Online Active Learning with Strong and Weak Annotators

Luigi Malagò[*]
Dipartimento di Informatica
Università degli Studi di Milano
malago@di.unimi.it

Nicolò Cesa-Bianchi
Dipartimento di Informatica
Università degli Studi di Milano
nicolo.cesa-bianchi@unimi.it

Jean-Michel Renders
Xerox Research Centre Europe
jean-michel.renders@xrce.xerox.com

Abstract

We study an online learning setting in which the learner actively decides when to query the binary label of the current data point. On each query, we assume the learner is given the option of obtaining the true label, which is drawn from the same source on which the learner's performance is measured, or a weak label, which is drawn from a noisy version of that source. Assuming weak labels are cheaper than true labels, the goal of the learner is to maximize predictive accuracy while keeping a desired proportion of true and weak labels. We propose an approach that chooses which label to query by learning the regions where the weak label is too noisy. This is done by using the information provided by the disagreement between weak and true labels. Preliminary experiments on a real-world dataset show evidence that our method of choosing labels performs better than less informed methods.

1 Introduction

Active learning [5, 10] is a powerful paradigm in which the learner is allowed to select, in a possibly adaptive manner, its own training data. This helps the learner to focus on the most informative data points, and thus achieve a better trade-off between the number of training labels and the predictive accuracy of the resulting model. We study this trade-off in the framework of online binary classification. Here the learner adaptively queries the binary labels of an observed sequence of data points, with the purpose of achieving small mistake rates across a given range of query rates.

Although the original motivation of active learning is that training labels are generally expensive to obtain, in many practical cases we may have more than one source of labels, with different costs per label. For example, consider the problem of identifying relevant documents in an industrial litigation. Since the number of potentially relevant document is typically large, training labels are provided by a pool of human annotators, who may include professionals with different experience and reliability. Whereas it is reasonable to take the annotations provided by an experienced attorney as ground truth, one may expect the labels of a unexperienced annotator to be of lower quality, and of corresponding lower cost.

This scenario can be easily cast in an online active learning protocol. The learner observes a sequence of data points (documents). Each time a new document is observed, the learner can either predict its label (as relevant or non-relevant), or issue a query to obtain the label from an annotator

[*]Current affiliation: Shinshu University, Nagano, Japan; email address malago@shinshu-u.ac.jp

and then use it for training. If a query is issued, the learner has the further option of querying a strong versus a weak annotator. We assume that strong annotators always return true (ground truth) labels, while weak annotators return noisy labels. If a query is not issued, the learner does not obtain any training signal for that data point. Now we are facing a more complex trade-off than that of standard online learning: the mistake rate must be minimized at a certain query rate where the fraction of strong labels does not exceed a given ratio. More in general, if annotators are associated with given costs per issued label, we may want to minimize the mistake rate at a given cost rate, leaving to the algorithm the choice of how often to query the weak and strong annotators.

Clearly enough, a learner is happier to pay a higher cost for a true label only when the corresponding weak label is expected to be very noisy. Now, it is plausible to assume that this noise rate should depend on the data point rather than being completely random. Namely, the unexperienced weak annotator is more likely to err on documents whose relevance is harder to judge. In other terms, the two annotators are expected to disagree more often as the data point approaches the Bayes decision surface for the relevance classification. Since we learn a decision surface for classifying relevance, we could set a threshold to decide when a weak label is too close to the Bayes surface (hence too noisy) and the true label should be queried instead. However, the behavior of the weak annotator noise around the Bayes surface may follow a complex pattern, not captured by a threshold. A more general approach, adopted in this paper, is that of training a classifier to predict when true and weak labels have a high chance of disagreement, and then use it to decide when to query the true label.

This plan appears to be worth implementing only if the overall cost of learning the disagreement between strong and weak annotators is not too big. However, notice that we do not want to reduce the original classification problem to the problem of classifying the disagreement, and then solve the latter to obtain a solution for the former. Instead we propose to exploit any information we can learn about the disagrement to obtain a gain in the way we mix weak and strong labels, compared to a non-informative random baseline. Moreover, we control the cost of learning by assuming that the additional cost of querying a weak label for the same document for which we already queried the true label can be disregarded (this is true when junior paralegals are way cheaper than experienced attorneys, which sounds like a reasonable hypothesis). In this case, we may query the weak label whenever we query the true label, and use the disagreement of this pair to feed a fresh training point to the algorithm for predicting disagreement labels. This new training point will hopefully help to pick more accurately the next true label to query, and so on. In the protocol we describe, the training set for the disagreement classifier is a subset of the points sampled by the relevance classifier, thus, we implicitly require that the points of disagreement become denser in the filtered stream, as a consequence of the model of the disagreement and of the query rule.

