Data Acquisition for Domain Adaptation of Closed-Box Models

by

Yiwei Liu

A Thesis Submitted to
the Faculty of Graduate Studies, York University
in Partial Fulfillment of the Requirements
for the Degree of Master of Science

GRADUATE PROGRAM IN ELECTRICAL ENGINEERING AND COMPUTER SCIENCE
YORK UNIVERSITY
TORONTO, ONTARIO
JANUARY 2024

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Machine learning (ML) marketplace provides customers with various ML-based solutions to accelerate their business. Models in the ML market are often available as closed boxes, but when they are applied to new domains, their performance may suffer from distribution shifts. Prior techniques cannot address this problem, because they are either impractical to use or against the nature of closed-box models. Instead, to enhance a closed-box classification model when performance discrepancies occur in the target domain, we propose to acquire extra data to construct a "padding" model to help the original closed box with its classification weaknesses in the target domain. Our solution consists of a "weakness detector" to discover the deficiency of the original closed-box model and the Augmented Ensemble approach to combine the source and the padding model for better performance in the target domain and further diversifying the ML marketplace. Extensive experiments on several popular benchmark datasets confirm the superiority and robustness of our proposed framework over baseline approaches.

Acknowledgements

I would like to thank all the people who made this thesis possible. My supervisor, Professor Xiaohui Yu, has generously mentored and supported me through my Master's studies. Professor Nick Koudas, Dr. Yifan Li and Dr. Yueting Chen give me valuable advice and help during my research journey. I also thank my supervisory committee member, Professor Aijun An, for her insightful feedback on this thesis. Lastly, I am deeply grateful to my incredible family and friends for their unwavering encouragement and support over the past two years.

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1 Motivation

As more and more industries adopt machine learning (ML) techniques to facilitate their business, ML marketplaces are becoming a significant resource for ML business solutions. The ML marketplace consists of data markets and model markets. Data markets such as Amazon Data Exchange [5] and Databricks Marketplace [17] offer datasets for customers to build or improve their models, while model markets provide customers with trained models in various forms. For example, Amazon Rekognition [4] and Google Cloud Vision [22] sell the use of model APIs; Open Model Zoo [44] provides downloadable versions of the models with different licensing terms. Meanwhile, models in the ML market are often closed boxes due to model sellers' business privacy concerns and the convenience for customers to use.

However, when a customer applies a closed-box model to a domain (the *target domain*) that is different from where the closed box is trained (the *source domain*), the closed box may experience performance degradation because of the distribution discrepancy. Figure 1.1 illustrates an example of this problem. The current closed-box classifier from the ML market distinguishes class *large mammals* and class *medium-sized mammals*, and it is trained on a dataset with *bear* and *lion* images for *large mammals* while *raccoon* and *skunk* images for *medium-sized mammals*. However, a customer applies this classifier to his validation dataset drawn from the target domain with another two categories (*wolf* images for *large mammals* and *fox* images for *medium-sized mammals*). Then, the closed-box model can have a poor performance in classifying *wolf* and *fox*, and the overall validation accuracy of $\mathcal{M}_{\mathbf{s}}$ may not be as good as what is advertised for this classifier in the ML market. Therefore, it is necessary to study how to improve the closed-box model for these new categories. Interestingly, if a

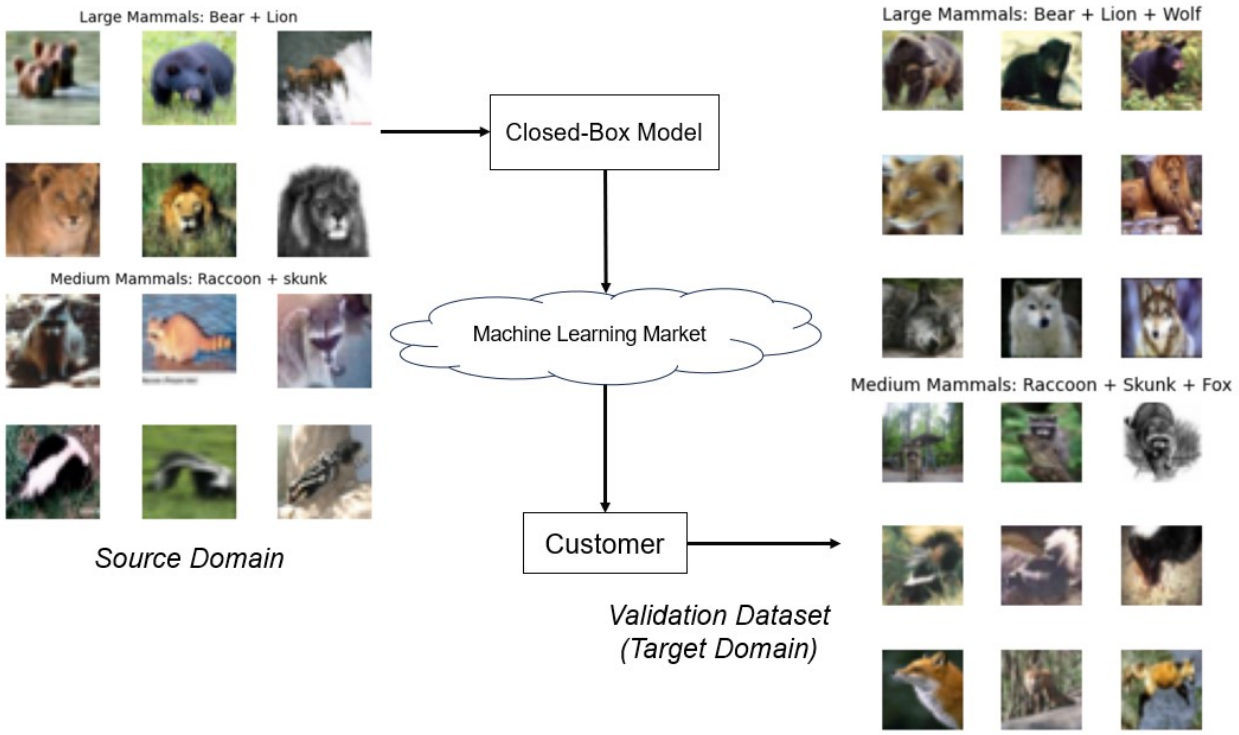

Figure 1.1: a Closed-box Model Encounters a Shifted Target Domain

customer makes a better model for the target domain than the present closed box, they can sell this new model back to the market and add up the market diversification.

One promising approach recently proposed for improving ML model accuracy is to collect additional data from the ML market to refine existing models [36]. However, this method does not apply to closed-box models without accessible model parameters or source data to customers. Other research, specifically on domain adaptation for closed-box models, transfers the capability of the closed box to a new model by knowledge distillation and update distilled models for the target domain by semi-supervised learning [65] [56]. However, closed-box models from the ML market are usually built from a large collection of data. It can be costly to train an entirely new model that outperforms the original closed-box model in the target domain. Especially when the new categories described in Figure 1.1 only account for a small percentage of the target domain. Instead, the original model can be re-purposed for the overlap between the target and the source domain.

To improve a closed-box model for a target domain, this thesis focuses on classification

models and proposes a novel domain adaptation framework that externally augments the existing closed-box model with extra data from the ML market to create a new model for the target domain. After a customer detects accuracy degradation (relative to the model's advertised accuracy) of a given closed-box classification model $\mathcal{M}_\mathbf{s}$ on his validation dataset from the target domain, the customer purchases data from the ML market, subject to a specified budget, to build a new "padding" model $\mathcal{M}_\mathbf{p}$ with better accuracy on data that model $\mathcal{M}_\mathbf{s}$ does not handle well in the target domain.

1.2 Methodology

When implementing the proposed framework, we have two challenges: (1) How to make the best use of limited budgets to collect labeled data from the market and build the padding model $\mathcal{M}_\mathbf{p}$ to assist the source closed-box classification model $\mathcal{M}_\mathbf{s}$ in the target domain? (2) How to employ $\mathcal{M}_\mathbf{p}$ to enhance $\mathcal{M}_\mathbf{s}$ for the target domain where labels are always revealed with delays?

For the first challenge, we build a "weakness detector", WeDe, that detects when $\mathcal{M}_\mathbf{s}$ performs poorly in the target domain. WeDe extracts features of $\mathcal{M}_\mathbf{s}$'s weaknesses by classifying $\mathcal{M}_\mathbf{s}$'s correct and incorrect predictions in the validation set. WeDe also generates *weakness scores* to estimate data utility in constructing $\mathcal{M}_\mathbf{p}$ complementary to $\mathcal{M}_\mathbf{s}$ in the target domain. In other words, WeDe acts as a proxy of $\mathcal{M}_\mathbf{s}$ during data acquisition from the market, and it avoids privacy leakage or misuse of $\mathcal{M}_\mathbf{s}$ caused by data sellers. Based on WeDe, we also introduce two data acquisition methods, which vary in the frequency of data acquisition and WeDe update.

For the second challenge, we propose to use WeDe with an approach named Augmented Ensemble (AE) to create $\mathcal{M}_\mathbf{t}$ as an ensemble of $\mathcal{M}_\mathbf{s}$ and $\mathcal{M}_\mathbf{p}$ for $\mathbf{T}$, whether classification probabilities are available or not from $\mathcal{M}_\mathbf{s}$. Compared to applying only $\mathcal{M}_\mathbf{p}$ to the target domain, AE also employs $\mathcal{M}_\mathbf{s}$ in the domain intersection between the target and the source, and thus AE is more beneficial to domain adaptation of closed-box models, especially when the data acquisition budget is not large enough to build a powerful $\mathcal{M}_\mathbf{p}$ for the entire target domain.

1.3 Contribution and Outline

In this thesis, we make three contributions to improving closed-box classification models for a new domain with data acquisition.

- **Efficient and secure data acquisition with a given budget.** We design a model weakness detector by learning about prediction mistakes of the closed-box model in a validation set drawn from the new domain, and then we use the detector to acquire data that is most likely to be the weakness of the closed box with a limited budget. Data acquisition with the weakness detector protects the privacy of our model and the target data.

- **Effective and versatile ensemble for the closed-box model with a complementary one.** A new model is built from acquired data to help the existing closed-box model with its weaknesses in the target domain. We apply both the new model and the closed box to the target domain. Then, based on the accessibility of $\mathcal{M}_{\mathbf{s}}$'s outputs, we aggregate predictions from the two models as the output of an improved closed box for the target domain.

- **Extensive and empirical test of our proposed solution.** We design classification tasks of different scales on two popular datasets with deliberate domain shifts and compare our solution with multiple baseline methods. Experiments show that our solution outperforms baselines, especially when the data acquisition budget is small, or the data acquisition source covers more data categories than the target domain. We also extensively analyze factors that impact our solution, providing insights for customers to choose suitable data acquisition strategies for different application scenarios.

The rest of the thesis is organized as follows. Chapter 2 reviews related work. Chapter 3 formally defines notations and our research objective: data acquisition for domain adaptation of closed-box classification models. Chapter 4 elaborates on our proposed methodologies to accomplish our objective. Chapter 5 presents extensive experimental results and analysis, as well as a discussion on future work. Chapter 6 summarizes how our data acquisition

framework works effectively for domain adaptation of closed-box models and points out future directions for extending our framework.

2 Related Work

We introduce related work on the problem of data acquisition for domain adaptation of closed-box models, and we also describe their limitations to solving this problem or their ties with our proposed solution. First, we present studies on the machine-learning market, especially those on data acquisition in the machine-learning market. Next, we introduce three research areas related to data acquisition: active learning (acquiring labels for data already possessed by the customer), data augmentation (an alternative practice for increasing model training data), and model learning curves (guidance to the design and application of data acquisition methods). Lastly, we discuss prior work on domain adaptation for closed-box models.

2.1 Machine Learning Market

A Machine Learning (ML) market usually takes the form of model markets and data markets for ML applications. Literature on model markets and data markets is detailed in the following two subsections.

2.1.1 Model Market

Chen et al. present the very first model market design with an arbitrage-free mechanism and model versioning [12]. The arbitrage-free mechanism prevents a collection of weak models from being even cheaper than a single model with a competitive quality as the collection, which protects the benefit of model sellers. Model versioning provides buyers with variations of an optimal model with different prices. The price of a model variation is proportional to

the performance of this variation. Model versioning can be done by injecting noise into the parameters of the optimal model.

Liu et al. propose a more comprehensive model market prototype *Dealer* [40]. *Dealer* has three components: data owners, model buyers, and market brokers connecting the first two parties. Data owners' revenues depend on their data quality and data privacy preservation. Model buyers can choose from multiple model versions with different prices and qualities. Brokers in this prototype are responsible for maximizing data owners' revenues and ensuring that model quality satisfies buyers with different needs and investment budgets.

Some research even integrates federated learning (FL) to protect data owners' privacy. Instead of sending data to brokers for training models, data owners can train models by themselves and then upload model gradients to brokers for aggregation under an FL framework [69]. Sun et al. propose algorithms to maximize the broker's profits from model trading revenues and model training costs in an FL-based model market [60].

2.1.2 Data Market

Agarwal et al. rigorously design a mechanism for a fair and trustful data market for ML tasks [3]. However, they do not discuss the concerns of data sellers and buyers. Data sellers aim to maximize their compensation from their data but would also like to preserve data privacy. On the other hand, data buyers focus on data acquisition for building or improving their ML models [45]. Regarding these concerns, Fernandez et al. design a more practical and comprehensive data market as a platform of trading data [19]. Also, to avoid strategic data purchases in the market, Fernandez introduces three market protections to data pricing schemes [9]. In the next two sections, we introduce more work in the design of data markets to address the concerns of sellers and buyers.

2.1.2.1 Compensation Maximization for Data Sellers

Data sellers' compensation depends on how much their data contributes to model training and how much data privacy they want to preserve. Therefore, to maximize sellers' compen-

sation, their data utility should be properly formulated based on the contribution of their data and the data privacy they sacrifice.

One popular model training data evaluation practice uses the Shapley Value from game theories. Shapley Value estimates the contribution of individual members to their group by how much the removal of these individuals affects group outcomes. Ghorbani et al. introduce a Shapley Value-based framework to evaluate the influence of training data for ML models and propose efficient algorithms for computing Shapley Value [21]. Jia et al. also present methods to facilitate the computation of Shapley Value in the context of ML models, but they introduce additional constraints to evaluating the influence of training data [30]. In addition to Shapley Value estimation by data removal, the impact of training data can be assessed with data perturbation [33][34], or gradient descents during model training [48].

Data privacy preservation is another factor in data valuation for sellers. Data sellers protect their data by differential privacy, which adds noise to the parameters of trained models or gradients when models are trained in a federated learning system [12][40][69]. Apart from differential privacy, sharded model training also helps preserve data privacy. A sharded system is a cluster of individual computing nodes with training data, and the output of this cluster is an aggregation of model predictions from all the involved nodes. Xia et al. design a new Shapley Value-computation method to evaluate the contribution of data sellers in a shared system [63].

2.1.2.2 Data Acquisition for Data Buyers

To use budgets efficiently, data buyers first need to know what kind of data is the most helpful in training models from scratch or improving existing models, and then they can adopt acquisition strategies for different purposes.

In terms of data valuation for training models, Yoon et al. build a comprehensive data value estimator by reinforcement learning [66]. Samples with the highest estimated values are selected. Predictions from the trained model act as rewards for updating the data value estimator. This procedure requires a lot of model retraining. Therefore, it is not a practical solution to data valuation for model training. Another way to quantify the value of training

data is by measuring the similarity between training and testing data [24] [64]. The intuition behind it is that if the distributions of model training and the testing data are close, models tend to exhibit good testing performance.

As for collecting additional data instances for model improvement, Li et al. use a domain classifier for the original and the new domain to quantify data novelty and propose two adaptive acquisition strategies based on the estimated data novelty [36]. When the source of data acquisition has a mixture of distributions, Chai et al. perform clustering to locate a homogeneous subdomain in the acquisition source and then build a reinforcement learning agent to decide from which cluster they collect data [10].

However, none of the aforementioned data acquisition work accommodates our setting with closed-box models from the ML market. First, data acquisition for model training is orthogonal to our goal of improving existing closed-box models. Second, in prior studies on data acquisition, customers should provide the seller with the information of model source data for evaluating the novelty of candidate data, but the source data of closed-box models are also inaccessible to customers.

Model vulnerability identification is also helpful in building up models from the data perspective but does not require the exposure of model information. Jain et al. build a binary classifier of correct and incorrect predictions of a target model to reveal the pattern of this model's prediction mistakes [29]. However, their model mistake detector is per data class and is useless in unlabeled target domains. Abid et al. disclose intuitive causes of model prediction mistakes by studying model outputs after removing certain data features [2], but the impact of data features on model vulnerabilities is orthogonal to our data acquisition for fixing ineffective closed-box models. Other research tries to understand model predictions from input data. Ilyas et al. unveil the connection between training and testing data given a model. They treat training data as inputs while model predictions as outputs for a regression equation. Coefficients in such equations reflect how training data impact model predictions on test data [28]. Cong et al. work on complicated models behind cloud APIs and identify the most influential data features on model performance by solving an overdetermined equation system built from linear regions of target models [16]. However, it is still unclear how to

apply these studies to reveal model vulnerabilities directly.

2.1.3 Summary

The sophisticated design of machine-learning (ML) markets satisfies three participants: sellers, buyers, and brokers. However, prior data acquisition methods with the ML market cannot accommodate our objective: we purchase data for domain adaption of closed-box models whose parameters and source data are both inaccessible to sellers. Hence, it is essential to develop a new strategy to accomplish our goal of data acquisition for domain adaption of closed-box models.

2.2 Active Learning

By active learning, the customer requests labels for his data and uses newly labeled data to improve his models. The selection of data for labeling depends on the prediction uncertainty or feature diversity of selection candidates. Prediction uncertainty is often measured by the entropy of classification probabilities from the model. Guo et al. formulate the acquisition of a batch of data labels as an optimization problem that aims to maximize classification accuracy on labeled data and minimize classification uncertainty on unlabeled simultaneously[23]. Houlsby et al. extend active learning to the Gaussian Process Classifier, a non-parametric model by Bayesian Active Learning by Disagreement (*BALD*) [26]. They compute the mutual information between model predictions and model parameters. If a model is very uncertain about a given data point, this data point reveals much information about the model parameter. However, *BALD* only works for label acquisition by a single instance, so it demands a lot of model retraining. To overcome the intensive model retraining in *BALD*, Kirsch et al. extend the *BALD* approach for a batch-mode acquisition of data labels [32]. There are also other measurements of model prediction uncertainty, e.g. the maximum classification probability and the margin between the top two probabilities [13].

Active learning based on feature diversity employs core-set selection or representative learning to select a bath of data for labeling. Sener et al. choose a core set of unlabeled

data by minimizing the distance between every point in this core set and all the data points outside the core set [55]. However, their research may not work well in high-dimensional data. To adapt active learning to high-dimensional data, Sinha et al. propose to adopt a variational auto-encoder (VAE) to learn the embedding space of data and an adversarial classifier to distinguish labeled and unlabeled data in the embedding space [59]. VAE aims to fool the classifier that all data are labeled while the classifier needs to discriminate between labeled and unlabeled data.

Some active learning research combines prediction uncertainty and feature diversity to select data for labeling. Prabhu et al. first cluster unlabeled datasets by uncertainty and then from each cluster, they select data points closest to the cluster center [47]. This method proves to be more beneficial than those solely based on uncertainty or diversity.

However, active learning is orthogonal to our work which focuses on acquiring complete data points from data sellers rather than asking for labels of data owned by customers themselves.

2.3 Data Augmentation

Data augmentation aims to enrich training data and make models more robust. Typical data augmentation research focuses on two types of data: images and relational data [58]. Traditional image data augmentation includes adding noise, color, rotation, and cropping. More recently, some deep learning techniques have been integrated with data augmentation. Adversarial data augmentation perturbs the source domain to maximize the prediction entropy of the current model [67]. A higher mutual information of this kind indicates a larger domain shift after augmentation. Generative adversarial models are also used in creating synthetic images [6]. Given an image from a class, a generator network creates fake images also from this class but appearing differently from the given image as told by a discriminator. However, three important questions remain for image data augmentation: what is the best augmentation strategy for a given task? How to evaluate the quality of synthetic data? How much synthetic data is needed?

