Noname manuscript No.
(will be inserted by the editor)

Agreeing to disagree: active learning with noisy labels without crowdsourcing

Mohamed-Rafik Bouguelia · Slawomir Nowaczyk · K.C. Santosh · Antanas Verikas

Received: date / Accepted: date

Abstract We propose a new active learning method for classification, which handles label noise without relying on multiple oracles (i.e., crowdsourcing). We propose a strategy that selects (for labeling) instances with a high influence on the learned model. An instance x is said to have a high influence on the model h, if training h on x (with label $y = h(x)$) would result in a model that greatly disagrees with h on labeling other instances. Then, we propose another strategy that selects (for labeling) instances that are highly influenced by changes in the learned model. An instance x is said to be highly influenced, if training h with a set of instances would result in a committee of models that agree on a common label for x but disagree with $h(x)$. We compare the two strategies and we show, on different publicly available datasets, that selecting instances according to the first strategy while eliminating noisy labels according to the second strategy, greatly improves the accuracy compared to several benchmarking methods, even when a significant amount of instances are mislabeled.

Keywords Active Learning · Classification · Label Noise · Mislabeling

1 Introduction

In order to learn a classification model, supervised learning algorithms need a training dataset where each instance is manually labeled. With a large amount of unlabeled instances, one needs to manually label as much instances as possible. Such instances are randomly selected by a human labeler or oracle (i.e., passive learning). With this setting, the learning methods need huge labeled data to produce an optimized classifier. Note that labeling is costly and time consuming. Semi-supervised learning methods like [21] learn using both labeled and unlabeled data, and can therefore be used to reduce the labeling cost to some extent. Nonetheless, instead of randomly selecting the instances to be labeled, active learning methods allow to further reduce the labeling cost by allowing interaction between the learning algorithm and the oracle. Unlike a passive learning, active learning lets the learner choose which instances are more appropriate for labeling, according to an informativeness measure.

The main problem that active learning addresses is about defining informativeness in a way that reduces the number of instances to be labeled along with the improvement of the classifier's performance. This is an important problem because in most real-world applications a large amount of unlabeled data are cheaply available as compared to the labeled ones.

We refer to [24] for a survey of active learning strategies. The most widely used active learning strategies are based on uncertainty sampling [19, 26, 27, 11]. Those strategies select instances in regions of the feature space, where the classifier is most uncertain about its prediction. Such instances are typically close to the decision boundary and allow to fine-tune the boundary regardless of the change which is made to the classifier. Examples of those strategies are presented in [14]. Other uncertainty based active learning methods, such as [5], define uncertainty in terms of the change that a weighted instance brings to the model so that the model changes its prediction regarding this instance. The classifier is then considered uncertain about its prediction, if a small weight is sufficient to change the predicted label. Active

Mohamed-Rafik Bouguelia, Slawomir Nowaczyk, Antanas Verikas
Center for Applied Intelligent Systems Research, Halmstad University, Halmstad 30118, Sweden
E-mail: {mohbou, slawomir.nowaczyk, antanas.verikas}@hh.se

K.C. Santosh
Department of Computer Science, The University of South Dakota, 414 E Clark St, Vermillion, SD 57069, USA
E-mail: santosh.kc@usd.edu

learning strategies based on query-by-committee [10] can also be regarded as an uncertainty sampling because they select instances on which the members of the committee are most uncertain. Those methods implicitly assume that the decision boundary is stable and needs just to be finely tuned. Indeed, stability of a decision boundary is expected to increase as training progresses (in terms of the number of labeled instances). However, since our objective is to reduce the labeling cost, active learning is initialized with few labeled instances and starts with a poor decision boundary. Therefore, the performance of those strategies may be limited. In the light of the aforementioned issues, some active learning strategies define informative instances as those having a great influence on the model [33, 6, 25, 23]. The influence of a candidate instance can be measured by the reduction in the overall uncertainty of the model [33, 23], the change in the probabilistic output of the model [6], or most commonly the change in specific parameters of the model before and after training on that instance. As an example, the authors in [25] use this strategy specifically with a discriminative probabilistic model where gradient-based optimization is used. The influence of an instance on the model is measured by the magnitude of the training gradient if the model is trained on that instance. However, unlike the uncertainty-based active learning, those strategies highly depend on the type of the used classification model, because they evaluate the change in specific parameters of the model. The active learning method that we propose in this paper, measures the informativeness of instances by their influence on the predictive capability of the classification model, not on its parameters, and can be used with any base classification model.

Further, most existing active learning methods assume that the labels given by the oracle are perfectly correct. However, the oracle is usually subject to accidental labeling errors, especially in complex applications such as document analysis [15], entity recognition in text [7], biomedical image processing [1] and video annotation [29], where the labeling task is tedious and time consuming. Such labeling errors not only reduce the accuracy of the classifier, but also mislead the active learner, causing it to query for the label of instances that are not necessarily informative. Many existing methods, like [17] and those surveyed in [9], address the problem of noisy labels in a passive supervised learning setting. Such methods allow to repeatedly correct or remove the possibly mislabeled instances from a dataset of labeled instances which is already available. This is different from the active learning setting where labeling errors only affects instances whose labels are queried during training. Such instances are located in regions of high informativeness, which naturally makes labeling errors not equally likely for all possible instances in the feature space. In this context, active learning with noisy labels is primarily tackled in the literature based on crowdsourcing techniques [16, 32, 13, 2]. However, those techniques can not be used with a single oracle because they rely on the redundancy of labels that are queried for each instance from multiple oracles, which induces a high additional labeling cost. Very few methods that are independent of a specific classifier, try to address this problem without relying on crowdsourcing. A strategy proposed in [20] rely on the classifier's confidence to actively ask the correction of the suspected (i.e., possibly mislabeled) instances from an expert, however, the learning in itself is passive. The same strategy has been investigated for active learning in [12] and [4], where suspected instances can be relabeled or discarded. The active learning method proposed in [33], suggests that a suspiciously mislabeled instance is the one that minimizes the expected entropy over the unlabeled dataset if it is labeled with a new label other than the one given by the oracle. Some more restrictive active learning methods like [28, 8] assume that labeling errors are due to uncertain domain knowledge of the oracle. They address this problem by modeling the knowledge of the oracle and querying for the label of an instance only if it is part of his/her knowledge. However, those methods do not handle labeling errors that are simply due to inattention, and they require the oracle to remain the same over time. The active learning method that we propose in this paper, handles labeling errors without using crowdsourcing and without making assumptions about the oracle's domain knowledge.