Note that learning the disagreement between the annotators can be naturally viewed as an online filtering problem (see [9] for a review), since the learner only observes false positive mistakes during the training of the disagreement classifier (when the prediction is for agreement, then the true label is not queried, and the disagreement between weak and true labels can not be assessed). Since online filtering can be implemented using the same algorithmic techniques as online active learning, cf. [2], our approach effectively reduces the problem of online active learning with strong and weak annotators to online active learning/filtering with a single annotator.

The problem of online active learning with multiple annotators has been previously studied in [6]. Unlike our setting, their model assumes that all annotators generate noisy labels, and the label with the smallest noise rate for the current instance is declared to be the true label of that instance. A setting similar to ours, in which the learner has access to a weak and a strong annotator, is the one in [12], albeit in a batch setting and only from a purely theoretical viewpoint. The work [13] studies a similar batch setting for image categorization, where annotators of different quality are available. Their criterion for choosing among annotators is based on the estimated information gain from each label. In other papers, e.g., [11], there is no access to the ground truth, which is estimated by taking a majority vote over the labels generated by the annotators. The work [7] adopts an exploration-exploitation approach to find out the best annotator for each instance in a batch active setting.

We give a more formal description of our approach in Section 2. The learning algorithm used in the implementation is illustrated in Section 3. In Section 4 we derive a concentration inequality for a mixed estimator based on regularized least squares. This allows us to derive an upper bound on the regret of a classifier that implements our approach. Finally, Section 5 describes the experimental setting and reports on preliminary the results over a real-world dataset.

2 Online Learning from Disagreement

In this section we describe our general approach for learning from weak and strong annotators. The main idea is to query the true label only when we predict it has a fair chance of disagreeing with the weak label. Notice that according to our disagreement model, presented in Section 4, the weak labels are perturbed versions of the strong labels over certain regions of the space of observations, so that we cannot expect to be able to classifying them with high accuracy, and in particular with precision higher than $1/2$. As discussed in the introduction, we assume that we can afford to query also the weak annotator whenever we decide to query the strong one. Moreover, since we would like to measure the mistake rate irrespectively of the number of queried labels, we evaluate the performance of the classifier by comparing the prediction on the current data point before the corresponding label (weak or true) is queried, with the ground truth label given by the strong annotator.

In the following, we use $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots \in \mathbb{R}^d$ to denote the stream of data points represented as d-dimensional feature (column) vectors $\boldsymbol{x}_t$. We write $y_t^w, y_t^s \in \{-1, +1\}$ to denote, respectively, the weak and the strong (ground truth) label associated with $\boldsymbol{x}_t$. Our goal is to control the mistakes of the relevance classifier on the true labels, irrespectively of which true and weak labels are observed, and under a given constraint on the number of queries to the weak and strong annotators. In a certain sense, for a fixed finite-time horizon, we would like to maximize the performance of the mixed classifier for any given query rate of sampled data points (weak or strong labels) and any given ratio between the number of weak and strong labels.

In the following we describe our approach to actively learning from strong and weak annotators. The main idea is that of training a relevance classifier via a disagreement classifier, whose goal is to predict the disagreement labels defined by the product $-y_t^w \times y_t^s \in \{-1, +1\}$.

Algorithm 1 Active learning from strong and weak annotators

```
1:  for each time step t = 1, 2, ... do
2:      observe instance x_t ∈ R^d
3:      predict true label y_t^s with ŷ ∈ {−1, +1} according to the relevance classifier
4:      if relevance classifier decides a query on x_t then
5:          query weak label y_t^w
6:          if disagreement classifier predicts a disagreement between y_t^w and y_t^s over x_t then
7:              query true label y_t^s
8:              update the relevance classifier using y_t^s
9:              update the disagreement classifier using −y_t^w × y_t^s
10:         else
11:             update the relevance classifier using y_t^w
12:         end if
13:     end if
14: end for
```

Algorithm 1 illustrates our general approach, in which we are simultaneously training a relevance and a disagreement classifier. At each time step t, we decide whether to make a query on the current point (line 4). In case the query is issued, first we query the weak annotator (line 5), and then we evaluate the disagreement classifier to decide whether to query also the strong annotator (line 6). If we obtain the true label (line 7), we use it to update the relevance classifier (line 8), and both labels to update the disagreement classifier (line 9). If we do not query the strong annotator, then we just update the relevance classifier using the label provided by the weak annotator (line 11). The approach we described does not depend on the specific choice of relevance and disagreement classifiers, and in particular on their query rules. Although it is reasonable to assume that regions of high disagreement are concentrated around the Bayes decision surface of the relevance classification, the shape of these regions could be complex. For this reason, for the disagreement classification problem we use a kernel-based classifier. A detailed implementation of a margin-based selective sampler classifier based on regularized least-squares is described in Algorithm 2, in Section 3.

By learning the disagreement, we want to query true labels only when they are likely to be useful. Hence, a reasonable baseline against which to measure our approach is a completely random choice of the same number of true labels.