The enrichment of relational data lies in expanding predictive data attributes from table joins in public databases. Chepurko et al. propose the first end-to-end system for relational data augmentation by joining tables from a public data repository and the customer's database [15]. They first construct table representations from a plan of joining tables. After the plan is executed, essential attributes from the joined table are returned to the customer using a random noise injection-based feature selection algorithm. Zhao et al. improve this end-to-end attribute augmentation system with a more powerful relational data embedding design [68]. Liu et al. adopt reinforcement learning to develop a more intelligent data attribute augmentation system [39]. They build two kinds of agents to decide which tables to join for selecting useful data attributes. One agent uses a traditional multi-arm-bandit algorithm, while the other employs a deep neural network for optimizing attribute selection rewards, which are the performance of models trained from data with expanded attributes. However, it is still unsure if additional attributes can address prediction errors caused by distribution shifts.

In summary, some limitations exist in widely applying data augmentation to model training, especially for domain adaptation of models. Therefore, studying the acquisition from the data market is still necessary.

2.4 Model Learning Curves

Model performance is often subject to factors like the size of the training dataset and the choice of model training parameters. Since we aim to improve closed-box models, we are interested in how those factors affect model performance. The model learning curve is one of the tools for analyzing such factors. When a learning curve is modeled by the size of the training dataset, this curve often presents three stages from empirical experiments: (1) model predictions are almost like random guesses when the training dataset is too small; (2) prediction errors decrease following a power-law rule when the training data is enough; 3) prediction errors stop declining when the training data is too much and has no more new information for the model to learn [25]. Rosenfeld et al. even manage to formulate and extrapolate such learning curves [51]. When Tae et al. improve model fairness by

collecting data from representative groups of a third-party dataset, they also use the model learning curve of each group to maintain overall accuracy [61]. We apply the idea of model learning curves to our empirical investigation of factors that affect our proposed solution in section 5.4.

2.5 Domain Adaption for Machine Learning Models

Updating machine learning models for a new domain usually involves distribution shift detection for identifying the discrepancy between the target and the source domain, and fine-tuning models to new domains by continual learning. However, closed-box models have private model parameters and source data, which make it impossible to adopt distribution shift detection and continual learning to accomplish domain adaptation of closed-box models. Then, some researchers develop domain adaptation techniques specifically for closed-box models, but the large need for target data limits the wide application of those techniques. In this section, we first introduce prior work on distribution shift detection, continual learning, and ensemble learning, a relevant topic on model training. After that, we discuss some work on improving closed-box models for a new domain.

2.5.1 Distribution Shift Detection

There are two areas in the research of detecting distribution shifts. One of them starts by learning representations of in-distribution data and then checks if testing data has similar representations. The degree of distribution shifts is returned after comparison. The other area in distribution shift detection research uses a two-sample test, a statistical tool to check whether the two samples are drawn from significantly different distributions given a significance level. However, measuring the distribution shift degree in the two-sample test is not as easy as representation-learning-based techniques.

2.5.1.1 Representation Learning

D. Abati et al. employ the variational autoencoder to learn representations of in-distribution data and adopt reconstruction errors of testing data to measure the distribution shift degree [1]. Pidhorskyi et al. also use the decoder-encoder network for representation learning on in-distribution data, but they use a probability to estimate the distribution shift degree [46]. Ruff et al. adopt one-class SVM to capture the distribution of normal data, and all the out-of-distribution data tends to fall outside the hypersphere of one-class SVM [52].

2.5.1.2 Two Sample Test

Friedman proposes compressing multivariate samples in the two-sample test into univariate ones with a binary classifier to facilitate calculation [20]. Recently, Lopez-Paz et al. present extended theoretical and empirical analyses of Friedman's idea and showcase the application of the classifier-based two-sample-test in evaluating image generative models [42]. A comprehensive study of two-sample-test-based distribution shift detection from Rabanser et al. shows that dimension reduction techniques like autoencoders, which do not necessarily generate univariate data, achieve better results than classifier-based methods under certain kinds of distribution shifts [49].

2.5.1.3 Summary

If model buyers identify distribution shifts in the target domain, they can collect data from the new distribution to build up their models. However, distribution-shift detection requires access to the source data of models, and closed-box models hide their training data from the public. Therefore, the detection methods mentioned above cannot address our target problem.

2.5.2 Ensemble Learning

Ensemble learning is often employed to alleviate model overfitting when the model training dataset is too small, and it also boosts the overall performance when the training data is imbalanced in data categories [53]. In ensemble learning, a collection of models is constructed by *bagging* or *boosting*. The *bagging* approach trains a group of independent models of the same structure but with different training data sampled from the original dataset with replacement. By contrast, the *boosting* strategy builds models one by one and re-weights training data based on predictions from the previous model [54]. Data incorrectly predicted by the current model are attached with more weight when they are used in training the next model. Results from models in the ensemble can be aggregated by simple majority voting or with weights based on each model's strength [53]. In our proposed solution, we weigh predictions from the original closed box and an assistant model according to the relation between inputs and the two models and then combine weighted predictions as the output of a model ensemble.

2.5.3 Continual Learning

Models can be sequentially updated to tasks from new domains while reserving knowledge of previous data domains by *Continual Learning* (CL). Replay-based and regularization-based CL are two mainstreams in CL research [43]. Replay-based CL usually stores data from old tasks in data buffers. When a new task comes, data of the new task is mixed with samples from buffers to retrain models [11]. Shin et al. employ generative models to simulate data from old tasks to relax the restriction from buffer sizes [57]. On the other hand, regularization-based CL approaches freeze specific model parameters to avoid forgetting what has been learned when models are being updated [31]. Frozen parameters can be selected by model weight uncertainty from Bayesian Network [18]. Buzzega et al. even propose an approach that is a mixture of replay-based and regularization-based CL techniques to deal with general cases of updating models [7].

Some CL research overlaps with ensemble learning to balance forgetting and learning

during model updates. Liang et al. assemble old and new models with a parameter optimized on a validation set [38]. Cano et al. handle streaming CL tasks with imbalanced and constantly changing data domains by adaptive model ensembles [8]. Wen et al. design an effective computation method to speed up ensemble learning and also improve a CL problem with a long task sequence [62].

Continual learning is widely applied to update models for new domains. However, since closed-box models do not expose their parameters to the public, we cannot use continual learning to modify closed-box models.

2.5.4 Domain Adaptation for Closed-Box Models

Previous research on domain adaptation of closed-box models builds a new model for the target domain by transferring knowledge from the closed-box model and refining the transferred model. For example, Liang et al. accomplish domain adaptation of closed-box models by training a new model with knowledge distillation where pseudo-labels from the "teacher" closed box are filtered to supervise the training of the new model [37]. Rather than filtering out noisy information from the closed box to train the target model by clean supervision, Yang et al. break the target domain into two parts, one similar to the source domain while the other diverges from the source, by the cross-entropy loss between closed box predictions and a pair of networks distilled from the closed box. They use domain divisions to refine distilled models in a semi-supervised manner and obtain the final model for the target domain [65]. To protect sensitive information in the target domain, Shi et al. distill the closed-box model with third-party data and variations of those data from the target domain [56].

These techniques can require much data for training new models, but customers only have limited samples from the target domain for validating purchased models before deployment. Hence, existing domain adaptation research for closed-box models still has limitations to be put into practice.

2.5.5 Summary

Constrained by the unique property of closed-box models, prior work in domain adaption of normal machine learning models cannot address our research problem. Although researchers have recently proposed to use predictions of the closed-box model to build another model for the target domain, they may need lots of target data to train the new model, which makes their studies less applicable. Therefore, it is necessary to design a more practical but effective solution for domain adaptation of closed-box models.

3 Preliminaries and Problem Definition

In this chapter, we first formally describe all the notations used in this thesis and then rigidly define the problem of data acquisition for improving closed-box models.

3.1 Preliminaries

Source Domain and Target Domain. A customer would like to have a classification model $\mathcal{M}_{\mathbf{t}}$ for a *target domain* $\mathbf{T} = \{X, Y\}$, where X is the feature space and Y is the label space. The customer owns a small and labeled *validation set* $\mathcal{D}_V$ from the target domain, $\mathcal{D}_V \subset \mathbf{T}$, while labels of any other data from $\mathbf{T}$ are revealed with delays. To accomplish the classification task in $\mathbf{T}$, the customer selects a suitable closed-box model $\mathcal{M}_{\mathbf{s}}$ from the model market according to the advertised model performance. We call the data domain that $\mathcal{M}_{\mathbf{s}}$ is trained on the *source domain* $\mathbf{S}$.

Data Pool. The data pool, denoted as $\mathcal{D} \subset \mathbf{T}$, is a labeled dataset provided by a seller from the data market that allows the customer to purchase samples to enhance the model performance. Every purchase of samples from $\mathcal{D}$ comes at a price. We assume a scenario where the customer is restricted by budget B to acquire no more than B samples from the data pool. Data acquisition can be executed in several rounds. Each round involves the purchase of one or multiple samples from $\mathcal{D}$.

Interaction with the Data Market. A round of interaction between the customer and the seller in the data market consists of two steps: (1) In the i-th round, the customer sends to the seller a data valuation function $F_i(\mathbf{x})$ where $\mathbf{x}$ is an input data instance from $\mathcal{D}$, and a data acquisition budget for this round, b_i. Furthermore, the customer aims to preserve

his model and target data privacy as much as possible. (2) The seller computes $F_i(\mathbf{x})$ on relevant data (i.e., data instances within the label space Y) and then returns to the customer those instances with the highest $F_i(\mathbf{x})$ values, $\mathcal{D}'$, such that $|\mathcal{D}'| = b_i$ and $\sum_i b_i \leq B$.

3.2 Problem Definition

Definition 1 (Data Acquisition for Domain Adaptation of the Closed-Box Model)**.** *Given a data acquisition budget B, a closed-box model $\mathcal{M}_\mathbf{s}$, a data pool $\mathcal{D}$, $\mathcal{D} \subset \mathbf{T}$, where $\mathbf{T}$ is the target domain, and a validation set $\mathcal{D}_V \subset \mathbf{T}$, the objective of data acquisition for domain adaptation of $\mathcal{M}_\mathbf{s}$ is to identify the B samples to acquire from $\mathcal{D}$ (denoted by $\mathcal{D}_B$) to optimize the accuracy of a target model $\mathcal{M}_\mathbf{t}$ in $\mathbf{T}$.*

For example in Figure 1.1, the customer has to discriminate two mammal classes in $\mathbf{T}$. Class *large mammal* is composed of *wolf*, *lion*, and *bear* images, while class *medium-sized mammal* is made of *fox*, *skunk*, and *raccoon* images. From the model market, the customer purchases a closed-box model $\mathcal{M}_\mathbf{s}$ with an advertised accuracy of 90% in classifying the two target classes. To validate the accuracy of $\mathcal{M}_\mathbf{s}$ in $\mathbf{T}$, the customer applies $\mathcal{M}_\mathbf{s}$ to a validation set $\mathcal{D}_V$ where images are labeled and follow the distribution of $\mathbf{T}$. In fact, $\mathcal{M}_\mathbf{s}$ is trained on a domain $\mathbf{S}$ without *wolf* for *large mammals* or *fox* for *medium-sized*. As a result, the customer only observes a validation accuracy of 70%. To build a better model $\mathcal{M}_\mathbf{t}$ for $\mathbf{T}$, the customer decides to use a budget B to purchase data from $\mathcal{D}$ which only covers images of the two target mammal classes.

4 Methodology

In this chapter, we first provide an overview of the proposed solution framework and then describe individual components in this framework.

4.1 Overview

Our proposed solution to data acquisition for domain adaptation of closed-box models is illustrated in Figure 4.1. To construct the model $\mathcal{M}_{\mathbf{t}}$ for the target domain $\mathbf{T}$ with the source model $\mathcal{M}_{\mathbf{s}}$ and data acquisition budget B, the customer first trains a binary classifier WeDe by correct and incorrect predictions of $\mathcal{M}_{\mathbf{s}}$ in the validation set $\mathcal{D}_V$ from $\mathbf{T}$ to learn about the feature of $\mathcal{M}_{\mathbf{s}}$'s weaknesses. Then, the customer acquires a set of data instances $\mathcal{D}_B$ from a data pool $\mathcal{D}$ by providing the seller with budget B and a data valuation function $F(\mathbf{x})$ influenced by WeDe. The acquired data $\mathcal{D}_B$ is used to train a padding model $\mathcal{M}_{\mathbf{p}}$ to deal with weaknesses of $\mathcal{M}_{\mathbf{s}}$ in $\mathbf{T}$. $\mathcal{M}_{\mathbf{p}}$ can have any kind of classification architecture. Finally, the output of $\mathcal{M}_{\mathbf{t}}$ in $\mathbf{T}$ is constructed with Augmented Ensemble (AE) by combining predictions from $\mathcal{M}_{\mathbf{s}}$ and $\mathcal{M}_{\mathbf{p}}$ according to target data characteristics identified by WeDe. Solid arrows in Figure 4.1 demonstrate the data acquisition process, while dashed arrows illustrate Augmented Ensemble.

4.2 Data Acquisition

Data valuation is fundamental for data acquisition because we want to invest more in useful data instances for domain adaptation when our acquisition budget is limited. Therefore, in this section, we first design a data valuation function $F(\mathbf{x})$ to evaluate the utility of data

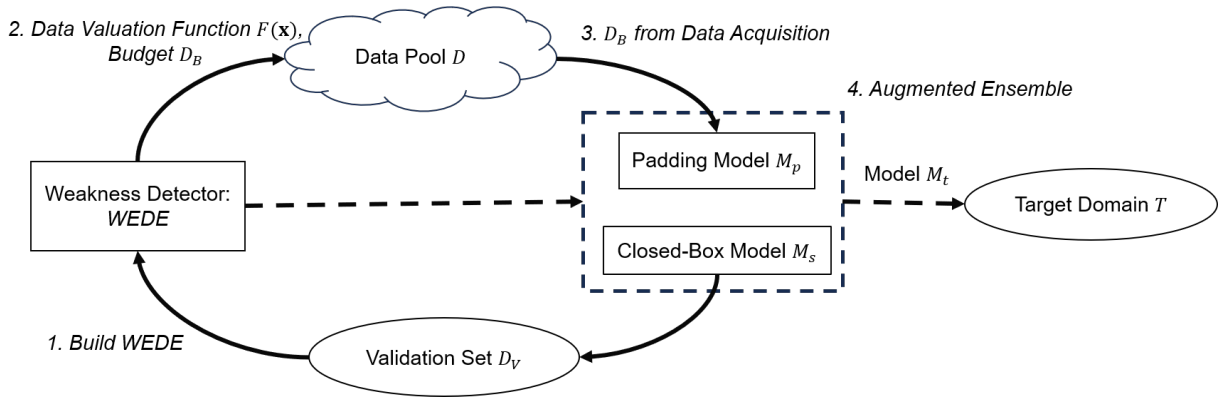

Figure 4.1: Data Acquisition and Augmented Ensemble for Improving a Closed-box Model in a Target Domain

in constructing a padding model $\mathcal{M}_\mathbf{p}$, not necessarily having an identical architecture with $\mathcal{M}_\mathbf{s}$, for where $\mathcal{M}_\mathbf{s}$ performs poorly in the target domain $\mathbf{T}$, and then we introduce two data acquisition strategies using $F(\mathbf{x})$ and budget B.

4.2.1 Data Utility Estimation

We aim to construct $\mathcal{M}_\mathbf{p}$ with extra data to fix the deficiency of $\mathcal{M}_\mathbf{s}$ in $\mathbf{T}$, which is often caused by distribution shifts. An intuitive approach to acquiring such data is to apply $\mathcal{M}_\mathbf{s}$ to the data pool $\mathcal{D}$ and collect $\mathcal{M}_\mathbf{s}$'s incorrect predictions in $\mathcal{D}$. However, due to privacy concerns, customers do not expect their $\mathcal{M}_\mathbf{s}$ to be exposed to $\mathcal{D}$ that belongs to data sellers. Then, to discover the weakness of $\mathcal{M}_\mathbf{s}$ in $\mathcal{D}$ without leaking $\mathcal{M}_\mathbf{s}$, we propose to use a weakness detector WeDe built from $\mathcal{M}_\mathbf{s}$'s prediction results in $\mathcal{D}_V$, a private and labeled dataset from $\mathbf{T}$. Also, if an instance from $\mathcal{D}$ is more likely to be the weakness of $\mathcal{M}_\mathbf{s}$ than the others, this instance should be more important to the construction of $\mathcal{M}_\mathbf{p}$. Based on this intuition, we employ *weakness score* associated with weakness features extracted by WeDe to estimate data utility.

4.2.1.1 Weakness Detector WeDe

To extract the features of $\mathcal{M}_\mathbf{s}$'s weakness in $\mathbf{T}$ and derive the associated *weakness score* $g(\mathbf{x})$ of an instance $\mathbf{x}$ from $\mathcal{D}$, we build a weakness detector WeDe to discriminate correct and incorrect predictions of $\mathcal{M}_\mathbf{s}$ in $\mathcal{D}_V$ as suggested by the research on model vulnerability interpretation [29]. We define incorrect predictions as class c_w while correct predictions as class $c_{\not w}$ and provide two designs of the binary classification task for WeDe.

Support Vector Machine (SVM)-based WeDe. SVM linearly separates data of class c_w and class $c_{\not w}$ in a hyperspace to maximize the margin between data instances from different classes but nearest to the class boundary. The distance of a data instance from the boundary to the side of class c_w is used as the weakness score. For example, in Figure 4.2, the solid arrow indicates the direction in which weakness scores increase. The circle on the boundary has a score of 0.

Logistic Regression-based WeDe. Logistic regression builds a linear function in the data space $\mathbb{R}^n$ to estimate the probability that a point in $\mathbb{R}^n$ belongs to class c_w. Such probabilities are seen as weakness scores.

In summary, the weakness score of a data instance $\mathbf{x}$ from $\mathcal{D}$ is defined as $g(\mathbf{x})$, which is a distance from the class boundary to c_w if WeDe is based on SVM, or a prediction probability of c_w when WeDe is built by Logistics Regression.

