Regret–Oracle Complexity Tradeoffs in Agnostic Online Learning

Idan Attias * **Steve Hanneke**† **Arvind Ramaswami**‡

Abstract

Agnostic online learning is classically solved via a reduction to the realizable setting, utilizing Littlestone's Standard Optimal Algorithm (SOA) as a base learner. However, the SOA can be computationally intractable to implement even for a single round. To overcome this barrier, recent work in oracle-efficient online learning replaces the SOA with a realizable base learner that accesses the concept class exclusively through an offline empirical risk minimization (ERM) oracle. While such agnostic learners achieve expected regret $\tilde{\mathcal{O}}\big(\sqrt{T \cdot 2^{\mathcal{O}(d_{\mathrm{LD}})}}\big)$, which is near-optimal in the horizon T for fixed Littlestone dimension d_{LD}, they require $\mathcal{O}\big(T^{2^{\mathcal{O}(d_{\mathrm{LD}})}}\big)$ oracle calls. In this work, we significantly improve this oracle complexity while relying on an even weaker primitive: a weak-consistency oracle, which merely decides whether a given labeled dataset is realizable. At the core of our approach is an adaptive and dynamic agnostic-to-realizable reduction that actively prunes non-realizable label sequences on the fly. By using the VC dimension (d_{VC}) to bound the number of dynamically maintained active paths, our algorithm reduces the total query complexity down to $\mathcal{O}(T^{d_{\mathrm{VC}}+1})$ while perfectly preserving near-optimal expected regret. Crucially, this dynamic pruning also yields a memory reduction over the standard reduction. Furthermore, we formally quantify the regret–oracle complexity tradeoff, providing upper bounds that smoothly interpolate between restricted query budgets and attainable expected regret. We complement these with lower bounds proving that any learner restricted to $Q = o(\sqrt{T})$ queries must suffer an expected regret of $\Omega(T/Q)$.

1 Introduction

Online learning is a fundamental framework for sequential prediction. On each round, the learner observes an instance, predicts a label, and then observes the true label, incurring a loss. The primary goal is to minimize regret: the learner's excess cumulative loss relative to the best concept in the class in hindsight [CBL06, SS+12, MRT12, SSBD14]. This framework captures a broad range of learning problems and has become a central tool in optimization, game theory, and sequential decision-making.

A growing line of work investigates whether online learning can be implemented efficiently using only offline optimization primitives. Offline-oracle models have successfully yielded computationally efficient algorithms across a broad range of decision-making problems [KV05, DHK+11, AHK+14, SKS16, HK16, FR20, DHL+20, SLX22, HHSY22, HRS24, BRS24]. In such models, the learner does not manipulate the concept class directly; instead, it interacts with the class through an oracle, typically an empirical risk minimization (ERM) oracle. This perspective is particularly natural, as ERM is the foundational optimization primitive underlying much of statistical learning[4].

*Institute for Data, Econometrics, Algorithms, and Learning (IDEAL), hosted by UIC and TTIC; idanattias88@gmail.com.

†Department of Computer Science, Purdue University; steve.hanneke@gmail.com.

‡Department of Computer Science, Purdue University; ramaswa4@purdue.edu.

[4]For a comprehensive literature review, see Section A.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

We focus on standard binary online classification with the 0-1 loss over a general, possibly infinite, concept class $\mathcal{C}$. In the realizable setting, Littlestone's Standard Optimal Algorithm (SOA) perfectly characterizes online learnability via the Littlestone dimension, d_{LD} [Lit88]. In the agnostic setting, where the sequence of labels may be arbitrary, Ben-David et al. [BDPSS09] established a regret bound of $\mathcal{O}(\sqrt{Td_{\mathrm{LD}}\log T})$ by reducing the agnostic problem to the realizable one using the SOA as a subroutine, where T is the number of rounds. Alon et al. [ABED+21] later proved a tight optimal regret bound of $\mathcal{O}(\sqrt{Td_{\mathrm{LD}}})$, highlighting the fact that the Littlestone dimension definitively characterizes agnostic online learnability. More finely, [FHMM23] show that in the k-realizable setting, the optimal expected regret is $\Theta(\sqrt{kd_{\mathrm{LD}}} + d_{\mathrm{LD}})$, which can be substantially smaller than the worst-case rate when $k \ll T$.

While these algorithms achieve optimal regret, they are computationally intractable in general. Executing the SOA requires repeatedly computing the Littlestone dimension of restricted subclasses, a task known to be intractable to approximate within a constant factor even when the concept class and the set of features are finite [FL98, MR17, Man22]. Moreover, Hasrati and Ben-David [HBD23] demonstrated that there exist recursively enumerable representable classes with a finite Littlestone dimension for which no computable SOA exists at all. This computational barrier motivates the oracle-efficient framework formalized by Assos et al. [AAD+23] and Kozachinskiy and Steifer [KS24]. In this model, the online learner accesses the concept class exclusively through an offline ERM oracle: given a labeled dataset, the oracle returns a concept minimizing the empirical error.