More specifically, the proposed active learning method relies on two strategies. In order to measure the informativeness of an instance x under a classification model h, the first strategy (see Section 3) trains h on x and evaluates its output on other unlabeled instances, whereas the second strategy (see Section 4) trains h on other instances and evaluates the output on x. A modification of the later strategy (see Section 6), allows to characterize mislabeled instances. The experimental evaluation that we present in Section 7 shows that querying labels according to the first strategy and reducing noisy labels according to the second strategy, allows to improve the performance of the active learner compared to several commonly used active learning strategies from the literature.

2 Preliminaries and notations

A brief summary of the active learning can be generalized as follows. Let $X \subseteq \mathbb{R}^d$ be a d dimensional feature space. The input $x \in X$ is called an instance. Let Y be a finite set of classes where each class $y \in Y$ is a discrete value called class label. The classifier is then a function h that associates an instance $x \in X$ with a class $y \in Y$ (see Eq. 1). Most classifiers not only return the predicted class y but also give

a score or an estimate of the posterior probability $P(y|x,h)$, i.e., probability that x belongs to class y under the model h.

$$h : \left| \begin{array}{l} \mathbf{X} \longrightarrow \mathbf{Y} \\ x \longmapsto y = h(x). \end{array} \right. \quad (1)$$

Let U be a set of unlabeled instances, L be the set of labeled instances that are queried so far, and h be the current classification model trained on L. In active learning, the learner is given the set U and has to iteratively select an instance $x \in U$ in order to ask an oracle for the corresponding class label $y \in Y$ and add it to L. In this way, the goal is to learn an efficient classification model $h : X \longrightarrow Y$ using a minimum number of queried labels.

The uncertainty based active learning strategies select (for labeling) instances for which the model h is most uncertain about their class. For example, if $y_1 = \max_{y \in Y} P(y|x,h)$ is the most probable class label for x, then the most common uncertainty strategy simply selects instances with a low confidence $P(y_1|x,h)$ or with a high conditional entropy $-\sum_{y \in Y} P(y|x,h) \log P(y|x,h)$ which is a measure of uncertainty.

A general active learning procedure is illustrated in Algorithm 1. The input is a classification model h, a set of unlabeled instances U and an initial set of labeled instances L. At each iteration the algorithm queries for the true class label of the instance which maximizes some informativeness measure F (e.g., uncertainty).

Algorithm 1 General pool-based active learning

1: **Input:** classifier h, unlabeled set U, initial labeled set L, informativeness measure F
2: **repeat**
3: Train h on L
4: Select $\bar{x} = \text{argmax}_{x \in U} F(x)$
5: Query for y the label of $\bar{x}$ from an oracle
6: $L \leftarrow L \cup \{(\bar{x}, y)\}$
7: $U \leftarrow U - \{\bar{x}\}$
8: **until** Labeling budget exhausted

In the next sections, we will use the notation h_x to denote the classification model after being trained on some labeled instance x (i.e., trained on $L \cup \{(x,y)\}$), and the notation $\bar{x}$ to denote an unlabeled instance which is candidate for labeling.

3 Disagreement 1 (selecting the most influencing instance)

This strategy measures the informativeness of an unlabeled instance $\bar{x}$ (candidate for labeling) based on the disagreement between the current classification model h and $h_{\bar{x}}$ (the model after being trained on $\bar{x}$). If the two models greatly disagree on labeling the unlabeled instances of U, then $\bar{x}$ is informative, and its true class label is queried from an oracle.

In order to define the disagreement between two classification models a and b, let us assume that instances are drawn i.i.d. from an underlying probability distribution D. We can then define a metric d which represents the disagreement between a and b as follows:

$$d(a,b) = \mathop{\mathbb{P}}_{x \sim D}[a(x) \neq b(x)] \simeq \frac{|\{x \in U : a(x) \neq b(x)\}|}{|U|}. \quad (2)$$

As shown in Eq. 2, we can practically define this metric as the number of unlabeled instances on which the two models disagree about their predicted labels.

Let us consider the candidate instance $\bar{x} \in U$ for labeling. If we decide to query for the true class label of $\bar{x}$, then $d(h, h_{\bar{x}})$ would express how many instances are affected by this decision. In order to compute the informativeness of $\bar{x}$ before querying for its true label, we define $h_{\bar{x}}$ as the classification model trained on $(\bar{x}, h(\bar{x}))$. In this way, if we query for the true label of $\bar{x}$, then the resulting model will most likely be $h_{\bar{x}}$ because $h(\bar{x})$ is the most probable class label for $\bar{x}$. Based on Eq. 2 we can define the informativeness of $\bar{x}$ as

$$F_1(\bar{x}) = \sum_{x \in U} \mathbb{1}(h(x) \neq h_{\bar{x}}(x)), \quad (3)$$

where $\mathbb{1}(C)$ is the 0-1 indicator function of condition C, defined as

$$\mathbb{1}(C) = \begin{cases} 1 & \text{if } C \text{ is true} \\ 0 & \text{otherwise.} \end{cases}$$

Note that in Eq. 3, the informativeness of the candidate instance $\bar{x}$ is determined by training a model $h_{\bar{x}}$ on $\bar{x}$ and testing on every instance $x \in U$.

Instead of expressing disagreement 1 as *how many instances* are affected (i.e., their predicted label change), we can also express it as *how much* those instances are affected. This is done by introducing a weight as described in Eq. 4

$$F_1'(\bar{x}) = \sum_{x \in U} \Big[\mathbb{1}(h(x) \neq h_{\bar{x}}(x)) \times w_x \Big], \quad (4)$$

where $w_x = |\max_{y \in Y} P(y|x, h_{\bar{x}}) - \max_{y \in Y} P(y|x,h)|$ is the difference in the confidence of the predicted label of x under $h_{\bar{x}}$ and h respectively.

4 Disagreement 2 (selecting the most influenced instance)

We define another disagreement measure that we call disagreement 2. While the objective of disagreement 1 was to favour instances having a large impact on the model output, the objective of disagreement 2 is to favour the instances whose label is wrongly predicted by the current classification model h.

Let us consider the committee (or ensemble) of classification models $C = \{h_x : x \in U\}$. The disagreement 2 strategy computes how many models in the committee C disagree with h on the label of a candidate instance $\bar{x}$. If many members of the committee disagree with $h(\bar{x})$, then $h(\bar{x})$ is likely to be wrong, and the true label of $\bar{x}$ is worth querying:

$$F_2(\bar{x}) = \sum_{x \in U} \mathbb{1}(h(\bar{x}) \neq h_x(\bar{x})). \tag{5}$$

Note that while Eq. 5 looks similar to Eq. 3, it is different. Eq. 3 trains a model $h_{\bar{x}}$ on the candidate instance $\bar{x}$ and tests the output on every instance $x \in U$, and Eq. 5 trains a model h_x on each instance $x \in U$ and tests on $\bar{x}$.

