Multiview Boosting by Controlling the Diversity and the Accuracy of View-specific Voters

Anil Goyal, Emilie Morvant, Pascal Germain, Massih-Reza Amini

Multiview Boosting by Controlling the Diversity and the Accuracy of View-specific Voters

Anil Goyal[a,b], Emilie Morvant[a], Pascal Germain[c], Massih-Reza Amini[b]

[a] *Univ Lyon, UJM-Saint-Etienne, CNRS, Institut d'Optique Graduate School, Laboratoire Hubert Curien UMR 5516, F-42023, Saint-Etienne, France*
[b] *Univ. Grenoble Alps, Laboratoire d'Informatique de Grenoble, AMA, Centre Equation 4, BP 53, F-38041 Grenoble Cedex 9, France*
[c] *Inria Lille - Nord Europe, Modal Project-Team, 59650 Villeneuve d'Ascq, France*

Abstract

In this paper we propose a boosting based multiview learning algorithm, referred as `PB-MVBoost`, which iteratively learns *i)* weights over view-specific voters capturing view-specific information, and *ii)* weights over views by optimizing a PAC-Bayes multiview C-Bound that takes into account the accuracy of view-specific classifiers and the diversity between the views. We derive a generalization bound for this strategy following the PAC-Bayes theory which is a suitable tool to deal with models expressed as weighted combination over a set of voters. Different experiments on three publicly available datasets show the efficiency of the proposed approach with respect to state-of-art models.

Keywords: Multiview Learning, PAC-Bayes, Boosting

1. Introduction

With the tremendous generation of data, there are more and more situations where observations are described by more than one view. This is for example the case with multilingual documents that convey the same information in different languages or images that are naturally described according to different set of features (for example SIFT, HOG, CNN, etc). In this paper, we study the related machine learning problem that consists in finding an efficient classification model from different information sources that describe the observations. This topic, called multiview (or multimodal) learning [1, 2, 3], has been expanding over

the past decade, spurred by the seminal work of Blum and Mitchell on co-training [4] (with only two views). The aim is to learn a classifier which performs better than classifiers trained over each view separately (called here view-specific classifier). Usually, this is done by directly concatenating the representations (early fusion) or by combining the predictions of view-specific classifiers (late fusion) [5]. In this work, we stand in the latter situation. Concretely, we study a two-level multiview learning strategy based on the PAC-Bayesian theory (introduced by McAllester [6] for monoview learning). This theory provides Probably Approximately Correct (PAC) generalization guarantees for models expressed as a weighted combination over a set of functions/voters (*i.e.*, for a weighted majority vote). In this framework, given a *prior* distribution over a set of functions, called voters, $\mathcal{H}$ and a learning sample, one aims at learning a *posterior* distribution over $\mathcal{H}$ leading to a well-performing majority vote; each voter from $\mathcal{H}$ is weighted by its probability to appear according to the posterior distribution. Note that, PAC-Bayesian studies have not only been conducted to characterize the error of such weighted majority votes [7, 8, 9, 10, 11], but have also been used to derive theoretically grounded learning algorithms (such as for supervised learning [10, 12, 13, 14, 15] or transfer learning [16]). To tackle multiview learning in a PAC-Bayesian fashion, we propose to define a two-level hierarchy of prior and posterior distributions over the views: *i)* for each view v, we consider a prior P_v and a posterior Q_v distributions over view-specific voters to capture view-specific information and *ii)* a hyper-prior π_v and a hyper-posterior ρ_v distributions over the set of views to capture the accuracy of view-specific classifiers and diversity between the views (see Figure 1). Following this distributions' hierarchy, we define a multiview majority vote classifier where the view-specific classifiers are weighted according to posterior and hyper-posterior distributions. By doing so, we extend the classical PAC-Bayesian theory to multiview learning with more than two views and derive a PAC-Bayesian generalization bound for our multiview majority vote classifier.

From a practical point of view, we design an algorithm based on the idea of boosting [17, 18, 19, 20], an ensemble method well known to be able to

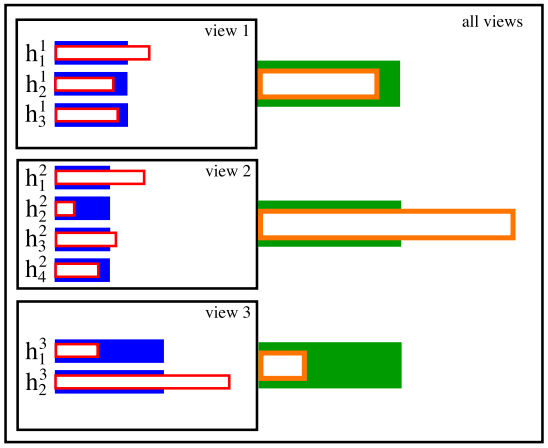


Figure 1: Illustration of the multiview distributions hierarchy with 3 views. For all views $v \in \{1, 2, 3\}$, we have a set of voters $\mathcal{H}_v = \{h_1^v, \ldots, h_{n_v}^v\}$ on which we consider prior P_v view-specific distribution (in blue), and we consider a hyper-prior π distribution (in green) over the set of 3 views. The objective is to learn a posterior Q_v (in red) view-specific distributions and a hyper-posterior ρ distribution (in orange) leading to a good model. The length of a rectangle represents the weight (or probability) assigned to a voter or a view.

learn well-performing majority vote. Our boosting-based multiview learning algorithm, called `PB-MVBoost`, deals with the two-level hierarchical learning strategy. `PB-MVBoost` is then an ensemble method that outputs a multiview classifier expressed as a weighted combination of view-specific voters. It is important to notice that controlling the diversity between the view-specific classifiers or the views is a key element in multiview learning [21, 22, 23, 24, 25, 15]. Therefore, to learn the weights over the views, we minimize an upper-bound on the error of the majority vote, called the multiview C-bound [11, 14, 22], allowing us to control a trade-off between accuracy and diversity. Concretely, at each iteration of our multiview algorithm, we learn *i)* weights over view-specific voters based on their ability to deal with examples on the corresponding view (capturing view-specific information), and *ii)* weights over views by minimizing the multiview C-bound. To show the potential of our algorithm, we empirically evaluate our approach on $\texttt{MNIST}_1$, $\texttt{MNIST}_2$ and `Reuters` RCV1/RCV2 collections[26, 21]. We observe that our algorithm `PB-MVBoost`, empirically minimizes the multiview

C-Bound over iterations, and leads to good performances even when the classes are unbalanced. We compare PB-MVBoost with a previously developed multiview algorithm, denoted by $\texttt{Fusion}^{\texttt{Cq}}_{\texttt{all}}$ [22], which first learns the view-specific voters at the base level of the hierarchy, and then, combines the predictions of view-specific voters using a PAC-Bayesian algorithm CqBoost [14]. From the experimental results, it came out that PB-MVBoost is more stable across different datasets and computationally faster than $\texttt{Fusion}^{\texttt{Cq}}_{\texttt{all}}$.

In the next section, we discuss some related works. In Section 3, we present the PAC-Bayesian multiview learning framework [22]. In Section 4, we derive our multiview learning algorithm PB-MVBoost. Before concluding in Section 6, we experiment our algorithm in Section 5.

2. Related Work

Learning a weighted majority vote is closely related to ensemble methods [27, 28]. In the ensemble methods literature, it is well known that we desire to combine voters that make errors on different data points [24]. Intuitively, this means that the voters disagree on some data points. This notion of disagreement (or agreement) is sometimes called diversity between classifiers [29, 30, 24]. Even if there is no consensus on the definition of "diversity", controlling it while keeping good accuracy is at the heart of a majority of ensemble methods: indeed if all the voters agree on all the points then there is no interest to combine them, only one will be sufficient. Similarly, when we combine multiple views (or representations), it is known that controlling diversity between the views plays a vital role for learning the final majority vote [21, 22, 23, 25]. Most of the existing ensemble-based multiview learning algorithms try to exploit either view consistency (agreement between views) [31, 32, 33] or diversity between views [34, 22, 35, 36] in different manners. Janodet et al. [31] proposed a boosting based multiview learning algorithm for two views, called 2-Boost. At each iteration, the algorithm learns the weights over the view-specific voters by maintaining a single distribution over the learning examples. Conversely, Koço et

al. [32] proposed Mumbo that maintains separate distributions for each view. For each view, the algorithm reduces the weights associated with the examples hard to classify, and increases the weights of those examples in the other views. This trick allows a communication between the views with the objective to maintain view consistency. Compared to our approach, we follow a two-level learning strategy where we learn (hyper-)posterior distributions/weights over view-specific voters and views. In order to take into account accuracy and diversity between the views, we optimize the multiview C-Bound (an upper-bound over the risk of multiview majority vote learned, see *e.g.* [11, 14, 22]).

Xu and Sun [34] proposed EMV-AdaBoost, an embedded multiview Adaboost algorithm, restricted to two views. At each iteration, an example contributes to the error if it is misclassified by any of the view-specific voters and the diversity between the views is captured by weighting the error by the agreement between the views. Peng et al. [35, 36] proposed variants of Boost.SH (boosting with SHared weight distribution) which controls the diversity for more than two views. Similarly than our approach, they maintain a single global distribution over the learning examples for all the views. To control the diversity between the views, at each iteration they update the distribution over the views by casting the algorithm in two ways: *i)* a multiarmed bandit framework (`rBoost.SH`) and *ii)* an expert strategy framework (`eBoost.SH`) consisting of set of strategies (distribution over views) for weighing views. At the end, their multiview majority vote is a combination of T weighted base voters, where T is the number of iterations for boosting. Whereas, our multiview majority vote is a weighted combination of the view-specific voters over all the weighted views.

Furthermore, our approach encompasses the one of Amini et al. [21] and Xiao and Guo [33]. Amini et al. [21] proposed a Rademacher analysis for expectation of individual risks of each view-specific classifier (for more than two views). Xiao and Guo [33] derived a weighted majority voting Adaboost algorithm which learns weights over view-specific voters at each iteration of the algorithm. Both of these approaches maintain a uniform distribution over the views whereas our algorithm learns the weights over the views such that they capture diversity

between the views. Moreover, it is important to note that Sun et al. [37] proposed a PAC-Bayesian analysis for multiview learning over the concatenation of views but limited to two views and to a particular kind of voters: linear classifiers. This has allowed them to derive a SVM-like learning algorithm but dedicated to multiview with exactly two views. In our work, we are interested in learning from more than two views and without any restrictions on the classifier type. Contrary to them, we followed a two-level distributions' hierarchy where we learn weights over view-specific classifiers and weights over views.