The central question is therefore:

What regret is achievable in agnostic online learning when the learner is restricted to offline oracle access, and how many oracle calls are necessary?

In the realizable setting, an exponential mistake bound in d_{LD} is attainable using ERM access, and such an exponential dependence is unavoidable for ERM oracles [AAD+23, AHR25, KS24]. Translating this to the agnostic setting, prior reductions achieve regret $\tilde{\mathcal{O}}\big(\sqrt{T \cdot 2^{\mathcal{O}(d_{\mathrm{LD}})}}\big)$, but suffer from a staggering oracle query complexity of $\mathcal{O}\big(T^{2^{\mathcal{O}(d_{\mathrm{LD}})}}\big)$. While originally analyzed with a full ERM oracle, the core mechanisms of these reductions can be adapted to use a *weak consistency oracle*[5] [AHR25]. This weaker primitive merely decides whether a labeled sample is realizable (i.e., it solves a decision problem rather than an optimization one), and has recently been shown to suffice for efficient PAC learning without requiring the full power of ERM [DG24].

Our work demonstrates that this doubly-exponential oracle complexity is not inherent. The key structural observation is that standard agnostic-to-realizable reduction obliviously maintains a multitude of pseudo-label histories that are strictly impossible for the underlying class $\mathcal{C}$ to produce. To overcome this, we introduce an adaptive and dynamic reduction that continuously issues weak-consistency queries to prune unrealizable histories the exact moment they arise. By Sauer's Lemma, the number of realizable labelings of the first t observed instances is strictly bounded by $\mathcal{O}(t^{d_{\mathrm{VC}}})$, where d_{VC} is the VC dimension. This shows that while the Littlestone dimension dictates the *information-theoretic price* (regret) of online classification, it does not dictate the *computational cost* (in terms of oracle complexity), which can be bounded by an arbitrarily smaller quantity: the VC dimension.

Our Contributions. We summarize our main contributions in Figure 1, mapping the Pareto frontier between the total number of weak-consistency oracle (Definition 2.1) queries and the attainable expected regret, compared to previously known results. Our technical contributions are as follows:

1. **A general agnostic-to-realizable reduction via dynamic pruning (Section 4).** We design a novel reduction (Algorithm 2) that seamlessly wraps any deterministic realizable base learner having a mistake bound M. Instead of suffering the computational blowup caused by explicitly maintaining all possible mistake schedules $\mathcal{O}(T^M)$ [BDPSS09], our dynamic-pruning mechanism reduces the active state space, maintaining at most $\mathcal{O}(t^{d_{\mathrm{VC}}})$ active paths per round t. This structural improvement yields three immediate consequences:
   - *Total Oracle Efficiency:* When instantiated with existing oracle-efficient base learners ($M = 2^{\mathcal{O}(d_{\mathrm{LD}})}$), our reduction matches their expected regret of $\tilde{\mathcal{O}}\big(\sqrt{T \cdot 2^{\mathcal{O}(d_{\mathrm{LD}})}}\big)$. Crucially, it

[5] In [AHR25], the algorithm relying on a weak consistency oracle is proven for the realizable case but can be adapted to the agnostic setting as well.

improves the total query complexity from $\mathcal{O}\big(T^{2^{\mathcal{O}(d_{\mathrm{LD}})}}\big)$ required by prior work down to just $\mathcal{O}(T^{d_{\mathrm{VC}}+1})$. Since $d_{\mathrm{VC}} \le d_{\mathrm{LD}}$ (and can be arbitrarily smaller), this constitutes an exponential to double-exponential reduction in dimension dependence.

- *Improved Memory Bound:* Even independently of oracle restrictions, when wrapping the Standard Optimal Algorithm ($M = d_{\mathrm{LD}}$), our reduction achieves the minimax optimal regret of $\tilde{\mathcal{O}}(\sqrt{T d_{\mathrm{LD}}})$. While the standard reduction must explicitly track a tree of $\mathcal{O}(T^{d_{\mathrm{LD}}})$ experts, our active pruning compresses the memory by maintaining only $\mathcal{O}(t^{d_{\mathrm{VC}}})$ paths per round t.
- *Instance-Dependent Bound:* By adaptively tuning the learning rate based on the empirically best active path, we achieve a first-order expected regret bound of $\mathcal{O}\big(\sqrt{L^* M \log T} + M \log T\big)$, where L^* is the true optimal loss in hindsight. This interpolates between the fully adversarial regime and the realizable setting.

