Discover-Then-Rank Unlabeled Support Vectors in the Dual Space for Multi-Class Active Learning

Dayou Yu [1] **Weishi Shi** [2] **Qi Yu** [1]

Abstract

We propose to approach active learning (AL) from a novel perspective of discovering and then ranking potential support vectors by leveraging the key properties of the dual space of a sparse kernel max-margin predictor. We theoretically analyze the change of a hinge loss in the dual form and provide both the upper and lower bounds that are deeply connected to the key geometric properties induced by the dual space, which then help us identify various types of important data samples for AL. These bounds inform the design of a novel sampling strategy that leverages class-wise evidence as a key vehicle, formed through an affine combination of dual variables and kernel evaluation. We construct two distinct types of sampling functions, including 1) discovery, which focuses on samples with low total evidence from all classes to support exploration, and 2) ranking, which aims to further refine the decision boundary. These two functions are automatically arranged into a two-phase active sampling process to balance exploration and exploitation. Experiments on various real-world data demonstrate the state-of-the-art AL performance achieved by our model.

1. Introduction

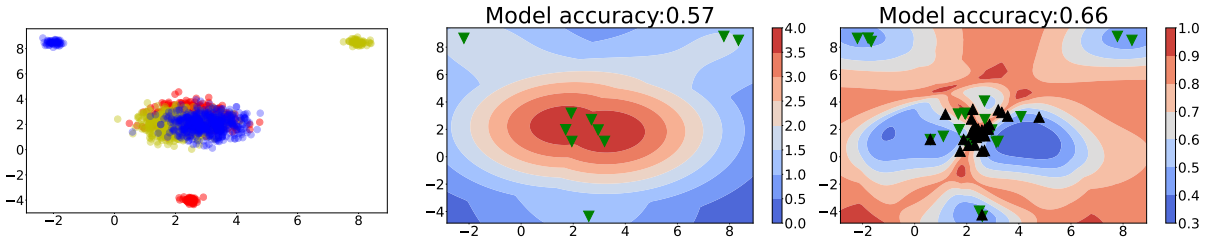
In many specialized domains, such as medicine and military operations, the cost of collecting high-quality labels for training a supervised learning model can be prohibitive. Active learning (AL) provides a viable solution to address label scarcity by allowing a machine learning model (active learner) to sample the instances actively, aiming to build a more accurate model with fewer labeled data instances. Exploration and exploitation are two essential aspects of active sampling when searching a large unlabeled candidate space. The former exposes the active learner to data samples that are dissimilar to the current training set. This behavior helps the learner develop a general understanding of the data distribution. The latter aims at improving the predictive performance by leveraging the existing information.

While sampling strategies designed to balance between exploration and exploitation have shown improved AL performance (Osugi et al., 2005; Yin et al., 2017), existing approaches primarily rely on intuitive heuristics, which lack a principled way to precisely quantify sampling behaviors that correspond to exploration and exploitation, respectively. In this paper, we propose a novel framework to systematically unify exploration and exploitation, aiming to maximize their respective contribution in AL. As a key innovation, we derive theoretical loss bounds from the dual space of a sparse kernel max-margin predictor (*e.g.,* a support vector machine or SVM) and leverage the bounds to design a principled discover-then-rank strategy to sample the important data instances (*i.e.,* potential support vectors or SVs).

The advantage of using a sparse kernel max-margin model is twofold. First, sparse kernel machines offer competitive predictive performance, especially when the training data is scarce. Their generalization capacity depends on the margin instead of the dimensionality of the feature space (Mohri et al., 2018). This can significantly reduce the risk of overfitting, making it fundamentally more advantageous than most deep learning models in the "small data" regime, where AL is commonly applied. For example, in many specialized domains, a realistic total annotation budget is typically less than $1,000$ labels (Tan et al., 2021), which is far from sufficient for training a decent-sized deep neural network that can predict well. Second, the decision boundary can be adequately characterized by a few data instances (*i.e.,* SVs), reducing the computational complexity of candidate searching, which is one of the major concerns in pool-based AL sampling function design. Ideally, if active sampling can perfectly recover all the SVs from a large unlabeled pool, it has the potential to minimize the total labeling cost. The proposed strategy, while built upon a strong theoretical underpinning, can be intuitively interpreted by leveraging

[1]Golisano College of Computing and Information Sciences, Rochester Institute of Technology [2]Department of Computer Science and Engineering, University of North Texas. Correspondence to: Qi Yu <qi.yu@rit.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.



(a) Data distribution (b) F_{disc} sampling (c) F_{rank} sampling

Figure 1: Demonstration of D-TRUST sampling: (a) Distribution of a dataset with three classes (*i.e.,* blue, red, and yellow) where three majority clusters of each class are located close to the center and three minority clusters are located far away from the center; (b) actively selected samples by iteration 5; (c) actively selected samples by iteration 100.

the geometric structure of the dual space. The dual space properties intuitively draw a connection from finding important data samples to improving the model from the AL perspective. In particular, the two stages of the discover-then-rank strategy can be seen as first positioning the decision boundaries by discovering as many potential SVs as possible (through exploration) and then ranking the SVs based on their contribution to the decision boundary (through exploitation) to populate the margin.

Built upon the key properties of the dual variables, our theoretical results quantify both the lower and upper bounds on the change of the hinge loss in its dual form when a new data sample is included. Since only SVs may change the hinge loss, the lower bound allows us to identify potential SVs with a theoretical guarantee. We further develop a label-independent approximation of the lower bound to support active sampling when the labels are not available. Meanwhile, the upper bound justifies the use of dual variables to measure the importance of data samples based on their impact to the decision boundary. It offers a theoretical underpinning that sampling and labeling these important data samples can lead to a faster convergence on AL with reduced annotation cost.

To support the extension to multiple classes, we propose to conduct class-wise decomposition of the decision function to derive the evidence of assigning a data sample to each class. The class-wise evidence allows us to differentiate samples that are far away from the current training data or located close to a mixed group of samples from multiple classes. The theoretical analysis also indicates that these samples are important because they either contribute more to the lower bound or have large dual variable values. We propose two sampling functions accordingly: F_{disc} and F_{rank}, which focus on selecting each type of samples, respectively. These two functions are automatically arranged into a two-phase active sampling process that starts with the discovery and then transits to the ranking of unlabeled SVs to most effectively balance exploration and exploitation. We refer to this process that *Discover-Then-Rank Unlabeled Support vecTors* as D-TRUST. Furthermore, we automatically adjust the transition between the discovery and ranking phases by leveraging the geometric distribution of SVs with distinct dual variable values.

Figure 1 demonstrates the sampling process using the proposed D-TRUST framework, which first relies on F_{disc} for effective exploration and then switches to F_{rank} for fine-tuning. The data distribution contains 3 classes marked as red, yellow, and blue. The green and black triangles denote the samples added to the training set by F_{disc} and F_{rank}, respectively. The heatmap indicates the value of the sampling scores: (b) F_{disc} at 5 iterations; (c) F_{rank} at 100 iterations. Samples in regions with warmer colors are favored by D-TRUST during that iteration. The F_{disc} contour map highly aligns with the distribution of the current training instances, thus can indicate whether a candidate instance is far away and worthy of exploring. The F_{rank} contour map highlights the most conflicting regions at decision boundaries, thus can indicate whether a candidate instance can help fine-tune the current decision boundary.

Our main contribution is threefold:

- we provide the theoretical bounds on the change of hinge loss in its dual form and present insights on how to leverage the lower/upper bounds to support active sampling,
- we propose two novel sampling functions by conducting class-wise decomposition of the decision function for multi-class AL, in order to find the important SVs that improve the lower/upper bounds,
- we automatically arrange the two sampling functions into a two-phase active sampling process to properly balance exploration and exploitation.

We conduct extensive experiments on both synthetic and real data to verify the important theoretical properties of the proposed D-TRUST framework and demonstrate the state-of-the-art AL performance by comparing with competitive baseline models.

2. Related Work

In this section, we discuss existing works that are most relevant to ours. We divide these works into three categories,

including sampling criterion, active learner selection, and sampling behavior.

Uncertainty-based sampling. Uncertainty-based sampling approaches are widely used in the pool-based AL scenario (Lewis & Gale, 1994). The common sampling mechanisms include least confident sampling (Lewis & Gale, 1994; Lewis & Catlett, 1994), best-versus-second-best sampling (Culotta & McCallum, 2005; Settles & Craven, 2008; Scheffer et al., 2001; Li & Guo, 2013), entropy-based sampling (Shannon, 1948), and mutual-information-based sampling (Huo & Tang, 2014). These measurements require models to provide probabilistic prediction capabilities. While there are ways to convert to probabilistic outputs for models like SVMs (Platt, 1999), the inaccurately calibrated probabilities may hurt active learning. Our approach utilizes the proposed class-wise evidence instead of probabilities. We also measure the second-order uncertainty, a concept developed under evidential theory and subjective logic (Jøsang, 2016) that offers a fine-grained analysis on the source of uncertainty to better support active sampling.

Related AL models. When choosing an AL model, the adaptability of the model to small data is essential. In the small data regime, sparse kernel machines, such as SVM, can also achieve competitive performance, while their sparsity can reduce the training cost. Compared to the deep AL models (Shen et al., 2017; Yang et al., 2017; Yoo & Kweon, 2019), SVM (Tong & Koller, 2001) does not require a large batch size while having good interpretability as the SVs stand for important data samples. A major challenge for SVM-based AL frameworks is the extension to multi-class scenarios (Jiang & Gupta, 2019). Common extensions include one-versus-rest (OVR) (Liu & Zheng, 2005) and one-versus-one (OVO) (Xu et al., 2019). The OVO approach is limited because the sampling function has to consider too many decision boundaries. OVR also has the limitation that each binary classifier is highly imbalanced. We propose a class-wise evidence measure to represent the information of each class and make the integration more intuitive.