3. The Multiview PAC-Bayesian Framework

3.1. Notations and Setting

In this work, we tackle multiview binary classification tasks where the observations are described with $V \geq 2$ different representation spaces, *i.e.*, views. Let $\mathcal{V}$ be the set of these V views. Formally, we focus on tasks for which the input space is $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_V$, where $\forall v \in \mathcal{V},\ X_v \subseteq \mathbb{R}^{d_v}$ is a d_v-dimensional input space, and the binary output space is $\mathcal{Y} = \{-1, +1\}$. We assume that $\mathcal{D}$ is a fixed but unknown distribution over $\mathcal{X} \times \mathcal{Y}$. We stand in the PAC-Bayesian supervised learning setting where an observation $\mathbf{x} = (x^1, x^2, \ldots, x^V) \in \mathcal{X}$ is given with its label $y \in \mathcal{Y}$, and is independently and identically drawn (*i.i.d.*) from $\mathcal{D}$. A learning algorithm is then provided with a training sample S of n examples *i.i.d.* from $\mathcal{D}$: $S = \{(\mathbf{x}_i, y_i)\}_{i=1}^n \sim (\mathcal{D})^n$, where $(\mathcal{D})^n$ stands for the distribution of a n-sample. For each view $v \in \mathcal{V}$, we consider a view-specific set $\mathcal{H}_v$ of voters $h : \mathcal{X}_v \to \mathcal{Y}$, and a prior distribution P_v on $\mathcal{H}_v$. Given a *hyper-prior* distribution π over the views $\mathcal{V}$, and a multiview learning sample S, our PAC-Bayesian learner objective is twofold: *i)* finding a posterior distribution Q_v over $\mathcal{H}_v$ for all views $v \in \mathcal{V}$, and *ii)* finding a *hyper-posterior* distribution ρ on the set of the views $\mathcal{V}$. This defines a hierarchy of distributions illustrated on Figure 1. The learned distributions express a multiview weighted majority vote[1]

[1] In the PAC-Bayesian literature, the weighted majority vote is sometimes called the Bayes classifier.

defined as

$$B_\rho(\mathbf{x}) = \text{sign}\left[\mathop{\mathbb{E}}_{v\sim\rho}\mathop{\mathbb{E}}_{h\sim Q_v} h(x^v)\right]. \tag{1}$$

Thus, the learner aims at constructing the posterior and hyper-posterior distributions that minimize the true risk $R_{\mathcal{D}}(B_\rho)$ of the multiview weighted majority vote

$$R_{\mathcal{D}}(B_\rho) = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \mathbb{1}_{[B_\rho(\mathbf{x})\neq y]},$$

where $\mathbb{1}_{[\pi]} = 1$ if the predicate π is true and 0 otherwise. The above risk of the deterministic weighted majority vote is closely related to the Gibbs risk $R_{\mathcal{D}}(G_\rho)$ defined as the expectation of the individual risks of each voter that appears in the majority vote. More formally, in our multiview setting, we have

$$R_{\mathcal{D}}(G_\rho) \;=\; \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}}\mathop{\mathbb{E}}_{v\sim\rho}\mathop{\mathbb{E}}_{h\sim Q_v} \mathbb{1}_{[h(x^v)\neq y]},$$

and its empirical counterpart is

$$R_S(G_\rho) = \frac{1}{n}\sum_{i=1}^{n}\mathop{\mathbb{E}}_{v\sim\rho}\mathop{\mathbb{E}}_{h\sim Q_v} \mathbb{1}_{[h(x_i^v)\neq y_i]}.$$

In fact, if B_ρ misclassifies $\mathbf{x}\in\mathcal{X}$, then at least half of the view-specific voters from all the views (according to hyper-posterior and posterior distributions) makes an error on $\mathbf{x}$. Then, it is well known [38, 39, 11] that $R_{\mathcal{D}}(B_\rho)$ is upper-bounded by twice $R_{\mathcal{D}}(G_\rho)$:

$$R_{\mathcal{D}}(B_\rho) \leq 2R_{\mathcal{D}}(G_\rho).$$

In consequence, a generalization bound for $R_{\mathcal{D}}(G_\rho)$ gives rise to a generalization bound for $R_{\mathcal{D}}(B_\rho)$.

There exist tighter relations [9, 11, 40], such as the C-Bound [40, 11] which captures a trade-off between the Gibbs risk $R_{\mathcal{D}}(G_\rho)$ and the disagreement between pairs of voters. This latter can be seen as a measure of diversity among the voters involved in the majority vote [41, 15], that is a key element to control from a multiview point of view [1, 21, 22, 24, 25]. The C-Bound can be extended to our multiview setting as below.

Lemma 1 (Multiview C-Bound). *Let $V \geq 2$ be the number of views. For all posterior $\{Q_v\}_{v=1}^V$ distributions over $\{\mathcal{H}_v\}_{v=1}^V$ and hyper-posterior ρ distribution over views $\mathcal{V}$, if $R_{\mathcal{D}}(G_\rho) < \frac{1}{2}$, then we have*

$$R_{\mathcal{D}}(B_\rho) \leq 1 - \frac{\big(1 - 2R_{\mathcal{D}}(G_\rho)\big)^2}{1 - 2d_{\mathcal{D}}(\rho)} \tag{2}$$

$$\leq 1 - \frac{\Big(1 - 2\,\mathbb{E}_{v\sim\rho} R_{\mathcal{D}}(G_{Q_v})\Big)^2}{1 - 2\,\mathbb{E}_{v\sim\rho} d_{\mathcal{D}}(Q_v)}\,, \tag{3}$$

where $d_{\mathcal{D}}(\rho)$ is the expected disagreement between pairs of voters defined as

$$d_{\mathcal{D}}(\rho) = \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}_{\mathcal{X}}} \mathop{\mathbb{E}}_{v\sim\rho} \mathop{\mathbb{E}}_{v'\sim\rho} \mathop{\mathbb{E}}_{h\sim Q_v} \mathop{\mathbb{E}}_{h'\sim Q_{v'}} \mathbb{1}_{[h(x^v)\neq h'(x^{v'})]},$$

and $R_{\mathcal{D}}(G_{Q_v})$ and $d_{\mathcal{D}}(Q_v)$ are respectively the true view-specific Gibbs risk and the expected disagreement defined as

$$R_{\mathcal{D}}(G_{Q_v}) = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{h\sim Q_v} \mathbb{1}_{[h(x^v)\neq y]}\,,$$

$$d_{\mathcal{D}}(Q_v) = \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}_{\mathcal{X}}} \mathop{\mathbb{E}}_{h\sim Q_v} \mathop{\mathbb{E}}_{h'\sim Q_v} \mathbb{1}_{[h(x^v)\neq h'(x^v)]}.$$

Proof. Similarly than done for the classical C-Bound [11, 40], Equation (2) follows from the Cantelli-Chebyshev's inequality (we provide the proof in Appendix B).

Equation (3) is obtained by rewriting $R_{\mathcal{D}}(G_\rho)$ as the ρ-average of the risk associated to each view, and lower-bounding $d_{\mathcal{D}}(\rho)$ by the ρ-average of the disagreement associated to each view. First we notice that in the binary setting where $y \in \{-1, 1\}$ and $h : \mathcal{X} \to \{-1, 1\}$, we have $\mathbb{1}_{[h(x^v)\neq y]} = \frac{1}{2}(1 - y\,h(x^v))$, and

$$\begin{aligned} R_{\mathcal{D}}(G_\rho) &= \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{v\sim\rho} \mathop{\mathbb{E}}_{h\sim Q_v} \mathbb{1}_{[h(x^v)\neq y]} \\ &= \frac{1}{2}\left(1 - \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{v\sim\rho} \mathop{\mathbb{E}}_{h\sim Q_v} y\,h(x^v)\right) \\ &= \mathop{\mathbb{E}}_{v\sim\rho} R_{\mathcal{D}}(G_{Q_v})\,. \end{aligned}$$

Moreover, we have

$$d_{\mathcal{D}}(\rho) = \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}_{\mathcal{X}}} \mathop{\mathbb{E}}_{v\sim\rho} \mathop{\mathbb{E}}_{v'\sim\rho} \mathop{\mathbb{E}}_{h\sim Q_v} \mathop{\mathbb{E}}_{h'\sim Q_{v'}} \mathbb{1}_{[h(x^v)\neq h'(x^{v'})]}$$

$$= \frac{1}{2}\left(1 - \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}_\mathcal{X}} \mathop{\mathbb{E}}_{v\sim\rho} \mathop{\mathbb{E}}_{v'\sim\rho} \mathop{\mathbb{E}}_{h\sim Q_v} \mathop{\mathbb{E}}_{h\sim Q_{v'}} h(x^v) \times h'(x^{v'})\right)$$
$$= \frac{1}{2}\left(1 - \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}_\mathcal{X}} \left[\mathop{\mathbb{E}}_{v\sim\rho} \mathop{\mathbb{E}}_{h\sim Q_v} h(x^v)\right]^2\right).$$

From Jensen's inequality (Theorem 4, in Appendix) it comes

$$d_\mathcal{D}(\rho) \geq \frac{1}{2}\left(1 - \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}_\mathcal{X}} \mathop{\mathbb{E}}_{v\sim\rho} \left[\mathop{\mathbb{E}}_{h\sim Q_v} h(x^v)\right]^2\right)$$
$$= \mathop{\mathbb{E}}_{v\sim\rho}\left[\frac{1}{2}\left(1 - \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}_\mathcal{X}} \left[\mathop{\mathbb{E}}_{h\sim Q_v} h(x^v)\right]^2\right)\right]$$
$$= \mathop{\mathbb{E}}_{v\sim\rho} d_\mathcal{D}(Q_v)\,.$$

By replacing $R_\mathcal{D}(G_\rho)$ and $d_\mathcal{D}(\rho)$ in Equation (2), we obtain

$$1 - \frac{\left(1 - 2R_\mathcal{D}(G_\rho)\right)^2}{1 - 2d_\mathcal{D}(\rho)} \leq 1 - \frac{\left(1 - 2\,\mathbb{E}_{v\sim\rho} R_\mathcal{D}(G_{Q_v})\right)^2}{1 - 2\,\mathbb{E}_{v\sim\rho} d_\mathcal{D}(Q_v)}\,.$$

■

Equation (2) suggests that a good trade-off between the Gibbs risk and the disagreement between pairs of voters will lead to a well-performing majority vote. Equation (3) controls the diversity among the views (important for multiview learning [21, 22, 23, 25]) thanks to the disagreement's expectation over the views $\mathbb{E}_{v\sim\rho}\, d_\mathcal{D}(Q_v)$.

3.2. The General Multiview PAC-Bayesian Theorem

In this section, we give a general multiview PAC-Bayesian theorem [22] that takes the form of a generalization bound for the Gibbs risk in the context of a two-level hierarchy of distributions. A key step in PAC-Bayesian proofs is the use of a *change of measure inequality* [39], based on the Donsker-Varadhan inequality [42]. Lemma 2 below extends this tool to our multiview setting.

Lemma 2. *For any set of priors $\{P_v\}_{v=1}^V$ over $\{\mathcal{H}_v\}_{v=1}^V$ and any set of posteriors $\{Q_v\}_{v=1}^V$ over $\{\mathcal{H}_v\}_{v=1}^V$, for any hyper-prior distribution π on views $\mathcal{V}$ and hyper-posterior distribution ρ on $\mathcal{V}$, and for any measurable function $\phi : \mathcal{H}_v \to \mathbb{R}$, we have*

$$\mathop{\mathbb{E}}_{v\sim\rho} \mathop{\mathbb{E}}_{h\sim Q_v} \phi(h) \leq \mathop{\mathbb{E}}_{v\sim\rho} \mathrm{KL}(Q_v\|P_v) + \mathrm{KL}(\rho\|\pi) + \ln\left(\mathop{\mathbb{E}}_{v\sim\pi} \mathop{\mathbb{E}}_{h\sim P_v} e^{\phi(h)}\right).$$

Proof. Deferred to Appendix C ∎

Based on Lemma 2, the following theorem gives a generalization bound for multiview learning. Note that, as done by Germain et al. [10, 11] we rely on a general convex function $D : [0,1] \times [0,1] \to \mathbb{R}$, which measures the "deviation" between the empirical and the true Gibbs risk.