2. **The regret–query tradeoff (Section 4.1).** We formally map the tradeoff between the available oracle query budget and the attainable expected regret. We prove that for any interpolation parameter $c \in (0, 1)$, there exists an online learner requiring at most $\mathcal{O}(T^{1+cd_{\mathrm{VC}}})$ total oracle queries while guaranteeing an expected regret bounded by $\mathcal{O}\big(T^{1-c/2}\sqrt{d_{\mathrm{LD}} + M \log T}\big)$. This smoothly interpolates between the low-query regime, yielding larger but still strictly sublinear regret, and the high-query regime, where our main reduction achieves near-optimal regret.
3. **A query-budget lower bound (Section 5).** We prove that a heavy dependence on the oracle budget is information-theoretically unavoidable in the low-query regime. Even for elementary concept classes with Littlestone dimension $d_{\mathrm{LD}} = 1$, any learner restricted to $Q = o(\sqrt{T})$ queries to an ERM oracle must suffer an expected regret of $\Omega(T/Q)$. This strictly improves upon prior lower bounds [AHR25] in the constrained $o(\sqrt{T})$ query regime, formally showing that any constant-query learner must suffer linear regret (as opposed to $\sqrt{T}$ in previous work).
4. **k-Realizable Online Learning with Oracles (Section F).** We give an oracle-efficient reduction from k-realizable online learning, where some concept in the class makes at most k mistakes on the sequence, to realizable online learning. This yields a deterministic learner with mistake bound $2^{\mathcal{O}(L_k)}$, where $L_k = \mathcal{O}\left(d_{\mathrm{LD}} + k\right)$ is the k-Littlestone dimension [FHMM23]. The learner uses $\mathcal{O}\left(T^{k+1}2^{\mathcal{O}(L_k)}\right)$ weak-consistency oracle calls, or $2^{\mathcal{O}(L_k)}$ ERM oracle calls. This complements our instance-dependent bound (Corollary 4.4) by providing a mistake guarantee independent of the horizon T, albeit with a worse dependence on k.

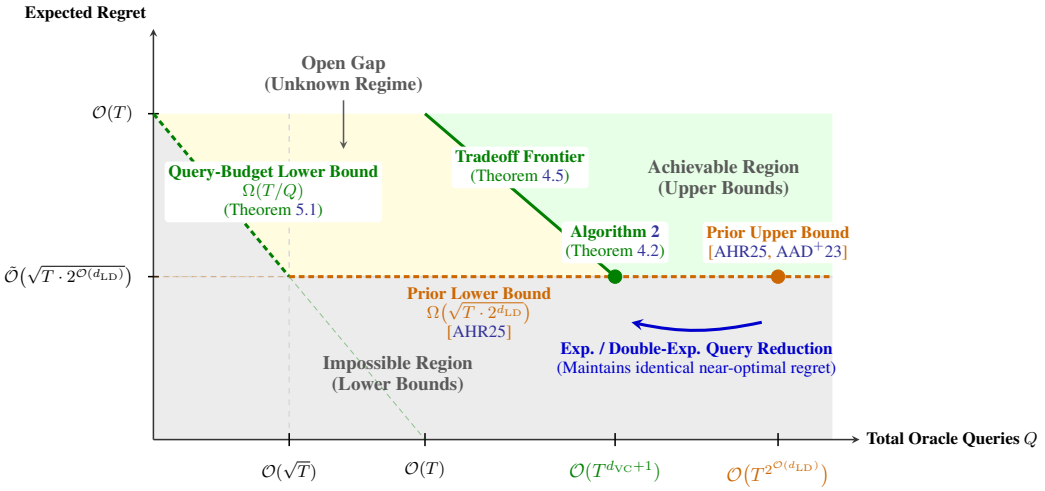

Figure 1: **Expected regret vs. total oracle queries tradeoff**. Our results are marked in green, whereas previous results appear in orange. The yellow region highlights the gap between the known upper and lower bounds. We believe that closing this gap will require fundamentally new ideas, potentially including oracle-efficient algorithms for agnostic online learning that go beyond the agnostic-to-realizable reduction, stronger query-budget lower bounds, or both.

2 The Learning Model: Agnostic Online Learning with Offline Oracle Access

We start with a definition of the online learning model where the interaction of the learner with the concept class is done only through an oracle. This is in contrast to the standard online model, where the learner knows the concept class in advance. The learning protocol is a sequential game between a learner and an adversary. Let $\mathcal{C} \subset \{0,1\}^{\mathcal{X}}$ be a concept class, where $\mathcal{X}$ is the instance space and $\{0,1\}$ is the label space. Let $\mathcal{F}$ be a family of concept classes, known to the learner (e.g., classes with finite Littlestone dimension d_{LD}), such that $\mathcal{C}$ is an unknown concept class from $\mathcal{F}$ chosen by the adversary. Suppose the learner only has oracle access to $\mathcal{C}$ via an oracle $\mathcal{O}$. The sequential game proceeds for T rounds, as follows. For each $t \in [T]$:

1. The adversary chooses $(x_t, y_t) \in \mathcal{X} \times \{0,1\}$.
2. The learner observes x_t, picks a distribution $\Delta_t \in \Delta(\{0,1\})$, and predicts $\hat{y}_t \sim \Delta_t$.
3. The adversary reveals $y_t \in \{0,1\}$ and the learner suffers a loss $\mathbb{I}[\hat{y}_t \neq y_t]$.

