Positive unlabeled learning via wrapper-based adaptive sampling

Pengyi Yang[1], Wei Liu[2], Jean Yang[1]
[1]Charles Perkins Centre, School of Mathematics and Statistics, University of Sydney, Australia
[2]Advanced Analytics Institute, University of Technology Sydney, Australia
{pengyi.yang, jean.yang}@sydney.edu.au; wei.liu@uts.edu.au

Abstract

Learning from positive and unlabeled data frequently occurs in applications where only a subset of positive instances is available while the rest of the data are unlabeled. In such scenarios, often the goal is to create a discriminant model that can accurately classify both positive and negative data by modelling from labeled and unlabeled instances. In this study, we propose an adaptive sampling (AdaSampling) approach that utilises prediction probabilities from a model to iteratively update the training data. Starting with equal prior probabilities for all unlabeled data, our method "wraps" around a predictive model to iteratively update these probabilities to distinguish positive and negative instances in unlabeled data. Subsequently, one or more robust negative set(s) can be drawn from unlabeled data, according to the likelihood of each instance being negative, to train a single classification model or ensemble of models.

1 Introduction

Traditional supervised learning algorithms require labels of both positive and negative instances for building a binary classification model. In various applications, however, obtaining negative data could be difficult especially in domains that lack precise knowledge and definition of negative instances [Calvo *et al.*, 2007]. For example, defining genes that are unrelated to a disease is difficult as there can be genes that are unknown to be related to the disease and therefore contaminating the negative sample set. In such cases, positive unlabeled learning techniques are proposed to model from labeled positive instances augmented with unlabeled instances comprising both unknown positive and negative instances [Denis *et al.*, 2005; Li *et al.*, 2009].

Current techniques proposed for positive unlabeled learning can roughly be categorised into (1) heuristic, (2) bias-based, (3) one-class, and (4) bootstrap sampling approaches. Heuristic approaches often partition the learning process into two steps where negative instances are firstly identified by using heuristic methods such as information retrieval techniques [Li and Liu, 2003], Bayesian methods [Liu *et al.*, 2003], Expectation Maximization algorithms [Nigam *et al.*, 1998; Liu *et al.*, 2002], or domain knowledge [Yang *et al.*, 2012], and a final classification model is created using labeled positive instances and unlabeled negative instances identified in the first step. One key disadvantage of most heuristic approaches is the requirement of a pre-defined threshold to determine either to include or exclude a potential negative instance obtained from unlabeled data for model training. Finding the optimal threshold for negative instance selection could be data dependent and often influence greatly on the accuracy of the final model. The lack of formality for many heuristic methods also limited their generality.

Bias-based approaches, on the other hand, treat all unlabeled data as negative instances and employ a traditional learning procedure, except a "bias" is introduced to weight the classification model and/or the cost function towards positive class in that predictions made from unlabeled data are penalised less for been positive to account for unknown positive instances been labeled as negatives [Elkan and Noto, 2008]. This approach was utilised for learning biased SVM [Liu *et al.*, 2003], logistic model [Lee and Liu, 2003] and Bayes classifier [Nigam *et al.*, 2000]. Elkan and Noto [Elkan and Noto, 2008] have subsequently formulated the bias-based approach in a general framework that could be used with a large selection of classification models. However, bias-based approaches often rely on training data for estimating the "bias" to be applied for model correction. Hence, part of the training data need to be utilised for bias estimation or a cross-validation procedure is conducted. This is unattractive especially when the training data are limited because the estimation of the "bias" coefficient could deviate significantly, causing under- or over-correction and thus poor classification model.

Alternatively, positive unlabeled learning can also be formulated as a one-class learning problem where only positive labels are used for training a classification model [Li *et al.*, 2011]. This has give rise to a set of methods that adhere to the same principle of one-class learning but tailored for positive unlabeled learning [Denis *et al.*, 2002; Calvo *et al.*, 2007]. Given the similarity between one-class learning and positive unlabeled learning, many more one-class learning algorithms can also be easily tuned for positive unlabeled learning [Khan and Madden, 2014]. The drawback of adjusting one-class learning methods for positive unlabeled learning however is that they generally rely on generative classification models and ignore unlabeled data.

Therefore, more labeled positive instances may be required to achieve comparable performance to methods that effectively utilise both labeled and unlabeled instances.

Recently, methods based on bootstrap sampling have been proposed to create ensemble of models for positive unlabeled learning [Mordelet and Vert, 2014; Claesen *et al.*, 2015; Yang *et al.*, 2016]. In such settings, unlabeled instances are treated as negatives and bootstrap sampling are performed on unlabeled instances and subsequently concatenated with labeled positive instances to train base classifiers that form an ensemble. The key idea is to take advantage of instability of predictions, caused potentially by the random inclusion of unlabeled positive instances, and aggregate a final stable prediction. These bootstrap sampling approaches adhere to and exploit the advantages of bagging-like procedure [Breiman, 1996]. Nevertheless, since all unlabeled instances are treated as negative data, the random subsets sampled from unlabeled data still contains incorrect labels. The base classifiers, therefore, still suffer from unwanted label noise which propagates and affects the performance of the final ensemble model.