4.2.1.2 Weakness Score as Data Utility Estimates

We propose two ways to treat the weakness score $g(\mathbf{x})$ as a utility estimate of $\mathbf{x}$ for training $\mathcal{M}_\mathbf{p}$, either using this score directly (which we call U-WS) or the probability distribution of this score (termed U-WSD). Since computing U-WS is straightforward, we focus our following discussion on the computation of U-WSD. Suppose class c_w (i.e., the data instances that are incorrectly predicted by $\mathcal{M}_\mathbf{s}$) follow a probability distribution characterized by parameters $\theta_\mathbf{w}$ and the distribution of class $c_{\not w}$ is characterized by $\theta_{\not \mathbf{w}}$. Then, the probability

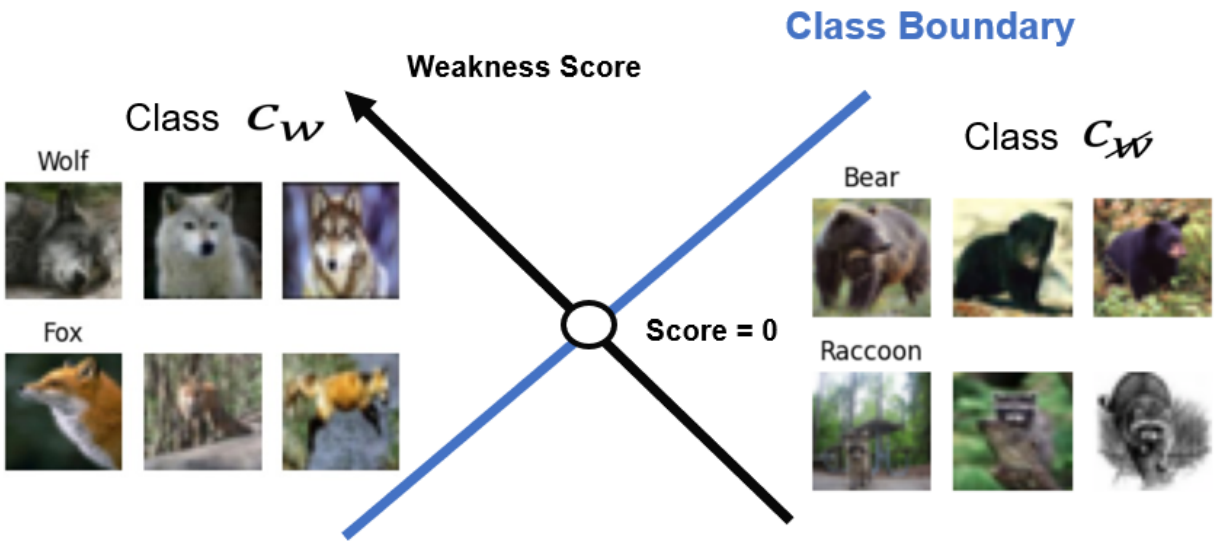

Figure 4.2: Support Vector Machine-based WEDE

According to Figure 1.1, validation images of wolves and foxes categories tend to be incorrectly predicted by $\mathcal{M}_\mathbf{s}$, while those of bears and raccoons are correct predictions. SVM-based WEDE generates weakness scores by distinguishing the two kinds of predictions after image encoding.

that $\mathbf{x}$ is drawn from the weakness distribution $\theta_\mathbf{w}$ of $\mathcal{M}_\mathbf{s}$ is

$$f(\mathbf{x}) = P(\theta_\mathbf{w}|\mathbf{x}) = \frac{P(\mathbf{x}|\theta_\mathbf{w})P(\theta_\mathbf{w})}{P(\mathbf{x}|\theta_\mathbf{w})P(\theta_\mathbf{w}) + P(\mathbf{x}|\theta_{\not\mathbf{w}})P(\theta_{\not\mathbf{w}})}, \tag{4.1}$$

where $P(\theta)$ is the prior probability of a distribution with parameter θ, and $P(\mathbf{x}|\theta)$ is the distribution density on $\mathbf{x}$. To facilitate the computation of Equation 4.1, we represent the distribution θ with a *weakness score distribution* θ' whose density is obtained by kernel density estimation. Figure 4.3 and Figure 4.4 respectively illustrate ${\theta_\mathbf{w}}'$ and ${\theta_{\not\mathbf{w}}}'$, weakness score distributions of c_w and $c_{\not w}$, from SVM-based WEDE and Logistic Regression-based WEDE. Then, Equation 4.1 is rewritten to

$$f(\mathbf{x}) = P({\theta_\mathbf{w}}'|g(\mathbf{x})) = \frac{P(g(\mathbf{x})|{\theta_\mathbf{w}}')P({\theta_\mathbf{w}}')}{P(g(\mathbf{x})|{\theta_\mathbf{w}}')P({\theta_\mathbf{w}}') + P(g(\mathbf{x})|{\theta_{\not\mathbf{w}}}')P({\theta_{\not\mathbf{w}}}')}, \tag{4.2}$$

Hence, $f(\mathbf{x})$ estimates the probability that a data instance $\mathbf{x}$ is the weakness of $\mathcal{M}_\mathbf{s}$. This weakness probability is also the acquisition utility of $\mathbf{x}$ in building $\mathcal{M}_\mathbf{p}$ for enhancing $\mathcal{M}_\mathbf{s}$ in $\mathbf{T}$.

Compared to U-WS, U-WSD gives customers more flexibility in selecting desired data

from $\mathcal{D}$. With U-WSD, customers can set a preferred probability that instances come from the weakness distribution of $\mathcal{M}_\mathbf{s}$.

In summary, WEDE-based data utility estimators, U-WS and U-WSD, serve as the data valuation function $F(\mathbf{x})$ in the interaction with the data seller, i.e.,

$$F(\mathbf{x}) = \begin{cases} g(\mathbf{x}) & \text{if data utility estimator is U-WS,} \\ f(\mathbf{x}) & \text{if data utility estimator is U-WSD} \end{cases} \tag{4.3}$$

For U-WS, the customer only needs to send WEDE to the seller, and the seller applies WEDE to instances in $\mathcal{D}$. For U-WSD, the customer has to send the weakness probability estimation function f (including the weakness score distribution parameters θ_w and $\theta_{\not w}$) in addition to WEDE. Both U-WS and U-WSD prevent leaking the privacy and usage of $\mathcal{M}_\mathbf{s}$ by hiding $\mathcal{M}_\mathbf{s}$ from the data seller.

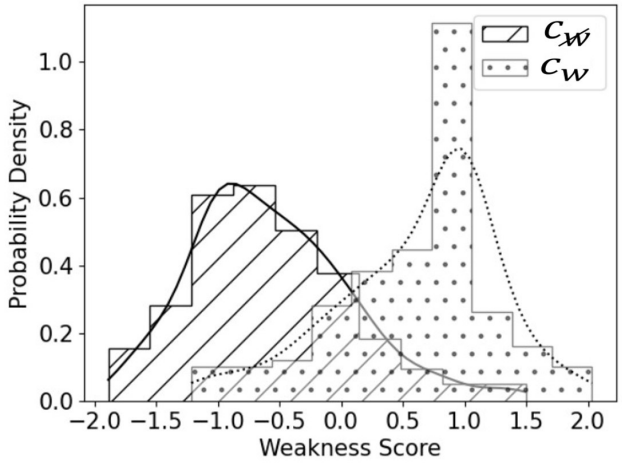

Figure 4.3: Distribution of Weakness Score from SVM-based WEDE

This weakness score distribution is generated by SVM-based WEDE in the validation set in the 1st run of experiments of the *Core-Session* task.

4.2.2 Acquisition Strategy

The customer acquires data in a one-shot or a sequential manner. The *One-shot Strategy* returns $\mathcal{D}_B$ from $\mathcal{D}$ by using up budget B at once. On the contrary, the *Sequential Strategy*

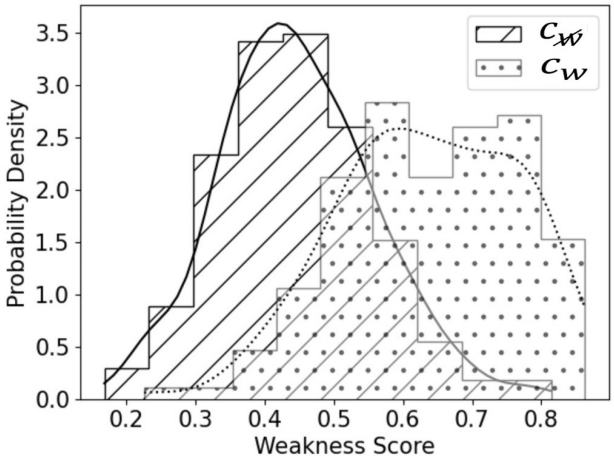

Figure 4.4: Distribution of Weakness Score from Logistic Regression-based WeDe
This weakness score distribution is generated by Logistic Regression-based WeDe in the validation set in the 10th run of experiments of the *Core-Session* task.

operates by multiple rounds. The i-th acquisition round consumes budget b_i, which is part of the total budget B and $\sum_i b_i \leq B$. After each round, WeDe is retrained on all the data instances that have been acquired so far, as well as the validation set $\mathcal{D}_V$. That is, the training data of WeDe after the i-th acquisition round is $\mathcal{D}_V \cup \{\mathcal{D}_{b_j}\}, j \in [0, i]$. Regardless of the acquisition strategy, after B is exhausted, all the acquired data instances $\mathcal{D}_B$ are used to train $\mathcal{M}_{\mathbf{p}}$. The logic of the *Sequential Strategy* is illustrated in Figure 4.5.

4.3 Augmented Ensemble

We build a model ensemble $\mathcal{M}_{\mathbf{t}}$ with $\mathcal{M}_{\mathbf{s}}$ and $\mathcal{M}_{\mathbf{p}}$ for better accuracy in the target domain $\mathbf{T}$ where labels are released with delays. Compared to only applying $\mathcal{M}_{\mathbf{s}}$ to $\mathbf{T}$, $\mathcal{M}_{\mathbf{t}}$ achieves a better performance in $\mathbf{T}$ by including $\mathcal{M}_{\mathbf{p}}$ that is specialized in addressing the prediction weakness of $\mathcal{M}_{\mathbf{s}}$ in $\mathbf{T}$.

For this purpose, we propose a method called Augmented Ensemble (AE) to aggregate predictions from $\mathcal{M}_{\mathbf{s}}$ and $\mathcal{M}_{\mathbf{p}}$ and produce the output of $\mathcal{M}_{\mathbf{t}}$ under two ubiquitous application scenarios: (1) $\mathcal{M}_{\mathbf{s}}$ outputs only *hard labels*, which are predicted class labels for inputs, and (2) in addition to hard labels, $\mathcal{M}_{\mathbf{s}}$ also returns *soft labels*, which are prediction proba-

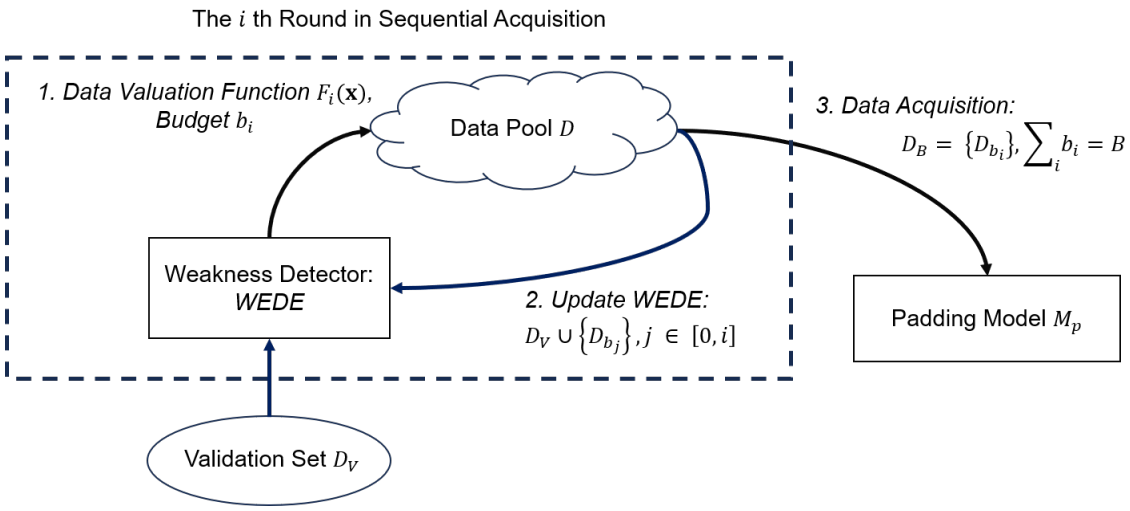

Figure 4.5: Sequential Strategy

bilities for every class. $\mathcal{M}_\mathbf{p}$ always gives customers both hard and soft labels, because it is built by customers rather than purchased as a closed box from the ML market.

In the hard-label-only scenario, we can only use the predicted labels from $\mathcal{M}_\mathbf{s}$ and have to decide to trust $\mathcal{M}_\mathbf{s}$ or $\mathcal{M}_\mathbf{p}$ when their predictions are contradictory. To best exploit the strength of each model, we aim to assign $\mathcal{M}_\mathbf{s}$'s weaknesses in $\mathbf{T}$ to $\mathcal{M}_\mathbf{p}$ for prediction while leaving others to $\mathcal{M}_\mathbf{s}$. Then, we design weakness-probability-based and data-valuation-based criteria for sending data from $\mathbf{T}$ to $\mathcal{M}_\mathbf{s}$ or $\mathcal{M}_\mathbf{p}$. If a target data instance meets a criterion, the prediction of this instance will solely rely on $\mathcal{M}_\mathbf{p}$, otherwise $\mathcal{M}_\mathbf{s}$. We term the AE strategy with criteria for target data assignment as *AE-C*.

When soft labels are also available from $\mathcal{M}_\mathbf{s}$, AE-C is still applicable. However, given that soft labels tell us the prediction probability of $\mathcal{M}_\mathbf{s}$ on every class, we also weight soft labels from $\mathcal{M}_\mathbf{s}$ and $\mathcal{M}_\mathbf{p}$ by each model's anticipated contribution to dealing with $\mathcal{M}_\mathbf{s}$'s weaknesses in $\mathbf{T}$. Weighted soft labels are then aggregated to produce the final predicted label. Given the characteristic of WEDE, we also design two ways of calculating weights for soft label combinations. We name the AE strategy with weights for prediction aggregation as *AE-W*.

In this section, we first introduce the AE-C strategy with two designs of criteria for assigning target data to $\mathcal{M}_\mathbf{s}$ or $\mathcal{M}_\mathbf{p}$ for prediction. AE-C is a general practice whether soft

labels of $\mathcal{M}_\mathbf{s}$ are accessible or not. Next, we describe the AE-W strategy with sophisticated designs of weights in aggregating soft labels from $\mathcal{M}_\mathbf{s}$ and $\mathcal{M}_\mathbf{p}$.

4.3.1 Augmented Ensemble with Criteria for Target Data Assignment

When $\mathcal{M}_\mathbf{s}$ only returns hard labels, the model ensemble $\mathcal{M}_\mathbf{t}$ outputs a predicted class label $h_t(\mathbf{z})$ of a target data instance $\mathbf{z}$ solely based on the prediction result $h_p(\mathbf{z})$ or $h_s(\mathbf{z})$ from $\mathcal{M}_\mathbf{p}$ and $\mathcal{M}_\mathbf{s}$ respectively. Since $\mathcal{M}_\mathbf{p}$ is designed for the weakness of $\mathcal{M}_\mathbf{s}$ in the target domain, it is natural to select $h_p(\mathbf{z})$ as $h_t(\mathbf{z})$ if $\mathbf{z}$ is likely to be incorrectly predicted by $\mathcal{M}_\mathbf{s}$. To identify such target data for $\mathcal{M}_\mathbf{p}$, we employ a criterion based on the weakness probability, which we call *AE-C-WP*. Besides, $\mathcal{M}_\mathbf{p}$ is also reliable when $\mathbf{z}$ comes from the distribution of its training data, which is $\mathcal{D}_B$ acquired from $\mathcal{D}$ under a budget B. To evaluate the relation between $\mathbf{z}$ and $\mathcal{D}_B$, we apply the data valuation function $F(\mathbf{x})$ to $\mathbf{z}$ and $\mathcal{D}_B$, and then compare the valuation result of $\mathbf{z}$ with the smallest value from $\mathcal{D}_B$. We term this AE-C strategy as *AE-C-DV*. More details of the two AE-C strategies are in the following sections.

4.3.1.1 Augmented Ensemble with Weakness Probability-based Criterion (AE-C-WP)

We use the weakness probability estimation function $f(\mathbf{x})$ from Equation 4.2 to calculate how likely $\mathbf{z}$ is to be misclassified by $\mathcal{M}_\mathbf{s}$. If the probability $f(\mathbf{z})$ is not less than a given criterion c, the prediction of $\mathcal{M}_\mathbf{p}$ on $\mathbf{z}$, $h_p(\mathbf{z})$, is considered as the final decision $h_t(\mathbf{z})$, i.e.,

$$h_t(\mathbf{z}) = \begin{cases} h_p(\mathbf{z}) & \text{if } f(\mathbf{z}) \geq c \\ h_s(\mathbf{z}) & \text{if } f(\mathbf{z}) < c \end{cases} \tag{4.4}$$

Criterion c is set manually. For example, customers can perform a grid search of c in a validation set to select c with the best result for AE-C-WP.

When WEDE is implemented with Logistic Regression, the weakness scores $g_{\text{logistic}}(\mathbf{z})$ of $\mathbf{z}$ is also a probability that $\mathbf{z}$ belongs to $\mathcal{M}_\mathbf{s}$'s weaknesses c_w. Hence, for Logistic Regression-

based WeDe, the AE-C-WP mechanism can be rewritten to

$$h_t(\mathbf{z}) = \begin{cases} h_p(\mathbf{z}) & \text{if } g_{\text{logistic}}(\mathbf{z}) \geq c \\ h_s(\mathbf{z}) & \text{if } g_{\text{logistic}}(\mathbf{z}) < c \end{cases} \tag{4.5}$$

4.3.1.2 Augmented Ensemble with Data Valuation-based Criterion (AE-C-DV)

We can estimate the distribution of $\mathcal{M}_\mathbf{p}$'s training data $\mathcal{D}_B$ by the range of its valuation results returned by the function $F(\mathbf{x})$. All the data within $\mathcal{D}_B$'s distribution should be no less than the smallest value c in $\mathcal{D}_B$. Then, if $F(\mathbf{z})$ is at least c, we assume $\mathbf{z}$ falls in $\mathcal{D}_B$'s distribution, and we take $h_p(\mathbf{z})$ as the final prediction $h_t(\mathbf{z})$, i.e.,

$$h_t(\mathbf{z}) = \begin{cases} h_p(\mathbf{z}) & \text{if } F(\mathbf{z}) \geq c \\ h_s(\mathbf{z}) & \text{if } F(\mathbf{z}) < c \end{cases}, c = \min(F(d_i)), d_i \in \mathcal{D}_B \tag{4.6}$$

4.3.2 Augmented Ensemble with Weights for Prediction Aggregation

When $\mathcal{M}_\mathbf{s}$ provides prediction probabilities of every class as soft labels for a target data instance $\mathbf{z}$, as well as hard labels, we propose to aggregate soft labels from both $\mathcal{M}_\mathbf{s}$ and $\mathcal{M}_\mathbf{p}$ to produce $h_t(\mathbf{z})$ as the predicted label from the ensemble $\mathcal{M}_\mathbf{t}$. We denote the soft label of class k from $\mathcal{M}_\mathbf{s}$ and $\mathcal{M}_\mathbf{p}$ on a given $\mathbf{z}$ as $p_{\text{s},k}(\mathbf{z})$ and $p_{\text{p},k}(\mathbf{z})$, respectively. In the aggregation of $p_{\text{s},k}(\mathbf{z})$ and $p_{\text{p},k}(\mathbf{z})$, $p_{\text{p},k}(\mathbf{z})$ should be attached more importance than $p_{\text{s},k}(\mathbf{z})$ when $\mathcal{M}_\mathbf{s}$ is not expected to predict $\mathbf{z}$ correctly. As such, we consider the weight of $p_{\text{p},k}(\mathbf{z})$ in the soft-label aggregation as the probability that $\mathbf{z}$ is from the distribution of $\mathcal{M}_\mathbf{s}$'s weaknesses. Then, we use the weakness probability estimation function $f(\mathbf{z})$ from Equation 4.2 to approximate the weight of $p_{\text{p},k}(\mathbf{z})$, and $1 - f(\mathbf{z})$ for the weight of $p_{\text{s},k}(\mathbf{z})$, and obtain the prediction probability of $\mathbf{z}$ belonging to class j generated by $\mathcal{M}_\mathbf{t}$, i.e.,

$$p_{\text{t},k}(\mathbf{z}) = (1 - f(\mathbf{z}))p_{\text{s},k}(\mathbf{z}) + f(\mathbf{z})p_{\text{p},k}(\mathbf{z}). \tag{4.7}$$

The final classification decision from $\mathcal{M}_\mathbf{t}$, $h_t(\mathbf{z})$, is the class k with the highest probability generated by Equation 4.7, i.e., $h_t(\mathbf{z}) = \text{argmax}_k p_{\text{t},k}(\mathbf{z})$.