Let H_{t-1} denote the history (transcript) prior to round t, including all past examples, predictions, and oracle queries and responses. Unless stated otherwise, the adversary is adaptive: it may choose (x_t, y_t) as a possibly randomized function of H_{t-1}, independent of the learner's internal randomness for round t. At the start of the round, only the instance x_t is revealed to the learner.

No realizability assumption is made in the agnostic setting. The regret of algorithm $\mathcal{A}$ with respect to $\mathcal{C}$ is defined as

$$\mathrm{Reg}_T(\mathcal{A}; \mathcal{C}) = \sum_{t=1}^{T} \mathbb{1}[\hat{y}_t \neq y_t] - \min_{c \in \mathcal{C}} \sum_{t=1}^{T} \mathbb{1}[c(x_t) \neq y_t].$$

For adaptive adversaries, both terms are evaluated on the transcript before taking expectation. Thus an adaptive expected-regret guarantee R_T means

$$\sup_{\mathcal{C} \in \mathcal{F}} \sup_{\mathrm{Adv} \in \mathfrak{A}_{\mathrm{ad}}} \mathbb{E}\left[\mathrm{Reg}_T(\mathcal{A}; \mathcal{C})\right] \leq R_T,$$

where $\mathfrak{A}_{\mathrm{ad}}$ denotes the class of adaptive adversaries described above, and the expectation is with respect to both the learner's randomness and any randomness of the adversary.

Let $N_T(\mathcal{A})$ be the (random) total number of oracle calls that $\mathcal{A}$ makes to $\mathcal{O}(\mathcal{C})$ over rounds $1, \ldots, T$. We use N_T for this realized query count and

$$Q_T(\mathcal{A}) = \sup_{\mathcal{C} \in \mathcal{F}} \sup_{\mathrm{Adv} \in \mathfrak{A}_{\mathrm{ad}}} \mathbb{E}\left[N_T(\mathcal{A})\right]$$

for the worst-case expected query complexity.

Definition 2.1 (Oracle Types) We define the two oracles that are used in this paper. Let $\mathcal{C}$ be the concept class we have access to.

- "Weak Consistency": Given any sequence $S = ((x_i, y_i))_{i=1}^n \in (\mathcal{X} \times \{0,1\})^n$, the Weak Consistency Oracle returns "realizable" if there exists some $c \in \mathcal{C}$ consistent with S. Otherwise, it returns "not realizable". This is the oracle used for the upper bounds in this paper (Theorem 4.1, Theorem 4.5).
- ("Agnostic") ERM: Given any sequence $S = ((x_i, y_i))_{i=1}^n \in (\mathcal{X} \times \{0,1\})^n$, the Agnostic ERM Oracle returns a a concept with the minimal error: $\mathrm{argmin}_{c \in \mathcal{C}} \sum_{(x,y) \in S} \mathbb{I}[c(x) \neq y]$. This oracle is used for the lower bound in Theorem 5.1.

Other variants of ERM oracles were defined in [AHR25], such as a restricted ERM oracle (which is applied only to the observed data throughout the interaction) and a realizable ERM oracle (which returns a concept only for realizable datasets), neither of which is used in our work. Note that our lower bound holds for the strongest form of such an ERM oracle. See [AHR25] for the precise definitions of these oracle models. Other terminology (VC dimension and Littlestone dimension) is defined in Section B.

3 Background: Agnostic to Realizable Reduction and its Suboptimality

Before presenting our oracle-efficient algorithm, we rigorously formalize the classic information-theoretic reduction from agnostic to realizable online learning established by [BDPSS09] (see Algorithm 1). Detailing this algorithm is essential for isolating the source of its computational barrier and for clarifying how the weak consistency oracle enables us to perfectly simulate and prune unrealizable experts in the construction.

Let $\mathcal{A}$ denote a deterministic algorithm that makes at most M mistakes on any sequence of data strictly realizable by some $c \in \mathcal{C}$. The reduction treats $\mathcal{A}$ as a black box and simulates a massive ensemble of online "experts." The goal of each expert is to perfectly output the sequence of labels generated by the unknown optimal concept $c^* \in \mathcal{C}$. Because c^* belongs to $\mathcal{C}$, its label sequence is strictly realizable. Thus, if the base learner $\mathcal{A}$ were to learn from the labels of c^*, it would be guaranteed to make at most M prediction mistakes. Each expert attempts to perfectly simulate c^* by guessing the exact subset of rounds where these mistakes occur.

