A Stopping Criterion for Transductive Active Learning

Daniel Kottke[1](✉), Christoph Sandrock[1], Georg Krempl[2], and Bernhard Sick[1]

[1] University of Kassel, Wilhelmshöher Allee 73, 34121 Kassel, Germany
{daniel.kottke,christoph.sandrock,bsick}@uni-kassel.de
[2] Utrecht University, Princetonplein 5, 3584 CC Utrecht, The Netherlands
g.m.krempl@uu.nl

Abstract. In transductive active learning, the goal is to determine the correct labels for an unlabeled, known dataset. Therefore, we can either ask an oracle to provide the right label at some cost or use the prediction of a classifier which we train on the labels acquired so far. In contrast, the commonly used (inductive) active learning aims to select instances for labeling out of the unlabeled set to create a generalized classifier, which will be deployed on unknown data. This article formally defines the transductive setting and shows that it requires new solutions. Additionally, we formalize the theoretically cost-optimal stopping point for the transductive scenario. Building upon the probabilistic active learning framework, we propose a new transductive selection strategy that includes a stopping criterion and show its superiority.

Keywords: Stopping criteria · Active learning · Transduction

1 Introduction

In classification, the goal is to create a classifier that predicts the true labels for unlabeled instances. Therefore, the classifier needs a set of instance-label pairs (i. e., the training set) which is often not directly available. Fortunately, unlabeled data is usually available at a low cost. However, labeling data is often expensive. Thus, active learning may reduce the annotation cost by selecting instances for labeling that help the classifier in its training progress the most [24].

In this article, we propose to distinguish inductive and transductive active learning. To visualize the difference between both scenarios, we give the following examples: (1) We aim to train a general model to identify protected animals on high-resolution satellite images to surveil their population. In this inductive learning example, we aim to build a general classifier as we want to use it periodically and not only on the images of the initial set (i. e., the test data is unknown). (2) After a natural disaster destroyed some buildings, we search

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-26412-2_29.

M.-R. Amini et al. (Eds.): ECML PKDD 2022, LNAI 13716, pp. 468–484, 2023.
https://doi.org/10.1007/978-3-031-26412-2_29

for survivors. Hence, we take satellite images to find collapsed buildings across the affected regions. In that transductive context, it is important to classify the collected images correctly as their evaluation decides between life and death. In such a transductive scenario, the performance on the collected data is important. Hence, it might be beneficial to use the classifier mainly for simple cases and annotate difficult cases manually even if they do not improve the classifier's performance much. Mixed inductive-transductive scenarios are also possible, where the generalization of the performance beyond the collected data might be relevant. However, to highlight the characteristics and consequences of each scenario, and due to space limitations, this paper will focus on disjoint scenarios.

Up until now, almost all literature refers to inductive active learning and only a few works exist that mention the transductive scenario. Tong [26, p. 15] even argued that the transductive scenario is a special case of inductive active learning and, therefore, solving the inductive case is sufficient. Recently, some articles [16,23] consider transductive active learning but they did not mention its distinct difference to the standard inductive setting in detail.

When deploying classifiers that have been trained with active learning, it is crucial to decide when to stop acquiring more labels [11]. Therefore, cost-sensitive stopping criteria balance misclassification and annotation costs [6,19]. In the inductive scenario, it is difficult to reliably estimate the misclassifications cost because the number of instances to be classified after deployment is often unknown. As we already know the instances to be classified in the transductive scenario, it is straightforward to define and evaluate stopping criteria.

Within this article, our contributions are:

1. We formally define and describe transductive active learning and show that it is beneficial to develop transductive selection strategies (Hypothesis A).
2. We propose a new transductive selection strategy and show its superiority (Hypothesis B). Therefore, we additionally introduce the minimum aggregated cost score, which is a new transductive, cost-based evaluation measure that considers annotation and misclassification costs.
3. We propose a new cost-based stopping criterion for transductive active learning which outperforms its competitors (Hypothesis C).

Next, we discuss the related work, followed by the problem definition, the probabilistic active learning framework, the extension to the transductive case, and our new stopping criterion. Our evaluation is based on three hypotheses.

2 Background and Related Work

In the early 1970 s, Vapnik introduced the concept of transductive inference, which he discussed in more detail in his later publications, e. g. [29, pp. 339ff.]. Both concepts mainly differ in the availability of an evaluation set. In inductive inference, the evaluation set is unknown, whereas it is known for transductive inference. The concept of transduction became especially relevant in the area of *semi-supervised learning* [4, pp. 453ff.]. Here, labels are only partially available, and the assumption is that incorporating the unlabeled instances can improve the classifier's performance. One approach is to successively label the most certain unlabeled instances based on the current classification results. Thereby, the approaches incorporate the structure of the data to build more realistic classification hypotheses [25,27]. In this paper, we extend this idea to active learning.

The main idea of active learning is to actively ask for information that helps best to improve the classifier's predictions [24]. In general, the active learning cycle starts with an initially unlabeled set of instances. A selection strategy successively selects some of these instances and then, an oracle provides the corresponding class labels for these instances. After updating the classifier, the cycle restarts. The main focus of active learning research is on finding an appropriate selection strategy. The most commonly used is uncertainty sampling [14], which selects instances where the classifier is most uncertain. These uncertainty scores are mainly based on probabilistic predictions. Query-by-committee [15] builds a classifier ensemble and selects instances where its members disagree the most. Expected error reduction [22] optimizes the generalization error by simulating potential label acquisitions and thereby provides a decision-theoretic score. Chapelle [3] observed that the used probabilities can be unreliable for only a few labels. Hence, he introduced a prior on the classes for regularization. Value of information [9] differs from expected error reduction in the way that it evaluates the generalization error only on the unlabeled instances and assumes that an unlabeled instance is correct after labeling. In probabilistic active learning [12], the generalization error for both, the current and the simulated (with the additional label) classifier, is evaluated on the same probability distribution.

The term transduction also appears in different contexts in active learning literature. Varying from our definition of transductive active learning, the authors of [7,20] use the term transduction as a technique of propagating labels to the remaining unlabeled data by using the predictions of the classifier. This self-labeling approach is used to create a more robust classifier as it is known from semi-supervised learning. Yu et al. [31] propose a transductive experimental design. Instead of using discrete classes as in classification tasks, they train a model for noisy, continuous targets. Balasubramanian et al. [1] present a selection strategy in the online-based setting. New instances are labeled if the current estimated performance of the classifier is insufficient. As they know this new instance when evaluating it, they use the term transductive learning.

Ishibashi and Hino [8] recently summarized existing stopping criteria for active learning. They divide them into three categories: (1) Accuracy-based approaches (e. g., [13]) evaluate the predictive error of the classifier on unlabeled data or already queried data. (2) Confidence-based approaches (e. g., [30]) use the uncertainty of the model on the remaining unlabeled data to determine the stopping point. (3) Stability-based approaches (e. g., [2]) consider the changes in the model parameters and stop if the model does not change much anymore.

In their survey, Pullar-Strecker et al. [19] compare different stopping criteria and define a cost measure based on the combined cost from annotation and misclassification. Their results indicate that previously proposed stopping criteria based on the accuracy per label tend to stop learning early, while stopping criteria based on classification changes tend to stop late. They conclude that criteria should consider the trade-off between annotation and misclassification costs.

