An Expanded Benchmark that Rediscovers and Affirms the Edge of Uncertainty Sampling for Active Learning

Anonymous authors
Paper under double-blind review

Abstract

Active Learning (AL) addresses the crucial challenge of enabling machines to efficiently gather labeled examples through strategic queries. Among the many AL strategies, Uncertainty Sampling (US) stands out as one of the most widely adopted. US queries the example(s) that the current model finds uncertain, proving to be both straightforward and effective. Despite claims in the literature suggesting superior alternatives to US, community-wide acceptance remains elusive. In fact, existing benchmarks present conflicting conclusions on the continued competitiveness of US. In this study, we review the literature on AL strategies in the last decade and build the most comprehensive open-source AL benchmark to date to understand the relative merits of different AL strategies. The benchmark surpasses existing ones by encompassing a broader coverage of strategies, models, and data. Through our extensive evaluation, we uncover fresh insights into the often-overlooked issue of model compatibility in the context of US to clarify the conflicting conclusions in existing benchmarks. Notably, our findings affirm that when paired with compatible models, US maintains a competitive edge over other strategies. These findings have practical implications and provide a concrete recipe for AL practitioners—by adopting compatible query-oriented and task-oriented models for US as the first-hand choice, empowering them to make informed decisions in their work.

1 Introduction

Supervised learning models can achieve competitive results with sufficient high-quality labeled data. However, acquiring such data can be costly in specific domains. This situation calls for Active Learning (AL), a learning paradigm that strategically selects the most valuable unlabeled examples for labeling. AL has the capability of achieving better performance with lower labeling costs, which has been widely studied and applied in various domains, such as computer vision Li & Guo (2013); Demir et al. (2015); Beluch et al. (2018), natural language processing Liu et al. (2021); Schröder et al. (2021); Kishaan et al. (2020), and biology and medical fields Hao et al. (2020); Nath et al. (2020); Logan et al. (2022).

Among the many AL strategies, Uncertainty Sampling (US) stands out as a straightforward and efficient query strategy by selecting the most uncertain examples for labeling based on the model's prediction confidence. US has demonstrated success across multiple applications Kishaan et al. (2020); Narayanan et al. (2020); Nath et al. (2020); while US is widely used, several AL studies have developed more sophisticated query strategies to address specific limitations in particular scenarios Donmez et al. (2007); Huang et al. (2010); Li et al. (2015). However, these strategies often lack a fair and unified comparison across different contexts.

Two large-scale benchmarks for pool-based AL have been developed to evaluate existing strategies for binary classification on tabular data Yang & Loog (2018); Zhan et al. (2021). However, they present conflicting conclusions regarding the preferred query strategies. While Yang & Loog (2018) suggested that the straightforward US strategy excels across the majority of datasets, Zhan et al. (2021) argued that Learning Active Learning (LAL) Konyushkova et al. (2017) outperforms US.

Table 1: Comparison between Yang & Loog (2018); Zhan et al. (2021) and our benchmark. (D) means aspects of datasets; (M) means aspects of base models; (Q) means aspects of query strategies; (A) means aspects of analysis; (O) means aspects of an open source tool. Our benchmark fetches up lacking query strategies in Yang & Loog (2018) and lacking analysis in Zhan et al. (2021) to provide a comprehensive comparison.

		Yang & Loog (2018)	Zhan et al. (2021)	Ours
(D)	More than 100K examples	✓		✓
(D)	More than 400 features	✓		✓
(M)	LR	✓		✓
(M)	RBFSVM		✓	✓
(M)	RF			✓
(Q)	Model uncertainty	✓	✓	✓
(Q)	Bayesian uncertainty			✓
(Q)	Data diversity		✓	✓
(Q)	Hybrid criteria	✓	✓	✓
(Q)	Redesigned learning framework		✓	✓
(A)	AUBC	✓	✓	✓
(A)	Average ranking	✓		✓
(A)	Comparison with Uniform	✓		✓
(O)	Released datasets		✓	✓
(O)	Unified AL protocol			✓
(O)	Analysis tools			✓

Given the lack of consistent comparisons across diverse contexts and the contradictory conclusions drawn from the previous two extensive benchmarks, there is a critical need for a benchmark that accurately represents the current state of AL techniques in this field. Therefore, this work aims to build the most comprehensive AL benchmark compared to previous benchmarks, focusing on datasets, base models, query strategies, and analysis aspects, as highlighted in Table 1. Our benchmark could be the most comprehensive open-source framework to date, crafted by integrating a transparent and unified interface. This unified interface cooperates with existing GitHub repositories, such as libact Yang et al. (2017), Google AL playground Yilei "Dolee" Yang (2017), ALiPy Tang et al. (2019), ModAL Danka & Horvath, scikit-activeml Kottke et al. (2021), and sets a new standard for future research.

Subsequently, we assess the performance of query strategies specifically for binary classifications on tabular data, which is widely used in various real-world applications due to its structured nature and the availability of diverse datasets. Our benchmarking results show that US remains competitive on most datasets. Furthermore, we uncover the reason for the substandard performance of US in Zhan et al. (2021), the incompatibility between a model used within US querying and a model being evaluated for the tasks degrades the performance. Through careful study, we affirm that US maintains a competitive edge over other strategies when used with compatible settings on Logistic Regression (LR), Radial Basis Function kernel Support Vector Machine (RBFSVM), Random Forest (RF), and Gradient Boosting Decision Tree (GBDT). In summary, we recommend adopting US with compatible settings as a first-hand choice for practitioners, providing a clear baseline for AL in real-world usage from the community.

In this work, we make the following contributions:

- To our knowledge, our benchmark is the most comprehensive, surpassing existing benchmarks in terms of datasets, models, query strategies, and analyses.

- We re-benchmark existing strategies for tabular datasets, demonstrating the US's competitiveness on most datasets, and, importantly, uncover profound insights into the often-overlooked issue of model compatibility in the context of US.
- We offer a reproducible and open-source benchmarking framework, which includes preparing datasets, an active learning process, and analysis tools to facilitate future research in the community.

2 Preliminary

Settles's literature survey initiated the study of pool-based active learning (AL) techniques. In this section, we extend Settles (2012) to the current state of pool-based AL research, addressing the gap created by the lack of an open-source benchmark and highlighting significant developments in query strategies over the last decade. We also introduce the experimental protocol of our benchmark, which facilitates a deeper understanding of the critical components involved in pool-based AL, helping readers to comprehensively evaluate the efficacy of different query strategies in this domain.

2.1 Literature survey of pool-based active learning

Settles (2012) formalized the pool-based active learning protocol as follows:

Initial setup The process begins with a small labeled pool $D_\mathrm{l} = \{(x_1, y_1), \ldots, (x_N, y_N)\}$, where $|D_\mathrm{l}| = N$ is the number of labeled examples, and a large unlabeled pool $D_\mathrm{u} = \{x_{N+1}, \ldots, x_{N+M}\}$, where $|D_\mathrm{u}| = M$ is the number of unlabeled examples; and an oracle O that provides ground truth labels.

Execution setup The active learning algorithm operates over T rounds within a total query budget, where each round involves querying the label of one unlabeled example from D_u until the budget is exhausted.

Query steps in each round

1. **Query**: Employ the query strategy $\mathcal{Q}$ to select an example x_j from D_u.
2. **Label**: Acquire the label y_j for x_j from an oracle $O(x_j) = y_j$.
3. **Update pools**: Move the new labeled example from D_u to D_l, i.e., $D_\mathrm{l} \leftarrow D_\mathrm{l} \cup \{(x_j, y_j)\}$, $D_\mathrm{u} \leftarrow D_\mathrm{u} \setminus \{(x_j)\}$.
4. **Update the model**: Retrain the model using the updated labeled pool D_l.

Prediction on the test set Finally, we train the model $\mathcal{G}$ on the latest labeled pool D_l and make predictions on new examples from the unseen testing set D_te.

The critical element in pool-based active learning is the query strategy $\mathcal{Q}$. A naïve uniform sampling (Uniform) method randomly selects unlabeled examples for labeling. Uniform does not utilize active learning strategies and serves primarily as a baseline. The overarching goal of active learning is to develop a query strategy that outperforms the Uniform baseline, and there are already numerous query strategies available today. Settles (2012) classifies these query strategies into three categories: **model uncertainty**, **expected model changing**, and **representation exploiting**.

Model uncertainty Uncertainty Sampling (US) is a prevalent query strategy in pool-based active learning, where it selects examples for labeling based on the degree of uncertainty regarding the model's prediction. US assumes that examples about which the model is most uncertain are likely to yield the highest information gain upon being labeled. Various measures can be employed to quantify uncertainty, including the margin score and entropy of the predictions of an examples in the unlabeled pool returned by the current model. In binary classification scenarios, using margin and entropy scores are equivalent in terms of defining model uncertainty (see Appendix B.5). Previous works have found that US is a strong baseline for most pool-based active learning problems Cawley (2011); Yang & Loog (2018); Karamcheti et al. (2021); Schröder et al. (2021); Bahri et al. (2022).

In contrast to US, which relies on a single model to quantify uncertainty, Query By Committee (QBC) Seung et al. (1992) quantifies uncertainty through multiple models to address the sampling bias in US Settles (2012). QBC operates on the principle of disagreement among a committee of models, each representing a different model derived from the training set. Specifically, QBC selects the unlabeled example where there is the maximal disagreement among the committee members. Disagreement is measured by voting entropy, defined as the entropy of the distribution of the committee's votes. A higher voting entropy of an example indicates more significant disagreement and, consequently, a higher value for querying.

Expected model changing Previous query strategies aim to query the most informative example for the current model. In this category, we strive to query the most informative example to reduce the model's error in the future. For instance, Expected Error Reduction (EER) estimates the total future output uncertainty over an unlabeled pool and queries the most uncertain example. Similarly, Variance Reduction (VR) estimates the variance of the model's output based on its Fisher information, which estimates the inverse of the lower bound on the variance of the model's parameters Cover (1999).

Representation exploiting US and QBC might perform poorly due to outliers or sampling bias that results in querying a non-representative example during the query process Dasgupta & Hsu (2008); Yang et al. (2015); Shui et al. (2020). Although EER and VR take the input distribution into account via estimating expected future error over all unlabeled examples, these methods are computationally expensive, making them unsuitable for large datasets Settles (2012). Hierarchical Sampling (Hier) is a model-free representation sampling method that exploits hierarchical clustering to explore the data structure of the unlabeled pool Dasgupta & Hsu (2008). Hier randomly selects an example from the subtree of the hierarchical clustering tree to obtain its label. Then, the tree structure is iteratively updated by making the labels in the cluster more pure and focusing on the remaining impure clusters. In contrast with model-free approaches, Density-Weight Uncertainty Sampling (DWUS) Nguyen & Smeulders (2004) assumes that informative examples should have both high uncertainty and be representative of the data distribution. Therefore, DWUS designs a weighted uncertainty score by averaging an example's similarity to the remaining examples in the training set.

The query strategies mentioned in Settles (2012) are long-standing. However, the survey should be updated with the latest approaches. Graph Density (Graph) is also a model-free representation sampling method that exploits cluster structure by applying graph-based clustering techniques to the unlabeled pool without depending on any model. Similar to Graph, Core-Set uses K-Means clustering on the embedding space extracted from the data transformation (such as deep convolutional neural networks) and then queries unlabeled examples closest to the centers of clusters. Sener & Savarese (2018) show that Core-Set works well on image classification tasks. Besides Graph and Core-Set, we could categorize recent query strategies into three categories: **hybrid criteria**, **Bayesian method**, and **redesigned learning framework**.

Hybrid criteria Several works study the combination of uncertainty and diversity information to improve previous query strategies. For example, Hinted Support Vector Machine (HintSVM) Li et al. (2015) focused on selecting an example of an updated decision boundary that passes through unqueried regions instead of reducing its margin only. QUerying Informative and Representative Examples (QUIRE) Huang et al. (2010) formulated the informativeness and representativeness with kernel matrices, which characterizes the similarity between labeled examples and unlabeled examples, to select an example with large self-similarity and large similarity to most remaining examples in the unlabeled pool. Representative Marginal Cluster Mean Sampling (MCM) Xu et al. (2003) queries examples within the model's margin closest to the K-Means centers in the embedding space, which inherits the benefits from Core-Set and US. Recently, Batch Mode Discriminative and Representative (BMDR) and Self-Paced Active Learning (SPAL) have been designed to query examples with elaborated empirical risk minimization Wang & Ye (2015); Tang & Huang (2019). BMDR queries the example that expects to minimize the empirical risk on the labeled and unlabeled pools using a self-learning approach and distribution difference between the labeled pool and training set. Following the objective function of BMDR, SPAL modifies the constraint of the objective function (1) to improve BMDR's performance. Please refer to Appendix B.5 for the detailed formulation of BMDR and SPAL.

Bayesian method Although QBC aims to query the most disagreeable example, the voter entropy might ignore each model's confidence regarding its predictions, potentially reducing efficiency. To address this issue, Bayesian Active Learning by Disagreement (BALD) Houlsby et al. (2011) queries the most uncertain example across the ensemble models but confident in the single model. This approach can be interpreted as the conditional mutual information between the model's prediction and its parameters. BALD aims to query the example with high conditional mutual information, where the model's prediction is uncertain, but the model's parameters are certain.

Redesigned learning framework As the number of query strategies increases, some are designed to automatically select the optimal strategy from multiple heuristic query strategies. For example, Active Learning By Learning (ALBL) treats the learning problem as a multi-armed bandit problem Hsu & Lin (2015). It thus selects the optimal strategy from a set of query strategies and queries the example based on this strategy that maximizes the estimated reward at each round. Learning Active Learning (LAL) formulates the query process as a regression problem to learn the strategy from various types of toy data Konyushkova et al. (2017). LAL queries the example from the learned regression function, which predicts the potential error reduction.

Besides previous query strategies for conventional machine learning models, such as Logistic Regression (LR), and Radial Basis Function kernel Support Vector Machine (RBFSVM), Beck et al. (2021) and Zhan et al. (2022) compared additional query strategies designed for deep learning models used in computer vision classification tasks. Their results show that US outperforms data diversity-based sampling strategies (Core-Set, Variational Adversarial Active Learning) Sinha et al. (2019). Moreover, hybrid criteria query strategies, such as Batch Active learning by Diverse Gradient Embeddings (BADGE) Ash et al. (2019), Learning Loss for Active Learning (LPL) Yoo & Kweon (2019), and Wasserstein Adversarial Active Learning (WAAL) Shui et al. (2020), achieve competitive results better than US. Although modern techniques such as BADGE, LPL, and WAAL demonstrate outstanding performance for deep active learning scenarios, they cannot be directly applied to the current implementation of tree-based models such as Random Forest (RF) Pedregosa et al. (2011), XGBoost Chen & Guestrin (2016). Therefore, our work excludes these query strategies and encourages future work to extend the benchmark to deep learning models.

2.2 Experimental protocol for the benchmark

Section 2.1 depicts an abstract process of pool-based active learning. To concretize the experimental protocol for the benchmark, we illustrate the framework in Figure 1. In this framework, we define the training set as the union of the labeled pool and unlabeled pool, denoted as $D_{\text{tr}} = D_{\text{l}} \cup D_{\text{u}}$. First, we split the dataset into disjoint training and testing sets, i.e., $D_{\text{tr}} \cap D_{\text{te}} = \emptyset$, to simulate a real-world learning scenario. After splitting the dataset, we sample from the labeled pool D_{l} within D_{tr} and leave the remaining examples as the unlabeled pool D_{u} to set up the initial environment. Furthermore, we isolate a query-oriented model $\mathcal{H}$ from the task-oriented model $\mathcal{G}$ in Section 2.1. The query-oriented model is used for selecting the most informative example during the query step while the task-oriented model is used for prediction on the test set, as depicted in Figure 1.

In most cases, the same models are used for both the query and the task. However, several query strategies are model-free or do not require the use of the same models. For instance, advanced query strategies by Yoo & Kweon (2019); Sinha et al. (2019) have been shown to be beneficial when using different models for querying informative examples and for training a classifier. To distinguish the relationship between models, we define **model compatibility** as the setting where the example obtained by the query-oriented model might or might not be the same when using the task-oriented model. In this work, we denote the compatible query-oriented and task-oriented models for Uncertainty Sampling as US-Compatible (US-C) and non-compatible models as US-Non-Compatible (US-NC). Section 5.1 studies the impact of model compatibility on US to clarify the conflicting conclusion in previous benchmarks.

The benchmark aims to provide a standardized framework for evaluating and comparing different query strategies in a fair manner. Following Guyon et al. (2010; 2011); Desreumaux & Lemaire (2020); Zhan et al. (2021), we utilize the Area Under the Budget Curve (AUBC) as a summary metric to quantify the results of learning curves. A learning curve tracks the performance of model $\mathcal{G}$ at each round of the active learning

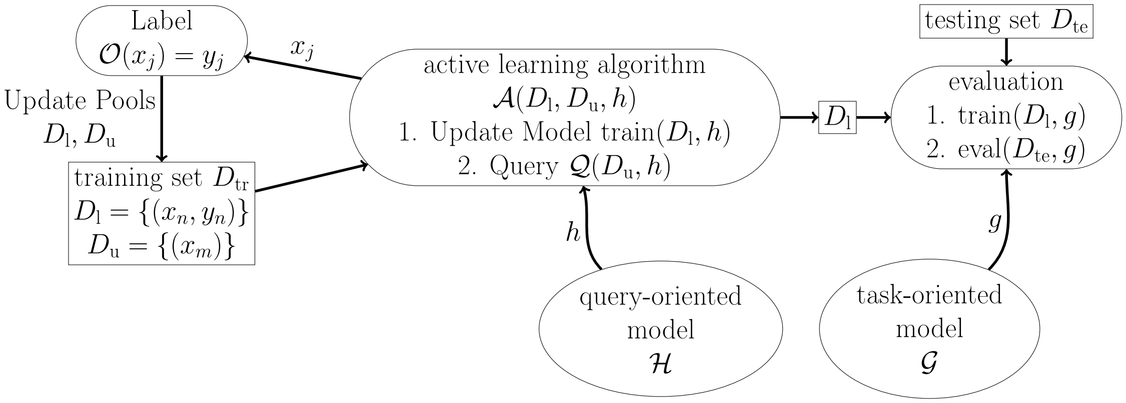

Label
$\mathcal{O}(x_j) = y_j$
x_j
Update Pools
$D_{\mathrm{l}}, D_{\mathrm{u}}$
training set D_{tr}
$D_{\mathrm{l}} = \{(x_n, y_n)\}$
$D_{\mathrm{u}} = \{(x_m)\}$
active learning algorithm
$\mathcal{A}(D_{\mathrm{l}}, D_{\mathrm{u}}, h)$
1. Update Model train(D_{l}, h)
2. Query $\mathcal{Q}(D_{\mathrm{u}}, h)$
D_{l}
testing set D_{te}
evaluation
1. train(D_{l}, g)
2. eval(D_{te}, g)
h
g
query-oriented
model
$\mathcal{H}$
task-oriented
model
$\mathcal{G}$

Figure 1: The Framework of Active Learning Experiments. Rectangles represent datasets including labeled pool, unlabeled pool, and test set. Rounded rectangles represent processes including an active learning algorithm, labeling, and evaluation. Circles represent models. In this work, we differentiate the relationship between two models: task-oriented and query-oriented.

process, typically using evaluation metrics such as accuracy. AUBC provides a concise way to compare the overall performance of different learning curves of query strategies. Figure 2 demonstrates that US, BALD, and LAL achieve higher accuracy more quickly than Uniform, corresponding to the mean AUBC of US (85.78%) and BALD (85.72%), which are better than LAL (85.52%), Uniform (84.77%), and Core-Set (84.47%) in detail. Furthermore, we report the accuracy of the task-oriented model under different labeled data sizes and data utilization rates of the query strategy for more detail.

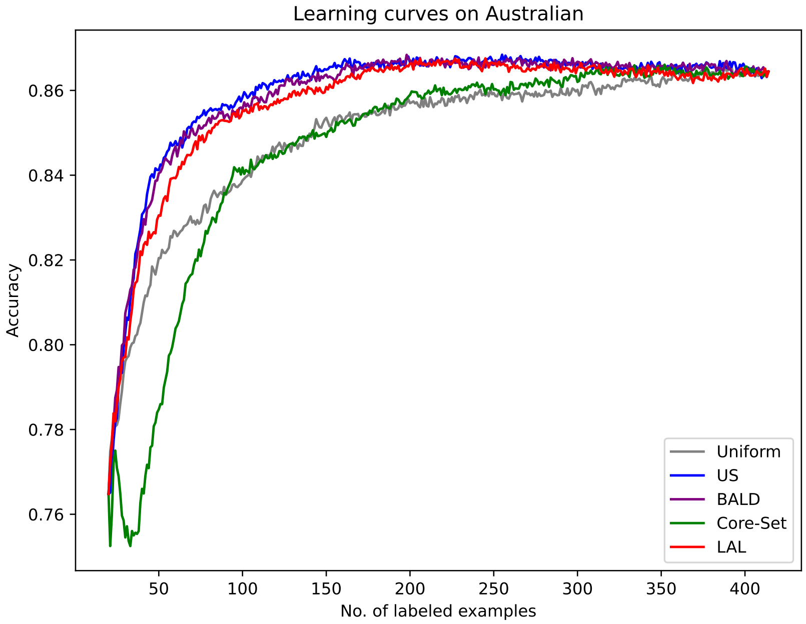


Figure 2: The learning curves (test accuracy vs. number of labeled examples) of query strategies on *Australian* dataset.