This study extends on bootstrap sampling approaches by introducing a novel wrapper-based adaptive sampling (AdaSampling) procedure. Similar to previously proposed methods [Mordelet and Vert, 2014; Claesen *et al.*, 2015], initially all unlabeled instances are treated as negative examples and are equally likely to be selected for model training. Then AdaSampling differs from bootstrap sampling approaches in that the procedure "wraps" around a classification model and prediction uncertainties of unlabeled instances from the model are incorporated for each subsequent iterations of sampling to reduce the probability of selecting potential unknown positive instances as negative examples for model training (Figure 1). AdaSampling is generic and can be applied with any learning model that outputs classification probabilities. It does not require additional training data for bias estimation nor a heuristic procedure for selecting negative instances. This allows both labeled and unlabeled data to be utilised for model training without introducing any prediction bias in the learning model. Furthermore, AdaSampling approach can be easily extended for ensemble learning where different negative instances are drawn and combined with labeled positive to create diverse base classifiers. This enables ensemble model with AdaSampling to make effective usage of unlabeled data while also prevent potential noise propagation from applying bootstrap sampling directly to all unlabeled data.

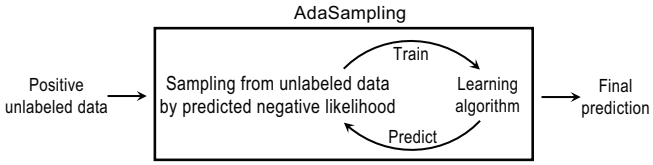

Figure 1: Schematic illustration of AdaSampling procedure.

Our empirical studies suggest that AdaSampling requires very few iterations to accurately distinguish unlabeled positive and negative instances even with very high positive to negative instance ratio in unlabeled data. We next compared AdaSampling based single and ensemble models with the state-of-the-art bias-based approach and bootstrap sampling approach using Support Vector Machine (SVM) and k-Nearest Neighbours (kNN) and a panel of evaluation metrics on several real-world datasets with different ratios of unlabeled positive instances. Our experimental results demonstrate that models trained with AdaSampling technique significantly improve on classification for both SVM and kNN, and their performance compared favourably to state-of-the-art methods. Together, this study offers a conceptually simple, flexible, yet powerful approach for positive unlabeled learning.

2 Methods

2.1 AdaSampling

It is helpful to view the positive unlabeled learning problem as discriminating positive and negative instances from a dataset with negative examples been contaminated by hidden positive instances. In this formulation, the problem of positive unlabeled learning is reduced to a traditional classification problem where all unlabeled instances are treated as negative instances. Let us denote the labeled instances as $\mathbf{L}$ (i.e. $y = 1$), unlabeled instances as $\mathbf{U}$ and assume that there are m labeled and n unlabeled instances. In positive unlabeled learning where the label information is not available for $\mathbf{U}$, a traditional classifier can be trained by labeling all $\mathbf{U}$ as negatives (i.e. y=0), sampling with equal probability from all instances in $\mathbf{U}$ and combining them with $\mathbf{L}$:

$$[\mathbf{D}^0, \boldsymbol{y}] = [\mathbf{L}, \boldsymbol{y} = \mathbf{1}] \cup [\mathbf{S}^0, \boldsymbol{y} = \mathbf{0}] \tag{1}$$

where $\mathbf{S}^0 \subset \mathbf{U}$ and the superscript 0 of $\mathbf{S}^0$ and $\mathbf{D}^0$ is the iteration index (cf., Alg. 1). A classification model can be fitted using this training dataset (Eq. 1):

$$p(y|\boldsymbol{x}) = h_\theta(\boldsymbol{x}; [\mathbf{D}^0, \boldsymbol{y}]) \tag{2}$$

The above classification model (Eq. 2) is the starting point of AdaSampling where an instance $s \in \mathbf{U}$ will be selected to be a negative example for subsequent training with a probability of $1 - p(y = 1|s)$. The training data can be updated after a set of instances $\mathbf{S}^i \subset \mathbf{U}$ are selected (with replacement):

$$[\mathbf{D}^i, \boldsymbol{y}] = [\mathbf{L}, \boldsymbol{y} = \mathbf{1}] \cup [\mathbf{S}^i, \boldsymbol{y} = \mathbf{0}] \tag{3}$$

where i indexes iteration of sampling. By utilising the prediction probabilities of the fitted model on instances from $\mathbf{U}$, AdaSampling can update the training dataset $[\mathbf{D}^i, \boldsymbol{y}]$ (Eq. 3) to reduce the chance of selecting unlabeled positive instances as negative training examples. This will lead to updated prediction probabilities $p^i(y|\boldsymbol{x}_1), ..., p^i(y|\boldsymbol{x}_{m+n})$ of all instances including the n instances of $\mathbf{U}$ from which AdaSampling can be repeatedly utilised to update the training dataset.

2.2 AdaSampling for classification

Single model

AdaSampling can be used with various classification algorithms for positive unlabeled learning. Akin to wrapper based feature selection [Kohavi and John, 1997], here the procedure wraps around a classification model to iteratively prioritise

negative instances from unlabeled data. This procedure therefore tunes the data with respect to a given predictive model. Criteria such as the following:

$$\frac{1}{m+n}\sum_{j=1}^{m+n}\left|p^i(y|\boldsymbol{x}_j)-p^{i-1}(y|\boldsymbol{x}_j)\right|<\varepsilon \qquad (4)$$

can be utilised for termination of the iterative sampling and the predicted probabilities from the final iteration can be used for classification of instances from both the labeled and unlabeled data. Here $j=1,...,m+n$ is the index of all instances in the data. We set ε to be 0.01, requiring smaller than 1% change in mean prediction probabilities of all instances for the process to terminate.