Dimitrakakis et al. [6] introduce a cost-sensitive scenario with a parameter balancing the annotation and misclassification cost. They propose two stopping criteria that compare the expected performance gain and the annotation cost caused by querying an instance. The first one uses convergence properties to estimate the performance gain, while the second one builds on a probabilistic classifier serving this purpose. This idea uses the generalization error of expected error reduction [22] which has been extended in [9,10]. The stopping criterion proposed in [8] compares the performance gain of a parameterized model with the acquisition cost of new labels. As shown in [19], the balancing parameter used by [6,8] is not directly applicable in real-world applications. This is because both articles consider an inductive setting where the size of the evaluation set is implicitly included in their parameters. However, even parameterizing the evaluation set size directly, as proposed by [19], may not solve the problem as it is hard to be estimated. In transduction, the evaluation set is given, which allows us to define a more intuitive and general cost function. To our knowledge, the transductive setting has not been investigated in a cost-sensitive scenario.

3 Problem Definition

For this section, we use a slightly adapted version of Vapnik's [29, p. 15] definition of "learning from examples". A learning task consists of: (1) a generator of random vectors (the instances) $\boldsymbol{x} \in \mathbb{R}^D$, drawn independently from a fixed but unknown probability distribution function $p(\boldsymbol{x})$, (2) an oracle that returns an output value (the label) $y \in \mathcal{Y}$, according to a conditional distribution function $p(y|\boldsymbol{x})$, also fixed but unknown, and (3) a classifier f that aims to predict the oracle's outputs.

In pool-based active learning, we have a dataset $\mathcal{D} = \{(\boldsymbol{x}_1, y_1), ..., (\boldsymbol{x}_N, y_N)\}$, where all instances $\boldsymbol{x}_i$ but only a few/no labels y_i are known to the learner, and $\mathcal{D} \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}, y) = p(y|\boldsymbol{x}) \cdot p(\boldsymbol{x})$. Specifically, the learner has access to[1]:

1. A small or empty set of initially labeled instances $\mathcal{L}_0 \subseteq \mathcal{D}$.
2. A set of initially unlabeled instances $\mathcal{U}_0 = \{\boldsymbol{x} : (\boldsymbol{x}, y) \in \mathcal{D} \setminus \mathcal{L}_0\}$.
3. An oracle o that returns the label $y = o(\boldsymbol{x})$ for every $(\boldsymbol{x}, y) \in \mathcal{D}$.

In each iteration $i \geq 1$, a *selection strategy* selects one instance from the candidate pool $\tilde{\boldsymbol{x}} \in \mathcal{U}_{i-1}$ with the goal to improve the performance of the classifier. The selected instance $\tilde{\boldsymbol{x}}$ is labeled by the oracle with $\tilde{y} = o(\tilde{\boldsymbol{x}})$, added to the set of labeled instances and removed from the candidate pool.

$$\mathcal{L}_i = \mathcal{L}_{i-1} \cup \{(\tilde{\boldsymbol{x}}, \tilde{y})\} \tag{1}$$

$$\mathcal{U}_i = \mathcal{U}_{i-1} \setminus \{\tilde{\boldsymbol{x}}\} \tag{2}$$

[1] We assume that the instances are unique to simplify the notation. This is not a limitation as one can easily drop this assumption by addressing instance-label pairs through their index.

After each iteration, the classifier is updated on the current labeled set which we denote by $f^{\mathcal{L}_i}$. Note that $\mathcal{U}_i$ only contains instances, whereas $\mathcal{D}$ and $\mathcal{L}_i$ consist of instance-label pairs. For readability purposes, we write $\mathcal{U}$ and $\mathcal{L}$ without the indices if possible.

In *transductive active learning*, the goal is to determine the correct labels for all instances in $\mathcal{D}$. As we assume that the oracle provided the true labels for instances in $\mathcal{L}$, we only need the classifier to predict the labels for instances in $\mathcal{U}$. To simplify the notation, we define a meta-classifier $g_f^{\mathcal{L}}$ that returns the known labels for instances in the labeled set and uses the classifier $f^{\mathcal{L}}$ to predict the unknown labels. This is necessary as we cannot be sure that $f^{\mathcal{L}}(\boldsymbol{x}) = y$ for all $(\boldsymbol{x}, y) \in \mathcal{L}$.

$$g_f^{\mathcal{L}}(\boldsymbol{x}) = \begin{cases} y & \text{if } (\boldsymbol{x}, y) \in \mathcal{L} \\ f^{\mathcal{L}}(\boldsymbol{x}) & \text{else} \end{cases} \tag{3}$$

We define the *transductive risk* as the sum of classification losses L over $\mathcal{D}$. As stated above, it is sufficient to evaluate over $\mathcal{U}$.

$$R_{\mathcal{D}}^{\text{tr}}(f^{\mathcal{L}}) = \sum_{(\boldsymbol{x},y)\in\mathcal{D}} L(y, g_f^{\mathcal{L}}(\boldsymbol{x})) = \sum_{\boldsymbol{x}\in\mathcal{U}} L(o(\boldsymbol{x}), f^{\mathcal{L}}(\boldsymbol{x})) = R_{\mathcal{U}}^{\text{tr}}(f^{\mathcal{L}}) \tag{4}$$

Throughout this article, we use the zero-one loss that compares the true label y with the prediction $f^{\mathcal{L}}(\boldsymbol{x})$ label and returns 0 if the prediction is correct and 1 otherwise.

$$L(y, f^{\mathcal{L}}(\boldsymbol{x})) = \begin{cases} 0 & y = f^{\mathcal{L}}(\boldsymbol{x}) \\ 1 & \text{otherwise} \end{cases} \tag{5}$$

In *inductive active learning*, we aim to train a classifier for every (possibly unknown) instance $\boldsymbol{x} \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x})$ with the goal of generalization. Consequently, we do not know the evaluation instances during training in the inductive setting. The distribution $p(\boldsymbol{x}, y)$ is usually approximated with a labeled validation set. As in [29], the (inductive) risk is defined as follows.

$$R(f^{\mathcal{L}}) = \underset{p(\boldsymbol{x},y)}{\mathbb{E}}\left[L(y, f^{\mathcal{L}}(\boldsymbol{x}))\right] = \underset{p(\boldsymbol{x})}{\mathbb{E}}\left[\underset{p(y|\boldsymbol{x})}{\mathbb{E}}\left[L(y, f^{\mathcal{L}}(\boldsymbol{x}))\right]\right] \tag{6}$$

The transductive active learning setting differs from the inductive one in two ways: (1) One knows the data used to evaluate the model beforehand, and one does not need to build a generalized model. (2) One can exclude data from being predicted by the classifier by asking for the label from the oracle.

4 From Inductive to Transductive Active Learning

We build our selection strategy for transductive active learning upon the probabilistic active learning framework [12] that estimates the expected risk reduction when a candidate instance is selected for label acquisition. In the first subsection, we summarize the existing method for the inductive scenario and derive the equations for the transductive case in the second subsection.