Theorem 1. *Let $V \geq 2$ be the number of views. For any distribution $\mathcal{D}$ on $\mathcal{X} \times \mathcal{Y}$, for any set of prior distributions $\{P_v\}_{v=1}^V$ over $\{\mathcal{H}_v\}_{v=1}^V$, for any hyper-prior distributions π over $\mathcal{V}$, for any convex function $D : [0,1] \times [0,1] \to \mathbb{R}$, for any $\delta \in (0,1]$, with a probability at least $1-\delta$ over the random choice of $S \sim (\mathcal{D})^n$, for all posterior $\{Q_v\}_{v=1}^V$ over $\{\mathcal{H}_v\}_{v=1}^V$ and hyper-posterior ρ over $\mathcal{V}$ distributions, we have:*

$$D\left(R_S(G_\rho), R_{\mathcal{D}}(G_\rho)\right) \leq \frac{1}{m}\Bigg[\mathop{\mathbb{E}}_{v\sim\rho} \mathrm{KL}(Q_v\|P_v) + \mathrm{KL}(\rho\|\pi) + \ln\left(\frac{1}{\delta}\mathop{\mathbb{E}}_{S\sim(\mathcal{D})^n}\mathop{\mathbb{E}}_{v\sim\pi}\mathop{\mathbb{E}}_{h\sim P_v} e^{nD(R_S(h),R_{\mathcal{D}}(h))}\right)\Bigg].$$

Proof. First, note that $\mathop{\mathbb{E}}_{v\sim\pi}\mathop{\mathbb{E}}_{h\sim P_v} e^{nD(R_S(h),R_{\mathcal{D}}(h))}$ is a non-negative random variable. Using Markov's inequality, with $\delta \in (0,1]$, and a probability at least $1-\delta$ over the random choice of the multiview learning sample $S \sim (\mathcal{D})^n$, we have

$$\mathop{\mathbb{E}}_{v\sim\pi}\mathop{\mathbb{E}}_{h\sim P_v} e^{n\,D(R_S(h),R_{\mathcal{D}}(h))} \leq \frac{1}{\delta}\mathop{\mathbb{E}}_{S\sim(\mathcal{D})^n}\mathop{\mathbb{E}}_{v\sim\pi}\mathop{\mathbb{E}}_{h\sim P_v} e^{nD(R_S(h),R_{\mathcal{D}}(h))}.$$

By taking the logarithm on both sides, with a probability at least $1-\delta$ over $S \sim (\mathcal{D})^n$, we have

$$\ln\left[\mathop{\mathbb{E}}_{v\sim\pi}\mathop{\mathbb{E}}_{h\sim P_v} e^{n\,D(R_S(h),R_{\mathcal{D}}(h))}\right] \leq \ln\left[\frac{1}{\delta}\mathop{\mathbb{E}}_{S\sim(\mathcal{D})^n}\mathop{\mathbb{E}}_{v\sim\pi}\mathop{\mathbb{E}}_{h\sim P_v} e^{nD(R_S(h),R_{\mathcal{D}}(h))}\right] \quad (4)$$

We now apply Lemma 2 on the left-hand side of Inequality (4) with $\phi(h) = n\,D(R_S(h), R_{\mathcal{D}}(h))$. Therefore, for any Q_v on $\mathcal{H}_v$ for all views $v \in \mathcal{V}$, and for any ρ on views $\mathcal{V}$, with a probability at least $1-\delta$ over $S \sim (\mathcal{D})^n$, we have

$$\ln\left[\mathop{\mathbb{E}}_{v\sim\pi}\mathop{\mathbb{E}}_{h\sim P_v} e^{n\,D(R_S(h),R_{\mathcal{D}}(h))}\right] \geq n\mathop{\mathbb{E}}_{v\sim\rho}\mathop{\mathbb{E}}_{h\sim Q_v} D(R_S(h),R_{\mathcal{D}}(h)) - \mathop{\mathbb{E}}_{v\sim\rho}\mathrm{KL}(Q_v\|P_v) - \mathrm{KL}(\rho\|\pi)$$

$$\geq\ n\,D\left(\mathop{\mathbb{E}}_{v\sim\rho}\mathop{\mathbb{E}}_{h\sim Q_v} R_S(h), \mathop{\mathbb{E}}_{v\sim\rho}\mathop{\mathbb{E}}_{h\sim Q_v} R_{\mathcal{D}}(h)\right) - \mathop{\mathbb{E}}_{v\sim\rho} \mathrm{KL}(Q_v\|P_v) - \mathrm{KL}(\rho\|\pi),$$

where the last inequality is obtained by applying Jensen's inequality on the convex function D. By rearranging the terms, we have

$$D\left(\mathop{\mathbb{E}}_{v\sim\rho}\mathop{\mathbb{E}}_{h\sim Q_v} R_S(h), \mathop{\mathbb{E}}_{v\sim\rho}\mathop{\mathbb{E}}_{h\sim Q_v} R_{\mathcal{D}}(h)\right) \leq \frac{1}{m}\Bigg[\mathop{\mathbb{E}}_{v\sim\rho} \mathrm{KL}(Q_v\|P_v) + \mathrm{KL}(\rho\|\pi) + \ln\left(\frac{1}{\delta}\mathop{\mathbb{E}}_{S\sim(\mathcal{D})^n}\mathop{\mathbb{E}}_{v\sim\pi}\mathop{\mathbb{E}}_{h\sim P_v} e^{n\,D(R_S(h),R_{\mathcal{D}}(h))}\right)\Bigg].$$

Finally, the theorem statement is obtained by rewriting

$$\mathop{\mathbb{E}}_{v\sim\rho}\mathop{\mathbb{E}}_{h\sim Q_v} R_S(h) = R_S(G_\rho), \tag{5}$$

$$\mathop{\mathbb{E}}_{v\sim\rho}\mathop{\mathbb{E}}_{h\sim Q_v} R_{\mathcal{D}}(h) = R_{\mathcal{D}}(G_\rho)\,. \tag{6}$$

■

Compared to the classical single-view PAC-Bayesian Bound of Germain et al. [10, 11], the main difference relies on the introduction of the view-specific prior and posterior distributions, which mainly leads to an additional term $\mathbf{E}_{v\sim\rho}\,\mathrm{KL}(Q_v\|P_v)$ expressed as the expectation of the view-specific Kullback-Leibler divergence term over the views $\mathcal{V}$ according to the hyper-posterior distribution ρ.

Theorem 1 provides tools to derive PAC-Bayesian generalization bounds for a multiview supervised learning setting. Indeed, by making use of the same trick as Germain et al. [10, 11], by choosing a suitable convex function D and upper-bounding $\mathop{\mathbb{E}}_{S\sim(\mathcal{D})^n}\mathop{\mathbb{E}}_{v\sim\pi}\mathop{\mathbb{E}}_{h\sim P_v} e^{n\,D(R_S(h),R_{\mathcal{D}}(h))}$, we obtain an instantiation of Theorem 1. In the next section, we give an example of this kind of deviation through the approach of Catoni [7], that is one of the three classical PAC-Bayesian Theorems [6, 7, 8, 43].

3.3. An Example of Instantiation of the Multiview PAC-Bayesian Theorem

To obtain the following theorem which is a generalization bound with the Catoni [7]'s point of view, we put D as $D(a,b) = \mathcal{F}(b) - C\,a$ where $\mathcal{F}$ is a convex function $\mathcal{F}$ and $C > 0$ is a real number [10, 11].

Corollary 1. *Let $V \geq 2$ be the number of views. For any distribution $\mathcal{D}$ on $\mathcal{X} \times \mathcal{Y}$, for any set of prior distributions $\{P_v\}_{v=1}^V$ on $\{\mathcal{H}\}_{v=1}^V$, for any hyper-prior distributions π over $\mathcal{V}$, for any $\delta \in (0,1]$, with a probability at least $1-\delta$ over the random choice of $S \sim (\mathcal{D})^n$ for all posterior $\{Q_v\}_{v=1}^V$ and hyper-posterior ρ distributions, we have:*

$$R_{\mathcal{D}}(G_\rho)$$
$$\leq \frac{1}{1-e^{-C}}\left(1-\exp\left[-\left(C\, R_S(G_\rho) + \frac{1}{n}\left[\mathop{\mathbb{E}}_{v\sim\rho} \mathrm{KL}(Q_v\|P_v) + \mathrm{KL}(\rho\|\pi) + \ln\tfrac{1}{\delta}\right]\right)\right]\right).$$

Proof. Deferred to Appendix D. ∎

This bound has the advantage of expressing a trade-off between the empirical Gibbs risk and the Kullback-Leibler divergences.

3.4. A Generalization Bound for the C-Bound

From a practical standpoint, as pointed out before, controlling the multiview C-Bound of Equation (3) can be very useful for tackling multiview learning. The next theorem is a generalization bound that justify the empirical minimization of the multiview C-bound (we use in our algorithm `PB-MVBoost` derived in Section 4).