Algorithm 1 summarises this procedure in pseudocode.

Algorithm 1: AdaSampling for single model

Data: Positive unlabeled data $\mathbf{L}$ and $\mathbf{U}$
Result: Predicted label of all instances $\boldsymbol{y}$

$\boldsymbol{p}^0 \leftarrow \mathbf{1}$; // initialise probability vector for all instances
$\mathbf{S}^0 \leftarrow$ sampling($\mathbf{U}$, $\boldsymbol{p}^0_{\mathbf{U}}$); // select negative instances with probability $\boldsymbol{p}^0$ from $\mathbf{U}$
$[\mathbf{D}^0, \boldsymbol{y}] \leftarrow [\mathbf{L}, \boldsymbol{y}=\mathbf{1}] \cup [\mathbf{S}^0, \boldsymbol{y}=\mathbf{0}]$; // label initial training data
$i \leftarrow 0$;
do
  $i \leftarrow i+1$;
  // train a model and classify all instances
  $p^i(y|\boldsymbol{x}_1),...,p^i(y|\boldsymbol{x}_{m+n}) \leftarrow$ predict($h_\theta(\boldsymbol{x}; [\mathbf{D}^{i-1}, \boldsymbol{y}]), \mathbf{L} \cup \mathbf{U}$);
  // adaptive sampling from $\boldsymbol{x} \in \mathbf{U}$ w.r.t updated probabilities
  $\mathbf{S}^i \leftarrow$ sampling($\mathbf{U}$, $\boldsymbol{p}^i_{\mathbf{U}}$);
  $[\mathbf{D}^i, \boldsymbol{y}] \leftarrow [\mathbf{L}, \boldsymbol{y}=\mathbf{1}] \cup [\mathbf{S}^i, \boldsymbol{y}=\mathbf{0}]$;
while *Eq.* $4 > \varepsilon$;
$\boldsymbol{y} \leftarrow$ classify($h_\theta(\boldsymbol{x}; [\mathbf{D}^i, \boldsymbol{y}]), \mathbf{L} \cup \mathbf{U}$);

Ensemble of models

Alternatively, we can apply weighted sampling from $\mathbf{U}$ using $\boldsymbol{p}^i_{\mathbf{U}}$, from the last AdaSampling interaction, as weights to create different negative subsets $\mathbf{S}^*_k$ $(k=1,...K)$. This allows for creating base models b_k $(k=1,...K)$ each trained on a different training set $[\mathbf{L}, \boldsymbol{y}=\mathbf{1}] \cup [\mathbf{S}^*_k, \boldsymbol{y}=\mathbf{0}]$ for ensemble prediction. The key advantage of this procedure for ensemble learning is that prediction uncertainties of U are exploited multiple times to make effective usage of instances in U, avoiding potential high variance introduced by training a single model for classification. Algorithm 2 summarises AdaSampling based ensemble learning procedure in pseudocode.

Algorithm 2: AdaSampling for ensemble of models

Data: Positive unlabeled data $\mathbf{L}$ and $\mathbf{U}$
Result: Predicted label of all instances $\boldsymbol{y}$

$\boldsymbol{p}^0 \leftarrow \mathbf{1}$;
$\mathbf{S}^0 \leftarrow$ sampling($\mathbf{U}$, $\boldsymbol{p}^0_{\mathbf{U}}$);
$[\mathbf{D}^0, \boldsymbol{y}] \leftarrow [\mathbf{L}, \boldsymbol{y}=\mathbf{1}] \cup [\mathbf{S}^0, \boldsymbol{y}=\mathbf{0}]$;
$i \leftarrow 0$;
do
  $i \leftarrow i+1$;
  $p^i(y|\boldsymbol{x}_1),...,p^i(y|\boldsymbol{x}_{m+n}) \leftarrow$ predict($h_\theta(\boldsymbol{x}; [\mathbf{D}^{i-1}, \boldsymbol{y}]), \mathbf{L} \cup \mathbf{U}$);
  $\mathbf{S}^i \leftarrow$ sampling($\mathbf{U}$, $\boldsymbol{p}^i_{\mathbf{U}}$);
  $[\mathbf{D}^i, \boldsymbol{y}] \leftarrow [\mathbf{L}, \boldsymbol{y}=\mathbf{1}] \cup [\mathbf{S}^i, \boldsymbol{y}=\mathbf{0}]$;
while *Eq.* $4 > \varepsilon$;
// create an ensemble of models
$h^E_\theta \leftarrow Null$;
for $k \in 1...K$ **do**
  $\mathbf{S}^*_k \leftarrow$ sampling($\mathbf{U}$, $\boldsymbol{p}^i_{\mathbf{U}}$);
  $[\mathbf{D}^k, \boldsymbol{y}] \leftarrow [\mathbf{L}, \boldsymbol{y}=\mathbf{1}] \cup [\mathbf{S}^*_k, \boldsymbol{y}=\mathbf{0}]$;
  $h^E_\theta \leftarrow h^E_\theta \bigcup h^{b_k}_\theta(\boldsymbol{x}; [\mathbf{D}^k, \boldsymbol{y}])$;
end
$\boldsymbol{y} \leftarrow$ classify($h^E_\theta; \mathbf{L} \cup \mathbf{U}$);

3 Experimental procedure

This section summarises the datasets used for evaluation and describe the performance evaluation strategy.