Formally, an expert is uniquely parameterized by a *mistake schedule*: a binary vector $I \in \{0,1\}^T$ subject to the sparsity constraint $\|I\|_1 \leq M$. The t-th coordinate of this vector, $I_t \in \{0,1\}$, acts as an instruction for the expert at round t: if $I_t = 1$, the expert assumes its internal base learner $\mathcal{A}_I$ is about to make a mistake (i.e., disagree with c^*), and forcibly flips its prediction. The total number of valid experts is $N = \sum_{j=0}^{M} \binom{T}{j} = \mathcal{O}(T^M)$.

Algorithm 1 Agnostic-to-Realizable Reduction [BDPSS09]

Subroutines: Base learner $\mathcal{A}$ for the realizable setting that makes at most M mistakes.

Initialize: **(i)** A uniform probability distribution over all valid mistake schedules. Let $N = \sum_{j=0}^{M} \binom{T}{j}$ be the total number of valid experts. For all $I \in \{0,1\}^T$ such that $\|I\|_1 \leq M$, set the initial weight $w_0(I) = \frac{1}{N}$. **(ii)** Initialize a separate instance of the base learner $\mathcal{A}_I$ for each expert. **(iii)** Let η denote the learning rate.

For $t = 1, 2, \ldots, T$:

1. Receive $x_t \in \mathcal{X}$.
2. **Expert Predictions.** For each schedule I:
   (a) Query internal base learner to get prediction $p_t(I) = \mathcal{A}_I(x_t)$.
   (b) Generate pseudo-label: $v_t(I) = p_t(I) \oplus I_t$.
   (If $I_t = 1$, the expert assumes $\mathcal{A}_I$'s prediction is wrong and flips it).
3. **Predict:** Run Hedge (Exponential Weights) over the experts. Predict $\hat{y}_t = 1$ with probability proportional to the sum of weights of experts predicting 1: $P_t(1) = \frac{\sum_{I:v_t(I)=1} w_{t-1}(I)}{\sum_I w_{t-1}(I)}$
4. **Update:** Observe true label y_t.
   (a) Update weights for each I: $w_t(I) = w_{t-1}(I) \cdot \exp(-\eta \cdot \mathbb{I}[v_t(I) \neq y_t])$.
   (b) Advance base learners: Feed $(x_t, v_t(I))$ to $\mathcal{A}_I$ as the "true" label.

To play the agnostic game, the algorithm runs Hedge over this entire pool of experts. The statistical guarantee of this reduction relies on the survival of the *Optimal Expert*. For the optimal hypothesis $c^* \in \mathcal{C}$ in hindsight, let I^* be the exact rounds where $\mathcal{A}$ disagrees with $c^*(x_t)$. Since tracking c^* generates the strictly realizable sequence $v^* = (c^*(x_1), \ldots, c^*(x_T))$, the base learner is guaranteed to make $\leq M$ mistakes. Thus, $\|I^*\|_1 \leq M$. The expert parameterized by I^* flawlessly tracks c^*, guaranteeing an expected agnostic regret of $\mathcal{O}(\sqrt{MT\log(T)})$. The base learner is chosen to be the Standard Optimal Algorithm (SOA), which achieves the optimal mistake bound of d_{LD} in the realizable setting, and the resulting agnostic regret bound is near-optimal.

Bottleneck 1: unrealizable experts and Sauer's Lemma. The reduction is oblivious to the structure of the concept class $\mathcal{C}$. Consequently, it maintains every pseudo-label sequence $v_{1:t}$ containing at most d_{LD} mistakes, each corresponding to a labeling (dichotomy) over the observed points $x_1, \ldots, x_t$. By Sauer's Lemma, the number of realizable dichotomies induced by $\mathcal{C}$ on any set of t points is bounded by $\mathcal{O}(t^{d_{\mathrm{VC}}})$. This means that plenty of the $\mathcal{O}(T^{d_{\mathrm{LD}}})$ sequences tracked by the

standard reduction are *unrealizable experts* – labelings that could never be produced by any actual hypothesis in the class.

Bottleneck 2: SOA intractability and large oracle complexity. A second fundamental limitation lies in the computational intractability of the SOA itself. Executing the SOA requires computing the Littlestone dimension at each round, a task known to be computationally hard even when the concept class and feature space are finite [MR17, Man22]. Consequently, a natural computational model relying solely on calls to an offline Empirical Risk Minimization (ERM) oracle has been proposed [AAD+23, KS24, AHR25]. For such oracle-based learners, the best achievable realizable mistake bound is $\Omega(2^{d_{\mathrm{LD}}})$, and an expected regret of $\Omega(\sqrt{T2^{d_{\mathrm{LD}}}})$ in the agnostic case [AHR25]. However, achieving this regret using the reduction in Algorithm 1 requires maintaining $\mathcal{O}(T^{2^{d_{\mathrm{LD}}}})$ active experts, which inherently demands an intractable $\mathcal{O}(T^{2^{\mathcal{O}(d_{\mathrm{LD}})}})$ ERM oracle calls per round [AAD+23]. Our main goal is to reduce significantly this oracle complexity.

