{x}; \{\mathcal{Q}\}_{\text{train}}, N)$ and the right sided term is $\mathcal{H}_{LRF}(\omega_k|\mathbf{x}; \{\mathcal{Q}\}_{\text{train}}, N \to N-1)$ as per (3), and $\mathcal{H}_{LRF}(\cdot)$ consists of $N-1$ number of trees.

The process is subsequently repeated by recomputing the matrix $\mathbf{D}$ on this LRF for the remaining $(N-M-1)$ number of times. Since computing $\mathbf{D}$ is an exhaustive process, the stage can be avoided by removing the rows and column in $\mathbf{D}$ corresponding to the entry $\mathcal{T}_j$ to obtain a $(N-1)\times(N-1)$ sized matrix and repeat the search process to the next redundant pair to be removed. The *posteriori* of a LRF can hence be represented by modifying (1) as

$$p(\omega_k|\mathbf{x}) = \sum_{i=1}^{N} p(\mathcal{T}_i) p_{\mathcal{T}_i}(\omega_k|\mathbf{x}) \tag{8}$$

where $p(\mathcal{T}_i)$ is the *priori* of each tree to a decision making, s.t. in a RF with N trees, $p(\mathcal{T}_i) = \frac{1}{N}$ as in (1) and in the LRF, $p(\mathcal{T}_i) = \frac{w_i}{N}$ where w_i is the redundancy number of $\mathcal{T}_i$, which is computed as the number of trees $\{\mathcal{T}_j\} \forall i \neq j$ in the original RF in $\mathcal{H}(\cdot)$ which were similar to $\mathcal{T}_i$ in $\mathcal{H}_{LRF}(\cdot)$.

4. Experimental Results - Discussions

We experimentally demonstrate the potential of LRF through CD using the following three classification datasets: (i) synthetic dataset and (ii) skin segmentation dataset, (iii) ADL dataset from the UCI ML repository[1]. The synthetic dataset was self generated. We randomly select 50% of the samples for training N trees of a RF and 25% for discovering the LRF with M number of trees, from each of these datasets. Remaining 25% of data is used for validating our proposed method. We construct the *posteriori* as $\mathcal{P}(\mathcal{T}_i)$ in (5) for each tree $\mathcal{T}_i$ in the RF $\mathcal{H}(\cdot)$ by uniformly shattering the dynamic range $[x^{(d)}_{\min}, x^{(d)}_{\max}]$ of feature $x^{(d)}$ into 100 discrete steps over all $x^{(d)} \in \mathbf{x}$.

The RF in our experiment consisted of $N = 100$ number of trees. Bootstrap factor for $\mathcal{Q}_i \in \{\mathcal{Q}\}_{\text{train}}$ is chosen to be 2%. As a stopping criteria of growth for each tree we set $minLeaf = 10$. The testing time of

[1] https://archive.ics.uci.edu/ml/datasets.html

Table 1. Comparison of the time complexity of RF before and after landscaping for different datasets.

DATASET	RF (SEC.)	LRF (SEC.)	Δ TIME
SYNTHETIC	1.2814 ± 0.27	0.8748 ± 0.18	-32%
SKIN	5.0184 ± 0.02	1.2237 ± 0.02	-76%
ADL	1.3526 ± 0.03	0.6212 ± 0.02	-54%

Table 2. Comparison of average classification error of RF before and after landscaping for different datasets.

DATASET	RF	LRF
SYNTHETIC	0.0582 ± 0.07	0.0585 ± 0.07
SKIN	0.0137 ± 0.09	0.0128 ± 0.08
ADL	0.2533 ± 0.09	0.2549 ± 0.17

the RF and LRF were computed for three different datasets across 15 runs of independent experiments and is reported in Table 1. The performance of the RF and LRF were computed in terms of percentage of error in decision making and reported in Table 2.

Close observation of the Fig. 2 where error vs. number of removed trees is presented, shows that initially the error is constant even after removal of significant number of trees. This corresponds to the deletion of redundant trees. The execution speedup is proportional to the number of redundant trees that can be removed in the due process. In Table 1 we can observe a $32 - 76\%$ reduction in time complexity of the LRF compared to the RF for other datasets.

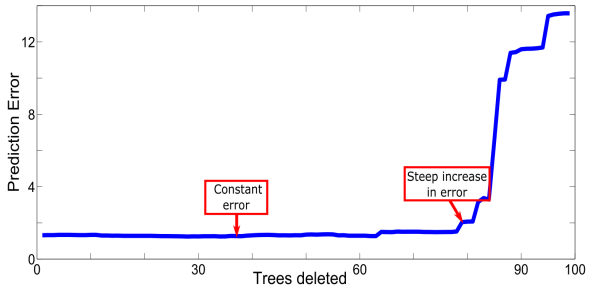

Figure 2. Illustration of the change in performance error as the trees are removed systematically for LRF through CD.

5. Conclusion

In this paper, we have proposed a method for reducing the computational complexity of RF during deployment through a post training refinement (landscaping) framework. The main features of this presented framework are (i) notable increase in execution speed and (ii) performance retention of the LRF after CD. This is done by proper identification of redundant trees which are highly correlated with others and selective removal in the forest followed with bias compensation. This increases the speed while preserving the functionality of the learned framework. We experimentally verified our claims for three different datasets which substantiates our claims with exhibiting speedup with $32 - 76\%$ reduction in execution time of the LRF while preserving overall classification performance.

Dual Submission

This paper has been accepted at the *22nd National Conference on Communications, 2016, IIT Guwahati, Assam.* url: http://www.iitg.ernet.in/ncc2016/

References

Bernard, S., Heutte, L. and Adam,S. *Towards a better understanding of random forests through the study of strength and correlation*, *Emerging Intelligent Comput. Tech., Appl.*, pp. 536–545, 2009

Bosch, A., Zisserman, A. and Munoz,X. *Image classification using random forests and ferns*, *Prof. Int. Conf. Comp. Vis.*, pp. 1–8, 2007

Dollár, P. and Zitnick, C.L. *Structured forests for fast edge detection*, *Proc. Int. Conf. Comp. Vis.*, pp.1841–1848, 2013.

Fernández-Delgado, M., Cernadas, E., Barro, S. and Amorim,D. *Do we need hundreds of classifiers to solve real world classification problems?*, *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 3133–3181, 2014

Gall, J. and Lempitsky, V. *Class-specific hough forests for object detection*, *Decision forests for computer vision and medical image analysis*, pp.143–157, 2013.

Jianhua, L. *Divergence measures based on the Shannon entropy*, *IEEE Trans. Inf. Th.* vol. 37, no. 1, pp. 145-151, 1991.

Moshkov and Mikhail, J. *Time complexity of decision trees*, *Trans. Rough Sets III*, pp. 244–459, 2005.

Yu, G., Goussies, N., Yuan, J., and Liu,Z. *Fast action detection via discriminative random forest voting and top-k subvolume search*, *IEEE Trans. Multimedia*, vol. 13, no. 3,pp. 507–517, 2011