Besides, since Logistic Regression-based WEDE directly outputs probabilities of c_w as weakness scores $g_{\text{logistic}}(\mathbf{z})$, we can replace $f(\mathbf{z})$ with $g_{\text{logistic}}(\mathbf{z})$ in Equation 4.7, and then we have

$$p_{\text{t},k}(\mathbf{z}) = (1 - g_{\text{logistic}}(\mathbf{z}))p_{\text{s},k}(\mathbf{z}) + g_{\text{logistic}}(\mathbf{z})p_{\text{p},k}(\mathbf{z}). \tag{4.8}$$

However, compared to Equation 4.7, Equation 4.8 is only applied to combining $\mathcal{M}_{\mathbf{s}}$ and $\mathcal{M}_{\mathbf{p}}$ after data acquisition by Logistic Regression-based WEDE.

5 Experiments

We test our proposed framework consisting of WeDe-based data acquisition and Augmented Ensemble (AE) for combining the source closed-box model $\mathcal{M}_{\mathbf{s}}$ and the padding model $\mathcal{M}_{\mathbf{p}}$ on various image classification tasks, and compare our framework with applicable baselines. We also empirically analyze factors that affect the performance of our framework. All the experiments are repeated ten times, and the average results are reported. Experiment code and configurations are available at our GitHub repository: https://github.com/t07902301/data-acquisition.

5.1 Experiment Setup

5.1.1 Dataset

Our experiments use the datasets *Core-50* [41] and *Cifar-100* [35]. Table 5.2 shows a label hierarchy of the two datasets. Images in *Core-50* are grouped into ten categories, each with five object classes and 11 session classes. For example, a category of *ball* in *Core-50* can be divided by objects like tennis balls and footballs, or by sessions with indoor or dim backgrounds. *Cifar-100* is an image set of animals from 20 super-classes, each with five sub-classes. For example, a superclass of *tree* has some subclasses like *maple tree* and *pine tree*. From *Core-50* and *Cifar-100*, we set up multi-class classification tasks: *Core-Object*, *Core-Session*, *Cifar-20-Class*, and *Cifar-4-Class*. *Core-Object* and *Core-Session* contain ten categories. *Cifar-20-Class* involves all 20 superclasses, while *Cifar-4-Class* only has 4 superclasses of animals. For each task, we first split the dataset into four parts: (1) a training set for building model $\mathcal{M}_{\mathbf{s}}$, (2) a data pool for data acquisition $\mathcal{D}$, (3) a validation

Task \ Dataset	Validation Set	Test Set	Data Pool
Core-Session	550	1100	10450
Core-Object	550	1100	10450
Cifar-20-Class	2000	8000	30000
Cifar-4-Class	500	500	6000

Table 5.1: Dataset Size

set $\mathcal{D}_V$, and (4) a test set for model evaluation. Table 5.1 exhibits the size of validation, test set, and the data pool in each task.

To evaluate our proposal, we create a target domain by simulating data distribution shifts from the source domain. To this end, for each task, we remove data of predefined labels in level 2 of the hierarchy from the training set of $\mathcal{M}_{\mathbf{s}}$ but leave other datasets untouched. This way, the target domain, from which parts (2)-(4) are drawn, is expected to have a "wider" distribution (with more *level-2* labels from the hierarchy) than the source domain, to which part (1) belongs. Specifically, for *Core-Object*, we remove one object class from each of the ten categories; for *Core-Session*, we remove one session class from each category; for *Cifar-4-Class*, we remove two subclasses from each superclass; for *Cifar-20-Class*, one subclass is removed from each superclass. We further investigate the impact of distribution shift by varying the number of *level-2* labels removed from the datasets in subsection 5.4.2.

Dataset	Level 1	Level 2	
Core-50	Category:10	Object: 5	Session: 11
Cifar-100	Superclass: 20	Subclass: 5	

Table 5.2: Dataset Label Hierarchy

5.1.2 Model and WeDe Implementation

For both $\mathcal{M}_{\mathbf{s}}$ and $\mathcal{M}_{\mathbf{p}}$, we adopt Squeezenet [27] in *Core-Session* and *Core-Object*. To save model training time in *Cifar-4-Class* and *Cifar-20-Class*, we add a linear classification layer

after feature extraction by a pretrained Resnet20 [14]. We train all models with 20 epochs and return the training checkpoint with the smallest loss in the validation set $\mathcal{D}_V$. SGD with 0.9 momentum and 5 x 10^{-4} weight decay is used to optimize model parameters. The batch size of $\mathcal{M}_\mathbf{p}$'s training data is 256 for *Cifar-20-Class* and 64 for other tasks. We adjust the learning rate for model parameter optimization batch-by-batch. It increases linearly from 0 to 0.1 by the end of the 10th epoch and then linearly declines to 0 at the end of training.

We determine the class boundary in SVM-based WEDE by the *SVC* optimization package for task *Core-Session*, *Core-Object*, and *Cifar-4-Class*, while the *LinearSVC* package for *Cifar-4-Class* with a much larger dataset. We also perform a grid search to determine the best hyperparameters in constructing WEDE.

5.1.3 Evaluation Metrics

Model Ensemble $\mathcal{M}_\mathbf{t}$. We use *accuracy improvement* to measure acquisition performance. It is defined as the classification accuracy of $\mathcal{M}_\mathbf{t}$ on the test set minus the accuracy of $\mathcal{M}_\mathbf{s}$ on the same dataset, where classification accuracy refers to the number of correct predictions divided by the total number of predictions.

WEDE. By the confusion matrix in Table 5.3, we define ***Weakness Detection Accuracy*** as how many weaknesses detected by WEDE actually belong to c_w. This metric is computed as

$$\frac{\text{True } c_w}{\text{True } c_w + \text{False } c_w}$$

.

Truth \ Prediction	c_w	$c_{\not w}$
c_w	True c_w	False $c_{\not w}$
$c_{\not w}$	False c_w	True $c_{\not w}$

Table 5.3: Confusion Matrix of WEDE

5.1.4 Environment

Our experiment uses an NVIDIA RTX A5000 GPU and Pytorch 1.12.1 for model training and testing, and scikit-learn 1.3.0 for building SVM-based and Logistic Regression-based WeDe. The vision transformer *ViT-B/32* from OpenAI [50] encodes images into input vectors for WeDe. We adopt Scipy 1.10.1 for kernel density estimation in weakness probability function $f(\mathbf{x})$. We use 0 to set the seed for all random number generators in our experiments.

5.2 Baselines

We implement four acquisition methods and one alternative model ensemble approach as baselines.

Random Acquisition: We randomly sample instances from the data pool $\mathcal{D}$ within budget B and use these samples to build $\mathcal{M}_\mathbf{p}$. This is equivalent to setting $F(\mathbf{x})$ as a random number generator in data acquisition. The training data of $\mathcal{M}_\mathbf{p}$ is expected to be from the same distribution as the test set, and thus we only apply $\mathcal{M}_\mathbf{p}$ to the test set. Random acquisition examines the necessity of creating a model ensemble from $\mathcal{M}_\mathbf{s}$ and $\mathcal{M}_\mathbf{p}$.

Probability-at-Ground-Truth (PGT) Acquisition: We select labeled instances from $\mathcal{D}$ with the lowest probabilities on their ground-truth labels from $\mathcal{M}_\mathbf{s}$. $F(\mathbf{x})$ for a labeled instance $\mathbf{x}$ is set as the negative value of $\mathcal{M}_\mathbf{s}$'s prediction probability on the true label of $\mathbf{x}$. This baseline returns the most extreme weaknesses of $\mathcal{M}_\mathbf{s}$.

Random Weakness Acquisition: Instead of sampling from the entire data pool, this baseline randomly draws $\mathcal{D}_B$ from incorrect predictions of $\mathcal{M}_\mathbf{s}$ in $\mathcal{D}$. $F(\mathbf{x})$ in this baseline has two components: $F_1(\mathbf{x})$ for collecting prediction mistakes of $\mathcal{M}_\mathbf{s}$ in $\mathcal{D}$ and a random number generator $F_2(\mathbf{x})$ for sampling from the results of $F_1(\mathbf{x})$.

PGT acquisition and *random weakness acquisition* select the weakness of $\mathcal{M}_\mathbf{s}$ differently from our WeDe-based method. After these two acquisition, we also apply $\mathcal{M}_\mathbf{s}$ and $\mathcal{M}_\mathbf{p}$ to the test set in conjunction with AE.

Entropy-based Acquisition: We consider the high prediction entropy of $\mathcal{M}_\mathbf{s}$ as an

alternative exhibition of $\mathcal{M}_\mathbf{s}$'s weakness. $F(\mathbf{x})$ equals the prediction entropy computed by $-\sum_k p_{\mathrm{s},k}(\mathbf{x})\log_2(p_{\mathrm{s},k}(\mathbf{x}))$, where $p_{\mathrm{s},k}(\mathbf{x})$ is $\mathcal{M}_\mathbf{s}$'s prediction probability of class k given an input $\mathbf{x}$. $\mathcal{M}_\mathbf{p}$ from this baseline is tested with AE.

PGT, *Random Weakness*, and *Entropy-based* acquisition baselines are all hypothetical, as they require customers to provide data sellers with $\mathcal{M}_\mathbf{s}$ for data valuation. But customers prefer to keep their purchased source model$\mathcal{M}_\mathbf{s}$ private and protect the privacy and usage of their source models.

Average Ensemble: Unlike dynamic estimation of aggregation weight in AE-W (subsection 4.3.2), weights of both $p_{\mathrm{s},j}(\mathbf{x})$ and $p_{\mathrm{p},j}(\mathbf{x})$ are set to be 0.5 on any test data.

5.3 Results

We first present how distribution shifts strongly affect the performance of $\mathcal{M}_\mathbf{s}$ in the test set. Then we compare data utility estimation options which vary in the design WeDe and utility estimators. We analyze Augmented Ensemble under two scenarios regarding the availability of soft labels from $\mathcal{M}_\mathbf{s}$: (1) *hard-label-only*: $\mathcal{M}_\mathbf{s}$ does not provide customers with soft labels of predictions. (2) *soft-label*: customers can access soft labels from $\mathcal{M}_\mathbf{s}$.

Next, we discuss the comparison of data acquisition strategies, one-shot and sequential WeDe-based methods, and three baselines, and provide an in-depth analysis of the two WeDe-based strategies. When implementing the sequential strategy, we set two acquisition rounds for *Core-Object* and *Core-Session* and three for *Cifar-20-Class*. Each round is assigned the same budget. Also, for sequential acquisition, we use the final WeDe in AE.

At last, we investigate how the complexity of the padding model, degree of distribution shift, and distribution of the data pool $\mathcal{D}$ impact the performance of our proposed framework.

5.3.1 Impact of Distribution Shifts

As we can see from statistics in Table 5.4, classification accuracy significantly drops after distribution shifts, or data of new *level-2* labels, are introduced to the test set across all

three tasks. Distribution shifts account for at least 20% of the test set, but they occupy around half of the misclassifications of $\mathcal{M}_\mathbf{s}$ in the test set. These statistics demonstrate that our generated distribution shifts in the test set strongly affect the performance of $\mathcal{M}_\mathbf{s}$.

Task	Accuracy Before and After Distribution Shifts	Distribution Shifts in the Test Set	Misclassifications from Distribution Shifts
Cifar-20-Class	74.386% → 67.012%	20%	37.886%
Cifar-4-Class	81.467% → 61.16%	40%	71.434%
Core-Object	86.24% → 74.42%	20%	57%
Core-Session	86.96% → 74%	33%	63.5%

Table 5.4: Statistics on Distribution Shifts

5.3.2 Data Utility Estimation

In this section, we first evaluate the difference between SVM and Logistic Regression in classifying class c_w and $c_{\not w}$ in WEDE, as well as their acquisition results from one-shot and sequential strategies. Then, we compare U-WS and U-WSD in estimating data utility for building the padding model $\mathcal{M}_\mathbf{p}$ to enhance the source model $\mathcal{M}_\mathbf{s}$ for the target domain. We adopt AE-C-WP with a criterion of 0.5 to create model ensembles in the hard-label-only scenario.

5.3.2.1 SVM vs. Logistic Regression as WEDE

We first compare the weakness detection accuracy of SVM and Logistic Regression as WEDE in Figure 5.1. SVM and Logistic Regression exhibit similar performance in distinguishing classes c_w and $c_{\not w}$. After the sequential acquisition, they are close in terms of detecting $\mathcal{M}_\mathbf{s}$'s weaknesses as shown in Figure 5.2.

Then, we examine the accuracy improvement of the target model $\mathcal{M}_\mathbf{t}$ built from data acquired by these two implementations of WEDE and associated U-WS-based utility estimation. For Logistic Regression-based WEDE, we test both the weakness probability

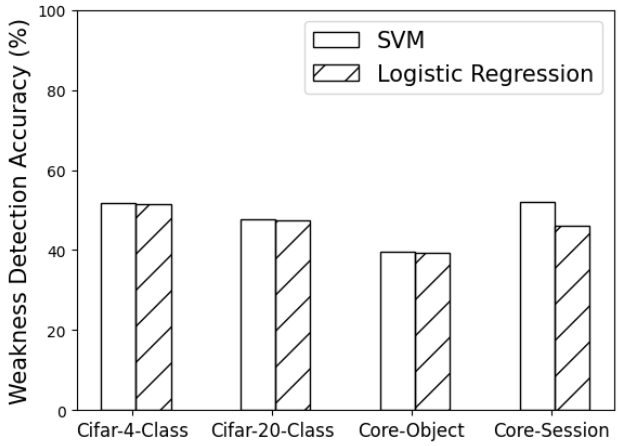

Figure 5.1: Weakness Detection Accuracy of Two WEDE Designs

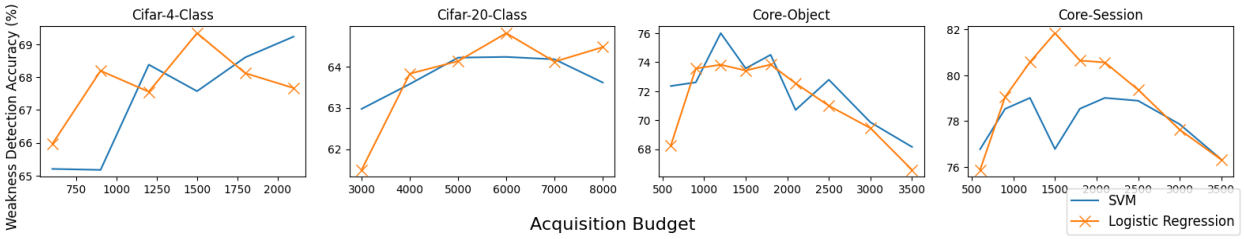

Figure 5.2: Weakness Detection Accuracy of Two WEDE Designs after Sequential Acquisition

estimation function $f(\mathbf{x})$ and the weakness score function $g(\mathbf{x})$ as weights in AE-W (Equation 4.7 , Equation 4.8) or probabilities in AE-C-WP (Equation 4.4, Equation 4.5). Results are shown in Figure 5.3, Figure 5.4,Figure 5.5 and Figure 5.6. We observe that in either application scenario, there is a negligible difference between SVM and Logistic Regression in implementing effective WEDE for the construction of $\mathcal{M}_\mathbf{t}$ for the test set. Therefore, SVM and Logistic Regression have similar functionality as WEDE for data acquisition.

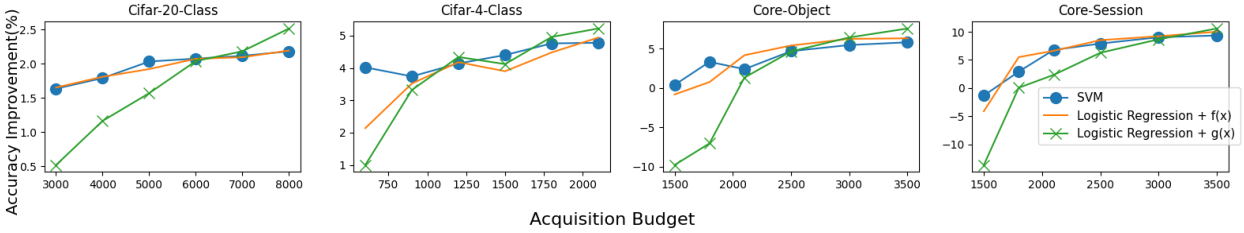

Figure 5.3: One-Shot Acquisition with Two WEDE Designs in the Hard-Label-Only Scenario

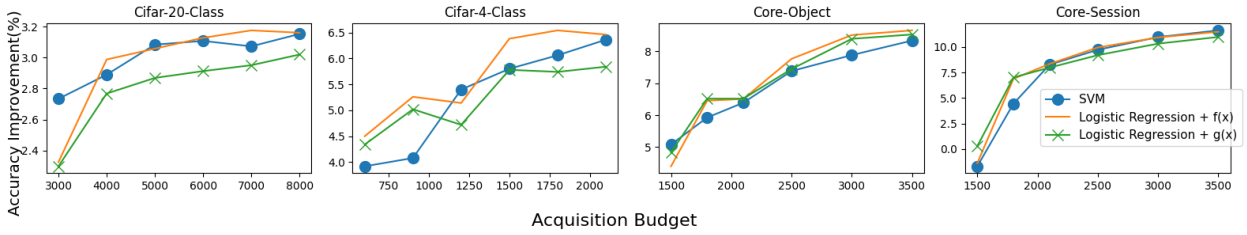

Figure 5.4: Sequential Acquisition with Two WEDE Designs in the Hard-Label-Only Scenario

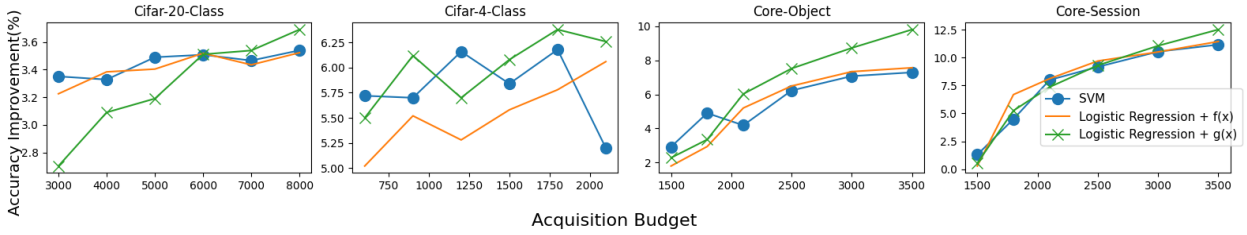

Figure 5.5: One-Shot Acquisition with Two WEDE Designs in the Soft-Label Scenario

5.3.2.2 U-WS vs. U-WSD as Data Utility Estimator

First, to find out the relation between U-WS and U-WSD, we sort data instances by their utility from each estimator in descending order and select instances of the top 100 utilities from $\mathcal{D}$. Then, we compare the utilities of selected instances with those returned by the counterpart estimator. In Figure 5.7, we display the top 100 weakness scores and their corresponding weakness probabilities, and we observe that those probabilities are almost the top 100 as well, which means data collected by U-WS and U-WSD is largely overlapped, despite the two estimators work in different ways.

Accordingly, from Figure 5.10 , Figure 5.11, Figure 5.8, and Figure 5.9, we can conclude that there is not much difference between the two data utility estimators U-WS and U-WSD in improving classification accuracy of $\mathcal{M}_{\mathbf{t}}$ in the test set with either one-shot or sequential acquisition strategy.