Bypassing the bottlenecks. This establishes the core motivation of our work. By utilizing an offline ERM oracle, or formally, a strictly weaker *weak consistency oracle* (Definition 2.1), to actively verify realizability and dynamically prune impossible dichotomies, we bypass both bottlenecks simultaneously. First, at any round t, we collapse the active memory footprint of the reduction from $\mathcal{O}(T^{d_{\mathrm{LD}}})$ down to the true capacity of the class, $\mathcal{O}(t^{d_{\mathrm{VC}}})$[6]. Second, when using an oracle-efficient realizable base learner, this active pruning reduces the total oracle complexity from $\mathcal{O}\left(T^{2^{\mathcal{O}(d_{\mathrm{LD}})}}\right)$ down to $\mathcal{O}\left(T^{d_{\mathrm{VC}}+1}\right)$.

4 Adaptive Dynamic Expert Pruning Tree

In this section, we demonstrate that if the learner is granted access to an offline optimization oracle that checks whether a dataset is strictly realizable in $\mathcal{C}$, namely a weak consistency oracle, the double-exponential tracking tree of Algorithm 1 can be significantly improved. This leads to a substantial reduction in the total number of offline oracle calls compared to [AAD+23], by using even a weaker oracle, and also shows that the memory requirements of the general reduction can be improved.

Algorithm Overview. Our `ADEPT` reduction (Algorithm 2) bypasses the $\mathcal{O}(T^M)$ combinatorial explosion by shifting the tracking mechanism from abstract mistake schedules to explicitly tracking valid label prefixes. At each round t, we propose binary extensions $u = v \circ b$ for each active parent prefix $v \in V_{t-1}$ and apply two filtering mechanisms to shrink the state space. First, *realizability pruning* queries the weak consistency oracle: if a sequence u contradicts the concept class $\mathcal{C}$, it evaluates to `UNREALIZABLE` and is discarded. By Sauer's Lemma, this dynamically bounds the active tree width to $\mathcal{O}(t^{d_{\mathrm{VC}}})$ paths. Second, *mistake pruning* simulates the internal base learner $\mathcal{A}$ along each surviving path. If a sequence exhausts the learner's guaranteed mistake budget (i.e., $k(u) > M$), it is permanently pruned. Crucially, because the true optimal hypothesis $c^* \in \mathcal{C}$ is strictly realizable and bounded by M mistakes, its true sequence of labels is guaranteed to safely survive both filters.

To safely run Hedge (Exponential Weights) over this dynamically pruned active set V_t, we must account for the varying "future capacity" of each path. An active prefix $u \in V_t$ represents a bundle of full-length experts as in Algorithm 1, sharing the exact same history up to time t. To assign each prefix the weight it would receive under Algorithm 1, we define the combinatorial weight of a sequence u as $W_t(u) = \sum_{j=0}^{M-k(u)} \binom{T-t}{j}$. This counts the exact combinatorial volume of valid future paths extending from u. By scaling the standard Hedge update by this future capacity, yielding the unnormalized weight $w_t(u) = W_t(u) \cdot \exp(-\eta L_{t-1}(v))$, our aggregated prediction outputs the identical probability distribution as explicitly running Hedge over the full $\mathcal{O}(T^M)$ global experts. Consequently, our reduction inherits optimal expected agnostic regret guarantees while operating strictly within the bounded $\mathcal{O}(t^{d_{\mathrm{VC}}})$ memory footprint. Figure 2 provides a visualization of the this algorithm over timesteps.

We formally present our main theorem characterizing the general efficiency of our algorithm.

[6]Here and throughout, this memory comparison refers to the number of maintained high-level expert states, equivalently active realizable pseudo-label prefixes; the raw implementation memory is obtained by multiplying this active-state count by the per-prefix state size of the chosen realizable learner, as captured by $S_{\mathcal{A}}(t)$ in Theorem 4.1.

Algorithm 2 Adaptive Dynamic Expert Pruning Tree (ADEPT)

Subroutines: Base learner $\mathcal{A}$ for the realizable setting that makes at most M mistakes, weak consistency oracle.

Initialize: **(i)** Active set of pseudo-label prefixes: $V_0 = \{\emptyset\}$, **(ii)** Base learner mistake counts: $k(\emptyset) = 0$, **(iii)** True cumulative losses against observed labels: $L_0(\emptyset) = 0$, **(iv)** Learning rate: $\eta = \sqrt{\frac{8M \ln(eT/M)}{T}}$.