3 Selective Sampling Algorithms

The approach to the problem of mixing strong and weak labels we described in the previous section is based on the training of two online selective sampling classifiers, for relevance and disagreement. However, our method does not depend on the choice of a specific selective sampler algorithm. In our experiments we used the SS algorithm from [2], a linear kernelizable algorithm for online learning based on the ridge regression model. Whenever a new label is queried, the classifier is updated according to the Regularized Least Squares (RLS) estimator

$$\boldsymbol{w}_{t+1} = \operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^d} \left(\sum_{s=1}^{t} \left(\boldsymbol{w}^\top \boldsymbol{x}_s - y_s\right)^2 + \|\boldsymbol{w}\|^2 \right) . \tag{1}$$

Let $S_t = \left[\boldsymbol{x}_1, \ldots, \boldsymbol{x}_t\right]$ be the input matrix and $\boldsymbol{y}_t = (y_1, \ldots, y_t)^\top$ the label vector. The closed-form expression for $\boldsymbol{w}_{t+1}$ is given by $\boldsymbol{w}_{t+1} = \left(I + S_t S_t^\top\right)^{-1} S_t \, \boldsymbol{y}_t$. In the case of selective sampling, the input matrix and the label vector only contain the queried examples. When dealing with unbalanced classes, it may be useful to add a weight parameter $c > 1$ to the estimator, to weight the points differently, according to their label.

Algorithm 2 The SS selective sampler

Parameter: $K \geq 0, c > 0$
Initialization: $\boldsymbol{w}_1 = 0, A_1 = I, N_0 = 0$
for each time step $t = 1, 2, \ldots$ **do**
  observe instance $\boldsymbol{x}_t \in \mathbb{R}^d$
  $\widehat{\Delta}_t = \boldsymbol{w}_t^\top \boldsymbol{x}_t$
  predict label $y_t \in \{-1, +1\}$ with $\operatorname{sgn}(\widehat{\Delta}_t)$
  if $\widehat{\Delta}_t^2 \leq \frac{K}{N_t} \ln t$ **then**
    Query label y_t and let $N_t = N_{t-1} + 1$
    $c_t = \begin{cases} 1 & \text{if } y_t = -1 \, , \\ c & \text{if } y_t = 1 \, . \end{cases}$
    $A_{t+1} = A_t + c_t^2 \boldsymbol{x}_t \boldsymbol{x}_t^\top$
    $\boldsymbol{w}_{t+1} = A_{t+1}^{-1}(A_t \boldsymbol{w}_t + c_t^2 y_t \boldsymbol{x}_t)$
  else
    $A_{t+1} = A_t, \boldsymbol{w}_{t+1} = \boldsymbol{w}_t$ and $N_t = N_{t-1}$
  end if
end for

Let $\widehat{\Delta}_t = \boldsymbol{w}_t^\top \boldsymbol{x}_t$ be the current margin, and let N_t be the number of queried labels up to time t. The query rule (i.e., the condition that triggers a query) for SS is based on the margin, which is compared at each time step to a threshold directly proportional to $\log t$ and inversely proportional to N_t, i.e.,

$$\widehat{\Delta}_t^2 \leq \frac{K \ln t}{N_t} \tag{2}$$

where the parameter $K > 0$ controls the query rate. The pseudo-code for SS appears in Algorithm 2.

The query rule is based on a probabilistic model for the generation of the labels Y_t: there exists a vector $\boldsymbol{u} \in \mathbb{R}^d$, with $|\Delta_t| \leq 1$ for all $t > 0$, such that $\mathbb{P}(Y_t = 1) = (1 + \Delta_t)/2$, where $\Delta_t = \boldsymbol{u}^\top \boldsymbol{x}_t$. Under this model, $\widehat{\Delta}_t$ is a statistical estimate of Δ_t. Now, the SS query rule fires when $\left|\widehat{\Delta}_t\right|$ is too small to guarantee that $\operatorname{sgn}(\Delta_t) = \operatorname{sgn}(\widehat{\Delta}_t)$ with high enough probability.

SS is a margin-based selective sampler algorithm that employs a RLS model. Other algorithms use the same estimator but different query rules. For instance, in the DGS algorithm [6] the estimated margin is compared against an adaptive data-dependent threshold. Other approaches to selective sampling have been proposed, where the query rule does not depend on the margin. For example in BBQ [8, 3], where a query is issued whenever the estimated variance of $\widehat{\Delta}_t$ is higher than a threshold polynomial in $1/t$.

4 Analysis of the Mixed Model

In this section we extend the selective sampling setting to a mixed model with two different label sources: the strong and the weak annotator. In order to bound both the number of mistakes and the number of queries of the learner, we need to derive a concentration inequality for the mixed estimator which allows us to control in probability the quantity $|\widehat{\Delta}_t - \Delta_t|$, for all $t > 0$. This step is propaedeutic to the analysis of different RLS selective samplers, where the mistake rate is usually decomposed over the time steps when the query is issued and it is not issued. Indeed, it can be proved that the former summation is controlled by a quantity proportional to $(\widehat{\Delta}_t - \Delta_t)^2$, cf. [8, 6].