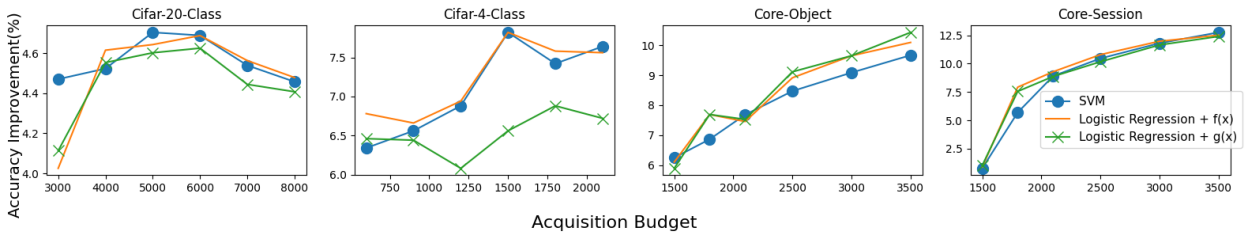

Figure 5.6: Sequential Acquisition with Two WeDe Designs in the Soft-Label Scenario

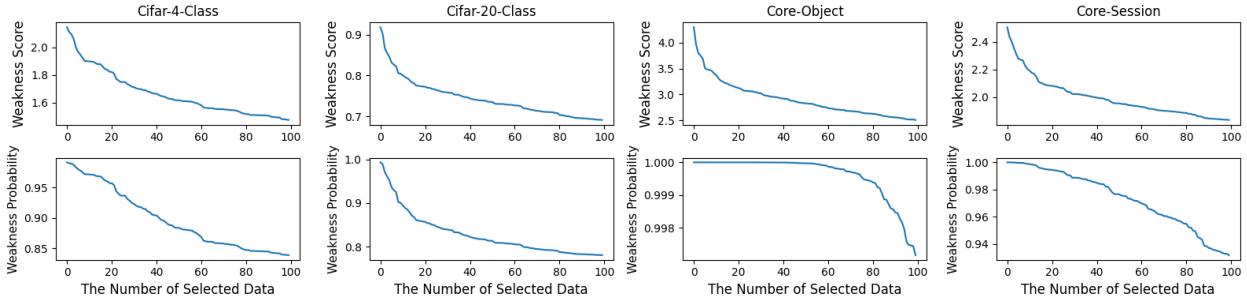

Figure 5.7: Descending Weakness Scores and Corresponding Weakness Probability

Instances with top 100 weakness scores are chosen from the data pool of the 1st run of experiments. Weakness probabilities of selected instances are shown on the bottom row.

5.3.2.3 Summary

After a comprehensive comparison of SVM and Logistic Regress as WeDe, as well as U-WS and U-WSD as data utility estimators, we notice that different designs of WeDe or data utility estimators all lead to effective data acquisition. Then in the following experiments, we only adopt U-WS with SVM-based WeDe to estimate data utility.

5.3.3 Augmented Ensemble

This section compares different AE strategies under the hard-label-only and the soft-label scenario and demonstrates the necessity of using AE after our proposed data acquisition.

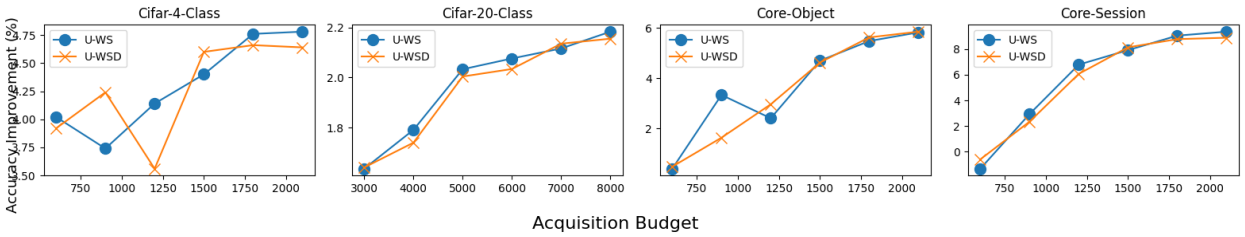

Figure 5.8: One-Shot Acquisition with Two Data Utility Estimation Options in the Hard-Label-Only Scenario

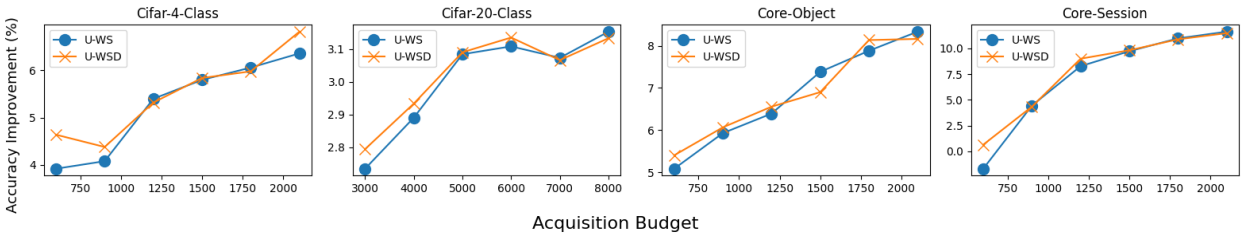

Figure 5.9: Sequential Acquisition with Two Data Utility Estimation Options in the Hard-Label-Only Scenario

5.3.3.1 Hard-Label-Only Scenario: AE-C-WP vs. AE-C-DV

Criteria in AE-C-WP can be adjusted straightforwardly since they are probabilities in terms of $\mathcal{M}_\mathbf{s}$'s weakness. We vary the criterion in AE-C-WP from 0.4 to 0.8, explore its influence in AE-C-WP, and compare AE-C-WP with AE-C-DV.

As shown in Figure 5.12, when the one-shot acquisition is adopted, AE-C-DV is close to AE-C-WP with the best accuracy improvement. But in the case of sequential acquisition, we observe that except *Cifar-20-Class*, AE-C-DV extremely underperforms AE-C-WP with any criterion (Figure 5.13). Regarding the effect of criteria on AE-C-WP, it can be observed that criterion 0.4 leads to better accuracy improvement than higher criteria across all classification tasks whichever acquisition strategy is used. Besides, we show the range of results from benchmark test data assignment when $\mathcal{M}_\mathbf{p}$ is built from one-shot acquired data. All the misclassifications of $\mathcal{M}_\mathbf{s}$ in the test set are sent to $\mathcal{M}_\mathbf{p}$, while others are left for $\mathcal{M}_\mathbf{s}$. This test data assignment benchmark does not waste any functionality of $\mathcal{M}_\mathbf{s}$ or $\mathcal{M}_\mathbf{p}$ in the test set, but it is only hypothetical because we do not know $\mathcal{M}_\mathbf{s}$'s misclassifications right after

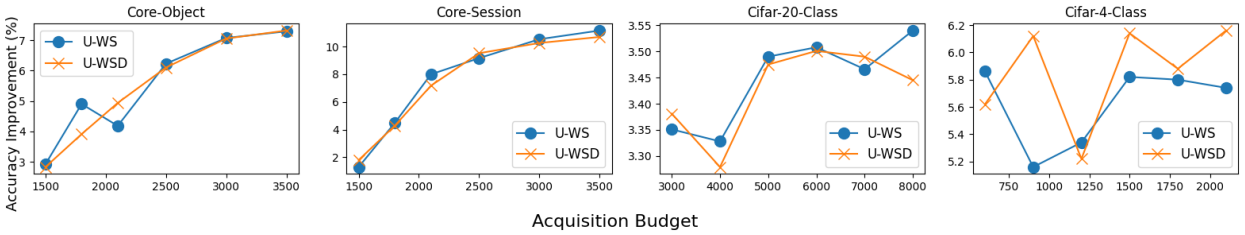

Figure 5.10: One-Shot Acquisition with Two Data Utility Estimation Options in the Soft-Label Scenario

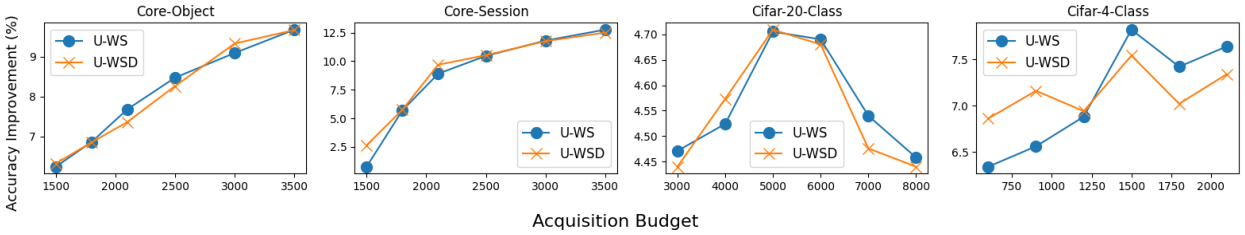

Figure 5.11: Sequential Acquisition with Two Data Utility Estimation Options in the Soft-Label Scenario

model deployment where labels of data are always revealed in delays.

Next, we investigate the poor performance of AE-C-DV when it is associated with sequential acquisition in task *Core-Session*, *Core-Object*, and *Cifar-4-Class*, and then we discuss the influence of criteria on AE-C-WP.

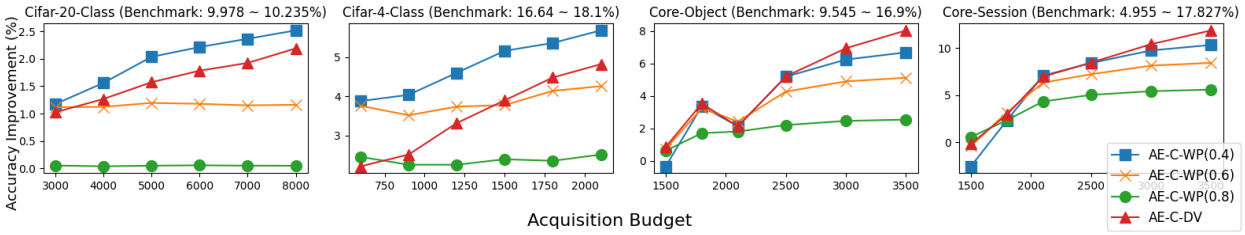

Figure 5.12: One-Shot Acquisition with AE-C in Hard-Label-Only Scenario

Given the specialization of $\mathcal{M}_\mathbf{p}$, AE-C aims to make $\mathcal{M}_\mathbf{p}$ work on more $\mathcal{M}_\mathbf{s}$'s weaknesses but fewer non-weaknesses. Then, to understand why the performance of AE-C-DV drops a lot after the sequential acquisition, we check how much test data misclassified by $\mathcal{M}_\mathbf{s}$ goes to $\mathcal{M}_\mathbf{p}$ under AE-C-DV and AE-C-WP with a criterion of 0.4. In Figure 5.14, we show the

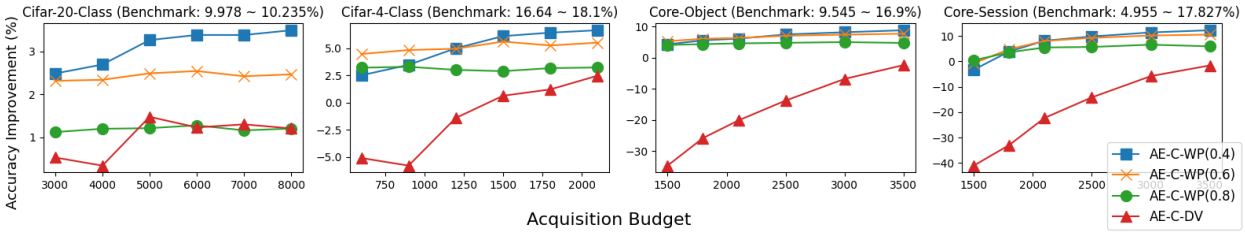

Figure 5.13: Sequential Acquisition with AE-C in Hard-Label-Only Scenario

percentage of $\mathcal{M}_\mathbf{s}$'s weaknesses assigned to test $\mathcal{M}_\mathbf{p}$ after sequential acquisition. Similarly, we discover how many $\mathcal{M}_\mathbf{s}$'s non-weaknesses are assigned to test $\mathcal{M}_\mathbf{p}$ in Figure 5.15. AE-C-DV assigns almost all the test data to $\mathcal{M}_\mathbf{p}$, which severely hurts the leverage of both $\mathcal{M}_\mathbf{p}$ and $\mathcal{M}_\mathbf{s}$.

Such extreme test data misassignment can be caused by an over-fitted WeDe after the sequential acquisition. Figure 5.16 compares the weakness detection accuracy of WeDe in the test set and $\mathcal{D}_B$ which is part of the training set for WeDe. WeDe shows much higher weakness detection accuracy in $\mathcal{D}_B$ than the test set, which means WeDe fits too closely to $\mathcal{D}_B$ and cannot generalize well to other datasets. Moreover, AE-C-DV applies the data valuation function $F(\mathbf{x})$ associated with over-fitted WeDe in $\mathcal{D}_B$ to generate a criterion for test data assignment, while $F(\mathbf{x})$ works less effectively in the test set. Criteria predetermined from $\mathcal{D}_B$ by $F(\mathbf{x})$ then mislead test data assignment in AE-C-DV.

For AE-C-WP, the target data assignment criterion is a controllable weakness probability. As indicated in Figure 5.17, although a higher criterion of AE-C-WP gives $\mathcal{M}_\mathbf{p}$ test data with a larger concentration on $\mathcal{M}_\mathbf{s}$'s weaknesses c_w, it also misses more $\mathcal{M}_\mathbf{s}$'s weaknesses c_w to test $\mathcal{M}_\mathbf{p}$. But a low criterion also includes too many non-weaknesses of $\mathcal{M}_\mathbf{s}$. Hence, in actual applications, customers may need preliminary trials to select a suitable criterion for AE-C-WP.

In summary, when data for building the padding model is acquired by one-shot, it is safe to use either AE-C-DV or AE-C-WP to produce the model ensemble, but AE-C-WP is more robust to sequential acquisition than AE-C-DV. In subsequent experiments, we only adopt AE-C-WP to implement the AE-C model ensemble method.

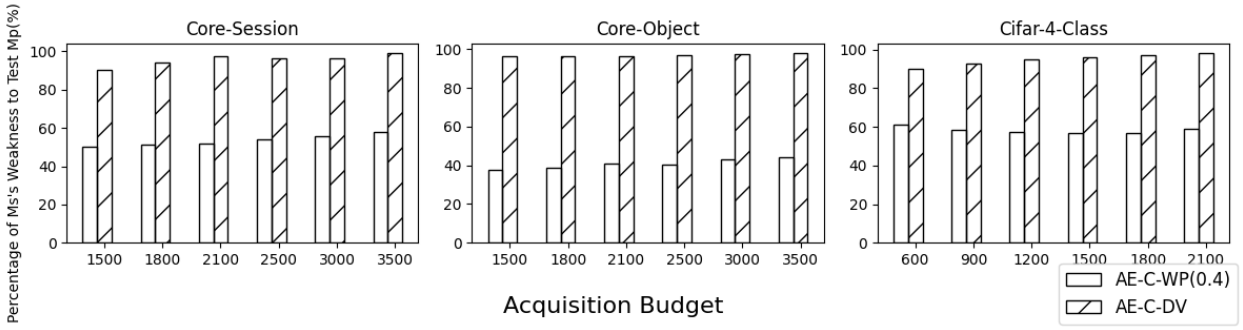

Figure 5.14: $\mathcal{M}_\mathbf{s}$'s Incorrect Predictions over Criteria from WeDe after Sequential Acquisition

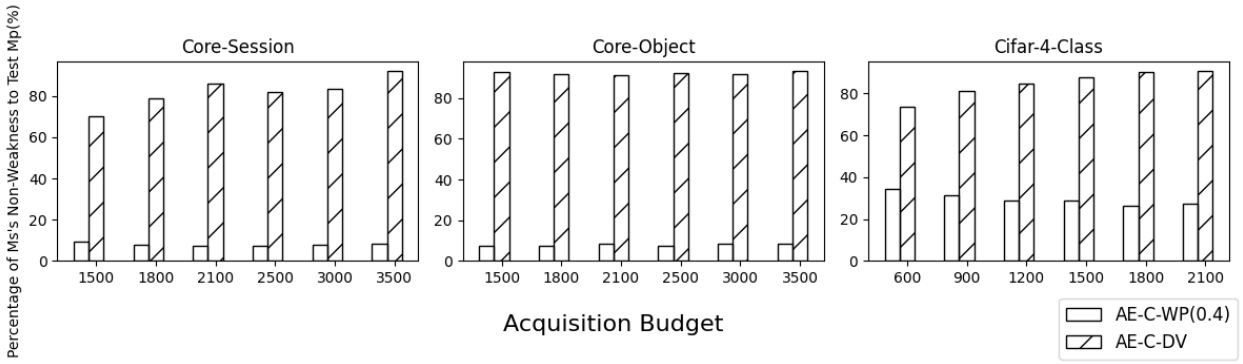

Figure 5.15: $\mathcal{M}_\mathbf{s}$'s Correct Predictions over Criteria from WeDe after Sequential Acquisition

5.3.3.2 Soft-Label Scenario: AE-W vs. AE-C

First, we investigate the distribution of weakness probability $f(\mathbf{x})$ as the weight of $\mathcal{M}_\mathbf{p}$'s predictions in AE-W and a tool for test data assignment in AE-C-WP. Figure 5.17 exhibits an overlap of the $f(\mathbf{x})$ distribution between $\mathcal{M}_\mathbf{s}$'s weaknesses c_w and non-weaknesses $c_{\not w}$. As a result, in AE-W, $\mathcal{M}_\mathbf{p}$ takes more weight than $\mathcal{M}_\mathbf{s}$ in predicting some instances from the none-weakness $c_{\not w}$. In AE-C-WP, $\mathcal{M}_\mathbf{p}$ is given some $c_{\not w}$ instances when the data assignment criterion is not high enough. However, $\mathcal{M}_\mathbf{p}$ specializes in c_w rather than $c_{\not w}$, and thus such overlap in $f(\mathbf{x})$ distribution affects $\mathcal{M}_\mathbf{p}$ in boosting the accuracy improvement of the model ensemble $\mathcal{M}_\mathbf{t}$.

Despite the overlap of $f(\mathbf{x})$ distribution between c_w and $c_{\not w}$, AE-W allows $\mathcal{M}_\mathbf{s}$ and $\mathcal{M}_\mathbf{p}$ to work together on every data instance. Even if $f(\mathbf{x})$ is more than 0.5 in the prediction

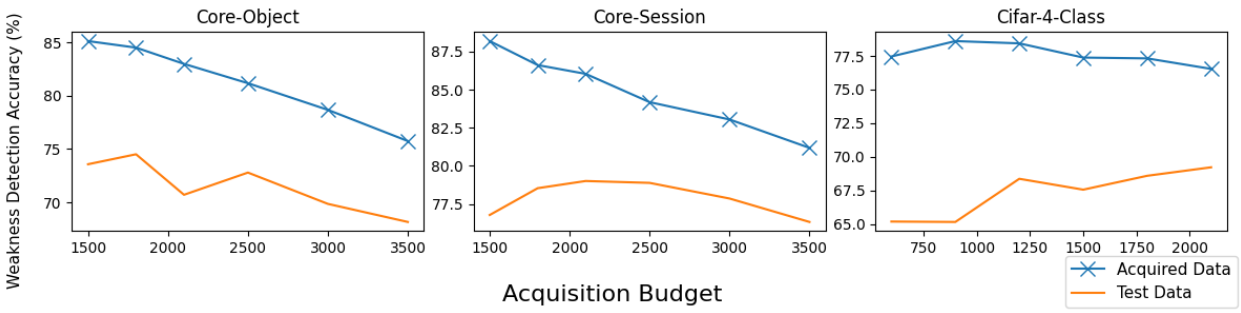

Figure 5.16: Weakness Detection Accuracy of WeDe in Two Related Datasets after Sequential Acquisition

aggregation for $\mathbf{x}$ of $c_{\not\psi}$ and $\mathcal{M}_\mathbf{p}$ outputs incorrect predictions, $\mathcal{M}_\mathbf{s}$ can still reverse the aggregation result by its specialization in $c_{\not\psi}$. On the contrary, AE-C-WP adopts only one model's prediction as the ensemble output $h_t(\mathbf{x})$. When AE-C-WP gives $\mathbf{x}$ of $c_{\not\psi}$ to $\mathcal{M}_\mathbf{p}$, $h_t(\mathbf{x})$ is often incorrect.

Next, we compare AE-W with AE-C-WP in accuracy improvement of $\mathcal{M}_\mathbf{t}$. We choose 0.4 as the test data assignment criterion in AE-C-WP since this criterion returns the best accuracy as shown in subsubsection 5.3.3.1. In Figure 5.18 and Figure 5.19, AE-W ensembles are superior to AE-C-DV ensembles, whichever acquisition strategy is adopted. This result demonstrates that AE-W exploits the strength of $\mathcal{M}_\mathbf{s}$ and $\mathcal{M}_\mathbf{p}$ for the target domain better than AE-C-WP.