4.1 The Probabilistic Active Learning Framework

To estimate the inductive risk, we need to estimate the unknown distributions $p(\boldsymbol{x})$ and $p(y|\boldsymbol{x})$ in Eq. 6. As suggested by [12,21], we approximate $p(\boldsymbol{x})$ using a Monte Carlo approach with an unlabeled set $\mathcal{E} \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x})$. Here, we use $\mathcal{E} = \{\boldsymbol{x} \colon (\boldsymbol{x}, y) \in \mathcal{L}\} \cup \mathcal{U}$. We estimate $p(y|\boldsymbol{x})$ with $p([)\mathcal{L}]y\boldsymbol{x}$ using the data in $\mathcal{L}$ [3,12,17]. The probability is based on a kernel frequency estimate $\boldsymbol{k}_{\boldsymbol{x}}^{\mathcal{L}}$ that contains the number of samples for every class near $\boldsymbol{x}$ using the similarity/kernel $K(\cdot,\cdot)$. By using a Bayesian approach that introduces a prior $\boldsymbol{\epsilon} \in \mathbb{R}_+^{|\mathcal{Y}|}$, the probability $\mathrm{p}^{\mathcal{L}}(y|\boldsymbol{x})$ is given by the y-th element of the normalized vector $\boldsymbol{k}_{\boldsymbol{x}}^{\mathcal{L}} + \boldsymbol{\epsilon}$.

$$\mathrm{p}^{\mathcal{L}}(y|\boldsymbol{x}) = \frac{(\boldsymbol{k}_{\boldsymbol{x}}^{\mathcal{L}} + \boldsymbol{\epsilon})_y}{||\boldsymbol{k}_{\boldsymbol{x}}^{\mathcal{L}} + \boldsymbol{\epsilon}||_1} \qquad k_{\boldsymbol{x},y}^{\mathcal{L}} = \sum_{\substack{(\boldsymbol{x}',y')\in\mathcal{L} \\ y'=y}} K(\boldsymbol{x}, \boldsymbol{x}') \tag{7}$$

The inductive risk of a classifier is estimated as follows.

$$\hat{R}_{\mathcal{E},p^{\mathcal{L}}}(f^{\mathcal{L}}) = \frac{1}{|\mathcal{E}|} \sum_{\boldsymbol{x}\in\mathcal{E}} \sum_{y\in\mathcal{Y}} \mathrm{p}^{\mathcal{L}}(y|\boldsymbol{x}) L(y, f^{\mathcal{L}}(\boldsymbol{x})) \approx R(f^{\mathcal{L}}) \tag{8}$$

For a given candidate $\tilde{\boldsymbol{x}} \in \mathcal{U}$, we calculate the probabilistic gain (xgain) as the expectation value over all possible labeling outcomes $\tilde{y} \in \mathcal{Y}$ of the estimated inductive risk reduction. Therefore, we compare the inductive risks (estimated on $\mathcal{E}$ and $p^{\mathcal{L}^+}$) of the current classifier $f^{\mathcal{L}}$ and the simulated classifier $f^{\mathcal{L}^+}$ that includes the candidate with $\mathcal{L}^+ = \mathcal{L} \cup (\tilde{\boldsymbol{x}}, \tilde{y})$. Since we want to maximize the gain, we consider the negative risk reduction.

$$\mathrm{xgain}(\tilde{\boldsymbol{x}}, \mathcal{L}, \mathcal{E}) = - \mathop{\mathbb{E}}_{p^{\mathcal{L}}(\tilde{y}|\tilde{\boldsymbol{x}})} \left[\hat{R}_{\mathcal{E},p^{\mathcal{L}^+}}(f^{\mathcal{L}^+}) - \hat{R}_{\mathcal{E},p^{\mathcal{L}^+}}(f^{\mathcal{L}}) \right] \tag{9}$$

$$= - \sum_{\tilde{y}\in\mathcal{Y}} \mathrm{p}^{\mathcal{L}}(\tilde{y}|\tilde{\boldsymbol{x}}) \left[\frac{1}{|\mathcal{E}|} \sum_{\boldsymbol{x}\in\mathcal{E}} \sum_{y\in\mathcal{Y}} \mathrm{p}^{\mathcal{L}^+}(y|\boldsymbol{x}) \Big(L\big(y, f^{\mathcal{L}^+}(\boldsymbol{x})\big) - L\big(y, f^{\mathcal{L}}(\boldsymbol{x})\big) \Big) \right] \tag{10}$$

$$= - \sum_{\tilde{y}\in\mathcal{Y}} \frac{(\boldsymbol{k}_{\tilde{\boldsymbol{x}}}^{\mathcal{L}} + \boldsymbol{\beta})_{\tilde{y}}}{||\boldsymbol{k}_{\tilde{\boldsymbol{x}}}^{\mathcal{L}} + \boldsymbol{\beta}||_1} \cdot \frac{1}{|\mathcal{E}|} \sum_{\boldsymbol{x}\in\mathcal{E}} \sum_{y\in\mathcal{Y}} \frac{(\boldsymbol{k}_{\boldsymbol{k}_{\boldsymbol{x}}}^{\mathcal{L}^+} + \boldsymbol{\alpha})_y}{||\boldsymbol{k}_{\boldsymbol{k}_{\boldsymbol{x}}}^{\mathcal{L}^+} + \boldsymbol{\alpha}||_1} \Big(L(y, f^{\mathcal{L}^+}(\boldsymbol{x})) - L(y, f^{\mathcal{L}}(\boldsymbol{x})) \Big) \tag{11}$$

The vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are the priors of the label distribution of the evaluation sample $\boldsymbol{x}$ and the candidate $\tilde{\boldsymbol{x}}$, respectively. They can be interpreted as the number of pseudo-labels added to each region of the dataset. High numbers lead to high regularization of the probabilities and vice versa. As proposed in [12], we set $\boldsymbol{\alpha} = \boldsymbol{\beta} = (10^{-3}, \ldots, 10^{-3})$.

The selection strategy chooses the candidate instance $\tilde{\boldsymbol{x}}^*$ that maximizes the probabilistic gain.

$$\tilde{\boldsymbol{x}}^* = \mathop{\arg\max}_{\tilde{\boldsymbol{x}}\in\mathcal{U}} \{\mathrm{xgain}(\tilde{\boldsymbol{x}}, \mathcal{L}, \mathcal{E})\} \tag{12}$$

4.2 Transductive Probabilistic Active Learning

The goal of transductive active learning is to determine the correct label for all instances in the dataset $\mathcal{D}$. As we assume that the oracle is omniscient, we know that the labels in $\mathcal{L}$ are already correct. To get the label of the remaining instances in $\mathcal{U}$, we can either ask the oracle (and be certain that it is correct) or use the classifier's predictions $f^{\mathcal{L}}(\boldsymbol{x})$. In the latter case, we run into the risk of making mistakes.

