
Adapting to Online Distribution Shifts in Deep Learning: A Black-Box Approach

Dheeraj Baby¹ Boran Han¹ Shuai Zhang¹ Cuixiong Hu¹ Yuyang Wang¹ Yu-Xiang Wang^{1,2}

¹Amazon

²University of California San Diego

Correspondence to: dheerajbaby@gmail.com

Abstract

We study the well-motivated problem of *online distribution shift* in which the data arrive in batches and the distribution of each batch can change arbitrarily over time. Since the shifts can be large or small, abrupt or gradual, the length of the relevant historical data to learn from may vary over time, which poses a major challenge in designing algorithms that can automatically adapt to the best “attention span” while remaining computationally efficient. We propose a meta-algorithm that takes any network architecture and any Online Learner (OL) algorithm as input and produces a new algorithm which *provably* enhances the performance of the given OL under non-stationarity. Our algorithm is efficient (it requires maintaining only $O(\log(T))$ OL instances) and adaptive (it automatically chooses OL instances with the ideal “attention” length at every timestamp). Experiments on various real-world datasets across text and image modalities show that our method *consistently* improves the accuracy of user specified OL algorithms for classification tasks. Key novel algorithmic ingredients include a *multi-resolution instance* design inspired by wavelet theory and a *cross-validation-through-time* technique. Both could be of independent interest.

1 Introduction

Many real-world Machine Learning (ML) problems can be cast into the framework of online learning where a model continuously learns from an online datastream. For example, consider the task of classifying the gender from high-school yearbook images. Suppose that the data is presented in

Framework 1 Batched online interaction protocol for classification

- 1: Initialize learner’s hypothesis $h_1 : \mathcal{X} \rightarrow \mathcal{Y}$.
 - 2: **for** each round $t \in [T] := \{1, \dots, T\}$ **do**
 - 3: Nature samples $n_t > 1$ covariate-label pairs $(x_1, y_1), \dots, (x_{n_t}, y_{n_t})$ iid from a distribution \mathcal{D}_t on the space $\mathcal{X} \times \mathcal{Y}$.
 - 4: The covariates are revealed to the learner.
 - 5: Learner predicts their labels using h_t .
 - 6: True labels are revealed to the learner.
 - 7: Learner updates its hypothesis to h_{t+1} using (a part of) the revealed labelled data.
 - 8: **end for**
-

an online manner where at each timestamp, the learner is asked to classify images from that timestamp. An online ML model will continuously adjust its parameters based on the data it received sequentially. Shift in the data distribution across the timestamps in a datastream constitutes a significant challenge in the design of online learning algorithms. For instance, in the case of high-school gender classification, the appearance characteristics of a population, such as fashion style or racial diversity, can evolve slowly over time. Such distribution shifts can cause models learned using old data to yield poor performance on the most recent or relevant data distribution. On the other hand, one can leverage an adaptively selected portion of the old data if the distribution is smoothly evolving. An ideal goal is to maximise the accuracy attainable at each timestamp (rather than maximising the average across all timestamps). Effectively handling this problem poses a common challenge in practical applications.

On the other hand, modeling how the data distribution evolves over time requires one to make restrictive assumptions while designing an effective learning algorithm. Unfortunately, most often the distribution shifts are caused by complex confounders which are hard to model [Zhu et al., 2014]. Consequently such assumptions may not be satisfied or even verifiable in practice. This leads to the phenomenon where the strong assumptions about the evolution of distribution can only contribute to more noise than signal into the process of algorithm design.

To study this challenge closely, we consider the problem of online classification under distribution shifts without explicitly modeling the evolution of such shifts. A typical protocol for online classification is presented in Framework 1. In this paper, we use the term **online learner (OL)** to mean any learner that operates under the protocol in Framework 1. There is a rich body of work [Besbes et al., 2015, Zhang et al., 2018a] (see Sec. 2 for a broader overview) that studies principled ways of handling non-stationarity under convex loss functions. However, due to the huge success of deep learning, many of the modern ML systems use deep nets where the convexity assumptions are violated. This limits the applicability of methods that only handle convex losses to relatively simple use cases such as logistic regression, SVMs or fine-tuning the linear layer of a neural network.

Moving forward, our goal will be to effectively adapt to distribution shifts without imposing convexity assumptions on losses. For online learning problems, one can continually update the parameters of the underlying network based on the new data as it sequentially arrives. For example we can use online gradient descent or continual learning algorithms such as [Zenke et al., 2017] in hope to control the generalization error at each round in the online protocol. However, as noted in [Yao et al., 2022], the performance of such methods can be limiting under distribution shifts. In this paper, we provide a meta-algorithm that takes in an arbitrary OL as a black-box and produces a new algorithm that has better classification accuracy under distribution shift. The black-box nature allows us to leverage the success of deep learning while still being able to adapt to distribution shift. This makes our method more widely applicable in practice in comparison to methods that only handle convex losses. Our key contributions are as follows:

- We develop a meta-algorithm AWE (Accuracy Weighted Ensemble, Alg. 2) that takes a black-box OL as input and improve its performance for online classification problem under distribution shifts. The method primarily contains two parts: (1) Multi-Resolution Instances (MRI) and (2) Cross-Validation-Through-Time (CVTT).
- We obtain strong theoretical guarantees. For the MRI design, we show that it covers at least a fourth of all datapoints from most recent distribution (Theorem 1). We also give bounds for the generalization error and dynamic regret of AWE (Theorems 2, 7).
- We conduct experiments on various datasets with *in-the-wild* distribution shifts across image and text modalities and find that our method *consistently* leads to improved performance (Sec. 6) while incurring only a logarithmic overhead in memory and time in comparison to the base OL algorithm.

Notes on key technical novelties.

a) black-box nature: We remark that the idea of enhancing

the performance of a black-box algorithm under non-stationarity has been proposed in [Daniely et al., 2015] (see Appendix C for a comparison). The algorithm in [Daniely et al., 2015] as well as ours are both based on running multiple instances of a base learner, where each instance is trained from a unique time-point in history and combining their predictions at a future time-point. However, algorithmic components that facilitate our black-box reduction differs from theirs in two aspects: i) **Data-efficient instances.** In Appendix C.1, by taking the specific case of a piece-wise stationary distribution shift, we show an example where the Geometric Covering intervals of [Daniely et al., 2015] fails to guarantee existence of an instance that has been trained on adequate amount of data from most recent distribution. Our MRI construction of maintaining base learner instances fixes this problem (see Theorem 1) leading to a more data-efficient way of instantiating the base learners. ii) **Faster regret rates.** Suppose in Framework 1, after each round, N labelled datapoints are revealed. The scheme in [Daniely et al., 2015] guarantees an average regret of $O(1/\sqrt{|I|})$ against the best instance in any interval I . However, when the data distribution is slowly varying (or constant in the best case) within interval I , our scheme lead to a faster average regret of $O(1/\sqrt{N|I|})$. This is attributed to the CVTT technique which pools together datapoints from similar distribution in adjacent timestamps while estimating the loss of each instance in the recent distribution. The high accuracy estimates of losses allows us to quickly learn/identify the most appropriate instance for the recent distribution leading to fast regret rates. (see Appendix C.2 for further details)

b) logarithmic overhead: Our MRI construction requires to maintain only a pool of $O(\log T)$ OL instances while guaranteeing the existence of instances in the pool that has been trained on reasonable amount of data exclusively from the recent distribution (see Theorem 1).

c) comparison to [Mazzetto and Upfal, 2023]: We remark that a recent break-through due to [Mazzetto and Upfal, 2023] also provides a way for adapting to distribution shifts. However, their method involves solving *multiple* ERM procedures at *each* timestamp which is hard to deploy in online datastreams. In contrast, we introduce novel algorithmic components (see Sec. 4.2) that facilitate the adaptation of any *user specified black-box* OL algorithms while obviating the need to solve multiple ERM procedures. The reason why [Mazzetto and Upfal, 2023] needs to perform multiple ERM procedures is that in their algorithm they need to compute the maximum mean discrepancy wrt a large hypothesis class. Our key observation is that (under piece-wise stationary distributions) we can compress a large hypothesis class to a set of finite learnt models with at-least one model being good for making predictions for the most recent distribution (see Theorem 1). Hence it suffices to compute the maximum mean discrepancy wrt this *finite* set of models thereby leading to computational savings.

2 Related Work

In this section, we briefly recall the works that are most related to our study. Adapting to distribution shifts under convex losses is well studied in literature [Hazan and Seshadhri, 2007, Zhang et al., 2018a, Zhang et al., 2018b, Cutkosky, 2020, Baby and Wang, 2021a, Zhao et al., 2020, Zhao et al., 2022, Baby and Wang, 2022, Baby and Wang, 2023, Baby et al., 2023b]. The strong requirement of convexity of losses limits their applicability to deep learning based solutions. Further, none of these works aim at optimizing Eq.(1). Methods in [Awasthi et al., 2023, Jain and Shenoy, 2023] provide a non-black-box way to adapt to distribution shift in offline problems. There are also various online learning algorithms coming from rich body of literature involving continual learning and invariant risk minimization. Examples include but not limited to [Zenke et al., 2017, Kirkpatrick et al., 2017, Zhai et al., 2023, Li and Hoiem, 2016, Lee et al., 2017, Aljundi et al., 2018, Rebuffi et al., 2017, Chaudhry et al., 2019, Cai et al., 2021]. We refer the reader to [Delange et al., 2021, Yang et al., 2023] for a detailed literature survey. The aforementioned algorithms can be taken as the input OL for our methodology. Examples of schemes that tackle distribution shift under limited amount of labeled data include [Lipton et al., 2018, Bai et al., 2022, Wu et al., 2021, Baby et al., 2023a, Garg et al., 2023, Rosenfeld and Garg, 2023]. However, they require structural assumptions like label shift on the distribution shift owing to scarce labelled data available and hence form a complementary direction to our work.

We emphasize that our objective is not to attain the best classification accuracy across all algorithms. Instead, our goal is to propose an effective meta-algorithm that can enhance the accuracy of any given online algorithm for classification under distribution shifts.

3 Problem Setting

In this section, we define the notations used and metrics of interest that we aim to control. We use $[T] := \{1, \dots, T\}$ and $[a, b] := \{a, \dots, b\} \subseteq [T]$. Suppose we are at the beginning of round t . Let \mathcal{X} denote the covariate space and \mathcal{Y} denote the label space. Suppose that the data distribution at round t is \mathcal{D}_t . Let i denote an OL instance. Let $\text{Acc}_t(i) := E_{(x,y) \sim \mathcal{D}_t} [\mathbb{I}\{i(x) = y\}]$ where $i(x)$ is the prediction of the model i for covariate x and \mathbb{I} is the binary indicator function. The accuracy $\text{Acc}_t(i)$ is the population level accuracy of model i for the data distribution at round t . The black-box OL we take in as input to AWE will focus on updating the parameters of an underlying neural network architecture. Let \mathcal{H} be the hypothesis class defined by the underlying neural network classifier. Let $h_t^* = \arg\max_{h \in \mathcal{H}} \text{Acc}_t(h)$ be the best classifier for making prediction at round t . Suppose h_t is the classifier used by our algorithm to make predictions for round t . One of the

Algorithm 3 `refineAccuracy`: Inputs: 1) An OL instance M ; 2) A terminal time τ and 3) Failure probability δ .

- 1: Let $n(r)$ denote the number of hold out-data points accumulated in the interval $[\tau - r + 1, \dots, \tau]$.
- 2: $S(r, \delta) := \sqrt{\log(T/\delta)/r} + \sqrt{20 \log T/r}$; Initialize $r = 1$.
- 3: **while** $r \leq \tau/2$ **do**
- 4: Let $u_M(r)$ be the empirical accuracy for the model M estimated using hold-out data in the rounds $[\tau - r + 1, \dots, \tau]$.
- 5: If $|u_M(r) - u_M(2r)| \leq 4S(n(r), \delta)$ then $r \leftarrow 2 \cdot r$; Else return $u(r)$
- 6: **end while**
- 7: Return $u_M(r)$

metrics we are interested in is controlling the instantaneous regret:

$$\text{Err}(t) = \text{Acc}_t(h_t^*) - \text{Acc}_t(h_t), \quad (1)$$

for the maximum number of rounds possible. Doing so implies that the accuracy of our algorithm stays close to the best attainable performance by any classifier in \mathcal{H} across *most* rounds. However, in rounds where the data distribution is very different from the past seen distributions, any algorithm will have to pay an unavoidable price proportional to the discrepancy between the distributions.