Balancing exploration-exploitation in AL. Some existing works randomly choose between the nearest and furthest instances with a probability threshold (Osugi et al., 2005). Another solution is to use unsupervised learning (Wang et al., 2017; Lin et al., 2016) or self-supervised learning (Mahmood et al., 2022) to compare the similarities and differences between the unlabeled pool and the current training set to navigate sampling more purposefully. Recent works leverage manifold-preserving graph reduction to achieve promising exploration-exploitation balance using AL models such as Gaussian Processes and SVMs (Zhou & Sun, 2015; Xie, 2021). However, those approaches require additional structural information to perform sampling, which limits their applications in many domains. In order to benefit from balanced exploration-exploitation, some works have used the numbers of intersecting margins to approximate data density and guide the sampling behaviors (Osugi et al., 2005; Demir et al., 2010). Alternatively, QUIRE (Huang et al., 2010) takes a min-max view and measures the informativeness and representativeness of an instance using approximated true labels with predictions. The convex hull-based sampling (Shi & Yu, 2018) encourages exploration by penalizing the instances within the convex hull of SVs in the predicted class. However, most of the related designs rely on solving additional optimization problems, which can be computationally expensive and not suitable for AL. In our approach, we use a label-independent approximation of the change to the hinge loss, and an automatic transitioning mechanism to balance exploration and exploitation with no additional computational overhead.

3. A Theoretical Foundation for Dual Space Active Sampling

We present our theoretical foundation to conduct active sampling in the dual space by proving both the lower and upper bounds on the change of the hinge loss. Intuitively, since a non SV is located outside of the margin, it does not cause the change to the loss. Thus, the ability to quantify the loss change to the model can effectively locate potential SVs to support active sampling. However, evaluating the loss change in its primal form is challenging due to lack of labels in the AL setting. We propose to formulate the dual form of the hinge loss and leverage the special properties of the dual variables to derive the bounds.

3.1. Problem Setup

We consider a pool-based AL scenario. The labeled training set $\mathcal{L}$ contains $\{(\mathbf{x}_n, t_n)\}_{n=1}^{N_\mathcal{L}}$, in which $\mathbf{x}_n$ stands for the feature vector of the n-th data instance and $t_n \in \mathcal{M} = \{1, 2, ..., M\}$ stands for the label. The unlabeled pool $\mathcal{U}$ contains only feature vectors $\{(\mathbf{x}_i)\}_{i=1}^{N_\mathcal{U}}$, where $N_\mathcal{U} \gg N_\mathcal{L}$. At each active sampling iteration, a data instance is sampled from $\mathcal{U}$ using a certain sampling function: $\mathbf{x}^* = \arg\min_{\mathbf{x}\in\mathcal{U}} F(\mathbf{x})$, where $F(\cdot)$ measures the informativeness of a sample.

The support vector machine (SVM) classifier (Vapnik, 1968) has been commonly used for AL. It maximizes the margin between different classes by making $y(\mathbf{x}_n) \geq 1$ when $t_n = 1$ and $y(\mathbf{x}_n) \leq -1$ when $t_n = -1$, where $y(\mathbf{x})$ is the output from the decision function, given by

$$y(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + b = \sum_{n=1}^{N} a_n t_n k(\mathbf{x}_n, \mathbf{x}) + b \qquad (1)$$

where a_n is a dual variable. In the theoretical analysis, we assume a general form of y without the offset b. Extension to multiple classes is typically achieved through one-versus-rest (OVR) (Liu & Zheng, 2005) for both simplicity and

interpretation purposes. Then, we use $\{(\mathbf{x}_n, \tilde{t}_{nm})\}_{n=1}^N$ to denote each binary classification problem, where we transform the labels into $\{-1, 1\}$, $\tilde{t}_{nm} = 1$ if $t_n = m$.

3.2. Geometric Interpretation of Dual Variables

The learning objective of SVM introduces a margin constraint C that controls the level of penalty for misclassified samples. C controls the slack variables ξ_n's, which relax the hinge-loss condition: $\xi_n = 1 - t_n y(\mathbf{x}_n)$. It is also the upper bound for dual variables a_n's. During the optimization, some dual variables are reduced to zero. A data instance with a non-zero a_n is called a support vector (SV). The SVs alone can fully define the decision boundary, thus the solution is sparse. The decision function can be seen as a hyper-plane in the dual space spanned by the kernel basis $k(\mathbf{x}_n, \mathbf{x})$ with non-zero a_n. Among the SVs, those that are located on the margin satisfy $a_n < C$. These locations should be where the model makes relatively confident predictions according to the margin. For SVs within the margin, their dual variables satisfy $a_n = C$. Geometrically, these SVs can be on either side of the decision boundary, which implies a higher possibility of mis-classification. Intuitively, the larger the dual variable value is, the bigger impact this SV has on the decision boundary. In (Burges, 1998), a mechanical analogy shows the impact of SVs in an intuitive way. In particular, the SVM solutions can translate into the conditions of mechanical equilibrium:

$$\sum \text{Forces} = \sum_{n=1}^{N} a_n t_n \hat{\mathbf{w}} = 0 \tag{2}$$

$$\sum \text{Torques} = \sum_{n=1}^{N} \mathbf{s}_n \times (a_n t_n \hat{\mathbf{w}}) = 0 \tag{3}$$

where $\mathbf{s}_n$ stands for an SV. In this analogy, the force is proportional to a_n. Consequently, we aim to design a sampling function to efficiently collect the data samples with an estimated large a_n, which can potentially become important SVs when labeled, leading the model to faster convergence. Next, we show how the proposed active sampling function searches and ranks unlabeled data instances in this dual space.

3.3. Bounding the Change of Hinge Loss

In this section, we present our theoretical results that bound the change of the hinge loss. This will allow us to effectively avoid non SVs when conducting active sampling and focus on the important data instances that directly shape the final decision boundary. Since analyzing the exact change of loss without the label information is challenging in the primal form, we proceed to bound the change in the dual space, which is based on the dual form of the hinge loss defined below.

Definition 1 (Dual hinge loss). Given the optimal hinge loss in the primal form:

$$L_N = \min_{\mathbf{w},b,\boldsymbol{\xi}} \frac{\|\mathbf{w}\|^2}{2} + C\sum_{n=1}^{N} \xi_n, \quad \text{s.t.} \quad t_n y(\mathbf{x}_n) \geq 1 - \xi_n \wedge \xi_n \geq 0, n \in [N] \tag{4}$$

the corresponding dual hinge loss is defined as

$$G(\mathbf{a}_N) = \sum_{n=1}^{N} a_n - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} a_i a_j t_i t_j k(\mathbf{x}_i, \mathbf{x}_j) \tag{5}$$

where $\mathbf{a}_N = (a_1, ..., a_N)^\top$ are the dual variables that maximizes the dual problem of (4).

Since the primal problem is convex, there is no duality gap between the primal and dual solutions. Thus, we have $G(\mathbf{a}_N) = L_N$, which allows us to use the dual hinge loss as a proxy to analyze the original hinge loss. In particular, when including an SV $\mathbf{x}_{N+1}$ into the training set, it may reduce the margin (*i.e.,* by decreasing $1/\|\mathbf{w}\|$), which leads to the increase of $\frac{\|\mathbf{w}\|^2}{2}$ and/or introduce a positive slack variable ξ_{N+1} when $t_{N+1}y(\mathbf{x}_{N+1}) < 1$. As a result, we will observe an increase of the hinge loss: $L_{N+1} > L_N$. In contrast, for a non SV, the loss will remain the same. While quantifying the exact loss change is challenging in the primal form, we can bound this change by leveraging the dual hinge loss, as summarized in the following theorem.

Theorem 1. *Let $G_N = G(\mathbf{a'}_N)$ be the dual hinge loss with N samples and $G_{N+1} = G(\mathbf{a}_{N+1})$ be the loss with one additional sample. Assume that $\mathbf{a'}_N$ and $\mathbf{a}_{N+1}$ are the solutions of the corresponding dual problems. Let $y^{(N)}(\cdot)$ and $y^{(N+1)}(\cdot)$ denote the decision functions given by (1). We define the change of loss $\epsilon^{(N+1)} = G_{N+1} - G_N$. Then, $\epsilon^{(N+1)}$ can be bounded as follows:*

$$\epsilon^{(N+1)} \leq a_{N+1}\left[1 - (t_{N+1}y^{(N+1)}(\mathbf{x}_{N+1}))\right] + \frac{1}{2}a_{N+1}^2 k(\mathbf{x}_{N+1}, \mathbf{x}_{N+1}) \tag{6}$$

$$\epsilon^{(N+1)} \geq \begin{cases} 0, & t_{N+1}y^{(N)}(\mathbf{x}_{N+1}) > 1 \\ \frac{\left[1 - t_{N+1}y^{(N)}(\mathbf{x}_{N+1})\right]^2}{2k(\mathbf{x}_{N+1}, \mathbf{x}_{N+1})}, & \textit{Otherwise} \end{cases} \tag{7}$$

Proof sketch. To bound the change of loss $\epsilon^{(N+1)}$, we introduce $\mathbf{a'}_{N+1} = (\mathbf{a'}_N, 0)^\top$ which extends $\mathbf{a'}_N$ with a 0 entry, $\mathbf{e}_{N+1} = (0, .., 0, 1)^\top$, which is the unit vector in the $(N+1)$-th direction, and $\beta \geq 0$, which is a hypothetical dual variable. By definition, $\mathbf{a'}_N$ and $\mathbf{a}_{N+1}$ are the maximizers of G_N and G_{N+1}, respectively. Thus

$$\begin{aligned} \epsilon^{(N+1)} &\leq G_{N+1}(\mathbf{a}_{N+1}) - G_{N+1}(\mathbf{a}_{N+1} - a_{N+1}\mathbf{e}_{N+1}) \\ \epsilon^{(N+1)} &\geq \max_{\beta} G_{N+1}(\mathbf{a'}_{N+1} + \beta\mathbf{e}_{N+1}) - G_N(\mathbf{a'}_N) \end{aligned} \tag{8}$$

Expanding both the left-hand and right-hand sides, we get the conclusion of Theorem 1. The complete proof is provided in Appendix B. □

While Theorem 1 bounds the change on the hinge loss, the results are not directly applicable for active sampling. First, the upper bound only indicates largest possible change with no guarantee on the minimum change. While the lower bound provides such a guarantee and a non-zero lower bound indicates a SV, evaluating the lower bound still requires the label information (*i.e.,* t_{N+1}), which is not available during sampling. In the following corollary, we first simplify the bounds by considering a common type of kernel function and then derive a label-independent approximation of the lower bound to support active sampling.