For $t = 1, 2, \ldots, T$:

1. Receive instance $x_t \in \mathcal{X}$. Initialize the next active set $V_t = \emptyset$.
2. **For** each active parent prefix $v = (v_1, \ldots, v_{t-1}) \in V_{t-1}$:
   (a) **Base Learner Step:** Get prediction $p_t = \mathcal{A}(v, x_t)$.
   (b) **For** each proposed extension bit $b \in \{0, 1\}$:
      i. Let $u = v \circ b$ denote the proposed sequence.
      ii. **Realizability Pruning:** Query the weak consistency oracle on the dataset:
      $$S = \big\{(x_1, v_1), \ldots, (x_{t-1}, v_{t-1}), (x_t, b)\big\}$$
      iii. **If** the oracle returns UNREALIZABLE, discard extension u and **continue**.
      iv. **Mistake Update:** Set $k(u) = k(v) + \mathbb{I}[b \neq p_t]$.
      v. **If** $k(u) \leq M$:
      - Add u to V_t.
      - Algorithm 1 Weight: $W_t(u) = \sum_{j=0}^{M-k(u)} \binom{T-t}{j}$.
      - Hedge Weight: $w_t(u) = W_t(u) \cdot \exp(-\eta L_{t-1}(v))$.
3. **Predict:** Output $\hat{y}_t \in \{0, 1\}$, predicting 1 with probability: $P_t(\hat{y}_t = 1) = \frac{\sum_{u \in V_t : u_t = 1} w_t(u)}{\sum_{u \in V_t} w_t(u)}$
4. **Update:** Observe true label y_t. For each active sequence $u \in V_t$, set: $L_t(u) = L_{t-1}(u_{1:t-1}) + \mathbb{I}[u_t \neq y_t]$.

Theorem 4.1 (Main Theorem: Regret, Memory, and Computational Guarantees) *Let $\mathcal{C}$ be a concept class with VC dimension d_{VC}. Let $\mathcal{A}$ be a deterministic base learner for the realizable setting that makes at most M mistakes on any strictly realizable sequence, requires $S_{\mathcal{A}}(t)$ memory per state, and makes $Q_{\mathcal{A}}(t)$ queries to a weak consistency oracle per round. Against any adaptively chosen adversarial sequence of length T, the ADEPT reduction (Algorithm 2) guarantees:*

- *Regret: An expected regret bounded by $\mathbb{E}[\mathrm{Regret}_T] \leq \mathcal{O}\left(\sqrt{T \cdot M \log T}\right)$.*
- *Memory Complexity: A dynamically bounded active tree size of at most $\mathcal{O}(t^{d_{\mathrm{VC}}})$ valid paths at any round t, yielding a global memory of size $\mathcal{O}\left(t^{d_{\mathrm{VC}}} \cdot S_{\mathcal{A}}(t)\right)$.*
- *Query Complexity: At most $\mathcal{O}\left(t^{d_{\mathrm{VC}}} \cdot (1 + Q_{\mathcal{A}}(t))\right)$ total queries to the weak consistency oracle per round.*

Consequences of the Main Theorem. The structural decoupling of the mistake bound (M) and the capacity of the concept class (d_{VC}) in Theorem 4.1 yields the following consequences for agnostic online learning. First, instantiating the base learner with an oracle-efficient realizable algorithm significantly reduces the total number of oracle calls required in the agnostic setting.

Theorem 4.2 (Total Oracle Efficiency) *Let $\mathcal{A}$ be the oracle-efficient realizable learner of [AHR25, AAD+23], which guarantees $M = 2^{\mathcal{O}(d_{\mathrm{LD}})}$. Algorithm 2 (ADEPT) uses $\mathcal{O}(t^{d_{\mathrm{VC}}})$ weak consistency calls per round (and $\mathcal{O}\left(T^{d_{\mathrm{VC}}+1}\right)$ overall), while achieving the same regret $\tilde{\mathcal{O}}\left(\sqrt{T \cdot 2^{\mathcal{O}(d_{\mathrm{LD}})}}\right)$, which is near-optimal for an oracle-based algorithm.*

In comparison, Algorithm 1 requires $\mathcal{O}\left(T^{2^{\mathcal{O}(d_{\mathrm{LD}})}}\right)$ oracle calls. Since $d_{\mathrm{VC}} \leq d_{\mathrm{LD}}$ (and can be arbitrarily smaller), the improvement is at least exponential and can be double exponential in the dimension for some classes. Second, even when focusing purely on the minimax optimal regret, our reduction provides an exponential memory reduction over Algorithm 1.