We also provide guarantees for the dynamic regret given by:

$$R_{\text{dynamic}} = \sum_{t=1}^T \text{Err}(t) = \sum_{t=1}^T \text{Acc}_t(h_t^*) - \text{Acc}_t(h_t). \quad (2)$$

Controlling $\text{Err}(t)$ is more challenging than controlling the dynamic regret because the former can imply later. Bounding $\text{Err}(t)$ leads to a more stringent control of accuracy at each round in the datastream. This is one of the key formalizations that differentiates our setting from [Daniely et al., 2015]. The control over instantaneous regret is translated to our experiments (see Sec. 6) by the improved accuracy of our meta-algorithm across *most* of the timestamps in the data stream (see also Remark 4).

4 Algorithm

In this section, we present our algorithm for handling distribution shifts and elaborate on the intuition behind the design principles. A formal treatment will be presented in the next section. Along the way we also explain the challenges faced and how they are overcome via algorithmic components. The full algorithm is presented in Alg.2.

Throughout the design of the algorithm, we assume that distribution of data at round t is same as the distribution of the data at time $t + 1$ for most rounds. (More precisely, the number of times the distribution switches is assumed

Algorithm 2 AWE: inputs: 1) A black-box online learning algorithm; 2) Failure probability δ and 3) Split probability p .

```

1: Initialize  $w_1 := \mathbf{1} \in \mathbb{R}^{\log_2 T}$ .
2: Enumerate set of rules  $\mathbf{r} = \{r^1, r^2, \dots, r^m\}$ 
3: for  $t \in [T]$ : do
4:   Get  $n_t$  covariates  $x_t(1), \dots, x_t(n_t)$ .
5:   Compute  $\mathcal{A}_t := \text{ACTIVE}(t)$  (See Eq.(3)).
6:   for  $i \in n_t$ : do
7:     Predict a category by giving  $x_t(i)$  as input to
       currentModel
8:   end for
9:   Get labels  $y_1, \dots, y_{n_t}$ . Let  $p$  fraction of the data be
     allocated to a training fold and remaining to a
     hold-out fold.
10:  Train models in  $\mathcal{A}_t$  with the training fold using the
     online learning algorithm given as input.
11:  Compute  $\mathcal{A}_{t+1} := \text{ACTIVE}(t+1)$ .
12:  For each model  $i \in \mathcal{A}_{t+1}$ , compute the accuracy
     estimate  $\widehat{\text{Acc}}_{t+1}^{(i)} = \text{refineAccuracy}(i, t, \delta)$ 
     (see Alg.3).
13:  Convert the accuracy values  $\{\widehat{\text{Acc}}_{t+1}^{(i)}\}_{i=1}^{|\mathcal{A}_{t+1}|}$  to
     weights  $w_{t+1} \in \mathbb{R}^{|\mathcal{A}_{t+1}|}$  (Eq.(4)).
14:  Construct the model  $E_{t+1} : x \rightarrow$ 
      $\arg\max_{k \in [K]} \sum_{i \in \mathcal{A}_{t+1}} w_{t+1}(i) \text{logit}_i[k]$ , where
      $\text{logit}_i \in \mathbb{R}^K$  is the logits of the model  $i \in \mathcal{A}_{t+1}$  for
     a given input covariate.
15:  Let  $i_{t+1}^* = \arg\max_i \widehat{\text{Acc}}_{t+1}^{(i)}$ .
16:  if  $\text{refineAccuracy}(E_{t+1}, t, \delta) >$ 
      $\text{refineAccuracy}(i_{t+1}^*, t, \delta)$ , then set
     currentModel as  $E_{t+1}$  else set
     currentModel as  $i_{t+1}^*$ .
17: end for
    
```

to grow sub-linearly over time). Such an assumption can be considered as relatively weak. On the other hand, if the number of points where the distribution switches grow linearly with time, it can be shown that learning is impossible in such a regime [Zhang et al., 2018a]. We do not assume any prior knowledge/modelling assumptions on where the switches happen.

The input to AWE is a user-specified OL. We would like to improve the accuracy of the given OL under distribution shifts. To motivate our techniques informally, consider the following thought experiment. Suppose in the interval $[1, t_0]$ the data is generated from a distribution \mathcal{D}_1 while in the interval $[t_0, T]$ it is from a sufficiently different distribution \mathcal{D}_2 . If we know that a change in distribution has happened at time t_0 , then we can start a new OL instance from time t_0 . We remind the reader that when a new instance is started, its internal states will also be re-initialized. However absent such knowledge, one naive idea is to start

a new instance at every past timestamp in the datastream. Then combine their predictions (based on validation accuracies) at a future timestamp. Unfortunately, such a scheme can be computationally expensive making it less attractive in practice. Further the question of how to combine the predictions from various instances in a statistically efficient manner also remains unclear.

It is challenging to maintain a pool of instances such that: 1) the growth rate of the pool's size is slow and 2) An instance that has been trained on adequate amount of data from the most recent distribution exists in the pool. In the next section, we discuss our solution to address the above problems. The solution is based on carefully adapting the idea of Multi Resolution Analysis (MRA) from wavelet theory [Mallat, 1999].

4.1 Multi-resolution Instances (MRI)

For the sake of simplicity, let the time horizon T be a power of 2. We define $M := \log_2 T$ resolutions. In each resolution $i \in [1, M]$ we define two collection of intervals as follows:

- $R := \{[1 + (k-1)T/2^{i-1}, kT/2^{i-1}] \text{ for } k \in \mathbb{N}\}$
- $B := \{[1 + (k-1)T/2^{i-1} + T/2^i, kT/2^{i-1} + 3 \cdot T/2^i] \text{ for } k \in \mathbb{N}\}$,

where we disregard the timepoints in an interval that exceeds the horizon T . We remark that intervals in the collection B are not present in the MRA defined by usual wavelet theory. However, we include them to quickly pick up distribution shifts as formalised in Theorem 1. We refer the reader to Appendix B for a specific example demonstrating the motivation for including the set B . See Fig.1 for a depiction of the intervals defined by the MRI construction.

With each interval, we associate an OL instance (hence interval and its associated instance are used interchangeably moving forward). For example, associated with an interval $[a, b]$ we define an OL instance that starts its operation at round a and it is only used to make predictions within the interval $[a, b]$. For making a prediction at an intermediate time $t \in [a, b]$ this OL instance will only be trained on the data that is seen in the duration $[a, t-1]$. However, we remind the reader that prediction made by this instance at round t may not be the final prediction submitted by the overall meta-algorithm. We remark that the instance defined by the interval $[a, b]$ is no longer used for making predictions at rounds $t > b$.

For any round t , define

$$\text{ACTIVE}(t) := \{u \in R \cup B | t \in u\}. \quad (3)$$

This defines the collection of instances that are active at round t . The meta-algorithm will form a prediction only based on the active models. Due to construction of the

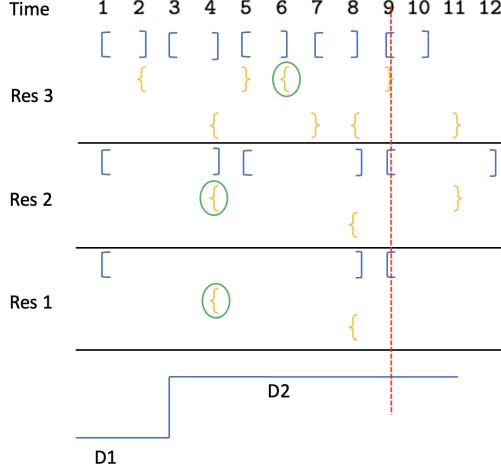


Figure 1: The figure shows the configuration of Multi Resolution Instances (MRI). Brackets of type $[]$ belongs to the collection R and type $\{ \}$ belongs to collection B (see Sec. 4.1). Consider the scenario where the data distribution has changed from timestamp 3 and remained stable afterwards. Suppose we are at the beginning of round 9 and after each round we get n training data points. So we have seen $6n$ labelled data points from distribution \mathcal{D}_2 . $\text{ACTIVE}(9)$ corresponds to those intervals that include the timestamp 9. The circled intervals has seen at least $3n$ data points from distribution \mathcal{D}_2 thereby ensuring models that are present in the active set with good performance under distribution \mathcal{D}_2 . A formal result of the data utilization efficiency of the MRI construction is proved in Theorem 1.

intervals, it is straight-forward to see that at any round $t \in [T]$, we have $|\text{ACTIVE}(t)| = O(\log T)$.

In Theorem 1, we show that there exists a model in the MRI pool that has seen sufficient amount of data from a new distribution if a shift happens. Consequently we are able to maintain only logarithmic number of instances while still guaranteeing that there exists at least one instance in the pool that is efficient to make predictions for the most recent distribution. We emphasize that such a property is achieved without imposing strong modelling assumptions on the evolution of the shift.

4.2 Cross-Validation-Through-Time (CVTT)

Now that we have a collection of instances, we next turn our attention to address the statistical challenge of combining the instances to make predictions that can lead to high accuracy. As mentioned in Framework 1, the labels for all covariates are revealed after each round t . We then split the data into a training fold and a hold-out fold (line 11 in Alg. 2). Data from the training fold is fed to the OL instances in $\text{ACTIVE}(t)$ to resume their training.

At round $t + 1$, we make predictions using the models in $\mathcal{A}_{t+1} := \text{ACTIVE}(t + 1)$. Recall that throughout the design

of our algorithm, we assume that the distribution of data at round t is close to the distribution of the data at time $t + 1$ for most rounds. Under such an assumption, the empirical accuracy for models in \mathcal{A}_{t+1} computed using the hold-out data at round t should be a rough estimate for the accuracy of those models for the data revealed at round $t + 1$. Since we have only a limited amount of validation data, even if the distribution at time $t + 1$ is identical to that at time t , such an estimate can be misleading. On the other-hand if the data distribution during the interval $[r, t]$ is relatively stable, then we can use all the hold-out data collected within $[r, t]$ to get a better estimate of the accuracy. To get such refined accuracy estimates (which in essence does a CVTT) for each model in \mathcal{A}_{t+1} , we use the recent advancement in [Mazzetto and Upfal, 2023] adapted to our setting in Alg. 3. This idea of estimating refined accuracy for each instance that are present in a sparse pool of instances is what eliminates the need of using the techniques in [Mazzetto and Upfal, 2023] that rely on expensive ERM computations. In contrast, techniques in [Mazzetto and Upfal, 2023] require to compute a metric of the form $\sup_{M \in \mathcal{H}} |u_M(r) - u_M(2r)|$ for a very large hypothesis class \mathcal{H} defined by the neural net architecture (cf. Line 4 of Alg. 3 for definition of $u_M(r)$). They use ERM for this purpose. However by virtue of our MRI construction and Theorem 1, we compress the large hypothesis class \mathcal{H} to a finite set of *learnt* models (with at least one model that has seen at least a fourth of the datapoints from the current distribution). Hence it suffices only to compute such discrepancy metrics over a finite set formed by candidates for near optimal hypothesis for the current round. This facilitate the speedup over their methods. To the best of our knowledge, the idea of using techniques in [Mazzetto and Upfal, 2023] to facilitate a CVTT has not been used before in literature.