Corollary 1. *When considering a square exponential kernel (or RBF kernel), the bounds on the change of the hinge loss can be simplified as*

$$\epsilon^{(N+1)} \leq a_{N+1}\left[1-(t_{N+1}y^{(N+1)}(\mathbf{x}_{N+1}))\right]+\frac{1}{2}a_{N+1}^2 \tag{9}$$

$$\epsilon^{(N+1)} \geq \begin{cases} 0, & t_{N+1}y^{(N)}(\mathbf{x}_{N+1})>1 \\ \frac{\left[1-t_{N+1}y^{(N)}(\mathbf{x}_{N+1})\right]^2}{2}, & \textit{Otherwise} \end{cases} \tag{10}$$

Furthermore, adding a sample $\mathbf{x}_{N+1}$ *with a small* $|y^{(N)}(\mathbf{x}_{N+1})|$ *guarantees a large change on the lower bound of the dual hinge loss.*

The proof of the simplified bound is straightforward by plugging in $k(\mathbf{x},\mathbf{x})=1$ for a RBF kernel. For the second part of the corollary, we can see that the lower bound is decided by $[1-t_{N+1}y^{(N)}(\mathbf{x}_{N+1})]^2$. While the label t_{N+1} is unavailable, we may still ensure this term is large when the absolute value of decision function evaluated on $\mathbf{x}_{N+1}$ (*i.e.,* $|y^{(N)}(\mathbf{x}_{N+1})|$) is small.

3.4. Mapping Hinge Loss Bounds to Active Sampling Functions for Exploration and Exploitation

The theoretical results as summarized by Theorem 1 and Corollary 1 provide important guidance to the design of the two sampling functions, including $\mathrm{F}_{\mathrm{disc}}$ and $\mathrm{F}_{\mathrm{rank}}$, as well as the two-phase D-TRUST sampling process to optimally balance exploration and exploitation. First, since the decision function can be evaluated without the labels, Corollary 1 suggests a theoretically sound way for actively sampling unlabeled instances that are likely to be SVs. In the simple binary case, it reduces to a commonly used sampling strategy that chooses data samples close to the current decision boundary. However, such a strategy cannot differentiate a sample's specific contribution in terms of exploration or exploitation. Furthermore, extension to multi-class setting is nontrivial as multiple decision boundaries are simultaneously involved. We propose to further conduct class-wise decomposition of the decision function to derive the *evidence* that supports assigning the data sample to each class (see Section 4.1 for details). Such a decomposition can identify two distinct scenarios, both of which can lead to a small $|y^{(N)}(\mathbf{x}_{N+1})|$. Intuitively, the first scenario corresponds to the case when the unlabeled sample $\mathbf{x}_{N+1}$ is far away from all the current training samples, *i.e.,* $k(\mathbf{x}_n,\mathbf{x}_{N+1})$ is small $\forall n\in[N]$; the second scenario corresponds to the case when $\mathbf{x}_{N+1}$ is located close to a mixed group of training samples from multiple classes. The first type of samples are instrumental for exploring the unlabeled pool while the second type can effectively exploit the labeled samples to refine the current decision boundary.

Besides the lower bound, the upper bound given in (9) implies that an SV with a large dual variable can incur a more significant change to the model to ensure faster convergence. From a geometric perspective, these SVs are located within the margin whose dual variables reach the maximum value as C. Intuitively, a small C corresponds to a large margin, which leads to a large number of SVs with many of them located within the margin and their dual variables tied at C. This makes it more difficult to identify the most important SVs and labeling all the SVs may incur a high annotation cost. In contrast, a large C will shrink the margin that reduces the number of SVs. Important SVs can also be more easily identified as only a few will be within the margin. While this can effectively reduce the annotation cost, a small margin can compromise the generalization capability of the actively learned model as evidenced by the margin based generalization bound (Mohri et al., 2018). In next section, we will formally investigate the impact of the margin constraint C on identifying the important samples to support fast convergence in AL as well as its important relationship to the total annotation cost.

4. Discover-Then-Rank Active Learning

We present the two-phase Discover-Then-Rank active learning process in this section based on the the theoretical results obtained in Section 3. We start by conducting decision function decomposition to derive the class-wise evidence to handle multiple classes. We then define the two sampling functions: $\mathrm{F}_{\mathrm{disc}}$ and $\mathrm{F}_{\mathrm{rank}}$, which focus on exploration and exploration of AL, respectively. Finally, we show how to integrate these two functions into an unified sampling process, which we refer to as D-TRUST.

4.1. Decision Function Decomposition

Given a set $\mathcal{M}$ of classes, we denote $y_m(\cdot)$ as the decision function of class $m\in\mathcal{M}$ obtained under OVR and a_{nm}'s the corresponding dual variables. We further decompose this decision function by collecting the class-wise evidence given by

$$D_m(\mathbf{x}) = \sum_{n=1,t_n=m}^{N} a_{nm} k(\mathbf{x}_n, \mathbf{x}) \tag{11}$$

Given an unlabeled sample $\mathbf{x}_{N+1}$, if $D_m(\mathbf{x}_{N+1})$ is small for all $m \in \mathcal{M}$, it implies (i) $|y_m^{(N)}(\mathbf{x}_{N+1})|$ is small, which guarantees a large change to the dual hinge loss if $\mathbf{x}_{N+1}$ is labeled, and (ii) $\mathbf{x}_{N+1}$ is far away from training samples from all classes, making it effective for exploration if labeled. In contrast, if $D_m(\mathbf{x}_{N+1})$ is simultaneously large for multiple classes, then $|y_m^{(N)}(\mathbf{x}_{N+1})|$ is still small as t_{nm} and $t_{nm'}$ take opposite signed in the decision function, which cancels out their corresponding class-wise evidences. This implies that $\mathbf{x}_{N+1}$ is located in a conflicting region and labeling it essentially exploits the current labeled data to improve the decision boundary. In what follows, we present the two sampling functions: Discovery (F_{disc}) and Ranking (F_{disc}), which leverage the class-wise evidences to identify two different types of samples as described above to best support exploration and exploitation of AL, respectively.

4.2. Exploration-Based Sampling Through Evidence-Aware Discovery

The Discovery function aims to discover data samples that are potentially SVs and can quickly shape the decision boundary. First, we construct the discovery function using the combination of proposed class-wise evidence measures. Then, we use uncertainty assistance to avoid the unstable issue in the beginning and propose a dynamic update mechanism for the augmented $\hat{\mathrm{F}}_{\text{disc}}$.

Evidence-aware discovery. Built upon the definition of class-wise evidence, we define Discovery sampling using a $\min\max()$ function that chooses data samples with small evidences from all classes

$$\mathbf{x}^* = \arg\min_{\mathbf{x}\in\mathcal{U}} \mathrm{F}_{\text{disc}}(\mathbf{x}), \quad \mathrm{F}_{\text{disc}}(\mathbf{x}) = \max_{m\in\mathcal{M}} \mathrm{D}_m(\mathbf{x}) \tag{12}$$

As we introduced in Section 4.1, each D_m provides a class-wise evidence for how much a data sample is supported by class m. By using the $\min\max()$ function, F_{disc} places a focus on data instances in the unlabeled pool that are dissimilar to the positive SVs from all the classes. It is most suitable for the early stage of AL, where the training data is limited and exploring the data space is critical.

Uncertainty assistance. To avoid constantly sampling outliers that are far away from the current training set, we design an augmented Discovery function by integrating F_{disc} with dual space induced uncertainty measurements to effectively explore the unknown but interesting regions in the data space. In particular, we consider a pair-wise confusion metric $\mathrm{UN}_{\text{pair}}$ and a general confusion metric UN_{all}. The pair-wise metric is to find the instance $\mathbf{x}^*$ with the smallest difference between its top two class-wise decision function values:

$$\mathrm{UN}_{\text{pair}}(\mathbf{x}) = y_{\hat{m}_1}(\mathbf{x}) - y_{\hat{m}_2}(\mathbf{x}) \tag{13}$$

where $\hat{m}_1, \hat{m}_2 \in \mathcal{M}$ are the labels with the two largest decision functions for $\mathbf{x}$. The pairwise measure only considers the top two probable classes and ignore the rest. To accommodate a large label space, we use the softmax function to convert $y_m(\mathbf{x})$ to a probability score $p(m|\mathbf{x})$ and compute the entropy over all classes, leading to the general confusion metric UN_{all}. The overall augmented Discovery function is

$$\hat{\mathrm{F}}_{\text{disc}}(\mathbf{x}) = \mathrm{F}_{\text{disc}}(\mathbf{x}) + \lambda \mathrm{UN}_{\text{pair}}(\mathbf{x}) - \gamma \mathrm{UN}_{\text{all}}(\mathbf{x}) \tag{14}$$

The two additional uncertainty metrics favor samples that are potentially misclassified and overall confusing, which facilitate $\hat{\mathrm{F}}_{\text{disc}}(\mathbf{x})$ in choosing informative samples while exploring the data space.

Dynamic update mechanism. We then propose an update mechanism that controls the impact of each factor in the sampling score using two balancing weights λ and γ. These two balancing parameters are both non-negative, as the model should choose samples with maximum UN_{all} score and minimum $\mathrm{UN}_{\text{pair}}$ and $\mathrm{F}_{\text{disc}}(\mathbf{x})$ scores.

The dynamic update mechanism has two turning points in the early stage, which determine the choice between the two following specific update rules:

$$\lambda = \lambda_0 - i\Delta\lambda, \quad \gamma = 0 \tag{15}$$

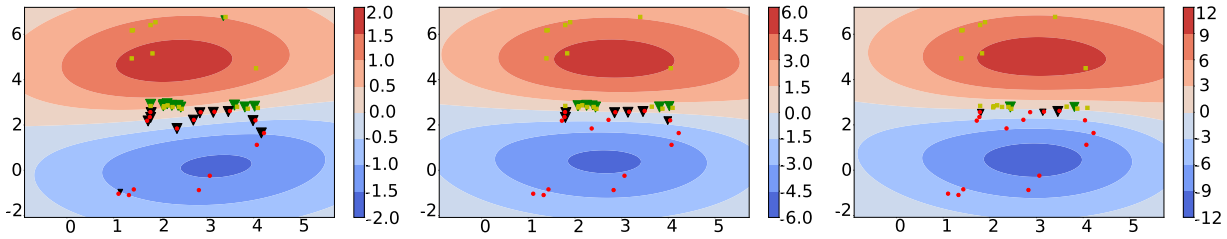
$$\lambda = \max\{\lambda_{old} - \Delta\lambda, 0\}, \quad \gamma = \gamma_0 + i\Delta\gamma \tag{16}$$

where λ_0, γ_0 are set during the initialization, i is the number of iterations, and $\Delta\lambda, \Delta\gamma$ are the changing rate.