Given a sequence $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots \in \mathbb{R}^d$ of observations, we assume that the $\{-1, +1\}$-valued labels associated with each $\boldsymbol{x}_t$ are realizations of the random variables Y_t^s (strong label) and Y_t^w (weak label). At each time step t the learner makes a prediction of the unknown strong label Y_t^s. When a query is issued, the relevance classifier is updated with either label Y_t^s or Y_t^w, according to the prediction of the disagreement classifier. When the strong label is queried, we assume the weak label comes for free, so that the disagreement classifier can be updated using the label $-Y_t^s Y_t^w$. We define the following probabilistic model for the strong label source,

$$\mathbb{P}(Y_t^s = 1) = \frac{1 + \Delta_t}{2}, \tag{3}$$

with $\Delta_t = \boldsymbol{u}^\top \boldsymbol{x}_t$, for all $t \geq 1$, where $\boldsymbol{u} \in \mathbb{R}^d$ is a fixed and unknown vector, such that $|\boldsymbol{u}^\top \boldsymbol{x}_t| \leq 1$, for all $t > 0$. Under this hypothesis it is easy to verify that $\mathbb{E}[Y_t^s] = \Delta_t$. We define the probabilistic model of the weak labeler as a perturbed version of the strong labeler, by restricting the perturbation over specific regions of the space of the observations. More specifically, let $0 \leq p < \frac{1}{2}$ be an unknown noise rate and let $\{\overline{\Omega}, \widetilde{\Omega}\}$ be an unknown partition of the domain of the observations $\Omega \subseteq \mathbb{R}^d$. For each $t > 0$ we have that

$$\mathbb{P}(Y_t^w = 1) = \begin{cases} \mathbb{P}(Y_t^s = 1) & \text{if } \boldsymbol{x}_t \in \overline{\Omega}\,, \\ (1-p)\mathbb{P}(Y_t^s = 1) + p\,\mathbb{P}(Y_t^s = -1) & \text{if } \boldsymbol{x}_t \in \widetilde{\Omega}\,. \end{cases} \tag{4}$$

Therefore

$$\mathbb{E}[Y_t^w] = \begin{cases} \boldsymbol{u}^\top \boldsymbol{x}_t & \text{if } \boldsymbol{x}_t \in \overline{\Omega}\,, \\ (1-2p)\boldsymbol{u}^\top \boldsymbol{x}_t & \text{if } \boldsymbol{x}_t \in \widetilde{\Omega}\,. \end{cases} \tag{5}$$

We use the notation $\Delta_t^w = \mathbb{E}[Y_t^w]$. Note that $\operatorname{sgn}\big(\Delta_t\big) = \operatorname{sgn}\big(\Delta_t^w\big)$ are Bayes optimal classifiers for the data model. The estimate at time t of $\boldsymbol{u}$ is $\boldsymbol{w}_t$, and it is computed via Regularized Least Squares (RLS) over a subset of the observations $(\boldsymbol{x}_t, Y_t)$. The observed labels Y_t have a different distribution according to which source is queried at time t. We predict Y_t^s according to $\operatorname{sgn}\big(\widehat{\Delta}_t\big)$, with $\widehat{\Delta}_t = \boldsymbol{w}_t^\top \boldsymbol{x}_t$. Let the variable Z_t be the output of the query rule for the relevance classifier at each time step, i.e., $Z_t = 1$ implies that a query is issued at time t, so that $N_T = \sum_{t=1}^T Z_t$. At each time step t for which $Z_t = 1$, the choice whether to query the strong or the weak label is determined by the output of a disagreement classifier. The probability of the two labelers to disagree over $\bar{\Omega}$ is given by $(1 - \Delta^2)/2$, while over $\widetilde{\Omega}$ it is increased by $p\Delta^2$, yet remaining lower than $1/2$ for $p < 1/2$. This implies that the disagreement classifier is trained with highly unbalanced classes. This is the reason why a weight parameter c is used to rebalance the labels and avoid constantly negative predictions of disagreement. Let the variable W_t be the output of the query rule for the disagreement classifier at each time step, so that for $Z_t = 1$, $W_t = 0$ implies that at time t a query is issued to the strong classifier, i.e., $Y_t = Y_t^s$. If $Z_t = 1$ and $W_t = 1$ a query is issued to the weak classifier and $Y_t = Y_t^w$. For $Z_t = 0$ we set $W_t = 0$.