Once the accuracy for each model in \mathcal{A}_{t+1} is estimated, we form an ensemble model E_{t+1} from those constituent models with weights specified by:

$$w_i^{t+1}(i) = \widehat{\text{Acc}}_{t+1}^{(i)}, \quad (4)$$

where w_i^{t+1} is the weight assigned to model i in round $t + 1$ and $\widehat{\text{Acc}}_{t+1}^{(j)}$ is the estimated accuracy for model j . Then based on the refined accuracy estimate of all models (line 16), we pick a model to make predictions at line 7. This ensembling model is mainly introduced to get better performance in practice while not hurting the theoretical guarantees. It is perfectly possible to prefer any other ensembling scheme as well and still the guarantees of Theorem 2 remains valid. The overall algorithm is displayed in Alg. 2.

5 Theory

In this section, we present theoretical justifications of the algorithmic components of AWE. All proofs are deferred to Appendix B. For the sake of simplicity assume that we

receive n labelled training data points and m hold-out data points after the end of each timestamp in Step 11 of Alg. 2. Next theorem shows the ability of MRI to maintain instances in the pool that can lead to good predictions for the most recent distribution.

Theorem 1. *Suppose we are at the beginning of a timestamp $t + 1$ and the data distribution has remained constant from some round $t_0 < t + 1$. Let this distribution be \mathcal{D} . We have labelled hold-out data available till round t . There exists at least one instance in the MRI pool that is active at a given round and satisfies at least one of the following properties:*

- *All the training data seen from the model is from distribution \mathcal{D} . Further it has seen at least $(t - t_0 + 1)n/2$ points from distribution \mathcal{D} .*
- *The model been only trained on data from \mathcal{D} . Further the number of points seen by the model is at least $(t - t_0 + 1)n/4$.*

Next, we attempt to understand the statistical efficiency of AWE. We define some notations first. Suppose the number of classes is 2. Let $\mathcal{F} := \{E_t\} \cup \mathcal{A}_t$. Let $f_t^* = \arg\max_{f \in \mathcal{F}} \text{Acc}_t(f)$. $\hat{f}_t := \arg\max_{f \in \mathcal{F}} \widehat{\text{Acc}}_t(f)$. Let $h_t^* = \arg\max_{h \in \mathcal{H}} \text{Acc}_t(h)$.

Theorem 2. *Assume the notations defined in Sec. 3. Suppose we are at the beginning of round t and that the data is sampled independently across timestamps. Assume that the data distribution (say \mathcal{D}) has remained constant in $[t - r, t]$. Then with probability at least $1 - 4\delta \log T \log mT$, instantaneous regret at round t for AWE,*

$$\text{Acc}_t(h_t^*) - \text{Acc}_t(\hat{f}_t) = \text{Acc}_t(h_t^*) - \text{Acc}_t(f_t^*) + \tilde{O}\left(\sqrt{1/mr}\right),$$

where \tilde{O} hides logarithmic factors in T, m, r and $1/\delta$.

Remark 3. *Theorem 2 must be interpreted in the light of Theorem 1. The first term $\text{Acc}_t(h_t^*) - \text{Acc}_t(g_t^*)$ depends on the generalization behaviour of the online learning algorithm given as input to AWE. However, due to Theorem 1, it is guaranteed that there exists at least one model in $\text{ACTIVE}(t)$ that has seen an $\Omega(mr)$ data points from the distribution \mathcal{D} when the data distribution has remained constant in $[t - r, t]$. This helps to keep the first term small. However, a theoretical bound on first term will depend on the specific OL algorithm used. For example, if we use ERM as the base learner, the first term can be bound by $O(1/\sqrt{nr})$. The second term $\tilde{O}(\sqrt{1/mr})$ reflects improvement in generalization obtained by adaptively using all the hold-out data from previous r rounds where the distribution has remained unchanged. We remark that the prior knowledge of r (or the change-point) is not required.*

Remark 4. *Let i_0 be the OL instance started from timestamp 1. By the construction of MRI in Sec. 4.1, $i_0 \in \text{ACTIVE}(t)$ for all rounds t . Whenever $g_t^* = i_0$, the above theorem guarantees that the accuracy of AWE is comparable to that of i_0 at the asymptotically decaying margin of*

$O(1/\sqrt{mr})$. On the other hand, the distribution shifts can potentially cause the existence of new models $g_t^ \in \mathcal{G}$ such that $\text{Acc}_t(g_t^*) \gg \text{Acc}_t(i_0)$. In such scenarios, Theorem 2 again guarantees that the accuracy of AWE (with map $\hat{\mathcal{W}}$) is comparable to that of $\text{Acc}_t(g_t^*)$ thereby improving the performance over the instance i_0 . This solidifies the ability of AWE to enhance the performance of a user-given OL.*

The statement of Theorem 2 translates into an excess risk bound against the pool of MRI instances. If we choose the input OL to AWE as ERM, one can get an excess risk bound of $O(1/\sqrt{mr})$ against an entire hypothesis class across which ERM is performed. However in an online setting, running ERM at each round can be computationally prohibitive. Hence the alternate choices for the OL (for example continual learning algorithms) becomes more useful. In such a scenario, Theorem 2 guarantees an excess risk bound against an optimally restarted instantiation of the chosen OL – thanks to the data-efficiency guarantees of MRI construction (Theorem 1).

Even-though we presented our analysis for piecewise stationary distributions, the extension to slowly varying distributions is straightforward by standard discretization arguments. In the analysis we can divide the time horizon into intervals where distribution is slowly varying and apply Theorem 2 to the mean distribution within the bin while paying a small additive price proportional to how much the distribution at a round deviates from the mean. In Appendix B, we characterize the dynamic regret (see Eq.(2)) of AWE on slowly changing distribution shifts. Though AWE has the drawback of storing all the previous hold-out data points, we remark that size of hold-out data is generally much smaller than the training data set size. In practice, one can limit to store only the hold-out data from a fixed amount of past timestamps. While refined accuracy calculation necessitates a forward inference pass on this data, the efficiency of modern GPUs prevents significant performance issues.

6 Experiments

6.1 Empirical Study on Real-world Datasets

All our experiments are conducted on the WildTime benchmark [Yao et al., 2022]. WildTime constitutes a suite of datasets for classification problems across image, text and tabular modalities that exhibit natural distribution shift. In this paper, our focus is on the image and text modalities. For the image modality, the WildTime benchmark comprises the FMOW dataset, while for text, it includes the HuffPost and Arxiv datasets. We direct the reader to [Yao et al., 2022] for more elaborate details regarding the datasets. All the experiments were conducted on NVIDIA A100 GPUs.

For each of the dataset, we consider different black-box OL algorithms as input to AWE (Alg. 2). We then compare the accuracy of each OL and AWE method across every time

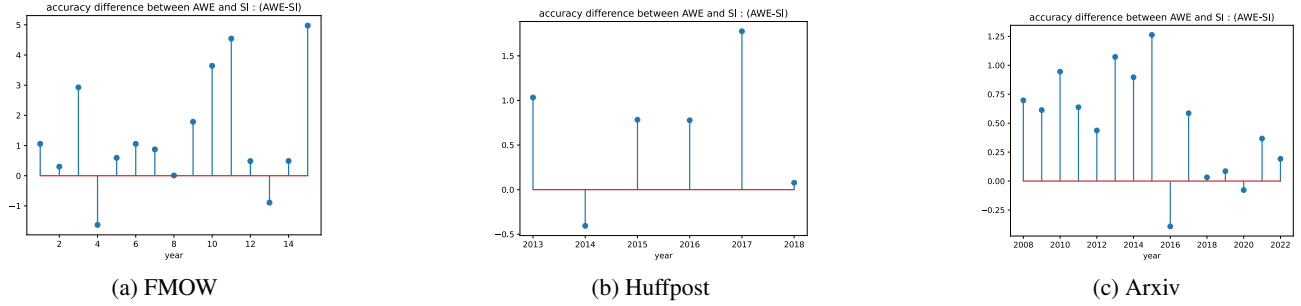


Figure 2: % accuracy differences across various timestamps when AWE is run with SI as the online learning algorithm. We report similar results for other OL algorithms and the fraction of timestamps where AWE improves (or does not degrade) the performance of the base OL in Appendix D.

Table 1: Performance statistics for image (FMOw dataset) and text (Huffpost & Arxiv datasets) modalities. We report the difference in average classification accuracy (%) across all timestamps obtained by a black-box scheme minus that of the input OL.

Input OL	FMOw		Huffpost		ArXiv	
	SAOL [Daniely et al., 2015]	AWE	SAOL [Daniely et al., 2015]	AWE	SAOL [Daniely et al., 2015]	AWE
SI [Zenke et al., 2017]	-4.19 ± 0.119	1.52 ± 0.067	-0.70 ± 0.068	0.70 ± 0.065	-3.97 ± 0.159	0.45 ± 0.009
FT	-3.71 ± 0.104	1.83 ± 0.073	0.71 ± 0.06	0.72 ± 0.069	-3.95 ± 0.14	0.56 ± 0.01
IRM [Arjovsky et al., 2019]	-6.16 ± 0.132	0.55 ± 0.04	0.37 ± 0.049	0.98 ± 0.08	-2.67 ± 0.131	0.13 ± 0.005
EWC [Kirkpatrick et al., 2017]	-3.50 ± 0.101	2.20 ± 0.08	-0.53 ± 0.059	0.72 ± 0.069	-3.87 ± 0.157	0.36 ± 0.008
CORAL [Sun and Saenko, 2016]	-2.98 ± 0.093	3.24 ± 0.097	0.18 ± 0.034	1.10 ± 0.085	-1.16 ± 0.087	1.67 ± 0.018

stamp. To clarify further, at any timestamp, we compute the accuracy only using the labelled data that is revealed towards the end of that timestamp (see Framework 1). We explore the performance of five different online learning algorithms: Synaptic Intelligence (SI) [Zenke et al., 2017], Invariant Risk Minimisation (IRM) [Arjovsky et al., 2019], FineTuning (FT), Elastic Weight Consolidation (EWC) [Kirkpatrick et al., 2017], and CORAL [Sun and Saenko, 2016]. We use the same neural network architectures and hyper-parameter choices for the OL as used by the eval-stream setting in [Yao et al., 2022]. We refer the reader to [Yao et al., 2022] for comprehensive information about these design choices.

The performance compared with SI along each timestamp is shown in Fig. 2 for each dataset. Similar results for other OL algorithms are presented in Appendix D. We combine the predictions of a model at round t using the accuracy estimates based on the data revealed until time $t-1$. Occasional drops in performance at certain timestamps may result from abrupt changes in the distribution. If the distribution at time t is sufficiently different from that at time $t-1$, the accuracy estimates we use to combine the instances can be not useful for making predictions at round t . However, our algorithm demonstrates a rapid adjustment to the new distribution, as evidenced by performance improvements shortly after the timestamps where a performance decline was observed.

To statistically summarize the performance, we report the % accuracy differences (i.e., $AWE - OL$) across various time

stamps. in Table 1. Additionally, in Appendix D, we detail the win/draw/lose numbers along the timestamps. We find that in well above 50% of the timestamps across all cases, AWE can improve the performance of the base OL instance. This indicates the robustness of AWE to deliver improved performance under *in-the-wild distribution shifts*.

We also compare AWE with Strongly Adaptive Online Learning (SAOL) [Daniely et al., 2015]. To the best of our knowledge, SAOL is the only black-box adaptation scheme applicable to online non-convex setting. All runs are repeated with 3 random seeds. We compute the difference in average accuracy (i.e $AWE - OL$ or $SAOL - OL$) over all timestamps attained by both schemes in Table 1. In contrast, we find that SAOL often degrades the performance of the input OL which can be alluded to the reasons presented in part (a) of notes on technical novelties in Sec. 1.