Due to these specific characteristics of the transductive scenario, we need to adapt the estimate in Eq. 7 such that the probability for the correct label y for labeled instances $\boldsymbol{x}$ with $(\boldsymbol{x}, y) \in \mathcal{L}$ is 1.

$$\mathbb{p}^{\mathcal{L}}_{\mathrm{tr}}(y|\boldsymbol{x}) = \begin{cases} 1 & (\boldsymbol{x}, y) \in \mathcal{L} \\ 0 & (\boldsymbol{x}, y') \in \mathcal{L} \wedge y \neq y' \\ \mathbb{p}^{\mathcal{L}}(y|\boldsymbol{x}) & \text{otherwise} \end{cases} \tag{13}$$

To calculate the probabilistic gain in the transductive setting, we use the same estimation idea as before, but with the transductive risk. The first step follows the simplification in Eq. 4.

$$\hat{R}^{\mathrm{tr}}_{\mathcal{D},p^{\mathcal{L}}_{\mathrm{tr}}}(f^{\mathcal{L}}) = \hat{R}^{\mathrm{tr}}_{\mathcal{U},p^{\mathcal{L}}_{\mathrm{tr}}}(f^{\mathcal{L}}) = \sum_{\boldsymbol{x}\in\mathcal{U}} \sum_{y\in\mathcal{Y}} \mathbb{p}^{\mathcal{L}}_{\mathrm{tr}}(y|\boldsymbol{x}) \cdot L(y, g^{\mathcal{L}}_{f}(\boldsymbol{x})) \approx R^{\mathrm{tr}}_{\mathcal{U}}(f^{\mathcal{L}}) \tag{14}$$

This estimate allows us to define the estimated risk reduction in the transductive setting as follows:

$$\begin{aligned}
&\Delta\hat{R}^{\mathrm{tr}}_{\mathcal{D},p^{\mathcal{L}^+}_{\mathrm{tr}}}(f^{\mathcal{L}^+}, f^{\mathcal{L}}) = \hat{R}^{\mathrm{tr}}_{\mathcal{U},p^{\mathcal{L}^+}_{\mathrm{tr}}}(f^{\mathcal{L}^+}) - \hat{R}^{\mathrm{tr}}_{\mathcal{U},p^{\mathcal{L}^+}_{\mathrm{tr}}}(f^{\mathcal{L}}) && (15)\\
&= \sum_{\boldsymbol{x}\in\mathcal{U}} \sum_{y\in\mathcal{Y}} \mathbb{P}^{\mathcal{L}^+}_{\mathrm{tr}}(y|\boldsymbol{x}) \Big(L(y, g^{\mathcal{L}^+}_{f}(\boldsymbol{x})) - L(y, g^{\mathcal{L}}_{f}(\boldsymbol{x})) \Big) && (16)\\
&= \sum_{\boldsymbol{x}\in\mathcal{U}\setminus\{\tilde{\boldsymbol{x}}\}} \sum_{y\in\mathcal{Y}} \mathbb{P}^{\mathcal{L}^+}_{\mathrm{tr}}(y|\boldsymbol{x}) \Big(L\big(y, f^{\mathcal{L}^+}(\boldsymbol{x})\big) - L\big(y, f^{\mathcal{L}}(\boldsymbol{x})\big) \Big) && \\
&\quad - \sum_{y\in\mathcal{Y}} \mathbb{P}^{\mathcal{L}^+}_{\mathrm{tr}}(y|\tilde{\boldsymbol{x}}) \big(L\big(y, \tilde{y}\big) - L\big(y, f^{\mathcal{L}}(\tilde{\boldsymbol{x}})\big) \big) && (17)\\
&= \sum_{\boldsymbol{x}\in\mathcal{U}\setminus\{\tilde{\boldsymbol{x}}\}} \sum_{y\in\mathcal{Y}} \mathbb{P}^{\mathcal{L}^+}_{\mathrm{tr}}(y|\boldsymbol{x}) \Big(L\big(y, f^{\mathcal{L}^+}(\boldsymbol{x})\big) - L\big(y, f^{\mathcal{L}}(\boldsymbol{x})\big) \Big) - L\big(\tilde{y}, f^{\mathcal{L}}(\tilde{\boldsymbol{x}})\big)\ . && (18)
\end{aligned}$$

In Eq. 17, we separate $\tilde{\boldsymbol{x}}$ from $\mathcal{U}$ as the candidate serves two purposes. In the first part of the equation, we estimate the *inductive* risk reduction for the remaining unlabeled instances resulting from the improvement of the model with the additional label. In the second part, we assume that the label $\tilde{y}$ is correct. Therefore, we only need to consider the case $y = \tilde{y}$ as $\mathbb{P}^{\mathcal{L}^+}_{\mathrm{tr}}(\tilde{y}|\tilde{\boldsymbol{x}}) = 1$ and $\mathbb{P}^{\mathcal{L}^+}_{\mathrm{tr}}(y|\tilde{\boldsymbol{x}}) = 0$ for $y \neq \tilde{y}$. Hence, we simplify that term to $L(\tilde{y}, f^{\mathcal{L}}(\tilde{\boldsymbol{x}}))$.

Analogous to Eq. 9, the transductive probabilistic gain is calculated as follows:

$$\text{xgain}^{\text{tr}}(\tilde{\boldsymbol{x}}, \mathcal{L}, \mathcal{D}) = - \mathop{\mathbb{E}}_{p^{\mathcal{L}}_{\text{tr}}(\tilde{y}|\tilde{\boldsymbol{x}})} \left[\Delta \hat{R}^{\text{tr}}_{\mathcal{D},p([)\mathcal{L}^{+}]}(f^{\mathcal{L}^{+}}, f^{\mathcal{L}}) \right] \tag{19}$$

$$= - \sum_{\tilde{y} \in \mathcal{Y}} \frac{(\boldsymbol{k}^{\mathcal{L}}_{\tilde{\boldsymbol{x}}} + \boldsymbol{\beta})_{\tilde{y}}}{||\boldsymbol{k}^{\mathcal{L}}_{\tilde{\boldsymbol{x}}} + \boldsymbol{\beta}||_1} \cdot \sum_{\boldsymbol{x} \in \mathcal{U} \setminus \{\tilde{\boldsymbol{x}}\}} \sum_{y \in \mathcal{Y}} \frac{(\boldsymbol{k}^{\mathcal{L}^{+}}_{\boldsymbol{k}_{\boldsymbol{x}}} + \boldsymbol{\alpha})_y}{||\boldsymbol{k}^{\mathcal{L}^{+}}_{\boldsymbol{k}_{\boldsymbol{x}}} + \boldsymbol{\alpha}||_1} \Big(L(y, f^{\mathcal{L}^{+}}(\boldsymbol{x})) - L(y, f^{\mathcal{L}}(\boldsymbol{x})) \Big)$$

$$+ \sum_{\tilde{y} \in \mathcal{Y}} \frac{(\boldsymbol{k}^{\mathcal{L}}_{\tilde{\boldsymbol{x}}} + \boldsymbol{\beta})_{\tilde{y}}}{||\boldsymbol{k}^{\mathcal{L}}_{\tilde{\boldsymbol{x}}} + \boldsymbol{\beta}||_1} \cdot L\big(\tilde{y}, f^{\mathcal{L}}(\tilde{\boldsymbol{x}})\big) \tag{20}$$