Table 2: Settings of query-oriented models $\mathcal{H}$ for specific query strategies $\mathcal{Q}$.

$\mathcal{Q}$	$\mathcal{H}$	Reason of choice
HintSVM	RBFSVM	the implementation in libact
QUIRE	RBFSVM	the implementation in libact
QBC	LR($C = 0.1$), RBFSVM, RF, Linear Discriminant Analysis	the inheritance of Zhan et al. (2021)
ALBL	Combination of multiple $\mathcal{Q}$ with same $\mathcal{H}$: US, HintSVM	the default settings in libact
LAL	RF	the implementation in ALiPy

3 Experimental settings

We employ most of the settings outlined in the prior benchmark Zhan et al. (2021). For each dataset D, we reserve 40% as the unseen test set D_{te} for performance evaluation. Then, for the remaining 60%, we randomly sample 20 examples as the initial labeled pool D_{l} and leave the others as the unlabeled pool D_{u}.

$$\begin{aligned} D &= D_{\text{tr}} \cup D_{\text{te}}, \quad |D_{\text{te}}| = 0.4|D|, \\ D_{\text{tr}} &= D_{\text{l}} \cup D_{\text{u}}, \quad |D_{\text{l}}| = 20. \end{aligned}$$

In the following, we clarify the differences and expansions in our benchmark compared to the previous benchmarks Yang & Loog (2018); Zhan et al. (2021).

Focus on fundamental binary classification. The previous benchmark simultaneously evaluated query strategies that either support or do not support multi-class may have affected the validity of claims when comparing results across different aspects Zhan et al. (2021). Therefore, we restrict our evaluation to binary-class datasets to ensure consistency and fairness.

Include comprehensive datasets with a unified format. We select 26 binary datasets from Zhan et al. (2021) and Yang & Loog (2018) and ensure consistency in the source datasets and the composition of the initial labeled and unlabeled pools. For instance, we scaled raw data features to $[-1, 1]$ for all datasets.[1] We added three datasets from the other tabular data benchmark Grinsztajn et al. (2022) to expand the benchmark coverage. These datasets were selected based on their large-scale and high-dimensional properties to better reflect real-world scenarios. Please refer to Table 10 for the properties of 29 datasets.

Include broad types of query strategies. Zhan et al. (2021) extended the query strategies from Yang & Loog (2018) to 17 query strategies. However, the redundancy of query strategies, such as US and Informative Cluster Diverse (InfoDiv) (See Appendix B.5 for more detail.), may lead to repetitive and limited insights into the benchmark. Therefore, we only keep the most representative 12 query strategies: US, QBC, Hier, Graph, Core-Set, HintSVM, QUIRE, DWUS, MCM, BMDR, ALBL, and LAL. We further expand the benchmark to explore a broader range of query strategies by including BALD, a popular query strategy in deep learning Gal et al. (2017).

Adopt a tree-based model. Previous benchmarks studied Logistic Regression and RBFSVM.[2] In this work, we further studied tree-based models such as XGBoost Chen & Guestrin (2016) and Random Forest Breiman (2001) (See Appendix D), as recommended by the earlier benchmark for tabular datasets Grinsztajn et al. (2022). To clarify the relationship between query-oriented and task-oriented models, we report some query strategies that do not use tree-based model as the query-oriented model in Table 2.

We disclose the construction of the initial labeled pool, data preprocessing steps, and the choice of models, which can significantly impact the experimental results Ji et al. (2023). This information saves participants

[1] We retained the original scaling for some of the LIBSVM datasets, such as *Heart*, *Ionosphere*, and *Sonar*, which were already scaled to the range of $[-1, 1]$.

[2] We also reproduce the previous benchmarks with different base models in Appendix C

time examining the settings and critical considerations for designing active learning experiments. Next, we report the issues encountered and solutions when we conduct experiments.

Handle errors and exceptions of experiments. Because current modules cannot support *cold-start* problems, we run the experiments repeatedly and skip any seed that lacks labels in the training or test set at the initial setup. For execution, we set a maximum running time of 72 hours for executing a query strategy on a dataset to ensure completion within a reasonable time (Denote 'TLE' in Table 5).

This section outlines the necessary information to conduct experiments for the benchmark. In our implementation and report, we strive to ensure the reproducibility of all results under these specific settings and processes corresponding to Figure 1. Furthermore, we compare our settings and results with the existing benchmark Zhan et al. (2021) in Appendix B and Appendix C, covering any additional modifications or improvements needed.

4 Benchmarking results

This section presents the benchmarking results for XGBoost in Table 3. We repeated experiments 100 times for small datasets with a size less than 2000 ($K_{\mathrm{S}} = 100$) and 10 times for large datasets ($K_{\mathrm{L}} = 10$). We set a total query budget of 3000 to reduce running time for large datasets. Next, we verify the superiority of Uncertainty Sampling over other query strategies. Furthermore, we investigate whether existing query strategies bring more benefits than Uniform for each dataset. In addition, we reproduce the benchmarking results from Zhan et al. (2021) with RBFSVM in Appendix C and constructed the new benchmark for RF in Appendix D.

4.1 Verify superiority

Referring to Table 3, we observe that US attains the highest mean AUBC among all query strategies on 18 datasets, indicating its superior performance compared to other query strategies on average. The remaining dominant query strategies are LAL and BALD, which achieve the highest AUBC on 6 and 4 datasets, respectively.

Besides AUBC, we also observe learning curves from different perspectives. Specifically, we check the model's accuracy with varying ratios of labeled examples on each dataset. Table 4 shows the model's accuracy with 20% labeled examples on each dataset, and US outperforms other query strategies on more than half (15) datasets. Please refer to Appendix A for more comparisons under other ratios.

Finally, we verify the ranking performance of query strategies across multiple datasets. Specifically, we assess the average and standard deviation of the rankings by seeds of the query strategy on each dataset. Then, we apply the Friedman test with a 5% significance level to test for statistical significance. The p-values of the Friedman test are less than 5% for all datasets, indicating that the performance differences between query strategies are statistically exist. Table 5 demonstrates that US ranks first on 18 datasets, and LAL, BALD, and MCM often achieve second and third ranks.

These results show that the straightforward and efficient US outperforms others on most datasets. These outcomes also correspond to previous work claiming US is the strong baseline with LR Yang & Loog (2018) and RBFSVM, which we re-benchmarked in Appendix C. We recommend that practitioners initiate their pool-based active learning projects with US.

4.2 Verify usefulness

We investigate the *usefulness* of query strategies in Section 4.2. The analysis of *usefulness* can uncover which query strategy brings more benefits than Uniform, offering practitioners a reality check on the effectiveness of a query strategy. Specifically, we investigate the improvement of the optimal stopping point of query strategies over Uniform. The optimal stopping point is the point where the model achieves the target accuracy with the least number of labeled examples. We refer to the *data utilization rate* Culver et al. (2006), which is the number of labeled examples to achieve the target accuracy divided by the number of

Table 3: Benchmarking results of XGBoost. The numbers are mean AUBC (↑ is better). We report the baseline method (Uniform), the best query strategy with its mean AUBC (BEST_QS, BEST), and the worst query strategy with its mean AUBC (WORST_QS, WORST) across datasets in Table 3.

	Uniform	BEST_QS	BEST	WORST_QS	WORST
Appendicitis	81.51%	US	82.85%	DWUS	80.74%
Sonar	75.10%	US	76.06%	Core-Set	74.11%
Parkinsons	84.24%	US	86.63%	HintSVM	83.12%
Ex8b	84.21%	LAL	85.16%	DWUS	83.05%
Heart	78.37%	BALD	79.35%	HintSVM	77.83%
Haberman	67.69%	US	69.17%	HintSVM	66.82%
Ionosphere	87.96%	US	89.95%	DWUS	81.85%
Clean1	76.54%	US	78.99%	Graph	76.30%
Breast	95.57%	LAL	96.31%	DWUS	91.85%
Wdbc	94.07%	LAL	95.24%	HintSVM	93.96%
Australian	84.77%	US	85.78%	HintSVM	83.89%
Diabetes	72.62%	US	73.62%	HintSVM	71.49%
Mammographic	79.46%	BALD	80.78%	DWUS	78.80%
Ex8a	92.06%	Core-Set	94.07%	HintSVM	84.75%
Tic	90.11%	US	90.65%	DWUS	89.11%
German	72.68%	US	74.03%	DWUS	71.78%
Splice	91.89%	US	93.76%	DWUS	89.51%
Gcloudb	87.85%	LAL	88.68%	QUIRE	85.99%
Gcloudub	92.98%	US	94.30%	DWUS	86.12%
Checkerboard	98.72%	LAL	99.49%	DWUS	86.83%
Spambase	93.16%	US	94.51%	HintSVM	91.09%
Banana	87.70%	LAL	88.45%	HintSVM	79.70%
Phoneme	85.78%	US	87.77%	DWUS	82.42%
Ringnorm	93.76%	US	95.46%	Core-Set	64.58%
Twonorm	95.43%	US	96.39%	HintSVM	83.38%
Phishing	94.20%	US	96.24%	DWUS	91.68%
Covertype	74.11%	US	76.64%	DWUS	61.34%
Bioresponse	72.92%	BALD	74.50%	Core-Set	72.04%
Pol	96.03%	BALD	97.62%	HintSVM	90.52%

labeled examples required by Uniform. In this benchmark, we set the target accuracy as the accuracy with the total query budget minus 0.01. Table 6 shows the data utilization rate of the optimal stopping point of query strategies over Uniform. We observe that US, BALD, MCM, and LAL achieve a higher data utilization rate than Uniform on most datasets.

To further investigate the usefulness of US, we check the improved accuracy (τ) of US, BALD, Core-Set, and LAL over Uniform on effective dataset (*Covertype*) and ineffective dataset (*Checkerboard*) on average with different scales of the total budget. Figure 3 shows that the performance of US and BALD gains significant benefits on large scale dataset. However, US suffers from the sampling bias on *Checkerboard* with a small budget, while BALD is more stable. We notice that a query strategy with a good performance brings more benefits at the early stage of the learning process.

5 Beyond the benchmarking results

In this section, we aim to clarify the conflicting conclusions between our benchmark and the previous work Zhan et al. (2021) and extend the benchmark to real-world datasets used in another tabular data

Table 4: Accuracy (↑ is better) of the model with 20% labeled examples: We report the accuracy of the model with 20% labeled examples on each dataset. The scores with **bold** indicate the best performance on a dataset. 'TLE' means a query strategy exceeds the time limit.

	Uniform	US	QBC	BALD	Hier	Graph	Core-Set	HintSVM	QUIRE	DWUS	MCM	BMDR	ALBL	LAL
Sonar	67.82%	67.29%	66.95%	66.73%	66.45%	67.83%	66.25%	65.88%	67.99%	**68.15%**	66.82%	TLE	67.25%	67.32%
Parkinsons	79.26%	**80.71%**	79.36%	80.51%	79.40%	78.90%	78.68%	78.41%	78.60%	79.69%	80.10%	78.81%	79.91%	80.27%
Ex8b	80.55%	80.69%	79.98%	79.37%	80.24%	79.13%	81.11%	80.32%	81.26%	78.76%	80.45%	**81.26%**	80.74%	80.60%
Heart	75.73%	77.19%	75.06%	**77.36%**	75.59%	76.31%	76.69%	74.82%	76.36%	75.09%	75.41%	75.91%	76.36%	76.39%
Haberman	69.24%	70.61%	69.20%	70.15%	68.54%	68.96%	67.68%	68.43%	67.69%	68.51%	**71.04%**	69.15%	68.67%	70.50%
Ionosphere	83.59%	86.96%	83.84%	86.78%	84.18%	82.28%	83.24%	81.84%	80.82%	74.91%	84.02%	80.28%	86.73%	**87.21%**
Clean1	68.34%	**69.86%**	68.03%	69.10%	68.57%	68.21%	66.74%	67.59%	68.42%	68.34%	67.36%	TLE	69.35%	69.31%
Breast	95.17%	**96.73%**	95.17%	96.72%	95.54%	95.29%	94.54%	94.77%	94.93%	90.07%	96.57%	94.77%	96.16%	96.66%
Wdbc	92.91%	95.34%	92.50%	95.36%	92.91%	92.99%	93.04%	91.94%	92.66%	92.54%	95.26%	92.68%	94.75%	**95.55%**
Australian	83.63%	**85.55%**	83.77%	85.33%	83.87%	83.90%	82.97%	81.95%	82.75%	82.51%	84.86%	83.58%	83.28%	85.13%
Diabetes	71.84%	**73.96%**	72.34%	73.35%	72.07%	72.38%	71.73%	70.14%	71.81%	70.54%	72.85%	72.16%	71.78%	72.80%
Mammographic	79.66%	**82.51%**	79.66%	82.30%	79.48%	79.17%	79.52%	78.70%	80.41%	79.39%	82.09%	80.03%	80.33%	81.67%
Ex8a	88.22%	88.55%	87.81%	88.75%	87.94%	89.41%	91.69%	78.04%	78.12%	78.06%	88.41%	TLE	85.21%	**91.88%**
Tic	89.25%	**90.43%**	89.23%	90.38%	89.30%	89.72%	89.33%	88.12%	89.83%	85.76%	89.58%	TLE	89.32%	89.15%
German	71.23%	**72.90%**	71.37%	72.73%	71.08%	71.48%	70.46%	71.61%	71.06%	68.76%	71.97%	TLE	71.55%	72.11%
Splice	89.07%	**92.95%**	88.88%	92.65%	89.00%	89.25%	84.99%	85.74%	88.99%	84.69%	91.33%	TLE	88.65%	90.25%
Gcloudb	87.82%	**89.55%**	87.98%	89.24%	87.95%	88.19%	88.27%	84.62%	84.61%	85.48%	89.49%	87.96%	88.35%	89.36%
Gcloudub	91.25%	**94.11%**	91.45%	92.40%	91.92%	92.28%	88.58%	83.11%	85.69%	80.24%	91.69%	89.76%	88.91%	93.61%
Checkerboard	98.76%	96.89%	98.80%	99.46%	99.35%	98.95%	98.59%	91.40%	88.72%	79.82%	99.46%	99.09%	98.26%	**99.80%**
Spambase	92.47%	94.84%	92.69%	**94.91%**	92.54%	92.64%	92.06%	88.95%	TLE	92.54%	94.61%	TLE	92.66%	94.68%
Banana	87.46%	87.93%	87.46%	87.65%	87.53%	87.50%	88.02%	71.37%	TLE	74.94%	88.49%	TLE	86.99%	**88.94%**
Phoneme	83.97%	87.13%	83.73%	**87.24%**	84.66%	84.13%	84.70%	80.83%	TLE	78.60%	86.36%	TLE	83.73%	86.52%
Ringnorm	92.79%	95.75%	93.33%	**95.77%**	92.35%	92.40%	51.86%	57.27%	TLE	55.71%	95.33%	TLE	92.45%	92.55%
Twonorm	94.99%	**96.61%**	95.04%	96.53%	95.07%	95.24%	95.73%	79.31%	TLE	94.26%	96.60%	TLE	95.52%	95.90%
Phishing	93.41%	**96.19%**	93.01%	96.04%	92.83%	93.35%	93.18%	91.75%	TLE	88.11%	95.70%	TLE	94.37%	95.94%
Covertype	72.30%	75.26%	72.05%	**75.26%**	TLE	64.54%	TLE	62.90%	TLE	59.97%	TLE	TLE	TLE	73.92%
Bioresponse	69.96%	**73.14%**	71.05%	72.87%	70.48%	71.11%	66.59%	66.70%	TLE	69.96%	71.34%	TLE	TLE	72.07%
Pol	95.29%	**98.19%**	95.52%	98.10%	95.40%	86.38%	93.68%	86.75%	TLE	95.29%	97.58%	TLE	95.18%	97.75%

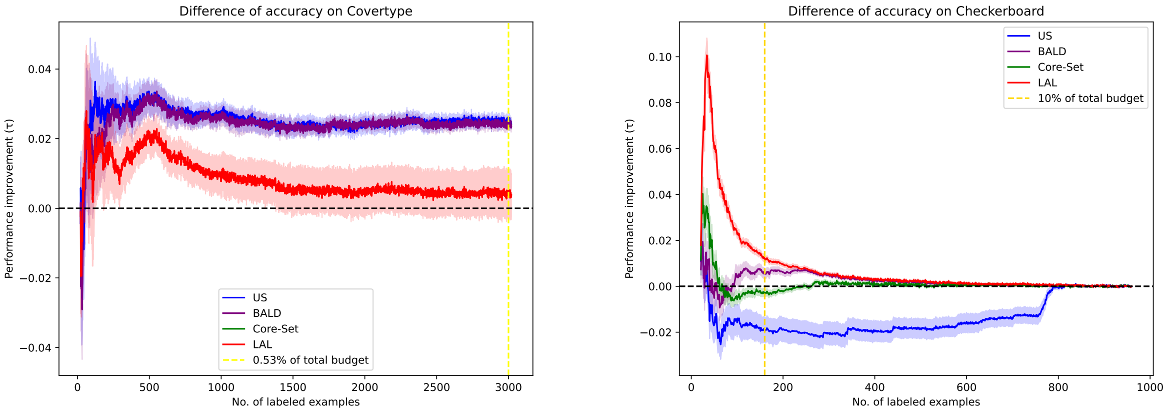


Figure 3: Mean difference of accuracy (improvement) of a query strategy from Uniform on *Covertype* (left) and *Checkerboard* (right). Note that there are no results of Core-Set on *Covertype* due to the time limit (TLE).

benchmark Grinsztajn et al. (2022). First, we study the impact of **model compatibility**. Second, we expand the benchmark by evaluating the usefulness of Uncertainty Sampling on three real-world datasets.

5.1 Impact of non-compatible models for uncertainty sampling

In contrast to the broader performance comparisons in earlier sections, Section 5.1 focuses on the **model compatibility** with US. Our investigation demonstrates that the incompatibility between query-oriented and task-oriented models significantly influences the performance of US. An example of model incompatibility

Table 5: Average Ranking of Query Strategies (↓ is better): We report query strategies with the best average ranking. The scores with ^{1}, ^{2}, or ^{3} mean the 1st, 2nd and 3rd performance on a dataset. 'TLE' means a query strategy exceeds the time limit.