5.3.3.3 Soft-Label Scenario: AE-W vs. Average Ensemble

Figure 5.20 and Figure 5.21 compare AE-W and *Average Ensemble* in creating the model ensemble $\mathcal{M}_\mathbf{t}$ with different acquisition strategies. AE-W performs better than Average Ensemble, except for the one-shot acquisition in *Core-Oject* where WeDe has a weakness detection accuracy of only 39.48%. Such inaccurate WeDe affects the generation of weights in aggregating soft labels as the output of $\mathcal{M}_\mathbf{t}$. AE-W does better than Average Ensemble because Average Ensemble assumes $\mathcal{M}_\mathbf{s}$ and $\mathcal{M}_\mathbf{p}$ contribute equally to every prediction ensemble, while AE-W allows the two models to take more charge in their respective specializations.

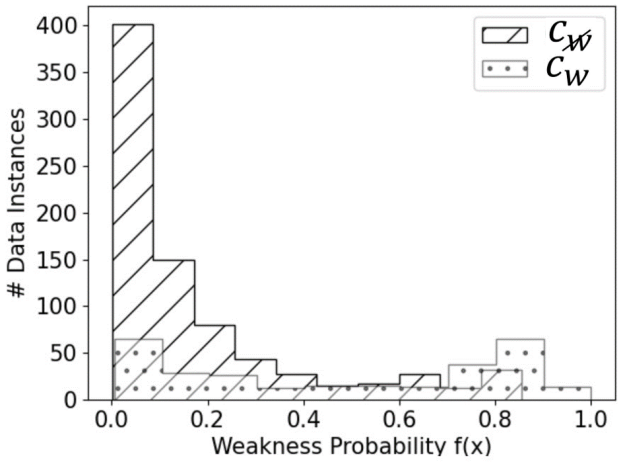

Figure 5.17: Distribution of Weakness Probability

This weakness probability distribution is generated from the test set in the 1st run of experiments of the *Core-Session* task.

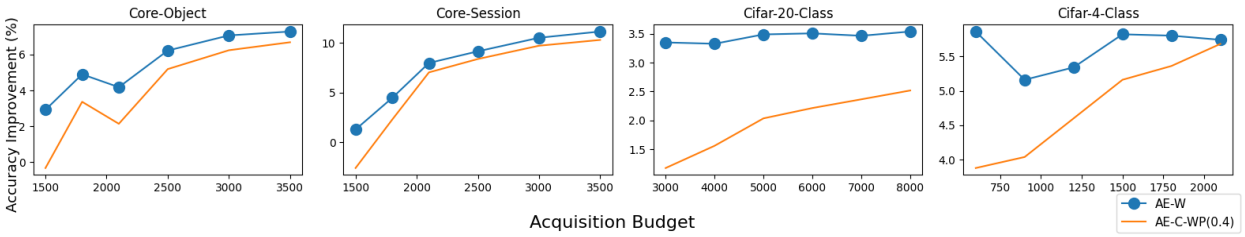

Figure 5.18: One-Shot Acquisition with AE-W or AE-C in Soft-Label Scenario

5.3.3.4 Apply $\mathcal{M}_\mathbf{p}$ without AE

We experiment with only applying $\mathcal{M}_\mathbf{p}$ to the entire test set and check the necessity of creating a model ensemble by AE. Figure 5.22 shows our WEDE-based acquisition has negative accuracy improvement, which means the padding model $\mathcal{M}_\mathbf{p}$ itself is worse than the source model $\mathcal{M}_\mathbf{s}$ for the entire test set. Hence, after we acquire data by WEDE to build $\mathcal{M}_\mathbf{p}$, we must construct a model ensemble of $\mathcal{M}_\mathbf{p}$ and $\mathcal{M}_\mathbf{s}$ for a shifted target domain.

5.3.3.5 Summary

It is essential to create a model ensemble $\mathcal{M}_\mathbf{t}$ with AE and apply $\mathcal{M}_\mathbf{t}$ to the target domain after our WEDE-based acquisition. Given the characteristic of $\mathcal{M}_\mathbf{s}$'s outputs, AE should

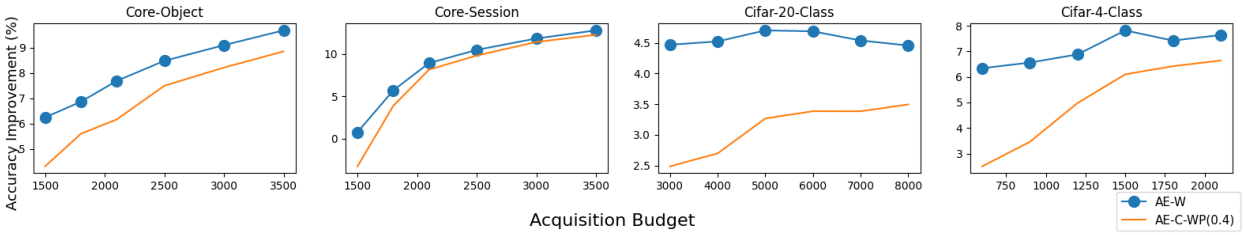

Figure 5.19: Sequential Acquisition with AE-W or AE-C in Soft-Label Scenario

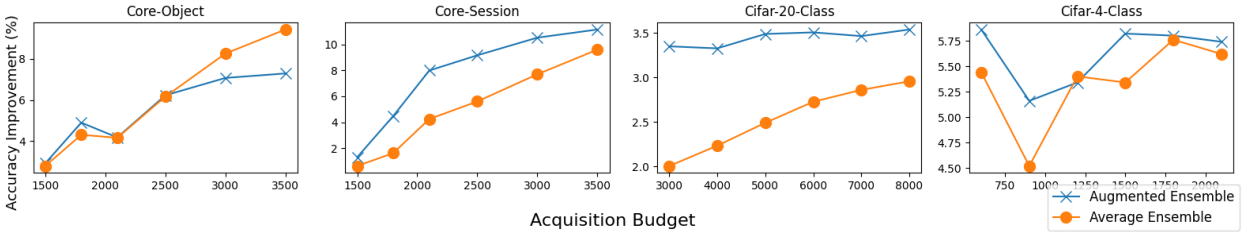

Figure 5.20: One-Shot Acquisition with AE-W or *Average Ensemble* in Soft-Label Scenario

be implemented in different ways. When $\mathcal{M}_\mathbf{s}$ only provides hard labels to customers, we may create $\mathcal{M}_\mathbf{t}$ with either AE-C-WP or AE-C-DV. No matter what acquisition strategy is adopted, AE-C-WP achieves better accuracy than AE-C-DV. However, for the best leverage of AE-C-WP, some preliminary experiments on various target data assignment criteria are needed to find a suitable criterion. On the other hand, when soft labels are also available from $\mathcal{M}_\mathbf{s}$, AE-W outperforms AE-C in producing accurate $\mathcal{M}_\mathbf{t}$. Consequently, in the following experiments, we adopt AE-C-WP for the hard-label-only scenario, while AE-W for the soft-label scenario.

5.3.4 Data Acquisition Strategy

In this section, we first present a detailed analysis of our proposed one-shot and sequential strategy, and then we show the accuracy improvement of all the acquisition methods. We also analyze the composition of data acquired by strategies biased to $\mathcal{M}_\mathbf{s}$'s weaknesses. Finally, we explore the impact of acquisition utility thresholds.

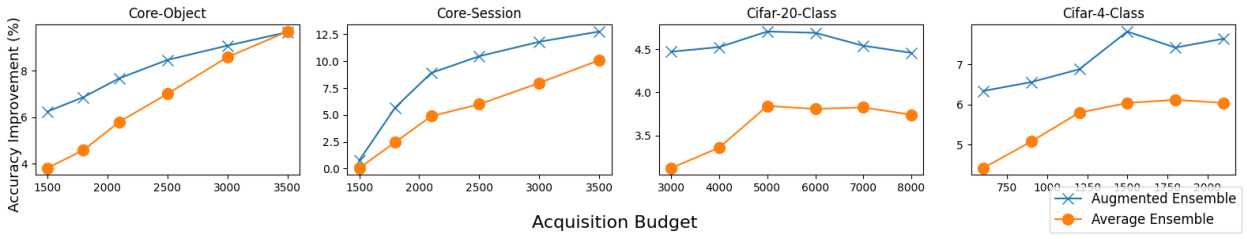

Figure 5.21: Sequential Acquisition with AE-W or *Average Ensemble* in Soft-Label Scenario

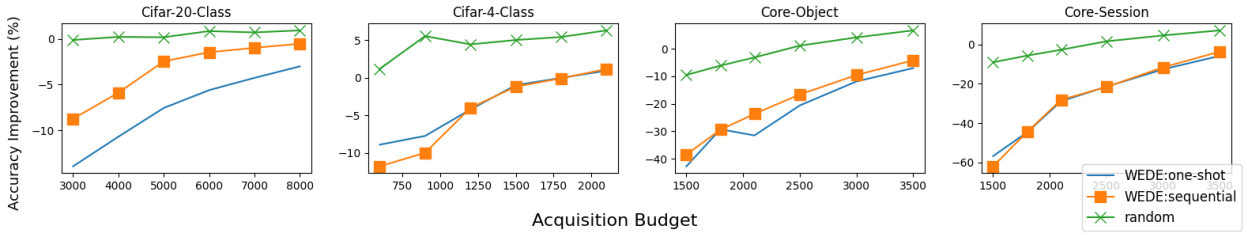

Figure 5.22: Apply $\mathcal{M}_{\mathbf{p}}$ in the Test Set without AE

5.3.4.1 One-shot vs. Sequential

We first examine each round of the sequential acquisition. Each acquisition round in the sequential strategy is set to have the same budget. As the acquisition goes on round by round, more instances of class c_w ($\mathcal{M}_{\mathbf{s}}$'s weaknesses in the target domain) are added to the validation set $\mathcal{D}_V$ to retrain WEDE as shown in Figure 5.23. Since WEDE has more data of c_w to learn, we witness an improvement in weakness detection accuracy as shown in Figure 5.24.

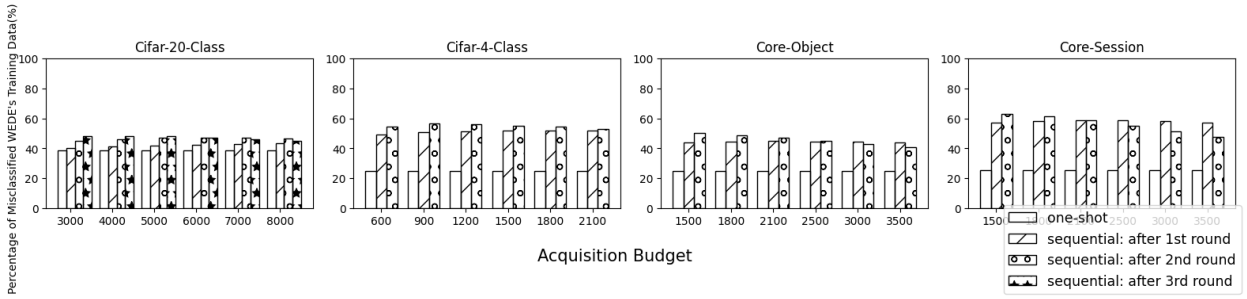

Figure 5.23: Mis-classification by $\mathcal{M}_{\mathbf{s}}$ in WEDE's Training Data

Figure 5.25 is an example of the weakness probability distribution after sequential acqui-

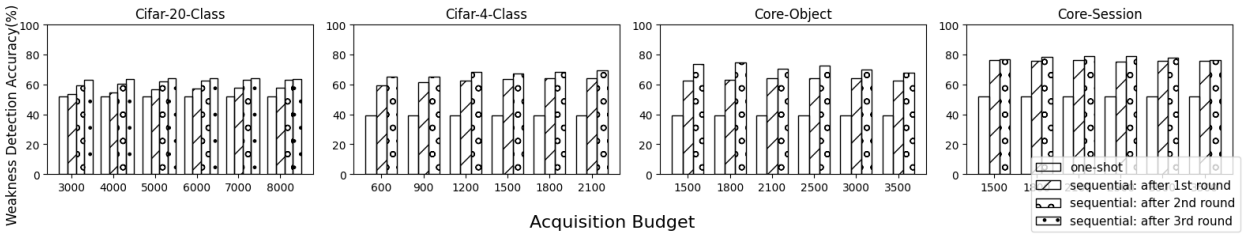

Figure 5.24: Weakness Detection Accuracy in WeDe-Based Acquisition

sition improves WeDe. Compared to Figure 5.17, the distribution overlap between c_w and $c_{\not w}$ greatly shrinks. Less non-weakness and more weakness of $\mathcal{M}_\mathbf{s}$ are estimated with a high weakness probability.

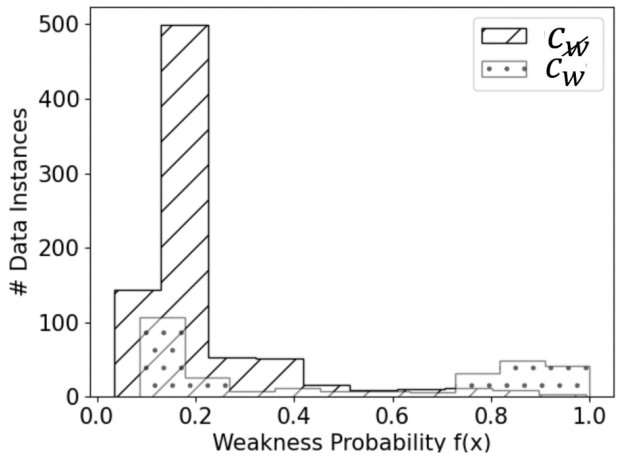

Figure 5.25: Distribution of Weakness Probability from WeDe after Sequential Acquisition
This weakness probability distribution is generated from the test set in the 1st run of *Core-Session* experiments when WeDe is updated by sequential acquisition with a budget of 1500.

Then, in Figure 5.26, we observe that $f(\mathbf{x})$ with WeDe updated by sequential acquisition estimates more test data, which is misclassified by $\mathcal{M}_\mathbf{s}$, with a weakness probability over 0.5, except for *Cifar-4-Class* and *Cifar-20-Class*. Also, fewer test data correctly classified by $\mathcal{M}_\mathbf{s}$ has a weakness probability estimation larger than 0.5 as shown in Figure 5.27. This change in estimated weakness probability for test data leads to better data assignment in AE-C and better model ensemble weights in AE-W. Besides, in the task of *Cifar-4-Class* and *Cifar-20-Class*, the big declines of non-weaknesses with $f(\mathbf{x})$ over 0.5 makes up for the drops in the weakness with $f(\mathbf{x})$ over 0.5. As shown in Figure 5.28 and Figure 5.29, sequential acquisition

does better than one-shot in the accuracy improvement of $\mathcal{M}_\mathbf{t}$ across all classification tasks.

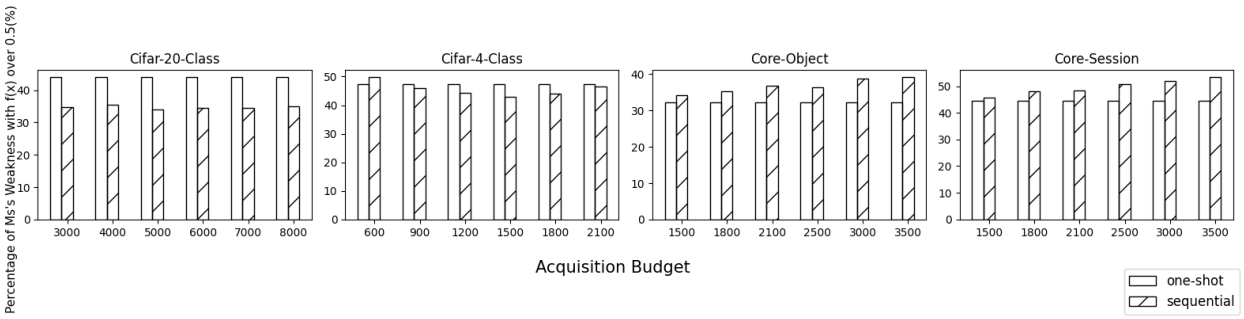

Figure 5.26: $\mathcal{M}_\mathbf{s}$'s Incorrect Predictions with Weakness Probability over 0.5

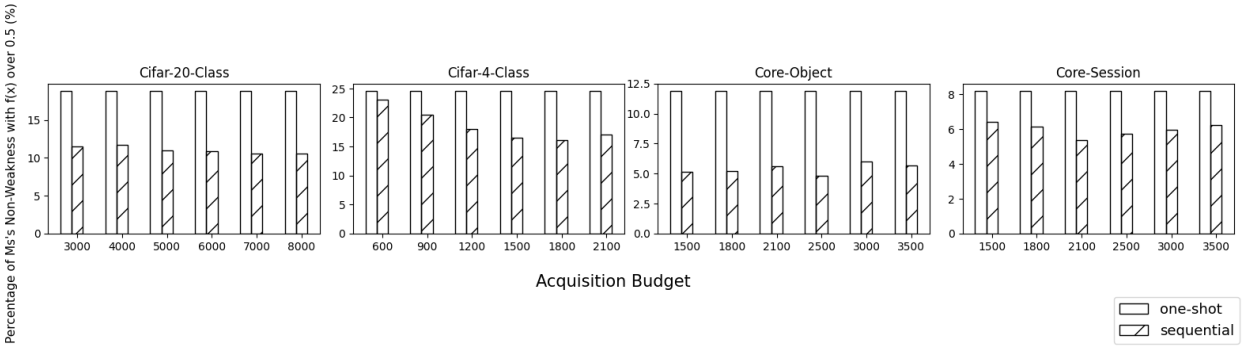

Figure 5.27: $\mathcal{M}_\mathbf{s}$'s Correct Predictions with Weakness Probability over 0.5

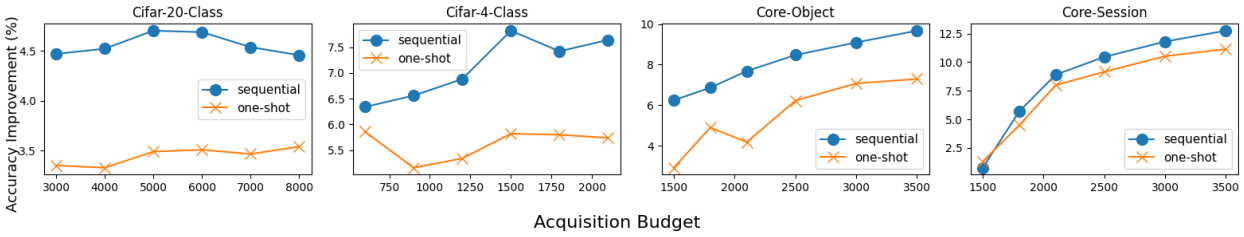

Figure 5.28: One-Shot Acquisition vs. Sequential Acquisition in Soft-Label Scenario

5.3.4.2 WeDe-based Acquisition vs. Baselines

We compare WeDe-based acquisition and baselines in building accurate $\mathcal{M}_\mathbf{t}$ for the test set. In the hard-label-only scenario, we ignore PGT and the entropy-based acquisition because they both require soft labels from $\mathcal{M}_\mathbf{s}$. Note that in actual application, it is often undesired

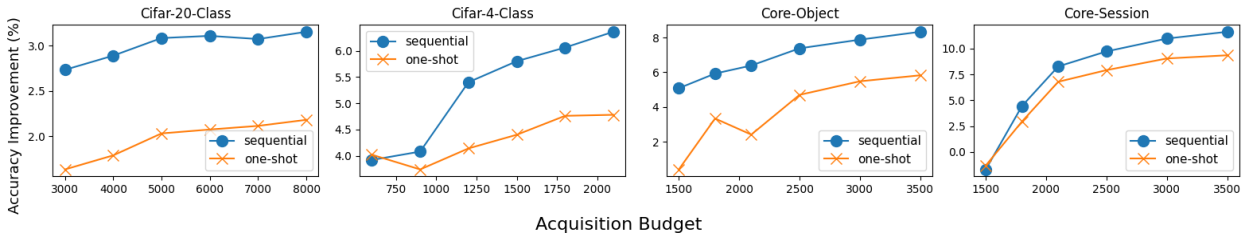

Figure 5.29: One-Shot Acquisition vs. Sequential Acquisition in Hard-Label-Only Scenario

for customers to adopt PGT, random weakness, and entropy-based baselines because customers have to send their $\mathcal{M}_\mathbf{s}$ to data sellers for data valuation in these strategies, which violates their need to protect the privacy and usage of $\mathcal{M}_\mathbf{s}$.