Theorem 2. *Let $V \geq 2$ be the number of views. For any distribution $\mathcal{D}$ on $\mathcal{X}\times\mathcal{Y}$, for any set of prior distributions $\{P_v\}_{v=1}^V$, for any hyper-prior distributions π over views $\mathcal{V}$, and for any convex function $D : [0,1]\times[0,1] \to \mathbb{R}$, with a probability at least $1-\delta$ over the random choice of $S \sim (D)^n$ for all posterior $\{Q_v\}_{v=1}^v$ and hyper-posterior ρ distributions, we have:*

$$R_{\mathcal{D}}(B_\rho) \leq 1 - \frac{\left(1 - 2 \mathop{\mathbb{E}}_{v\sim\rho} \mathtt{sup}\ \left(\mathbf{r}^{\delta/2}_{Q_v,\mathcal{S}}\right)\right)^2}{1 - 2 \mathop{\mathbb{E}}_{v\sim\rho} \mathtt{inf}\ \mathbf{d}^{\delta/2}_{Q_v,\mathcal{S}}},$$

where

$$\mathbf{r}^{\delta/2}_{Q_v,\mathcal{S}} = \left\{ r : \mathrm{kl}(R_S(Q_v)\|r) \leq \frac{1}{n}\left[\mathrm{KL}(Q_v\|P_v) + \ln\frac{4\sqrt{m}}{\delta}\right] \text{ and } r \leq \frac{1}{2}\right\}, \quad (7)$$

$$\text{and } \mathbf{d}^{\delta/2}_{Q_v,\mathcal{S}} = \left\{ d : \mathrm{kl}(d^S_{Q_v}\|d) \leq \frac{1}{n}\left[2.\,\mathrm{KL}(Q_v\|P_v) + \ln\frac{4\sqrt{m}}{\delta}\right]\right\}. \quad (8)$$

Proof. Similarly to Equations (23) and (24) of [11], we define the sets $\mathbf{r}^{\delta/2}_{Q_v,\mathcal{S}}$ (Equation (7)) and $\mathbf{d}^{\delta/2}_{Q_v,\mathcal{S}}$ (Equation (8)) for our setting. Finally, the bound is obtained (from Equation (3) of Lemma 1) by replacing the view-specific Gibbs risk $R_{\mathcal{D}}(G_{Q_v})$ by its upper bound $\texttt{sup}\ \mathbf{r}^{\delta/2}_{Q_v,\mathcal{S}}$ and expected disagreement $d_{\mathcal{D}}(Q_v)$ by its lower bound $\texttt{inf}\ \mathbf{d}^{\delta/2}_{Q_v,\mathcal{S}}$. ■

4. The PB-MVBoost algorithm

In this section we exploit our two-level hierarchical strategy (see Figure 1) in order to learn a well-performing weighted combination of view-specific voters (or views) as in Equation (1). Therefore, we propose to follow a well-known approach to learn weighted combination of voters, that is boosting. Indeed, boosting aims at combining a set of weak voters[2] to construct a good majority vote. Typically, boosting algorithms repeatedly learn a "weak" voter (using a learning algorithm) with different probability distribution over the learning sample S. Finally, it combines all the weak voters in order to have one single strong classifier which performs better than the individual weak voters. Recall that in multiview learning it is crucial to take into account the interactions between voters and views [21, 22, 23, 25]. We adapt this principle to our setting for combining a set of view-specific weak voters while taking into account the accuracy and diversity between them. We develop a multiview learning algorithm PB-MVBoost (see Algorithm 1), which allows to iteratively learn the set of view-specific classifiers that the algorithm will combine.

Concretely, for a given training set $S = \{(\mathbf{x}_i, y_i), \ldots, (\mathbf{x}_n, y_n)\} \in (\mathcal{X} \times \{-1, +1\})^n$ of size n, our algorithm PB-MVBoost maintains a distribution over the examples which is initialized as uniform. Then at each iteration, V view-specific weak classifiers are learned according to the current distribution $\mathcal{D}_t$ (Step 5), and their corresponding errors ϵ^t_v are estimated (Step 6).

[2] In boosting, the performance of a weak classifier is only slightly better than random guessing.

Algorithm 1 PB-MVBoost

Input: Training set $S = (\mathbf{x}_i, y_i), \ldots, (\mathbf{x}_n, y_n)$, where $\mathbf{x}_i = (x^1, x^2, \ldots, x^V)$ and $y_i \in \{-1, 1\}$.

For each view $v \in \mathcal{V}$, a view-specific hypothesis set $\mathcal{H}_v$.

Number of iterations T.

for $\mathbf{x}_i \in S$ **do**

$\mathcal{D}_1(\mathbf{x}_i) \leftarrow \frac{1}{n}$

$\forall v \in \mathcal{V}\ \rho_v^1 \leftarrow \frac{1}{V}$

for $t = 1, \ldots, T$ **do**

$\forall v \in \mathcal{V},\ h_v^t \leftarrow \text{argmin}_{h \in \mathcal{H}_v} \underset{(\mathbf{x}_i, y_i) \sim \mathcal{D}_t}{\mathbb{E}} \left[\mathbb{1}_{[h(x_i^v) \neq y_i]}\right]$

Compute error: $\forall v \in \mathcal{V},\ \epsilon_v^t \leftarrow \underset{(\mathbf{x}_i, y_i) \sim \mathcal{D}_t}{\mathbb{E}} \left[\mathbb{1}_{[h_v^t(x_i^v) \neq y_i]}\right]$

Compute voter weights (taking into account view specific information):

$$\forall v \in \mathcal{V}, Q_v^t \leftarrow \frac{1}{2}\left[\ln\left(\frac{1-\epsilon_v^t}{\epsilon_v^t}\right)\right]$$

Optimize the multiview C-Bound to learn weights over the views

$$\rho^t \leftarrow \text{argmax}_\rho \quad \frac{\left[1 - 2\sum_{v=1}^V \rho_v r_v^t\right]^2}{1 - 2\sum_{v=1}^V \rho_v d_v^t}$$

$$\text{such that} \quad \sum_{v=1}^V \rho_v = 1, \quad \rho_v \geq 0 \quad \forall v \in \{1, \ldots, V\}$$

$$\text{where } \forall v \in \mathcal{V},\ r_v^t \leftarrow \underset{(\mathbf{x}_i, y_i) \sim \mathcal{D}_t}{\mathbb{E}} \underset{h \sim \mathcal{H}_v}{\mathbb{E}} \left[\mathbb{1}_{[h(x_i^v) \neq y_i]}\right]$$

$$\forall v \in \mathcal{V},\ d_v^t \leftarrow \underset{(\mathbf{x}_i, y_i) \sim \mathcal{D}_t}{\mathbb{E}} \underset{h, h' \sim \mathcal{H}_v}{\mathbb{E}} \left[\mathbb{1}_{[h(x_i^v) \neq h'(x_i^v)]}\right]$$

for $\mathbf{x}_i \in S$ **do**

$$\mathcal{D}_{t+1}(\mathbf{x}_i) \leftarrow \frac{\mathcal{D}_t(\mathbf{x}_i) \exp\left(-y_i \sum_{v=1}^V \rho_v^t Q_v^t h_v^t(x_i^v)\right)}{\sum_{j=1}^n \mathcal{D}_t(\mathbf{x}_j) \exp\left(-y_j \sum_{v=1}^V \rho_v^t Q_v^t h_v^t(x_j^v)\right)}$$

Return: For each view $v \in \mathcal{V}$, weights over view-specific voters and weights over views, *i.e.*, ρ^T

Similarly to the Adaboost algorithm [18], the weights of each view-specific classifier $(Q_v^t)_{1\leq v\leq V}$ are then computed with respect to these errors as

$$\forall v \in \mathcal{V}, Q_v^t \leftarrow \frac{1}{2}\left[\ln\left(\frac{1-\epsilon_v^t}{\epsilon_v^t}\right)\right].$$

To learn the weights $(\rho_v)_{1\leq v\leq V}$ over the views, we optimize the multiview C-Bound, given by Equation (3) of Lemma 1 (Step 8 of algorithm), which in our case writes as a constraint maximization problem:

$$\begin{aligned} \max_{\rho} \quad & \frac{\left[1-2\sum_{v=1}^{V}\rho_v r_v^t\right]^2}{1-2\sum_{v=1}^{V}\rho_v d_v^t}, \\ s.t. \quad & \sum_{v=1}^{V}\rho_v = 1, \quad \rho_v \geq 0 \quad \forall v \in \{1,...,V\}. \end{aligned}$$

where r_v is the view-specific Gibbs risk, and d_v the expected disagreement over all view-specific voters defined as follows.

$$r_v^t = \mathop{\mathbb{E}}_{(\mathbf{x}_i,y_i)\sim\mathcal{D}_t} \mathop{\mathbb{E}}_{h\sim\mathcal{H}_v} \mathbb{1}_{[h(x_i^v)\neq y_i]}, \tag{9}$$

$$d_v^t = \mathop{\mathbb{E}}_{(\mathbf{x}_i,y_i)\sim\mathcal{D}_t} \mathop{\mathbb{E}}_{h,h'\sim\mathcal{H}_v} \mathbb{1}_{[h(x_i^v)\neq h'(x_i^v)]}. \tag{10}$$

Intuitively, the multiview C-Bound tries to diversify the view-specific voters and views (Equation (10)) while controlling the classification error of the view-specific classifiers (Equation (9)). This allows us to control the accuracy and the diversity between the views which is an important ingredient in multiview learning [34, 22, 35, 36, 15].

In Section 5, we empirically show that our algorithm minimizes the multiview C-Bound over the iterations of the algorithm (this is theoretically justified by the generalization bound of Theorem 2). Finally, we update the distribution over training examples $\mathbf{x}_i$ (Step 9), by following the Adaboost algorithm and in a way that the weights of misclassified (resp. well classified) examples by the final weighted majority classifier increase (resp. decrease).

$$\mathcal{D}_{t+1}(\mathbf{x}_i) \leftarrow \frac{\mathcal{D}_t(\mathbf{x}_i)\exp\left(-y_i\sum_{v=1}^{V}\rho_v^t Q_v^t h_v^t(x_i^v)\right)}{\sum_{j=1}^{n}\mathcal{D}_t(\mathbf{x}_j)\exp\left(-y_j\sum_{v=1}^{V}\rho_v^t Q_v^t h_v^t(x_j^v)\right)}.$$

Intuitively, this forces the view-specific classifiers to be consistent with each other, which is important for multiview learning [31, 32, 33]. Finally, after T iterations of the algorithm, we learn the weights over the view-specific voters and weights over the views leading to a well-performing weighted multiview majority vote defined as

$$B_\rho(\mathbf{x}) = \text{sign}\left(\sum_{v=1}^{V} \rho_v^T \sum_{t=1}^{T} Q_v^t h_v^t(x^v)\right).$$

4.1. A note on the Complexity of `PB-MVBoost`

The complexity of learning a decision tree classifier is $O(d\, nlog(n))$, where d is the depth of the decision tree. We learn the weights over the views by optimizing Equation (3) (Step 8 of our algorithm) using SLSQP method which has time complexity of $O(V^3)$. Therefore, the overall complexity is $O\left(T\ \left(V^3 + V\, d_v\, n.log(n)\right)\right)$. Note that it is easy to parallelize our algorithm: by using V different machines, we can learn the view-specific classifiers and weights over them (Steps 4 to 7).

5. Experimental Results

In this section, we present experiments to show the potential of our algorithm `PB-MVBoost` on the following datasets.