	US	QBC	BALD	Hier	Graph	Core-Set	HintSVM	QUIRE	DWUS	MCM	BMDR	ALBL	LAL
Appendicitis	4.83^1	7.50	5.51^3	7.63	8.30	7.15	7.34	7.82	9.12	5.20^2	7.85	6.92	5.83
Sonar	4.79^1	6.38	4.97^2	6.83	7.52	8.35	8.04	7.79	6.71	6.19	TLE	5.39	5.04^3
Parkinsons	3.39^1	8.41	3.79^2	7.03	9.45	8.37	10.92	8.47	8.74	4.59	7.26	6.34	4.24^3
Ex8b	5.31^2	7.95	6.34	7.60	8.28	5.83	9.09	7.42	10.07	5.08^1	7.13	5.50	5.40^3
Heart	4.89^2	7.72	4.76^1	7.66	8.17	7.18	9.65	8.34	6.43	6.33	7.10	7.04	5.73^3
Haberman	4.38^1	7.69	4.56^2	7.04	8.10	8.97	9.76	9.55	6.07	4.74^3	7.07	7.95	5.12
Ionosphere	2.67^1	7.19	2.89^2	7.12	8.22	8.28	9.38	10.10	12.74	4.51	10.08	4.30	3.52^3
Clean1	2.61^1	8.13	2.96^2	8.33	8.76	7.87	8.47	7.39	8.59	5.30	TLE	5.53	4.06^3
Breast	3.98^3	8.72	3.72^2	7.14	10.02	7.43	8.88	7.50	12.97	4.24	8.64	4.88	2.88^1
Wdbc	3.67^3	9.47	3.44^2	8.15	9.35	8.41	9.57	9.02	9.54	3.95	9.59	3.71	3.13^1
Australian	2.80^1	7.75	3.20^2	7.70	7.47	8.97	10.38	8.36	8.65	4.66	8.30	8.32	4.44^3
Diabetes	3.64^1	7.15	4.35^2	6.88	7.09	7.17	10.44	8.44	10.01	4.98^3	7.20	8.14	5.51
Mammographic	3.46^3	8.69	3.30^1	7.67	8.07	9.09	8.24	9.06	9.54	3.41^2	7.51	9.07	3.89
Ex8a	4.37^3	6.10	4.42	5.90	6.16	1.70^1	11.68	10.75	10.24	4.43	TLE	8.94	3.31^2
Tic	2.65^1	5.66	2.99^2	6.47	6.84	8.49	9.32	8.31	9.24	4.59^3	TLE	7.28	6.16
German	3.10^1	7.29	3.52^2	7.81	7.23	7.41	6.87	8.24	10.92	4.14^3	TLE	6.02	5.45
Splice	1.52^1	7.30	1.86^2	7.18	8.61	9.29	9.82	6.40	11.62	3.37^3	TLE	6.62	4.41
Gcloudb	4.24^3	7.25	4.69	7.51	8.19	5.93	10.71	11.04	11.67	4.02^2	7.07	5.56	3.12^1
Gcloudub	2.52^1	6.41	3.68^3	4.91	7.13	8.75	12.44	10.70	12.45	4.51	7.60	7.05	2.85^2
Checkerboard	6.37	7.15	4.94	5.03	7.44	6.54	11.36	11.48	12.81	3.72^2	4.58^3	8.34	1.24^1
Spambase	1.50^1	7.80	1.70^2	6.80	8.10	8.30	11.00	TLE	7.80	3.30^3	TLE	6.20	3.50
Banana	5.20	5.70	5.70	3.60^3	8.20	3.00^2	10.60	TLE	10.40	5.00	TLE	7.20	1.40^1
Phoneme	1.60^1	8.10	2.00^2	5.20	8.40	6.40	10.00	TLE	10.90	3.30	TLE	7.00	3.10^3
Ringnorm	1.40^1	5.10	1.60^2	6.30	8.00	10.50	9.00	TLE	10.50	3.00^3	TLE	6.30	4.30
Twonorm	1.30^1	7.90	1.70^2	8.50	7.40	6.00	11.00	TLE	10.00	3.00^3	TLE	5.20	4.00
Phishing	1.40^1	7.90	1.60^2	7.20	8.70	6.20	10.10	TLE	10.90	3.60	TLE	5.00	3.40^3
Covertype	1.40^1	3.80	1.90^2	TLE	5.00	TLE	TLE	TLE	6.00	TLE	TLE	TLE	2.90^3
Bioresponse	1.90^2	6.40	1.60^1	6.40	6.20	8.60	TLE	TLE	6.60	3.70	TLE	TLE	3.60^3
Pol	1.80^2	5.80	1.50^1	6.20	9.80	9.20	11.00	TLE	6.70	4.00	TLE	7.30	2.70^3

is that the previous benchmark adopted US with LR($C = 1$) as the query-oriented model and RBFSVM as the task-oriented model Zhan et al. (2021).[3] Through careful analysis, we found that when non-compatible models are used (denoted as US-NC), the performance of US (denoted as US-C) notably drops, as shown in Table 12. This drop is primarily due to the misalignment of the decision boundaries between the query-oriented and task-oriented models, which can lead the query-oriented model to select samples that are not the most uncertain for the task-oriented model, as illustrated in Figure 4. In summary, our benchmarking highlights that by utilizing compatible models, US-C consistently performs better than US-NC on average.

We compare different combinations of query-oriented and task-oriented models based on LR, RBFSVM, and RF. Figure 5 and Appendix C.3 emphasize that compatible model pairs perform better than non-compatible model pairs for US, evident across 22 datasets, where the optimal AUBC score occurs with compatible models, i.e., the highest AUBC score is found along the diagonal. Although some results demonstrate that non-compatible models are slightly better than compatible models, such as *Splice* and *Banana* in Figure 13, these instances were exceptions rather than the norm in our benchmark.

In summary, we advocate for the default use of compatible model parings in US for practical applications. This setting simplifies the model selection process and can potentially yield better performance across various datasets.

[3] See `https://github.com/SineZHAN/ComparativeSurveyIJCAI2021PoolBasedAL/blob/master/Algorithm/baseline-google-binary.py#L242`

Table 6: Data utilization rate (↓ is better): We report the data utilization rate of query strategies. The scores with *italic style* indicate that the query strategy does not provide more benefits than Uniform. 'TLE' means a query strategy exceeds the time limit.

	US	QBC	BALD	Hier	Graph	Core-Set	HintSVM	QUIRE	DWUS	MCM	BMDR	ALBL	LAL
Appendicitis	77.96%	93.97%	71.10%	88.47%	84.87%	77.22%	73.31%	81.74%	*101.34%*	73.73%	88.66%	72.73%	71.81%
Sonar	94.66%	98.91%	92.74%	*103.50%*	*109.41%*	*115.15%*	*107.62%*	*110.06%*	*106.54%*	*103.24%*	*103.20%*	94.73%	93.96%
Parkinsons	65.14%	*100.21%*	70.28%	86.74%	*116.07%*	99.84%	*123.64%*	*103.34%*	*101.74%*	74.59%	92.52%	90.79%	67.29%
Ex8b	90.08%	*109.88%*	93.86%	95.59%	*107.28%*	91.14%	*126.40%*	*104.15%*	*129.39%*	90.31%	*103.05%*	90.49%	93.70%
Heart	86.90%	*107.95%*	84.11%	*107.17%*	97.55%	*103.64%*	*113.07%*	*103.62%*	*109.48%*	97.06%	95.77%	91.87%	89.85%
Haberman	82.19%	95.22%	82.67%	*111.45%*	*105.52%*	*129.80%*	*161.28%*	*141.81%*	*107.57%*	92.66%	*104.88%*	*118.27%*	*100.55%*
Ionosphere	61.01%	*111.80%*	63.87%	*105.88%*	*108.81%*	96.76%	*119.75%*	*111.23%*	*209.02%*	76.15%	*139.82%*	78.44%	65.46%
Clean1	75.75%	*103.42%*	76.96%	99.94%	*104.20%*	94.65%	97.48%	95.94%	*100.00%*	82.71%	99.69%	88.50%	82.04%
Breast	69.35%	*109.64%*	66.14%	89.34%	*147.47%*	95.73%	*100.94%*	95.69%	*494.88%*	71.37%	*133.41%*	66.72%	49.26%
Wdbc	64.80%	*133.42%*	62.73%	*108.66%*	*131.75%*	*109.41%*	*138.49%*	*116.93%*	*117.16%*	68.42%	*119.39%*	64.54%	59.28%
Australian	77.16%	*115.07%*	79.27%	*107.96%*	*128.48%*	*144.82%*	*159.60%*	*127.39%*	*130.52%*	92.17%	*128.82%*	*111.12%*	85.45%
Diabetes	93.03%	*102.01%*	*107.05%*	89.91%	*114.95%*	*108.80%*	*181.30%*	*112.29%*	*171.75%*	94.00%	*105.48%*	*101.62%*	76.47%
Mammographic	92.07%	*120.26%*	78.18%	*124.13%*	*128.84%*	*112.22%*	*237.91%*	*103.86%*	*202.17%*	88.01%	*119.19%*	82.66%	79.15%
Ex8a	80.82%	*101.76%*	82.20%	97.18%	*110.12%*	62.62%	*164.20%*	*158.36%*	*134.14%*	79.50%	84.62%	*153.98%*	69.71%
Tic	80.79%	*128.22%*	79.62%	*118.52%*	*137.88%*	*151.43%*	*147.20%*	*111.52%*	*164.54%*	*105.34%*	TLE	*137.67%*	*118.43%*
German	99.67%	*119.68%*	*103.79%*	*125.82%*	*122.31%*	*139.69%*	*114.99%*	*124.65%*	*190.73%*	*113.27%*	TLE	*113.41%*	95.44%
Splice	50.12%	*109.42%*	52.06%	*110.84%*	*101.56%*	96.64%	*116.59%*	97.02%	*165.10%*	58.39%	*105.95%*	99.16%	77.40%
Gcloudb	73.44%	*124.36%*	71.91%	*103.12%*	*145.65%*	78.12%	*222.56%*	*289.43%*	*152.25%*	72.98%	85.25%	72.23%	64.86%
Gcloudub	67.80%	*118.36%*	79.22%	*102.15%*	*131.47%*	*168.09%*	*310.10%*	*198.38%*	*303.37%*	87.16%	*156.37%*	*117.76%*	66.35%
Checkerboard	*231.82%*	*140.07%*	93.83%	*110.34%*	*115.64%*	*154.47%*	*529.48%*	*543.18%*	*660.77%*	72.37%	*105.94%*	*153.28%*	39.50%
Spambase	25.23%	97.59%	28.02%	87.65%	*121.97%*	*106.69%*	*196.13%*	TLE	*101.91%*	37.69%	TLE	82.08%	37.18%
Banana	*111.31%*	95.73%	*121.28%*	*106.59%*	*117.47%*	65.76%	*448.34%*	*393.05%*	*574.50%*	*103.93%*	TLE	*132.50%*	43.66%
Phoneme	35.59%	97.05%	36.87%	73.34%	*103.54%*	80.84%	*107.40%*	TLE	*165.20%*	43.68%	TLE	90.82%	44.67%
Ringnorm	31.06%	91.33%	32.57%	*111.24%*	*102.61%*	*208.70%*	*158.46%*	TLE	*195.48%*	44.30%	TLE	94.07%	73.23%
Twonorm	34.59%	*115.91%*	36.78%	*112.05%*	*103.37%*	75.10%	*529.37%*	TLE	*173.40%*	43.62%	TLE	73.05%	74.54%
Phishing	27.08%	*131.84%*	28.23%	*117.78%*	*137.85%*	*109.86%*	57.30%	TLE	*244.87%*	37.25%	TLE	55.39%	31.95%
Covertype	41.33%	*109.86%*	40.35%	TLE	*116.98%*	TLE	TLE	TLE	*117.86%*	TLE	TLE	TLE	46.23%
Bioresponse	64.95%	*105.42%*	64.58%	89.29%	96.76%	91.78%	TLE	TLE	*100.00%*	73.95%	TLE	TLE	66.72%
Pol	29.38%	96.56%	27.82%	96.41%	98.11%	*152.58%*	34.08%	TLE	*100.00%*	41.06%	TLE	*126.70%*	31.84%

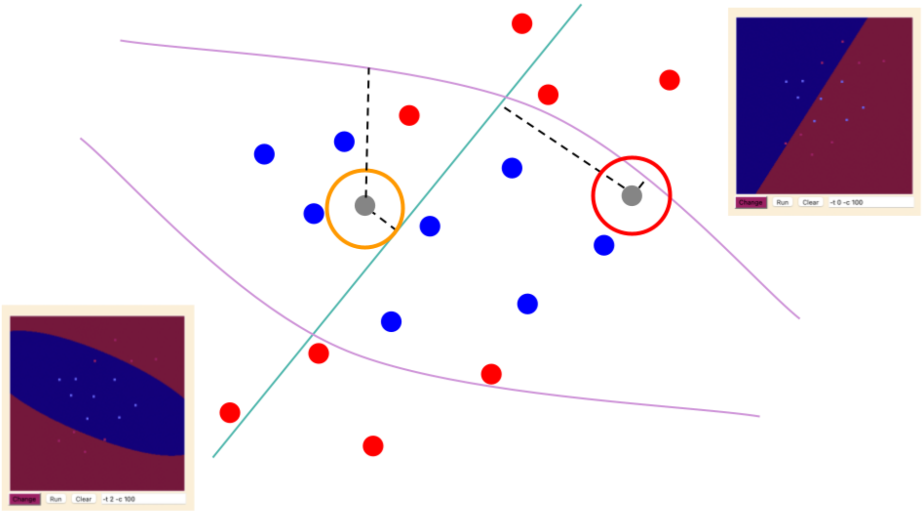


Figure 4: Given RBFSVM as the task-oriented model, we study the non-compatible query-oriented model with LR($C = 0.1$). The red and blue points represent labeled examples. The gray points represent unlabeled examples. The cyan and magenta lines indicate the decision boundaries of query models LR($C = 0.1$) and RBFSVM trained on current labeled examples. If we adopt US, the non-compatible setting queries a sample (orange circle), which is most uncertain to LR($C = 0.1$) rather than the most uncertain sample to RBFSVM (red circle).

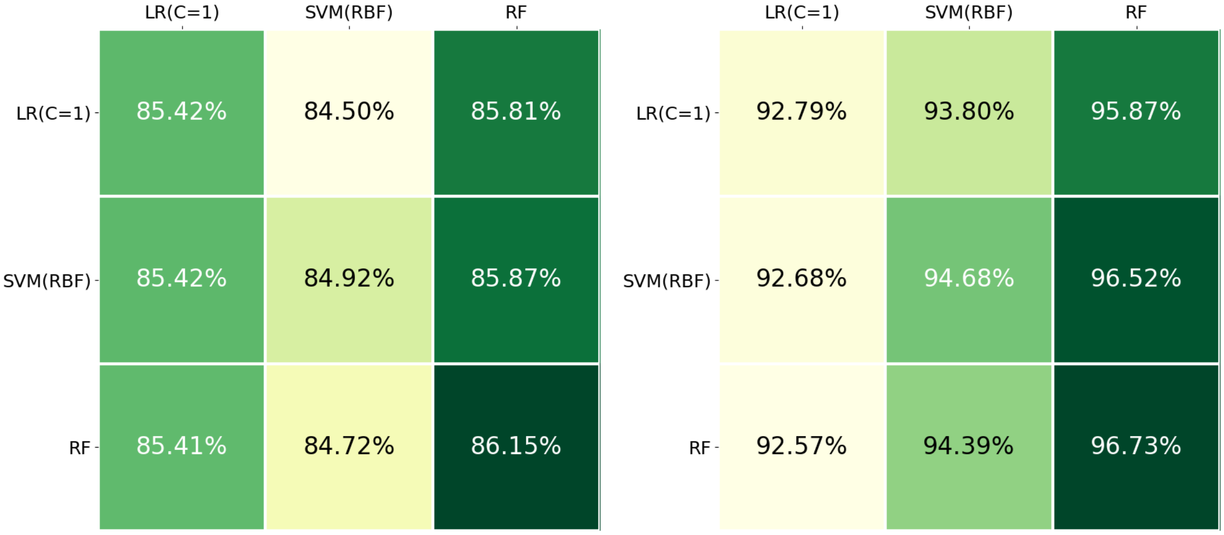


Figure 5: Mean AUBC of a query-oriented model (rows) and a task-oriented model (columns) on *Australian* (left) and *Phishing* (right)

5.2 Extending the usefulness of uncertainty sampling

We extend the existing benchmark to real-world datasets used in another tabular data benchmark Grinsztajn et al. (2022) to demonstrate the usefulness of US within our current benchmark and its potential applicability and benefits across a more comprehensive array of real-world datasets. Real-world datasets include a larger number of examples, such as *Pol* and *Covertype*, and higher dimensions, such as *Bioresponse*. By extending our evaluation to these datasets, we aim to illustrate that the consistent usefulness of US is not limited to the existing benchmark.

In Figure 6, similar to Section 4.2, US could bring more benefits than Uniform at the early stage. These results affirm that US has potential as an applicable approach across large-scale and high-dimension scenarios, which encourages the exploration of US in broader applications.

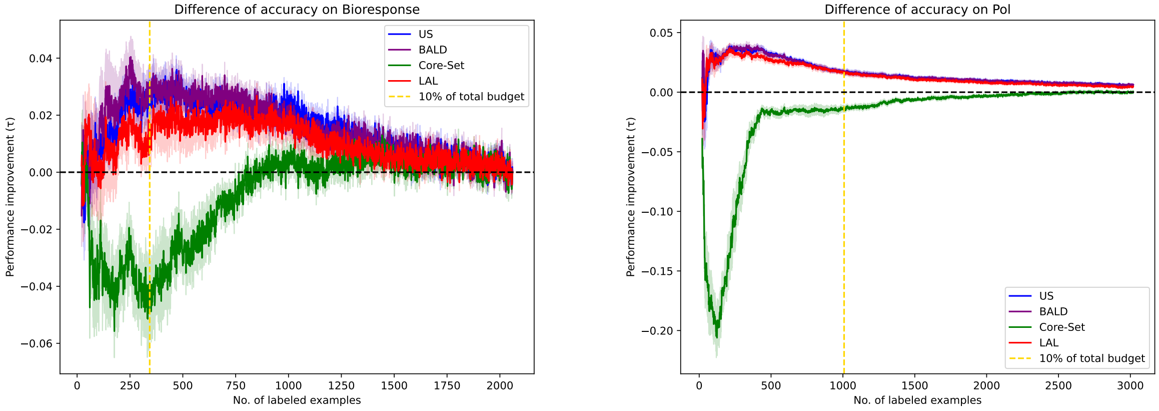


Figure 6: Mean difference of AUBC (improvement) of a query strategy from Uniform on *Bioresponse* (left) and *Pol* (right).

6 Conclusion

This work presents the most comprehensive survey and open-source benchmark for active learning to date. Our benchmark, with its transparent and unified interface, incorporates existing GitHub repositories, providing a thorough and up-to-date comparison of active learning query strategies. We equip Uncertainty Sampling with compatible models and affirm that it remains superior to other active learning strategies as well as Uniform Sampling on most of the datasets. Furthermore, we discover that Uncertainty Sampling can be affected by the incompatibility between query-oriented and task-oriented models, resulting in discrepancies between previous benchmarks. Our affirmation suggests Uncertainty Sampling with compatible query-oriented and task-oriented models as a first-hand choice for practitioners. These insights not only enhance the community's comprehension of current active learning strategies but also establish a foundation for future research with this practical guide. We anticipate extending our framework to encompass diverse domains like vision and languages and incorporating various models such as deep neural networks, as outlined in Appendix F for future exploration.

Broader Impact Statement

Active learning is a long-term research topic in machine learning, yet achieving a consensus on the best strategies within the community is challenging. This work starts from the tabular data to build the most comprehensive open-source active learning benchmark to date. We affirm that Uncertainty Sampling (US) remains superior to other active learning strategies and Uniform on most datasets. We also clarify conflicting conclusions in previous benchmarks by carefully verifying previous settings. Our work will benefit the active learning community by providing a transparent and unified framework for evaluating active learning strategies compared to a strong baseline–US with compatible settings. We hope our work will help practitioners check the reality of existing active learning strategies and settings for different domains. Moreover, re-examine the potential issues in existing benchmarks, such as neglected settings and unpublished analysis steps.

References

Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. *arXiv preprint arXiv:1906.03671*, 2019.

Dara Bahri, Heinrich Jiang, Tal Schuster, and Afshin Rostamizadeh. Is margin all you need? an extensive empirical study of active learning on tabular data. *arXiv preprint arXiv:2210.03822*, 2022.

Nathan Beck, Durga Sivasubramanian, Apurva Dani, Ganesh Ramakrishnan, and Rishabh Iyer. Effective evaluation of deep active learning on image classification tasks. *arXiv preprint arXiv:2106.15324*, 2021.

William H Beluch, Tim Genewein, Andreas Nürnberger, and Jan M Köhler. The power of ensembles for active learning in image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9368–9377, 2018.

Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

Clemens-Alexander Brust, Christoph Käding, and Joachim Denzler. Active learning for deep object detection. *arXiv preprint arXiv:1809.09875*, 2018.

Wenbin Cai, Muhan Zhang, and Ya Zhang. Batch mode active learning for regression with expected model change. *IEEE transactions on neural networks and learning systems*, 28(7):1668–1681, 2016.

Gavin C Cawley. Baseline methods for active learning. In *Active Learning and Experimental Design workshop In conjunction with AISTATS 2010*, pp. 47–57. JMLR Workshop and Conference Proceedings, 2011.

Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Rita Chattopadhyay, Zheng Wang, Wei Fan, Ian Davidson, Sethuraman Panchanathan, and Jieping Ye. Batch mode active sampling based on marginal probability distribution matching. *KDD : proceedings. International Conference on Knowledge Discovery & Data Mining*, 2012:741–749, 2012.

Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pp. 785–794, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4232-2. doi: 10.1145/2939672.2939785. URL `http://doi.acm.org/10.1145/2939672.2939785`.

Thomas M Cover. *Elements of information theory*. John Wiley & Sons, 1999.

Matt Culver, Deng Kun, and Stephen Scott. Active learning to maximize area under the roc curve. In *Sixth International Conference on Data Mining (ICDM'06)*, pp. 149–158, 2006. doi: 10.1109/ICDM.2006.12.

Tivadar Danka and Peter Horvath. modAL: A modular active learning framework for Python. URL `https://github.com/modAL-python/modAL`. available on arXiv at `https://arxiv.org/abs/1805.00979`.

Sanjoy Dasgupta and Daniel Hsu. Hierarchical sampling for active learning. In *Proceedings of the 25th international conference on Machine learning*, pp. 208–215, 2008.

Emre Demir, Zehra Cataltepe, Umit Ekmekci, Mateusz Budnik, and Laurent Besacier. Unsupervised Active Learning For Video Annotation. In *ICML Active Learning Workshop 2015*, Lille, France, July 2015. URL `https://hal.science/hal-01350092`.

Louis Desreumaux and Vincent Lemaire. Learning active learning at the crossroads? evaluation and discussion. *arXiv e-prints*, art. arXiv:2012.09631, December 2020.

Pinar Donmez, Jaime G. Carbonell, and Paul N. Bennett. Dual strategy active learning. In Joost N. Kok, Jacek Koronacki, Raomon Lopez de Mantaras, Stan Matwin, Dunja Mladenič, and Andrzej Skowron (eds.), *Machine Learning: ECML 2007*, pp. 116–127, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg. ISBN 978-3-540-74958-5.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL `http://archive.ics.uci.edu/ml`.