We measure the performance of the learner using the notion of cumulative regret,

$$R_T = \sum_{t=1}^{T} \Big(\mathbb{P}\big(Y_t^s \widehat{\Delta}_t < 0\big) - \mathbb{P}\big(Y_t^s \Delta_t < 0\big) \Big) \tag{6}$$

computing the expected difference in the number of mistakes made by predicting each strong (ground truth) label Y_t^s using the estimate $\boldsymbol{w}_t$ versus the Bayes optimal classifier defined through $\boldsymbol{u}$. Moreover, we want to control the number of queried labels, of both weak and strong annotators. According to our scenario, we fix the cost of the strong label to be κ times higher than the cost of

the weak label. We can provide an upper bound for the total cost C_T of the labels, which allows to control the budget spent to query the strong and weak annotators. The total cost depends on the number of queries N_T of the relevance classifier and on the number of queries of the disagreement classifier $D_T = \sum_{t=1}^{T} Z_t(1 - W_t)$, i.e.,

$$C_T = \sum_{t=1}^{T} \Big(Z_t W_t + (1+\kappa) Z_t (1 - W_t) \Big) = \sum_{t=1}^{T} \Big(Z_t + \kappa\, Z_t (1 - W_t) \Big) = N_T + \kappa\, D_T \quad (7)$$

where the factor $1 + \kappa$ includes the cost of a query to the strong annotator and the fact that whenever the strong annotator is queried, the weak annotator is queried too.

Consider the confusion matrix for the disagreement classifier, where the positive disagreement label is given by a perturbed label, when $\boldsymbol{x}_t \in \widetilde{\Omega}$, and the positive prediction by $1 - W_t$. The false negatives (those points where the weak source is queried and the label is perturbed) affect the convergence of the RLS mixed estimator $\boldsymbol{w}_t$ to $\boldsymbol{u}$ by introducing a bias term. The presence of false negatives could be avoided by querying consistently the strong expert. However, this would increase the cost per query, since we suppose the labels of the strong expert to be more expensive. Given a fixed budget we have a tradeoff between number of labels queried and convergence of the RLS mixed estimator.

In order to characterize such bias as a function of the performance of the disagreement classifier, we proceed as follows. We adapt the approach in [1] to our context, and we derive a concentration inequality for the mixed estimator that explicitly depends on the confusion matrix of the disagreement classifier. Given any fixed sequence $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N_t}$, notice that the random variables Y_t^s and Y_t^w are $\mathcal{F}_t$-measurable, i.e., their value is determined at time t, given all previous observations up to time $t - 1$. We define the residual

$$\varepsilon_t = \begin{cases} Y_t^s - \boldsymbol{u}^\top \boldsymbol{x}_t & \text{if } W_t = 0 \vee \boldsymbol{x}_t \in \overline{\Omega}\,, \\ Y_t^w - (1-2p)\boldsymbol{u}^\top \boldsymbol{x}_t & \text{otherwise,} \end{cases}$$

and we observe that since $\mathbb{E}[\varepsilon_t \mid \mathcal{F}_{t-1}] = 0$, $\big\langle \epsilon_t \big\rangle_t$ is a martingale difference sequence. Because conditioned on the observations, the RLS estimator at time t does not depend on their ordering, we partition the matrix S_t associated with the queried points into two submatrices $\big[\overline{S}_t, \widetilde{S}_t\big]$. Let $\overline{S}_T = [\boldsymbol{x}_t]$, with $t \in 1, \ldots, T$ such that $Z_t = 1$ and $W_t = 0 \vee (W_t = 1 \wedge \boldsymbol{x}_t \in \overline{\Omega})$, and $\widetilde{S}_T = [\boldsymbol{x}_t]$, with $t \in 1, \ldots, T$ when $Z_t = 1$ and $W_t = 1 \wedge \boldsymbol{x}_t \in \widetilde{\Omega}$. Then, $\widetilde{S}_t$ is the matrix associated with the FNs of the disagreement classifier (the weak source is queried and the returned label is perturbed) and $\overline{S}_t$ the matrix associated with the remaining query points, i.e., the TPs, FPs and TNs (either the strong source is queried or the label of $\boldsymbol{x}_t$ is not perturbed). Using the same criterion, we partition the labels in $\boldsymbol{Y}$ into $\overline{\boldsymbol{Y}}_t$ and $\widetilde{\boldsymbol{Y}}_t$, and the vector of residuals $\boldsymbol{\epsilon}_t$ into $\overline{\boldsymbol{\epsilon}}_t$ and $\widetilde{\boldsymbol{\epsilon}}_t$.

Then $A_t = (S_t S_t^\top + \lambda \mathbb{1})$ can be written as $(\overline{S}_t \overline{S}_t^\top + \widetilde{S}_t \widetilde{S}_t^\top + \lambda \mathbb{1})$, where λ is the regularization coefficient. We have that