As shown in Figure 5.30 and Figure 5.31, sequential acquisition achieves the most accuracy gains in either application scenario.

Random acquisition returns $\mathcal{D}_B$ exactly from the target domain $\mathbf{T}$, but it does not apply any of $\mathcal{M}_\mathbf{s}$ to the target domain, which means $\mathcal{M}_\mathbf{p}$ has to prepare for $\mathbf{T}$ from the beginning by learning about $\mathcal{D}_B$. $\mathcal{D}_B$ acquired by a small budget B even fails to train a $\mathcal{M}_\mathbf{p}$ with competitive performance as $\mathcal{M}_\mathbf{s}$ in $\mathbf{T}$.

Instead, biased acquisition methods associated with the model ensemble approach AE exploit the capability of $\mathcal{M}_\mathbf{s}$ in $\mathbf{T}$ in addition to $\mathcal{M}_\mathbf{p}$. As such, they achieve better results than random acquisition under small budgets. However, they also exhaust $\mathcal{M}_\mathbf{s}$'s weaknesses in $\mathcal{D}$ faster than random acquisition because they have acquisition bias in $\mathcal{M}_\mathbf{s}$'s weaknesses. As a result, we observe that as B increases, accuracy improvement from biased strategies slows down while the improvement from random acquisition goes up steadily.

Note that the performance of random acquisition in *Cifar-20-Class* increases slower than the other two tasks even if the budget increases faster with a step of 1000. Because $\mathcal{M}_\mathbf{s}$ from *Cifar-20-Class* is trained from a much larger dataset, it is harder for random acquisition to generate $\mathcal{M}_\mathbf{p}$ to do better than $\mathcal{M}_\mathbf{s}$ in the target domain.

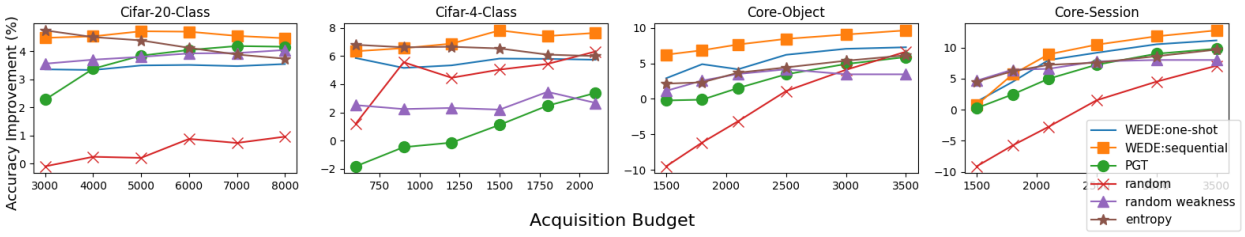

Figure 5.30: WEDE-based Acquisition vs. Baselines in Soft-Label Scenario

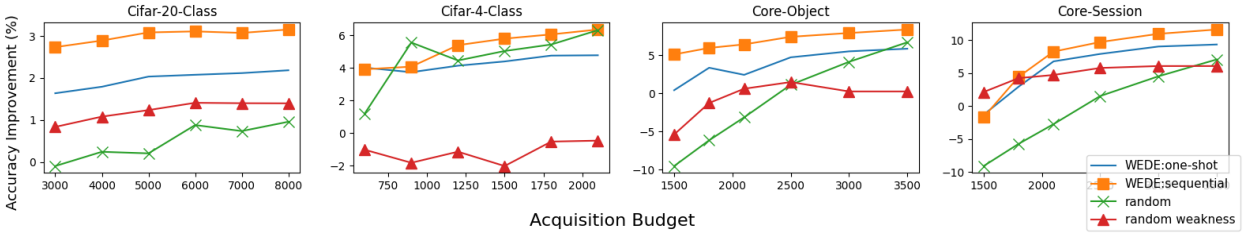

Figure 5.31: WEDE-based Acquisition vs. Baselines in Hard-Label-Only Scenario

5.3.4.3 Invalid Data from Biased Acquisition of $\mathcal{M}_\mathbf{s}$'s Weakness

By far, we consider $\mathcal{M}_\mathbf{s}$'s prediction mistakes as its weakness. However, not every data instance from $\mathcal{D}$ misclassified by $\mathcal{M}_\mathbf{s}$ contributes to the effective construction of $\mathcal{M}_\mathbf{p}$. Due to noise or mislabelling, data can be extremely difficult and even impossible to be correctly predicted with any classifiers. We define such data as *invalid data* for acquisition because they do not train $\mathcal{M}_\mathbf{p}$ to help $\mathcal{M}_\mathbf{s}$ with challenging distribution shifts in $\mathbf{T}$ but to work on unnecessary and almost intractable tasks.

One-shot, sequential, *entropy-based*, *Probability-at-Ground-Truth (PGT)*, and *random weakness* strategies are all biased in selecting $\mathcal{M}_\mathbf{s}$'s weaknesses from $\mathcal{D}$. They are also prone to return invalid data for the construction of $\mathcal{M}_\mathbf{p}$. To summarize invalid data acquired by these biased strategies, we build a benchmark model $\mathcal{M}_\mathbf{b}$ from a dataset that is exactly from the target domain, and $\mathcal{M}_\mathbf{b}$ also performs relatively well in the test set. Then, we consider every sample misclassified by $\mathcal{M}_\mathbf{b}$ as invalid data for acquisition. For tasks *Cifar-4-Class* and *Cifar-20-Class*, we implement $\mathcal{M}_\mathbf{b}$ by frozen feature extraction layers from a Mobilenet trained on the entire *Cifar-100* [14] and a linear classification layer where the number of output neurons corresponds to how superclasses are involved in each task. $\mathcal{M}_\mathbf{b}$ has an accu-

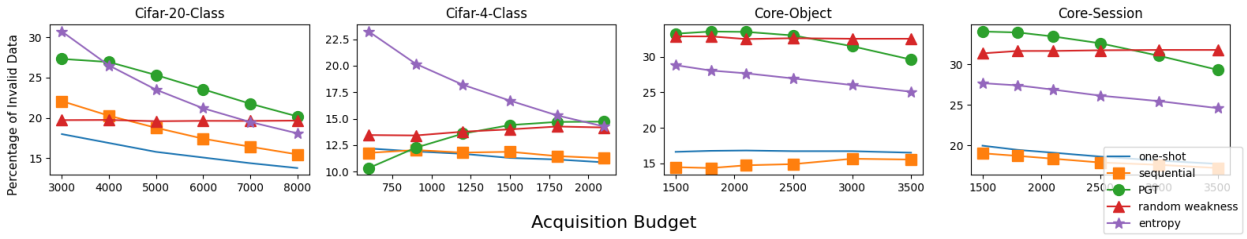

Figure 5.32: Invalid Data after Acquisition

racy of 78.12% and 79.178% in the test set for *Cifar-4-Class* and *Cifar-20-Class* respectively. *Core-Session* and *Core-Object* share the same $\mathcal{M}_{\mathbf{b}}$ implemented by Squeezenet, with nearly 87% test accuracy. Figure 5.32 displays the percentage of invalid data acquired by different methods.

Sequential acquisition with WeDe returns almost the least invalid data in all tasks except *Cifar-20-Class* where it has a few more invalid data than one-shot acquisition. The advantage of sequential acquisition over other biased strategies in selecting valid data from $\mathcal{D}$ lies in the fact that the weakness of $\mathcal{M}_{\mathbf{s}}$ caused by distribution shifts exhibits common features, while invalid data is often heterogeneous and rarely shows similar features. Then, WeDe-based method is more likely to choose valid data than its counterparts. Although the one-shot acquisition also uses WeDe, it does not have WeDe as accurate as that from the sequential counterpart as discussed in the comparison between one-shot and sequential acquisition in subsubsection 5.3.4.1, and thus it returns more invalid data in most cases. Accordingly, our observation of WeDe-based and baselines in subsubsection 5.3.4.2 shows that sequential acquisition primarily with the least invalid data returned brings about the most powerful $\mathcal{M}_{\mathbf{p}}$ in tackling the weakness of $\mathcal{M}_{\mathbf{s}}$ from $\mathbf{T}$.

5.3.4.4 Utility Threshold

We explore the impact of data utility thresholds on accuracy improvements when we adopt one-shot acquisition. We do not try the sequential acquisition, because after the first acquisition round, $\mathcal{D}_B$ returned by each utility threshold differs, and WeDe is also updated differently. Then in the next acquisition, it is hard to keep a consistent number of data in-

stances returned by different thresholds. Besides, to make utility thresholds meaningful, we choose the utility estimator U-WSD with SVM-based WeDe, which returns probabilities as data utility estimates.

We experiment with utility thresholds of 0.5, 0.6, and 0.7. For each threshold, we sample a certain number of data instances whose utility is within the given threshold. Specifically, we first count how many data instances are selected by threshold 0.7 and then sample the same number of instances when the threshold is 0.5 or 0.6. As a result, data returned by the three different thresholds has consistent sizes, and only utility thresholds influence the final accuracy improvement.

Figure 5.33 shows the percent of invalid acquired data under different utility thresholds. A higher threshold returns a slightly higher percentage of invalid data. This is because the weakness detection accuracy of WeDe in the one-shot acquisition is still not good enough, and thus data with a relatively high utility from WeDe can be invalid. Accordingly, as shown in Figure 5.34, the highest threshold does not necessarily lead to the biggest accuracy improvement.

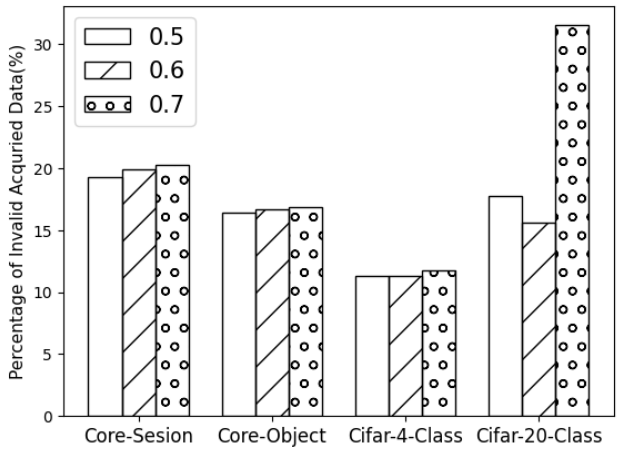

Figure 5.33: Invalid Data Acquired by Different Thresholds in U-WSD

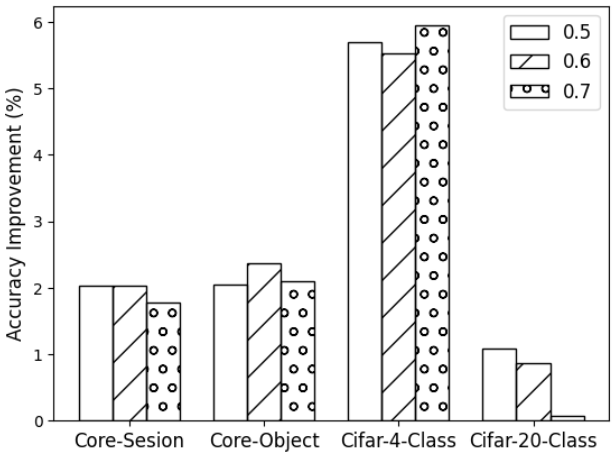

Figure 5.34: One-Shot Acquisition with Thresholds in U-WSD under Soft-Label Scenario

5.4 Factors that Impact WeDe-based Data Acquisition

We vary three factors that affect WeDe-based acquisition: the complexity of the padding model $\mathcal{M}_{\mathbf{p}}$, the degree of distribution shift, and the distribution of $\mathcal{D}$. We compare WeDe-based acquisition with random acquisition, the only practical baseline.

In Figure 5.35 and Figure 5.36, the Y-axis labeled "WeDe over Random" refers to the classification accuracy improvement achieved by WeDe-based acquisition over that achieved by random acquisition.

We perform sequential acquisition with data utility estimation by U-WS and SVM-based WeDe in all the following experiments, except that *Cifar-2-class*, a binary classification task from *Cifar-100*, uses the one-shot acquisition. *Cifar-2-class* has a data pool of 3000 images, a validation set, and a test set both with 100 images. The distribution shift in this task makes the accuracy of $\mathcal{M}_{\mathbf{s}}$ drop from 96.25% to 84.6%.

5.4.1 Complexity of the Padding Model

To study how $\mathcal{M}_{\mathbf{p}}$'s complexity influences the accuracy improvement of $\mathcal{M}_{\mathbf{t}}$, we fix the implementation of $\mathcal{M}_{\mathbf{s}}$ (Squeezenet for *Core-Object* and SVM for *Cifar-2-Class*) but vary that of $\mathcal{M}_{\mathbf{p}}$ with four classification model architectures of increasing complexity: SVM with a

linear kernel (512 parameters), Squeezenet (0.6M parameters), Resnet-18 (11M parameters), and Resnet-34 (22M parameters). We train models from scratch. SVM outputs a distance between an instance and the class boundary, which is incompatible with the prediction probability from other model implementations. In *Cifar-2-class*, $\mathcal{M}_{\mathbf{s}}$ is an SVM. So for this task, we build the ensemble by AE-C-WP when $\mathcal{M}_{\mathbf{p}}$ is not implemented with SVM.

From Figure 5.35, we observe that larger model complexity leads to a more significant advantage of WEDE-based acquisition over random acquisition. This is especially obvious when the budget is small; as the acquisition budget increases, the advantage of WEDE-based method becomes less apparent. This observation aligns with the intuition that $\mathcal{M}_{\mathbf{p}}$ with higher complexity requires more training data and random acquisition works poorly when the acquisition budget is small.

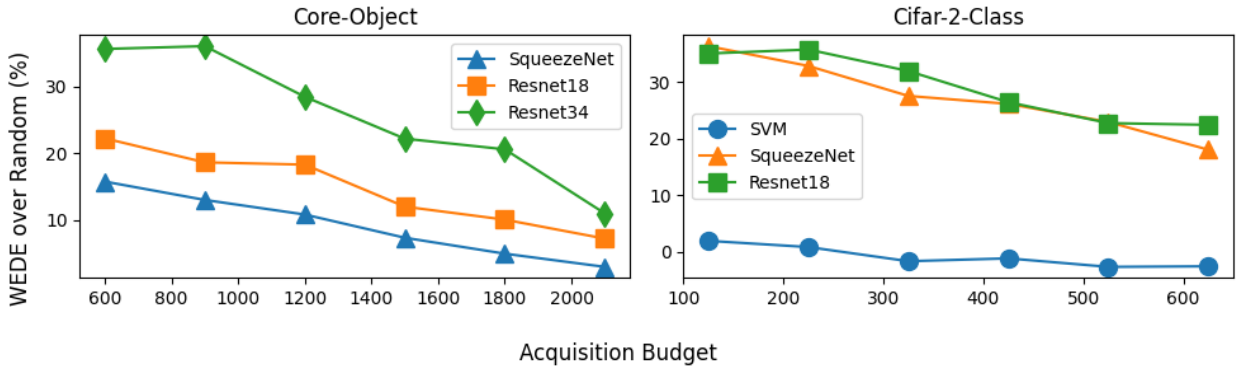

Figure 5.35: Acquisition under Different Model Complexity

5.4.2 Degree of Distribution Shift

We generate various degrees of distribution shift by removing 1, 2, or 3 subclasses from each superclass in *Cifar-2-class*, and 2, 3, or 5 sessions from each category in *Core-50*. Datasets before and after the data removal are considered to be drawn from source and target domains. The more subclasses/sessions are removed, the higher the discrepancy is between source and target domains.

Figure 5.36 shows the effect of distribution shift degrees, where each curve represents the accuracy improvement obtained by WEDE-based acquisition over random acquisition with a

different number of sessions/subclasses removed. As the degree of distribution shift becomes greater, e.g. from 2 sessions to 5 sessions removed from each category (with each category originally containing 11 sessions) or from 1 subclass to 3 subclasses removed from each superclass (with each superclass originally containing 5 subclasses), WEDE-based acquisition presents less advantage over random acquisition. This result agrees with the intuition that model $\mathcal{M}_{\mathbf{s}}$ is of little use to the prediction ensemble if the target domain is significantly different from the source domain.

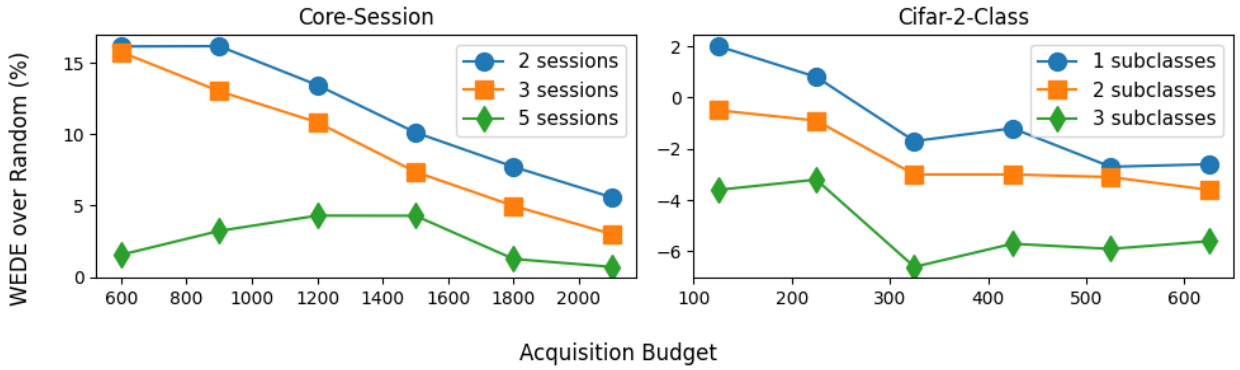

Figure 5.36: Acquisition under Different Distribution Shifts

5.4.3 Distribution of the Data Pool

We design a data pool $\mathcal{D}$ with a larger coverage than the target domain $\mathbf{T}$ and test the robustness of our proposed method in the presence of distribution discrepancy between $\mathcal{D}$ and $\mathbf{T}$. *Cifar-2-Class* and *Cifar-4-Class* are modified for this purpose. In *Cifar-2-Class*, the test set and the validation set $\mathcal{D}_V$ only cover subclasses of bus, streetcar, tank, and train, while $\mathcal{D}$ has all the subclasses. The modified domain of *Cifar-4-Class* is illustrated in 5.37.