Sandra Ebert, Mario Fritz, and Bernt Schiele. Ralf: A reinforced active learning formulation for object class recognition. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3626–3633, 2012. doi: 10.1109/CVPR.2012.6248108.

Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. In *International conference on machine learning*, pp. 1183–1192. PMLR, 2017.

Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? *Advances in Neural Information Processing Systems*, 35:507–520, 2022.

Isabelle Guyon, Gavin Cawley, Gideon Dror, and Vincent Lemaire. Design and analysis of the wcci 2010 active learning challenge. In *The 2010 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, 2010. doi: 10.1109/IJCNN.2010.5596506.

Isabelle Guyon, Gavin C. Cawley, Gideon Dror, and Vincent Lemaire. Results of the active learning challenge. In Isabelle Guyon, Gavin Cawley, Gideon Dror, Vincent Lemaire, and Alexander Statnikov (eds.), *Active Learning and Experimental Design workshop In conjunction with AISTATS 2010*, volume 16 of *Proceedings of Machine Learning Research*, pp. 19–45, Sardinia, Italy, 16 May 2011. PMLR. URL `https://proceedings.mlr.press/v16/guyon11a.html`.

Zhongkai Hao, Chengqiang Lu, Zhenya Huang, Hao Wang, Zheyuan Hu, Qi Liu, Enhong Chen, and Cheekong Lee. Asgn: An active semi-supervised graph neural network for molecular property prediction. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 731–752, 2020.

Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011.

Wei-Ning Hsu and Hsuan-Tien Lin. Active learning by learning. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pp. 2659–2665, January 2015.

Sheng-jun Huang, Rong Jin, and Zhi-Hua Zhou. Active learning by querying informative and representative examples. In J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta (eds.), *Advances in Neural Information Processing Systems*, volume 23. Curran Associates, Inc., 2010. URL `https://proceedings.neurips.cc/paper/2010/file/5487315b1286f907165907aa8fc96619-Paper.pdf`.

Yilin Ji, Daniel Kaestner, Oliver Wirth, and Christian Wressnegger. Randomness is the root of all evil: More reliable evaluation of deep active learning. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 3932–3941, 2023. doi: 10.1109/WACV56688.2023.00393.

Siddharth Karamcheti, Ranjay Krishna, Li Fei-Fei, and Christopher Manning. Mind your outliers! investigating the negative impact of outliers on active learning for visual question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 7265–7281, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.564. URL `https://aclanthology.org/2021.acl-long.564`.

Jeeveswaran Kishaan, Mohandass Muthuraja, Deebul Nair, and Paul-Gerhard Plöger. Using active learning for assisted short answer grading. 2020.

Ksenia Konyushkova, Raphael Sznitman, and Pascal Fua. Learning active learning from data. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL `https://proceedings.neurips.cc/paper/2017/file/8ca8da41fe1ebc8d3ca31dc14f5fc56c-Paper.pdf`.

Daniel Kottke, Adrian Calma, Denis Huseljic, GM Krempl, Bernhard Sick, et al. Challenges of reliable, realistic and comparable active learning evaluation. In *Proceedings of the Workshop and Tutorial on Interactive Adaptive Learning*, pp. 2–14, 2017.

Daniel Kottke, Marek Herde, Tuan Pham Minh, Alexander Benz, Pascal Mergard, Atal Roghman, Christoph Sandrock, and Bernhard Sick. scikit-activeml: A Library and Toolbox for Active Learning Algorithms. *Preprints*, 2021. doi: 10.20944/preprints202103.0194.v1. URL `https://github.com/scikit-activeml/scikit-activeml`.

Jan Kremer, Kim Steenstrup Pedersen, and Christian Igel. Active learning with support vector machines. *Wiley Int. Rev. Data Min. and Knowl. Disc.*, 4(4):313–326, jul 2014. ISSN 1942-4787. doi: 10.1002/widm.1132. URL `https://doi.org/10.1002/widm.1132`.

Chun-Liang Li, Chun-Sung Ferng, and Hsuan-Tien Lin. Active learning using hint information. *Neural computation*, 27(8):1738–1765, 2015.

Xin Li and Yuhong Guo. Adaptive active learning for image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 859–866, 2013.

Qiang Liu, Yanqiao Zhu, Zhaocheng Liu, Yufeng Zhang, and Shu Wu. Deep active learning for text classification with diverse interpretations. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pp. 3263–3267, 2021.

Yash-yee Logan, Mohit Prabhushankar, and Ghassan AlRegib. Decal: Deployable clinical active learning. *arXiv preprint arXiv:2206.10120*, 2022.

Po-Yi Lu, Chun-Liang Li, and Hsuan-Tien Lin. A more robust baseline for active learning by injecting randomness to uncertainty sampling. In *Proceedings of the AI and HCI Workshop @ ICML*, July 2023.

Carsten T Lüth, Till J Bungert, Lukas Klein, and Paul F Jaeger. Toward realistic evaluation of deep active learning algorithms in image classification. *arXiv preprint arXiv:2301.10625*, 2023.

Prateek Munjal, Nasir Hayat, Munawar Hayat, Jamshid Sourati, and Shadab Khan. Towards robust and reproducible active learning using neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 223–232, 2022.

Stephen Mussmann and Percy Liang. On the relationship between data efficiency and error for uncertainty sampling. In *International Conference on Machine Learning*, pp. 3674–3682. PMLR, 2018.

S Deepak Narayanan, Apoorv Agnihotri, and Nipun Batra. Active learning for air quality station deployment. 2020.

V. Nath, Dong Yang, Bennett A. Landman, Daguang Xu, and Holger R. Roth. Diminishing uncertainty within the training pool: Active learning for medical image segmentation. *IEEE Transactions on Medical Imaging*, 40:2534–2547, 2020.

Hieu T Nguyen and Arnold Smeulders. Active learning using pre-clustering. In *Proceedings of the twenty-first international conference on Machine learning*, pp. 79, 2004.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Christopher Schröder, Andreas Niekler, and Martin Potthast. Revisiting uncertainty-based query strategies for active learning with transformers. In *Findings*, 2021. URL `https://api.semanticscholar.org/CorpusID:235828923`.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018. URL `https://openreview.net/forum?id=H1aIuk-RW`.

Burr Settles. *Active Learning*, volume 6. Springer, 7 2012. doi: 10.2200/S00429ED1V01Y201207AIM018. URL `https://www.morganclaypool.com/doi/abs/10.2200/S00429ED1V01Y201207AIM018?ai=1ge&mi=6e3g68&af=R`.

H Sebastian Seung, Manfred Opper, and Haim Sompolinsky. Query by committee. In *Proceedings of the fifth annual workshop on Computational learning theory*, pp. 287–294, 1992.

Changjian Shui, Fan Zhou, Christian Gagné, and Boyu Wang. Deep active learning: Unified and principled method for query and training. In *International Conference on Artificial Intelligence and Statistics*, pp. 1308–1318. PMLR, 2020.

Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. Variational adversarial active learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5972–5981, 2019.

Ying-Peng Tang and Sheng-Jun Huang. Self-paced active learning: Query the right thing at the right time. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):5117–5124, Jul. 2019. doi: 10.1609/aaai.v33i01.33015117. URL `https://ojs.aaai.org/index.php/AAAI/article/view/4445`.

Ying-Peng Tang, Guo-Xiang Li, and Sheng-Jun Huang. ALiPy: Active learning in python. Technical report, Nanjing University of Aeronautics and Astronautics, 1 2019. URL `https://github.com/NUAA-AL/ALiPy`. available as arXiv preprint `https://arxiv.org/abs/1901.03802`.

Alexandru Tifrea, Jacob Clarysse, and Fanny Yang. Uniform versus uncertainty sampling: When being active is less efficient than staying passive. *arXiv preprint arXiv:2212.00772*, 2022.

Holger Trittenbach, Adrian Englhardt, and Klemens Böhm. An overview and a benchmark of active learning for outlier detection with one-class classifiers. *Expert Systems with Applications*, 168:114372, 2021. ISSN 0957-4174. doi: https://doi.org/10.1016/j.eswa.2020.114372. URL `https://www.sciencedirect.com/science/article/pii/S0957417420310496`.

Zheng Wang and Jieping Ye. Querying discriminative and representative samples for batch mode active learning. *ACM Trans. Knowl. Discov. Data*, 9(3), feb 2015. ISSN 1556-4681. doi: 10.1145/2700408. URL `https://doi.org/10.1145/2700408`.

Dongrui Wu, Chin-Teng Lin, and Jian Huang. Active learning for regression using greedy sampling. *Information Sciences*, 474:90–105, 2019.

Zhao Xu, Kai Yu, Volker Tresp, Xiaowei Xu, and Jizhi Wang. Representative sampling for text classification using support vector machines. In Fabrizio Sebastiani (ed.), *Advances in Information Retrieval*, pp. 393–407, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg. ISBN 978-3-540-36618-8.

Yao-Yuan Yang, Shao-Chuan Lee, Yu-An Chung, Tung-En Wu, Si-An Chen, and Hsuan-Tien Lin. libact: Pool-based active learning in python. Technical report, National Taiwan University, 10 2017. URL `https://github.com/ntucllab/libact`. available as arXiv preprint `https://arxiv.org/abs/1710.00379`.

Yazhou Yang and Marco Loog. A benchmark and comparison of active learning for logistic regression. *Pattern Recognition*, 83:401–415, 2018. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2018.06.004. URL `https://www.sciencedirect.com/science/article/pii/S0031320318302140`.

Yi Yang, Zhigang Ma, Feiping Nie, Xiaojun Chang, and Alexander G Hauptmann. Multi-class active learning by uncertainty sampling with diversity maximization. *International Journal of Computer Vision*, 113:113–127, 2015.

rostamiz Yilei “Dolee” Yang. Active learning playground. `https://github.com/google/active-learning`, 2017. URL `https://github.com/google/active-learning`.

Donggeun Yoo and In So Kweon. Learning loss for active learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 93–102, 2019.

Tianning Yuan, Fang Wan, Mengying Fu, Jianzhuang Liu, Songcen Xu, Xiangyang Ji, and Qixiang Ye. Multiple instance active learning for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5330–5339, 2021.

Xueying Zhan, Huan Liu, Qing Li, and Antoni B. Chan. A comparative survey: Benchmarking for pool-based active learning. In Zhi-Hua Zhou (ed.), *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pp. 4679–4686. International Joint Conferences on Artificial Intelligence Organization, 8 2021. doi: 10.24963/ijcai.2021/634. URL `https://doi.org/10.24963/ijcai.2021/634`. Survey Track.

Xueying Zhan, Qingzhong Wang, Kuan-hao Huang, Haoyi Xiong, Dejing Dou, and Antoni B Chan. A comparative survey of deep active learning. *arXiv preprint arXiv:2203.13450*, 2022.

Hongjing Zhang, SS Ravi, and Ian Davidson. A graph-based approach for active learning in regression. In *Proceedings of the 2020 SIAM International Conference on Data Mining*, pp. 280–288. SIAM, 2020.

Zhisong Zhang, Emma Strubell, and Eduard Hovy. A survey of active learning for natural language processing. *arXiv preprint arXiv:2210.10109*, 2022.

Table 7: Accuracy of the model with 10% labeled examples: We report the accuracy of the model with 10% labeled examples on each dataset. The scores with **bold** mean the best performance on a dataset. 'TLE' means a query strategy exceeds the time limit.

	Uniform	US	QBC	BALD	Hier	Graph	Core-Set	HintSVM	QUIRE	DWUS	MCM	BMDR	ALBL	LAL
Ionosphere	74.99%	75.43%	74.94%	74.91%	74.57%	74.86%	75.82%	74.91%	75.30%	74.31%	74.83%	75.06%	75.25%	**75.96%**
Clean1	62.60%	62.65%	62.32%	62.39%	**63.36%**	61.67%	62.52%	61.93%	62.60%	62.60%	61.44%	61.71%	62.82%	62.88%
Breast	93.33%	94.95%	93.13%	94.68%	94.26%	90.32%	93.61%	93.74%	93.56%	90.49%	94.65%	93.18%	94.73%	**95.88%**
Wdbc	90.54%	90.28%	90.58%	91.38%	90.09%	87.78%	91.33%	90.49%	90.96%	90.07%	89.88%	89.29%	91.92%	**92.32%**
Australian	81.00%	**83.13%**	81.17%	82.96%	81.37%	80.44%	76.49%	78.11%	77.38%	79.75%	80.85%	80.20%	80.11%	82.36%
Diabetes	70.22%	**72.07%**	70.52%	71.40%	70.85%	69.98%	69.25%	69.34%	68.88%	69.30%	70.90%	70.29%	70.43%	71.68%
Mammographic	79.92%	81.45%	79.38%	**81.76%**	79.46%	79.40%	79.61%	78.65%	79.07%	78.72%	81.32%	79.29%	80.26%	81.56%
Ex8a	79.53%	79.62%	79.70%	78.66%	79.49%	78.32%	**83.99%**	76.19%	76.52%	71.73%	78.38%	82.90%	79.19%	82.25%
Tic	86.54%	**88.12%**	86.63%	87.95%	86.46%	84.66%	87.33%	85.39%	85.46%	81.19%	87.07%	TLE	87.58%	87.09%
German	69.23%	**70.75%**	69.46%	70.37%	69.11%	69.35%	68.96%	69.18%	69.28%	68.47%	69.64%	TLE	69.29%	70.45%
Splice	83.16%	**84.98%**	82.06%	84.91%	82.45%	75.17%	76.06%	78.42%	83.03%	79.81%	81.08%	80.61%	81.79%	82.71%
Gcloudb	86.36%	89.11%	86.39%	88.86%	87.14%	83.93%	87.18%	84.21%	84.60%	84.17%	88.84%	87.09%	87.57%	**89.15%**
Gcloudub	88.19%	88.91%	88.41%	87.93%	88.47%	86.93%	86.82%	82.97%	84.66%	79.88%	86.87%	87.70%	86.01%	**90.77%**
Checkerboard	97.32%	95.81%	97.08%	97.67%	97.93%	96.57%	96.78%	90.18%	87.09%	75.27%	98.66%	98.23%	94.78%	**99.80%**
Spambase	90.77%	**93.85%**	89.93%	93.62%	90.63%	90.97%	88.96%	85.38%	TLE	90.85%	93.11%	TLE	91.09%	93.04%
Banana	85.76%	83.20%	86.17%	82.74%	86.53%	83.44%	87.51%	69.76%	62.64%	70.87%	84.51%	TLE	85.29%	**88.43%**
Phoneme	81.66%	**85.13%**	80.97%	84.77%	82.30%	80.62%	81.45%	78.38%	TLE	76.87%	83.62%	TLE	81.47%	83.64%
Ringnorm	89.55%	93.89%	89.92%	**94.02%**	87.82%	57.35%	54.17%	55.70%	TLE	57.73%	92.03%	TLE	88.21%	89.32%
Twonorm	93.97%	95.92%	93.56%	**95.95%**	93.31%	93.73%	94.45%	80.79%	TLE	92.31%	95.70%	TLE	94.50%	94.42%
Phishing	92.17%	94.62%	92.06%	**94.84%**	91.76%	91.18%	92.22%	90.97%	TLE	87.74%	93.80%	TLE	93.20%	94.28%
Covertype	70.98%	**73.70%**	70.22%	73.43%	TLE	64.46%	TLE	61.73%	TLE	58.76%	67.50%	TLE	66.92%	71.90%
Bioresponse	66.64%	68.38%	66.74%	**68.99%**	65.45%	65.64%	63.52%	60.64%	TLE	66.64%	67.77%	TLE	66.55%	67.88%
Pol	93.24%	96.67%	93.27%	**96.84%**	92.96%	84.26%	85.70%	75.77%	TLE	93.24%	95.24%	TLE	93.67%	96.15%

A Detailed benchmarking results of XGBoost

We present more settings of the benchmarking results for XGBoost for verifying the superiority of query strategies in Section 4.1. We check the accuracy of the model with different ratios, e.g., 10% and 30% of labeled examples on each dataset. Tables 7 and 8 also confirm that US outperforms other query strategies on most datasets. It is worth mentioning that LAL achieves good performance on *Gcloudb*, *Gcloudub*, and *Checkerboard* when the ratio of labeled examples is 10%. However, these datasets are synthetic, and their features may be more similar to the pre-trained datasets used by LAL, resulting in LAL's exceptional performance on these datasets.

B Revision of Zhan et al. (2021)

In this section, we reveal and revise descriptions in Zhan et al. (2021) to study the conflicting conclusions in previous benchmarks and provide clear information to the active learning community. We appreciate that Zhan et al. (2021) published their source code on GitHub.[4] Thus we could examine the difference from our settings.

B.1 Experimental Settings

Inputs and base models. At the initial setup, Zhan et al. (2021) employed a random split of 60% of the dataset for the training set and the remaining 40% for the testing set. No pre-processing was applied to the dataset, and fixed random seeds were used to ensure consistency in the training and testing sets across repeated experiments. They used an RBFSVM as the task-oriented model for evaluating the query strategies.

Query strategies. To compare the performance of 17 query strategies, they implemented random sampling and all query strategies using different libraries. The libact library provided implementations for Uncertainty Sampling (US), Query by Committee (QBC), Hinted Support Vector Machine (HintSVM), QUerying Informative and REpresentative Examples (QUIRE), Active Learning by Learning (ALBL), Den-

[4] `https://github.com/SineZHAN/ComparativeSurveyIJCAI2021PoolBasedAL`

Table 8: Accuracy of the model with 30% labeled examples: We report the accuracy of the model with 30% labeled examples on each dataset. The scores with **bold** mean the best performance on a dataset. 'TLE' means a query strategy exceeds the time limit.

	Uniform	US	QBC	BALD	Hier	Graph	Core-Set	HintSVM	QUIRE	DWUS	MCM	BMDR	ALBL	LAL
Sonar	71.18%	71.71%	70.73%	71.90%	70.89%	71.02%	67.89%	69.06%	70.20%	71.43%	70.10%	**72.37%**	71.75%	71.99%
Parkinsons	81.44%	**84.95%**	81.88%	84.38%	81.87%	81.95%	81.58%	80.50%	81.74%	81.74%	83.74%	82.67%	83.21%	84.08%
Ex8b	82.76%	84.21%	82.99%	83.14%	83.11%	82.54%	84.42%	81.99%	83.48%	81.54%	83.88%	83.51%	84.40%	**84.87%**
Heart	77.78%	**79.78%**	77.90%	79.50%	77.85%	77.84%	78.07%	76.76%	77.99%	77.84%	78.29%	78.49%	78.37%	79.27%
Haberman	68.87%	**71.21%**	68.41%	70.92%	69.58%	68.96%	68.54%	67.49%	67.65%	69.13%	70.33%	69.24%	68.85%	70.33%
Ionosphere	85.97%	**90.30%**	86.70%	90.27%	87.21%	87.41%	84.42%	84.07%	79.92%	78.34%	88.85%	81.04%	88.76%	89.99%
Clean1	72.59%	**74.79%**	72.52%	74.26%	72.36%	72.40%	71.87%	72.59%	72.87%	72.59%	72.54%	72.03%	73.59%	74.59%
Breast	95.68%	96.69%	95.77%	96.76%	96.26%	95.52%	96.08%	95.74%	95.99%	90.15%	96.64%	95.62%	96.36%	**96.81%**
Wdbc	93.73%	95.95%	93.77%	95.95%	94.28%	93.86%	94.07%	92.98%	93.94%	93.58%	95.94%	93.71%	95.75%	**95.99%**
Australian	84.59%	**86.25%**	84.62%	86.12%	84.56%	85.00%	84.50%	82.80%	84.79%	83.76%	86.00%	84.76%	84.91%	85.77%
Diabetes	72.60%	**74.27%**	72.84%	73.84%	72.72%	73.12%	73.19%	70.34%	72.86%	71.05%	73.59%	72.59%	72.54%	72.78%
Mammographic	79.53%	82.05%	79.27%	82.07%	79.42%	78.87%	79.10%	79.10%	79.60%	78.65%	**82.08%**	79.89%	79.17%	81.69%
Ex8a	91.08%	92.63%	91.30%	92.75%	91.20%	91.01%	**94.96%**	77.68%	80.40%	82.04%	93.09%	92.75%	88.08%	93.89%
Tic	90.12%	91.07%	90.12%	**91.29%**	90.11%	90.15%	89.58%	89.37%	90.21%	88.67%	90.78%	TLE	89.79%	90.11%
German	72.16%	**73.73%**	72.50%	73.56%	72.35%	72.71%	72.11%	72.40%	71.84%	70.83%	73.47%	TLE	72.96%	73.24%
Splice	91.53%	**95.47%**	91.53%	95.17%	91.36%	91.75%	90.43%	89.53%	91.76%	87.21%	94.79%	91.80%	91.82%	92.98%
Gcloudb	88.09%	89.06%	88.20%	89.03%	88.23%	88.33%	88.56%	84.96%	84.50%	85.99%	89.15%	88.26%	88.62%	**89.27%**
Gcloudub	92.67%	**95.38%**	92.52%	94.84%	93.37%	92.76%	90.66%	84.03%	87.36%	81.48%	94.49%	91.84%	91.76%	94.42%
Checkerboard	99.29%	97.22%	99.19%	99.76%	99.61%	99.16%	99.47%	91.51%	91.71%	79.49%	99.65%	99.56%	99.29%	**99.81%**
Spambase	93.10%	**95.14%**	93.22%	95.11%	93.45%	93.07%	92.99%	90.03%	TLE	93.16%	94.88%	TLE	93.49%	95.04%
Banana	88.02%	89.08%	87.83%	88.98%	87.86%	87.75%	87.96%	74.16%	79.78%	76.83%	**89.16%**	TLE	87.43%	88.98%
Phoneme	85.14%	**88.47%**	85.14%	88.20%	86.02%	84.93%	85.59%	83.45%	TLE	80.68%	88.15%	TLE	85.15%	88.00%
Ringnorm	94.22%	96.15%	94.59%	**96.32%**	93.38%	94.01%	51.04%	64.64%	TLE	55.09%	96.12%	TLE	94.06%	94.94%
Twonorm	95.67%	96.76%	95.67%	**96.77%**	95.76%	95.73%	96.03%	77.35%	TLE	95.08%	96.75%	TLE	96.02%	96.57%
Phishing	94.09%	96.56%	93.67%	96.55%	93.79%	93.98%	93.83%	91.96%	TLE	91.03%	96.38%	TLE	94.81%	**96.60%**
Covertype	73.54%	**76.30%**	73.16%	76.25%	TLE	65.82%	TLE	63.94%	TLE	60.21%	73.13%	TLE	69.72%	74.64%
Bioresponse	72.11%	74.10%	72.51%	**74.72%**	71.91%	73.20%	69.89%	69.11%	TLE	72.11%	73.33%	TLE	71.76%	73.78%
Pol	96.34%	**98.28%**	96.22%	98.25%	96.41%	96.70%	94.84%	91.99%	TLE	96.34%	98.14%	TLE	95.93%	98.25%

sity Weighted Uncertainty Sampling (DWUS), and Variation Reduction (VR). The Google library included Random Sampling (Uniform), k-Center-Greedy (KCenter or Core-Set), Margin-based Uncertainty Sampling (Margin), Graph Density (Graph), Hierarchical Sampling (Hier), Informative Cluster Diverse (InfoDiv), and Representative Sampling (MCM). The ALiPy library contributed Estimation of Error Reduction (EER), BMDR, SPAL, and LAL. Besides, they proposed the Beam-Search Oracle (BSO) as a reference to approximate the optimal sequence of queried samples that maximizes performance on the testing set, aiming to assess the potential improvement space for query strategies on specific datasets. Through reviewing their released source code, we identified differences between the task-oriented and query-oriented models for specific query strategies. Table 9 highlights the discrepancies between the two models for each query strategy.[5] In particular, Margin and US (US-C and US-NC in our notation) are variant settings for Uncertainty Sampling. We further discuss such differences in Section 5.1. In re-benchmarking (Appendix C), we adopt RBFSVM for a query strategy and evaluation by default.