Empirically, UN_{all} is less accurate, hence not effective when the training samples are very sparse. In contrast, $\mathrm{UN}_{\text{pair}}$ can measure the local uncertainty more correctly in the very early stage. Thus, λ is assigned a larger value at the beginning by (15). Later, we increase the impact of UN_{all} and reduce the weight of $\mathrm{UN}_{\text{pair}}$, meanwhile let $\mathrm{F}_{\text{disc}}(\mathbf{x})$ dominate by increasing $\Delta\lambda$ and transitioning into Eq. (16). More details are provided in Appendix D.3.

4.3. Exploitation-Based Sampling Through Active Ranking Based on Estimated Dual Variables

The Ranking function aims to identify data samples with the largest impact to the decision boundary and labeling these samples can ensure a fast convergence of AL with reduced annotation cost. The upper bound on the change of the dual hinge loss derived in both Theorem 1 and Corollary 1 shows that the dual variable can serve as good indicator to rank the data samples. However, designing the Ranking function faces two major challenges. First, the dual variables are tightly coupled with the margin constraint C and it is important to understand the impact of the margin constraint so that important samples can be properly separated to avoid a high annotation cost in AL while ensuring good generalization of the actively learned model. Second, the dual variable value is not available until the label is assigned,



(a) SVs of $C = 1$ (b) SVs of $C = 10$ (c) SVs of $C = 100$

Figure 2: SVs and magnitudes of corresponding dual variables compared to C: The size of the triangle markers indicates the magnitude of the corresponding a_n/C. The background contour map shows the decision function $y(\mathbf{x})$ from (1).

Table 1: Impact of margin constraint C

C value	1	10	100
Model accuracy	0.99	0.99	0.98
Number of SVs	21	10	6
Mean dual variable value of SVs	0.88	7.21	50.46
Number of $a_n = C$ SVs	17	6	2

making it not applicable for active sampling. We propose to leverage the geometric property of the dual space and use class-wise evidence to define a proxy measure of the dual variable values without label information.

Impact of the margin constraint. Since the margin constraint C upper bounds the dual variable values, in order to be able to rank the importance of data samples according to their dual variable values, it is necessary to set a relatively large C value. From the geometric perspective, a small C leads to a large margin, which allows many SVs to be located within the margin. As a result, their dual variables are all tied at C, making it impossible to differentiate their relative importance. This phenomenon is also empirically verified by results from a synthetic dataset as shown in Table 1 and Figure 2.

From the table, we see that the number of SVs decreases as we increase the C value. Meanwhile, each SV plays a more important role as reflected by much larger dual variable values on average. As shown in Figure 2, the SV locations are according to either $a_n = C$ or $a_n < C$, consistent with our geometric interpretation of dual variables in Section 3.2. As we can see in Figure 2 (a), when C is too small, the decision boundary is less accurate, and there are too many SVs within the margin. In Figure 2 (c), when C is too large, although the decision boundary is close to the optimal (Figure 2 (b)), there are too few SVs and it would cause a problem if there is overlapping in the dataset.

We have discovered that for larger C, there is potentially more distinction between important SVs and other SVs. However, setting a very large C value will hurt the generalization power of the model (reflected by a lower accuracy in Table 1). This can be more formally verified through the margin theory (Mohri et al., 2018):

$$R(h) \leq \hat{R}_\rho(h) + \frac{2}{\rho}\hat{\mathfrak{R}}_S(\mathcal{H}) + 3\sqrt{\frac{\log\frac{2}{\delta}}{2n}} \tag{17}$$

where $R(h)$ is the true generalization error, $\hat{R}_\rho(h)$ is the empirical error, $\mathcal{H}$ is a hypothesis set and $\hat{\mathfrak{R}}_S(\mathcal{H})$ is the empirical Rademacher complexity of $\mathcal{H}$. For any $\delta > 0$, $\forall h \in \mathcal{H}$, the above holds with probability at least $1 - \delta$ given fixed $\rho > 0$, where ρ is defined as the margin. In SVM, $C \propto \frac{1}{\rho^2}$ and if C is set too large, the margin will be too small. Thus, the coefficient $\frac{2}{\rho}$ of the Rademacher term will be large, which will result in a loose error bound.

Remark. Since the margin constraint C determines the size of the margin, which in turn controls the number of SVs, the selection of C can be connected with the available annotation budget under the AL setting if the goal is to sample and label the important SVs. Intuitively, for a relatively small annotation budget, a large C should be chosen (which may compromise the generalization as expected); for a larger budget, a smaller C could be used to improve the generalization power. However, it should still be larger than the passive setting to allow the AL model to identify more important instances during sampling by avoiding a large number dual variables to reach their upper bound at C.

A proxy measure of dual variables. Since the dual variable values are not available before the labels are assigned, we propose a proxy measure to approximate the dual variable value without using the label information. A key intuition is that an SV with a large dual variable (*e.g.*, $a = C$) is located within the margin. Geometrically, these SVs are usually located in a conflicting region in the data space that is surrounded by a mixed group of data samples from multiple classes. To this end, we propose to measure the level of conflict for an unlabeled sample using a dissonance-based measurement as a proxy to the dual variable:

$$\mathrm{F}_{\text{rank}}(\mathbf{x}) = \sum_{t\in\mathcal{M}} \left(\frac{D_t \sum_{j\neq t} D_m \mathrm{Bal}(D_m, D_t)}{\sum_{m\neq t} D_m} \right) \tag{18}$$

where $\mathrm{Bal}(D_m, D_t) = 1 - \frac{|D_m - D_t|}{D_m + D_t}$ if $D_m D_t \neq 0$ and 0 otherwise. Dissonance has also been formulated under the

Subjective Logic framework (Jøsang, 2016). As a second-order uncertainty, it is caused by the conflict of strong evidence. Hence, a higher F_{rank} score favors samples causing a high conflict among multiple classes. Different from F_{disc}, a high F_{rank} score requires large evidences from multiple classes simultaneously, which ensures that the data sample is located in a conflicting region w.r.t. the current training samples. Since such regions are more likely to be within the margin, we expect these data samples to have large dual variables. Having this, the Ranking sampling chooses a data sample with the largest F_{rank} score: $\mathbf{x}^* = \arg\max_{\mathbf{x}\in\mathcal{U}} \mathrm{F}_{\text{rank}}$.

4.4. Transition from Discovery to Ranking

While F_{disc} and F_{rank} perform exploration and exploitation effectively, it is important to arrange them properly to formulate an overall active sampling process to achieve a good balance between these two essential and complementary sampling behaviors. We propose to leverage the dual variable values of labeled samples (from the initial pool and the sampled ones) to determine a transition threshold from discovery to ranking based sampling. In particular, with effective exploration, more SVs can be identified that help to shape the correct decision boundary. When the decision boundary is shaping, more SVs will expand the margin that increase the number of SVs inside the margin. Since their corresponding dual variables are positive, when taking the average over all the labeled samples, the mean dual variable value should increase steadily. Such an increasing trend indicates the increase in the volume of the margin. When the volume is sufficiently large, we should transit into exploitation using ranking sampling. The transition criterion is triggered if the mean dual variable value surpasses a threshold based on the initial value:

$$\frac{\sum_n \sum_m a_{nm}}{N_{SV}} > \eta \frac{\sum_{n'} \sum_m a^{(0)}_{n'm}}{N^{(0)}_{SV}} \tag{19}$$

where N_{SV} and $N^{(0)}_{SV}$ are the total number of SVs in the current iteration and the initial pool, respectively. With this transition criterion, we connect our discover sampling and ranking sampling. We show the overall active sampling process in Algorithm 1 of Appendix C. The transition parameter η could be set by considering the margin constraint C based on our prior discussion on the impact of the margin constraint. Empirically, any $\eta > 1$ works well and we set $\eta = 1.5$ in our experiments.

5. Experiments

We conduct experiments on both synthetic and real data to assess the effectiveness of the proposed D-TRUST AL process. The synthetic experiments are designed to verify important theoretical properties and the desired sampling behavior of the two sampling functions. Limited by space, we present the synthetic experiment results in Appendix D. In the real data experiments, we compare D-TRUST with competitive AL baselines to demonstrate its state-of-the-art AL performance. We also investigate the impact of key model parameters through a detailed ablation study.

Table 2: Dataset property descriptions

Dataset	$N_{\mathcal{U}} + N_{\mathcal{L}}$	$\dim(\mathbf{x})$	$\lvert\mathcal{C}\rvert$	**Domain**
Dermatology 1	800	1391	50	Medical
Dermatology 2	868	1554	30	Medical
Yeast	1484	8	10	Biology
USPS	9298	256	10	Image
Auto-drive	58509	48	11	Auto
Penstroke	1144	500	26	Image

Datasets, experiment setup, and comparison baselines. We conduct AL experiments on six real-world datasets, which are summarized in Table 2. In these experiments, we start with the same labeled training set $\mathcal{L}_0$ and record the test accuracy of the model trained over each AL iteration. The detailed parameter settings are described in Appendix D. To demonstrate the effectiveness of D-TRUST, we compare with several competitive baselines: Convex Hull-based sampling (*MC-CH*) (Shi & Yu, 2018), QUIRE sampling (*QUIRE*) (Huang et al., 2010), Graph Density-based sampling (*GD*) (Ebert et al., 2012), Best vs Second Best sampling (*BvSB*) (Joshi et al., 2009), and Entropy-based sampling (*Entr*) (Wu et al., 2004).

AL performance comparison. For each model using different sampling approaches, we plot the model accuracy on a left-out test set during each AL iteration, through 500 iterations in total. In Figure 3, we show that the proposed D-TRUST converges faster than all the compared ones. All the models tested use SVM as the base classifier, so any difference in the predictive performance is a result of the design of the sampling function.