6.2 Ablation study

Ablation on resolution: To further substantiate the efficacy of the multi-resolution instance paradigm (Sec. 4.1), we compare the performance obtained by AWE and an alternative where we only maintain instances from a single resolution in the MRI construction. Using a single resolution has the effect of dividing the entire time horizon into fixed-size intervals along the sets R and B (that are potentially overlapping; Sec. 4.1). Each window is associated with an OL instance trained exclusively on the duration of

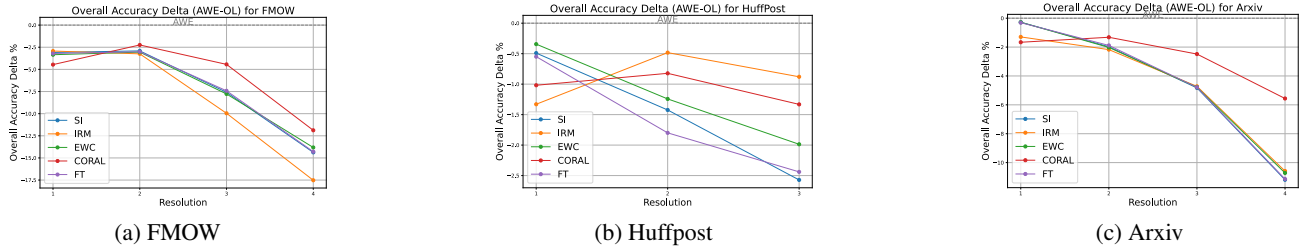


Figure 3: Ablation study across various resolutions. We compute the overall accuracy attained by AWE minus that attained by using only a single resolution in the MRI pool. We see that in most cases AWE outperforms the single resolution counterparts. Further, by virtue of using AWE, the user does not need to hand-tune the optimal resolution to use in an MRI pool.

that interval. We see in Fig.3 that in most of the cases AWE attains superior performance. Choosing an optimal window size (and hence the resolution number) requires prior knowledge of the type of shift. In contrast, AWE eliminates the need for this window-size selection by adaptively combining instances from multiple resolutions, automatically assigning a higher weight to the higher-performing resolutions during prediction. We refer the reader to Appendix D for more experimental results.

7 Conclusion and Future Work

We proposed a method to enhance the performance of any user given OL for classification by merging predictions from different historical OL instances. The MRI construction for limiting the instance-pool size could be of independent interest for tackling non-stationarity. Experiments across various real world datasets indicate that our method *consistently* improves the performance of user-specified OL algorithms. Future work include extending the methods to change point detection and online learning with limited feedback. Limitations can be found at Appendix A.

References

- [Aljundi et al., 2018] Aljundi, R., Babiloni, F., Elhoseiny, M., Rohrbach, M., and Tuytelaars, T. (2018). Memory aware synapses: Learning what (not) to forget. In *Computer Vision – ECCV 2018: 15th European Conference, Munich, Germany, September 8–14, 2018, Proceedings, Part III*.
- [Arjovsky et al., 2019] Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. (2019). Invariant risk minimization. *arXiv preprint arXiv:1907.02893*.
- [Awasthi et al., 2023] Awasthi, P., Cortes, C., and Mohri, C. (2023). Theory and algorithm for batch distribution drift problems. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*.
- [Baby et al., 2023a] Baby, D., Garg, S., Yen, T.-C., Balakrishnan, S., Lipton, Z. C., and Wang, Y.-X. (2023a). Online label shift: Optimal dynamic regret meets practical algorithms. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- [Baby and Wang, 2021a] Baby, D. and Wang, Y.-X. (2021a). Optimal dynamic regret in exp-concave online learning. In *Conference on Learning Theory*, pages 359–409. PMLR.
- [Baby and Wang, 2021b] Baby, D. and Wang, Y.-X. (2021b). Optimal dynamic regret in exp-concave online learning. In *COLT*.
- [Baby and Wang, 2022] Baby, D. and Wang, Y.-X. (2022). Optimal dynamic regret in LQR control. In *Advances in Neural Information Processing Systems*.
- [Baby and Wang, 2023] Baby, D. and Wang, Y.-X. (2023). Second order path variationals in non-stationary online learning. *AISTATS*.
- [Baby et al., 2023b] Baby, D., Xu, J., and Wang, Y.-X. (2023b). Non-stationary contextual pricing with safety constraints. *Transactions on Machine Learning Research*.
- [Bai et al., 2022] Bai, Y., Zhang, Y.-J., Zhao, P., Sugiyama, M., and Zhou, Z.-H. (2022). Adapting to online label shift with provable guarantees. In *Advances in Neural Information Processing Systems*.
- [Besbes et al., 2015] Besbes, O., Gur, Y., and Zeevi, A. (2015). Non-stationary stochastic optimization. *Operations research*, 63(5):1227–1244.
- [Bousquet et al., 2003] Bousquet, O., Boucheron, S., and Lugosi, G. (2003). Introduction to statistical learning theory. In *Advanced Lectures on Machine Learning, Lecture Notes in Computer Science*. Springer.
- [Cai et al., 2021] Cai, Z., Sener, O., and Koltun, V. (2021). Online continual learning with natural distribution shifts: An empirical study with visual data. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*.

- [Chaudhry et al., 2019] Chaudhry, A., Rohrbach, M., Elhoseiny, M., Ajanthan, T., Dokania, P. K., Torr, P. H. S., and Ranzato, M. (2019). On tiny episodic memories in continual learning. *arXiv: Learning*.
- [Cutkosky, 2020] Cutkosky, A. (2020). Parameter-free, dynamic, and strongly-adaptive online learning. In *ICML*.
- [Daniely et al., 2015] Daniely, A., Gonen, A., and Shalev-Shwartz, S. (2015). Strongly adaptive online learning. In *International Conference on Machine Learning*, pages 1405–1411.
- [Delange et al., 2021] Delange, M., Aljundi, R., Masana, M., Parisot, S., Jia, X., Leonardis, A., Slabaugh, G., and Tuytelaars, T. (2021). A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- [Garg et al., 2023] Garg, S., Erickson, N., Sharpnack, J., Smola, A., Balakrishnan, S., and Lipton, Z. (2023). Rls-bench: Domain adaptation under relaxed label shift. In *International Conference on Machine Learning (ICML)*.
- [Hazan and Seshadhri, 2007] Hazan, E. and Seshadhri, C. (2007). Adaptive algorithms for online decision problems. In *Electronic colloquium on computational complexity (ECCC)*, volume 14.
- [Jain and Shenoy, 2023] Jain, N. and Shenoy, P. (2023). Instance-conditional timescales of decay for non-stationary learning.
- [Kirkpatrick et al., 2017] Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., et al. (2017). Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*.
- [Lee et al., 2017] Lee, S.-W., Kim, J.-H., Jun, J., Ha, J.-W., and Zhang, B.-T. (2017). Overcoming catastrophic forgetting by incremental moment matching. In *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS’17*.
- [Li and Hoiem, 2016] Li, Z. and Hoiem, D. (2016). Learning without forgetting. In *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part IV*.
- [Lipton et al., 2018] Lipton, Z., Wang, Y.-X., and Smola, A. (2018). Detecting and correcting for label shift with black box predictors. In *International Conference on Machine Learning*.
- [Mallat, 1999] Mallat, S. (1999). *A wavelet tour of signal processing*. Elsevier.
- [Mazzetto and Upfal, 2023] Mazzetto, A. and Upfal, E. (2023). An adaptive algorithm for learning with unknown distribution drift. In *Advances in Neural Information Processing Systems*.
- [Mohri et al., 2012] Mohri, M., Rostamizadeh, A., and Talwalkar, A. (2012). *Foundations of Machine Learning*. The MIT Press.
- [Rebuffi et al., 2017] Rebuffi, S., Kolesnikov, A., Sperl, G., and Lampert, C. H. (2017). icarl: Incremental classifier and representation learning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*.
- [Rosenfeld and Garg, 2023] Rosenfeld, E. and Garg, S. (2023). (almost) provable error bounds under distribution shift via disagreement discrepancy. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- [Sun and Saenko, 2016] Sun, B. and Saenko, K. (2016). Deep CORAL: correlation alignment for deep domain adaptation. In *Computer Vision - ECCV 2016 Workshops - Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III*.
- [Wu et al., 2021] Wu, R., Guo, C., Su, Y., and Weinberger, K. Q. (2021). Online adaptation to label distribution shift. In *Advances in Neural Information Processing Systems*.
- [Yang et al., 2023] Yang, X., Song, Z., King, I., and Xu, Z. (2023). A survey on deep semi-supervised learning. *IEEE Trans. on Knowl. and Data Eng.*
- [Yao et al., 2022] Yao, H., Choi, C., Cao, B., Lee, Y., Koh, P. W., and Finn, C. (2022). Wild-time: A benchmark of in-the-wild distribution shift over time. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.
- [Zenke et al., 2017] Zenke, F., Poole, B., and Ganguli, S. (2017). Continual learning through synaptic intelligence. In *International Conference on Machine Learning*. PMLR.
- [Zhai et al., 2023] Zhai, R., Schroedl, S., Galstyan, A., Kumar, A., Steeg, G. V., and Natarajan, P. (2023). Online continual learning for progressive distribution shift (ocl-pds): A practitioner’s perspective. In *ICLR 2023 Workshop on Successful Domain Generalization*.
- [Zhang et al., 2018a] Zhang, L., Lu, S., and Zhou, Z.-H. (2018a). Adaptive online learning in dynamic environments. In *Advances in Neural Information Processing Systems (NeurIPS-18)*, pages 1323–1333.
- [Zhang et al., 2018b] Zhang, L., Yang, T., Zhou, Z.-H., et al. (2018b). Dynamic regret of strongly adaptive methods. In *International Conference on Machine Learning (ICML-18)*, pages 5877–5886.

[Zhao et al., 2022] Zhao, P., Xie, Y.-F., Zhang, L., and Zhou, Z.-H. (2022). Efficient methods for non-stationary online learning. In *Advances in Neural Information Processing Systems*.

[Zhao et al., 2020] Zhao, P., Zhang, Y., Zhang, L., and Zhou, Z.-H. (2020). Dynamic regret of convex and smooth functions. *NeurIPS*.

[Zhu et al., 2014] Zhu, C., Zhu, H., Ge, Y., Chen, E., and Liu, Q. (2014). Tracking the evolution of social emotions: A time-aware topic modeling perspective. In *2014 IEEE International Conference on Data Mining, ICDM 2014, Shenzhen, China, December 14-17, 2014*.

Checklist

In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all models and algorithms presented, check if you include:
 - (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
 - (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
 - (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes. Please check the supplementary zip file.]
2. For any theoretical claim, check if you include:
 - (a) Statements of the full set of assumptions of all theoretical results. [Yes]
 - (b) Complete proofs of all theoretical results. [Yes]
 - (c) Clear explanations of any assumptions. [Yes]
3. For all figures and tables that present empirical results, check if you include:
 - (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
 - (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
 - (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
 - (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
 - (a) Citations of the creator If your work uses existing assets. [Yes]
 - (b) The license information of the assets, if applicable. [Yes]
 - (c) New assets either in the supplemental material or as a URL, if applicable. [Yes]
 - (d) Information about consent from data providers/curators. [Yes]
 - (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]
5. If you used crowdsourcing or conducted research with human subjects, check if you include:
 - (a) The full text of instructions given to participants and screenshots. [Not Applicable]
 - (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
 - (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

A Limitations and Further Discussion

Limitations. One of the main limitations of our work is that we only try to optimize the classification accuracy at any round. However for the case of continual learning based algorithms, other metrics which measure the power of remembering old experiences to prevent catastrophic forgetting is also often optimized. Such metrics are often termed as backward transfer metrics. Our method do not explicitly optimize metrics that measure backward transfer, though it could be internally optimized by the different instances we maintain in the ensemble pool. Further, our algorithms do not take into account protecting the privacy of sensitive training data.

Further discussion. Adapting to distribution shifts is a well studied problem in online learning literature [Zhang et al., 2018a, Zhao et al., 2020, Baby and Wang, 2021b]. However, these works are studied under convexity assumptions on the loss functions. To the best of our knowledge adaptive regret minimization works like [Daniely et al., 2015] are the only ones that are also applicable to the deep learning framework. Eventhough they are backed by strong theoretical guarantees, their practical performance is curiously under-investigated in literature before. Our experiments in Section 6 indicate a large theory-practice gap for adaptive algorithm such as [Daniely et al., 2015] when applied to deep learning framework. Our work is a first step towards closing this gap.