5.1. Datasets

MNIST

MNIST is a publicly available dataset consisting of $70,000$ images of handwritten digits distributed over ten classes [26]. For our experiments, we generated 2 four-view datasets[3] where each view is a vector of $\mathbb{R}^{14\times 14}$. Similarly than done by Chen et al. [44], the first dataset ($\texttt{MNIST}_1$) is generated by considering 4 quarters of image as 4 views. For the second dataset ($\texttt{MNIST}_2$), we consider 4

[3] $\texttt{MNIST}_1$ and $\texttt{MNIST}_2$ datasets are available at `https://github.com/goyalanil/Multiview_Dataset_MNIST`

Strategy	MNIST$_1$		MNIST$_2$		Reuters	
	Accuracy	F_1	Accuracy	F_1	Accuracy	F_1
Mono	$.9034 \pm .001^{\downarrow}$	$.5353 \pm .006^{\downarrow}$	$.9164 \pm .001^{\downarrow}$	$.5987 \pm .007^{\downarrow}$	$.8420 \pm .002^{\downarrow}$	$.5051 \pm .007^{\downarrow}$
Concat	$.9224 \pm .002^{\downarrow}$	$.6168 \pm .011^{\downarrow}$	$.9214 \pm .002^{\downarrow}$	$.6142 \pm .013^{\downarrow}$	$.8431 \pm .004^{\downarrow}$	$.5088 \pm .012^{\downarrow}$
Fusion$_{\mathtt{dt}}$	$.9320 \pm .001^{\downarrow}$	$.5451 \pm .019^{\downarrow}$	$.9366 \pm .001^{\downarrow}$	$.5937 \pm .020^{\downarrow}$	$.8587 \pm .003^{\downarrow}$	$.4128 \pm .017^{\downarrow}$
MV-MV	$.9402 \pm .001^{\downarrow}$	$.6321 \pm .009^{\downarrow}$	$.9450 \pm .001^{\downarrow}$	$.6849 \pm .008^{\downarrow}$	$.8780 \pm .002^{\downarrow}$	$.5443 \pm .012^{\downarrow}$
rBoost.SH	$.9256 \pm .001^{\downarrow}$	$.5315 \pm .009^{\downarrow}$	$.9545 \pm .0007$	$.7258 \pm .005^{\downarrow}$	$.8853 \pm .002$	$.5718 \pm .011^{\downarrow}$
MV-AdaBoost	*.9514* $\pm .001$	$.6510 \pm .012^{\downarrow}$	*.9641* $\pm .0009$	$.7776 \pm .007^{\downarrow}$	$.8942 \pm .006$	$.5581 \pm .013^{\downarrow}$
MVBoost	$.9494 \pm .003^{\downarrow}$	*.7733* $\pm .009^{\downarrow}$	$.9555 \pm .002$	*.7910* $\pm .006^{\downarrow}$	$.8627 \pm .007^{\downarrow}$	$.5789 \pm .012^{\downarrow}$
Fusion$^{\mathtt{all}}_{\mathtt{Cq}}$	$.9418 \pm .002^{\downarrow}$	$.6120 \pm .040^{\downarrow}$	$.9548 \pm .003^{\downarrow}$	$.7217 \pm .041^{\downarrow}$	**.9001** $\pm .003$	**.6279** $\pm .019$
PB-MVBoost	**.9661** $\pm .0009$	**.8066** $\pm .005$	**.9674** $\pm .0009$	**.8166** $\pm .006$	*.8953* $\pm .002$	*.5960* $\pm .015^{\downarrow}$

Table 1: Test classification accuracy and F_1-score of different approaches averaged over all the classes and over 20 random sets of $n = 500$ labeled examples per training set. Along each column, the best result is in bold, and second one in italic. $^{\downarrow}$ indicates that a result is statistically significantly worse than the best result, according to a Wilcoxon rank sum test with $p < 0.02$.

overlapping views around the centre of images: this dataset brings redundancy between the views. These two datasets allow us to check if our algorithm is able to capture redundancy between the views. We reserve $10,000$ of images as test samples and remaining as training samples.

Multilingual, Multiview Text categorization

This dataset is a multilingual text classification data extracted from Reuters RCV1/RCV2 corpus[4]. It consists of more than $110,000$ documents written in five different languages (English, French, German, Italian and Spanish) distributed over six classes. We see different languages as different views of the data. We reserve 30% of documents as test samples and remaining as training data.

5.2. Experimental Protocol

While the datasets are multiclass, we transformed them as binary tasks by considering *one-vs-all* classification problems: for each class we learn a binary classifier by considering all the learning samples from that class as positive

[4]Reuters RCV1/RCV2 corpus is available at `https://archive.ics.uci.edu/ml/datasets/Reuters+RCV1+RCV2+Multilingual,+Multiview+Text+Categorization+Test+collection`

examples and the others as negative examples. We consider different sizes of learning sample S (150, 200, 250, 300, 500, 800, 1000) that are chosen randomly from the training data. Moreover, all the results are averaged over 20 random runs of the experiments. Since the classes are unbalanced, we report the accuracy along with F1-measure for the methods and all the scores are averaged over all the *one-vs-all* classification problems.

We consider two multiview learning algorithms based on our two-step hierarchical strategy, and compare the `PB-MVBoost`[5] algorithm described in Section 4, with a previously developed multiview learning algorithm [22], based on classifier late fusion approach [5], and referred to as $\texttt{Fusion}^{\texttt{all}}_{\texttt{Cq}}$. Concretely, at the first level, this algorithm trains different view-specific linear SVM models with different hyperparameter C values (12 values between 10^{-8} and 10^{3}). And, at the second level, it learns a weighted combination over the predictions of view-specific voters using PAC-Bayesian algorithm `CqBoost`[14] with a RBF kernel. Note that, algorithm `CqBoost` tends to minimize the PAC-Bayesian C-Bound [11] controlling the trade-off between accuracy and disagreement among voters. The hyperparameter γ of the RBF kernel is chosen over a set of 9 values between 10^{-6} and 10^{2}; and hyperparameter μ is chosen over a set of 8 values between 10^{-8} and 10^{-1}. To study the potential of our algorithms ($\texttt{Fusion}^{\texttt{all}}_{\texttt{Cq}}$ and `PB-MVBoost`), we considered following 7 baseline approaches:

- `Mono`: We learn a view-specific model for each view using a decision tree classifier and report the results of the best performing view.
- `Concat`: We learn one model using a decision tree classifier by concatenating features of all the views.
- $\texttt{Fusion}_{\texttt{dt}}$: This is a late fusion approach where we first learn the view-specific classifiers using 60% of learning samples. Then, we learn a final multiview weighted model over the predictions of the view-specific classifiers. For this approach, we used decision tree classifiers at both levels of learning.

[5] Code for `PB-MVBoost` is available at `https://github.com/goyalanil/PB-MVBoost`

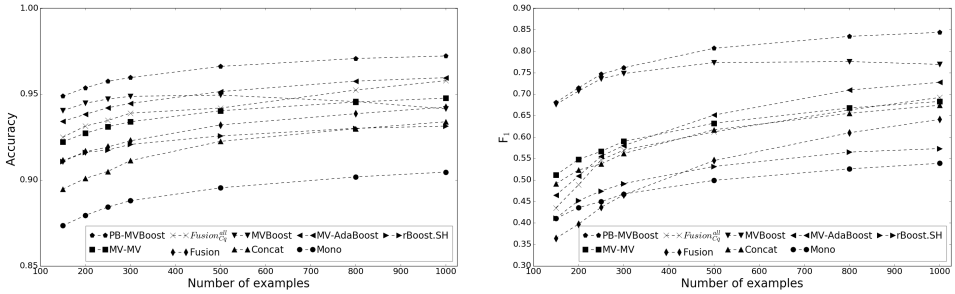
- `MV-MV`: We compute a multiview uniform majority vote (similar to approach followed by Amini et al. [21]) over all the view-specific classifiers' outputs in order to make final prediction. We learn view-specific classifiers using decision tree classifiers.

- `rBoost.SH`: This is the multiview learning algorithm proposed by Peng et al. [35, 36] where a single global distribution is maintained over the learning sample for all the views and the distribution over views are updated using multiarmed bandit framework. At each iteration, `rBoost.SH` selects a view according to the current distribution and learns the corresponding view-specific voter. For tuning the parameters, we followed the same experimental setting as Peng et al. [36].

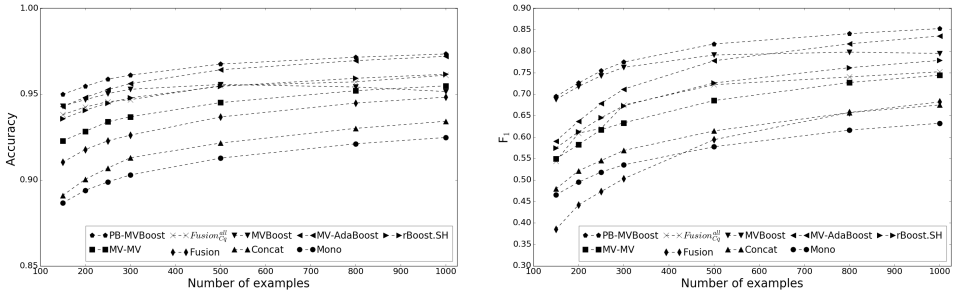
- `MV-AdaBoost`: This is a majority vote classifier over the view-specific voters trained using Adaboost algorithm. Here, our objective is to see the effect of maintaining separate distributions for all the views.

- `MVBoost`: This is a variant of our algorithm `PB-MVBoost` but without learning weights over views by optimizing multiview C-Bound. Here, our objective is to see the effect of learning weights over views on multiview learning.

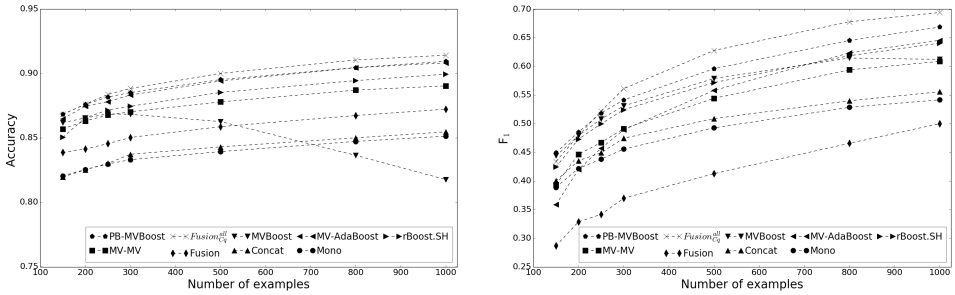
For all boosting based approaches (`rBoost.SH`, `MV-AdaBoost`, `MVBoost` and `PB-MVBoost`), we learn the view-specific voters using a decision tree classifier with depth 2 and 4 as a weak classifier for `MNIST`, and `Reuters` RCV1/RCV2 datasets respectively. For all these approaches, we kept $T = 100$ as the number of iterations. For optimization of multiview C-Bound, we used Sequential Least SQuares Programming (SLSQP) implementation provided by SciPy[6] [45] and the decision trees implementation from scikit-learn [46].



(a) $\texttt{MNIST}_1$



(b) $\texttt{MNIST}_2$



(c) `Reuters`

Figure 2: Evolution of accuracy and F_1-measure with respect to the number of labeled examples in the initial labeled training sets on $\texttt{MNIST}_1$, $\texttt{MNIST}_2$ and `Reuters` datasets.

5.3. Results

Firstly, we report the comparison of our algorithms $\texttt{Fusion}_{\texttt{Cq}}^{\texttt{all}}$ and PB-MVBoost (for $m = 500$) with all the considered baseline methods in Table 1. Secondly, Figure 2, illustrates the evolution of the performances according to the size of the learning sample. From the table, proposed two-step learning algorithm $\texttt{Fusion}_{\texttt{Cq}}^{\texttt{all}}$ is significantly better than the baseline approaches for Reuters dataset. Whereas, our boosting based algorithm PB-MVBoost is significantly better than all the baseline approaches for all the datasets. This shows that considering a two-level hierarchical strategy in a PAC-Bayesian manner is an effective way to handle multiview learning.