The proposed D-TRUST model has dominant performance on all datasets, showing strong adaptation power on different types of domains. MC-CH achieves the second best performance as it also effectively controls the exploration-exploitation balance. BvSB shows good performance in the early stage because of the local uncertainty is more reliable when the training set is small. Entr sampling shows a comparable performance with BvSB. However, on some datasets, it performs poorly initially, which is due to the inaccurately estimated probabilities from the limited amount of training data. GD and QUIRE also show competitive performance to BvSB sampling on certain datasets, but they struggle on the Dermatology datasets, which contain relatively large number of classes. D-TRUST also exhibits a clear advantage in the early stage of AL, which verifies the effectiveness of evidence-based exploration.

Ablation study. In this section, we show the impact of key model parameters in Figure 4. We initialize λ and γ in a range such that the scale of the different scores

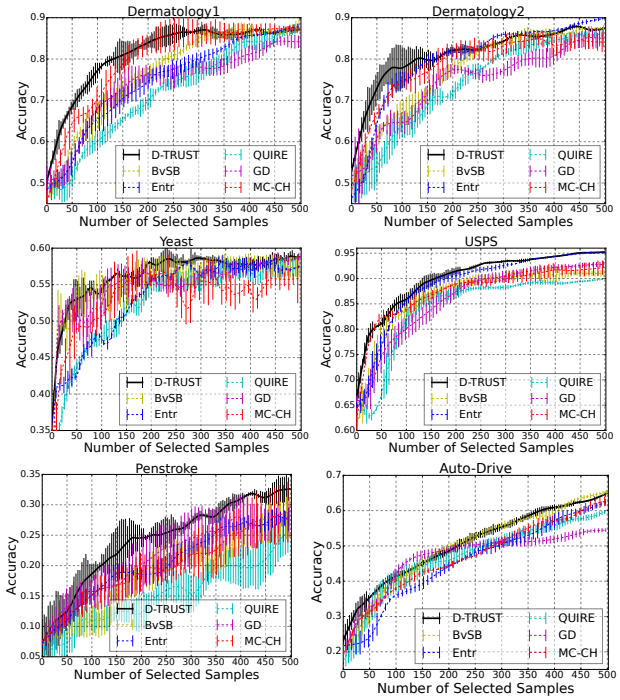
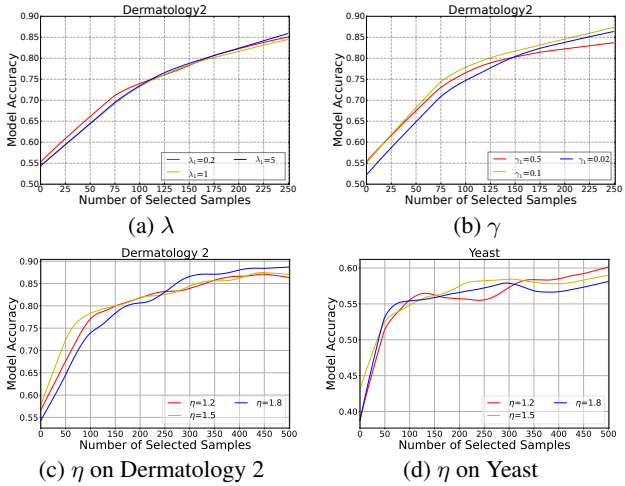


Figure 3: AL performance comparison



(a) λ (b) γ

(c) η on Dermatology 2 (d) η on Yeast

Figure 4: Impact of balancing and transition parameters

F_{disc}, UN_{all}, UN_{pair} are comparable. We set the update rates $\Delta\lambda$ and $\Delta\gamma$ to adjust λ and γ dynamically. The detailed update rules are introduced in Appendix D. We use λ_1 and γ_1 to denote the largest value that λ and γ take during the entire AL process and provide the ablation study on different λ_1 and γ_1 values. For the transition parameter η, we set the values based on the number of classes and the actual average dual variables on the datasets (which depend on C). Here we show two example datasets on the effect of η. More results are presented in Appendix D. The results show that D-TRUST is robust to the change of balancing parameters within a certain range. For the transition parameter, it affects the middle stage if the transition happens too early or too late, but the overall performance is still stable when η is around 1.5 for most datasets. Thus in real-world applications, these parameters can be chosen effortlessly, as long the different components in the F_{disc} function have similar scales.

6. Discussions and Future Directions

In this work, we primarily focus on the classical sample-starved AL setting, where data annotation is costly and labels are expensive to acquire. It is worth noting that a majority of AL research in recent years has shifted the focus to leveraging deep learning models for AL. As a result, most of these works assume that a large batch of actively sampled data instances can be labeled in each AL iteration to avoid frequent updates of a deep neural network (DNN). We believe it is important to re-ignite the interest in advancing the AL research in the classical sample-starved setting. We approach this classical problem from a novel perspective by leveraging the key properties of the dual space of a sparse kernel max-margin predictor. However, we do acknowledge the extension to DNNs as an important future direction. The extension is feasible by leveraging the recent efforts in bridging SVMs and DNNs (*e.g.,* (Tang, 2013)) or deep kernel learning methods (*e.g.,* (Wilson et al., 2016)). There are also interesting works that utilize max-margin models as a component in the meta-learning framework ((Lee et al., 2019)). Thus, we can also further investigate combining max-margin based AL with other learning paradigms such as meta-learning or reinforcement learning.

7. Conclusion

In this paper, we develop a theoretical foundation that allows us to conduct principled active sampling in the dual space induced by a sparse maximum-margin predictor. We derive both the lower and upper bounds of a hinge loss in the dual form, which can guide active sampling to identify potential SVs and those with a large impact on the decision boundary. A two-phase active sampling process dynamically transitions from discovery to ranking sampling by monitoring the change on the mean of the dual variable values. As a result, this process achieves an automatic balance between exploration and exploitation for effective active sampling. Experimental results and comparison with competitive baselines justify the effectiveness of the proposed dual space induced active sampling process.

Acknowledgements

This research was supported in part by an NSF IIS award IIS-1814450 and an ONR award N00014-18-1-2875. The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing any funding agency.

References

Burges, C. J. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, Jun 1998. ISSN 1573-756X. doi: 10.1023/A:1009715923555. URL https://doi.org/10.1023/A:1009715923555.

Culotta, A. and McCallum, A. Reducing labeling effort for structured prediction tasks. In *AAAI*, volume 5, pp. 746–751, 2005.

Demir, B., Persello, C., and Bruzzone, L. Batch-mode active-learning methods for the interactive classification of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 49(3):1014–1031, 2010.

Ebert, S., Fritz, M., and Schiele, B. Ralf: A reinforced active learning formulation for object class recognition. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3626–3633. IEEE, 2012.

Huang, S.-J., Jin, R., and Zhou, Z.-H. Active learning by querying informative and representative examples. In *Advances in neural information processing systems*, pp. 892–900, 2010.

Huo, L.-Z. and Tang, P. A batch-mode active learning algorithm using region-partitioning diversity for svm classifier. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(4):1036–1046, 2014.

Jiang, H. and Gupta, M. Minimum-margin active learning. *arXiv preprint arXiv:1906.00025*, 2019.

Jøsang, A. *Subjective logic*. Springer, 2016.

Joshi, A. J., Porikli, F., and Papanikolopoulos, N. Multi-class active learning for image classification. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2372–2379. IEEE, 2009.

Lee, K., Maji, S., Ravichandran, A., and Soatto, S. Meta-learning with differentiable convex optimization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10657–10665, 2019.

Lewis, D. D. and Catlett, J. Heterogeneous uncertainty sampling for supervised learning. In *Machine learning proceedings 1994*, pp. 148–156. Elsevier, 1994.

Lewis, D. D. and Gale, W. A. A sequential algorithm for training text classifiers. In *SIGIR'94*, pp. 3–12. Springer, 1994.

Li, X. and Guo, Y. Active learning with multi-label svm classification. In *IjCAI*, pp. 1479–1485. Citeseer, 2013.

Lin, C. H., Mausam, M., and Weld, D. S. Re-active learning: Active learning with relabeling. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

Liu, Y. and Zheng, Y. F. One-against-all multi-class svm classification using reliability measures. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, volume 2, pp. 849–854. IEEE, 2005.

Mahmood, R., Fidler, S., and Law, M. T. Low-budget active learning via wasserstein distance: An integer programming approach. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=v8OlxjGn23S.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. *Foundations of machine learning*. MIT press, 2018.

Osugi, T., Kim, D., and Scott, S. Balancing exploration and exploitation: A new algorithm for active machine learning. In *Fifth IEEE International Conference on Data Mining (ICDM'05)*, pp. 8–pp. IEEE, 2005.

Platt, J. C. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *ADVANCES IN LARGE MARGIN CLASSIFIERS*, pp. 61–74. MIT Press, 1999.

Scheffer, T., Decomain, C., and Wrobel, S. Active hidden markov models for information extraction. In *International Symposium on Intelligent Data Analysis*, pp. 309–318. Springer, 2001.

Settles, B. and Craven, M. An analysis of active learning strategies for sequence labeling tasks. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pp. 1070–1079, 2008.

Shannon, C. E. A mathematical theory of communication. *Bell system technical journal*, 27(3):379–423, 1948.

Shen, Y., Yun, H., Lipton, Z. C., Kronrod, Y., and Anandkumar, A. Deep active learning for named entity recognition. *arXiv preprint arXiv:1707.05928*, 2017.

Shi, W. and Yu, Q. An efficient many-class active learning framework for knowledge-rich domains. In *2018 IEEE International Conference on Data Mining (ICDM)*, pp. 1230–1235. IEEE, 2018.

Tan, W., Du, L., and Buntine, W. Diversity enhanced active learning with strictly proper scoring rules. *Advances in Neural Information Processing Systems*, 34, 2021.

Tang, Y. Deep learning using linear support vector machines. *arXiv preprint arXiv:1306.0239*, 2013.

Tong, S. and Koller, D. Support vector machine active learning with applications to text classification. *Journal of machine learning research*, 2(Nov):45–66, 2001.

Vapnik, V. Chervonenkis: On the uniform convergence of relative frequencies of events to their probabilities. *Soviet Mathematics: Doklady*, 9, 01 1968.

Wang, Z., Du, B., Zhang, L., Zhang, L., and Jia, X. A novel semisupervised active-learning algorithm for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(6):3071–3083, 2017.

Wilson, A. G., Hu, Z., Salakhutdinov, R., and Xing, E. P. Deep kernel learning. In *Artificial intelligence and statistics*, pp. 370–378. PMLR, 2016.

Wu, T.-F., Lin, C.-J., and Weng, R. C. Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine Learning Research*, 5(Aug):975–1005, 2004.