3.1 Synthetic datasets

Synthetic datasets were used to analyse the behaviour of AdaSampling. In particular, we simulated 100 labeled positive instances (denoted as $\mathbf{L}^+$) from a normal distribution $\mathcal{N}(6,\ 1)$ and 300 unlabeled negative instances (denoted as $\mathbf{U}^-$) from a normal distribution $\mathcal{N}(4,\ 1)$. Then, 50 or 100 unlabeled positive instances (denoted as $\mathbf{U}^+$) were added into the data to simulate "easy" and "hard" scenarios, respectively. Together, this gives two synthetic datasets where in the "easy" scenario there are 100 labeled positive instances and 350 unlabeled instances (a ratio of 1:0.5:3 for $\mathbf{L}^+$, $\mathbf{U}^+$ and $\mathbf{U}^-$), and in the "hard" scenario there are 100 labeled positive instances and 400 unlabeled instances (a ratio of 1:1:3 for $\mathbf{L}^+$, $\mathbf{U}^+$ and $\mathbf{U}^-$).

3.2 Real-world datasets and cross-validation

We utilised five classification benchmark datasets for performance evaluation. These include breast cancer diagnosis (Breast), prediction free electrons in the ionosphere data (Ionosphere), sonar prediction of mines vs. rocks (Sonar), the Wisconsin database of breast cancer (WDBC), and the Pima Indians diabetes dataset (Pima). All these datasets were obtained from UC Irvine Machine Learning Repository [Lichman, 2013]

To simulate positive unlabeled learning scenarios, we treated instances from the negative class as unlabeled and introduced 50% and 67% of unlabeled positive instances with respect to the positive class by randomly removing label information of 1/2 or 2/3 of instances from the positive class, creating an "easy" and a "hard" scenarios. This gives 2 configurations of each dataset on which the evaluation experiments were performed (Table 1).

Table 1: Summary of real-world datasets and configurations used for positive unlabeled learning.

Dataset	P	N	$\|\mathbf{L}^+\|$	$\|\mathbf{U}^+\|$	$\|\mathbf{L}^+\|/\|\mathbf{U}^+\|$
Breast (easy)	239	444	119	120	∼1
Breast (hard)	239	444	80	159	∼0.5
Ionosphere (easy)	126	225	63	63	1
Ionosphere (hard)	126	225	42	84	0.5
Sonar (easy)	97	111	49	48	∼1
Sonar (hard)	97	111	32	65	∼0.5
WDBC (easy)	212	357	106	106	1
WDBC (hard)	212	357	71	141	∼0.5
Pima (easy)	268	500	134	134	1
Pima (hard)	268	500	89	179	∼0.5

We utilised a multi-layered repetitive 5-fold cross-validation (CV) procedure to evaluate the performance of each method. Specifically, label information of instances from the positive class were randomly removed. This is repeated 5 times each with a different set of selected instances and comprise the first layer of randomisation. Subsequently, the data is split for 5-fold CV and this is repeated 10 times each with a different split point. This gives the second layer of randomisation each is nested with the first layer of 5-fold CV. The performance of each method is the average of each trail plus and minus the mean standard error with respect to a given evaluation matric described below.

3.3 Classification algorithms and evaluation metrics

We applied AdaSampling with support vector machine (SVM) and k-nearest neighbour (kNN) classification algorithms. SVM and kNN are typical examples of eager and lazy learning algorithms, respectively, and therefore represent two different methods that could be used together with AdaSampling. An SVM with radial basis function kernel (C=1) and a kNN with k=3 were used across all positive unlabeled methods as well as the baseline to provide objective comparison between each positive unlabeled learning methods.

The evaluation matrices utilised for performance comparison are sensitivity (Se), specificity (Sp), F_1 score, and geometric mean (GM). Area under the curve (AUC) is not included as a comparison metric because it is not effective for evaluating bias-based approach where the ranking of the predictions often remain the same [Elkan and Noto, 2008], leading to the same ROC curve but adjusted thresholds. Given that all benchmark datasets used in this study have roughly balanced class distribution, the F_1 score and geometric mean provide a good trade-off between sensitivity and specificity for method comparison.

Specifically, each of the matric is defined as following:

$$\text{Se} = \frac{TP}{TP+FN}; \quad \text{Sp} = \frac{TN}{FP+TN};$$

$$F_1 = \frac{2TP}{2TP+FP+FN}; \quad \text{GM} = \sqrt{\frac{TP}{TP+FN} \times \frac{TP}{TP+FP}};$$

where TP, TN, FP, and FN denote the number of true positives, true negatives, false positives, and false negatives, respectively.

4 Results

This section presents the experimental results from using synthetic datasets and performance comparison using real-world datasets. All the data and code are available from the project repository[1].

4.1 Analysis on synthetic datasets

We first evaluated on whether AdaSampling would allow classification algorithms to recover unlabeled positive instances from synthetic datasets. As can be seem from Figure 2, initially, unlabeled positive instances generally receive low classification probabilities with respect to (w.r.t) positive class. However, after only 2 iterations, classifiers coupled with AdaSampling procedure are able to drastically increase classification probabilities with respect to the positive class for most unlabeled positive instances. These results indicate that AdaSampling is highly effective in adaptive learning and converges in very few iterations.