In Figure 3, we compare proposed algorithms $\texttt{Fusion}_{\texttt{Cq}}^{\texttt{all}}$ and PB-MVBoost in terms of accuracy, F_1-score and time complexity for $m = 500$ examples. For MNIST datasets, PB-MVBoost is significantly better than $\texttt{Fusion}_{\texttt{Cq}}^{\texttt{all}}$. For Reuters dataset, $\texttt{Fusion}_{\texttt{Cq}}^{\texttt{all}}$ performs better than PB-MVBoost but computation time for $\texttt{Fusion}_{\texttt{Cq}}^{\texttt{all}}$ is much higher than that of PB-MVBoost. Moreover, in Figure 2, we can see that the performance (in terms of F_1-score) for $\texttt{Fusion}_{\texttt{Cq}}^{\texttt{all}}$ is worse than PB-MVBoost when we have less training examples ($n = 150$ and 200). This shows the proposed boosting based one-step algorithm PB-MVBoost is more stable and more effective for multiview learning.

From Table 1 and Figure 2, we can observe that MV-AdaBoost (where we have different distributions for each view over the learning sample) provides better results compared to other baselines in terms of accuracy but not in terms of F1-measure. On the other hand, MVBoost (where we have single global distribution over the learning sample but without learning weights over views) is better compared to other baselines in terms of F1-measure. Moreover, the performances of MVBoost first increases with an increase of the quantity of the training examples, then decreases. Whereas our algorithm PB-MVBoost provides the best results in terms of both accuracy and F1-measure, and leads to a monotonic increase of the performances with respect to the addition of labeled

[6] https://docs.scipy.org/doc/scipy/reference/optimize.minimize-slsqp.html

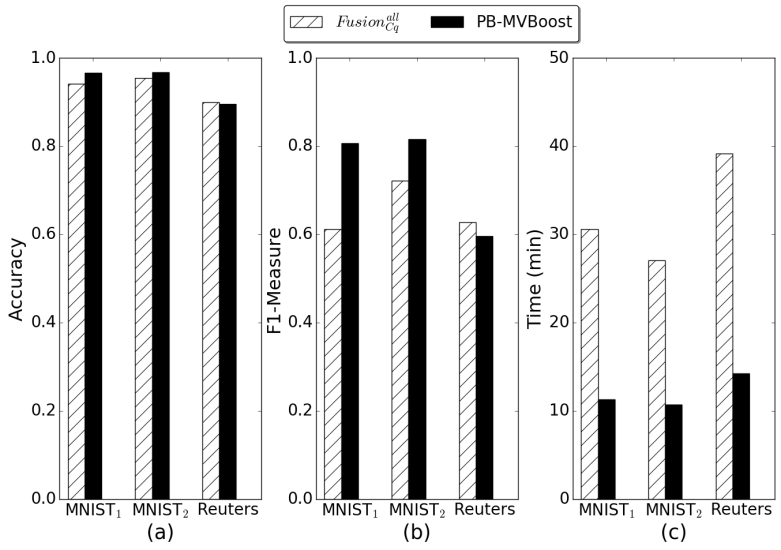
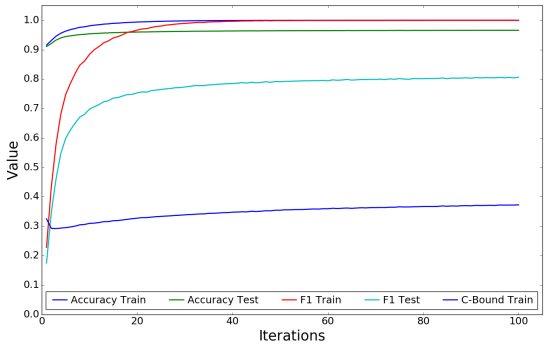


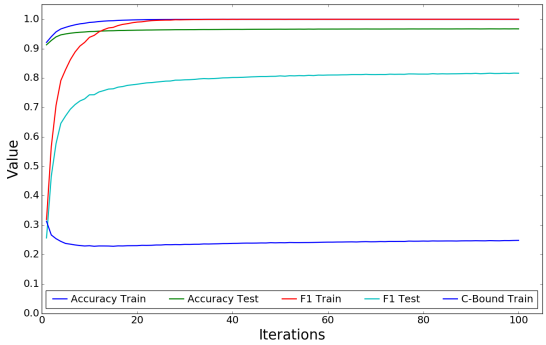
Figure 3: Comparison between $\texttt{Fusion}^{\texttt{all}}_{\texttt{Cq}}$ and `PB-MVBoost` in terms Accuracy (a), F1-Measure (b) and Time Complexity (c) for $n = 500$

examples. This confirms that by maintaining a single global distribution over the views and learning the weights over the views using a PAC-Bayesian framework, we are able to take advantage of different representations (or views) of the data.

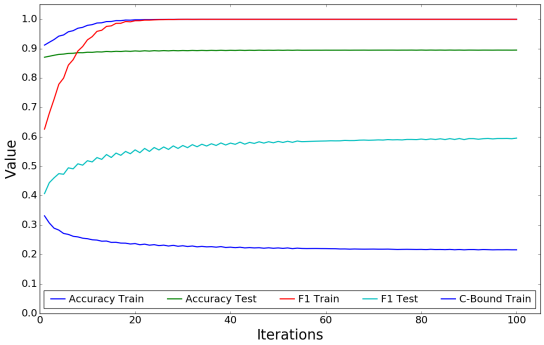
Finally, we plot behaviour of our algorithm `PB-MVBoost` over $T = 100$ iterations on Figure 4 for all the datasets. We plot accuracy and F1-measure of learned models on training and test data along with empirical multiview C-Bound on training data at each iteration of our algorithm. Over the iterations, the F1-measure on the test data keeps on increasing for all the datasets even if F1-measure and accuracy on the training data reach the maximal value. This confirms that our algorithm handles unbalanced data well. Moreover, the empirical multiview C-Bound (which controls the trade-off between accuracy and diversity between views) keeps on decreasing over the iterations. This validates that by combining the PAC-Bayesian framework with the boosting one, we can empirically ensure the view specific information and diversity between the views for multiview learning.



(a) $\texttt{MNIST}_1$



(b) $\texttt{MNIST}_2$



(c) `Reuters`

Figure 4: Plots for classification error and F1-measure on training and test data; and empirical multiview C-Bound on training data over the iterations for all datasets with $n = 500$.

6. Conclusion

In this paper, we provide a PAC-Bayesian analysis for a two-level hierarchical multiview learning approach with more than two views, when the model takes the form of a weighted majority vote over a set of functions/voters. We consider a hierarchy of weights modelled by distributions where for each view we aim at learning *i)* posterior Q_v distributions over the view-specific voters capturing the view-specific information and *ii)* hyper-posterior ρ_v distributions over the set of the views. Based on this strategy, we derived a general multiview PAC-Bayesian theorem that can be specialized to any convex function to compare the empirical and true risks of the stochastic multiview Gibbs classifier. We propose a boosting-based learning algorithm, called as `PB-MVBoost`. At each iteration of the algorithm, we learn the weights over the view-specific voters and the weights over the views by optimizing an upper-bound over the risk of the majority vote (the multiview C-Bound) that has the advantage of controlling a trade-off between accuracy and the diversity between the views. The empirical evaluation shows that `PB-MVBoost` leads to good performances and confirms that our two-level PAC-Bayesian strategy is indeed a nice way to tackle multiview learning. Moreover, we compare the effect of maintaining separate distributions over the learning sample for each view; single global distribution over views; and single global distribution along with learning weights over views on results of multiview learning. We show that by maintaining a single global distribution over the learning sample for all the views and learning the weights over the views is an effective way to deal with multiview learning. In this way, we are able to capture the view-specific information and control the diversity between the views. Finally, we compare `PB-MVBoost` with a two-step learning algorithm $\texttt{Fusion}^{\texttt{all}}_{\texttt{Cq}}$ which is based on PAC-Bayesian theory. We show that `PB-MVBoost` is more stable and computationally faster than $\texttt{Fusion}^{\texttt{all}}_{\texttt{Cq}}$.

For future work, we would like to specialize our PAC-Bayesian generalization bounds to linear classifiers [10] which will clearly open the door to derive theoretically founded multiview learning algorithms. We would also like to

extend our algorithm to *semi-supervised* multiview learning where one has access to an additional unlabeled data during training. One possible way is to learn a view-specific voter using pseudo-labels (for unlabeled data) generated from the voters trained from other views (as done for example in [47]). Another possible direction is to make use of unlabeled data while computing view-specific disagreement for optimizing multiview C-Bound. This clearly opens the door to derive theoretically founded algorithms for *semi-supervised* multiview learning using PAC-Bayesian theory. We would like to extend our algorithm to transfer learning setting where training and test data are drawn from different distributions. An interesting direction would be to bind the data distribution to the different views of the data, as in some recent zero-shot learning approaches [48]. Moreover, we would like to extend our work to the case of missing views or incomplete views e.g. Amini et al. [21] and Xu et al. [49]. One possible solution is to learn the view-specific voters using available view-specific training examples and adapt the distribution over the learning sample accordingly.

Acknowledgements

This work was partially funded by the French ANR project LIVES ANR-15-CE23-0026-03 and the "Région Rhône-Alpes".

Appendix

Appendix A. Mathematical Tools

Theorem 3 (Markov's ineq.). *For any random variable X s.t. $\mathbb{E}(|X|) = \mu$, for any $a > 0$, we have $\mathbb{P}(|X| \geq a) \leq \frac{\mu}{a}$.*

Theorem 4 (Jensen's ineq.). *For any random variable X, for any concave function g, we have $g(\mathbb{E}[X]) \geq \mathbb{E}[g(X)]$.*

Theorem 5 (Cantelli-Chebyshev ineq.). *For any random variable X s.t. $\mathbb{E}(X) = \mu$ and $\mathbf{Var}(X) = \sigma^2$, and for any $a > 0$, we have $\mathbb{P}(X - \mu \geq a) \leq \frac{\sigma^2}{\sigma^2 + a^2}$.*

Appendix B. Proof of C-Bound for Multiview Learning (Lemma 1)

In this section, we present the proof of Lemma 1, inspired by the proof provided by Germain et al. [11]. Firstly, we need to define the margin of the multiview weighted majority vote B_ρ and its first and second statistical moments.