The first part is equal to the inductive probabilistic gain evaluated on $\mathcal{U} \setminus \{\tilde{\boldsymbol{x}}\}$ multiplied by the number of instances in that set. This factor is necessary as the transductive risk is defined as the sum over all losses whereas the inductive risk uses the average loss. We call the second part of the equation the *candidate gain* (cgain) as it results from acquiring the correct label from the candidate instance. In summary, we can write the transductive probabilistic gain as the sum of the inductive and the candidate gain:

$$\text{xgain}^{\text{tr}}(\tilde{\boldsymbol{x}}, \mathcal{L}, \mathcal{U}) = |\mathcal{U} \setminus \{\tilde{\boldsymbol{x}}\}| \cdot \text{xgain}(\tilde{\boldsymbol{x}}, \mathcal{L}, \mathcal{U} \setminus \{\tilde{\boldsymbol{x}}\}) + \text{cgain}(\tilde{\boldsymbol{x}}, \mathcal{L}, \{\tilde{\boldsymbol{x}}\}) \, . \tag{21}$$

4.3 Illustrative Example

Figure 1 shows the inductive and the candidate gain for a synthetic 2-dimensional dataset with two classes. The 7 already labeled instances are marked with a gray circle. The classifier's decision boundary is given as a black line and the

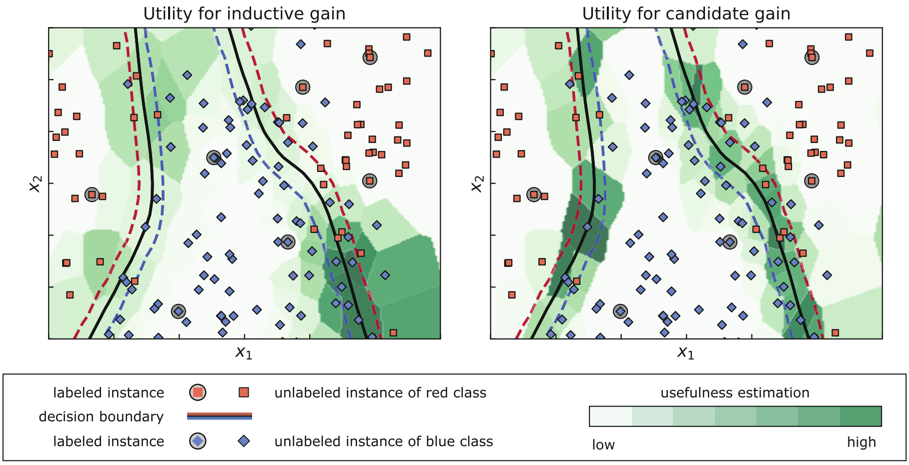


Fig. 1. Utility plots for the inductive and the candidate gain on a synthetic 2-dimensional dataset with 7 labels. (Color figure online)

dashed lines mark its confidence. The utilities are calculated for every unlabeled instance and are given as green surfaces (the color refers to the utility of the nearest instance). We see that the candidate gain (right plot) focuses on difficult instances in regions of high Bayesian error (near the decision boundary). Hence, it does not explore the data space but aims to ask the oracle to prevent the classifier from making wrong predictions. In contrast, the inductive gain (left plot) aims at improving the performance of the classifier. Therefore, it explores regions that are not yet covered with labels (upper left and lower right) and exploits the labels that already are available by refining the decision boundary. Moreover, we observe that regions of higher density (lower right) are preferred over regions with lower density (upper left) as labels have more impact on the classifier's performance there.

5 A Transductive Stopping Criterion

To define a stopping criterion for transductive active learning, we introduce a performance metric using an economic rationale. Therefore, we consider the most relevant kinds of costs involved in an active learning scenario: (1) The *annotation cost* $c_{AN} \in \mathbb{R}^{\geq 0}$ describes the cost of acquiring one label from an oracle, and (2) the *misclassification cost* $c_{ER} \in \mathbb{R}^{\geq 0}$ describes the cost induced by one wrong prediction of the classifier. Intuitively, the annotation cost is dependent on the number of acquired labels, whereas the misclassification cost usually decreases as more labels become available.

We define the *aggregated cost* as the sum of annotation and misclassification costs. Consequently, the aggregated cost can be written as follows for the i-th iteration of the active learning cycle.

$$\operatorname{aggcost}(f, \mathcal{L}_i, \mathcal{U}_i, c_{AN}, c_{ER}) = \underbrace{|\mathcal{L}_i| \cdot c_{AN}}_{\substack{\text{Annotation}\\\text{Cost}}} + \underbrace{R^{\text{tr}}_{\mathcal{U}_i}(f^{\mathcal{L}_i}) \cdot c_{ER}}_{\substack{\text{Misclassification}\\\text{Cost}}} \tag{22}$$

Hence, we assume that the annotation cost is a linear function considering fixed costs c_{AN} for annotating a single instance. We can easily generalize this by using some arbitrary cost function, which describes the cost of acquiring the labeled set $\mathcal{L}_i$, but this is not in the scope of this article. We determine the misclassification cost using the product of the estimated number of wrongly classified instances $R^{\text{tr}}_{\mathcal{U}_i}(f^{\mathcal{L}_i})$ and the cost for one error c_{ER}.

The optimal solution from an economic perspective is to achieve the *minimum aggregated cost* (mac), as shown in Eq. 23. Calculating the mac is equivalent to finding the optimal stopping point for the given costs.

$$\operatorname{mac}(f, c_{AN}, c_{ER}) = \min_i \left(\operatorname{aggcost}(f, \mathcal{L}_i, \mathcal{U}_i, c_{AN}, c_{ER})\right) \tag{23}$$

In this article, we assume to have a selection strategy that iteratively selects one sample. In each iteration of the active learning cycle, we have to decide whether to acquire the label of another instance or to stop querying new labels.

Consequently, we stop the acquisition as soon as the annotation cost c_{AN} exceeds the estimated cost reduction, based on the transductive probabilistic gain:

$$\text{Stop when } \mathit{\Delta} c_{ER} < c_{AN} \quad \text{with} \quad \mathit{\Delta} c_{ER} = \text{xgain}^{\text{tr}}(\tilde{\boldsymbol{x}}^*, \mathcal{L}_i, \mathcal{U}_i) \cdot c_{ER}\,. \tag{24}$$

6 Experimental Evaluation

This section presents our experimental evaluation and starts by describing the experimental setup including the used datasets, competitors, and visualizations. Our evaluation approach is based on three hypotheses as motivated in the introduction. For each contribution, we formulate one hypothesis, present the key findings, and provide a detailed discussion with plots and/or tables.

6.1 Setup, Datasets, and Competitors

All experiments have been implemented in Python using scikit-learn and scikit-activeml[2]. We conduct experiments with the following selection strategies: random sampling (rand), least confidence uncertainty sampling (lc) [14], epistemic uncertainty sampling (epis) [17], query by committee (qbc) [15] with the Kullback-Leibler divergence as a disagreement measure and bootstrapping to generate a committee of 10 classifiers, Monte Carlo expected error reduction (mc) [21] including the extension of Chapelle with $\epsilon = 10^{-3}$ (chap) [3], and value of information (voi) [9]. To show the benefits of the new transductive probabilistic active learning (xpal_tr), we also compare it to the inductive (standard) variant (xpal) [12]. The expected error based strategies mc, chap (with [6]), voi, and xpal_tr implement a cost-based stopping criterion. Whereas voi already evaluates only on the unlabeled instances, we use the unlabeled set as the evaluation set for mc and chap to ensure comparability in the transductive setting.