For comparison, we also create a filtered version of the data pool using a one-class SVM, an off-the-shelf tool in the *scikit-learn* library, trained on $\mathcal{D}_V$ to remove images from $\mathcal{D}$ for matching $\mathcal{D}$ with $\mathbf{T}$. *Cifar-2-Class* and *Cifar-4-Class* respectively have a filtering accuracy of 77% and 65%. Since *Cifar-4-Class* does not have an accurate data filter for $\mathcal{D}$, we ignore a filtered $\mathcal{D}$ for this task. Besides, although it is helpful to set a benchmark for comprehending the capabilities of our proposed approach, this filter operation is unrealistic because: first,

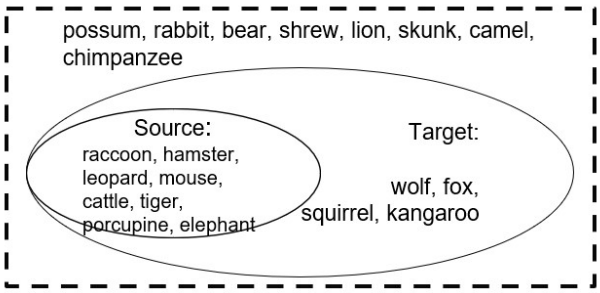

Figure 5.37: *Cifar-4-Class* with a Data Pool Larger than the Target Domain
The dashed rectangle marks the domain of the data pool.

if customers send their target data to data sellers for building a data pool filter to control the acquisition scope, they may leak sensitive information in their target data to the seller. Second, if customers build a filter by themselves, they are not able to evaluate the filtering accuracy in $\mathcal{D}$.

$\mathcal{M}_\mathbf{s}$ and $\mathcal{M}_\mathbf{p}$ in *Cifar-2-Class* are implemented with the normal SVM and we adopt the one-shot acquisition. As we can observe from Figure 5.38, when $\mathcal{D}$ is not filtered, both tasks benefit from WEDE-based acquisition more than the random method. Also, in *Cifar-2-Class*, the performance of WEDE-based acquisition is even close to what can be achieved on the hypothetically filtered $\mathcal{D}$. This result demonstrates the robustness of our WEDE-based method against a broader data pool.

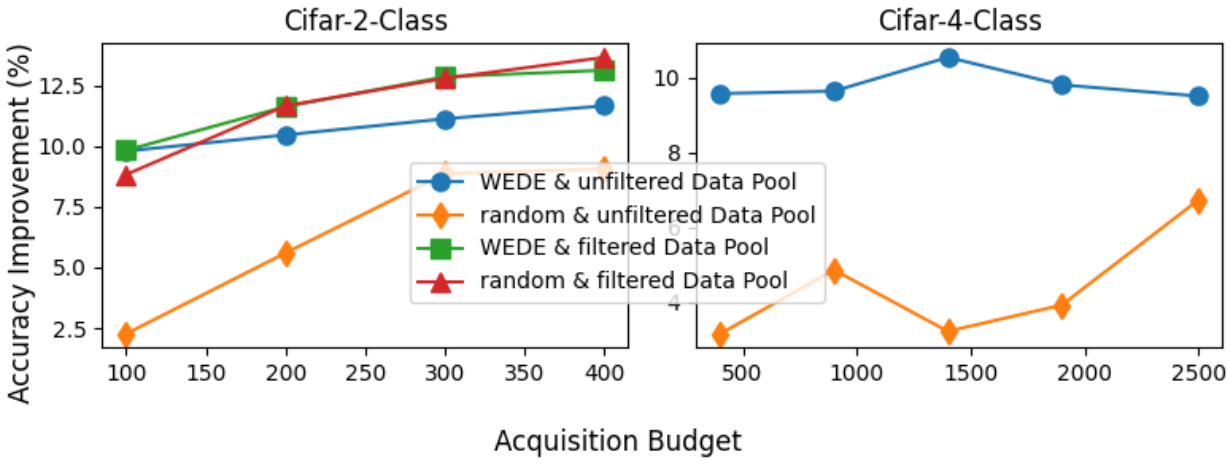

Figure 5.38: Acquisition from Data Pool with Different Distributions

5.5 Discussion and Future Work

Our proposed solution of domain adaptation of closed-box models applies whenever $\mathcal{M}_\mathbf{s}$ outputs prediction probabilities, and is demonstrated to be more effective than various baselines, especially when the data acquisition budget is not enough to construct a powerful $\mathcal{M}_\mathbf{p}$ for the entire domain, or when the data pool covers more data categories than the target domain.

However, our solution is subject to the complexity of $\mathcal{M}_\mathbf{p}$ and the domain discrepancy degree. Our analysis of these factors provides insights for customers to choose suitable data acquisition strategies for individual application cases. A more rigid and thorough study of how those factors affect our solution is encouraged in the future.

Although it is straightforward to describe the weakness of $\mathcal{M}_\mathbf{s}$ with incorrect predictions from $\mathcal{M}_\mathbf{s}$, invalid classification samples are included in c_w to build WeDe, which makes it challenging for WeDe to extract features of regular weakness caused by domain shifts. As we observe in subsubsection 5.3.4.4, such WeDe attach high utility scores to some invalid samples. To exclude invalid data from weakness feature extraction in WeDe, the soft label entropy from $\mathcal{M}_\mathbf{s}$ can be employed to define $\mathcal{M}_\mathbf{s}$'s weaknesses instead. According to our comparison of data acquisition strategies, the entropy-based method returns less invalid data than the *confidence-based* and *random weakness*, both of which define model weaknesses by incorrect predictions. However, a threshold is also needed to determine how high an entropy is for the model weakness, and the set-up of such thresholds may require some trials. Besides, this definition only works when soft labels are accessible from $\mathcal{M}_\mathbf{s}$.

Lastly, our framework can be extended to closed-box regression models. Since the targets of regression tasks are continuous, prediction weaknesses of regression models may be defined with error tolerance values.

6 Conclusion

In this thesis, we have developed an effective, versatile, and secure domain adaptation framework for closed-box models by sourcing extra data from the machine-learning market given an acquisition budget. This framework aims at commonplace scenarios when closed-box models suffer from performance degradation due to the disparity between source and target domains caused by distribution shifts. Besides, neither customers' models nor their target data is exposed to data sellers in this framework, which protects customers' privacy.

We first build a classifier WeDe to extract features of prediction mistakes made by the existing closed-box model in a dataset sampled from the target domain. Based on weakness scores from WeDe, we provide U-WS and U-WSD as two options for estimating data utility in enhancing the closed box. U-WS directly uses weakness scores as utility estimates, while U-WSD evaluates data utility by the probability that acquisition candidates are from the closed box's prediction weakness distribution. Data utilities generated by U-WS or U-WSD are then used to acquire data instances in a one-shot or sequential fashion. A new model is built from acquired data to help the closed box with its weaknesses in the target domain, and the specialty of this new model can be controlled by a data utility threshold during data acquisition. Finally, we introduce a comprehensive design of Augmented Ensemble (AE), considering the characteristics of $\mathcal{M}_{\mathbf{s}}$'s outputs, to combine predictions of the new model and the original closed box and produce an improved model for the target domain. The enhanced model can be sold back to the machine-learning market and also increase its diversity.

Extensive experiments and analyses demonstrate that our proposed framework outperforms baselines, especially when our data acquisition budget is small, or the data acquisition

source comes from a larger domain than the customer's target.

References

[1] D. Abati, A. Porrello, S. Calderara, and R. Cucchiara. Latent space autoregression for novelty detection. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 481–490, Los Alamitos, CA, USA, jun 2019. IEEE Computer Society.

[2] Abubakar Abid, Mert Yuksekgonul, and James Zou. Meaningfully debugging model mistakes using conceptual counterfactual explanations. In *Proceedings of the 39th International Conference on Machine Learning*, pages 66–88. PMLR, June 2022. ISSN: 2640-3498.

[3] Anish Agarwal, Munther Dahleh, and Tuhin Sarkar. A marketplace for data: An algorithmic solution. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, EC '19, page 701–726, New York, NY, USA, 2019. Association for Computing Machinery.

[4] Amazon. Amazon rekognition, 2023.

[5] Amazon. Aws data exchange, 2023.

[6] Antreas Antoniou, Amos Storkey, and Harrison Edwards. Data augmentation generative adversarial networks, 2018.

[7] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: A strong, simple baseline. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA, 2020. Curran Associates Inc.

[8] Alberto Cano and Bartosz Krawczyk. ROSE: robust online self-adjusting ensemble for continual learning on imbalanced drifting data streams. *Machine Language*, 111(7):2561–2599, July 2022.

[9] Raul Castro Fernandez. Protecting data markets from strategic buyers. In *Proceedings of the 2022 International Conference on Management of Data*, SIGMOD '22, page 1755–1769, New York, NY, USA, 2022. Association for Computing Machinery.

[10] Chengliang Chai, Jiabin Liu, Nan Tang, Guoliang Li, and Yuyu Luo. Selective data acquisition in the wild for model charging. *Proceedings of the VLDB Endowment*, 15(7):1466–1478, March 2022.

[11] Arslan Chaudhry, Albert Gordo, David Lopez-Paz, Puneet K. Dokania, and Philip Torr. Using hindsight to anchor past knowledge in continual learning, 2020.

[12] Lingjiao Chen, Paraschos Koutris, and Arun Kumar. Towards Model-based Pricing for Machine Learning in a Data Marketplace. In *Proceedings of the 2019 International Conference on Management of Data*, pages 1535–1552, Amsterdam Netherlands, June 2019. ACM.

[13] Weitao Chen, Rick Salay, Sean Sedwards, Vahdat Abdelzad, and Krzysztof Czarnecki. Accelerating the training of convolutional neural networks for image segmentation with deep active learning. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, page 1–7. IEEE Press, 2020.

[14] chenyaofo. Pytorch cifar models. `https://github.com/chenyaofo/pytorch-cifar-models/`, 2023.

[15] Nadiia Chepurko, Ryan Marcus, Emanuel Zgraggen, Raul Castro Fernandez, Tim Kraska, and David Karger. ARDA: automatic relational data augmentation for machine learning. *Proceedings of the VLDB Endowment*, 13(9):1373–1387, May 2020.

[16] Z. Cong, L. Chu, L. Wang, X. Hu, and J. Pei. Exact and consistent interpretation of piecewise linear models hidden behind apis: A closed form solution. In *2020 IEEE 36th*

International Conference on Data Engineering (ICDE), pages 613–624, Los Alamitos, CA, USA, apr 2020. IEEE Computer Society.

[17] Databricks. Databricks marketplace, 2023.

[18] Sayna Ebrahimi, Mohamed Elhoseiny, Trevor Darrell, and Marcus Rohrbach. Uncertainty-guided continual learning with bayesian neural networks. In *International Conference on Learning Representations*, 2020.

[19] Raul Castro Fernandez, Pranav Subramaniam, and Michael J. Franklin. Data market platforms: Trading data assets to solve data problems. *Proc. VLDB Endow.*, 13(12):1933–1947, jul 2020.

[20] J Friedman. On Multivariate Goodness-of-Fit and Two-Sample Testing. Technical Report SLAC-PUB-10325, 826696, January 2004.

[21] Amirata Ghorbani and James Zou. Data shapley: Equitable valuation of data for machine learning. In *International Conference on Machine Learning*, pages 2242–2251, 2019.

[22] Google. Google cloud vision api, 2023.

[23] Yuhong Guo and Dale Schuurmans. Discriminative batch mode active learning. In *Proceedings of the 20th International Conference on Neural Information Processing Systems*, NIPS'07, page 593–600, Red Hook, NY, USA, 2007. Curran Associates Inc.

[24] Kazuaki Hanawa, Sho Yokoi, Satoshi Hara, and Kentaro Inui. EVALUATION OF SIMILARITY-BASED EXPLANATIONS. page 26, 2021.

[25] Joel Hestness, Sharan Narang, Newsha Ardalani, Gregory Diamos, Heewoo Jun, Hassan Kianinejad, Md Mostofa Ali Patwary, Yang Yang, and Yanqi Zhou. Deep Learning Scaling is Predictable, Empirically, December 2017. arXiv:1712.00409 [cs, stat].

[26] Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian Active Learning for Classification and Preference Learning, December 2011. arXiv:1112.5745 [cs, stat].

[27] Forrest N. Iandola, Song Han, Matthew W. Moskewicz, Khalid Ashraf, William J. Dally, and Kurt Keutzer. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size, November 2016. arXiv:1602.07360 [cs].

[28] Andrew Ilyas, Sung Min Park, Logan Engstrom, Guillaume Leclerc, and Aleksander Madry. Datamodels: Predicting Predictions from Training Data, February 2022. Number: arXiv:2202.00622 arXiv:2202.00622 [cs, stat].

[29] Saachi Jain, Hannah Lawrence, Ankur Moitra, and Aleksander Madry. Distilling model failures as directions in latent space. In *The Eleventh International Conference on Learning Representations*, 2023.

[30] Ruoxi Jia, David Dao, Boxin Wang, Frances Ann Hubis, Nick Hynes, Nezihe Merve Gürel, Bo Li, Ce Zhang, Dawn Song, and Costas J. Spanos. Towards efficient data valuation based on the shapley value. In Kamalika Chaudhuri and Masashi Sugiyama, editors, *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pages 1167–1176. PMLR, 16–18 Apr 2019.

[31] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017.

[32] Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. *BatchBALD: Efficient and Diverse Batch Acquisition for Deep Bayesian Active Learning*. Curran Associates Inc., Red Hook, NY, USA, 2019.

[33] Pang Wei Koh and Percy Liang. Understanding Black-box Predictions via Influence Functions. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1885–1894. PMLR, July 2017. ISSN: 2640-3498.

[34] Pang Wei W Koh, Kai-Siang Ang, Hubert Teo, and Percy S Liang. On the Accuracy of Influence Functions for Measuring Group Effects. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[35] Alex Krizhevsky. Learning Multiple Layers of Features from Tiny Images.

[36] Yifan Li, Xiaohui Yu, and Nick Koudas. Data Acquisition for Improving Machine Learning Models. *Proceedings of the VLDB Endowment*, 14(10):1832–1844, June 2021. arXiv:2105.14107 [cs].

[37] Jian Liang, Dapeng Hu, Jiashi Feng, and Ran He. DINE: Domain Adaptation from Single and Multiple Black-box Predictors. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7993–8003, New Orleans, LA, USA, June 2022. IEEE.

[38] Mingfu Liang, Jiahuan Zhou, Wei Wei, and Ying Wu. Balancing Between Forgetting and Acquisition in Incremental Subpopulation Learning. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision – ECCV 2022*, volume 13686, pages 364–380. Springer Nature Switzerland, Cham, 2022. Series Title: Lecture Notes in Computer Science.

[39] Jiabin Liu, Chengliang Chai, Yuyu Luo, Yin Lou, Jianhua Feng, and Nan Tang. Feature augmentation with reinforcement learning. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, pages 3360–3372, 2022.

[40] Jinfei Liu, Jian Lou, Junxu Liu, Li Xiong, Jian Pei, and Jimeng Sun. Dealer: an end-to-end model marketplace with differential privacy. *Proceedings of the VLDB Endowment*, 14(6):957–969, February 2021.

[41] Vincenzo Lomonaco and Davide Maltoni. Core50: a new dataset and benchmark for continuous object recognition. In Sergey Levine, Vincent Vanhoucke, and Ken Goldberg, editors, *Proceedings of the 1st Annual Conference on Robot Learning*, volume 78 of *Proceedings of Machine Learning Research*, pages 17–26. PMLR, 13–15 Nov 2017.

[42] David Lopez-Paz and Maxime Oquab. Revisiting classifier two-sample tests. In *International Conference on Learning Representations*, 2017.

[43] Zheda Mai, Ruiwen Li, Jihwan Jeong, David Quispe, Hyunwoo Kim, and Scott Sanner. Online continual learning in image classification: An empirical survey. *Neurocomput.*, 469(C):28–51, jan 2022.

[44] OpenVINO. Open model zoo, 2023.

[45] Jian Pei and Xiaohui Yu. Data acquisition and valuation in data markets.

[46] Stanislav Pidhorskyi, Ranya Almohsen, Donald A. Adjeroh, and Gianfranco Doretto. Generative probabilistic novelty detection with adversarial autoencoders. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 6823–6834, Red Hook, NY, USA, 2018. Curran Associates Inc.

[47] V. Prabhu, A. Chandrasekaran, K. Saenko, and J. Hoffman. Active domain adaptation via clustering uncertainty-weighted embeddings. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8485–8494, Los Alamitos, CA, USA, oct 2021. IEEE Computer Society.

[48] Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. Estimating Training Data Influence by Tracing Gradient Descent. In *Advances in Neural Information Processing Systems*, volume 33, pages 19920–19930. Curran Associates, Inc., 2020.

[49] Stephan Rabanser, Stephan Günnemann, and Zachary C. Lipton. *Failing Loudly: An Empirical Study of Methods for Detecting Dataset Shift.* Curran Associates Inc., Red Hook, NY, USA, 2019.

[50] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision, February 2021. arXiv:2103.00020 [cs].

[51] Jonathan S. Rosenfeld, Amir Rosenfeld, Yonatan Belinkov, and Nir Shavit. A constructive prediction of the generalization error across scales. In *International Conference on Learning Representations*, 2020.

[52] Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4393–4402. PMLR, 10–15 Jul 2018.

[53] Omer Sagi and Lior Rokach. Ensemble learning: A survey. *WIREs Data Mining and Knowledge Discovery*, 8(4):e1249, 2018.

[54] Robert E. Schapire. A brief introduction to boosting. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'99, page 1401–1406, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc.

[55] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018.

[56] Yucheng Shi, Kunhong Wu, Yahong Han, Yunfeng Shao, Bingshuai Li, and Fei Wu. Source-free and black-box domain adaptation via distributionally adversarial training. *Pattern Recognition*, 143:109750, November 2023.

[57] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 2994–3003, Red Hook, NY, USA, 2017. Curran Associates Inc.

[58] Connor Shorten and Taghi M. Khoshgoftaar. A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, 6(1):60, December 2019.

[59] S. Sinha, S. Ebrahimi, and T. Darrell. Variational adversarial active learning. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5971–5980, Los Alamitos, CA, USA, nov 2019. IEEE Computer Society.

[60] Peng Sun, Xu Chen, Guocheng Liao, and Jianwei Huang. A profit-maximizing model marketplace with differentially private federated learning. In *IEEE INFOCOM 2022 - IEEE Conference on Computer Communications*, pages 1439–1448, 2022.

[61] Ki Hyun Tae and Steven Euijong Whang. Slice tuner: A selective data acquisition framework for accurate and fair machine learning models. In *Proceedings of the 2021 International Conference on Management of Data*, SIGMOD '21, page 1771–1783, New York, NY, USA, 2021. Association for Computing Machinery.

[62] Yeming Wen, Dustin Tran, and Jimmy Ba. Batchensemble: an alternative approach to efficient ensemble and lifelong learning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[63] Haocheng Xia, Jinfei Liu, Jian Lou, Zhan Qin, Kui Ren, Yang Cao, and Li Xiong. Equitable data valuation meets the right to be forgotten in model markets. *Proc. VLDB Endow.*, 16(11):3349–3362, jul 2023.

[64] Xi Yan, David Acuna, and Sanja Fidler. Neural Data Server: A Large-Scale Search Engine for Transfer Learning Data, March 2020. Number: arXiv:2001.02799 arXiv:2001.02799 [cs].

[65] Jianfei Yang, Xiangyu Peng, Kai Wang, Zheng Zhu, Jiashi Feng, Lihua Xie, and Yang You. DIVIDE TO ADAPT: MITIGATING CONFIRMATION BIAS FOR DOMAIN ADAPTATION OF BLACK-BOX PREDIC-. 2023.

[66] Jinsung Yoon, Sercan Arik, and Tomas Pfister. Data Valuation using Reinforcement Learning. In *Proceedings of the 37th International Conference on Machine Learning*, pages 10842–10851. PMLR, November 2020. ISSN: 2640-3498.

[67] Long Zhao, Ting Liu, Xi Peng, and Dimitris Metaxas. Maximum-entropy adversarial data augmentation for improved generalization and robustness. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA, 2020. Curran Associates Inc.

[68] Zixuan Zhao and Raul Castro Fernandez. Leva: Boosting Machine Learning Performance with Relational Embedding Data Augmentation. In *Proceedings of the 2022 International Conference on Management of Data*, pages 1504–1517, Philadelphia PA USA, June 2022. ACM.

[69] Shuyuan Zheng, Yang Cao, Masatoshi Yoshikawa, Huizhong Li, and Qiang Yan. Flmarket: Trading private models in federated learning. In *2022 IEEE International Conference on Big Data (Big Data)*, pages 1525–1534, 2022.