Experimental design. The active learning algorithm was stopped when the total budget was equal to the size of the unlabeled pool, $T = |D_{\text{u}}^{(0)}|$. They collected the testing accuracy at each round to construct a learning curve, and the AUBC was calculated to summarize the performance of a query strategy on a dataset. To ensure reliable results, they conducted $K_{\text{S}} = 100$ repeated experiments for small datasets ($n < 2000$) and $K_{\text{L}} = 10$ repeated experiments for large datasets ($n \geq 2000$), where n represents the size of the dataset. Finally, they compute the average AUBCs across repeated experiments for each query strategy on each dataset.

Analysis methods. Zhan et al. (2021) benchmarked the pool-based active learning for classifications on 35 datasets, including 26 binary-class and 9 multi-class datasets collected from LIBSVM and UCI Chang & Lin (2011); Dua & Graff (2017). They provided the data properties, such as the number of samples n, dimension d, and imbalance ratio IR, where the imbalance ratio is the proportion of negative labels to the

[5] The settings are different from their source code for Google and ALiPy[4].

Table 9: Settings of query-oriented models $\mathcal{H}$ for specific query strategies $\mathcal{Q}$ in (Zhan et al., 2021).

$\mathcal{Q}$	$\mathcal{H}$
US(Zhan et al. (2021)) , US-NC (Ours)	LR($C = 0.1$)
QBC	LR($C = 1$), SVM(Linear, `probability = True`), SVM(RBF, `probability = True`), Linear Discriminant Analysis
ALBL	Combination of QSs with same $\mathcal{H}$: US-C, US-NC, HintSVM
VR	LR($C = 1$)
EER	SVM(RBF, `probability = True`)

number of positive labels

$$\text{IR} = \frac{|\{(x_i, y_i) : y_i = +1\}|}{|\{(x_j, y_j) : y_j = -1\}|}.$$

They employed these metrics to analyze the results from different aspects to explain the results of the query strategy's performance on a dataset. We agree with their core idea of the analysis methods and believe their benchmark benefits the research community. However, we observe that the conclusion of their work differs from several previous works. For example, Zhan et al. (2021) claimed that LAL performs better on binary datasets than Uncertainty Sampling while not in the other benchmark Yang & Loog (2018). The evidence urges us to re-implement the active learning benchmark to clarify the conflicting claims.

B.2 Benchmarking datasets

Section 3 records the datasets used in the previous benchmark Zhan et al. (2021). However, we discover that the attributes of datasets are different. We report the revision in Table 10 via 'Zhan et al. (2021) → Our new version'.

B.3 Failure of the Reproducing Uniform

Table 13 (Table 14) shows the significant difference between ours and Zhan et al. (2021). We noticed an implementation error in the previous benchmark. In Google, Uniform assumes that the data has already been shuffled.[6] However, the implementation in Zhan et al. (2021) does not shuffle the unlabeled pool at first.[7]

```
[Code=Python]
dict_data,labeled_data,test_data,unlabeled_data = \
    split_data(dataset_filepath, test_size, n_labeled)

print(unlabeled_data)
# results of indices of unlabeled pool
#[3, 4, 5, 10, 11, 13, 15, 16, 20, 23, 24, 26, 27, 29, 30, \
# 31, 33, 36, 37, 41, 43, 44, 45, 49, 50, 51, 53, 54, 55, \
# 57, 63, 64, 65, 70, 73, 75, 77, 78, 79, 83, 84, 86, 87, \
# 88, 89, 91, 92, 95, 97, 102, 105, 110, 112, 114, 115, \
# 121, 122, 127, 128, 131, 132, 136, 137, 139, 140, 144, \
```

[6] `https://github.com/google/active-learning/blob/master/sampling_methods/uniform_sampling.py#L40`

[7] `https://github.com/SineZHAN/ComparativeSurveyIJCAI2021PoolBasedAL/blob/master/Algorithm/baseline-google-binary.py#L331`

Table 10: Benchmarking datasets and revision of Table 2 in Zhan et al. (2021) Note. *Pol*, *Bioresponse*, and *Covertype* are expanded datasets which are not included in Zhan et al. (2021).

	Property	r	d	n
Appendicitis	Real-life	4	7	106
Sonar	Real-life	1	60	108→208
Parkinsons	Real-life	3.06	22	195
Ex8b	Synthetic	1	2	206→210
Heart	Real-life	1	13	270
Haberman	Real-life	2	3	306
Ionosphere	Real-life	1	34	351
Clean1	Real-life	1	168→166	475→476
Breast	Real-life	1	10	478
Wdbc	Real-life	1	30	569
Australian	Real-life	1	14	690
Diabetes	Real-life	1	8	768
Mammographic	Real-life	1	5	830
Ex8a	Synthetic	1	2	863→766
Tic	Real-life	6	9	958
German	Real-life	2	20→24	1000
Splice	Real-life	1	61→60	1000
Gcloudb	Synthetic	1	2	1000
Gcloudub	Synthetic	2→2.03	2	1000
Checkerboard	Synthetic	1→1.82	2	1600
Spambase	Real-life	1→1.54	57	4601
Banana	Synthetic	1	2	5300
Phoneme	Real-life	2	5	5404
Ringnorm	Real-life	1	21→20	7400
Twonorm	Real-life	1	50→20	7400
Phishing	Real-life	1	30	2456→11055
Pol	Real-life	1	26	10082
Bioresponse	Real-life	1	419	3434
Covertype	Real-life	1	10	566602

```
# 148, 150, 151, 155, 157, 159, 160, 161, 162, 164, 165, \
# 167, 168, 172, 175, 176, 177, 178, 181, 182, 183, 184, \
# 185, 186, 187, 188, 190, 191, 193, 194, 197, 198, 199, \
# 202, 203, 204, 205, 207, 208]
```

We also modify their Uniform implementation by `shuffle` the `unlabeled_data`. Then, we can obtain similar results based on their source code, see Table 11.

```
[Code=Python]
dict_data,labeled_data,test_data,unlabeled_data = \
    split_data(dataset_filepath, test_size, n_labeled)
random.shuffle(unlabeled_data)
```

The unshuffled implementation in Google significantly impacts binary classification datasets, such as *Sonar*, *Clean1*, and *Spambase*. Also, it affects *Ex8a* and *German*, which enlarges the difference AUBCs between Uniform and other query strategies. Due to this experience, we suggest practitioners ensure the correctness of the baseline method by comparing different implementations before conducting the benchmarking experiments.

Table 11: Comparing different train/test/labeled splits on *Sonar*: first column is reprot and reproducing results in Zhan et al. (2021), second column in our implementation, and the third column is reproducing results after we revise Zhan et al. (2021)'s code.

Uniform	Report and code in Zhan et al. (2021)	Our code	Modified code in Zhan et al. (2021)
Google	**0.6274***	0.7513	0.7577
libact	-	0.7520	0.7543
ALiPy	-	0.7556	0.7579

B.4 Query Strategy and Implementation

We revise some descriptions of the query strategies in Zhan et al. (2021):

(1) 'Graph Density (Graph) is a typical parallel-form combined strategy that balances the uncertainty and representative based measure simultaneously via a time-varying parameter Ebert et al. (2012).'

(2) 'Marginal Probability based Batch Mode AL (Margin) Chattopadhyay et al. (2012) selects a batch that makes the marginal probability of the new labeled set similar to the one of the unlabeled set via optimization by Maximum Mean Discrepancy (MMD).'

(3) 'Kremer et al. (2014) proposed an SVM-based AL strategy by minimizing the distances between data points and classification hyperplane (HintSVM).'

Issue (1): Although Ebert et al. (2012) proposed the reinforcement learning method to select uncertainty and diversity sample(s) during the procedure, Google Yilei "Dolee" Yang (2017) does not implement the whole procedure but only the diversity sampling method.[8] Thus, we should categorize it as **diversity-based** method.
Issue (2): Google Yilei "Dolee" Yang (2017) does not use MMD to measure the distance. The implementation is uncertainty sampling with a margin score is mentioned in the survey paper Settles (2012). Therefore, we should categorize it to **uncertainty-based** method.
Issue (3): libact Yang et al. (2017) implemented HintSVM based on the work of Li et al. (2015) rather than Kremer et al. (2014).

B.5 Relationship between query strategies

We provide additional evidence to explain the relationship between query strategies, which supports our experimental results.

(1) US-C and InfoDiv should be the same when the query batch size is one.

(2) Different uncertainty measurements should be the same in the binary classification, indicating that different uncertainty measurements do not cause differences between US-C and US-NC.

(3) SPAL changes the condition of variables used for discriminative and representative objective functions in BMDR.

Issue (1): InfoDiv clusters unlabeled samples into several clusters, then selects uncertain samples and keeps the same cluster distribution simultaneously.[9] Therefore, it is the same when we set the $B = 1$ to query the

[8] https://github.com/google/active-learning/blob/master/sampling_methods/graph_density.py

[9] https://github.com/google/active-learning/blob/master/sampling_methods/informative_diverse.py

most uncertain sample. Zhan et al. (2021) provided the different numbers of US-C and InfoDiv in Table 4, which might have resulted from using the different batch sizes of these query strategies.
Issue (2): The least confidence, margin, and entropy are monotonic functions with a peak equal to $\mathbb{P}(y = +1 \mid x) = 0.5$ in binary classification, such that all of these uncertainty measurements would query the same point Settles (2012).
Issue (3) The optimization problem in BMDR is

$$\begin{aligned}\min_{\alpha^\top 1_{|D_u|}=b, w} \; & \sum_{i=1}^{|D_l|}(y_i - w^\top\phi(x_i))^2 + \lambda\|w\|^2 \\ &+ \sum_{i=1}^{|D_u|}\alpha_i\left[\|w^\top\phi(x_j)\|_2^2 + 2|w^\top\phi(x_j)|\right] \\ &+ \beta(\alpha^\top K_1\alpha + k\alpha),\end{aligned} \tag{1}$$

where $\phi(x)$ is the feature mapping, λ is the hyper-parameter for the regularization term, β is the hyper-parameter for the diversity term, $1_{|D_u|}$ means ones vector with length of the unlabelled pool $|D_u|$. K_1 is defined as $K_1 = \frac{1}{2}K_{UU}$, where K_{UU} means the kernel matrix with sub-index U of unlabelled pool D_u. SPAL only changes $\alpha^\top 1_{|D_u|} = b$ to $\alpha^\top e_{|D_u|} = b$.[10]

B.6 Comparison between Zhan et al. (2021) and Yang & Loog (2018)

Yang & Loog (2018) propose the first benchmark for pool-based active learning for the conventional Logistic Regression model. The work compares 10 query strategies that could be categorized into **model uncertainty** and **hybrid criteria**. In datasets, they adopt 44 binary datasets and follow data pre-processing in Chang & Lin (2011). To compare performance across different query strategies, they also use an Area Under the Learning Curve with accuracy to show the average performance of a query strategy, named AUBC in Zhan et al. (2021). Furthermore, they compare the performance of each query strategy by average rank and improvement (win/tie/loss) from random sampling, which has the same purpose as our work (See Section 4.1 and Section 4.2).

C Re-benchmarking results of Zhan et al. (2021)

After we accomplish experiments under the settings in Appendix B.1, we obtain the benchmarking results for RBFSVM with the form (query strategy, dataset, seed, $|D_l|$, accuracy) for each round. A (random) seed corresponds to the different training sets, test sets, and initial label pool splits for a dataset. We collect the accuracy at each round ($|D_\mathrm{l}|$, accuracy) to plot a learning curve for query strategy on a dataset with a seed and summarize it as the mean AUBC in Table 12. Our re-benchmarking results show that Uncertainty Sampling with compatible models (US-C) outperforms the other query strategies on most datasets.

C.1 Statistical comparison of re-benchmarking results

We show our re-benchmarking results for RBFSVM side-by-side with Zhan et al. (2021)'s Table 3 in Table 13. To determine if there is a statistical difference between the two benchmarking results, we construct the confidence interval with the t-distribution of mean AUBCs. If a result in Zhan et al. (2021) falls outside the interval, their mean significantly differs from ours. We notice significant differences in Uniform on several datasets in Table 13. Therefore, we focus on comparing Uniform in Table 14, demonstrating our mean and standard deviation (SD) AUBC of Uniform and the mean AUBC of Uniform reported by Zhan et al. (2021). There are 13, nearly half of the datasets, significantly different from the existing benchmark with significance level $\alpha = 5\%$. Furthermore, we perform better on most datasets except for *Parkinsons* and *Mammographic*. 1% of mean AUBC is larger than previous work on 8 datasets, especially for *Sonar*, *Clean1*, and *Spambase*. Following the same procedure of statistical testing, Table 15 demonstrates **BSO** of ours and (Zhan et al.,

[10] https://github.com/NUAA-AL/ALiPy/blob/master/alipy/query_strategy/query_labels.py#L1469

Table 12: Mean AUBC of Query Strategies: We report query strategies with mean of repeated experiments.

	Uniform	US-C	US-NC	InfoDiv	QBC	EER	VR	Hier	Graph	Core-Set	HintSVM	QUIRE	DWUS	MCM	BMDR	SPAL	ALBL	LAL
Appendicitis	83.95%	84.54%	84.49%	84.54%	84.41%	84.26%	83.95%	84.14%	84.19%	83.98%	83.90%	83.99%	84.21%	84.57%	84.18%	84.15%	84.49%	84.33%
Sonar	75.00%	77.88%	76.54%	77.88%	76.75%	75.73%	75.00%	75.54%	75.58%	74.54%	74.06%	75.11%	74.35%	77.51%	75.98%	TLE	76.28%	76.76%
Parkinsons	83.05%	85.31%	85.11%	85.31%	84.49%	84.51%	83.05%	83.57%	82.91%	83.56%	81.78%	83.05%	82.74%	85.27%	83.69%	83.85%	84.61%	84.63%
Ex8b	88.53%	89.81%	89.39%	89.81%	89.38%	89.36%	88.53%	88.81%	88.50%	89.15%	86.95%	87.86%	88.42%	89.77%	88.84%	TLE	88.74%	89.42%
Heart	80.51%	81.57%	81.20%	81.57%	81.30%	80.85%	80.51%	80.75%	80.54%	81.05%	80.39%	81.03%	80.57%	81.54%	80.65%	80.97%	81.18%	81.24%
Haberman	73.09%	72.99%	73.02%	72.99%	73.05%	73.14%	73.08%	73.01%	73.04%	72.67%	72.62%	72.46%	73.16%	72.92%	73.23%	TLE	72.97%	73.11%
Ionosphere	91.80%	93.00%	91.97%	93.00%	92.78%	92.49%	91.80%	92.04%	91.62%	91.34%	89.64%	90.15%	87.93%	92.96%	89.34%	92.32%	92.06%	92.65%
Clean1	81.79%	84.32%	83.41%	84.32%	83.42%	82.15%	TLE	81.86%	81.00%	79.02%	76.97%	81.81%	81.79%	84.16%	TLE	TLE	82.64%	83.34%
Breast	96.14%	96.34%	96.26%	96.34%	96.33%	96.31%	95.82%	96.17%	96.15%	96.28%	96.24%	96.24%	96.06%	96.34%	96.18%	TLE	96.26%	95.86%
Wdbc	95.39%	96.52%	95.97%	96.52%	96.26%	96.22%	95.39%	95.65%	95.40%	95.86%	95.58%	95.83%	95.04%	96.50%	95.12%	95.72%	96.12%	96.13%
Australian	84.83%	85.04%	84.59%	85.04%	84.94%	84.72%	84.83%	84.87%	84.69%	84.78%	84.44%	84.76%	84.73%	85.04%	84.73%	85.04%	84.86%	84.83%
Diabetes	74.24%	74.79%	74.32%	74.79%	74.72%	74.57%	74.24%	74.34%	74.24%	74.91%	74.56%	74.70%	72.27%	74.71%	74.23%	74.65%	74.43%	74.62%
Mammographic	81.25%	81.64%	81.65%	81.64%	81.61%	81.58%	81.23%	81.40%	81.22%	81.48%	80.94%	81.42%	79.95%	81.68%	81.32%	TLE	81.59%	81.39%
Ex8a	85.52%	88.01%	82.83%	88.01%	86.16%	85.22%	85.52%	86.10%	85.13%	85.55%	81.34%	80.95%	79.24%	87.80%	85.39%	TLE	84.19%	83.54%
Tic	87.18%	87.20%	87.18%	87.20%	87.19%	87.19%	87.18%	87.19%	87.20%	87.16%	87.19%	86.99%	87.10%	87.20%	87.19%	87.12%	87.18%	87.20%
German	73.40%	74.17%	73.87%	74.17%	73.96%	73.80%	73.40%	73.48%	73.54%	73.62%	73.06%	73.55%	72.69%	74.09%	TLE	73.62%	73.66%	74.00%
Splice	80.68%	82.28%	81.47%	82.28%	81.50%	80.73%	80.68%	80.62%	78.21%	74.76%	77.57%	80.35%	76.08%	82.39%	TLE	TLE	81.00%	80.45%
Gcloudb	89.50%	89.85%	88.58%	89.85%	89.73%	89.47%	89.50%	89.49%	89.40%	89.25%	87.55%	87.85%	88.62%	89.82%	89.54%	TLE	89.72%	89.46%
Gcloudub	94.40%	95.67%	94.89%	95.67%	95.36%	93.87%	94.40%	94.75%	94.40%	89.12%	89.35%	93.17%	93.62%	95.57%	93.77%	TLE	93.83%	94.76%
Checkerboard	97.81%	98.47%	91.34%	98.47%	97.02%	98.40%	97.81%	97.85%	97.37%	98.74%	92.42%	94.37%	90.45%	98.47%	98.32%	TLE	96.79%	96.41%
Spambase	91.03%	92.05%	90.10%	92.05%	91.90%	TLE	TLE	91.22%	90.73%	90.52%	89.85%	TLE	91.03%	92.00%	TLE	TLE	91.62%	90.62%
Banana	89.26%	87.87%	80.50%	87.87%	89.08%	TLE	89.25%	89.29%	88.48%	89.30%	85.10%	82.99%	81.64%	87.54%	TLE	TLE	88.51%	89.23%
Phoneme	82.54%	83.55%	82.11%	83.55%	83.18%	TLE	TLE	83.00%	82.09%	82.40%	80.83%	81.83%	81.37%	83.59%	TLE	TLE	82.47%	82.42%
Ringnorm	97.76%	97.86%	97.67%	97.86%	97.71%	TLE	TLE	97.66%	97.11%	94.77%	97.15%	TLE	93.46%	97.82%	TLE	TLE	97.69%	97.80%
Twonorm	97.53%	97.64%	97.55%	97.64%	97.60%	TLE	TLE	97.52%	97.54%	97.55%	97.36%	TLE	97.31%	97.63%	TLE	TLE	97.52%	97.61%
Phishing	93.82%	94.60%	93.91%	94.60%	94.41%	TLE	TLE	93.80%	93.27%	94.06%	92.96%	TLE	89.23%	94.49%	TLE	TLE	94.20%	94.29%

Table 13: We report our AUBCs (%) with Table 3 in Zhan et al. (2021) side-by-side. A score denoted with format: `Zhan et al. (2021)` → `ours`. The symbol '*' indicates a significant difference with the significance level $\alpha = 5\%$.

