

# Online Boosting for Anytime Transfer and Multitask Learning

## Supplementary Materials

**Boyu Wang and Joelle Pineau**

School of Computer Science  
McGill University, Montreal, Canada  
boyu.wang@mail.mcgill.ca, jpineau@cs.mcgill.ca

### Proof of Theorem 1

Let  $D_m^b$  be the weight distribution of batch TrAdaBoost algorithm,  $D_m^o$  be the weight distribution of OTB, which can be viewed as the normalized version of Poisson parameter  $\lambda$  in Algorithm 2. Lemma 1 shows the convergence property of  $D_m^o$ .

**Lemma 1.** *As  $N_S \rightarrow \infty$  and  $N_T \rightarrow \infty$ ,  $D_1^o \xrightarrow{P} D_1^b$ .*

Define  $h_m^b$  as the  $m$ th base learner of batch TrAdaBoost, and  $h_m^o$  as the analogous base learner of OTB. Lemma 2 states that if the weight vector  $D_m^o$  converges to  $D_m^b$ , the base learner  $h_m^o$  also converges to  $h_m^b$ .

**Lemma 2.** *If  $D_m^o \xrightarrow{P} D_m^b$ , and the base learners are naive Bayes classifiers, then  $h_m^o \xrightarrow{P} h_m^b$ .*

Let  $\epsilon_{S,m}^b = \sum_{x_n \in S_S} D_m^b(n) I(h_m^b(x_n) \neq y_n)$ ,  $\epsilon_{T,m}^b = \sum_{x_n \in S_T} D_m^b(n) I(h_m^b(x_n) \neq y_n)$ ,  $D_{T,m}^b = \sum_{x_n \in S_T} D_m^b(n)$ ; and  $\epsilon_{S,m}^o, \epsilon_{T,m}^o, D_{T,m}^o$  be their online approximation defined in line 22-24 of Algorithm 2. Lemma 3 states that  $\epsilon_{S,m}^o, \epsilon_{T,m}^o, D_{T,m}^o$  also converge to their batch counterparts given  $h_m^o$  converging to  $h_m^b$ .

**Lemma 3.** *If  $D_m^o \xrightarrow{P} D_m^b$ ,  $h_m^o \xrightarrow{P} h_m^b$ , and the base learners are naive Bayes classifiers, then  $\epsilon_{S,m}^o \xrightarrow{P} \epsilon_{S,m}^b$ ,  $\epsilon_{T,m}^o \xrightarrow{P} \epsilon_{T,m}^b$ , and  $D_{T,m}^o \xrightarrow{P} D_{T,m}^b$ .*

To prove the convergence of the ensemble of classifiers, we also need Lemma 4.

**Lemma 4.** *If  $X_1, X_2, \dots$  and  $X$  are discrete random variables and  $X_n \xrightarrow{P} X$ , then  $I(X_n = x) \xrightarrow{P} I(X = x)$  for all possible values  $x$ .*

We omit the proofs of these lemmas since they follow quite readily from Theorem in (Oza and Russell 2001), Lemma 2, Lemma 8, Lemma 9, and Lemma 4 in (Oza 2001). We only give the proof of the main theorem.

**Theorem 1.** *As  $N_S \rightarrow \infty$  and  $N_T \rightarrow \infty$ , if the base learners are naive Bayes classifiers, OTB converges to batch TrAdaBoost algorithm.*

*Sketch of the Proof.* The convergence of OTB can be proved by induction. For the first base learner, we have  $D_1^o \xrightarrow{P} D_1^b$

by Lemma 1. Then by Lemma 2 and Lemma 3, we have  $h_1^o \xrightarrow{P} h_1^b$ ,  $\epsilon_{S,1}^o \xrightarrow{P} \epsilon_{S,1}^b$ ,  $\epsilon_{T,1}^o \xrightarrow{P} \epsilon_{T,1}^b$ , and  $D_{T,1}^o \xrightarrow{P} D_{T,1}^b$ , which completes the proof of the base case.

Now suppose we have  $D_m^o \xrightarrow{P} D_m^b$ , we need to prove  $D_{m+1}^o \xrightarrow{P} D_{m+1}^b$ , which can be shown as follow.

Note that  $D_m^o(n)$  is normalized version of the Poisson parameter  $\lambda$  of the  $n$ th sample of online data stream. Therefore, by (2) and (3) in *Algorithm Outline* section, we have

$$D_{m+1}^o(n) = \begin{cases} \frac{D_m^o(n)}{1 + D_{T,m}^o - (1-\beta)\epsilon_{S,m}^o - 2\epsilon_{T,m}^o}, & h_m^o(x_n) = y_n \\ \frac{\beta D_m^o(n)}{1 + D_{T,m}^o - (1-\beta)\epsilon_{S,m}^o - 2\epsilon_{T,m}^o}, & h_m^o(x_n) \neq y_n \end{cases}$$

for a sample from source domain, and

$$D_{m+1}^o(n) = \begin{cases} \frac{D_m^o(n)}{1 + D_{T,m}^o - (1-\beta)\epsilon_{S,m}^o - 2\epsilon_{T,m}^o}, & h_m^o(x_n) = y_n \\ \frac{D_m^o(n)(D_{T,m}^o - \epsilon_{T,m}^o)}{\epsilon_{T,m}(1 + D_{T,m}^o - (1-\beta)\epsilon_{S,m}^o - 2\epsilon_{T,m}^o)}, & h_m^o(x_n) \neq y_n \end{cases}$$

for a sample from target domain. It can be verified that this weight update mechanism is identical to the distribution update step of batch TrAdaBoost (line 7 and line 9 of Algorithm 1). By the assumption  $D_m^o \xrightarrow{P} D_m^b$ , we have  $h_m^o \xrightarrow{P} h_m^b$  (Lemma 2),  $\epsilon_{S,m}^o \xrightarrow{P} \epsilon_{S,m}^b$ ,  $\epsilon_{T,m}^o \xrightarrow{P} \epsilon_{T,m}^b$ , and  $D_{T,m}^o \xrightarrow{P} D_{T,m}^b$  (Lemma 3). Also, note that both  $D_{m+1}^o(n)$  and  $D_{m+1}^b(n)$  are continuous functions of these convergent quantities, we have  $D_{m+1}^o(n) \xrightarrow{P} D_{m+1}^b(n)$ . Again, by Lemma 2 and Lemma 3, we have  $h_{m+1,N}^o \xrightarrow{P} h_{m+1,N}^b$ ,  $\epsilon_{S,m+1}^o \xrightarrow{P} \epsilon_{S,m+1}^b$ ,  $\epsilon_{T,m+1}^o \xrightarrow{P} \epsilon_{T,m+1}^b$ , and  $D_{T,m+1}^o \xrightarrow{P} D_{T,m+1}^b$ , which implies that all of the base learners returned by OTB converges to that returned by batch TrAdaBoost. By Lemma 4, we have  $\sum_{m=\lceil \frac{1}{M/2} \rceil}^M \log(\frac{1-\epsilon_{T,m}^o}{\epsilon_{T,m}^o}) I(h_m^o(x) = y) \xrightarrow{P} \sum_{m=\lceil \frac{1}{M/2} \rceil}^M \log(\frac{1-\epsilon_{T,m}^b}{\epsilon_{T,m}^b}) I(h_m^b(x) = y)$ , which implies  $H^o \xrightarrow{P} H^b$ .  $\square$

### References

- Oza, N. C., and Russell, S. 2001. Online bagging and boosting. In *AISTATS*, 105–112.
- Oza, N. C. 2001. *Online Ensemble Learning*. Ph.D. Dissertation, University of California, Berkeley.