Definition 1. *Let M_ρ is a random variable that outputs the margin of the multiview weighted majority vote on the example $(\mathbf{x}, y)$ drawn from distribution $\mathcal{D}$, given by*

$$M_\rho(\mathbf{x}, y) = \mathop{\mathbb{E}}_{v\sim\rho} \mathop{\mathbb{E}}_{h\sim Q_v} y\, h(x^v)\,.$$

The first and second statistical moments of the margin are respectively given by

$$\mu_1(M_\rho^{\mathcal{D}}) = \mathop{\mathbf{E}}_{(\mathbf{x},y)\sim\mathcal{D}} M_\rho(\mathbf{x}, y)\,, \tag{B.1}$$

and

$$\begin{aligned}\mu_2(M_\rho^{\mathcal{D}}) &= \mathop{\mathbf{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \big[M_\rho(\mathbf{x}, y)\big]^2 \\ &= \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}_X} y^2 \left[\mathop{\mathbb{E}}_{v\sim\rho} \mathop{\mathbb{E}}_{h\sim Q_v} h(x^v)\right]^2 = \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}_X} \left[\mathop{\mathbb{E}}_{v\sim\rho} \mathop{\mathbb{E}}_{h\sim Q_v} h(x^v)\right]^2.\end{aligned} \tag{B.2}$$

According to this definition, the risk of the multiview weighted majority vote can be rewritten as follows:

$$R_{\mathcal{D}}(B_\rho) = \mathop{\mathbb{P}}_{(\mathbf{x},y)\sim\mathcal{D}}\big(M_\rho(\mathbf{x}, y) \leq 0\big).$$

Moreover, the risk of the multiview Gibbs classifier can be expressed thanks to the first statistical moment of the margin. Note that in the binary setting where $y \in \{-1, 1\}$ and $h : \mathcal{X} \to \{-1, 1\}$, we have $\mathbb{1}_{[h(x^v)\neq y]} = \frac{1}{2}(1 - y\, h(x^v))$, and therefore

$$\begin{aligned}R_{\mathcal{D}}(G_\rho) &= \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{v\sim\rho} \mathop{\mathbb{E}}_{h\sim Q_v} \mathbb{1}_{[h(x^v)\neq y]} \\ &= \frac{1}{2}\left(1 - \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} \mathop{\mathbb{E}}_{v\sim\rho} \mathop{\mathbb{E}}_{h\sim Q_v} y\, h(x^v)\right) \\ &= \frac{1}{2}(1 - \mu_1(M_\rho^{\mathcal{D}}))\,.\end{aligned} \tag{B.3}$$

Similarly, the expected disagreement can be expressed thanks to the second statistical moment of the margin by

$$\begin{aligned} d_{\mathcal{D}}(\rho) &= \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}_{\mathcal{X}}} \mathop{\mathbb{E}}_{v\sim\rho} \mathop{\mathbb{E}}_{v'\sim\rho} \mathop{\mathbb{E}}_{h\sim Q_v} \mathop{\mathbb{E}}_{h'\sim Q_{v'}} \mathbb{1}_{[h(x^v)\neq h'(x^{v'})]} \\ &= \frac{1}{2}\left(1 - \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}_{\mathcal{X}}} \mathop{\mathbb{E}}_{v\sim\rho} \mathop{\mathbb{E}}_{v'\sim\rho} \mathop{\mathbb{E}}_{h\sim Q_v} \mathop{\mathbb{E}}_{h\sim Q_{v'}} h(x^v) \times h'(x^{v'})\right) \\ &= \frac{1}{2}\left(1 - \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}_{\mathcal{X}}} \Big[\mathop{\mathbb{E}}_{v\sim\rho} \mathop{\mathbb{E}}_{h\sim Q_v} h(x^v)\Big] \times \Big[\mathop{\mathbb{E}}_{v'\sim\rho} \mathop{\mathbb{E}}_{h'\sim Q_{v'}} h'(x^{v'})\Big]\right) \\ &= \frac{1}{2}\left(1 - \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}_{\mathcal{X}}} \left[\mathop{\mathbb{E}}_{v\sim\rho} \mathop{\mathbb{E}}_{h\sim Q_v} h(x^v)\right]^2\right) \\ &= \frac{1}{2}(1 - \mu_2(M_\rho^{\mathcal{D}}))\,. \end{aligned} \tag{B.4}$$

From above, we can easily deduce that $0 \leq d_{\mathcal{D}}(\rho) \leq 1/2$ as $0 \leq \mu_2(M_\rho^{\mathcal{D}}) \leq 1$. Therefore, the variance of the margin can be written as

$$\begin{aligned} \mathrm{Var}(M_\rho^{\mathcal{D}}) &= \mathop{\mathbf{Var}}_{(\mathbf{x},y)\sim\mathcal{D}}(M_\rho(\mathbf{x},y)) \\ &= \mu_2(M_\rho^{\mathcal{D}}) - (\mu_1(M_\rho^{\mathcal{D}}))^2. \end{aligned} \tag{B.5}$$

The proof of the C-bound

Proof. By making use of one-sided Chebyshev inequality (Theorem 5 of Appendix A), with $X = -M_\rho(\mathbf{x},y)$, $\mu = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}}(M_\rho(\mathbf{x},y))$ and $a = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} M_\rho(\mathbf{x},y)$, we have

$$\begin{aligned} R_{\mathcal{D}}(B_\rho) &= \mathop{\mathbb{P}}_{(\mathbf{x},y)\sim\mathcal{D}}\big(M_\rho(\mathbf{x},y) \leq 0\big) \\ &= \mathop{\mathbb{P}}_{(\mathbf{x},y)\sim\mathcal{D}}\left(-M_\rho(\mathbf{x},y) + \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} M_\rho(\mathbf{x},y) \geq \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} M_\rho(\mathbf{x},y)\right) \\ &\leq \frac{\mathop{\mathbf{Var}}_{(\mathbf{x},y)\sim\mathcal{D}}(M_\rho(\mathbf{x},y))}{\mathop{\mathbf{Var}}_{(\mathbf{x},y)\sim\mathcal{D}}(M_\rho(\mathbf{x},y)) + \left(\mathop{\mathbb{E}}_{(\mathbf{x},y)\sim\mathcal{D}} M_\rho(\mathbf{x},y)\right)^2} \\ &= \frac{\mathrm{Var}(M_\rho^{\mathcal{D}})}{\mu_2(M_\rho^{\mathcal{D}}) - \Big(\mu_1(M_\rho^{\mathcal{D}})\Big)^2 + \Big(\mu_1(M_\rho^{\mathcal{D}})\Big)^2} \\ &= \frac{\mathrm{Var}(M_\rho^{\mathcal{D}})}{\mu_2(M_\rho^{\mathcal{D}})} \end{aligned}$$

$$
\begin{aligned}
&= \frac{\mu_2(M_\rho^{\mathcal{D}}) - \Big(\mu_1(M_\rho^{\mathcal{D}})\Big)^2}{\mu_2(M_\rho^{\mathcal{D}})} \\
&= 1 - \frac{\Big(\mu_1(M_\rho^{\mathcal{D}})\Big)^2}{\mu_2(M_\rho^{\mathcal{D}})} \\
&= 1 - \frac{\Big(1 - 2\,R_{\mathcal{D}}(G_\rho)\Big)^2}{1 - 2\,d_{\mathcal{D}}(\rho)}\,.
\end{aligned}
$$

■

Appendix C. Proof of Lemma 2

We have

$$
\begin{aligned}
\mathop{\mathbb{E}}_{v\sim\rho}\mathop{\mathbb{E}}_{h\sim Q_v}\phi(h) &= \mathop{\mathbb{E}}_{v\sim\rho}\mathop{\mathbb{E}}_{h\sim Q_v}\ln e^{\phi(h)} \\
&= \mathop{\mathbb{E}}_{v\sim\rho}\mathop{\mathbb{E}}_{h\sim Q_v}\ln\left(\frac{Q_v(h)}{P_v(h)}\frac{P_v(h)}{Q_v(h)}e^{\phi(h)}\right) \\
&= \mathop{\mathbb{E}}_{v\sim\rho}\left[\mathop{\mathbb{E}}_{h\sim Q_v}\ln\left(\frac{Q_v(h)}{P_v(h)}\right) + \mathop{\mathbb{E}}_{h\sim Q_v}\ln\left(\frac{P_v(h)}{Q_v(h)}e^{\phi(h)}\right)\right].
\end{aligned}
$$

According to the Kullback-Leibler definition, we have

$$
\mathop{\mathbb{E}}_{v\sim\rho}\mathop{\mathbb{E}}_{h\sim Q_v}\phi(h) = \mathop{\mathbb{E}}_{v\sim\rho}\left[\mathrm{KL}(Q_v\|P_v) + \mathop{\mathbb{E}}_{h\sim Q_v}\ln\left(\frac{P_v(h)}{Q_v(h)}e^{\phi(h)}\right)\right].
$$

By applying Jensen's inequality (Theorem 4, in Appendix) on the concave function ln, we have

$$
\begin{aligned}
\mathop{\mathbb{E}}_{v\sim\rho}\mathop{\mathbb{E}}_{h\sim Q_v}\phi(h) &\le \mathop{\mathbb{E}}_{v\sim\rho}\left[\mathrm{KL}(Q_v\|P_v) + \ln\left(\mathop{\mathbb{E}}_{h\sim P_v}e^{\phi(h)}\right)\right] \\
&= \mathop{\mathbb{E}}_{v\sim\rho}\mathrm{KL}(Q_v\|P_v) + \mathop{\mathbb{E}}_{v\sim\rho}\ln\left(\frac{\rho(v)}{\pi(v)}\frac{\pi(v)}{\rho(v)}\mathop{\mathbb{E}}_{h\sim P_v}e^{\phi(h)}\right) \\
&= \mathop{\mathbb{E}}_{v\sim\rho}\mathrm{KL}(Q_v\|P_v) + \mathrm{KL}(\rho\|\pi) + \mathop{\mathbb{E}}_{v\sim\rho}\ln\left(\frac{\pi(v)}{\rho(v)}\mathop{\mathbb{E}}_{h\sim P_v}e^{\phi(h)}\right).
\end{aligned}
$$

Finally, we apply again the Jensen inequality (Theorem 4) on ln to obtain the lemma.

Appendix D. A Catoni-Like Theorem—Proof of Corollary 1

The result comes from Theorem 1 by taking $D(a,b) = \mathcal{F}(b) - Ca$, for a convex $\mathcal{F}$ and $C > 0$, and by upper-bounding $\mathop{\mathbb{E}}_{S\sim(\mathcal{D})^n} \mathop{\mathbb{E}}_{v\sim\pi} \mathop{\mathbb{E}}_{h\sim P_v} e^{nD(R_S(h),R_{\mathcal{D}}(h))}$. We consider $R_S(h)$ as a random variable following a binomial distribution of n trials with a probability of success $R(h)$. We have

$$
\begin{aligned}
&\mathop{\mathbb{E}}_{S\sim(\mathcal{D})^n} \mathop{\mathbb{E}}_{v\sim\pi} \mathop{\mathbb{E}}_{h\sim P_v} e^{n\, D(R_S(h),R_{\mathcal{D}}(h))} \\
&= \mathop{\mathbb{E}}_{S\sim(\mathcal{D})^n} \mathop{\mathbb{E}}_{v\sim\pi} \mathop{\mathbb{E}}_{h\sim P_v} e^{n\mathcal{F}(R_{\mathcal{D}}(h) - C\, n\, R_S(h))} \\
&= \mathop{\mathbb{E}}_{S\sim(\mathcal{D})^n} \mathop{\mathbb{E}}_{v\sim\pi} \mathop{\mathbb{E}}_{h\sim P_v} e^{n\mathcal{F}(R_{\mathcal{D}}(h))} \sum_{k=0}^{n} \Pr_{S\sim(\mathcal{D})^n}\left(R_S(h) = \frac{k}{n}\right) e^{-Ck} \\
&= \mathop{\mathbb{E}}_{S\sim(\mathcal{D})^n} \mathop{\mathbb{E}}_{v\sim\pi} \mathop{\mathbb{E}}_{h\sim P_v} e^{n\mathcal{F}(R_{\mathcal{D}}(h))} \sum_{k=0}^{n} \binom{n}{k} R_{\mathcal{D}}(h)^k (1 - R_{\mathcal{D}}(h))^{n-k} e^{-Ck} \\
&= \mathop{\mathbb{E}}_{S\sim(\mathcal{D})^n} \mathop{\mathbb{E}}_{v\sim\pi} \mathop{\mathbb{E}}_{h\sim P_v} e^{n\mathcal{F}(R_{\mathcal{D}}(h))} \left(R_{\mathcal{D}}(h)\, e^{-C} + (1 - R_{\mathcal{D}}(h))\right)^n .
\end{aligned}
$$

The corollary is obtained with $\mathcal{F}(p) = \ln \frac{1}{(1 - p[1 - e^{-C}])}$.