According to Eq. 5, it is likely that $h(\bar{x})$ is wrong if many members of the committee C disagree with $h(\bar{x})$. However, we can get more confidence about that if such committee members agree on a common label (different from $h(\bar{x})$). Therefore, another version of disagreement 2 which quantifies how much a committee of models disagree with $h(\bar{x})$ and agree on a common label for $\bar{x}$, would be expressed as follows:

$$\max_{y \in Y} \sum_{x \in U} \mathbb{1}(h(\bar{x}) \neq h_x(\bar{x}) \wedge h_x(\bar{x}) = y).$$

Instead of using the 0-1 indicator function $\mathbb{1}(h_x(\bar{x}) = y)$ to indicate the agreement of a committee member h_x on the label y for $\bar{x}$, we can rather consider the probability $P(y|\bar{x}, h_x)$, that is, the confidence of the committee member h_x about assigning the label y for $\bar{x}$. The weighted version of disagreement 2 can then be expressed (according to Eq. 6) as

$$F_2'(\bar{x}) = \max_{y \in Y} \sum_{x \in U} \Big[\mathbb{1}(h(\bar{x}) \neq h_x(\bar{x})) \times P(y|\bar{x}, h_x)\Big]. \tag{6}$$

5 Discussion on disagreements 1 and 2

To summarize, the simple version of disagreement 1 (Eq. 3) is quantifying how many predictions change if the model is trained on the candidate instance $\bar{x}$. The weighted version (Eq. 4) is quantifying how big is the change in those predictions. There is a relation between this proposed (disagreement 1) strategy and an optimal active learning strategy. Indeed, since the ultimate objective of active learning is to produce a high accuracy classifier with a minimum number of labeled training instances, an optimal strategy would be to select at each iteration the instance $\bar{x}$ that leads to the maximum increase in accuracy if labeled and used for training[1]. However, this strategy can never be used because it requires knowing beforehand the true class labels of the instances in U to evaluate the gain in accuracy of the classifier.

For simplification purposes, let us consider a binary classification task (i.e., with only two possible classes). Let y_x^* be the (unknown) true class label of an instance $x \in U$. The overall gain in accuracy of the model induced by training on a candidate instance $\bar{x}$ is expressed as

$$G = \frac{1}{|U|} \times \sum_{x \in U} g(x),$$

where $g(x)$ is the gain in accuracy regarding a single instance x.

$$g(x) = \mathbb{1}(h_{\bar{x}}(x) = y_x^*) - \mathbb{1}(h(x) = y_x^*)$$

The value of $g(x)$ would be 1 if the label of x is correctly predicted by $h_{\bar{x}}$ only, -1 if it is correctly predicted by h only, and 0 if the two models h and $h_{\bar{x}}$ predict the same label for x (either correctly or wrongly). To illustrate the relation between disagreement 1 and the optimal active learning, $g(x)$ can be re-written as a factor of $\mathbb{1}(h(x) \neq h_{\bar{x}}(x))$ (which is used in equations 3 and 4) as follows:

$$g(x) = \Big[\mathbb{1}(h_{\bar{x}}(x) = y_x^*) - \mathbb{1}(h_{\bar{x}}(x) \neq y_x^*)\Big] \times \mathbb{1}(h(x) \neq h_{\bar{x}}(x)). \tag{7}$$

While it is impossible to evaluate the left-hand side factor in Eq. 7 because y_x^* is unknown, it is obvious that if $\mathbb{1}(h_{\bar{x}}(x) \neq h(x)) = 0$, there is no gain in accuracy regarding the instance x. Therefore, the disagreement between h and $h_{\bar{x}}$ is a necessary (but not necessarily sufficient) condition to improve the accuracy.

Disagreement 2 is a measure of how likely $h(\bar{x})$ is wrong. The simple version of disagreement 2 (Eq. 5) is quantifying how many models disagree with h regarding the predicted label of the candidate instance $\bar{x}$. The weighted version (Eq. 6) is quantifying how much the different models commonly agree to disagree with h regarding the predicted label of $\bar{x}$. Conceptually, the proposed (disagreement 2) strategy has some similarity with the active learning strategies based on

[1] This is optimal given that we are only allowed to query for the label of one instance at each iteration, and it is only optimal for the given classifier.

query-by-committee and uncertainty sampling, because it allows to query for the label of instances on which h is uncertain (i.e., instances whose labels are likely wrongly predicted by h). However, those strategies define the most uncertain instance as the one being closest to the current decision boundary. This may result in querying for the label of an instance which has not a great impact on the decision boundary even if its label is wrongly predicted by h. Unlike those strategies, disagreement 2 selects instances that are not necessarily closest to the decision boundary, which makes them having a larger influence on the classification model.

It is worth mentioning that the disagreement 1 and the disagreement 2 measures have different objectives and there is not a simple linear correlation between them. This is demonstrated by Fig. 1 which shows the informativeness of some instances according to the weighted versions of disagreement 1 and disagreement 2, for different datasets.

In the next section, we show that a simple modification of the disagreement 2 measure, allows it to be used as a mislabeling measure to characterize noisy labels.

6 Dealing with noisy labels

As indicated in Section 1, the oracle is potentially subject to labeling errors. We consider random labeling errors, where the oracle has a probability $\alpha \in [0, 1]$ for giving a wrong label for each query (i.e., α represents the noise intensity).

Let (x_q, y_q) be a labeled instance from L whose label y_q was queried from the oracle. As a reminder, we previously used the notation h_x to denote the classification model after being trained on the instance x with its predicted label $h(x)$. In other words, h_x is the classifier trained on $L \cup \{(x, h(x))\}$. Let us now use the notation $h_{x \setminus x_q}$ to denote the same model as h_x, but trained after (temporarily) excluding the instance x_q from L.

According to the idea of disagreement 2, if many models highly agree on a common label for x_q, which is different from the queried label y_q (i.e., agreeing to disagree with y_q), then we can suspect y_q to be wrong. Therefore, a mislabeling measure can be expressed based on Eq. 6 for a labeled instance (x_q, y_q), as follows:

$$M(x_q, y_q) = \max_{y \in Y} \sum_{x \in U} \Big[\mathbb{1}(y_q \neq h_{x \setminus x_q}(x_q)) \times P(y | x_q, h_{x \setminus x_q}) \Big]. \quad (8)$$

A labeled instance with a high value of $M(x_q, y_q)$ is potentially mislabeled. In order to reduce the effect of such instance on the active learning, three main alternatives exist in the literature (with different mislabeling measures).