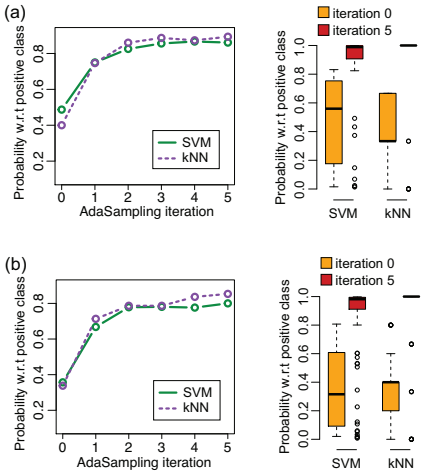

Figure 2: Evaluation of AdaSampling iteration on synthetic datasets. (a) Predicted probabilities of unlabeled positive instances with respect to (w.r.t) positive class in "easy" case (a ratio of 1:0.5:3 for $\mathbf{L}^+$, $\mathbf{U}^+$ and $\mathbf{U}^-$). (b) Predicted probabilities of unlabeled positive instances with respect to positive class in "hard" case (a ratio of 1:1:3 for $\mathbf{L}^+$, $\mathbf{U}^+$ and $\mathbf{U}^-$).

Figure 3 shows decision boundaries created by each classification algorithms. Baseline results correspond to classification by treating all unlabeled instances simply as negative class examples. Results from AdaSampling correspond to applying Alg. 1 to create final classification models. It is apparent that decision boundaries created by all classification models in baseline settings significantly over-penalise positive instances (brown straps of Figure 3). Expectedly, such over-penalisation increase with the number of unlabeled positive instances (compare results under brown strap in Figure 3 (a) and (b)). AdaSampling facilitates classification models to recover a large proportion of labeled as well as unlabeled positive instances from been over-penalised by reducing the

[1] https://github.com/PengyiYang/AdaSampling

Table 2: SVM prediction without or with positive unlabeled learning methods

	Se (%)	Sp (%)	F_1 (%)	GM (%)	Se (%)	Sp (%)	F_1 (%)	GM(%)
	Breast (easy)				Breast (hard))			
Original	97±0	96.6±0	95.5±0	95.5±0	97±0	96.6±0	95.5±0	95.5±0
Baseline	41.4±0.6	99.5±0	57.4±0.7	63.1±0.5	6.5±0.8	100±0	11.4±1.2	35±1.2
BiasMode	66.9±0.4	98.5±0.1	78.5±0.3	80±0.3	61.3±1.9	88.3±1.9	67.4±1	70.4±0.8
BagModel	57.7±0.4	99.2±0.1	72.1±0.4	74.8±0.3	19.2±0.6	99.7±0.1	31.4±0.8	42.3±0.7
AdaSingle	97.8±0.1	95.8±0.1	95.1±0.1	95.2±0.1	97.1±0.2	96.1±0.1	95±0.1	95.1±0.1
AdaEnsemble	97.9±0.1	95.8±0.1	**95.2±0**	**95.3±0**	97.5±0.1	96.1±0.1	**95.2±0.1**	**95.3±0.1**
	Ionosphere (easy)				Ionosphere (hard)			
Original	87±0.2	98.1±0	91.3±0.1	91.5±0.1	87±0.2	98.1±0	91.3±0.1	91.5±0.1
Baseline	15.7±1	99.9±0	25.5±1.4	40.2±1.2	0.7±0.1	100±0	1.3±0.2	NA
BiasModel	77.5±0.7	97.2±0.3	84.5±0.4	85.3±0.4	64.2±1.2	97.8±0.4	75.3±0.9	77.5±0.7
BagModel	54.3±0.9	99.5±0.1	68.8±0.9	72.4±0.7	27.3±1.1	99.9±0	41.5±1.3	50.8±1
AdaSingle	90.8±0.2	91.3±0.4	88.2±0.3	88.4±0.3	87.1±0.5	89.9±0.5	85.1±0.3	85.4±0.3
AdaEnsemble	91.2±0.2	92.6±0.3	**89.3±0.2**	**89.5±0.2**	87.9±0.5	91.4±0.5	**86.6±0.4**	**86.8±0.4**
	Sonar (easy)				Sonar (hard)			
Original	77.3±0.3	88.3±0.2	80.9±0.2	81.3±0.2	77.3±0.3	88.3±0.2	80.9±0.2	81.3±0.2
Baseline	23.8±1.4	99.8±0.1	36.8±1.9	49±1.3	7.9±1.1	100±0	13.4±1.8	38.1±0.9
BiasModel	55.3±0.8	87.9±1.2	64.7±0.5	66.8±0.5	45.3±2.4	81.3±3.3	52±0.9	56.7±0.6
BagModel	40.8±0.8	96.3±0.3	55.5±0.9	60.3±0.8	20.2±1.5	98.7±0.2	31.3±2.2	47.2±1
AdaSingle	63.7±0.5	77.4±1.1	67±0.4	67.5±0.4	54.8±1.3	75.9±2.2	59.5±0.6	61±0.6
AdaEnsemble	65.2±0.5	78.1±1.1	**68.5±0.4**	**68.9±0.4**	55.8±1.1	76.2±1.9	**60.5±0.5**	**61.7±0.4**
	WDBC (easy)				WDBC (hard)			
Original	95.6±0.1	98.7±0	96.6±0	96.7±0	95.6±0.1	98.7±0	96.6±0	96.7±0
Baseline	28.2±1.3	100±0	42.7±1.6	52.1±1.3	3.2±0.5	100±0	5.9±0.8	25.8±0.4
BiasModel	74.9±1.2	95.1±0.6	81.4±0.4	82.3±0.4	72±1.9	80.5±2.2	70.3±0.9	72.2±0.7
BagModel	51.8±0.3	100±0	67.8±0.3	71.7±0.2	15.5±1	100±0	25.8±1.5	40.1±1.2
AdaSingle	96.4±0.2	93.8±0.2	93.2±0.2	**93.3±0.1**	95.4±0.3	92.7±0.2	91.9±0.1	92.1±0.1
AdaEnsemble	96.6±0.1	93.5±0.1	**93.2±0.1**	**93.3±0.1**	95.5±0.3	93.1±0.2	**92.3±0.1**	**92.4±0.1**
	Pima (easy)				Pima (hard)			
Original	55.1±0.1	87.2±0.1	61.5±0.1	62±0.1	55.1±0.1	87.2±0.1	61.5±0.1	62±0.1
Baseline	1.2±0.1	99.8±0	2.3±0.2	NA	0.2±0.1	99.9±0	0.3±0.1	NA
BiasModel	98.1±0.3	4.2±0.7	52.1±0.1	59±0.1	98.3±0.3	2.4±0.4	51.7±0.1	58.7±0.1
BagModel	9.5±0.3	98.8±0.1	16.5±0.5	26.9±0.5	1.3±0.1	99.8±0	2.6±0.1	13.3±0
AdaSingle	82.7±0.3	62.6±0.3	65.5±0.2	67±0.2	79±0.2	64.4±0.2	64.4±0.1	65.6±0.1
AdaEnsemble	83.3±0.2	62.9±0.2	**66±0.1**	**67.5±0.1**	80.4±0.2	64.6±0.2	**65.3±0.1**	**66.5±0.1**