	Uniform	BSO	Avg	BEST	BEST_QS	WORST	WORST_QS
Appendicitis	84% → 83.95%	88% → 88.37%	84% → 84.25%	86% → 84.57%*	EER → MCM	83% → 83.90%*	DWUS → HintSVM
Sonar	62% → 74.63%*	83% → 88.40%*	76% → 75.60%	78% → 77.62%*	LAL → US-C	73% → 73.57%	HintSVM →HintSVM
Parkinsons	84% → 83.05%*	87% → 88.28%*	85% → 83.97%	86% → 85.31%*	QBC → US-C	83% → 81.78%	HintSVM →HintSVM
Ex8b	87% → 88.53%*	92% → 93.76%*	89% → 88.88%	91% → 89.81%*	SPAL → US-C	86% → 86.99%	HintSVM →HintSVM
Heart	81% → 80.51%	85% → 89.30%*	79% → 80.99%	83% → 81.57%*	InfoDiv → US-C	72% → 80.39%*	DWUS → HintSVM
Haberman	73% → 73.08%	75% → 78.96%*	73% → 72.95%	74% → 73.19%	BMDR → BMDR	72% → 72.44%	QUIRE → QUIRE
Ionosphere	90% → 91.80%*	93% → 95.45%*	91% → 91.59%	93% → 93.00%*	LAL → US-C	88% → 87.93%*	HintSVM → DWUS
Clean1	65% → 81.83%*	87% → 92.19%*	81% → 81.97%	84% → 84.25%*	LAL → US-C	75% → 76.95%	HintSVM →HintSVM
Breast	95% → 96.16%*	96% → 97.60%*	96% → 96.19%	96% → 96.32%*	SPAL → US-C	95% → 95.82%*	DWUS → VR
Wdbc	95% → 95.39%	97% → 98.41%*	96% → 95.87%	97% → 96.52%*	LAL → US-C	94% → 95.04%*	EER → DWUS
Australian	85% → 84.83%	88% → 90.46%*	85% → 84.82%	85% → 85.04%*	Core-Set → US-C	82% → 84.44%*	DWUS → HintSVM
Diabetes	74% → 74.24%*	78% → 82.57%*	74% → 74.42%	75% → 74.91%	Core-Set →Core-Set	69% → 72.27%*	EER → DWUS
Mammographic	82% → 81.30%*	84% → 85.03%*	82% → 81.44%	83% → 81.78%	MCM → MCM	80% → 79.99%*	EER → DWUS
Ex8a	84% → 85.39%*	87% → 88.28%*	84% → 84.62%	86% → 87.88%*	Hier → US-C	80% → 79.11%*	QUIRE → DWUS
Tic	87% → 87.18%	87% → 90.77%*	87% → 87.17%	87% → 87.20%*	EER → US-C	87% → 86.99%	QUIRE → QUIRE
German	73% → 73.40%*	78% → 82.08%*	74% → 73.65%	74% → 74.17%*	QBC → US-C	72% → 72.68%	DWUS → DWUS
Splice	81% → 80.75%	87% → 91.02%*	79% → 80.08%	82% → 82.34%*	QBC → MCM	68% → 75.18%*	EER → Core-Set
Gcloudb	89% → 89.52%	90% → 90.91%*	89% → 89.20%	90% → 89.81%*	Graph → US-C	87% → 87.48%	HintSVM →HintSVM
Gcloudub	94% → 94.37%	96% → 96.83%*	93% → 93.72%	95% → 95.60%*	QBC → US-C	86% → 89.29%*	EER → Core-Set
Checkerboard	98% → 97.81%	99% → 99.72%*	94% → 96.42%	99% → 98.74%	Core-Set →Core-Set	90% → 90.45%*	VR → DWUS
Spambase	69% → 91.03%*	- → -	88% → 91.14%	92% → 92.05%*	QBC → US-C	69% → 89.85%*	DWUS → HintSVM
Banana	90% → 89.26%	- → -	85% → 86.90%	89% → 89.30%*	Hier → Core-Set	78% → 80.50%*	QUIRE → US-NC
Phoneme	82% → 82.54%	- → -	82% → 82.49%	83% → 83.59%*	QBC → MCM	80% → 80.83%	HintSVM →HintSVM
Ringnorm	98% → 97.76%*	- → -	95% → 97.05%	98% → 97.86%*	LAL → US-C	80% → 93.46%	DWUS → DWUS
Twonorm	98% → 97.53%	- → -	98% → 97.54%	98% → 97.64%*	Core-Set → US-C	97% → 97.31%	DWUS → DWUS
Phishing	93% → 93.82%*	- → -	94% → 93.65%	95% → 94.60%*	LAL → US-C	92% → 89.23%*	Graph → DWUS

2021). This phenomenon is more evident in BSO than in Uniform. We still get significantly different and better performances on most datasets except for *Appendicitis*.

C.2 Verify usefulness

Zhan et al. (2021) verified the applicability of a query strategy by several aspects:

- *Low/high dimension view* (*LD* for $d < 50$, *HD* for $d \geq 50$),
- *Data scale view* (*SS* for $n < 1000$, *LS* for $n \geq 1000$),

Table 14: Reporducing Failure of **Uniform**

	Mean	SD	Zhan et al. (2021)	$\alpha = 5\%$	$\alpha = 1\%$
Appendicitis	83.95%	3.63%	83.6%	In	In
Sonar	74.63%	3.79%	61.7%	Out	Out
Parkinsons	83.05%	3.68%	84.0%	Out	In
Ex8b	88.53%	2.80%	86.6%	Out	Out
Heart	80.51%	2.79%	80.8%	In	In
Haberman	73.08%	2.70%	72.7%	In	In
Ionosphere	91.80%	1.78%	90.1%	Out	Out
Clean1	81.83%	1.94%	64.9%	Out	Out
Breast	96.16%	0.90%	95.4%	Out	Out
Wdbc	95.39%	1.30%	95.2%	In	In
Australian	84.83%	1.58%	84.6%	In	In
Diabetes	74.24%	1.52%	73.6%	Out	Out
Mammographic	81.30%	1.98%	81.9%	Out	Out
Ex8a	85.39%	2.17%	83.8%	Out	Out
Tic	87.18%	1.53%	87.0%	In	In
German	73.40%	1.73%	72.6%	Out	Out
Splice	80.75%	1.61%	80.6%	In	In
Gcloudb	89.52%	1.17%	89.3%	In	In
Gcloudub	94.37%	0.96%	94.2%	In	In
Checkerboard	97.81%	0.59%	97.8%	In	In
Spambase	91.03%	0.57%	68.5%	Out	Out
Banana	89.26%	0.38%	89.5%	In	In
Phoneme	82.54%	1.01%	82.2%	In	In
Ringnorm	97.76%	0.21%	97.6%	Out	In
Twonorm	97.53%	0.19%	97.6%	In	In
Phishing	93.82%	0.48%	92.6%	Out	Out

- *Data balance/imbalance view* (*BAL* for $\gamma < 1.5$, *IMB* for $\gamma \geq 1.5$).

They compare these aspects with a score

$$\delta_{q,s} = \max\{\overline{\text{AUBC}_{\text{BSO},s}}, \overline{\text{AUBC}_{\text{US},s}}, \dots, \overline{\text{AUBC}_{\text{LAL},s}}\} - \overline{\text{AUBC}_{q,s}},$$

Specifically, they grouped $\delta_{q,s}$ by different aspects to generate the metric for the report

$$\bar{\delta}_{q,v} = \frac{\sum_{s \in v} \delta_{q,s}}{|\{s \in v\}|},$$

where v is one of a dataset's dimension, scale, or class-balance views. We re-benchmark results and denote the rank of the query strategy with a superscript in Table 16. Table 16 shows that the US-C (InfoDiv) and MCM occupy the first and second ranks in different aspects, and the QBC keeps the third rank. The results are unlike those of Zhan et al. (2021) except for the QBC performance well on both of us. We explain the reason for the same performance of US-C and InfoDiv in Appendix B.5.

Using score $\bar{\delta}_{q,v}$ to ascertain the applicability of several query strategies is straightforward. However, it could bring an issue: BSO outperforms query strategies significantly on most datasets in our benchmarking results. We cannot exclude those remaining large-scale datasets without BSO, i.e., $n > 1000$, having the same pattern, such that their results could impact different aspects. Therefore, we replace $\bar{\delta}_{q,v}$ with the improvement of query strategy q over Uniform, i.e., $\tau_{q,s,k}$ in Section 4.2, because Uniform is the baseline and most efficient across all experiments, which is essential to complete.

Table 15: Reporducing Failure of **BSO**

	Mean	SD	Zhan et al. (2021)	$\alpha = 5\%$	$\alpha = 1\%$
Appendicitis	88.37%	2.95%	88.1%	In	In
Sonar	88.40%	2.84%	83.0%	Out	Out
Parkinsons	88.28%	3.19%	86.5%	Out	Out
Ex8b	93.76%	1.82%	92.4%	Out	Out
Heart	89.30%	2.47%	84.8%	Out	Out
Haberman	78.96%	3.05%	75.1%	Out	Out
Ionosphere	95.45%	1.42%	93.3%	Out	Out
Clean1	92.19%	1.69%	87.1%	Out	Out
Breast	97.60%	0.67%	96.1%	Out	Out
Wdbc	98.41%	0.65%	97.3%	Out	Out
Australian	90.46%	1.48%	87.8%	Out	Out
Diabetes	82.57%	1.70%	78.4%	Out	Out
Mammographic	85.03%	1.97%	84.4%	Out	Out
Ex8a	88.28%	2.03%	87.3%	Out	Out
Tic	90.77%	2.27%	87.3%	Out	Out
German	82.08%	2.01%	78.3%	Out	Out
Splice	91.02%	1.18%	87.1%	Out	Out
Gcloudb	90.91%	1.09%	90.1%	Out	Out
Gcloudub	96.83%	0.78%	96.3%	Out	Out
Checkerboard	99.72%	0.36%	99.2%	Out	Out

Table 16: Verifying Applicability with δ_i

	B	LD	HD	SS	LS	BAL	IMB
US-NC	4.77	4.16	8.12	5.36	3.96	5.09	4.39
QBC	3.83^3	3.15^3	7.57^3	5.02^3	2.20^3	4.05^3	3.57^3
HintSVM	5.91	4.92	11.37	6.77	4.73	6.25	5.51
QUIRE	5.96	5.08	11.54	6.13	5.60	6.94	4.98
ALBL	4.20	3.49	8.06	5.37	2.59	4.45	3.90
DWUS	6.20	5.46	10.24	6.71	5.50	6.83	5.46
VR	5.04	4.26	12.02	5.43	4.13	5.36	4.72
Core-Set	4.92	3.78	11.20	5.79	3.72	5.35	4.42
US-C	3.50^1	2.89^1	6.86^1	4.62^1	1.97^1	3.72^1	3.24^1
Graph	4.62	3.72	9.58	5.77	3.05	4.98	4.20
Hier	4.22	3.41	8.69	5.53	2.43	4.49	3.90
InfoDiv	3.50^1	2.89^1	6.86^1	4.62^1	1.97^1	3.72^1	3.24^1
MCM	3.56^2	2.94^2	6.98^2	4.68^2	2.03^2	3.80^2	3.27^2
EER	5.21	4.18	11.09	5.33	4.86	6.13	4.30
BMDR	5.61	4.57	11.50	5.77	5.11	6.33	4.89
SPAL	5.90	4.69	12.32	5.67	6.77	6.56	5.17
LAL	4.14	3.41	8.14	5.27	2.59	4.37	3.86

The other issue is heuristically grouping the views into a binary category and averaging the performance with the same views $\bar{\delta}_{q,v}$ without reporting SDs. These analysis methods may be biased when the properties of datasets are not balanced. To address this issue, we plot a matrix of scatter plots that directly demonstrates the improvement of US-C for each property on all datasets with different colors. Figure 7 shows a low correlation ($|r| < 0.4$) and no apparent patterns between properties and the improvement of US-C, indicating that Our analysis results do not support the claims of 'Method aspects' in the existing benchmark Zhan et al. (2021), either. In conclusion, we want to emphasize that **revealing the analysis methods is as important as the experimental settings** because the analysis method employed will influence the conclusion.

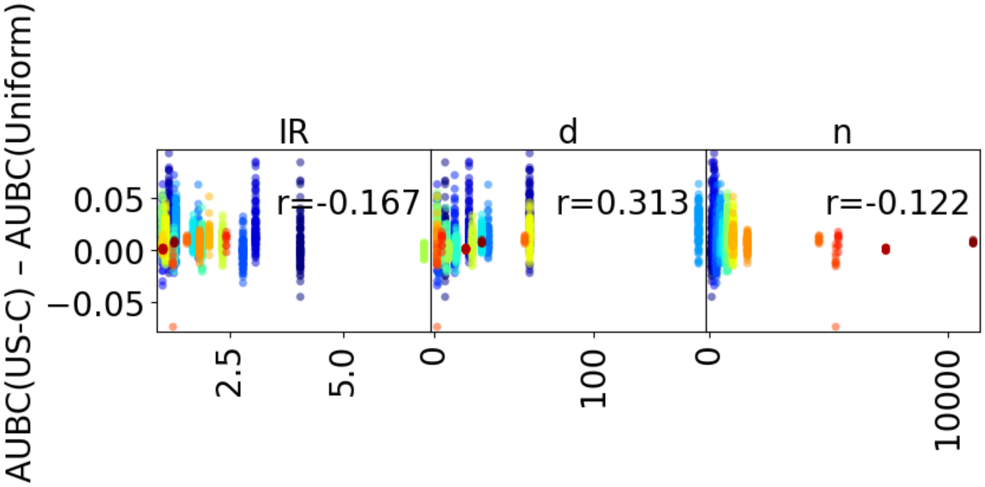


Figure 7: A matrix of scatter plots of the improvement of US-C

C.3 More on analysis of non-compatible models for uncertainty sampling

Section 5.1 demonstrates results involving different combinations of query-oriented and task-oriented models on *Checkerboard* and *Gcloudb* datasets. We reveal more datasets from Figure 8 to Figure 12. These results still hold for the compatible models for uncertainty sampling outperform non-compatible ones on most datasets, i.e., the diagonal entries of the heatmap are larger than non-diagonal entries. Figure 13 demonstrates that non-compatible models achieve slightly better performance than compatible models. When query-oriented and task-oriented models are heterogeneous, we conjecture that it could improve uncertainty sampling by exploring more diverse examples like the hybrid criteria approach Settles (2012); Sinha et al. (2019).

D Benchmarking results of Random Forest

This section follows the analysis procedure in Section 4 to benchmark Random Forest (RF) on the same datasets. The analysis results are listed as follows:

1. Verify the superiority by comparing the mean AUBC of query strategies in Table 17.
2. Verify the superiority by comparing the average accuracy of the model with 20% of the total budget in Table 18.
3. Verify the superiority by comparing the average ranking of query strategies in Table 19.
4. Verify the usefulness by comparing the data utilization rate of query strategies in Table 20.

These results are consistent with the previous benchmarking results in Section 4 and Appendix C. We conclude that the uncertainty sampling with compatible RF models gains superiority and usefulness for tabular datasets.

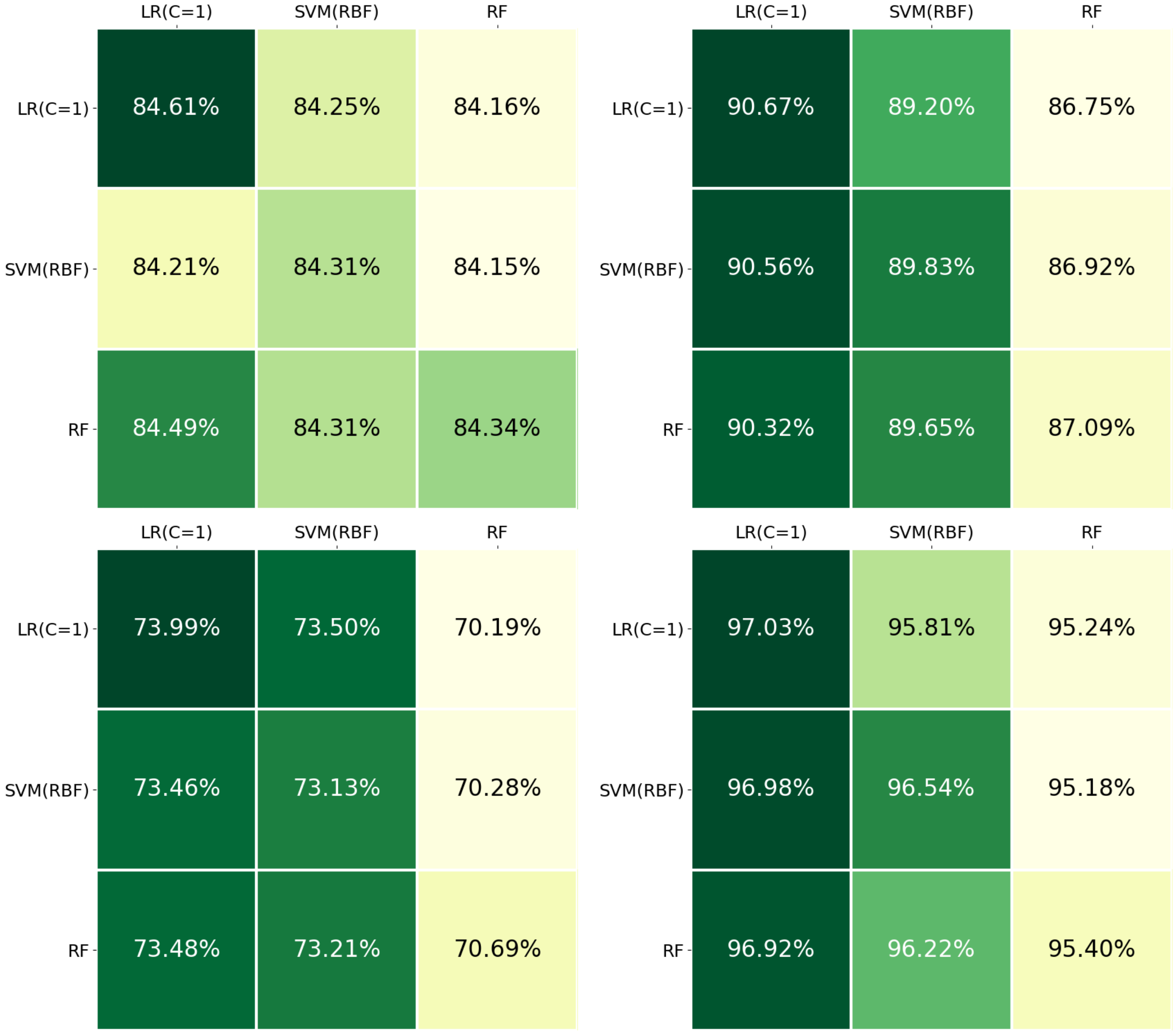


Figure 8: Mean AUBC of query-oriented model and task-oriented model on group 1. (Compatible LRs achieve best results.): *Appendicitis* (top-left), *Ex8b* (top-right), *Haberman* (bottom-left), and *Wdbc* (bottom-right).

E Computational resource

We test the time of an experiment for query strategy running on a dataset. Our resource is: *DELL PowerEdge R730* with CPU *Intel Xeon E5-2640 v3 @2.6GHz * 2* and memory *192 GB*. The results are reported in the supplementary material with path `active-learning-benchmark/results/rbfsvm-exps_rebenchmarkZhan2021/speedtest/README.md`. Note that this work does not optimize libact, Google, and ALiPy performance. If practitioners discover inefficient implementation, please contact us by mail or leave issues on GitHub.

F Limitations, related benchmarks, and future works

While we intentionally constrain our benchmark's scope to maintain fairness and reproducibility, this focus might give the impression of limitations. It is worth noting that prior active learning benchmarks focus on assessing query strategies within the context of advanced deep learning models, especially in image classification and visual question answering Beck et al. (2021); Karamcheti et al. (2021); Zhan et al. (2022).