The first alternative is to manually review and correct the label of the instance by an expert oracle. This alternative may induce an additional relabeling cost, because the expert is assumed to be reliable. The second alternative simply consists in removing the instance from L. Note that this alternative may occasionally remove an informative instance that was actually correctly labeled. The third alternative is softer than the previous ones. If the classifier accepts training with weighted instances [30, 22], then every instance in $(x_q, y_q) \in L$ can be weighted by a weight $\frac{1}{M(x_q, y_q)}$ which is the inverse of the mislabeling measure, that is, instances with a higher mislabeling measure have a smaller weight (i.e., smaller impact on the model). However, this alternative highly depends on the used classifier.

As one can notice, each alternative has its benefits and drawbacks. For our experiments, in order to remain independent of any specific classifier, we evaluate active learning with the proposed mislabeling measure (Eq. 8) against other mislabeling measures from the literature, for the two first alternatives (i.e., relabeling and removing). Those alternatives require to periodically select an instance with the highest mislabeling measure from L. To ensure a fair comparison between the mislabeling measures, we simply set this period to $\frac{1}{\alpha}$. For example, if $\alpha = 0.1$, then after each 10 queries, the most likely mislabeled instance is either relabeled or removed from the set L.

7 Experiments

In this section, we evaluate the proposed active learning strategies. First, we present the datasets and evaluation metrics as well as the benchmarking methods used for comparison. Then, we present the experimental results.

7.1 Datasets

We consider in our experimental evaluation seven different datasets of a variable size and number of classes, where six of them are considered as publicly available datasets and are described in the *UCI machine learning repository* [3]. The other dataset is a set of real-world administrative documents that are represented as bag of textual words (i.e., sparse vector containing the occurrence count of each word in the document). Table 1 shows a brief summary of each dataset including the number of categories (column classes), the dimensionality (column features), the number of instances (column size), and the percentage of instances kept for testing (column % test). This percentage is presented for each dataset as it is originally available in the UCI repository.

7.2 Evaluation metrics

There is no absolute measure for evaluating active learning strategies. Most authors demonstrate the performance

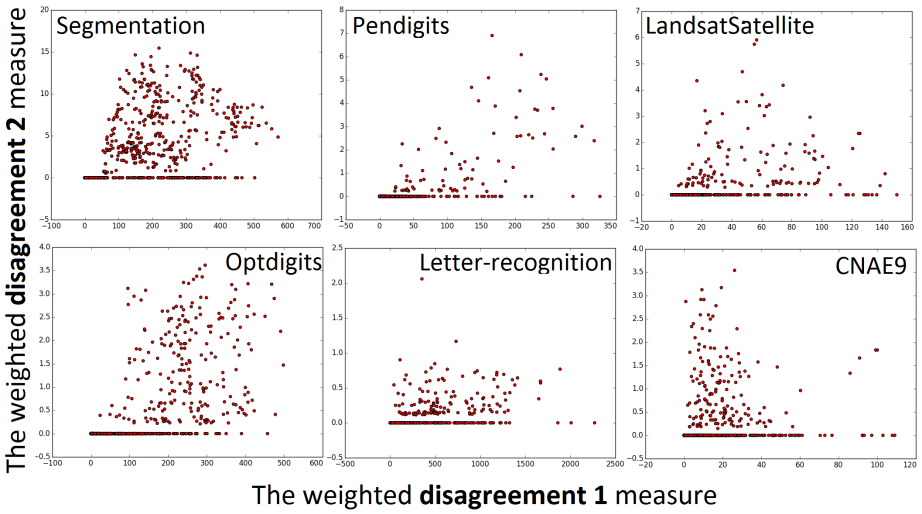

Fig. 1 Disagreement 1 with respect to disagreement 2 on different datasets

of active learning strategies visually by plotting curves of the classifier's accuracy (on a test set) with respect to the number of labeled samples used for training. The higher the curve, the better the active learning strategy. We also present some of such plots in our experiments (see Figs. 2 and 3). Nonetheless, this method does not provide a quantitative evaluation of the active learning strategy for the whole learning session. A straightforward way to achieve this is to compute the average accuracy for the whole active learning session:

$$\frac{1}{|L|}\sum_{t=1}^{|L|} Acc_t,$$

where Acc_t is the test accuracy achieved by the classifier when trained on the t'th labeled instance. However, this measure gives a higher score to strategies that achieve higher accuracy at the end of the learning session (where accuracy is pretty high) even if they perform relatively poorly at early stages (where accuracy is pretty low). Therefore, we propose an alternative evaluation measure which quantifies the performance of a given active learning strategy relatively to the optimal strategy (that we referred to in Section 5) and the random sampling strategy which selects instances independently of their informativeness. This measure aims to be independent of the dataset by quantifying the area between the maximum achievable accuracy and the accuracy which is achievable by a random sampling, and can be computed as

$$E(AL) = \frac{\sum_{t=1}^{|L|} Acc_t(AL) - Acc_t(RANDOM)}{\sum_{t=1}^{|L|} Acc_t(OPTIMAL) - Acc_t(RANDOM)},$$

where $Acc_t(AL)$ is the test accuracy achieved at time t for a given strategy AL.

For the results presented in this paper, we noticed that using the average accuracy measure will lead to the same conclusions as the proposed evaluation metric E, although this is not necessarily true in general. Nevertheless, the metric E have some readability benefits. It shows how much a given active learning strategy is close to the optimal one independently of the dataset. Moreover, it clearly gives a negative value if there is no benefit in using the active learning strategy over the usual passive learning where data is randomly selected by the oracle.

Table 1 Summary of the datasets characteristics

Dataset	classes	features	size	% test
LandsatSatellite	6	36	6435	31%
CNAE-9	9	856	1080	20%
Letter-recognition	26	16	26101	23%
Optdigits	10	64	5620	32%
Pendigits	10	16	10992	32%
Segmentation	7	19	2310	9%
Documents	24	277	1951	33%

7.3 Benchmarking methods

In order to evaluate the proposed active learning strategies (without considering noisy labels for now), we compare them to five active learning strategies described below.