$$\begin{aligned} \boldsymbol{w}_t &= A_{t-1}^{-1} S_{t-1} \boldsymbol{Y}_{t-1} = A_{t-1}^{-1} S_{t-1} \left(\begin{bmatrix} \overline{\boldsymbol{\epsilon}}_{t-1} \\ \widetilde{\boldsymbol{\epsilon}}_{t-1} \end{bmatrix} + \begin{bmatrix} \overline{S}_{t-1} \\ 0 \end{bmatrix} \boldsymbol{u} + (1-2p) \begin{bmatrix} 0 \\ \widetilde{S}_{t-1} \end{bmatrix} \boldsymbol{u} \right) \\ &= A_{t-1}^{-1} S_{t-1} \left(\boldsymbol{\epsilon}_{t-1} + \begin{bmatrix} \overline{S}_{t-1} \\ \widetilde{S}_{t-1} \end{bmatrix} \boldsymbol{u} - 2p \begin{bmatrix} 0 \\ \widetilde{S}_{t-1} \end{bmatrix} \boldsymbol{u} \right) \\ &= A_{t-1}^{-1} S_{t-1} \boldsymbol{\epsilon}_{t-1} + A_{t-1}^{-1} (S_{t-1} S_{t-1}^\top \pm \lambda \mathbb{1}) \boldsymbol{u} - 2p A_{t-1}^{-1} S_{t-1} \begin{bmatrix} 0 \\ \widetilde{S}_{t-1} \end{bmatrix} \boldsymbol{u} \\ &= A_{t-1}^{-1} S_{t-1} \boldsymbol{\epsilon}_{t-1} + \boldsymbol{u} - \lambda A_{t-1}^{-1} \boldsymbol{u} - 2p A_{t-1}^{-1} \widetilde{S}_{t-1} \widetilde{S}_{t-1}^\top \boldsymbol{u}\,. \end{aligned}$$

By multiplying both members by $\boldsymbol{x}_t$ and rearranging items, we get

$$\begin{aligned} \boldsymbol{x}_t^\top \boldsymbol{w}_t - \boldsymbol{x}_t^\top \boldsymbol{u} &= \boldsymbol{x}_t^\top A_{t-1}^{-1} S_{t-1} \boldsymbol{\epsilon}_{t-1} - \lambda \boldsymbol{x}_t^\top A_{t-1}^{-1} \boldsymbol{u} - 2p \boldsymbol{x}_t^\top A_{t-1}^{-1} \widetilde{S}_{t-1} \widetilde{S}_{t-1}^\top \boldsymbol{u} \\ &= \langle \boldsymbol{x}_t, S_{t-1} \boldsymbol{\epsilon}_{t-1} \rangle_{A_{t-1}^{-1}} - \lambda \langle \boldsymbol{x}_t, \boldsymbol{u} \rangle_{A_{t-1}^{-1}} - 2p \langle \boldsymbol{x}_t^\top, \widetilde{S}_{t-1} \widetilde{S}_{t-1}^\top \boldsymbol{u} \rangle_{A_{t-1}^{-1}}\,. \end{aligned}$$

Then, by applying the Cauchy-Schwartz inequality, we have

$$\begin{aligned} |\widehat{\Delta}_t - \Delta_t| &\le \|\boldsymbol{x}_t\|_{A_{t-1}^{-1}} \left(\|S_{t-1} \boldsymbol{\epsilon}_{t-1}\|_{A_{t-1}^{-1}} + \lambda \|\boldsymbol{u}\|_{A_{t-1}^{-1}} + 2p \left\| \widetilde{S}_{t-1} \widetilde{S}_{t-1}^\top \boldsymbol{u} \right\|_{A_{t-1}^{-1}} \right) \\ &\le \|\boldsymbol{x}_t\|_{A_{t-1}^{-1}} \left(\|S_{t-1} \boldsymbol{\epsilon}_{t-1}\|_{A_{t-1}^{-1}} + \lambda^{1/2} \|\boldsymbol{u}\| + 2p \|\boldsymbol{u}\| \left\| \widetilde{S}_{t-1} \widetilde{S}_{t-1}^\top \right\| \left\| A_{t-1}^{-1} \right\| \right)\,. \end{aligned}$$

We upper bound the spectral norm $\left\|\widetilde{S}_{t-1}\widetilde{S}_{t-1}^{\top}\right\|$ with $\widetilde{N}_{t-1}$, i.e., the number of perturbed points sampled from the weak sampler, where $\widetilde{N}_T = \sum_{t=1}^{T} Z_t(1-W_t)\mathbb{1}\left\{\boldsymbol{x}_t \in \widetilde{\Omega}\right\}$. The term $\left\|A_{t-1}^{-1}\right\|$ equals $\sqrt{\lambda_{\max}\left(\left(A_{t-1}^{-1}\right)^{\top}A_{t-1}^{-1}\right)} = \lambda_{\max}(A_{t-1}^{-1}) = \lambda_{\min}(A_{t-1})^{-1} = (\lambda_{\min}(S_{t-1}S_{t-1}^{\top}) + \lambda)^{-1}$. We can now apply Lemma 9 in [1]. It follows that with probability at least $1-\delta$

$$|\widehat{\Delta}_t - \Delta_t| \leq \|\boldsymbol{x}_t\|_{A_{t-1}^{-1}} \left(\sqrt{8\ln\left(\frac{|A_{t-1}|^{1/2}}{\delta}\right)} + \|\boldsymbol{u}\|\left(\lambda^{1/2} + 2p\widetilde{N}_{t-1}(\lambda_{\min}(A_{t-1}) + \lambda)^{-1}\right)\right). \tag{8}$$