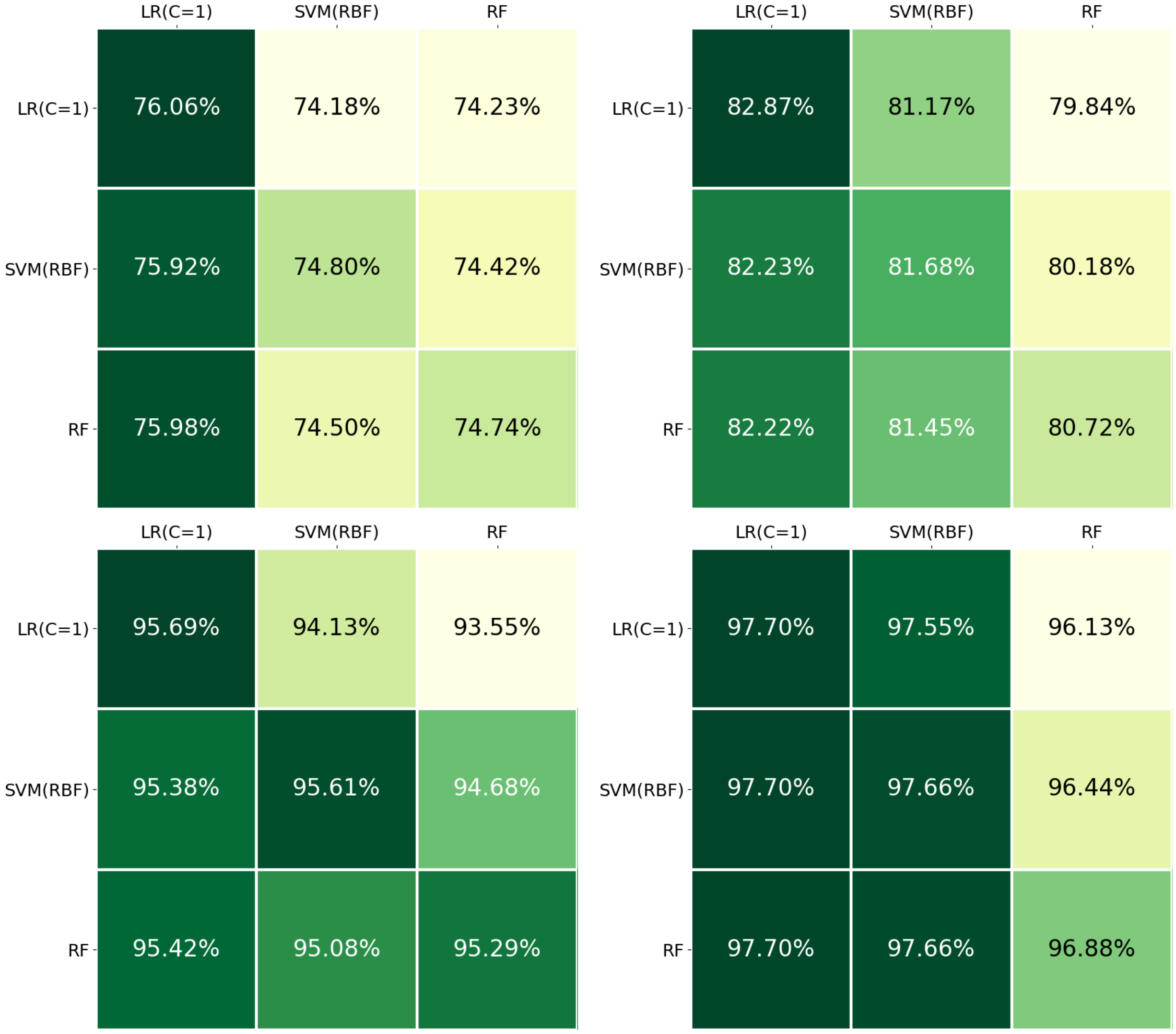


Figure 9: Mean AUBC of query-oriented model and task-oriented model on group 1. (Compatible LRs achieve best results.): *Diabetes* (top-left), *Mammographic* (top-right), *Gcloudub* (bottom-left), and *Twonorm* (bottom-right).

We encourage practitioners to explore active learning techniques in broader tasks and domains. For example, ample room exists to investigate active learning's applicability in areas like regression problems, object detection, and natural language processing Cai et al. (2016); Wu et al. (2019); Zhang et al. (2020); Yuan et al. (2021); Brust et al. (2018); Zhang et al. (2022).

Evaluating the performance of query strategy is a challenge in benchmarking. Kottke et al. (2017) and Trittenbach et al. (2021) propose metrics such as *Deficiency score*, *Data Utilization Rate*, *Start Quality*, and *Average End Quality* to summarize the performance of a query strategy from learning curves. Our implementation saves querying results at each round, enabling thorough analysis without costly re-runs, which empowers researchers to develop novel metrics and methods, driving advancements in active learning assessment.

The stability of experimental results is another challenge to a fair comparison. Studies by Ji et al. (2023); Lüth et al. (2023); Munjal et al. (2022) have revealed variations in performance metrics stemming from different query strategies, causing inconsistent results and claims in previous research. They suggest standardizing

Table 17: Benchmarking results of Random Forest. The numbers are mean AUBC (↑, %). We report the baseline method (Uniform), the best query strategy with its mean AUBC (BEST_QS, BEST), and the worst query strategy with its mean AUBC (WORST_QS, WORST) across datasets in Table 17.

	Uniform	BEST_QS	BEST	WORST_QS	WORST
Appendicitis	83.70%	US	84.48%	DWUS	83.12%
Sonar	75.66%	US	77.31%	HintSVM	74.75%
Parkinsons	84.31%	US	86.61%	HintSVM	82.79%
Ex8b	85.97%	US	87.07%	HintSVM	84.67%
Heart	80.19%	DWUS	80.93%	HintSVM	79.64%
Haberman	69.56%	US	70.61%	QUIRE	68.91%
Ionosphere	90.98%	BALD	92.08%	HintSVM	87.22%
Clean1	79.15%	BALD	82.09%	HintSVM	75.18%
Breast	96.42%	US	96.82%	DWUS	95.44%
Wdbc	94.29%	LAL	95.32%	HintSVM	93.92%
Australian	85.77%	US	86.20%	DWUS	85.55%
Diabetes	74.60%	LAL	74.97%	DWUS	73.59%
Mammographic	79.36%	LAL	80.82%	DWUS	78.82%
Ex8a	93.09%	BALD	95.50%	HintSVM	87.65%
Tic	86.36%	Core-Set	86.43%	DWUS	85.48%
German	74.02%	US	74.74%	DWUS	72.86%
Splice	90.49%	MCM	91.52%	Core-Set	84.17%
Gcloudb	88.33%	LAL	88.96%	QUIRE	86.50%
Gcloudub	93.83%	BALD	95.34%	HintSVM	87.38%
Checkerboard	99.24%	LAL	99.67%	DWUS	95.00%
Spambase	93.54%	BALD	94.74%	HintSVM	92.11%
Banana	88.25%	LAL	88.82%	DWUS	81.54%
Phoneme	86.63%	BALD	88.81%	HintSVM	84.75%
Ringnorm	94.15%	US	95.66%	Core-Set	70.55%
Twonorm	96.60%	BALD	96.88%	HintSVM	94.78%
Phishing	95.61%	US	96.68%	HintSVM	94.24%
Covertype	76.47%	US	79.20%	Uniform	76.47%
Bioresponse	73.57%	US	74.83%	Uniform	73.57%
Pol	96.58%	US	97.85%	Uniform	96.58%

experimental settings like data augmentation, neural network structures, and optimizers to address this. These findings emphasize the sensitivity of active learning algorithms to experimental settings, a critical consideration for future work.

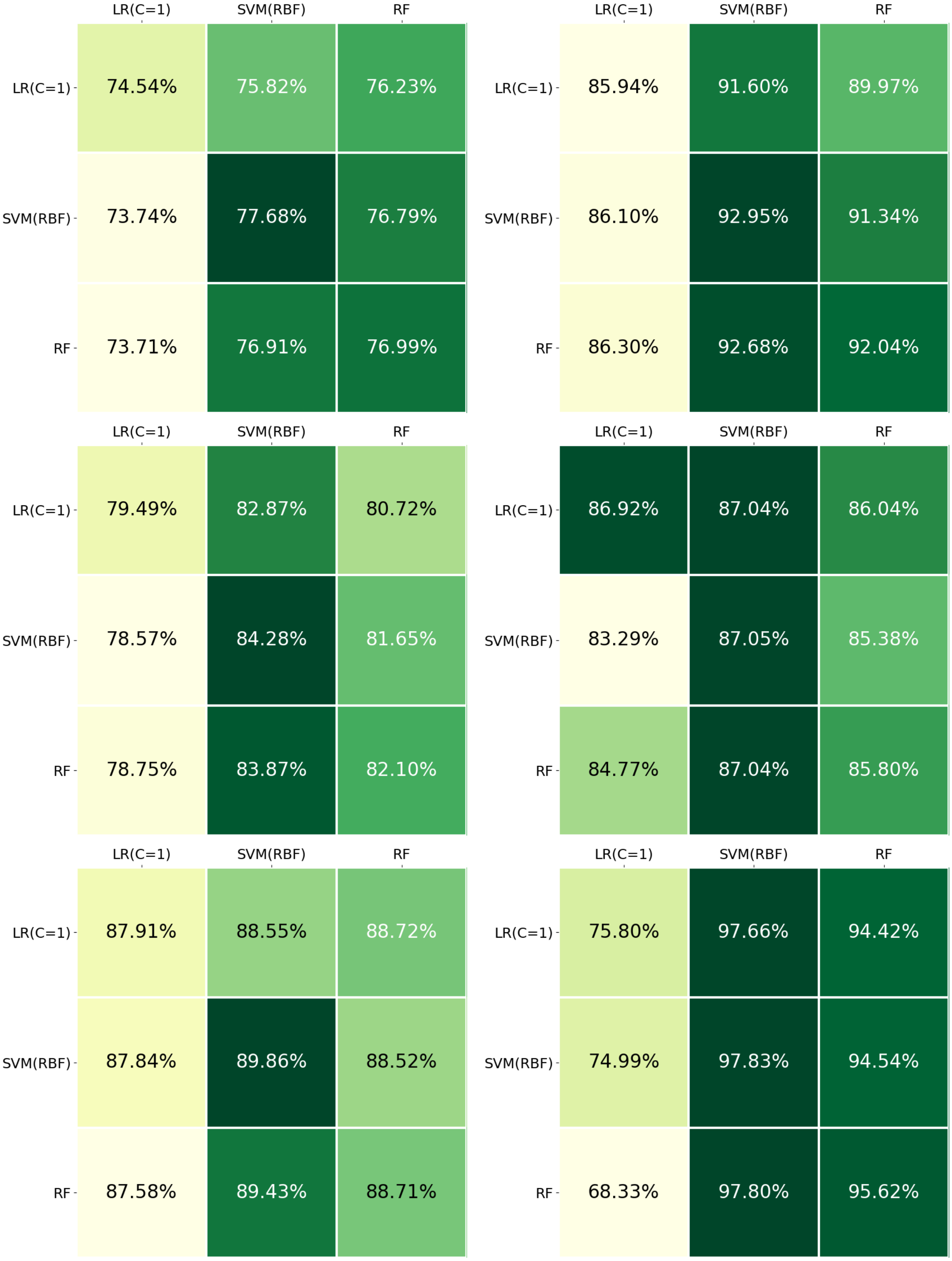
Previous benchmarks show that query strategies may not outperform Uniform in specific settings or tasks Yang & Loog (2018); Desreumaux & Lemaire (2020); Karamcheti et al. (2021); Munjal et al. (2022). Our findings, demonstrated in Table 6, also indicate that uncertainty sampling does not excel on datasets like *Checkerboard* and *Banana*. Several works study possible reasons for the failure of *uncertainty sampling* (Mussmann & Liang, 2018; Karamcheti et al., 2021; Tifrea et al., 2022) to realize the applicability of active learning algorithms. It underscores the need to explore robust baselines for pool-based active learning, particularly in real-world scenarios Lu et al. (2023).

Table 18: Accuracy of the model with 20% labeled examples: We report the accuracy of the model with 20% labeled examples on each dataset. The scores with **bold** mean the best performance on a dataset. 'TLE' means a query strategy exceeds the time limit.

	Uniform	US	QBC	BALD	Hier	Graph	Core-Set	HintSVM	QUIRE	DWUS	MCM	BMDR	ALBL	LAL
Sonar	68.98%	69.99%	69.96%	70.45%	69.11%	70.77%	68.57%	69.40%	70.11%	70.15%	69.81%	**70.92%**	70.33%	70.17%
Parkinsons	81.04%	81.94%	80.87%	**82.21%**	81.56%	81.04%	80.47%	80.54%	80.53%	81.05%	81.97%	81.41%	81.24%	81.94%
Ex8b	82.44%	83.40%	82.45%	83.62%	82.56%	82.04%	**83.83%**	81.85%	83.04%	82.61%	83.20%	83.43%	83.18%	83.33%
Heart	78.37%	**79.42%**	77.80%	78.97%	78.89%	79.08%	78.81%	77.40%	78.78%	79.05%	79.29%	79.40%	78.54%	78.83%
Haberman	70.11%	**71.63%**	70.24%	71.58%	70.67%	70.79%	69.34%	69.57%	68.76%	71.47%	71.41%	70.49%	70.39%	71.28%
Ionosphere	88.65%	91.34%	88.42%	**91.91%**	89.13%	88.50%	83.14%	82.30%	80.99%	84.30%	91.54%	TLE	88.73%	90.87%
Clean1	71.01%	72.94%	70.82%	**73.42%**	71.25%	71.27%	66.95%	66.69%	71.31%	70.98%	73.32%	68.87%	72.34%	72.54%
Breast	96.35%	97.18%	96.51%	97.14%	96.66%	96.36%	95.70%	96.36%	95.93%	95.20%	97.22%	96.31%	96.89%	**97.23%**
Wdbc	93.56%	96.18%	93.35%	**96.27%**	93.55%	93.62%	93.12%	92.69%	93.28%	93.46%	96.07%	93.85%	94.98%	96.18%
Australian	84.97%	**86.17%**	85.25%	86.09%	85.42%	85.34%	85.01%	84.69%	85.39%	84.45%	85.72%	85.00%	85.47%	85.98%
Diabetes	73.84%	74.13%	74.07%	**74.39%**	73.96%	73.86%	73.85%	73.07%	73.71%	72.36%	74.24%	73.50%	73.68%	74.33%
Mammographic	79.74%	**82.46%**	79.75%	82.23%	80.46%	79.61%	81.19%	80.97%	82.44%	80.13%	82.39%	79.95%	80.67%	82.22%
Ex8a	89.91%	**95.05%**	89.58%	95.02%	89.86%	91.50%	93.15%	77.99%	82.09%	80.45%	94.51%	91.23%	87.17%	94.46%
Tic	86.92%	86.92%	86.84%	86.96%	86.58%	**87.02%**	86.93%	86.55%	86.86%	84.93%	86.94%	86.63%	86.98%	TLE
German	72.86%	73.62%	72.64%	73.46%	72.66%	72.89%	72.75%	72.53%	72.67%	70.20%	73.41%	72.57%	73.22%	**73.72%**
Splice	87.65%	**88.77%**	87.41%	88.57%	87.76%	87.17%	70.04%	78.95%	87.15%	78.31%	88.76%	87.25%	87.53%	86.36%
Gcloudb	88.27%	89.25%	88.52%	89.14%	88.45%	88.69%	88.52%	84.99%	85.68%	86.68%	89.29%	88.48%	89.25%	**89.58%**
Gcloudub	92.02%	95.31%	92.24%	**95.68%**	93.05%	93.95%	90.16%	84.09%	86.48%	83.41%	95.47%	91.53%	89.65%	93.90%
Checkerboard	99.28%	99.14%	99.32%	99.13%	99.60%	99.58%	99.41%	93.15%	93.09%	93.32%	99.69%	99.48%	99.10%	**99.88%**
Spambase	93.09%	95.32%	92.93%	95.31%	92.98%	92.92%	92.17%	90.28%	91.81%	93.20%	**95.39%**	92.21%	92.80%	95.08%
Banana	87.97%	89.28%	88.00%	**89.30%**	87.92%	88.04%	88.43%	75.08%	74.97%	77.28%	89.25%	88.23%	87.27%	89.16%
Phoneme	84.44%	88.01%	84.41%	**88.30%**	85.67%	84.77%	85.51%	81.39%	83.14%	82.61%	87.89%	85.12%	84.46%	87.55%
Ringnorm	93.73%	**96.91%**	93.90%	96.80%	94.33%	92.80%	50.68%	56.12%	50.68%	60.49%	96.86%	74.30%	92.83%	90.30%
Twonorm	96.47%	96.83%	96.52%	**96.89%**	95.30%	96.60%	96.46%	93.07%	91.79%	96.15%	96.77%	TLE	96.07%	96.85%
Phishing	94.83%	**96.85%**	94.50%	96.72%	94.66%	94.61%	94.56%	92.91%	91.25%	92.79%	96.80%	TLE	95.57%	96.83%
Covertype	74.78%	76.91%	TLE	**76.97%**	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE
Bioresponse	70.63%	**73.03%**	TLE	72.64%	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE
Pol	95.94%	**98.23%**	TLE	98.22%	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE

Table 19: Average Ranking of Query Strategies: We report query strategies with the best average ranking. The scores with ^{1}, ^{2}, or ^{3} mean the 1st, 2nd and 3rd performance on a dataset. 'TLE' means a query strategy exceeds the time limit.

	US	QBC	BALD	Hier	Graph	Core-Set	HintSVM	QUIRE	DWUS	MCM	BMDR	ALBL	LAL
Appendicitis	4.81^1	7.76	5.44^2	7.30	7.74	7.31	8.43	7.89	8.85	5.63	7.81	6.47	5.56^3
Sonar	3.69^1	6.95	3.97^3	7.86	7.77	8.15	8.87	7.71	7.83	3.96^2	TLE	6.52	4.72
Parkinsons	2.97^1	8.55	3.08^2	6.81	10.06	9.53	11.61	9.34	7.17	3.72^3	7.77	6.66	3.73
Ex8b	4.50^2	8.08	4.37^1	7.56	8.60	6.04	10.10	8.26	8.90	5.13^3	7.28	6.82	5.36
Heart	6.06^3	7.49	6.26	7.46	8.40	7.86	9.42	8.24	4.90^1	5.74^2	6.34	6.63	6.20
Haberman	5.07^1	7.22	5.33	7.42	7.26	8.38	9.35	9.71	5.25^2	5.29^3	7.10	7.90	5.72
Ionosphere	2.43^2	6.53	2.40^1	6.29	7.27	9.96	11.50	11.55	11.25	2.90^3	8.03	7.17	3.72
Clean1	2.69^2	7.76	2.44^1	7.40	9.25	9.49	11.60	7.15	7.86	2.96^3	TLE	5.58	3.82
Breast	3.44^1	8.35	3.64^3	6.75	9.60	8.36	7.66	9.05	12.24	3.60^2	8.54	5.94	3.83
Wdbc	3.05^3	9.30	2.80^1	8.41	9.53	9.58	10.67	9.42	9.31	3.32	8.19	4.49	2.93^2
Australian	4.77^2	7.48	4.58^1	6.78	8.50	8.11	8.21	7.25	8.89	5.69	8.26	6.82	5.66^3
Diabetes	5.53^2	6.75	5.77^3	7.22	6.83	6.30	8.27	7.78	10.86	6.03	6.86	7.75	5.05^1
Mammographic	3.92^3	9.62	3.73^2	7.66	8.23	9.06	6.96	6.74	10.10	3.92	9.11	8.54	3.41^1
Ex8a	2.37^1	7.93	2.39^2	7.27	7.95	4.10	12.14	11.78	11.86	3.14^3	6.43	9.83	3.81
Tic	7.49	4.53^2	7.66	6.58	5.02	3.89^1	6.15	7.01	9.01	7.80	TLE	4.82^3	8.04
German	3.80^1	7.89	4.19^2	8.44	7.55	7.07	8.71	7.55	12.52	4.34^3	8.21	5.67	5.06
Splice	2.72^1	6.46	2.81^3	6.15	9.43	11.59	9.86	6.20	10.77	2.72^2	TLE	5.23	4.06
Gcloudb	4.81	7.34	4.76^3	7.65	8.25	6.73	11.28	10.51	10.25	4.90	6.76	4.53^2	3.23^1
Gcloudub	2.47^2	7.34	2.46^1	5.49	6.63	8.48	12.90	11.21	11.56	2.87^3	7.48	7.80	4.31
Checkerboard	3.00^2	7.67	3.25	6.47	7.49	7.20	12.18	11.67	11.81	3.15^3	5.83	8.38	2.90^1
Spambase	2.40^2	7.50	1.60^1	5.50	9.10	9.30	11.00	TLE	7.60	2.60^3	TLE	6.00	3.40
BaTLEa	3.10^3	6.90	2.90^2	7.00	9.60	5.60	11.60	12.00	12.40	3.30	6.10	8.30	2.20^1
Phoneme	2.40^3	8.60	1.90^1	5.00	8.60	6.50	11.60	9.60	11.30	2.20^2	TLE	6.80	3.50
Ringnorm	1.40^1	6.00	1.80^2	4.60	8.00	11.70	9.30	11.30	9.70	3.10^3	TLE	5.90	5.20
Twonorm	1.90^2	6.20	1.80^1	9.50	5.50	4.80	10.90	TLE	8.00	2.30^3	TLE	8.80	6.30
Phishing	1.50^1	7.00	2.30^3	5.70	7.50	5.80	9.80	TLE	9.20	2.20^2	TLE	4.00	TLE
Covertype	1.40^1	TLE	1.60^2	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE
Bioresponse	1.40^1	TLE	1.60^2	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE
Pol	1.30^1	TLE	1.70^2	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE



	LR(C=1)	SVM(RBF)	RF
LR(C=1)	74.54%	75.82%	76.23%
SVM(RBF)	73.74%	77.68%	76.79%
RF	73.71%	76.91%	76.99%

	LR(C=1)	SVM(RBF)	RF
LR(C=1)	85.94%	91.60%	89.97%
SVM(RBF)	86.10%	92.95%	91.34%
RF	86.30%	92.68%	92.04%

	LR(C=1)	SVM(RBF)	RF
LR(C=1)	79.49%	82.87%	80.72%
SVM(RBF)	78.57%	84.28%	81.65%
RF	78.75%	83.87%	82.10%

	LR(C=1)	SVM(RBF)	RF
LR(C=1)	86.92%	87.04%	86.04%
SVM(RBF)	83.29%	87.05%	85.38%
RF	84.77%	87.04%	85.80%

	LR(C=1)	SVM(RBF)	RF
LR(C=1)	87.91%	88.55%	88.72%
SVM(RBF)	87.84%	89.86%	88.52%
RF	87.58%	89.43%	88.71%

	LR(C=1)	SVM(RBF)	RF
LR(C=1)	75.80%	97.66%	94.42%
SVM(RBF)	74.99%	97.83%	94.54%
RF	68.33%	97.80%	95.62%

Figure 10: Mean AUBC of query-oriented model and task-oriented model on group 2. (Compatible SVMs achieve best results.): *Sonar* (top-left), *Ionosphere* (top-right), *Clean1* (middle-left), *Tic* (middle-left), *Gcloudb* (bottom-left), and *Ringnorm* (bottom-right).