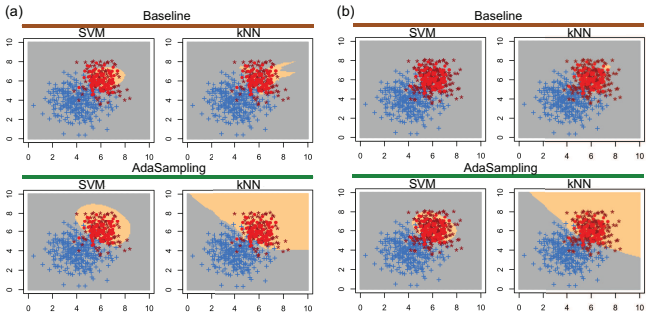

Figure 3: Comparison of baseline (i.e. treat all unlabeled instances as negative examples) and AdaSampling assisted classification on synthetic datasets. (a) Decision boundaries of SVM, kNN, Logit and LDA on "easy" dataset. (b) Decision boundaries of SVM, kNN, Logit and LDA on "hard" dataset.

chance of selecting unlabeled positive instances and therefore extending decision boundaries around positive instances (green straps of Figure 3).

4.2 Classification of real-world datasets

Table 2 and 3 compares SVM and kNN classification on five read-world datasets using baseline approach (i.e. treating all unlabeled instances as negative examples), bias-based approach ("BiasModel") described in [Elkan and Noto, 2008], bagging-like approach ("BagModel") described in [Mordelet and Vert, 2014], and AdaSampling-based single model ("AdaSingle") and ensemble of models ("AdaEnsemble") proposed in this study. The classification of SVM and kNN on the original dataset (i.e. both positive and negative instances are defined) are performed to provide a gold standard in each case.

Direct application of both SVM and kNN to positive unlabeled data gives low predictive sensitivities (Baseline, Table 2 and 3) and the sensitivity decreases with the increase of unlabeled positive instances ("hard" cases). BiasModel significantly improves predictive sensitivities in most cases but suffered from low specificities in Pima dataset classification. It appears that BiasModel over-corrected towards positive class in Pima dataset. This pointed to a potential problem of relying on correcting the initial classifier trained by treating all unlabeled instances as negatives. If a large number of unlabeled