Xie, X. Sampling active learning based on non-parallel support vector machines. *Neural Processing Letters*, 53 (3):2081–2094, 2021.

Xu, H., Bie, X., Feng, H., and Tian, Y. Multiclass svm active learning algorithm based on decision directed acyclic graph and one versus one. *Cluster Computing*, 22(3): 6241–6251, 2019.

Yang, L., Zhang, Y., Chen, J., Zhang, S., and Chen, D. Z. Suggestive annotation: A deep active learning framework for biomedical image segmentation. In *International conference on medical image computing and computer-assisted intervention*, pp. 399–407. Springer, 2017.

Yin, C., Qian, B., Cao, S., Li, X., Wei, J., Zheng, Q., and Davidson, I. Deep similarity-based batch mode active learning with exploration-exploitation. In *2017 IEEE International Conference on Data Mining (ICDM)*, pp. 575–584. IEEE, 2017.

Yoo, D. and Kweon, I. S. Learning loss for active learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 93–102, 2019.

Zhou, J. and Sun, S. Gaussian process versus margin sampling active learning. *Neurocomputing*, 167:122–131, 2015.

Appendix

Organization of Appendix. In this Appendix, we organize additional content as follows. We first provide the summary of the notations in Appendix A. Then, we present the complete proof for the theoretical analysis in the main paper in Appendix B. We provide an algorithm overview of the proposed D-TRUST active sampling strategy in Appendix C. Then, we present more experimental results, including the synthetic dataset and more illustrative examples, detailed real-data experimental settings, and additional results including more ablation studies in Appendix D. We discuss the potential societal impacts, limitations, and possible extensions of the current work in Appendix E. Finally, the link to the source code is provided in Appendix F.

A. Summary of Notations

Table 3: Summary of key notations with definitions

Notation	Definition	Type
$\mathbf{x}_n$	Feature vector of the n-th data sample	Observed
t_n	True label of $\mathbf{x}_n$	Observed
$\tilde{t}_{nm}$	Binary coded label of $\mathbf{x}_n$ for class t	Observed
$\phi(\mathbf{x}_n)$	Basis function at $\mathbf{x}_n$	Implicit
$\mathbf{w}, b$	Weight vector and intercept in decision function definition	Implicit
$y(\mathbf{x}_n)$	Decision function of $\mathbf{x}_n$	Computed
a_{nm}	Dual variable for $\mathbf{x}_n$ and class t	Computed (Dual representation)
$\mathbf{x}^*$	Selected data sample by sampling function	
D_m	Class-wise Evidence of label m	Computed
λ, γ	Balancing parameters in the unified sampling function	Hyperparameter
λ_0, γ_0, $\Delta\lambda, \Delta\gamma$, λ_1, γ_1	Initial values, update rates, and the largest values of balancing parameters in the unified sampling function	Hyperparameter
η	Threshold parameter for the transition from F_{disc} to F_{rank}	Hyperparameter

B. Proof of Theoretical Results

Proof of Theorem 1

Proof. First we write the Lagrangian form of the losses:

$$G_N(\mathbf{a'}_N) = \sum_{i=1}^{N} a'_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} a'_i a'_j t_i t_j k(\mathbf{x}_i, \mathbf{x}_j) \tag{20}$$

$$G_{N+1}(\mathbf{a}_{N+1}) = \sum_{i=1}^{N+1} a_i - \frac{1}{2}\sum_{i=1}^{N+1}\sum_{j=1}^{N+1} a_i a_j t_i t_j k(\mathbf{x}_i, \mathbf{x}_j) \tag{21}$$

We also define $\mathbf{a'}_{N+1} = (a'_1, a'_2, ..., a'_N, 0)^T$, $\mathbf{e}_{N+1} = (0, .., 0, 1)^T$, and $0 \leq \beta \leq C$. By definition, $\mathbf{a'}_N$ and $\mathbf{a}_{N+1}$ are the maximizers of G_N and G_{N+1}, thus

$$\begin{aligned}
&\max_{\beta} G_{N+1}(\mathbf{a}'_{N+1} + \beta\mathbf{e}_{N+1}) - G_N(\mathbf{a}'_N) \\
\leq& G_{N+1}(\mathbf{a}_{N+1}) - G_N(\mathbf{a}'_N) \\
\leq& G_{N+1}(\mathbf{a}_{N+1}) - G_{N+1}(\mathbf{a}_{N+1} - a_{N+1}\mathbf{e}_{N+1})
\end{aligned} \tag{22}$$

The lower bound:

$$\begin{aligned}
&\max_{\beta} G_{N+1}(\mathbf{a}'_{N+1} + \beta \mathbf{e}_{N+1}) - G_N(\mathbf{a}'_N) \\
=&\max_{\beta} \beta - \sum_{i=1}^{N} a'_i \beta t_i t_{N+1} k(\mathbf{x}_i, \mathbf{x}_{N+1}) - \frac{1}{2}\beta^2 k(\mathbf{x}_{N+1}, \mathbf{x}_{N+1}) \\
=&\max_{\beta} \beta(1 - t_{N+1} y^{(N)}(\mathbf{x}_{N+1})) - \frac{1}{2}\beta^2 k(\mathbf{x}_{N+1}, \mathbf{x}_{N+1})
\end{aligned} \tag{23}$$

We use $y^{(N)}(\mathbf{x}) = \sum_{i=1}^{N} a'_i \beta t_i t k(\mathbf{x}_i, \mathbf{x})$. The similar substitution is used in the following proof as well. When $1 - t_{N+1} y^{(N)}(\mathbf{x}_{N+1}) < 0$, the β solution for the max function is 0, thus the lower bound is also 0 in this case. Otherwise, the β solution is $\frac{1 - t_{N+1} y^{(N)}(\mathbf{x}_{N+1})}{2k(\mathbf{x}_{N+1}, \mathbf{x}_{N+1})}$, the corresponding lower bound is $\frac{(1 - t_{N+1} y^{(N)}(\mathbf{x}_{N+1}))^2}{2k(\mathbf{x}_{N+1}, \mathbf{x}_{N+1})}$. We use the RBF kernel which makes $k(\mathbf{x}_{N+1}, \mathbf{x}_{N+1}) = 1$. The result can be concluded as $\frac{(1 - t_{N+1} y^{(N)}(\mathbf{x}_{N+1}))^2}{2}$.

The upper bound:

$$\begin{aligned}
&G_{N+1}(\mathbf{a}_{N+1}) - G_{N+1}(\mathbf{a}_{N+1} - a_{N+1}\mathbf{e}_{N+1}) \\
=&a_{N+1} - \sum_{i=1}^{N+1} (t_i a_i)(t_{N+1} a_{N+1}) k(\mathbf{x}_i, \mathbf{x}_{N+1}) + \frac{1}{2} a_{N+1}^2 k(\mathbf{x}_{N+1}, \mathbf{x}_{N+1}) \\
=&a_{N+1}(1 - (t_{N+1} y^{(N+1)}(\mathbf{x}_{N+1}))) + \frac{1}{2} a_{N+1}^2 k(\mathbf{x}_{N+1}, \mathbf{x}_{N+1})
\end{aligned} \tag{24}$$

If $\mathbf{x}_{N+1}$ is not an SV given $\mathbf{a}_{N+1}$, above is equal to 0. Otherwise, $1 - (t_{N+1} y^{(N+1)}(\mathbf{x}_{N+1})) = \xi_{N+1}$. Because $0 \leq a_{N+1} \leq C$, the upper bound increases as a_{N+1} increases. We use the RBF kernel which makes $k(\mathbf{x}_{N+1}, \mathbf{x}_{N+1}) = 1$. The result can be concluded as $a_{N+1}\left[1 - (t_{N+1} y^{(N+1)}(\mathbf{x}_{N+1}))\right] + \frac{1}{2} a_{N+1}^2$. □

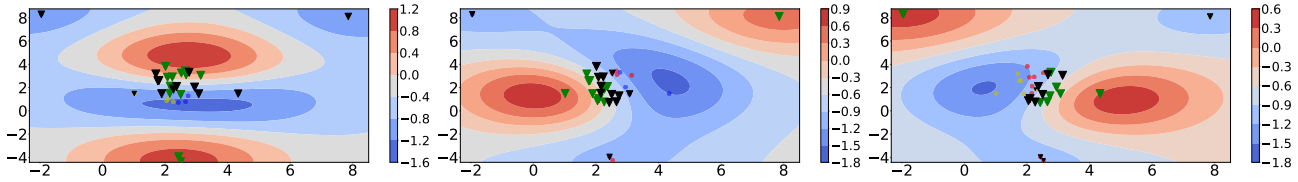
C. Algorithm

Algorithm 1 presents the detailed active sampling process.

D. Additional Experiments

D.1. Synthetic Data Results

Here we use the same synthetic dataset as Figure 1 to show the distribution of important SVs. In Figure 5, we show the SVs from each binary SVM and the current decision function built by them. As can be seen, an effective exploration-oriented model is able to discover the minority groups, thus the overall decision function $y(\mathbf{x})$ has two peaks in each binary SVM. We show that the SVs near the center of the majority groups have larger dual variable values, as indicated by a larger marker size, than the far away ones. The SVs with smaller dual variable values have less impact on the decision function.



(a) SVs of Class 1 (b) SVs of Class 2 (c) SVs of Class 3

Figure 5: SVs and magnitudes of corresponding dual variables a_n of each binary SVM. The color-coded circles mark the current training points. The triangle markers are the SVs, green stands for the positive class, and black stands for the negative class. The size of the triangle markers indicates the magnitude of the corresponding a_n. The background contour map shows the decision function $y_m(\mathbf{x})$ from (1).