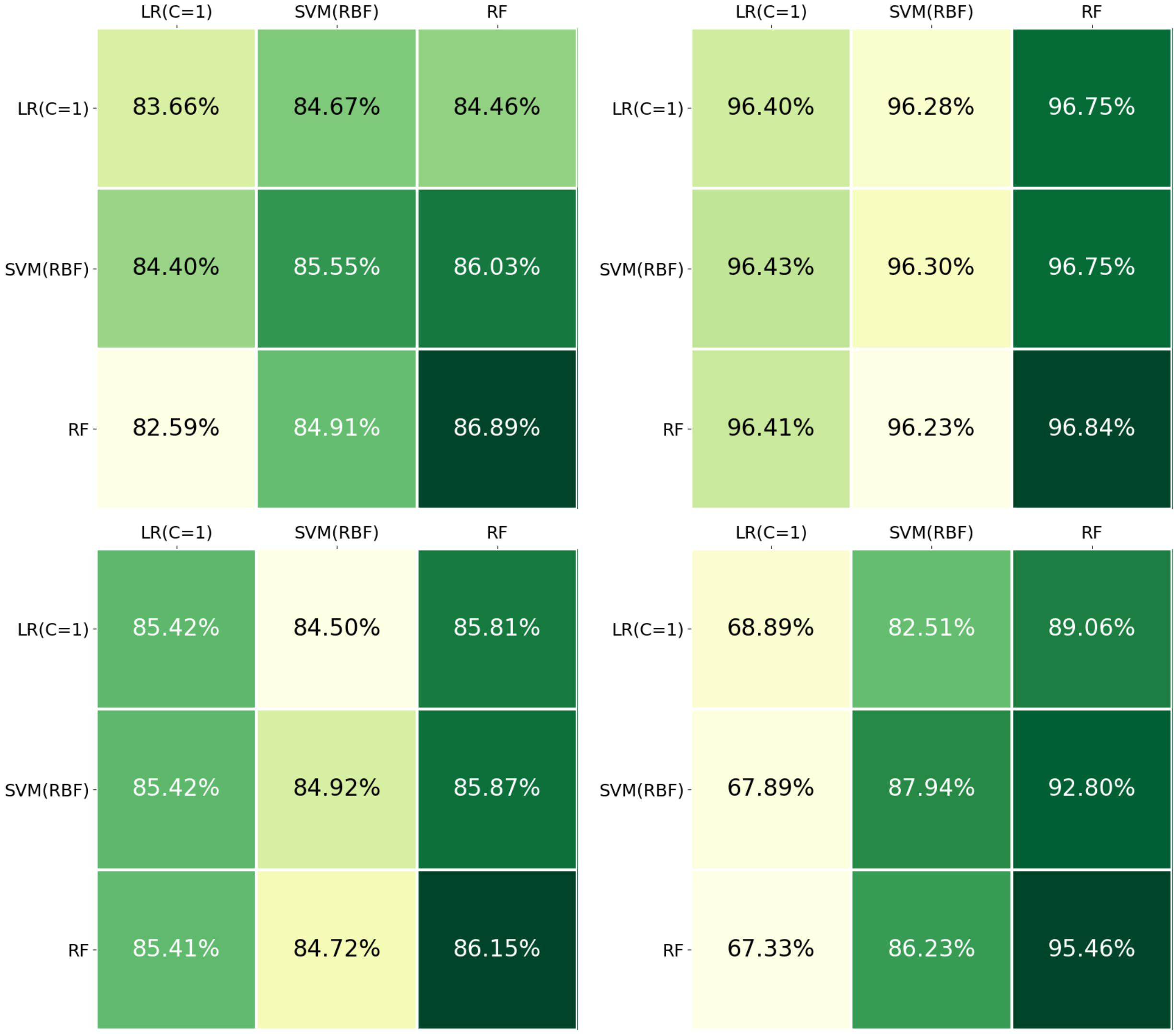
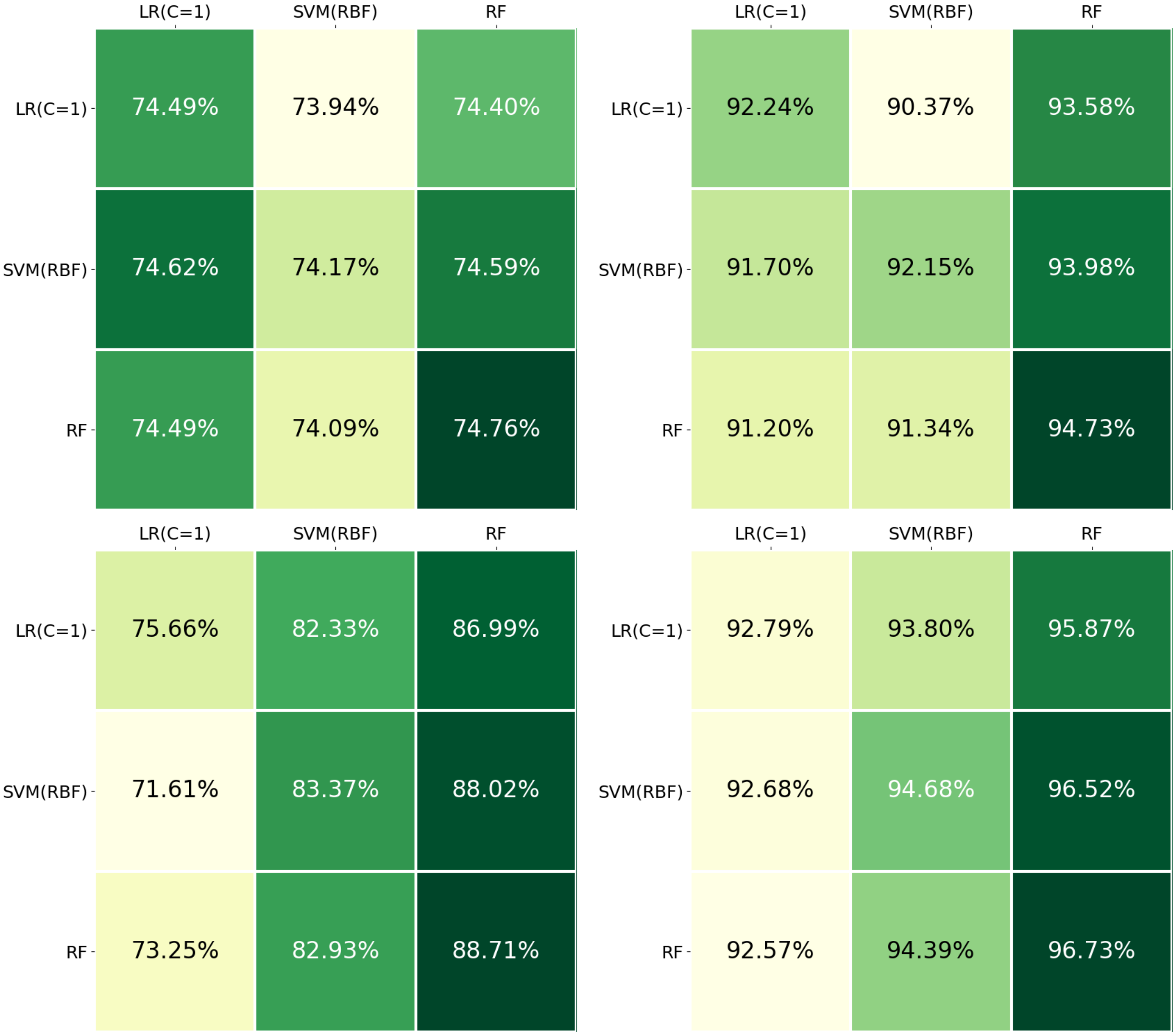


Figure 11: Mean AUBC of query-oriented model and task-oriented model on group 3. (Compatible RFs achieve best results.): *Parkinsons* (top-left), *Breast* (top-right), *Australian* (bottom-left), and *Ex8a* (bottom-right).



	LR(C=1)	SVM(RBF)	RF
LR(C=1)	74.49%	73.94%	74.40%
SVM(RBF)	74.62%	74.17%	74.59%
RF	74.49%	74.09%	74.76%

	LR(C=1)	SVM(RBF)	RF
LR(C=1)	92.24%	90.37%	93.58%
SVM(RBF)	91.70%	92.15%	93.98%
RF	91.20%	91.34%	94.73%

	LR(C=1)	SVM(RBF)	RF
LR(C=1)	75.66%	82.33%	86.99%
SVM(RBF)	71.61%	83.37%	88.02%
RF	73.25%	82.93%	88.71%

	LR(C=1)	SVM(RBF)	RF
LR(C=1)	92.79%	93.80%	95.87%
SVM(RBF)	92.68%	94.68%	96.52%
RF	92.57%	94.39%	96.73%

Figure 12: Mean AUBC of query-oriented model and task-oriented model on group 3. (Compatible RFs achieve best results.): *German* (top-left), *Spambase* (top-right), *Phoneme* (bottom-left), and *Phishing* (bottom-right).

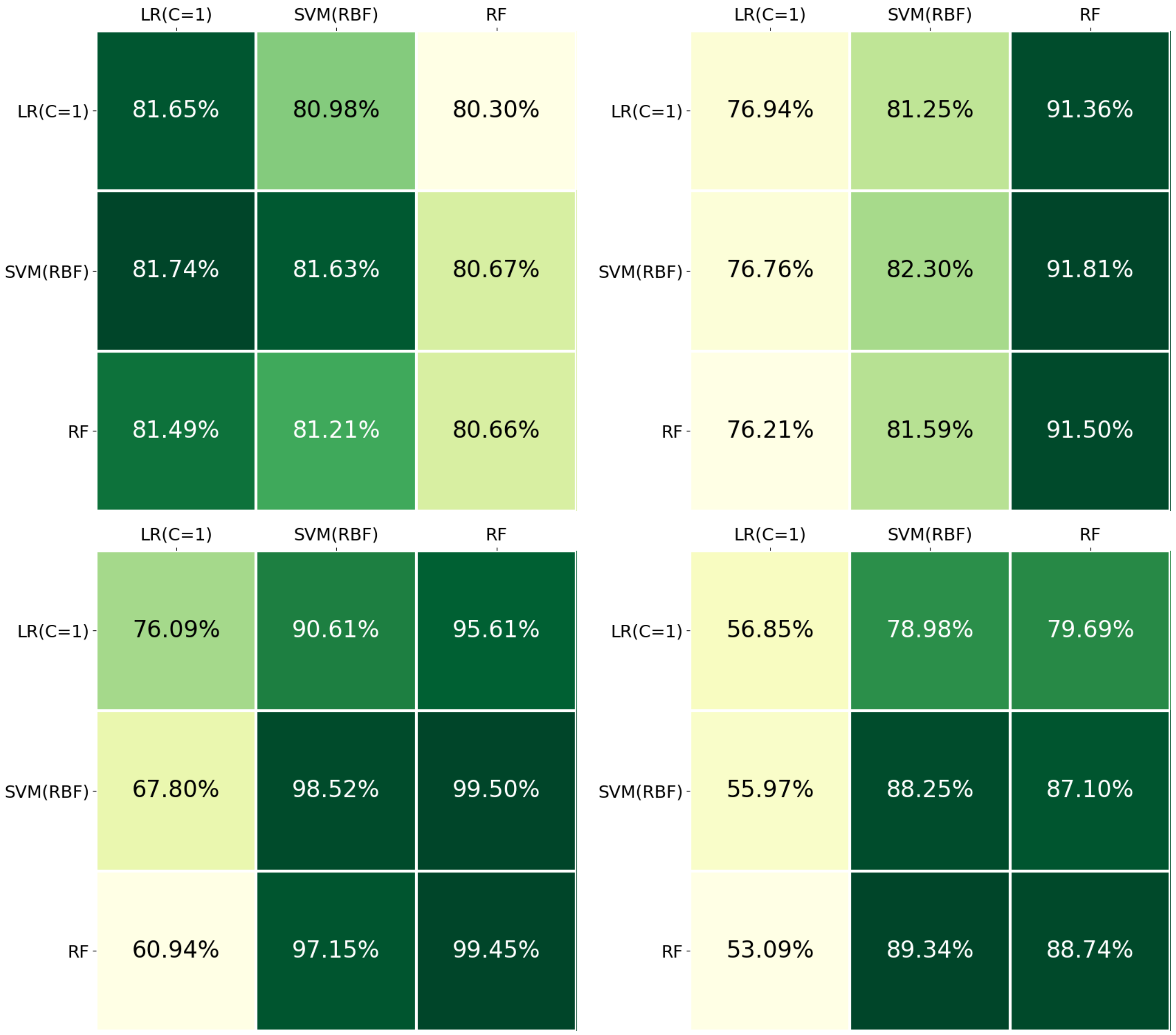


Figure 13: Mean AUBC of query-oriented model and task-oriented model on group 5. (Non-Compatible models achieve best results.): *Heart* (top-left), *Splice* (top-right), *Checkerboard* (bottom-left), and *Banana* (bottom-right).

Table 20: Data utilization rate of query strategies. The scores with [1], [2], or [3] mean the 1st, 2nd and 3rd performance on a dataset. ‘TLE’ means a query strategy exceeds the time limit.

	US	QBC	BALD	Hier	Graph	Core-Set	HintSVM	QUIRE	DWUS	MCM	BMDR	ALBL	LAL
Appendicitis	72.68%	88.27%	72.37%	83.77%	84.57%	78.46%	94.03%	79.59%	96.47%	73.37%	77.20%	75.18%	68.32%
Sonar	83.21%	96.93%	79.75%	*103.93%*	*105.19%*	*109.28%*	*113.21%*	*100.60%*	98.23%	82.09%	93.32%	94.02%	84.00%
Parkinsons	66.78%	*104.47%*	65.65%	89.01%	*115.53%*	*113.00%*	*125.06%*	*109.28%*	90.19%	71.19%	83.42%	90.02%	71.19%
Ex8b	72.10%	*100.51%*	75.02%	*108.36%*	*105.54%*	82.26%	*134.12%*	*107.66%*	*104.83%*	78.49%	97.42%	92.07%	78.03%
Heart	83.52%	93.96%	80.04%	87.83%	98.14%	*105.58%*	*126.91%*	*105.54%*	96.75%	85.71%	88.20%	94.19%	84.66%
Haberman	*108.22%*	*166.25%*	86.22%	*127.40%*	*117.93%*	*155.72%*	*194.88%*	*160.76%*	*110.28%*	84.09%	*108.16%*	*131.40%*	94.10%
Ionosphere	71.03%	*109.47%*	70.06%	*112.15%*	*118.36%*	*184.42%*	*204.09%*	*190.78%*	*266.93%*	75.05%	TLE	*117.39%*	78.14%
Clean1	66.38%	*101.51%*	68.54%	98.33%	*113.03%*	*105.73%*	*125.79%*	98.96%	99.31%	67.75%	*104.35%*	86.16%	75.82%
Breast	56.07%	91.36%	58.33%	83.40%	*161.72%*	92.76%	94.25%	90.92%	*342.11%*	58.36%	*124.45%*	78.14%	59.50%
Wdbc	48.49%	*118.85%*	49.57%	*104.03%*	*119.32%*	*113.68%*	*147.97%*	*112.58%*	*119.54%*	52.55%	94.83%	65.46%	52.33%
Australian	71.43%	95.51%	73.10%	98.52%	*121.25%*	*114.84%*	*106.65%*	*108.79%*	*125.02%*	73.67%	*101.76%*	92.53%	74.20%
Diabetes	95.33%	*104.14%*	92.27%	*104.10%*	*122.35%*	*109.48%*	*119.51%*	*116.73%*	*178.03%*	96.93%	*110.47%*	*117.40%*	92.93%
Mammographic	72.07%	*153.42%*	71.66%	89.03%	91.95%	81.27%	*103.54%*	87.55%	*128.24%*	64.89%	*101.07%*	93.26%	59.36%
Ex8a	43.75%	*111.29%*	42.78%	97.60%	*109.29%*	57.61%	*165.84%*	*176.49%*	*166.71%*	46.08%	88.94%	*142.32%*	54.80%
Tic	75.39%	96.17%	80.45%	92.89%	*124.21%*	82.87%	*140.92%*	*129.96%*	*209.75%*	89.96%	*111.72%*	61.44%	TLE
German	92.08%	*119.64%*	96.81%	*129.11%*	*122.34%*	*114.60%*	*144.97%*	*120.44%*	*237.03%*	*104.75%*	*136.86%*	*106.77%*	92.01%
Splice	77.98%	*108.12%*	79.25%	*101.68%*	*108.61%*	*164.32%*	*144.63%*	97.44%	*181.65%*	77.97%	*104.14%*	96.84%	84.31%
Gcloudb	61.60%	*147.10%*	60.36%	94.06%	*147.55%*	*104.00%*	*488.17%*	*423.89%*	*142.77%*	66.29%	98.02%	81.17%	64.59%
Gcloudub	46.41%	*105.08%*	48.27%	84.01%	85.21%	*119.68%*	*273.89%*	*186.53%*	*168.58%*	47.57%	*123.12%*	*103.64%*	59.03%
Checkerboard	80.42%	*125.73%*	70.49%	99.35%	79.21%	*124.08%*	*916.17%*	*801.92%*	*553.50%*	58.66%	*106.13%*	*141.07%*	50.63%
Spambase	22.96%	*109.32%*	19.14%	94.56%	*122.51%*	*132.64%*	*282.80%*	*207.05%*	96.33%	21.76%	TLE	*104.81%*	25.40%
Banana	65.49%	*116.15%*	47.70%	*131.01%*	*132.31%*	83.74%	*455.27%*	*396.93%*	*691.08%*	56.57%	*122.59%*	*194.17%*	52.98%
Phoneme	33.78%	*102.37%*	33.95%	68.43%	*100.16%*	72.78%	*116.82%*	92.83%	*107.17%*	34.62%	87.11%	83.28%	39.12%
Ringnorm	27.95%	*114.87%*	31.49%	*142.95%*	*250.54%*	*866.58%*	*817.08%*	*844.31%*	*731.19%*	36.33%	TLE	*208.99%*	*124.45%*
Twonorm	54.11%	90.13%	42.61%	*285.71%*	*114.89%*	97.52%	*902.62%*	TLE	*114.72%*	58.90%	TLE	*134.42%*	68.25%
Phishing	18.38%	*118.07%*	20.24%	*102.20%*	*142.07%*	98.98%	*215.34%*	TLE	*151.26%*	20.96%	TLE	59.51%	22.77%
Covertype	45.70%	TLE	46.78%	TLE	*116.98%*	TLE	TLE	TLE	*117.86%*	TLE	TLE	TLE	TLE
Bioresponse	70.86%	TLE	72.53%	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE
Pol	17.78%	TLE	17.21%	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE	TLE