We use a Parzen window classifier [18] with an RBF kernel as the classifier (similar to [3,12,17]). The main advantages of this classifier are the low number of parameters, the deterministic character, its probabilistic nature, and the fact that it is generic in a way that all methods can be used with that classifier. Using the same classifier for comparison is important as doing otherwise could induce additional biases. The bandwidth parameter of the kernel is set by the mean criterion [5].

We use 10 datasets from OpenML [28]. For simplicity, we remove all samples that contain missing values and standardize all features independently to zero mean and a standard deviation of one. We repeatedly (25 times) split all datasets randomly into two subsets. The first one, which contains 67% of the samples, is used for the active learning circle and builds the initially unlabeled set $\mathcal{U}_0$ according to Sect. 3. This set is used for evaluating the transductive setting. The remaining samples (33%) build the test set for the inductive setting.

[2] https://github.com/dakot/stopTransAL, https://github.com/scikit-activeml.

6.2 Visualization Techniques

To visualize the results, we provide learning curves (e. g., Fig. 2) showing the transductive (resp. inductive) risk. For each dataset and selection strategy, we averaged the risks after every iteration over the 25 repetitions. The goal is to achieve a low error fast.

We summarize these results in ranking tables (e. g., Fig. 3). There, we show the rank of each strategy for every dataset with respect to the area under the performance curve. We calculate the rank for each of the 25 repetitions independently and average these ranks into the final score. Depending on the evaluation goal, we define a baseline strategy that will be compared to all other competitors using a paired Wilcoxon signed-rank test. We identify if the evaluation score of the competitor is significantly higher (arrow up), significantly lower (arrow down), or not significantly different (no sign) than the baseline strategy (p-value .05). These are summarizes as win/tie/loss statistics.

Moreover, we evaluate the transductive scenario by plotting the aggregated cost (e. g., Fig. 4). There, we evaluate the aggregated cost (i. e., the sum of annotation and misclassification costs) for different cost ratios. Depending on the application this ratio might differ and the practitioner can find a suitable algorithm. In Fig. 4, we show the minimum aggregated cost as we identify the optimal stopping point for every selection strategy. Hence, we can assess the quality of selection strategies without the bias of a stopping criterion. In Fig. 5 and Fig. 6 (dashed lines), the aggregated cost is determined based on the proposed stopping point of a stopping criterion. The black lines in the aggregated cost plots show the naive baselines which are determined by the minimum cost between classifying all instances as one class without acquiring any label and acquiring all labels.

Due to the large variety of plots, we only show the most interesting results. You can find all plots in the supplemental material on github.

6.3 Results

Hypothesis A: It is beneficial to develop specific selection strategies for transductive active learning.

Key Findings: When comparing inductive and transductive probabilistic active learning, we show that xpal (inductive) wins when evaluated on the inductive risk, and xpal_tr wins for the transductive risk. Hence adapting the selection strategy is beneficial and solving the inductive case (considering generalization capabilities) is not sufficient to solve transductive active learning.

Detailed Discussion: In Fig. 2, we exemplary selected three datasets to show the inductive and the transductive risk for all selection strategies. We see that the transductive risk finishes at zero risk as there are no errors when all labels are acquired. In contrast, the inductive risk converges at the Bayesian error rate. In Fig. 3, we show the ranking statistics based on the area under the inductive/transductive risk curve as described in the previous subsection. Please note that epis is only valid for 2-class problems. The results show the superiority of xpal in the inductive case (rank 2.56 vs. rank 2.95) and of xpal_tr in the transductive case (rank 1.87 vs. 2.14). The reason for that is that xpal_tr specifically incorporated the acquisition of difficult instances into the target function through the candidate gain as discussed in Subsect. 4.3.

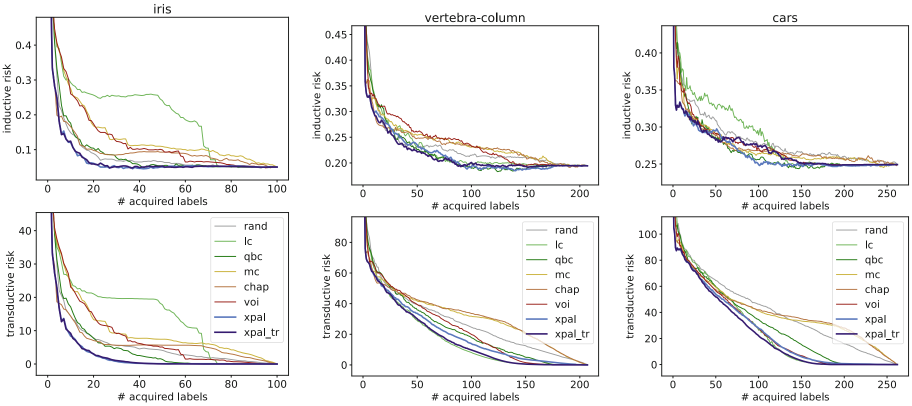
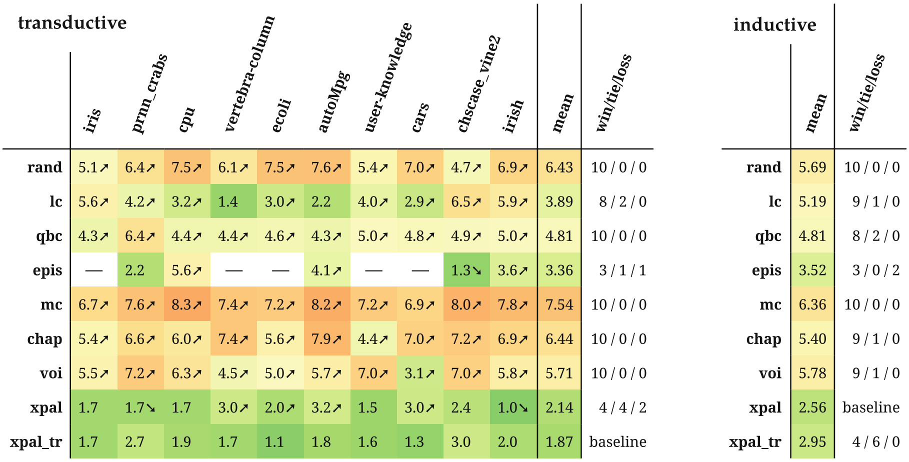


Fig. 2. Learning curves of selection strategies with respect to the inductive (upper) and the transductive (lower) risk.