1. **Entropy uncertainty** [24]. It determines the most uncertain instance with respect to all possible classes based on the entropy measure. This strategy selects at each iteration the instance $\bar{x}$ with the highest conditional entropy.
$\bar{x} = \text{argmax}_{x \in U} H(y|x,h)$,
where $H(y|x,h) = -\sum_{y \in Y} P(y|x,h) \log P(y|x,h)$.

2. **Least certain strategy** [14]. It selects at each iteration the instance $\bar{x}$ which is closest to the decision boundary. For classifiers that output an estimate of the prediction probability, this strategy is equivalent to selecting $\bar{x} = \text{argmin}_{x \in U} P(y_1|x,h)$, where $y_1 = \text{argmax}_{y \in Y} P(y|x,h)$ is the most probable class for x.

3. **Sufficient weight strategy** [5]. It computes for each instance x a *sufficient weight* which is defined as the smallest weight that should be associated with x so that the prediction of the classifier $h(x)$ changes from one label to another. Then, it selects (for labeling) the instance $\bar{x}$ with the smallest sufficient weight.

4. **Expected Entropy Reduction (EER)** [33, 23]. This strategy selects the instance $\bar{x}$ which minimizes the expected entropy of the model regarding all the other unlabeled instances. The expected entropy is computed by averaging over all possible labels $y_i \in Y$ for each instance $x \in U$.

$$\bar{x} = \underset{x \in U}{\text{argmin}} \sum_{y_i \in Y} p(y_i|x,h) \sum_{u \in U-x} H(y|u, h_{(x,y_i)}),$$

   where $h_{(x,y_i)}$ is the model after being trained on (x, y_i) and $H(y|u, h_{(x,y_i)})$ is the conditional entropy for the instance u as described in the first strategy.

5. **Change in probabilities** [6]. This strategy measures the variation between two models in terms of the difference in their probabilistic output. Let v_h denote a vector containing the prediction probabilities of the model h on all the available instances (labeled and unlabeled). Given the current model h and the model after being trained on an additional instance h_x, the informativeness of the instance x is measured proportionally to the distance between v_h and v_{h_x}.

In order to evaluate the active learning in the presence of noisy labels, we use the proposed mislabeling measure to filter mislabeled instances as described in Section 6. We compare the results based on two other mislabeling measures (described below) that are independent of the classifier, and has been used to mitigate the effect of noisy labels on active learning in [33, 12, 4].

1. **Entropy reduction based mislabeling measure** [33]. This measure suggests that a suspiciously mislabeled instance is the one that minimizes the expected entropy over the set U, if it is labeled with a new label other than the one given by the oracle.
2. **Margin based mislabeling measure** [12, 4]. This measure simply suggest that a mislabeled instance is the one having a high prediction probability and a low probability for the label given by the oracle. The mislabeling measure for a labeled instance $(x_q, y_q) \in L$ is then simply defined as $p(y_1|x, h_{L-x_q}) - p(y_q|x, h_{L-x_q})$, where y_1 is the most probable label for x_q, y_q is the label given by the oracle, and h_{L-x_q} is the model trained after excluding x_q from L.

SVM is used as a base classifier for all the considered active learning strategies. We consider two variants of the SVM classifier, a simple one (with a linear kernel) and a complex one (with an RBF kernel). We use the Python implementation of SVM which is available in the scikit-learn library [18]. Prediction probabilities are estimated and calibrated based on the SVM scores using the method described in [31]. For all the scenarios that we consider in this paper, the initial SVM model is trained on a fixed set of 50 initially labeled instances chosen randomly from U.

7.4 Results and analysis

We firstly evaluate the active learning without considering noisy labels. Table 2 shows the average E metric (defined in section 7.2) over all classifiers for each individual dataset. Table 3 shows the average E metric as well as the average accuracy over all datasets for each individual classifier (columns *RBF SVM* and *Linear SVM*), and for all datasets and classifiers (column *All SVMs*). Figs. 2 and 3 show the test accuracy of the model h with respect to the number of labeled samples (chosen according to different strategies) used to train h, for 7 different datasets and two variants of h (SVM with RBF kernel in the top of each figure, and SVM with a linear kernel in the bottom of each figure). Fig. 2 (respectively Fig. 3) compares the proposed simple and weighted versions of disagreement 1 (respectively disagreement 2) to the other reference strategies. For clarity purposes, the curve of only one baseline strategy is presented in the figures, however, results of all the considered strategies are summarized in tables 2 and 3.

First, we can observe that all the considered active learning strategies perform generally better than the passive random sampling. This can be seen from Figs. 2 and 3 where the active learning curves are predominantly higher, and from tables 2 and 3, where the E metric values are rarely negative. This confirms that any active learning method can, in general, improve the results over the usual passive learning (random sampling).

Second, we can see that the *simple* and *weighted* versions of the proposed *disagreement 1* and *disagreement 2* strategies, all achieve a better overall performance than the *Entropy* and the *Least certain* strategies. This can be directly observed from the column *All SVMs* of Table 3. Moreover, the *Entropy* and the *Least certain* strategies are occasionally less reliable than the random one. This is especially

Table 2 Average E metric over all classifiers for each dataset

Method	**Average E metric (%)**						
	CNAE9	LandsatSatellite	Letter-recognition	Documents	Optdigits	Pendigits	Segmentation
Entropy	45.28	17.96	-44.78	26.13	08.36	11.95	-24.00
Least certain	45.73	06.52	-43.21	47.57	24.18	11.07	3.92
Sufficient weight	49.38	24.33	7.01	46.54	29.85	47.21	40.10
EER	51.75	29.62	-21.72	50.80	26.40	32.45	22.13
Change in prob.	50.14	**34.37**	17.94	63.57	**52.25**	29.95	37.56
Disagreement 1	53.82	25.69	-3.93	**76.26**	44.44	44.19	40.53
Weighted disag. 1	**57.69**	30.79	**28.59**	71.15	51.65	**50.03**	43.74
Disagreement 2	53.56	20.27	13.64	66.95	42.19	46.07	**45.61**
Weighted disag. 2	54.31	33.79	21.60	72.61	46.15	37.33	34.90

true for the *Letter-recognition* dataset where the two strategies are significantly worse than the *random sampling* (see the *Letter-recognition* column in Table 2, and the *Letter-recognition* curves in Figs. 2 and 3). Indeed, for this dataset, the initial classifier achieves a low test accuracy (around 30%) and the learning progresses slowly. Therefore, selecting the most uncertain instances will allow to fine-tune a poorly defined decision boundary, which slows down the learning progress further. The same observation can be made for the *EER* strategy which achieved a lower performance than the random strategy on the *Letter-recognition* dataset (see Table 2). This is seemingly due to the fact that the EER strategy computes an expected entropy by averaging over all possible labels, which makes it less reliable when the number of classes is high (i.e., 26 classes for the *Letter-recognition* dataset as shown on Table 1).