References

[1] P. K. Atrey, M. A. Hossain, A. El-Saddik, M. S. Kankanhalli, Multimodal fusion for multimedia analysis: a survey, Multimedia Syst. 16 (6) (2010) 345–379.

[2] S. Sun, A survey of multi-view machine learning, Neural Comput Appl 23 (7-8) (2013) 2031–2038.

[3] T. Baltrušaitis, C. Ahuja, L.-P. Morency, Multimodal machine learning: A survey and taxonomy, IEEE Transactions on Pattern Analysis and Machine Intelligence 41 (2) (2019) 423–443.

[4] A. Blum, T. M. Mitchell, Combining Labeled and Unlabeled Data with Co-Training, in: COLT, 1998, pp. 92–100.

[5] C. Snoek, M. Worring, A. W. M. Smeulders, Early versus late fusion in semantic video analysis, in: ACM Multimedia, 2005, pp. 399–402.

[6] D. A. McAllester, Some PAC-Bayesian theorems, Machine Learning 37 (1999) 355–363.

[7] O. Catoni, PAC-Bayesian supervised classification: the thermodynamics of statistical learning, Vol. 56, Inst. of Mathematical Statistic, 2007.

[8] M. W. Seeger, PAC-Bayesian generalisation error bounds for gaussian process classification, JMLR 3 (2002) 233–269.

[9] J. Langford, J. Shawe-Taylor, PAC-Bayes & margins, in: NIPS, MIT Press, 2002, pp. 423–430.

[10] P. Germain, A. Lacasse, F. Laviolette, M. Marchand, PAC-Bayesian learning of linear classifiers, in: ICML, 2009, pp. 353–360.

[11] P. Germain, A. Lacasse, F. Laviolette, M. Marchand, J. Roy, Risk bounds for the majority vote: from a PAC-Bayesian analysis to a learning algorithm, JMLR 16 (2015) 787–860.

[12] E. Parrado-Hernández, A. Ambroladze, J. Shawe-Taylor, S. Sun, PAC-bayes bounds with data dependent priors, JMLR 13 (2012) 3507–3531.

[13] P. Alquier, J. Ridgway, N. Chopin, On the properties of variational approximations of Gibbs posteriors, ArXiv e-prints.
URL `http://arxiv.org/abs/1506.04091`

[14] J.-F. Roy, M. Marchand, F. Laviolette, A column generation bound minimization approach with PAC-Bayesian generalization guarantees, in: Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, 2016, pp. 1241–1249.

[15] E. Morvant, A. Habrard, S. Ayache, Majority vote of diverse classifiers for late fusion, in: Structural, Syntactic, and Statistical Pattern Recognition - Joint IAPR International Workshop, S+SSPR 2014, Joensuu, Finland, August 20-22, 2014. Proceedings, 2014, pp. 153–162. `doi:`

10.1007/978-3-662-44415-3_16.
URL https://doi.org/10.1007/978-3-662-44415-3_16

[16] P. Germain, A. Habrard, F. Laviolette, E. Morvant, A new PAC-Bayesian perspective on domain adaptation, in: Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016, 2016, pp. 859–868.
URL http://jmlr.org/proceedings/papers/v48/germain16.html

[17] Y. Freund, Boosting a weak learning algorithm by majority, Inf. Comput. 121 (2) (1995) 256–285. doi:10.1006/inco.1995.1136.
URL http://dx.doi.org/10.1006/inco.1995.1136

[18] Y. Freund, R. E. Schapire, A decision-theoretic generalization of on-line learning and an application to boosting, J. Comput. Syst. Sci. 55 (1) (1997) 119–139. doi:10.1006/jcss.1997.1504.
URL http://dx.doi.org/10.1006/jcss.1997.1504

[19] R. E. Schapire, A brief introduction to boosting, in: Proceedings of the 16th International Joint Conference on Artificial Intelligence - Volume 2, IJCAI'99, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999, pp. 1401–1406.
URL http://dl.acm.org/citation.cfm?id=1624312.1624417

[20] R. E. Schapire, The Boosting Approach to Machine Learning: An Overview, Springer New York, New York, NY, 2003. doi:10.1007/978-0-387-21579-2_9.
URL https://doi.org/10.1007/978-0-387-21579-2_9

[21] M.-R. Amini, N. Usunier, C. Goutte, Learning from Multiple Partially Observed Views - an Application to Multilingual Text Categorization, in: NIPS, 2009, pp. 28–36.

[22] A. Goyal, E. Morvant, P. Germain, M. Amini, PAC-Bayesian Analysis for a Two-Step Hierarchical Multiview Learning Approach, in: Machine Learning

and Knowledge Discovery in Databases - European Conference, ECML PKDD 2017, Skopje, Macedonia, September 18-22, 2017, Proceedings, Part II, 2017, pp. 205–221. doi:10.1007/978-3-319-71246-8_13.
URL https://doi.org/10.1007/978-3-319-71246-8_13

[23] O. Chapelle, B. Schlkopf, A. Zien, Semi-Supervised Learning, 1st Edition, The MIT Press, 2010.

[24] L. I. Kuncheva, Combining Pattern Classifiers: Methods and Algorithms, Wiley-Interscience, 2004.

[25] O. Maillard, N. Vayatis, Complexity versus agreement for many views, in: ALT, 2009, pp. 232–246.

[26] Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, in: Proceedings of the IEEE, 1998, pp. 2278–2324.

[27] T. G. Dietterich, Ensemble methods in machine learning, in: Multiple Classifier Systems, 2000, pp. 1–15.

[28] M. Re, G. Valentini, Ensemble methods: a review, Advances in machine learning and data mining for astronomy (2012) 563–582.

[29] J. F. Díez-Pastor, J. J. Rodríguez, C. I. García-Osorio, L. I. Kuncheva, Diversity techniques improve the performance of the best imbalance learning ensembles, Information Sciences 325 (2015) 98 – 117. doi:https://doi.org/10.1016/j.ins.2015.07.025.
URL http://www.sciencedirect.com/science/article/pii/S0020025515005186

[30] G. Brown, L. I. Kuncheva, "good" and "bad" diversity in majority vote ensembles, in: N. El Gayar, J. Kittler, F. Roli (Eds.), Multiple Classifier Systems, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010, pp. 124–133.

[31] J.-C. Janodet, M. Sebban, H.-M. Suchier, Boosting Classifiers built from Different Subsets of Features, Fundamenta Informaticae 94 (2009) (2009)

1–21. doi:10.3233/FI-2009-131.
URL https://hal.archives-ouvertes.fr/hal-00403242

[32] S. Koço, C. Capponi, A boosting approach to multiview classification with cooperation, in: D. Gunopulos, T. Hofmann, D. Malerba, M. Vazirgiannis (Eds.), Machine Learning and Knowledge Discovery in Databases, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, pp. 209–228.

[33] M. Xiao, Y. Guo, Multi-view adaboost for multilingual subjectivity analysis, in: COLING 2012, 24th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, 8-15 December 2012, Mumbai, India, 2012, pp. 2851–2866.
URL http://aclweb.org/anthology/C/C12/C12-1174.pdf

[34] Z. Xu, S. Sun, An algorithm on multi-view adaboost, in: K. W. Wong, B. S. U. Mendis, A. Bouzerdoum (Eds.), Neural Information Processing. Theory and Algorithms, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010, pp. 355–362.

[35] J. Peng, C. Barbu, G. Seetharaman, W. Fan, X. Wu, K. Palaniappan, Shareboost: Boosting for multi-view learning with performance guarantees, in: Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2011, Athens, Greece, September 5-9, 2011, Proceedings, Part II, 2011, pp. 597–612. doi:10.1007/978-3-642-23783-6_38.
URL https://doi.org/10.1007/978-3-642-23783-6_38

[36] J. Peng, A. J. Aved, G. Seetharaman, K. Palaniappan, Multiview boosting with information propagation for classification, IEEE Transactions on Neural Networks and Learning Systems PP (99) (2017) 1–13. doi:10.1109/TNNLS.2016.2637881.

[37] S. Sun, J. Shawe-Taylor, L. Mao, PAC-Bayes analysis of multi-view learning, Information Fusion 35 (2017) 117–131.

[38] J. Shawe-Taylor, J. Langford, PAC-Bayes & margins, Advances in Neural Information Processing Systems 15 (2003) 439.

[39] D. A. McAllester, PAC-Bayesian stochastic model selection, in: Machine Learning, 2003, pp. 5–21.

[40] A. Lacasse, F. Laviolette, M. Marchand, P. Germain, N. Usunier, PAC-Bayes bounds for the risk of the majority vote and the variance of the Gibbs classifier, in: NIPS, 2006, pp. 769–776.

[41] J. Roy, F. Laviolette, M. Marchand, From pac-bayes bounds to quadratic programs for majority votes, in: Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011, 2011, pp. 649–656.

[42] M. D. Donsker, S. S. Varadhan, Asymptotic evaluation of certain markov process expectations for large time, i, Communications on Pure and Applied Mathematics 28 (1) (1975) 1–47.

[43] J. Langford, Tutorial on practical prediction theory for classification, JMLR 6 (2005) 273–306.

[44] M. Chen, L. Denoyer, Multi-view generative adversarial networks, in: Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2017, Skopje, Macedonia, September 18-22, 2017, Proceedings, Part II, 2017, pp. 175–188. doi:10.1007/978-3-319-71246-8_11.
URL https://doi.org/10.1007/978-3-319-71246-8_11

[45] E. Jones, T. Oliphant, P. Peterson, et al., SciPy: Open source scientific tools for Python, [Online; accessed ¡today¿] (2001–).
URL http://www.scipy.org/

[46] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos,

D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[47] X. Xu, W. Li, D. Xu, I. W. Tsang, Co-labeling for multi-view weakly labeled learning, IEEE Transactions on Pattern Analysis and Machine Intelligence 38 (6) (2016) 1113–1125.

[48] R. Socher, M. Ganjoo, C. D. Manning, A. Ng, Zero-shot learning through cross-modal transfer, in: Advances in Neural Information Processing Systems 26, 2013, pp. 935–943.

[49] C. Xu, D. Tao, C. Xu, Multi-view learning with incomplete views, IEEE Transactions on Image Processing 24 (12) (2015) 5812–5825.