transductive

	iris	prnn_crabs	cpu	vertebra-column	ecoli	autoMpg	user-knowledge	cars	chscase_vine2	irish	mean	win/tie/loss
rand	5.1↗	6.4↗	7.5↗	6.1↗	7.5↗	7.6↗	5.4↗	7.0↗	4.7↗	6.9↗	6.43	10 / 0 / 0
lc	5.6↗	4.2↗	3.2↗	1.4	3.0↗	2.2	4.0↗	2.9↗	6.5↗	5.9↗	3.89	8 / 2 / 0
qbc	4.3↗	6.4↗	4.4↗	4.4↗	4.6↗	4.3↗	5.0↗	4.8↗	4.9↗	5.0↗	4.81	10 / 0 / 0
epis	—	2.2	5.6↗	—	—	4.1↗	—	—	1.3↘	3.6↗	3.36	3 / 1 / 1
mc	6.7↗	7.6↗	8.3↗	7.4↗	7.2↗	8.2↗	7.2↗	6.9↗	8.0↗	7.8↗	7.54	10 / 0 / 0
chap	5.4↗	6.6↗	6.0↗	7.4↗	5.6↗	7.9↗	4.4↗	7.0↗	7.2↗	6.9↗	6.44	10 / 0 / 0
voi	5.5↗	7.2↗	6.3↗	4.5↗	5.0↗	5.7↗	7.0↗	3.1↗	7.0↗	5.8↗	5.71	10 / 0 / 0
xpal	1.7	1.7↘	1.7	3.0↗	2.0↗	3.2↗	1.5	3.0↗	2.4	1.0↘	2.14	4 / 4 / 2
xpal_tr	1.7	2.7	1.9	1.7	1.1	1.8	1.6	1.3	3.0	2.0	1.87	baseline

inductive

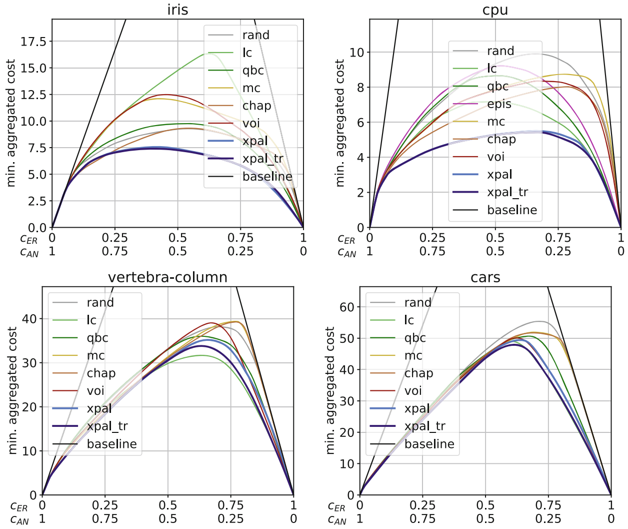
	mean	win/tie/loss
rand	5.69	10 / 0 / 0
lc	5.19	9 / 1 / 0
qbc	4.81	8 / 2 / 0
epis	3.52	3 / 0 / 2
mc	6.36	10 / 0 / 0
chap	5.40	9 / 1 / 0
voi	5.78	9 / 1 / 0
xpal	2.56	baseline
xpal_tr	2.95	4 / 6 / 0

Fig. 3. Ranking statistics with respect to the area under the transductive (left) and inductive (right) risk.

Hypothesis B: Our selection strategy xpal_tr performs best for the transductive risk and the minimum aggregated cost.

Key Findings: We show that transductive probabilistic active learning outperforms the other competitors in the transductive scenario on average when evaluated on the transductive risk and the minimum aggregated cost, i. e., the sum of the annotation and misclassification cost for the optimal stopping point.

Detailed Discussion: To evaluate this hypothesis, we consider the figures from Hypothesis A to evaluate the transductive risk and Fig. 4 to evaluate the minimum aggregated cost. The results show: (1) For the transductive risk, xpal_tr is only defeated significantly in three cases (2 times by xpal and once by epis). Whereas epis performs mediocre on cpu (rank 5.6), the ranks of xpal_tr are all between 1.1 and 3.0. Hence, xpal_tr seems to be fairly robust. (2) For the minimum aggregated cost, we see in the ranking statistics that the hardest competitors are xpal (4 wins, 4 ties, 2 losses), epis (3 wins, 2 losses), and lc (7 wins, 3 ties). All other competitors are defeated significantly on all 10 datasets. Hereby, epis is a special case as it seems to be quite competitive. Still, it is important to note that it only works on half of the datasets as it is only applicable to 2-class problems.



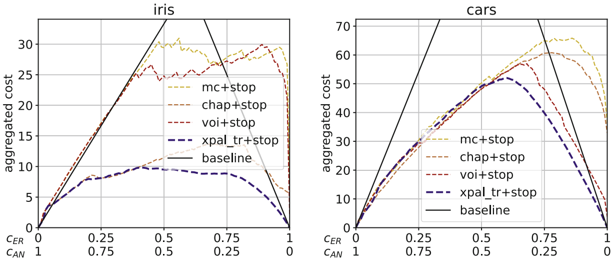
	mean	*win/tie/loss*
rand	6.41	10 / 0 / 0
lc	4.28	7 / 3 / 0
qbc	5.75	10 / 0 / 0
epis	4.30	3 / 0 / 2
mc	6.21	10 / 0 / 0
chap	5.30	10 / 0 / 0
voi	5.43	10 / 0 / 0
xpal	2.61	4 / 4 / 2
xpal_tr	2.37	baseline

Fig. 4. Minimum aggregated cost curves (left) and ranking statistics with respect to the area under the mac curve (right).

Hypothesis C: Our new stopping criterion performs best compared to existing methods.

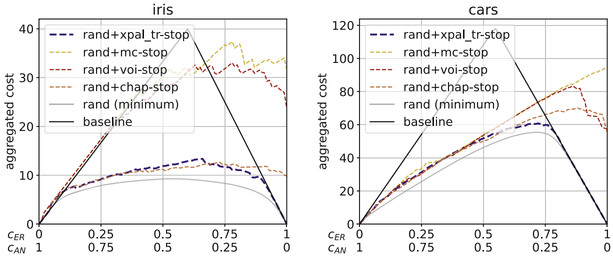
Key Findings: The selection strategy xpal_tr with the new stopping criterion outperforms the existing selection strategies that implement a stopping criterion (mc, chap, voi). To evaluate these stopping criteria independently from the selection strategy, we tested their performance together with random sampling to ensure comparability and show the superiority of our method.

Detailed Discussion: To evaluate the stopping criteria, we show the aggregated cost for the chosen stopping point with respect to the given cost ratios (left) and the ranking statistics (right): In Fig. 5, we evaluated the proposed combinations of a selection strategy and a stopping criterion. Figure 6 shows the results based on a random selection. We use random for the comparison as it induces the smallest bias on the selection. In this scenario, we cannot assume that the best candidate is always selected. Hence, we average the estimated misclassification cost reduction instead of choosing the one from the selected candidate to decide about stopping. Our method xpal_tr significantly outperforms all competitors on all datasets for both cases with only one exception (1 tie).



	mean	win/tie/loss
mc+stop	3.67	10 / 0 / 0
chap+stop	2.53	10 / 0 / 0
voi+stop	2.61	10 / 0 / 0
xpal_tr+stop	1.18	baseline

Fig. 5. Aggregated cost curves for selection strategies that implement a stopping criterion (left) and their ranks based on the area under these curves (right).



	mean	win/tie/loss
rand+mc-stop	3.75	10 / 0 / 0
rand+chap-stop	2.02	9 / 1 / 0
rand+voi-stop	3.12	10 / 0 / 0
rand+xpal_tr-stop	1.10	baseline

Fig. 6. Aggregated cost curves for different stopping criteria using rand as a selection strategy (left) and their ranks based on the area under these curves (right).