B Omitted Technical Details

An example motivating the necessity of including the B set in the MRI construction. We included the B set because it turned out to be important for proving Theorem 3.

Fig. 1 highlights a distribution shift scenario where usage of B set turns out to be useful. In the figure we are at the beginning of round 9 and we have seen $6n$ labelled datapoints from distribution D_2 as explained in the caption of the figure. We can see that all the active blue brackets starts only from time 9 and hence has not seen any data from distribution D_2 . But the circled brackets from set B in resolutions 1 and 2 has seen $5n$ data points from D_2 . This forms a particular scenario where the brackets from B turns out to be useful in getting a data coverage guarantee as stipulated by Theorem 1

Theorem 1. *Suppose we are at the beginning of a timestamp $t + 1$ and the data distribution has remained constant from some round $t_0 < t + 1$. Let this distribution be \mathcal{D} . We have labelled hold-out data available till round t . There exists at least one instance in the MRI pool that is active at a given round and satisfies at least one of the following properties:*

- *All the training data seen from the model is from distribution \mathcal{D} . Further it has seen at least $(t - t_0 + 1)n/2$ points from distribution \mathcal{D} .*
- *The model been only trained on data from \mathcal{D} . Further the number of points seen by the model is at least $(t - t_0 + 1)n/4$.*

Proof. Through out this proof we view intervals in $R \cup B$ and models in MRI synonymously since they are strongly associated with each other. We assume that time t_0 is not the start of any intervals in the MRI, otherwise the theorem is trivially satisfied.

Since we are the beginning of round $t + 1$, we have labelled training data available for rounds in $[t_0, t]$. Consider the smallest resolution in the MRI where there exists an interval $[a, b]$ in R that includes the duration $[t_0, t + 1]$. Let the length of such an interval be d . For brevity of notations let's define $t'_0 := t_0 - a + 1$ and $t' = t - a + 1$. Since this is the interval in the smallest resolution that covers $[t_0, t + 1]$, we must have that $t'_0 \leq d/2$.

Recall that the data distribution \mathcal{D} is constant across rounds $[t_0, t + 1]$. Note that $d > t'$. So if $t' - d/2 \geq d/2 - t' + 1$, then we can select the interval $[(a + b)/2 + 1, b] \in R$ that has seen at-least $(t - t_0 + 1)n/2$ from distribution \mathcal{D} . Since the current timepoint $t + 1 \in [a, b]$, we also have that $[(a + b)/2 + 1, b] \in \text{ACTIVE}(t + 1)$

In the case when $t'_0 \leq d/4$, the interval $[a, b]$ has seen all the training data in $[t_0, t]$. Notice that the interval $[1 + (3a + b)/4, (3a + b)/4 + d] \in B$, covers at-least $(t - t_0 + 1)n/2$ of data points from the distribution \mathcal{D} . Further this interval has seen only data from the distribution \mathcal{D} . Since the duration of this interval is d , we conclude that it is active at the current round.

Now we consider the case where $d/4 < t'_0 < d/2$ and $t' - d/2 < d/2 - t' + 1$. Consider the smallest resolution in MRI which contains a bracket $[e, f] \in R$ that fully covers the interval $[t_0, (a + b)/2]$. Clearly we must have $f = (a + b)/2$. Due to the smallest resolution property, we have that t_0 must be at-most $(e + f)/2$.

Since $t' - d/2 < d/2 - t' + 1$, we have that the number of data points seen within the interval $[(e + f)/2 + 1, f]$ must be at-least $(t - t_0 + 1)n/4$. Consequently the interval $[(e + f)/2 + 1, (5f - 3e)/2] \in B$ must also have seen

at-least $(t - t_0 + 1)n/4$ training data points from the distribution \mathcal{D} . Since $t' - d/2 < d/2 - t' + 1$, we must have $[(e + f)/2 + 1, (5f - 3e)/2] \in \text{ACTIVE}(t + 1)$. Further since \mathcal{D} has remained constant across $[t + 0, t + 1]$, we have that all the data seen by the interval $(e + f)/2 + 1, (5f - 3e)/2$ till now must be from \mathcal{D} . \square

Definition 5. Let \mathcal{F} be a function that maps \mathcal{Z} to $0, 1$. The shattering coefficient is defined as the maximum number of behaviours over n points.

$$S(\mathcal{F}, n) := \max_{z_{1:n} \in \mathcal{Z}} |\{(f(z_1), \dots, f(z_n)) : f \in \mathcal{F}\}|.$$

We say that subset $\mathcal{F}' \subseteq \mathcal{F}$ is an n -shattering-set if it is a smallest subset of \mathcal{F} such that for any $(f(z_1), \dots, f(z_n))$ there exists some $f' \in \mathcal{F}'$ such that $(f(z_1), \dots, f(z_n)) = (f'(z_1), \dots, f'(z_n))$.

Notations: We introduce some notations. Let $d = |\text{ACTIVE}(t)|$, $\text{ACTIVE}(t) := \{M_1, \dots, M_d\}$ and $\mathcal{G} := \{x \rightarrow \arg\max_{k \in [K]} (\sum_{j=1}^d w_j \text{logit}(M_j(x))[k]) : w_j \in \mathbb{R}\}$ denote a hypothesis class consisting of classifiers whose logits are weighted linear combination of that in $\text{ACTIVE}(t)$. Let the distribution of data at round t be D_t . Define $\text{Acc}_t(g) = E_{(X,Y) \sim D_t} [I\{g(X) = Y\}]$ and $\widehat{\text{Acc}}_t(g) := \text{refineAccuracy}(g, t, \delta)$ for a classifier g .

Theorem 2. Assume the notations defined in Sec. 3. Suppose we are at the beginning of round t and that the data is sampled independently across timestamps. Assume that the data distribution (say \mathcal{D}) has remained constant in $[t - r, t]$. Then with probability at least $1 - 4\delta \log T \log mT$, instantaneous regret at round t for AWE,

$$\text{Acc}_t(h_t^*) - \text{Acc}_t(\hat{f}_t) = \text{Acc}_t(h_t^*) - \text{Acc}_t(f_t^*) + \tilde{O}\left(\sqrt{1/mr}\right),$$

where \tilde{O} hides logarithmic factors in T, m, r and $1/\delta$.

Proof. We decompose the instantaneous regret at round t as

$$\text{Acc}_t(h^*) - \text{Acc}_t(\hat{f}) = \underbrace{\text{Acc}_t(h^*) - \text{Acc}_t(f^*)}_{T_1} + \underbrace{\text{Acc}_t(f^*) - \text{Acc}_t(\hat{f})}_{T_2} \quad (5)$$

We further proceed to bound T_2 as

$$\begin{aligned} T_2 &= \text{Acc}_t(f^*) - \widehat{\text{Acc}}_t(f^*) + \widehat{\text{Acc}}_t(\hat{f}) - \text{Acc}_t(\hat{f}) \\ &\quad \widehat{\text{Acc}}_t(f^*) - \widehat{\text{Acc}}_t(\hat{f}) \\ &\leq \text{Acc}_t(f^*) - \widehat{\text{Acc}}_t(f^*) + \widehat{\text{Acc}}_t(\hat{f}) - \text{Acc}_t(\hat{f}), \end{aligned} \quad (6)$$

where the last line is due to the fact that \hat{f} maximises the refined accuracy estimates among \mathcal{F} .

Next, we proceed to bound $|\widehat{\text{Acc}}_t(g) - \text{Acc}_t(g)|$ for any $g \in \mathcal{G}$. Note that $\mathcal{F} \subset \mathcal{G}$. Hence such a task will directly lead to a bound on terms of the form $|\widehat{\text{Acc}}_t(f) - \text{Acc}_t(f)|$ for any $f \in \mathcal{F}$. The reason we follow this path is because the refined accuracy of the model $E \in \mathcal{F}$ depends highly non-linearly on the past cross-validation data due to the weighted combination of models in ACTIVE_t where the weights itself are based on the corresponding models' refined accuracy estimate.

Subtle care needs to be exercised when bounding terms of the form $|\widehat{\text{Acc}}_t(g) - \text{Acc}_t(g)|$. Notice that we can write

$$\widehat{\text{Acc}}_t(g) := \frac{1}{n(g)} \sum_{i=1}^{n(g)} \frac{1}{m} \sum_{j=1}^m I\{g(x_j^{(t-i+1)}) = y_j^{(t-i+1)}\},$$

where x_v^n is the v^{th} covariate revealed at round n in Framework 1. Here $n(g)$ is the final value of r where the call to $\text{refineAccuracy}(g, t, \delta)$ stops. Since it depends on the hypothesis $g \in \mathcal{G}$ handling this is different from the usual way of

handling the excess risk in statistical learning theory [Bousquet et al., 2003] where the number of datapoints is independent of the hypothesis.

observation 1: To get around this issue, consider mt datapoints: $S = \{(x^1, y^1)_{1:m}, \dots, (x^t, y^t)_{1:m}\}$. Consider two hypothesis $g, g' \in \mathcal{G}$ such that $I\{g(x) = y\} = I\{g'(x) = y\}$ for any $(x, y) \in S$. In such a case it follows that $n(g) = n(g')$ and $\widehat{\text{Acc}}_t(g) = \widehat{\text{Acc}}_t(g')$.

Now consider the loss-class $\mathcal{R} = \{(x, y) \rightarrow I\{g(x) = y\} : g \in \mathcal{G}\}$ induced by \mathcal{G} . Let \mathcal{R}' be the mt -shattering-set (Def.5). Thus $|\mathcal{R}'| = S(\mathcal{R}, mt)$. Notice that each hypothesis in \mathcal{G} is associated with a loss-composed hypothesis in \mathcal{R}' . In view of observation 1, in order to bound the loss-composed-hypothesis random variable $|\widehat{\text{Acc}}_t(g) - \text{Acc}_t(g)|$ for any $g \in \mathcal{G}$ it suffices to take a union bound of Proposition 6 across all elements in \mathcal{R}' . This implies that

$$|\widehat{\text{Acc}}_t(g) - \text{Acc}_t(g)| = \tilde{O}(\sqrt{\log(S(\mathcal{R}, mt))/mr}), \quad (7)$$

with probability at-least $1 - \delta \cdot \log(S(\mathcal{R}, mt))$ for any $g \in \mathcal{G}$.

Each hypothesis in \mathcal{G} is a d -dimensional linear binary classifier. So VC dimension of \mathcal{G} is d . Since the VC dimension of \mathcal{R} is at-most twice that of \mathcal{G} we have that $\log(S(\mathcal{R}, mt)) = 2d$. Further since $|\text{ACTIVE}(t)| \leq 2 \log T$, combined with Sauer's lemma [Mohri et al., 2012] we have that $\delta \cdot \log(S(\mathcal{R}, mt)) \leq 4\delta \log T \log mT$.

Now putting everything together results in Theorem 2. □

Proposition 6. (due to Theorem 1 in [Mazzetto and Upfal, 2023]) For a fixed model $g \in \mathcal{G}$ we have that with probability at-least $1 - \delta$

$$|\text{Acc}_t(g) - \widehat{\text{Acc}}_t(g)| = \tilde{O}\left(\sqrt{\frac{1}{mr}}\right).$$

Proof. The proof is a direct consequence of the result in [Mazzetto and Upfal, 2023]. For the sake of completeness, we show how it follows from Theorem 1 in [Mazzetto and Upfal, 2023].

Let $(\mathcal{X} \times \mathcal{Y}, \mathcal{A})$ be a measurable space. By the assumption in Theorem 2, $(X_i, Y_i) \sim D_i$ are mutually independent random variables. Let $D_t^r := \frac{1}{r} \sum_{\tau=t-r+1}^t D_\tau(A)$ for all $A \in \mathcal{A}$. Let \hat{D}_t^r be the corresponding empirical distribution defined by

$$\hat{D}_t^r := \frac{|(X_\tau, Y_\tau) \in A : t-r+1 \leq \tau \leq t|}{mr},$$

where we recall that there are m hold-out data-points revealed after each round.