Third, from Table 3 we can see that the strategy based on the *change in probabilities* achieved the closest performance to our proposed strategies and a slightly better performance than the *simple disagreement 1* strategy. However, the *weighted version of disagreement 1* achieves the best overall result. This is confirmed by the column *All SVMs* which shows the average E metric over all datasets and classifiers. This proves that the instances chosen using the proposed *weighted disagreement 1* strategy are usually more informative.

Finally, the active learning results in the presence of noisy labels are summarized in Table 4. The *weighted disagreement 1* has been used as a query strategy, as it achieved the best performance in the previous experiments. Table 4 shows the average E metric over all datasets when different mislabeling measures (including the one proposed in Section 6) are used to filter (i.e., relabel or remove) the potentially mislabeled instances. Different intensities of noise α are considered. We can observe from Table 4 that for all the considered mislabeling measures, and for a fixed value of α, relabeling the potentially mislabeled instances improves the accuracy better than removing them. However, as discussed in Section 6, relabeling may require an additional cost, while removing doesn't. Further, Table 4 shows that the mislabeling measure that we proposed in Section 6 (which is based on the weighted disagreement 2 measure) allows to better mitigate the effect of noisy labels than the *margin* or the *entropy reduction* based mislabeling measures. This proves that relaying on a committee of models that highly agree on a common label and disagree with the label given by the oracle, allows to better characterize mislabeled instances, even when the noise intensity is significantly high.

8 Conclusion and future work

In this paper, we proposed a new active learning method which is able to query for the label of instances based on how much they impact the output of the classification model. The method is also able to measure how much the queried instance's label is likely to be wrong, based on models that agree to disagree with the current classification model, without relaying on crowdsourcing techniques. This method is generic and can be used with any base classifier. The experimental evaluation demonstrate that the proposed query strategy achieves a higher accuracy compared to several active learning strategies, and that the proposed mislabeling measure efficiently characterize mislabeled instances.

As a future work, we will focus on how to automatically estimate the noise intensity from the data and from previous interactions with the oracle, so that the number of relabeled or removed instances could be adapted according to this noise intensity. We will also focus on improving computational efficiency of the proposed method. Furthermore, we would like to investigate ways of combining the disagreement 1 and disagreement 2 strategies to benefit from their synergy. As a simple idea, Eq. 7 contains two factors that allow to improve the classifier's accuracy. The right-hand side factor has been used in disagreement 1, but the left-hand side factor is not possible to estimate because it requires knowing if the label of x has been correctly or incorrectly predicted. However, since disagreement 2 allows to charac-

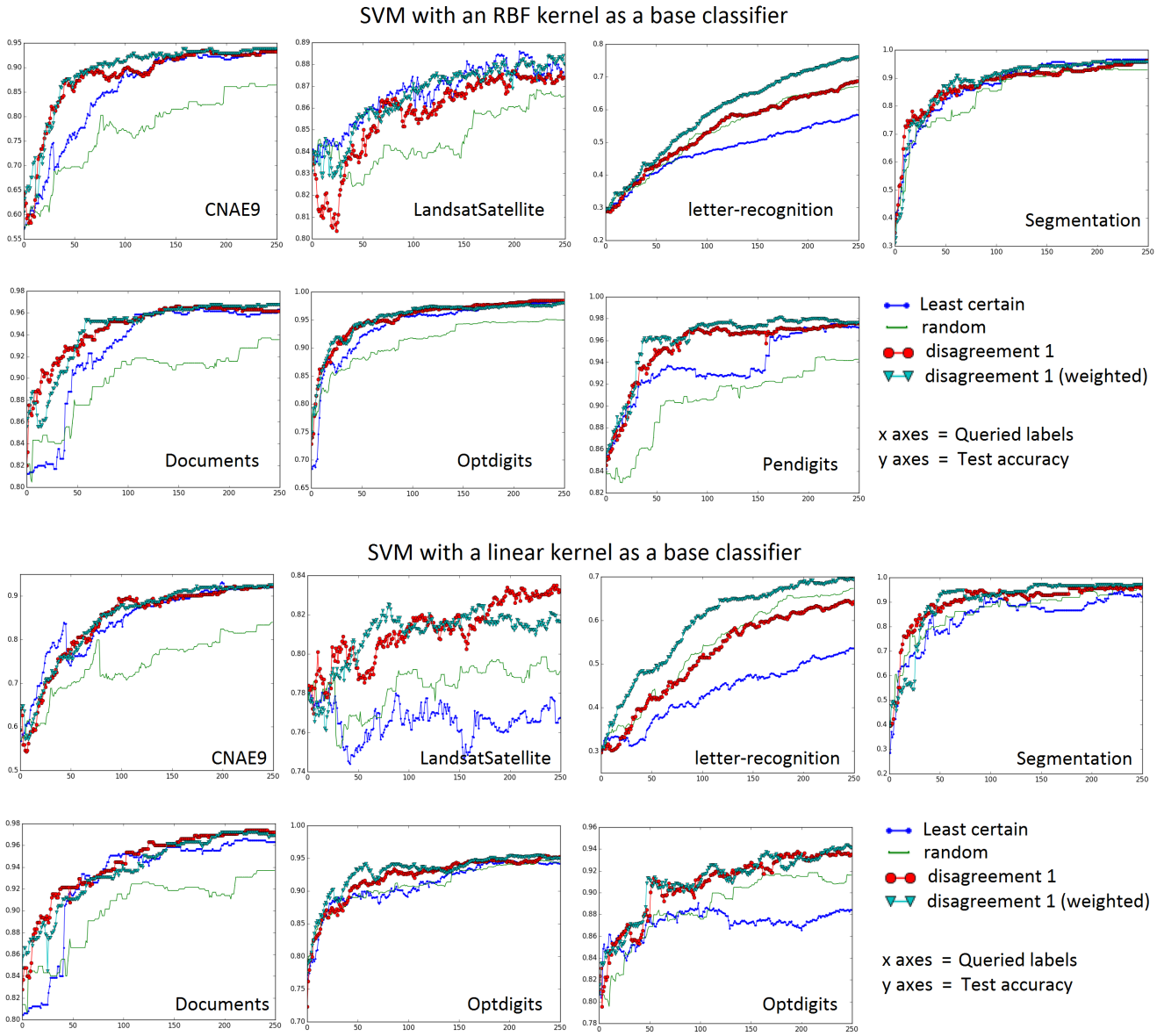

Fig. 2 Disagreement 1 strategy in comparison to uncertainty sampling (least certain strategy)