Algorithm 1: D-TRUST Active Sampling

Result: the final $\mathcal{L}$, test accuracy at each iteration

initialization: max number of iterations I, length scale parameter l for the RBF kernel
split the data into the labeled training set $\mathcal{L}$, the unlabeled candidate set $\mathcal{U}$ (size$> I$), and the test set $\mathcal{T}$
while *iterations* $i < I$ **do**
  adjust λ, γ according to i
  fit the classifier based on $\mathcal{L}$
  save the predictive accuracy acc based on $\mathcal{T}$
  get predicted probabilities p and dual variable values a_{nm} based on $\mathcal{U}$
  if $\frac{\sum_n \sum_m a_{nm}}{N_{SV}} \leq \eta \frac{\sum_{n'} \sum_m a^{(0)}_{n'm}}{N^{(0)}_{SV}}$ **then**
    compute D_m using
    $D_m = \sum_{n=1,\tilde{t}_{nm}=1}^{N_{\mathcal{L}}} a_{nm} k(\mathbf{x}_n, \mathbf{x})$
    compute sampling score
    $\mathrm{F}_{\text{disc}}(\mathbf{x}) = \max_{m \in \mathcal{C}} D_m$
    $\hat{\mathrm{F}}_{\text{disc}}(\mathbf{x}) = \mathrm{F}_{\text{disc}}(\mathbf{x}) + \lambda \mathrm{UN}_{\text{pair}}(\mathbf{x}) - \gamma \mathrm{UN}_{\text{all}}$
    select the instance $\mathbf{x}^* = \arg\min_{\mathbf{x} \in \mathcal{U}} \hat{\mathrm{F}}_{\text{disc}}(\mathbf{x})$
    $\mathcal{U} = \mathcal{U} \backslash \mathbf{x}^*$; $\mathcal{L} = \mathcal{L} \cup \{\mathbf{x}^*\}$
  else
    compute F_{rank} using $\mathrm{F}_{\text{rank}} = \mathrm{diss}(\mathbf{x})$
    select the instance $\mathbf{x}^* = \arg\max_{\mathbf{x} \in \mathcal{U}} \mathrm{F}_{\text{rank}}(\mathbf{x})$
    $\mathcal{U} = \mathcal{U} \backslash \mathbf{x}^*$; $\mathcal{L} = \mathcal{L} \cup \{\mathbf{x}^*\}$
  end
end

D.2. General Parameter Settings

In the main experiments, we apply the same margin threshold $C = 10$ for all datasets. We also adopt the same RBF kernel for all datasets. The characteristic length scale l for the kernel is chosen based on the scale of the input features $\mathbf{x}$. We provide a brief ablation study of C and l in Table 4.

Table 4: Model performance using various C and l

Dataset	Description		Method	Number of selected samples				
	C	l		100	200	300	400	500
Dermatology1	100	1	D-TRUST	77.0	87.0	91.0	93.0	92.5
			BvSB	76.0	84.0	86.0	88.0	87.5
	1	1	D-TRUST	78.0	87.0	88.5	89.5	89.0
			BvSB	73.5	86.5	88.0	88.0	88.0
	10	0.001	D-TRUST	72.5	81.5	86.0	88.0	88.5
			BvSB	67.5	80.0	86.0	86.0	86.5
USPS	100	0.01	D-TRUST	90.0	92.0	94.0	95.2	95.3
			BvSB	87.3	90.7	92.6	94.3	94.7
	1	0.01	D-TRUST	88.9	92.3	93.9	94.8	94.8
			BvSB	88.8	91.6	93.7	94.4	94.6
Penstroke	50	1	D-TRUST	15.7	18.0	25.4	29.6	29.9
			BvSB	15.1	17.1	16.9	23.4	25.7
	0.1	1	D-TRUST	16.9	20.7	26.9	29.6	31.4
			BvSB	13.0	19.8	22.5	24.0	25.4

D.3. Dynamic Update Rules of $\hat{\mathrm{F}}_{\text{disc}}$

In the discovery stage, we use a dynamic-update mechanism to fully exhibit the exploration power of F_{disc} under the uncertainty-guided regularization from $\mathrm{UN}_{\text{pair}}$ and UN_{all}. As introduced in the main paper, the UN_{all} and $\mathrm{UN}_{\text{pair}}$ measures are to help us not only find the samples that have the smallest overall $|y(\mathbf{x})|$, but also are confusing to the current model and more likely to be wrongly classified. We here provide more details about the update mechanism that controls the balance by

changing weights λ and γ.

The key idea of the dynamic-update mechanism is to have two turning points in the early stage. Empirically, UN_{all} is less accurate and effective when the training samples are very limited, while UN_{pair} can measure the local uncertainty more correctly in the very early stage. Thus, λ is assigned with a larger value at the beginning, while γ is kept at 0. During this period, the update rule is

$$\lambda = \lambda_0 - i\Delta\lambda, \quad \gamma = 0 \tag{25}$$

where λ_0 is set during the initialization, i is the number of iterations, and $\Delta\lambda$ is the changing rate.

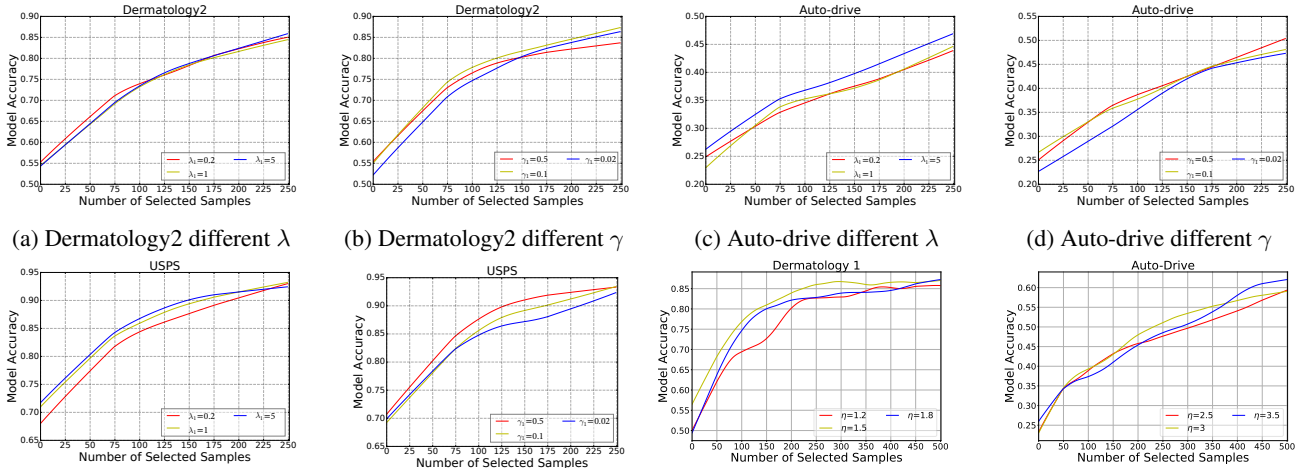
The first turning point is when a decent amount of data instances are added to $\mathcal{L}$. We increase the impact of UN_{all} and reduce the weight of UN_{pair}. We also want exploration to play a bigger role in this stage, which means $\text{F}_{\text{disc}}(\mathbf{x})$ should dominate. We thus increase $\Delta\lambda$ to make sure of this. At the same time, γ is slowly increased, so that we can smoothly transition into the exploitation-oriented later stage. The rate is controlled by a small $\Delta\lambda$. During this period, the update rule is

$$\lambda = \max\{\lambda_{old} - \Delta\lambda, 0\}, \quad \gamma = \gamma_0 + i\Delta\gamma \tag{26}$$

A larger λ_0 (*i.e.,* 5) is chosen for datasets with a large number of classes (*e.g.,* Derm 1&2) to allow DF_{pair} to stand out in the early phase of AL. Otherwise, λ is set smaller (*i.e.,* 0.5-1). We set γ_0 as 0 as we primarily trust DF_{pair} in the beginning. The $\Delta\lambda$ and $\Delta\gamma$ values are set to be 0.01-0.05 and 0.001-0.005, giving the parameters a moderate change. In the ablation study, we use λ_1 and γ_1 to denote the largest value that λ and γ take during the whole AL process. Finally, we use η to switch exploration to exploitation. We set a larger η for datasets with more classes, which usually require longer exploration due to the complex interaction among classes. The unlabeled pool also has an impact on η because a bigger pool requires more exploration. The specific value for each dataset is: [Derm1, 1.5], [Derm2, 1.2], [Yeast, 1.5], [USPS, 2.0], [Auto-drive, 3.0], [Penstroke, 1.5]. For practical use, as long as the balancing parameters are not set to extreme values such that the components of the sampling score are not comparable, the performance should not deteriorate much. One can always use a small hold-out dataset to cross-validate the parameters first.

D.4. Additional Ablation Study

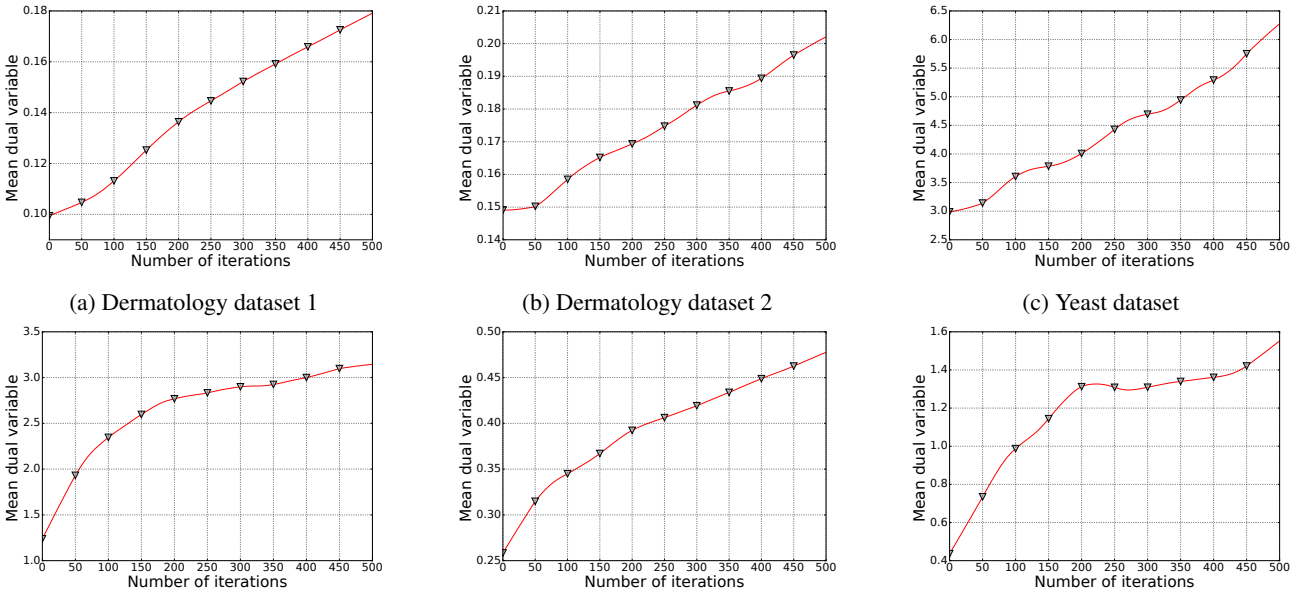
In this section, we provide more results from the ablation study on different λ_1 and γ_1 values, together with the transition parameter η. In Figure 6 (a)-(f) average predictive accuracy over the test set obtained from repeated experiments under different hyperparameter settings. The results show that D-TRUST is robust to the change of balancing parameters within a certain range. For example, the parameters can change to five times as big or as small without hurting the performance too much. Thus in real-world applications, these parameters can be chosen effortlessly, as long the different components in the F_{disc} function have similar scales.