For a fixed model $g \in \mathcal{G}$ consider the singleton function class $\mathcal{F} := \{(x, y) \mapsto I\{g(x) = y\}\}$. For any distributions P, Q on $(\mathcal{X} \times \mathcal{Y}, \mathcal{A})$ define:

$$P(g) := E_{(X,Y) \sim P}[I\{g(X) = Y\}],$$

and

$$\|P - Q\|_{\mathcal{F}} := |P(g) - Q(g)|.$$

Due to Hoeffding's lemma, we have that for any fixed $r \leq t$ and $\delta \in (0, 1)$,

$$\|D_t^r - \hat{D}_t^r\|_{\mathcal{F}} = O(\sqrt{\log(1/\delta)/(mr)}),$$

with probability at-least $1 - \delta$. Hence Assumption 1 in [Mazzetto and Upfal, 2023] holds.

Let \hat{r} be the final value where the $\text{refineAccuracy}(g, t, \delta)$ procedure (Alg.3) stops. Now Theorem 1 in [Mazzetto and Upfal, 2023] states that the output of $\text{refineAccuracy}(g, t, \delta)$ satisfies with probability $1 - \delta$ that

$$\|D_t - \hat{D}_t\|_{\mathcal{F}} = \tilde{O}\left(\min_{u \leq t} \left[\frac{1}{\sqrt{um}} + \max_{\tau < u} \|D_t - D_{t-\tau}\|_{\mathcal{F}} \right]\right). \quad (8)$$

Using the fact that the data distribution has remained constant in $[t - r, t]$ as in the assumption of Theorem 2 and upper bounding the minimum across u by plugging in $u = r$, we get that

$$|\text{Acc}_t(g) - \widehat{\text{Acc}}_t(g)| = \tilde{O}(1/\sqrt{mr}).$$

This completes the proof. □

In Section 5, our analysis focused on the case where the distribution shifts were assumed to be piece-wise stationary. Next, we relax that assumption and study the dynamic regret of (see Eq.(2)) AWE under the cases of slowly evolving shifts.

Theorem 7. Assume the notations used in Section 5 and Theorem 2. Consider an arbitrary partitioning \mathcal{P} of the time horizon into M bins as $[i_s, i_t]$ for $i = 1, \dots, M$. Define $V_{i_s:i_t} := \max_{u \in [i_s, i_t]} TV(D_{i_t}, D_u)$ which is the maximum total variation (from end time-point of the bin) of the data distribution within the i^{th} bin. Define $V := \sum_{i=1}^M V_{i_s:i_t+1}$. We have the following dynamic regret bound for AWE:

$$\begin{aligned} R_{\text{dynamic}} &= \sum_{k=1}^T \text{Acc}_k(h_k^*) - \text{Acc}_k(\hat{f}_k) \\ &= \sum_{k=1}^T \text{Acc}_k(h_k^*) - \text{Acc}_k(f_k^*) + \min_M \min_{\mathcal{P}} \sum_{i=1}^M \tilde{O}(\sqrt{dn_i/m} + n_i V_{i_s:i_t}), \end{aligned}$$

with probability at-least $1 - \delta$.

Proof. We continue our arguments from Eq.(8). Let $j \in [i_s, i_t]$. For any fixed hypothesis $g \in \mathcal{G}$ (see Notations section before the proof of Theorem 2 for the definition of \mathcal{G}) we get with probability at-least $1 - \delta$ that

$$|\text{Acc}_j(g) - \widehat{\text{Acc}}_j(g)| = \tilde{O}(1/\sqrt{mj} + V_{i_s:i_t}).$$

By taking a union bound over the hypothesis class \mathcal{G} similar to Eq.(7) and noting that the metric entropy of \mathcal{G} is $O(d)$ we conclude that

$$|\text{Acc}_j(g) - \widehat{\text{Acc}}_j(g)| = \tilde{O}(\sqrt{d/mj} + V_{i_s:i_t}).$$

Define $n_i := i_t - i_s + 1$ which is the length of the i^{th} bin. Next by taking a union bound across all time points within a bin (and after re-adjusting δ) and continuing from the decomposition in Eq.(6), we have with probability at-least $1 - \delta$ that

$$\begin{aligned} \sum_{j=i_s}^{i_t} \text{Acc}_j(f_j^*) - \text{Acc}_j(\hat{f}_j) &= \sum_{j=i_s}^{i_t} \tilde{O}(\sqrt{d/mj} + V_{i_s:i_t}) \\ &= \tilde{O}(\sqrt{dn_i/m} + n_i V_{i_s:i_t}) \end{aligned}$$

Now summing across all bins, using the decomposition in Eq.(5) and noting that the partition was selected arbitrarily results in the theorem. □

To better understand how the dynamic regret relates to the variation in the evolving data distributions, we provide the following corollary to Theorem 7.

Corollary 8. *Assume the notations of Theorem 7. We have with probability at-least $1 - \delta$ that AWE satisfies*

$$R_{\text{dynamic}} = \tilde{O}(T^{2/3}(V/m)^{1/3} + \sqrt{T/m}) + \sum_{k=1}^T \text{Acc}_k(h_k^*) - \text{Acc}_k(f_k^*)$$

Remark 9. *The expression in the RHS of the bound in Corollary 8 is composed of two terms. The first term characterizes the dynamic regret. The second term characterises the approximation error of the instance pool $\{E_t\} \cup \mathcal{A}_t$ (see AWE Alg. 2 for definitions of E_t and \mathcal{A}_t) in approximating the best hypothesis $h_t^* \in \mathcal{H}$. Since E_t is an ensemble model which can be more expressive than individual instances of the OL algorithm trained on disjoint pieces of the history, the second term can be potentially negative as well.*

Proof. Note that Theorem 7 holds for any partitioning schemes. Hence to further upperbound the dynamic regret, we can compute the bound in Theorem 7 for any specific partitioning scheme.

Consider the following partitioning of the time horizon into M bins such that $V_{i_s:i_t} \leq \epsilon$ while $V_{i_s:i_t+1} > \epsilon$. Suppose that $M > 1$. We have the following bound on number of bins

$$\begin{aligned} V &= \sum_{i=1}^M V_{i_s:i_t+1} \\ &\geq \epsilon M. \end{aligned}$$

The number of bins also must be at-least 1. Hence the number of bins obeys $M \leq 1 + V/\epsilon$. Instantiating Theorem 7 with the above partitioning results in

$$\begin{aligned} R_{\text{dynamic}} &= \sum_{k=1}^T \text{Acc}_k(h_k^*) - \text{Acc}_k(\hat{g}_k) \\ &\leq \sum_{k=1}^T \text{Acc}_k(h_k^*) - \text{Acc}_k(g_k^*) + \sum_{i=1}^M \tilde{O}(\sqrt{dn_i/m} + n_i V_{i_s:i_t}) \\ &\leq_{(a)} \sum_{k=1}^T \text{Acc}_k(h_k^*) - \text{Acc}_k(g_k^*) + \tilde{O}(\sqrt{dMT/m}) + T\epsilon \\ &\leq \sum_{k=1}^T \text{Acc}_k(h_k^*) - \text{Acc}_k(g_k^*) + \tilde{O}(\sqrt{(dT/m)(1 + V/\epsilon)}) + T\epsilon \\ &\leq \sum_{k=1}^T \text{Acc}_k(h_k^*) - \text{Acc}_k(g_k^*) + \tilde{O}(\sqrt{dT/m}) + \tilde{O}(\sqrt{VT/(m\epsilon)}), \end{aligned}$$

where line (a) is due to Cauchy Schwartz. Optimizing over ϵ and setting $\epsilon = (V/(mT))^{1/3}$ yields the theorem. □

Next, we compare our bound with existing dynamic regret bounds in the literature.

Remark 10. *We note that our bound is different and not directly comparable to a dynamic regret bound of the form $O(\sqrt{T(P+1)})$ where $P = \sum_{t=2}^T \text{TV}(D_t, D_{t-1})$ [Zhang et al., 2018a]. The reasons are as follows: 1) In the batched online setting of Fig. 1, the value of m (the number of points in the hold-out data set at each round) can be substantially larger than 1 as is the case with our experiments. 2) Though the variational V can be bounded above by P , such a bound can be very loose. So the value V can be much smaller than P . Hence in the batched online framework of Fig. 1, the bound given by Corollary 8 can be tighter than a regret bound of the form $O(\sqrt{T(P+1)})$.*

C Comparison to [Daniely et al., 2015]

In this section, we provide a close comparison between our work and that of [Daniely et al., 2015] both of which are black-box adaptation techniques.

C.1 Failure mode of Geometric Covering (GC) intervals

GC intervals developed in [Daniely et al., 2015] fails to satisfy a data coverage guarantee as stipulated by Theorem 1. We describe a minimalistic scenario where GC is insufficient to give a data coverage guaranteed by MRI as in Theorem 3.

Let $T = 10$. The GC intervals that spans this time horizon can be split into various resolution as follows.

$$Res_0 = [1, 1], [2, 2], \dots, [10, 10] \quad Res_1 = [2, 3], [4, 5], [6, 7], [8, 9], [10, 11] \quad Res_2 = [4, 7], [8, 11] \quad Res_3 = [8, 15]$$

Suppose The distribution shift is such that data at time 1 is generated from distribution D_1 and times $[2, 10]$ are generated from another distribution D_2 . Let the number of labelled examples revealed after each online round be N . Suppose we are the beginning of the online round $t = 9$. So we have seen $7N$ labelled data points from distribution D_2 .

We have $\text{ACTIVE}(9) = [9, 9], [8, 9], [8, 11], [8, 15]$. Since all the active intervals start from timepoint 8, the experts defined by these active intervals have only seen N labelled data points from distribution D_2 when we are at round 9. Since $N < 7N/4 < 7N/2$, it fails to provide a data coverage guarantee as stipulated by MRI in Theorem 3. On the other hand, in this example there exists a bracket in the MRI from the B set that can cover $4N$ data points from distribution D_2 . The under-coverage effect of GC can be more exaggerated when we consider longer time horizons.

C.2 Differences in problem setting and regret guarantees

We note that the setting considered in our paper differs from that of the usual black-box model selection in the adaptive online learning literature [Daniely et al., 2015]. We consider the setting of batched online learning (Framework 1) where the learner makes predictions for the labels of a collection of covariates (say N covariates) that are revealed at each online round with true labels revealed only after all the predictive labels are submitted. This is suited for real-world usecases where it is practical to receive a collective feedback. The main message is that by using our methods, one can attain faster average interval regret guarantees than that promised by SAOL [Daniely et al., 2015]. We explain this in detail below.

Conventional online algorithms operate in the regime of $N = 1$. However, if we define the loss suffered at a round as the average loss incurred by the algorithm for all covariates, then we can reduce the setting of Framework 1 to that studied in [Daniely et al., 2015] and use the black-box model selection algorithms studied there. But this approach can lead to serious drawbacks when applied to our setting of batched online learning: Suppose in an interval $I \subseteq [T]$, the data distribution is constant. Then the Strongly Adaptive guarantees in [Daniely et al., 2015] will lead to an average regret wrt the best model within interval I as $O(1/\sqrt{|I|} + 1/\sqrt{N}) = O(1/\sqrt{|I|})$ (where for later inequality, we suppose $N > |I|$). Here regret is measured wrt population level accuracy as in Theorem 2 and the term $1/\sqrt{N}$ is the artifact of concentration inequality to relate the empirical average loss at a round to its population analogue. However, by summing up the regret bound in Theorem 2, our model selection scheme lead to an average regret of $O(1/\sqrt{p \cdot N|I|})$ where p is the validation-train split ratio in AWE. Since p is selected such that $pN > 1$, such a rate leads to faster convergence than $O(1/\sqrt{|I|})$. As explained in notes on technical novelties in Section 1, this improved effect is attained by finding a weighted combination of active models with more weights allotted to models with best validation scores for the most recent distribution. Hence the distribution of the weights in our algorithm is more tilted (in comparison to that of SAOL in [Daniely et al., 2015]) towards models with high recent validation score. The CVTT component helps to obtain high accuracy validation scores for each model.