Table 3 Average E metric and average accuracy, over all datasets for each classifier

Method	**Average E metric (%) — Average accuracy (%)**		
	RBF SVM	**Linear SVM**	**All SVMs**
Entropy uncertainty	14.83 — 82.51	-3.15 — 80.15	5.84 — 81.33
Least certain	31.77 — 83.93	-4.40 — 79.84	13.68 — 81.89
Sufficient weight	38.82 — 85.09	31.01 — 83.90	34.92 — 84.49
EER	41.64 — 85.15	13.05 — 81.69	27.35 — 83.42
Change in prob.	48.09 — 86.08	33.57 — 83.87	40.83 — 84.97
Disagreement 1	47.62 — 85.81	32.66 — 83.51	40.14 — 84.66
Weighted disag. 1	**56.51 — 86.99**	**38.81 — 84.58**	**47.66 — 85.79**
Disagreement 2	51.09 — 86.46	31.28 — 83.56	41.18 — 85.01
Weighted disag. 2	48.70 — 86.06	37.22 — 84.47	42.96 — 85.27

Table 4 Average E metric over all datasets in the presence of noisy labels

Classifier	Method	Average E metric (%)							
		Relabeling noisy labels				Removing noisy labels			
		$\alpha = 0.1$	$\alpha = 0.2$	$\alpha = 0.3$	$\alpha = 0.4$	$\alpha = 0.1$	$\alpha = 0.2$	$\alpha = 0.3$	$\alpha = 0.4$
RBF SVM	No filtering	30.80	29.44	29.84	25.29	30.80	29.44	29.84	25.29
	Entropy reduc.	37.23	37.42	**50.12**	49.70	37.59	25.36	26.84	14.22
	Margin	41.44	37.84	41.80	48.60	35.04	16.75	**31.55**	17.01
	Proposed	**44.98**	**41.95**	49.50	**54.17**	**39.12**	**35.86**	30.80	**31.34**
Linear SVM	No filtering	21.94	29.83	29.65	09.12	21.94	29.83	29.65	09.12
	Entropy reduc.	40.79	38.43	62.02	**56.46**	**31.61**	22.77	30.77	26.26
	Margin	39.83	35.93	53.61	53.93	27.35	13.41	38.19	32.03
	Proposed	**40.81**	**46.99**	**64.70**	55.94	31.39	**31.71**	**41.23**	**33.24**
All SVMs	No filtering	26.37	29.64	29.75	17.21	26.37	29.64	29.75	17.21
	Entropy reduc.	39.01	37.93	55.53	53.08	34.60	24.07	28.80	20.24
	Margin	40.64	36.88	47.70	51.26	31.19	15.23	34.87	24.52
	Proposed	**42.90**	**44.47**	**56.41**	**55.06**	**35.25**	**33.78**	**36.02**	**32.29**

terize instances whose label is probably wrongly predicted, then it can be possibly used as a weight in place of the left-hand side factor of Eq. 7.

References

1. Abedini, M., N. Codella, J. Connell, R. Garnavi, M. Merler, S. Pankanti, J. Smith, and T. Syeda-Mahmood. 2015. A generalized framework for medical image classification and recognition. *IBM Journal of Research and Development* 59(2/3): 1–18.
2. Agarwal, A., R. Garg, and S. Chaudhury. 2013. Greedy search for active learning of ocr. *International Conference on Document Analysis and Recognition*.
3. Bache, K., and M. Lichman. 2013. Uci machine learning repository. http://archive.ics.uci.edu/ml. *Irvine, CA : University of California, School of Information and Computer Science*.
4. Bouguelia, M. R., Y. Belad, and A. Belad. 2015. Identifying and mitigating labelling errors in active learning. *International Conference on Pattern Recognition Applications and Methods*.
5. Bouguelia, M. R., Y. Belaid, and A. Belaid. 2016. An adaptive streaming active learning strategy based on instance weighting. *Pattern Recognition Letters* 70: 38–44.
6. Bouneffouf, D., R. Laroche, T. Urvoy, R. Fraud, and R. Allesiardo. 2014. Contextual bandit for active learning: Active thompson sampling. *International Conference on Neural Information Processing* 26(12): 405–412.
7. Ekbal, A., S. Saha, and U. K. Sikdar. 2014. On active annotation for named entity recognition. *International Journal of Machine Learning and Cybernetics*.
8. Fang, M., and X. Zhu. 2014. Active learning with uncertain labeling knowledge. *Pattern Recognition Letters* 43: 98–108.
9. Frnay, B., and M. Verleysen. 2014. Classification in the presence of label noise: a survey. *IEEE transactions on neural networks and learning systems* 25(5): 845–869.
10. Gilad-Bachrach, R., A. Navot, and N. Tishby. 2005. Query by committee made real. *Advances in neural information processing systems*.
11. Hamidzadeh, J., R. Monsefi, and H. S. Yazdi. 2016. Large symmetric margin instance selection algorithm. *International Journal of Machine Learning and Cybernetics* 7(1): 25–45.
12. Henter, D., A. Stahl, M. Ebbecke, and M. Gillmann. 2015. Classifier self-assessment: active learning and active noise correction for document classification. *IEEE International Conference on Document Analysis and Recognition*.
13. Ipeirotis, P. G., F. Provost, V. S. Sheng, and J. Wang. 2014. Repeated labeling using multiple noisy labelers. *Data Mining and Knowledge Discovery* 28(2): 402–441.
14. Kremer, J., K. Steenstrup Pedersen, and C. Igel. 2014. Active learning with support vector machines. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 4(4): 313–326.
15. Krithara, A., M. R. Amini, J. M. Renders, and C. Goutte. 2008. Semi-supervised document classification with a mislabeling error model. *European Conference on Information Retrieval*.