Equation (8) provides a concentration inequality for the mixed estimator which can be used to prove upper bounds for the regret in Equation (6). The inequality differs from the case when all labels come from the same (strong) expert by the term $2p\widetilde{N}_{t-1}\big(\lambda_{\min}(A_{t-1}) + \lambda\big)^{-1}$, introducing a bias in the estimator. In turn, the bias term depends both on the relevance classifier, determining which points enter in the estimator, and thus the minimum eigenvalues of A_{t-1}, and on the disagreement classifier, since $\widetilde{N}_{t-1}$ is proportional to the rate of FNs. Since a common approach to prove a bound for the regret requires a sum of the inequality over the rounds when the a label is asked, to obtain a meaningful result we need to ensure that the bias term does not grow linearly with time. Besides a proper selective sampling criterion for the relevance classifier which allow us to control the minimum eigenvalue of the design matrix, we also need to choose a disagreement classifier which ensures that the number of queried labels goes to zero fast enough, as N_t increases.

Notice that we have an explicit tradeoff between the bound over C_T and the bound over R_T, which depends on the performance of the disagreement classifier, indeed by minimizing D_T we reduce the total cost of the labels, however we also increase the probability of having large $\widetilde{N}_{t-1}$. Similarly, we can directly control $\widetilde{N}_{t-1}$ by increasing the positive rate of the disagrement classifier, penalizing the bound for the total cost.

5 Experiments

In this section we present a preliminary empirical evaluation of our system based on the Enron dataset, from the TREC 2010 Legal Track test collection [4]. This dataset was generated from a set of text documents, emails and text attachments, which are labeled as relevant or not with respect to 12 different topics (tasks) by humans with different levels of expertise. More specifically, labels were assigned to documents by a team of paralegal experts (the weak labelers) and by a team of legals experts (the strong labelers). The dataset contains 685,592 documents (emails or attachments), encoded according to a bag-of-words model based on a dictionary of 280,010 terms. Only a subset of the documents is classified by both legals and paralegals. For this reasons we focused on tasks 301 and 303, which contain the largest number of documents each having both a strong and a weak label (7,345 and 7,578 documents, respectively). A problem that we had to cope with is that, on these tasks, there is not a lot of disagreement between legal and paralegal labels. This implies that the gap in performance between the classifiers trained only on weak or strong labels is not very large.

Figure 1 compares the performance of our system (denoted Online Learning) on tasks 301 and 303 against some baselines. We used the SS selective sampler to implement both the relevance and the disagreement classifiers, as on our datasets SS performed better than BBQ. We ran experiments using the weighted variant of RLS. More specifically, we assigned a weight $c = 3$ to queried documents that were found relevant, and a weight $c = 8$ to documents for which we queried both labels and found them disagreeing. Indeed, the fraction of positive labels for the paralegal (legal) in tasks 301 and 303 is $11.08\%(11.28)\%$ and $14.58\%(12.66\%)$, respectively. The larger weight $c = 8$ is due to the fact, already mentioned in Section 4, that disagreement is caused by the presence of extra noise in the weak labels, and the rate of this extra noise is rather small.

Since learning the disagreement is not an easy task in general, whenever the disagreement classifier learns slower, the performance of the relevance classifier are affected, since the disagreement label is used to decide which expert to query. In order to tame this issue, we primed the disagreement classifier with a seed set of disagreement points chosen from regions where the weak labels are

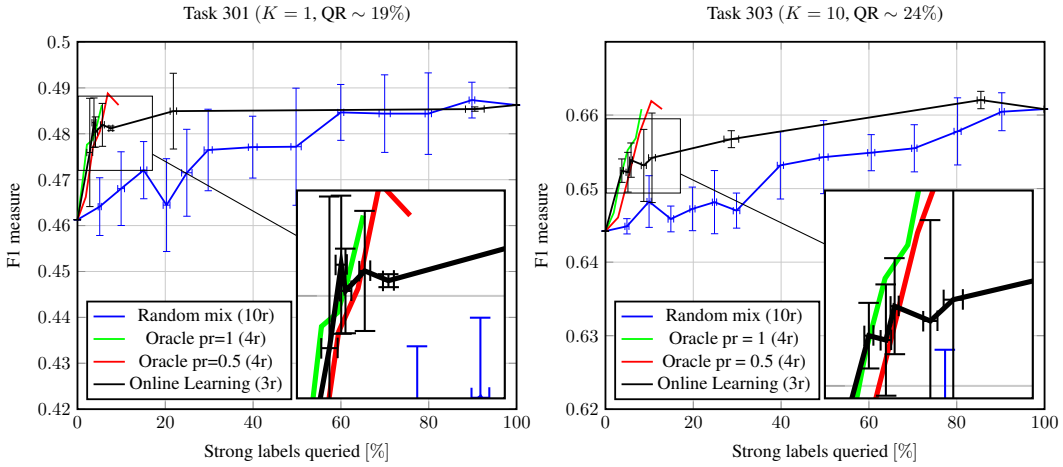

Figure 1: Experimental results on the Enron dataset for tasks 301 and 303. QR denotes the average query rate of the relevance classifier, controlled by the parameter K of the SS selective sampler.

perturbed. Note that a disagreement label is different from a strong label, which means that we can not use the seed set to train the relevance classifier directly.