D Additional Experimental Results on AWE

Experiments with AWE. We report the accuracy differences across various time stamps for different base OL algorithms. The experimental setting is same as that of the one described in Sec. 6. The results are displayed in Figures 4, 5, 6, 7 and Table 3. The trends noticed in Sec. 6 remain to hold uniformly.

Experiments with SAOL. The experimental results on per-step accuracy gain with SAOL black-box scheme from [Daniely et al., 2015] is shown in Figures 13, 14, 15, 16 and Table 4.

Experiments with Voting. Besides Eq.(4) majority voting is a commonly used model combination scheme in stationary problems mainly due to its computational and statistical efficiencies. We run experiments where instead of using the

Input OL	FMOW (SAOL)	Huffpost (SAOL)	Arxiv (SAOL)
	$\Delta\text{acc } \%$	$\Delta\text{acc } \%$	$\Delta\text{acc } \%$
SI	-2.47 ± 0.082	-0.17 ± 0.022	-2.95 ± 0.093
FT	-2 ± 0.077	0.03 ± 0.014	-3.04 ± 0.024
IRM	-4.06 ± 0.108	0.18 ± 0.034	-2.34 ± 0.083
EWC	2.05 ± 0.078	0.06 ± 0.013	-2.95 ± 0.093
CORAL	-0.40 ± 0.034	-0.27 ± 0.028	-0.48 ± 0.038

Table 2: Performance statistics for image (FMOW dataset) and text (Huffpost & Arxiv datasets) modalities. We report the difference in average classification accuracy (%) across all timestamps obtained by the majority voting-based black-box scheme minus that of the input OL. As discussed before, the majority voting-based variant of AWE can often degrade the performance, signifying the advantage of model selection via refined accuracy estimates.

	FMOW	Huffpost	Arxiv
SI	13/0/2	5/0/1	13/0/2
FT	14/0/1	5/0/1	12/0/3
IRM	9/1/5	6/0/0	8/0/7
EWC	12/2/1	4/0/2	12/0/3
CORAL	15/0/0	6/0/0	15/0/0

Table 3: The table summarizes win/draw/lose statistics for the AWE algorithm. We say that a win (draw/lose) happens at a time stamp if the accuracy of AWE is higher (equal/lower) than the input OL algorithm. We can see that in most cases, the fraction of timestamps where AWE does not degrade the accuracy of the base OL is well above 50%. This signifies the efficacy of AWE in optimizing the instantaneous regret at each round.

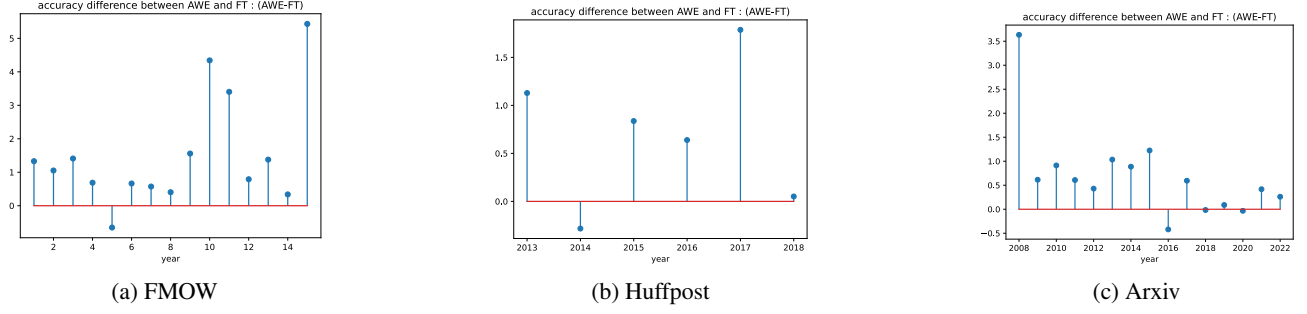


Figure 4: % accuracy differences across various timestamps when AWE is run with FT as the online learning algorithm.

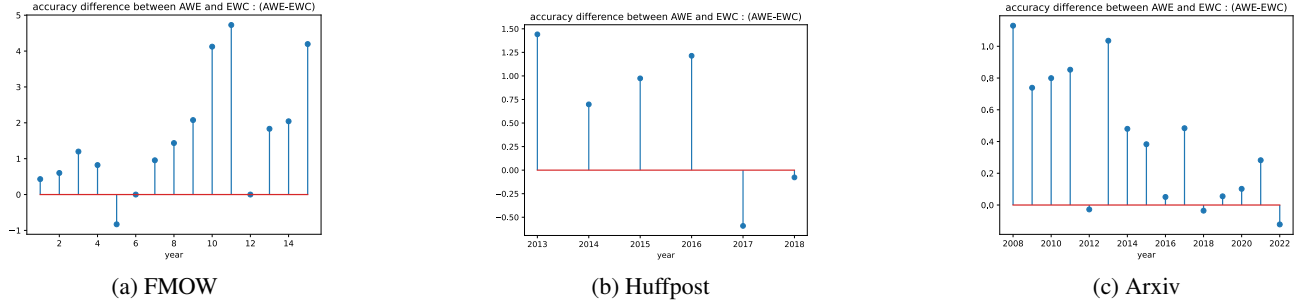


Figure 5: % accuracy differences across various timestamps when AWE is run with EWC as the online learning algorithm.

map given by Eq.(4) in AWE, at any round we output a prediction that is recommended by the majority of instances in $\text{ACTIVE}(t)$. The experimental results are reported in Tables 2 and 4 and Figures 8-12. As the experimental results show, a map that does not take into account the refined accuracy estimation can often lead to performance degradation. This provides evidence on the efficacy of more nuanced aggregation methods that also take into account the accuracy of the instances as in AWE. We remind the readers that Theorem 1 guarantees existence of at-least one model in the instance pool that has seen sufficient amount of data from the most recent distribution. However, the majority of instances can still have bad accuracy. Consequently, a majority voting strategy based model combination leads to poor performance.

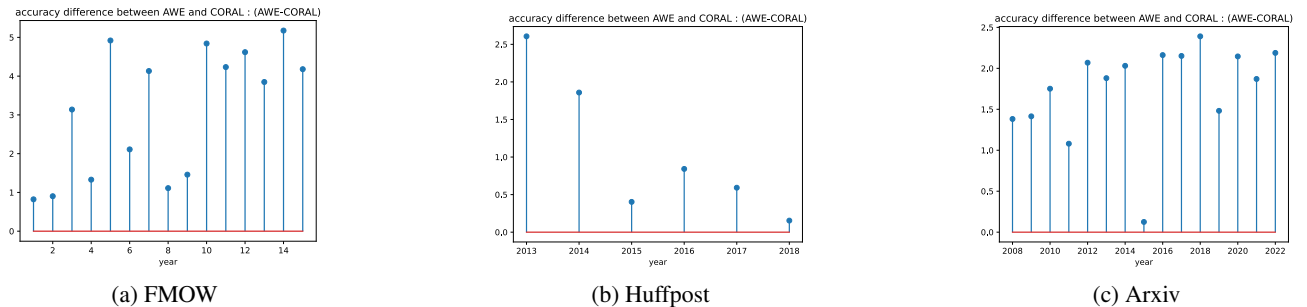


Figure 6: % accuracy differences across various timestamps when AWE is run with CORAL as the online learning algorithm.

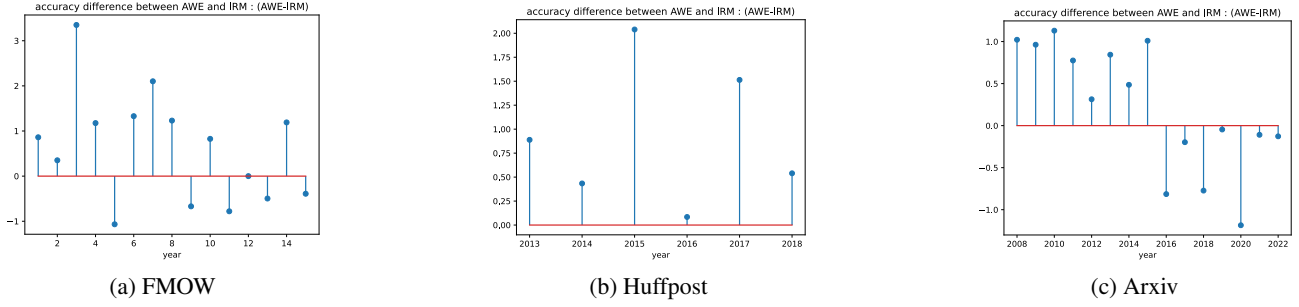


Figure 7: % accuracy differences across various timestamps when AWE is run with IRM as the online learning algorithm.

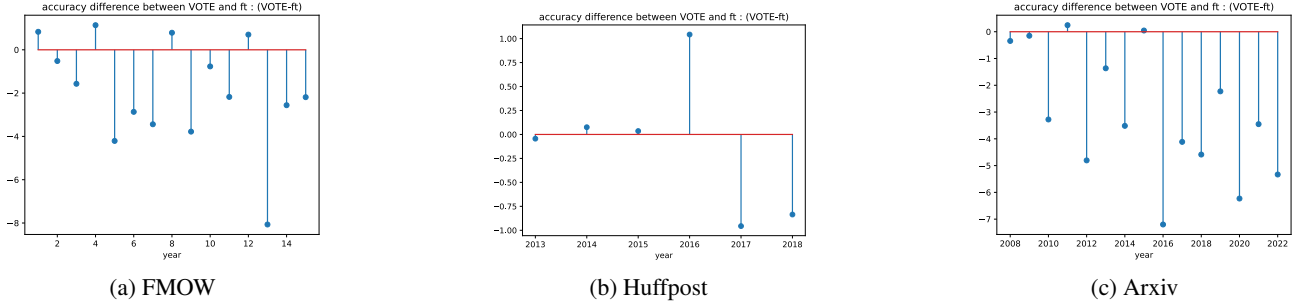


Figure 8: % accuracy differences across various timestamps when AWE with majority voting is run with FT as the online learning algorithm.

	Voting			SAOL		
	FMOW	Huffpost	Arxiv	FMOW	Huffpost	Arxiv
SI	3/0/12	3/0/3	2/0/13	1/8/6	0/3/3	1/8/6
FT	4/0/11	3/0/3	2/0/13	2/8/5	0/3/3	1/8/6
IRM	4/0/11	4/0/2	3/0/12	2/8/5	2/3/1	1/8/6
EWC	3/0/12	3/0/3	1/0/14	2/8/5	0/3/3	1/8/6
CORAL	7/0/8	2/0/4	4/0/11	1/8/6	2/3/1	2/8/5

Table 4: Table summarizing the win/draw/lose numbers for majority voting and SAOL algorithm from [Daniely et al., 2015]. We see that the number of timestamps where the black-box scheme improves the performance of the base OL is significantly lower than that of Table 3. For the case of Voting, this signifies that a weighting strategy that does not take into account the refined accuracy estimates may not be useful in practice. For the case of SAOL, this signifies that optimizing the cumulative regret (at the rate of $1/\sqrt{|I|}$, where I is the interval size for any window I) instead of instantaneous regret does not lead to significant gains in practice. See also section (a) of the notes of technical novelties.

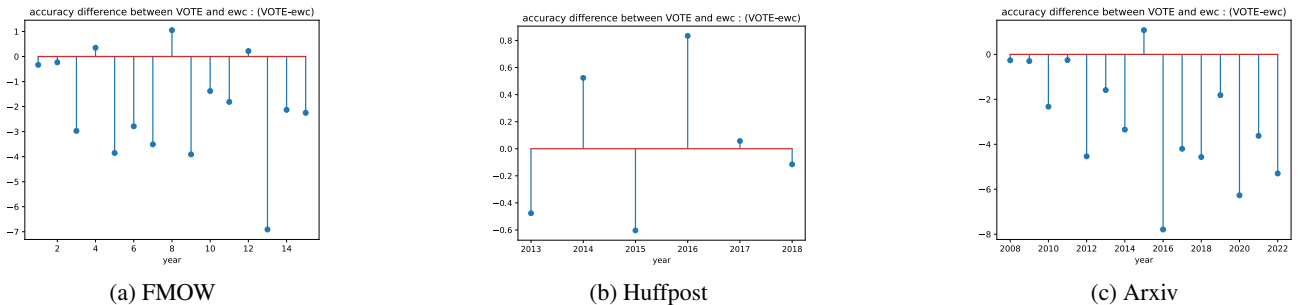


Figure 9: % accuracy differences across various timestamps when AWE with majority voting is run with EWC as the online learning algorithm.

