Abstract

Decision tree boosting is considered as an important and widely recognized method in image classification, despite dominance of the deep learning based approaches in this area. Provided with good image features, it can produce a powerful model with unique properties, such as strong predictive power, scalability, interpretability, etc. In this paper, we propose a novel tree boosting framework which capitalizes on the idea of using shallow, sparse and yet powerful oblique decision trees (trained with recently proposed \textit{Tree Alternating Optimization} algorithm) as the base learners. We empirically show that the resulting model achieves better or comparable performance (both in terms of accuracy and model size) against established boosting algorithms such as gradient boosting or AdaBoost in number of benchmarks. Further, we show that such trees can directly and efficiently handle multiclass problems without using one-vs-all strategy employed by most of the practical boosting implementations.

Work supported by NSF award IIS–2007147

AdaBoost and Decision Trees

Ensembles of decision trees have numerous applications in many tasks in image processing and computer vision. Among others, AdaBoost with decision tree base learners is often referred to as the best off-the-shelf classifier for various problems.

• Many papers and current implementations of AdaBoost use trees that are:
  • Axis-aligned (i.e. it uses a single feature at a decision node)
  • Trained with greedy top-down induction
  • However, axis-aligned trees are not very suitable for many problems, especially for the ones with correlated features (e.g. pixels of an image).
  • Greedy top-down induction produces suboptimal trees.
  • Because of these, boosting algorithms usually need to induce \( K \) (number of classes) such trees at each boosting step, which adds an extra overhead.
  • And, to find a suitable splitting criterion for specific objective functions (as is the case with many boosting algorithms) is not straightforward with these greedy algorithms.

Optimizing trees in AdaBoost.MH with TAO

Our proposal:

• to use oblique decision trees (i.e. trees with hyperplane splits at decision nodes)
• to use a non-greedy optimization algorithm to learn such trees

We adapt the recently proposed algorithm for learning decision trees, \textit{Tree Alternating Optimization (TAO)} for AdaBoost.MH and empirically evaluate its performance on image classification datasets.

Given a tree structure \( T \), a training set \( \{(x_n, y_n)\}_{n=1}^{N} \), and AdaBoost.MH weights \( \{w_n\}_{n=1}^{N} \), TAO directly optimizes the following base learner’s objective function with a guarantee of monotonically decreasing it:

\[
E(\Theta) = \sum_{n=1}^{N} \sum_{k=1}^{K} w_n \exp(-y_n \cdot T_k(x_n))
\]

Previous implementations of AdaBoost.MH used \( K \) (number of classes) trees at each boosting step. In this work we only fit a single oblique decision tree trained with TAO.

Algorithm 1: Learning a tree for AdaBoost.MH with TAO

input training set \( \{(x_n, y_n)\}_{n=1}^{N} \),
initial tree \( T_0(\cdot; \Theta) \) of depth \( \Delta \),
AdaBoost.MH weights \( \{w_n\}_{n=1}^{N} \),
repeat
for depth \( d = 0 \) to \( \Delta \) do
  for \( i \in \text{nodes at depth } d \) do
    if \( i \) is a leaf then
      \( \theta_i \leftarrow \text{find a constant vector to minimize eq. 1} \)
      (has a closed form solution);
    else
      \( \theta_i \leftarrow \text{fit a weighted binary classifier} \);
    end
  end
until convergence occurs or max iteration;
return trained tree \( T \)

Experiments

We experimentally evaluate AdaBoost.MH with TAO trees against established forest-based methods. Boosted TAO trees are smaller (fewer and shallower trees) yet consistently more accurate.