We compared the performance of our algorithm with that of a random mix implemented as follows: whenever the relevance classifier issues a query, a strong label is returned with probability p and a weak label with probability $1 - p$, where p is the desired mix rate. This baseline shows the performance of an uninformed strategy that only knows the desidered mix between weak and strong labels. We also ran two oracle versions of disagreement classifier (precision 1 and 0.5, respectively). The first classifier knows exactly all disagreement labels. The second one classifies detects perfectly when the strong and the weak label agree, but has only a 50% chance of recognizing documents whose strong and weak labels disagree. We view this as an approximation of the Bayes optimal classifier when $\mathbb{P}\big(Y = 1 \mid X \in \overline{\Omega}\big) = 0$ and $\mathbb{P}\big(Y = 1 \mid X \in \widetilde{\Omega}\big) = p \leq \frac{1}{2}$.

In Figure 1 we selected the parameter of the relevance classifier so that the overall query rate was 19% for task 301 and 24% for task 303. These query rates maximize the performance gap between querying strong labels only as opposed to querying weak labels only. The gap is exactly the difference in F1 measure between the 0% tick and the 100% tick in the percentage of queried strong labels. In each pane of the figure, the blue wiggly line shows the performance of the random mix as more strong labels are allowed in the mix. The black line shows the performance of our system using the disagreement classifier. The difference is not impressing, but consistent. Also, at low rates of strong labels our system closely track the performance of the 0.5-precision classifier, which in turn is not so different from the performance of the 1-precision classifier.

6 Conclusion

In this work we proposed an online active learning setting where there are two sources of labels: strong (ground truth) labels and weak (noisy) labels. We devised a methodology to analyze problems in this setting, and approached the task of the learner by introducing a classifier for the disagreement between the annotators. We described a concrete implementation of our methodology based on regularized least squares, and reported some preliminary although encouraging experimental results. The trade-off underlying the problem of learning with weak and strong annotators is delicate: if the weak source is too weak, then weak labels are not useful, if it is too strong, then one can learn from weak labels only. In real-world data, this trade-off is hard to achieve, and thus learning the disagreement could also be used to detect these extreme cases, where no mixing of weak and strong sources can be beneficial.

References

[1] Y. Abbasi-yadkori, D. Pál, and C. Szepesvári. Improved algorithms for linear stochastic bandits. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 2312–2320. Curran Associates, Inc., 2011.

[2] G. Cavallanti, N. Cesa-Bianchi, and C. Gentile. Learning noisy linear classifiers via adaptive and selective sampling. *Machine Learning*, 83(1):71–102, 2011.

[3] N. Cesa-Bianchi, C. Gentile, and F. Orabona. Robust bounds for classification via selective sampling. In *Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009, Montreal, Quebec, Canada, June 14-18, 2009*, pages 121–128, 2009.

[4] G. V. Cormack, M. R. Grossman, B. Hedin, and D. W. Oard. Overview of the TREC 2010 legal track, 2010. Dataset available at http://trec-legal.umiacs.umd.edu.

[5] S. Dasgupta. Two faces of active learning. *Theor. Comput. Sci.*, 412(19):1767–1781, Apr. 2011.

[6] O. Dekel, C. Gentile, and K. Sridharan. Robust selective sampling from single and multiple teachers. *Journal of Machine Learning Research*, 13:2655–2697, 2012.

[7] P. Donmez, J. Carbonell, and J. Schneider. Efficiently learning the accuracy of labeling sources for selective sampling. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 259–268. ACM, 2009.

[8] F. Orabona and N. Cesa-Bianchi. Better algorithms for selective sampling. In *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011*, pages 433–440, 2011.

[9] D. Sculley. *Advances in online learning-based spam filtering*. PhD thesis, Tufts University, August 2008.

[10] B. Settles. *Active Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool, 2012.

[11] V. Sheng, F. Provost, and P. Ipeirotis. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 614–622. ACM, 2008.

[12] R. Urner, S. Ben-David, and O. Shamir. Learning from weak teachers. In *Proceedings of the 15th International Conference on Artificial Intelligence and Statistics, AISTATS*, pages 1252–1260, 2012.

[13] S. Vijayanarasimhan and K. Grauman. Cost-sensitive active visual category learning. *International Journal of Computer Vision*, 91(1):24–44, 2011.