7 Conclusion and Outlook

In this article, we introduced and formalized the transductive active learning scenario. We showed that this scenario is not just a special case of the inductive one and that it requires new methods for instance selection. To address this problem, we proposed a novel transductive selection strategy based on the probabilistic active learning framework and experimentally showed that it performs better than the inductive version in the transductive setting. We introduced and motivated a target function for stopping criteria for transductive active learning that considers the misclassification and the annotation costs. Based on this target function, we introduced the minimum aggregated cost that evaluates stopping criteria based on how well they perform for different cost ratios. We used our strategy to derive a novel cost-based stopping criterion. The empirical evaluation showed that it outperforms existing criteria.

In the future, we aim to investigate how the prior influences the proposed methods (here set to 0.001 following [3,12]). In this article, we only considered fixed annotation and misclassification costs and omniscient oracles. However, it is often more realistic that instances have different annotation costs (e. g., dependent on the annotation time, or quality) or that instances have different misclassification costs (e. g., dependent on the instance's importance). Moreover, considering computational cost for the selection might be beneficial. Finally, we want to analyze how our stopping criterion can be used also with other active learning strategies such as uncertainty sampling.

References

1. Balasubramanian, V., Chakraborty, S., Panchanathan, S.: Generalized query by transduction for online active learning. In: International Conference on Computer Vision (Workshops), pp. 1378–1385 (2009)
2. Bloodgood, M., Vijay-Shanker, K.: A method for stopping active learning based on stabilizing predictions and the need for user-adjustable stopping. arXiv preprint arXiv:1409.5165 (2014)
3. Chapelle, O.: Active learning for Parzen window classifier. In: International Workshop on Artificial Intelligence and Statistics, vol. 5, pp. 49–56 (2005)
4. Chapelle, O., Schölkopf, B., Zien, A.: Semi-supervised learning. MIT Press (2010)
5. Chaudhuri, A., Kakde, D., Sadek, C., Gonzalez, L., Kong, S.: The mean and median criteria for kernel bandwidth selection for support vector data description. In: International Conference on Data Mining (Workshops), pp. 842–849 (2017)
6. Dimitrakakis, C., Savu-Krohn, C.: Cost-minimising strategies for data labelling: optimal stopping and active learning. In: International Symposium on Foundations of Information and Knowledge Systems, pp. 96–111 (2008)
7. Güttler, F.N., Ienco, D., Poncelet, P., Teisseire, M.: Combining transductive and active learning to improve object-based classification of remote sensing images. Remote Sens. Lett. **7**(4), 358–367 (2016)
8. Ishibashi, H., Hino, H.: Stopping criterion for active learning based on error stability. arXiv preprint arXiv:2104.01836 (2021)

9. Joshi, A.J., Porikli, F., Papanikolopoulos, N.: Multi-class active learning for image classification. In: Conference on Computer Vision and Pattern Recognition, pp. 2372–2379 (2009)
10. Kapoor, A., Horvitz, E., Basu, S.: Selective supervision: guiding supervised learning with decision-theoretic active learning. In: Int. Joint Conference on Artificial Intelligence, pp. 877–882 (2007)
11. Kottke, D., Calma, A., Huseljic, D., Krempl, G., Sick, B.: Challenges of reliable, realistic and comparable active learning evaluation. In: Workshop on Interactive Adaptive Learning, pp. 2–14 (2017)
12. Kottke, D., Herde, M., Sandrock, C., Huseljic, D., Krempl, G., Sick, B.: Toward optimal probabilistic active learning using a Bayesian approach. Mach. Learn. **110**(6), 1199–1231 (2021)
13. Laws, F., Schätze, H.: Stopping criteria for active learning of named entity recognition. In: International Conference on Computational Linguistics, pp. 465–472 (2008)
14. Lewis, D.D.: A sequential algorithm for training text classifiers. In: International ACM SIGIR Conference on Research and Development in Information Retrieval (1995)
15. McCallumzy, A.K., Nigamy, K.: Employing EM and pool-based active learning for text classification. In: International Conference on Machine Learning, pp. 359–367 (1998)
16. Min, F., Liu, F.L., Wen, L.Y., Zhang, Z.H.: Tri-partition cost-sensitive active learning through kNN. Soft. Comput. **23**(5), 1557–1572 (2019)
17. Nguyen, V.-L., Shaker, M.H., Hüllermeier, E.: How to measure uncertainty in uncertainty sampling for active learning. Mach. Learn. **111**, 89–122 (2021). https://doi.org/10.1007/s10994-021-06003-9
18. Parzen, E.: On estimation of a probability density function and mode. Ann. Math. Stat. **33**(3), 1065–1076 (1962)
19. Pullar-Strecker, Z., Dost, K., Frank, E., Wicker, J.: Hitting the target: stopping active learning at the cost-based optimum. arXiv preprint arXiv:2110.03802 (2021)
20. Reitmaier, T., Calma, A., Sick, B.: Transductive active learning-a new semi-supervised learning approach based on iteratively refined generative models to capture structure in data. Inf. Sci. **293**, 275–298 (2015)
21. Roy, N., McCallum, A.: Toward optimal active learning through Monte Carlo estimation of error reduction. In: International Conference on Machine Learning, pp. 441–448 (2001)
22. Roy, N., Mccallum, A., Com, M.W.: Toward optimal active learning through Monte Carlo estimation of error reduction. In: Proceedings of the International Conference on Machine Learning (ICML), p. 8. San Francisco, CA, USA (2001)
23. Scharei, K., Herde, M., Bieshaar, M., Calma, A., Kottke, D., Sick, B.: Automated active learning with a robot. Arch. Data Science, Ser. A **5**(1), 16 (2018)
24. Settles, B.: Active learning literature survey. Technical report, University of Wisconsin, Department of Computer Science (2010)
25. Sun, S., Hardoon, D.R.: Active learning with extremely sparse labeled examples. Neurocomputing **73**(16–18), 2980–2988 (2010)
26. Tong, S.: Active learning: theory and applications, Ph. D. thesis, Stanford (2001)
27. Triguero, I., García, S., Herrera, F.: Self-labeled techniques for semi-supervised learning: taxonomy, software and empirical study. Knowl. Inf. Syst. **42**(2), 245–284 (2015)
28. Vanschoren, J., van Rijn, J.N., Bischl, B., Torgo, L.: OpenML: networked science in machine learning. SIGKDD Explor. **15**(2), 49–60 (2013)

29. Vapnik, V.N.: Statistical learning theory. John Wiley & Sons, Inc. (1998)
30. Vlachos, A.: A stopping criterion for active learning. Comput. Speech Lang. **22**(3), 295–312 (2008)
31. Yu, K., Bi, J., Tresp, V.: Active learning via transductive experimental design. In: International Conference on Machine learning, pp. 1081–1088 (2006)