Table 3: kNN prediction without or with positive unlabeled learning methods

	Se (%)	Sp (%)	F_1 (%)	GM (%)	Se (%)	Sp (%)	F_1 (%)	GM (%)
	Breast (easy)				Breast (hard)			
Original	95.7±0.1	97.5±0	95.5±0	95.6±0	95.7±0.1	97.5±0	95.5±0	95.6±0
Baseline	48.6±0.6	99.1±0.1	64.3±0.6	68.3±0.5	25.6±0.4	99.5±0.1	40±0.5	49.2±0.5
BiasModel	82.1±0.3	97.7±0.1	88±0.2	88.3±0.2	67.8±0.5	98.2±0.1	79±0.4	80.3±0.3
BagModel	64.3±0.5	98.8±0.1	77.1±0.4	78.8±0.4	38.6±0.6	99.2±0.1	54.7±0.6	60.7±0.6
AdaSingle	97.2±0.1	97.2±0.1	**96±0.1**	**96±0.1**	95.2±0.3	97.4±0.1	95.2±0.1	95.2±0.1
AdaEnsemble	97.4±0.1	97±0.1	**96±0.1**	**96±0.1**	96.2±0.3	97.3±0.1	**95.6±0.1**	**95.7±0.1**
	Ionosphere (easy)				Ionosphere (hard)			
Original	60.4±0.3	98.1±0.1	73.2±0.2	75.4±0.2	60.4±0.3	98.1±0.1	73.2±0.2	75.4±0.2
Baseline	32±1.5	98.7±0.1	45.7±1.8	54.4±1.2	17.7±1.6	99.3±0.1	27.5±2.3	44.4±1.2
BiasModel	61±1.3	96.5±0.2	71.9±0.9	74±0.7	45.1±1.9	97.9±0.3	58.6±1.7	63.5±1.3
BagModel	42.1±1.9	98.3±0.1	55.9±2	61.3±1.6	24.9±2.2	99.1±0.2	36.2±2.8	51.6±1.6
AdaSingle	65.9±1.6	96.1±0.3	75±1.3	76.7±1	58.3±2	94.4±0.4	**67.5±1.6**	69.8±1.3
AdaEnsemble	67.2±1.5	97.1±0.2	**76.9±1.1**	**78.5±0.9**	56.4±2.4	96.4±0.3	66.8±2	**70.1±1.6**
	Sonar (easy)				Sonar (hard)			
Original	73.1±0.4	87.7±0.2	77.9±0.2	78.3±0.2	73.1±0.4	87.7±0.2	77.9±0.2	78.3±0.2
Baseline	40.6±2	93.8±0.2	53±2.2	57.1±2	22.7±1.9	97.2±0.3	33.5±2.5	46.5±1.9
BiasModel	55.3±1.1	85.9±0.6	**63.6±1**	**65.2±1**	37.4±1	91.7±0.6	49.7±1	54.1±0.9
BagModel	48.7±2	90.4±0.4	59.4±1.9	61.9±1.7	27.6±2	95.5±0.5	39±2.5	48±2
AdaSingle	66.1±1.6	60.7±0.9	62±0.9	62.5±0.9	54.3±1.6	60.8±1.6	53.7±0.7	54.3±0.7
AdaEnsemble	68±1.6	60.4±1.1	63.2±0.8	63.7±0.8	55.6±1.6	60.4±1.6	**54.6±0.7**	**55.2±0.7**
	WDBC (easy)				WDBC (hard)			
Original	87.8±0.1	95.9±0.1	90.1±0.1	90.2±0.1	87.8±0.1	95.9±0.1	90.1±0.1	90.2±0.1
Baseline	41.6±0.6	98.5±0.1	57.3±0.6	62.4±0.5	24±0.6	99.2±0	37.7±0.8	47.1±0.7
BiasModel	74.5±0.6	93.1±0.1	79.9±0.4	80.3±0.4	59±0.4	95.2±0.2	70.3±0.4	71.9±0.3
BagModel	55.2±0.6	97.8±0.1	69.1±0.5	71.7±0.4	34.2±0.6	98.7±0.1	49.6±0.7	56.3±0.6
AdaSingle	90.2±0.3	91.3±0.3	**88.1±0.1**	**88.2±0.1**	88.6±0.5	91.9±0.4	87.6±0.2	87.8±0.2
AdaEnsemble	90.7±0.3	90.9±0.3	**88.1±0.1**	**88.2±0.1**	89.8±0.5	92±0.4	**88.4±0.2**	**88.5±0.2**
	Pima (easy)				Pima (hard)			
Original	53.7±0.2	78.4±0.2	55.2±0.1	55.3±0.1	53.7±0.2	78.4±0.2	55.2±0.1	55.3±0.1
Baseline	19.2±0.2	93.4±0.2	28.9±0.3	33.9±0.3	8.6±0.3	97±0.1	14.8±0.4	22.3±0.5
BiasModel	59.2±0.3	67.1±0.3	53.6±0.2	53.9±0.2	43.5±0.4	77.1±0.2	46.6±0.3	46.8±0.3
BagModel	29.6±0.3	88.7±0.2	39±0.4	41.4±0.4	14.5±0.3	94.6±0.2	23±0.5	28.9±0.5
AdaSingle	77.2±0.4	59.4±0.4	61.1±0.2	62.5±0.2	74.2±0.4	60.4±0.5	59.8±0.3	61.1±0.3
AdaEnsemble	79.7±0.3	59.4±0.4	**62.4±0.2**	**64±0.2**	77±0.3	59.9±0.5	**61.2±0.2**	**62.5±0.2**

instances are utilised as negative examples, the classification models could be of very poor quality, resulting in invalid correction.

BagModel appears to improve moderately on predictive sensitivities but the overall performance is lower them other alternative positive unlabeled learning approaches according to F_1 score and geometric mean. This is expected as in BagModel no explicit mechanism is applied to deal with unlabeled positive instances. While the bootstrap sampling on unlabeled instances may avoid selecting unlabeled positive instances, this is not enforced as the sampling is completely random. In comparison, AdaSampling-based approaches achieved the highest prediction accuracy in terms of F_1 score and geometric mean in all tested datasets and in both "easy" and "hard" cases regardless the classification algorithms (i.e SVM and kNN). Moreover, AdaEnsemble outperformed AdaSingle in most cases, suggesting an added advantage of incorporating heterogenous models using AdaSampling. It is worth noting that in a few cases, the performance of AdaSampling-based approach even outperformed the gold standard where all original labels were used for learning. This suggest that AdaSampling not only can recover missing label information but also could identify and correct potential label noise in the original datasets.