(a) Dermatology2 different λ (b) Dermatology2 different γ (c) Auto-drive different λ (d) Auto-drive different γ

(e) USPS different λ (f) USPS different γ (g) Dermatology1 different η (h) Auto-drive different η

Figure 6: Ablation study on balancing and transitioning parameters



(a) Dermatology dataset 1 (b) Dermatology dataset 2 (c) Yeast dataset

(d) USPS dataset (e) Penstroke dataset (f) Auto-drive dataset

Figure 7: The increasing trend of mean dual variable values during AL

In Figure 7, we show the increasing trend of $\frac{\sum_n \sum_m a_{nm}}{N_{SV}}$ over the AL process. This trend is universal for different real-world datasets. For larger datasets with fewer classes, we can see that the increase slows down in the later stage of AL. However, from the accuracy results, we know that the model can still improve by fine-tuning the decision boundaries. Given the steady increase of the mean dual variable value, we use a fixed parameter η in our transition criterion: $\frac{\sum_n \sum_m a_{nm}}{N_{SV}} > \eta \frac{\sum_n \sum_m a^{(0)}_{nm}}{N^{(0)}_{SV}}$. For datasets with fewer classes, η can be assigned a larger value because of the more rapid increase in the beginning. For datasets with more classes, η should be smaller. In practice, η can be determined based on the number of classes and the C value. In Figure 6 (g) and (h), we show the performance using different η values on two more datasets, which exhibits some difference in the middle stage of AL but remains stable overall.

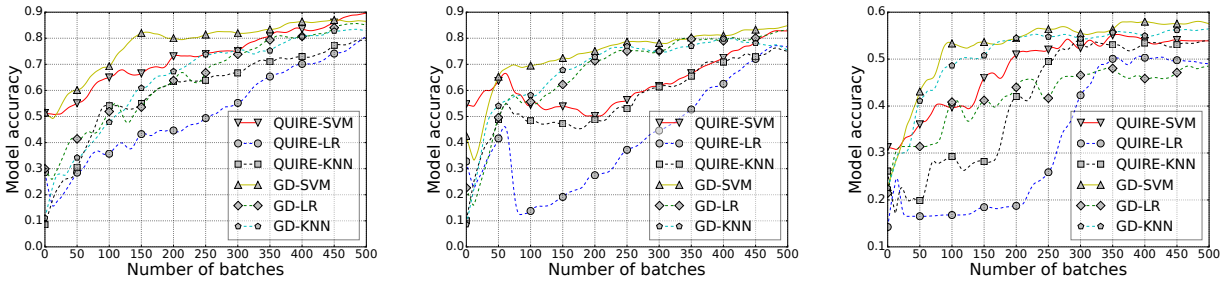
Additionally, in Figure 9, we show the AL results using different classifiers with the baseline sampling approaches. By showing this comparison, we justify our choice of SVM as the core model in the first place. As Figure 9 shows, for the same sampling strategy (QUIRE or Graph-Density), the overall performance using the linear regression (LR) or the k-nearest-neighbors (KNN) models are not so good as the SVM version.

Table 5: SV-recovery results

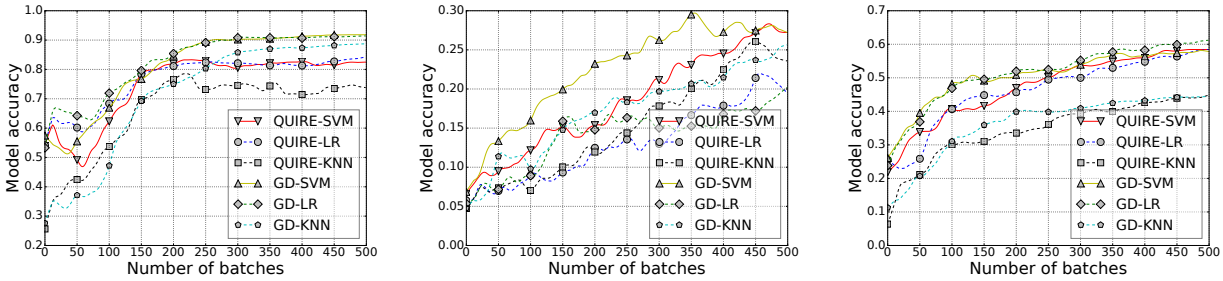
Dataset	Final N_L	$N_L + N_U$	$N_L/(N_L + N_U)$	True SV$/(N_L + N_U)$	Recall	Precision
Yeast	510	751	0.68	0.90	0.71	0.94
Auto-Drive	511	4411	0.12	0.82	0.13	0.92
USPS	510	4010	0.13	0.39	0.31	0.96

D.5. SV-recovery Experiments

In this subsection we show an interesting study on the effectiveness of SV-recovery rates of the proposed method. The purpose of the study is to verify how often the proposed method can discover potential SVs. The results are presented in a "precision-recall"-style analysis: in Table 5, we show all the SV-related information, where $N_L + N_U$ is the total number of samples, from which D-TRUST selects 500, resulting in N_L of labeled samples in the end (500 plus initial labeled pool). Then, we train a passive model using all $N_L + N_U$ samples, and the "true SVs" number is how many of the $N_L + N_U$ samples become SVs after the passive model is fully trained. The recall shows how many of the "true SVs" D-TRUST has recovered using the 500 labeling budget, and the precision shows how many of the N_L samples become SVs. From the table, we see that D-TRUST has a very high precision compared to the true SV ratio, and the recall is also better than $N_L/(N_L + N_U)$, which is the selection rate.



(a) Dermatology dataset 1 (b) Dermatology dataset 2 (c) Yeast dataset



(d) USPS dataset (e) Penstroke dataset (f) Auto-drive dataset

Figure 8: QUIRE and Graph Density sampling based on different classifiers, LR - logistic regression, KNN - k nearest neighbors

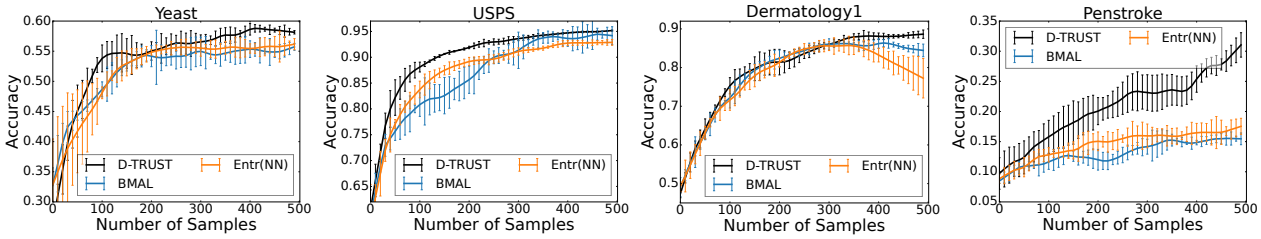


Figure 9: Batch-mode Simple Adaptation Comparison

D.6. Batch-mode Comparison

The proposed method considers the low-budget regime where single-batch is feasible. However, the exploration-exploitation mechanism can be further integrated with diversity-based or other types of batch-mode extensions. However, here we show a simple comparison in Figure 9 with a batch size of 10, which shows that D-TRUST already performs well compared to existing baselines (Yin et al., 2017; Shannon, 1948).

D.7. Complexity Analysis

Since SVM based models share the same training time, the complexity only differs at the sampling step. While the first-order uncertainty based sampling methods have the complexity of $O(NM)$, where N and M denote the numbers of samples and classes, respectively, our method has a sampling cost of $O(NM^2)$ because of the dissonance uncertainty evaluation. Table 6 below summarizes the averaged sampling times.

E. Discussion of Potential Societal Impacts and Limitations

Potential Societal Impacts. When applying the proposed model, the major impact to consider is regarding the selection of data samples. Since the model will be trained using very limited actively selected data samples, it is important to ensure the fairness of the sampling process to avoid potential bias in the trained model, especially when using the model to support

Table 6: Sampling time comparison

Dataset	D-TRUST	QUIRE	BvSB	GD	Entr	MCCH
Dermatology 1	14.18	8.52	8.97	0.02	9.33	126.76
Dermatology 2	13.07	7.29	6.83	0.03	7.04	156.84
Yeast	0.13	11.93	0.06	0.03	0.06	317.86
USPS	0.89	8.32	0.88	0.03	0.66	643.65
Auto-drive	1.63	20.59	0.83	0.04	0.50	628.57
Penstroke	2.46	13.73	1.91	0.03	1.49	184.76

critical decision-making.

Limitations and potential extensions. Since the theoretical results are derived upon the important properties of the dual space, the proposed approach will be used together with the sparse kernel max-margin models. Meanwhile, we clearly justify the advantage of using a sparse maximum-margin model in the "small data" regime for active learning, where many other models (*e.g.,* most deep learning models) become largely ineffective. However, we also recognize that potential extension of our work to DNNs is feasible by leveraging the recent efforts in bridging SVMs and DNNs (*e.g.,* (Tang, 2013)). Such a framework uses the second to last layer output $\mathbf{h}(\mathbf{x})$ as the SVM input and a L2-hinge loss. The solution is in the form of a weight vector, which can be used to solve for dual variables. Then, our sampling strategies can be straightforwardly applied. Intuitively the dual variables can be solved through a system of linear equations. More efficiently solving the dual variables can also be an interesting future direction. Since most deep learning models require a careful fine-tuning, we will leave this extension as a future direction as well.

F. Source Code

The data and source code for replicating the results are provided in this link: https://github.com/ritmininglab/D-TRUST.git