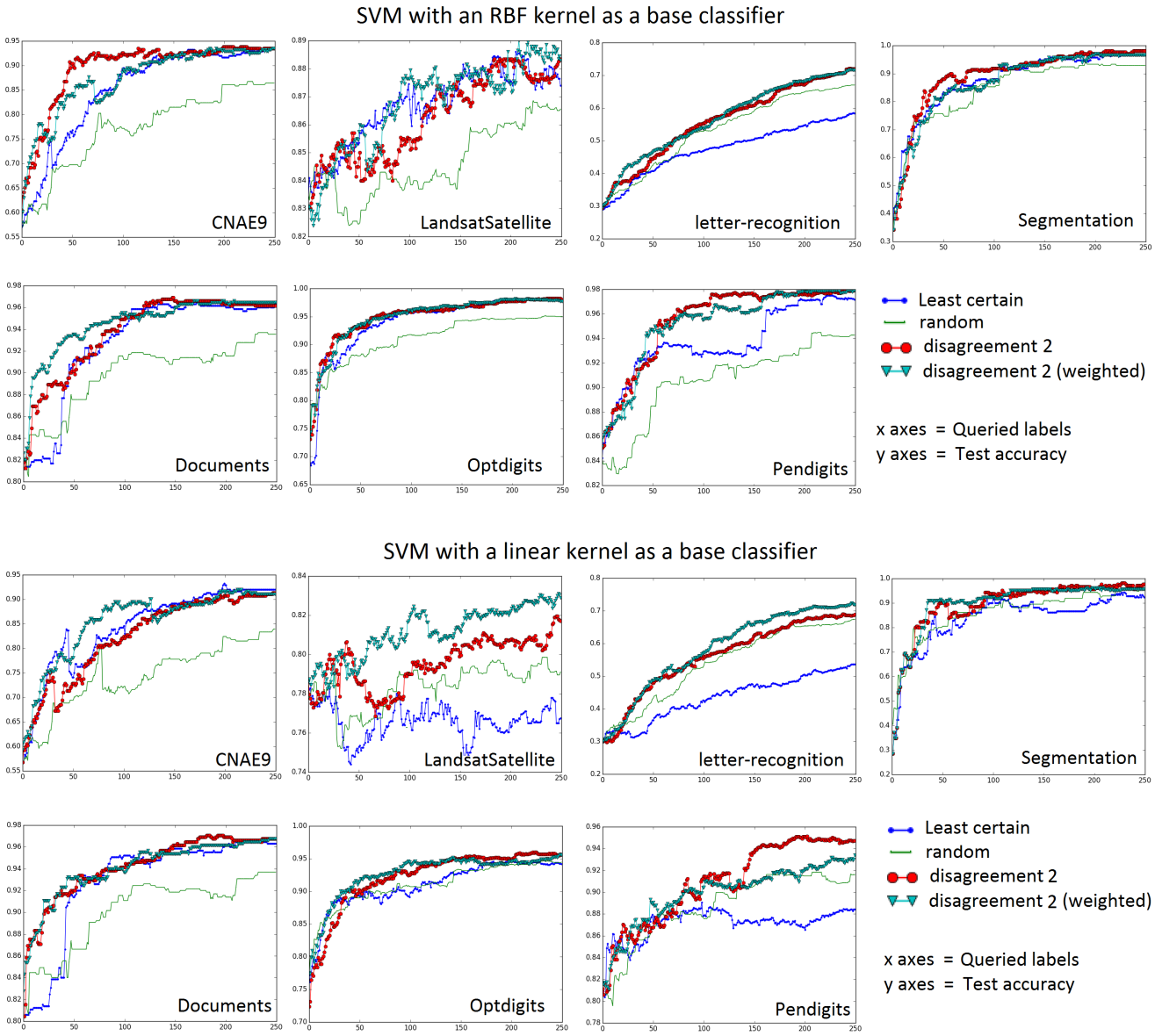

Fig. 3 Disagreement 2 strategy in comparison to uncertainty sampling (least certain strategy)

16. Lin, C. H., and D. S. Weld. 2016. Re-active learning: Active learning with relabeling. *AAAI Conference on Artificial Intelligence*.
17. Natarajan, N., I. S. Dhillon, P. K. Ravikumar, and A. Tewari. 2013. Learning with noisy labels. *In Advances in neural information processing systems*.
18. Pedregosa, F., 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, and E. Duchesnay. 2011. Scikit-learn: machine learning in python. *Journal of Machine Learning Research* 12: 2825–2830.
19. Ramirez-Loaiza, M. E., M. Sharma, G. Kumar, and M. Bilgic. 2016. Active learning: an empirical study of common baselines. *Data Mining and Knowledge Discovery*.
20. Rebbapragada, U., C. E. Brodley, D. Sulla-Menashe, and M. A. Friedl. 2012. Active label correction. *IEEE 12th International Conference on Data Mining*.
21. Ren, W., and G. Li. 2015. Graph based semi-supervised learning via label fitting. *International Journal of Machine Learning and Cybernetics*.
22. Rosenberg, A. 2012. Classifying skewed data: Importance weighting to optimize average recall. *INTERSPEECH*.
23. Roy, N., and A. McCallum. 2001. Toward optimal active learning through sampling estimation of error reduction. *International Conference on Machine Learning*.

24. Settles, B. 2012. Active learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning* 6(1): 1–114.
25. Settles, B., M. Craven, and S. Ray. 2008. Multiple-instance active learning. *Advances in neural information processing systems* 20: 1289–1296.
26. Sharma, M., and M. Bilgic. 2013. Most-surely vs. least-surely uncertain. *IEEE 13th International Conference on Data Mining*.
27. Small, K., and D. Roth. 2010. Margin-based active learning for structured predictions. *International Journal of Machine Learning and Cybernetics* 1(1-4): 3–25.
28. Tuia, D., and J. Munoz-Mari. 2013. Learning user's confidence for active learning. *IEEE Transactions on Geoscience and Remote Sensing* 51(2): 872–880.
29. Vijayanarasimhan, S., and K. Grauman. 2012. Active frame selection for label propagation in videos. *European Conference on Computer Vision, Springer Berlin Heidelberg*.
30. Wu, J., S. Pan, Z. Cai, X. Zhu, and C. Zhang. 2014. Dual instance and attribute weighting for naive bayes classification. *IEEE International Joint Conference on Neural Networks*.
31. Wu, T. F., C. J. Lin, and R. C. Weng. 2004. Probability estimates for multi-class classification by pairwise coupling. *The Journal of Machine Learning Research* 5: 975–1005.
32. Zhang, J., X. Wu, and V. S. Shengs. 2015a. Active learning with imbalanced multiple noisy labeling. *IEEE transactions on cybernetics* 45(5): 1095–1107.
33. Zhang, X. Y., S. Wang, and X. Yun. 2015b. Bidirectional active learning: a two-way exploration into unlabeled and labeled data set. *IEEE transactions on neural networks and learning systems* 26(12): 3034–3044.