5 Conclusion

In this study, we proposed an adaptive sampling approach, called AdaSampling, for positive unlabeled learning. The proposed approach inheres the spirit of wrapper classification in which a classification model is used iteratively to assess the likelihood of each instance with respect to each class category. AdaSampling is a flexible framework and can be utilised to optimise data for individual classification model as well as constructing more complex ensemble models. Our experimental results demonstrated that both the single classification model and the ensemble of models derived from AdaSampling perform significantly better than those without using AdaSampling and in most cases also outperform other stat-of-the-art approaches for positive unlabeled learning.

We note that with minor modifications, AdaSampling can be easily extended for (1) multi-class classification and (2) class label noise identification and correction. The current study forms the basis of our future work on these directions.

References

[Breiman, 1996] Leo Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

[Calvo *et al.*, 2007] Borja Calvo, Pedro Larrañaga, and José A Lozano. Learning bayesian classifiers from positive and unlabeled examples. *Pattern Recognition Letters*, 28(16):2375–2384, 2007.

[Claesen *et al.*, 2015] Marc Claesen, Frank De Smet, Johan AK Suykens, and Bart De Moor. A robust ensemble approach to learn from positive and unlabeled data using svm base models. *Neurocomputing*, 160:73–84, 2015.

[Denis *et al.*, 2002] Francois Denis, Remi Gilleron, and Marc Tommasi. Text classification from positive and unlabeled examples. In *Proceedings of the 9th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, IPMU'02*, pages 1927–1934, 2002.

[Denis *et al.*, 2005] François Denis, Rémi Gilleron, and Fabien Letouzey. Learning from positive and unlabeled examples. *Theoretical Computer Science*, 348(1):70–83, 2005.

[Elkan and Noto, 2008] Charles Elkan and Keith Noto. Learning classifiers from only positive and unlabeled data. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 213–220. ACM, 2008.

[Khan and Madden, 2014] Shehroz S Khan and Michael G Madden. One-class classification: taxonomy of study and review of techniques. *The Knowledge Engineering Review*, 29(03):345–374, 2014.

[Kohavi and John, 1997] Ron Kohavi and George H John. Wrappers for feature subset selection. *Artificial intelligence*, 97(1):273–324, 1997.

[Lee and Liu, 2003] Wee S Lee and Bing Liu. Learning with positive and unlabeled examples using weighted logistic regression. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pages 448–455, 2003.

[Li and Liu, 2003] Xiaoli Li and Bing Liu. Learning to classify texts using positive and unlabeled data. In *Proceedings of the 18th international joint conference on Artificial intelligence*, pages 587–592. Morgan Kaufmann Publishers Inc., 2003.

[Li *et al.*, 2009] Xiaoli Li, S Yu Philip, Bing Liu, and See-Kiong Ng. Positive unlabeled learning for data stream classification. In *SDM*, volume 9, pages 257–268. SIAM, 2009.

[Li *et al.*, 2011] Wenkai Li, Qinghua Guo, and Charles Elkan. A positive and unlabeled learning algorithm for one-class classification of remote-sensing data. *IEEE Transactions on Geoscience and Remote Sensing*, 49(2):717–725, 2011.

[Lichman, 2013] M. Lichman. UCI machine learning repository, 2013.

[Liu *et al.*, 2002] Bing Liu, Wee Sun Lee, Philip S Yu, and Xiaoli Li. Partially supervised classification of text documents. In *ICML*, volume 2, pages 387–394. Citeseer, 2002.

[Liu *et al.*, 2003] Bing Liu, Yang Dai, Xiaoli Li, Wee Sun Lee, and Philip S Yu. Building text classifiers using positive and unlabeled examples. In *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*, pages 179–186. IEEE, 2003.

[Mordelet and Vert, 2014] Fantine Mordelet and J-P Vert. A bagging svm to learn from positive and unlabeled examples. *Pattern Recognition Letters*, 37:201–209, 2014.

[Nigam *et al.*, 1998] Kamal Nigam, Andrew McCallum, Sebastian Thrun, and Tom Mitchell. Learning to classify text from labeled and unlabeled documents. In *Proceedings of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelligence*, pages 792–799. American Association for Artificial Intelligence, 1998.

[Nigam *et al.*, 2000] Kamal Nigam, Andrew Kachites McCallum, Sebastian Thrun, and Tom Mitchell. Text classification from labeled and unlabeled documents using em. *Machine learning*, 39(2-3):103–134, 2000.

[Yang *et al.*, 2012] Peng Yang, Xiao-Li Li, Jian-Ping Mei, Chee-Keong Kwoh, and See-Kiong Ng. Positive-unlabeled learning for disease gene identification. *Bioinformatics*, 28(20):2640–2647, 2012.

[Yang *et al.*, 2016] Pengyi Yang, Sean J Humphrey, David E James, Yee Hwa Yang, and Raja Jothi. Positive-unlabeled ensemble learning for kinase substrate prediction from dynamic phosphoproteomics data. *Bioinformatics*, 32(2):252–259, 2016.