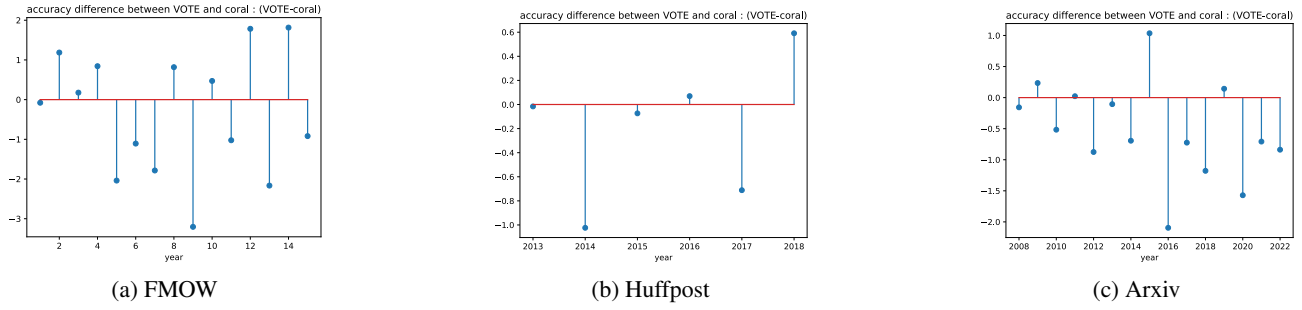


Figure 10: % accuracy differences across various timestamps when AWE with majority voting is run with CORAL as the online learning algorithm.

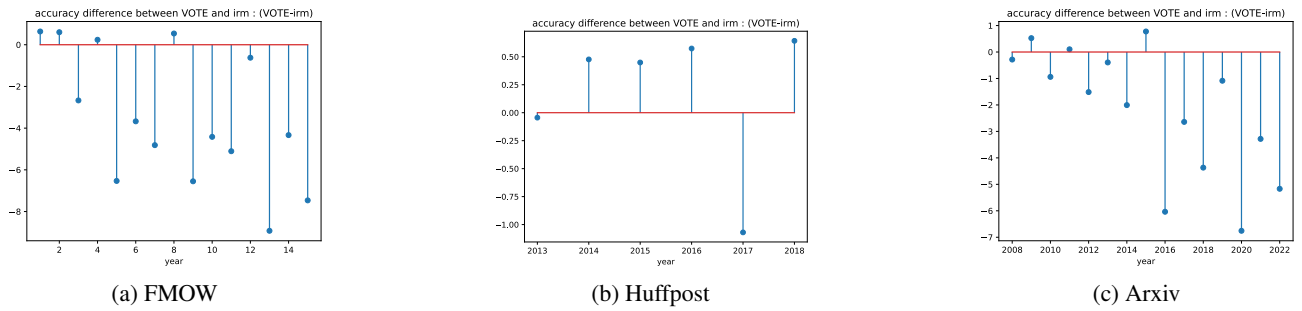


Figure 11: % accuracy differences across various timestamps when AWE with majority voting is run with IRM as the online learning algorithm.

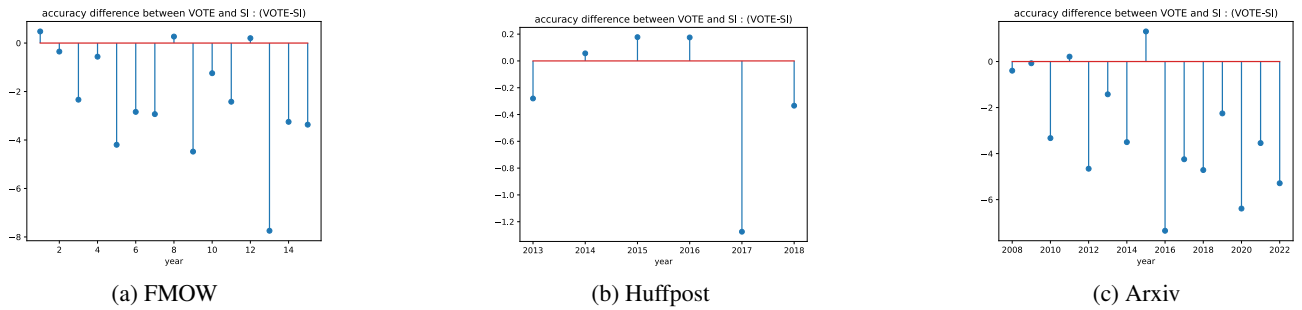


Figure 12: % accuracy differences across various timestamps when AWE with majority voting is run with SI as the online learning algorithm.

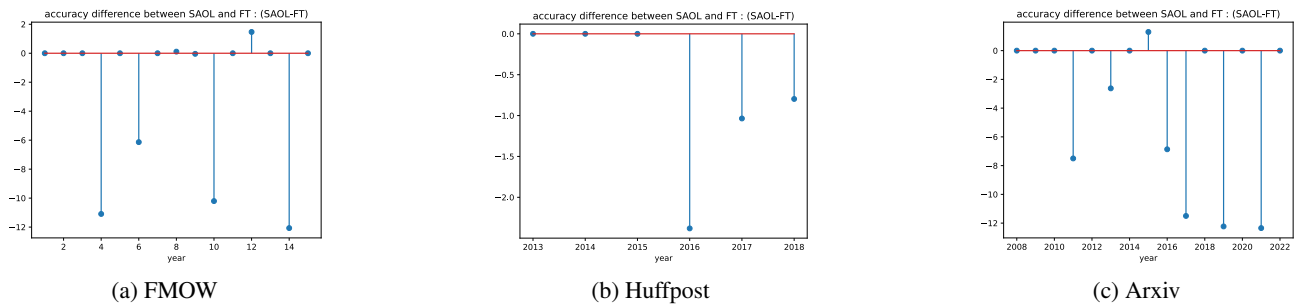


Figure 13: % accuracy differences across various timestamps when SAOL is run with FT as the online learning algorithm.

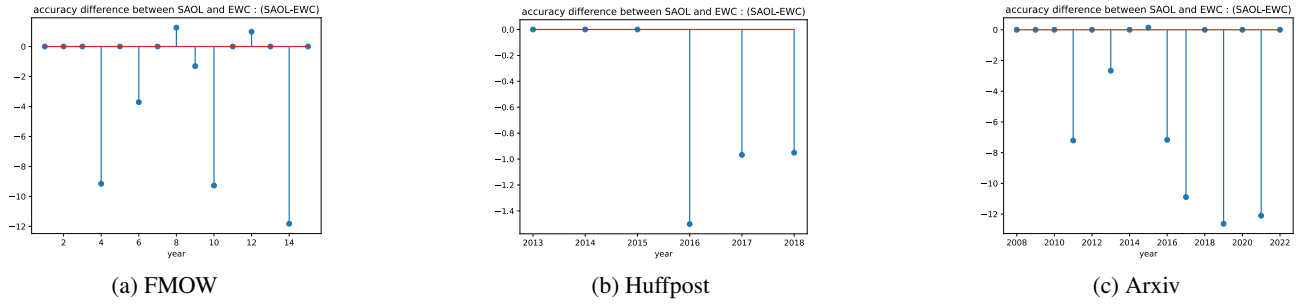


Figure 14: % accuracy differences across various timestamps when SAOL is run with EWC as the online learning algorithm.

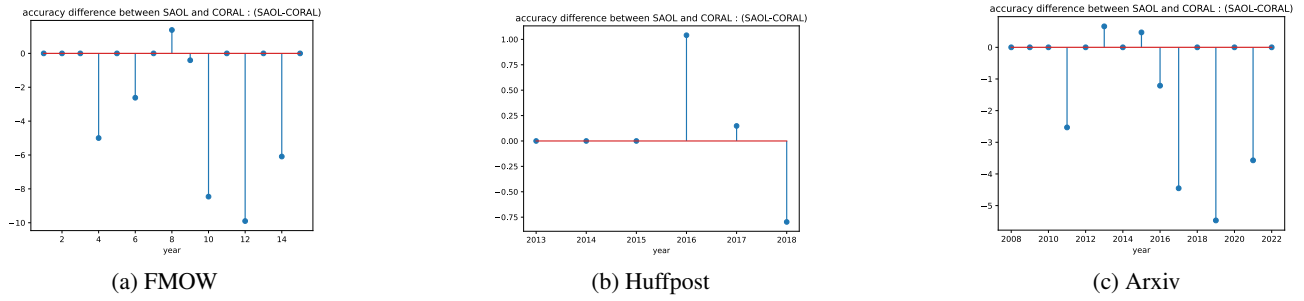


Figure 15: % accuracy differences across various timestamps when SAOL is run with CORAL as the online learning algorithm.

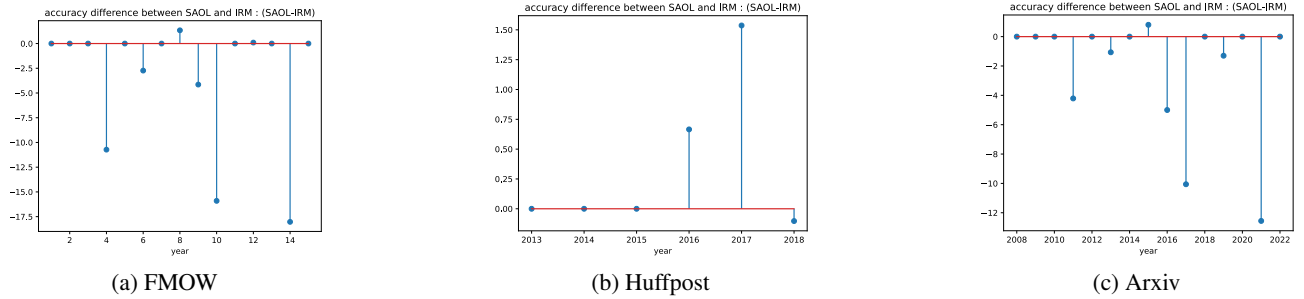


Figure 16: % accuracy differences across various timestamps when SAOL is run with IRM as the online learning algorithm.

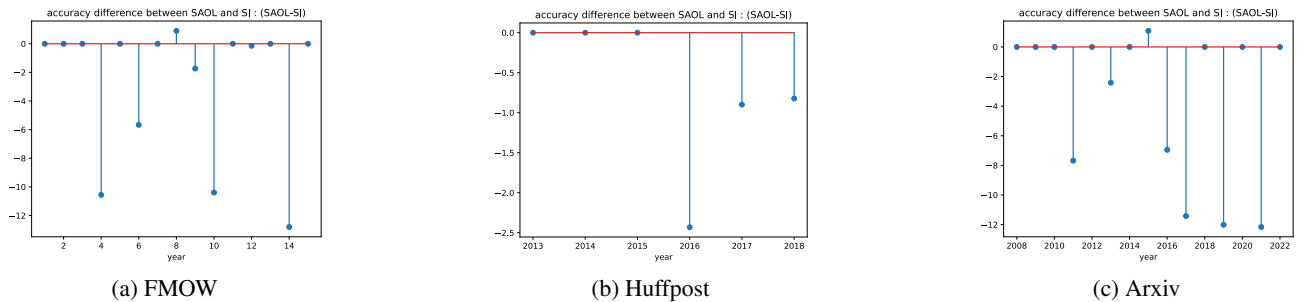


Figure 17: % accuracy differences across various timestamps when SAOL is run with SI as the online learning algorithm.