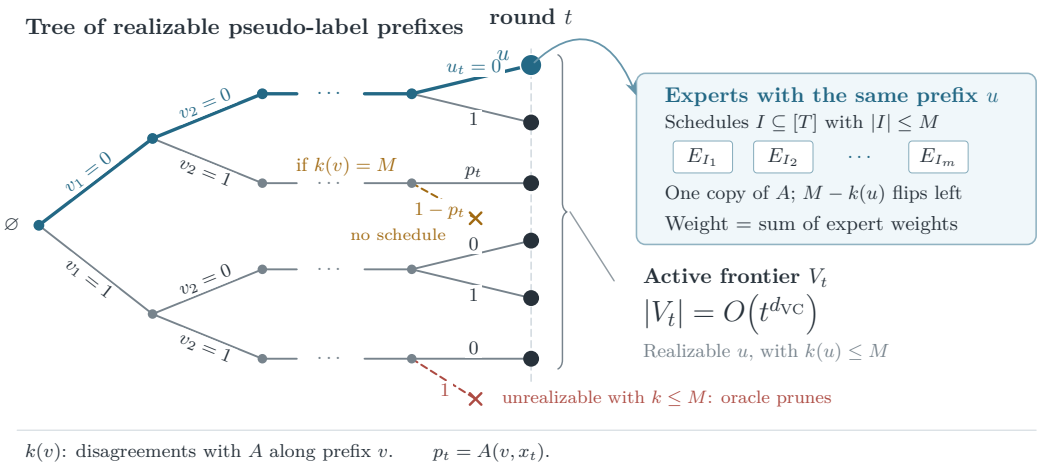

Figure 2: **A visualization of ADEPT** Above is a visualization of the realizable pseudo-label prefixes. After timesteps, prefixes either become pruned because it's no longer realizable, or because the number of mistakes on the pseudo-labels with respect to the base realizable algorithm has exceeded M.

Corollary 4.3 (Memory Efficiency via SOA) *Let $\mathcal{A}$ be the Standard Optimal Algorithm (SOA) [Lit88], achieving the optimal realizable mistake bound $M = d_{\mathrm{LD}}$. Algorithm 2 (ADEPT) matches the near-optimal $\tilde{\mathcal{O}}\left(\sqrt{T \cdot d_{\mathrm{LD}}}\right)$ expected regret while maintaining only $\mathcal{O}(t^{d_{\mathrm{VC}}})$ active paths per round. In comparison, Algorithm 1 must explicitly track $\mathcal{O}(T^{d_{\mathrm{LD}}})$ experts.*

Finally, our reduction can be made instance-dependent by simply adaptively tuning the learning rates.

Corollary 4.4 (Instance-Dependent Bound) *Setting $\eta_t \propto 1/\sqrt{L^*_{t-1} + 1}$, where $L^*_{t-1} = \min_{u \in V_{t-1}} L_{t-1}(u_{1:t-1})$ is the tracked empirical loss of the best active path, achieves a first-order expected regret*
$\mathcal{O}\left(\sqrt{L^* \cdot M \log T} + M \log T\right)$, *where L^* is the true optimal loss in hindsight. This interpolates between fully adversarial environments ($L^* \propto T$, yielding $\tilde{\mathcal{O}}(\sqrt{T \cdot M})$ regret) and the realizable setting ($L^* = 0$, bounding the total mistakes by $\mathcal{O}(M \log T)$, which recovers the base learner's realizable guarantee up to a logarithmic factor).*

The full proofs appear in Section C.

Proof Sketch of Theorem 4.1 The proof proceeds in the following technical steps:

Weight Monotonicity under Pruning: We first establish that the combinatorial future capacity weights obey the one-sided relation $W_{t-1}(v) \geq W_t(v \circ 0) + W_t(v \circ 1)$, where inactive extensions are assigned weight 0. Pascal's identity gives equality for the two unpruned potential extensions; after realizability or mistake-budget pruning, one of the extensions $v \circ b$ may be inactive, so only the inequality is guaranteed. This is sufficient for the potential argument, since pruning can only remove nonnegative mass from the active tree.

Regret Analysis via the Hedge Potential: Let $Z_t = \sum_{u \in V_t} W_t(u) \exp(-\eta L_t(u))$ be the Hedge potential function (the total weight of all active paths) at the end of round t. In standard analysis, the algorithm's cumulative loss is upper-bounded by tracking the decay of this potential function. Let $Z'_t = \sum_{u \in V_t} W_t(u) \exp(-\eta L_{t-1}(u_{1:t-1}))$ be the intermediate potential function before observing y_t. When our algorithm dynamically prunes branches, it discards strictly positive probability mass, which guarantees that the intermediate mass strictly shrinks ($Z'_t \leq Z_{t-1}$). Since a smaller potential function tightens the theoretical upper bound on the algorithm's loss, maintaining fewer experts can only improve, and never harm, the algorithmic regret (provided the optimal expert is not discarded). Letting $\ell_t = \mathbb{P}(\hat{y}_t \neq y_t)$ denote the expected prediction loss of the algorithm, applying Hoeffding's Lemma to the prediction distribution yields the standard recursive relation

$Z_t \le Z'_t \exp(-\eta\ell_t + \eta^2/8) \le Z_{t-1} \exp(-\eta\ell_t + \eta^2/8)$. Telescoping this to T yields the algorithmic upper bound: $Z_T \le Z_0 \exp\left(-\eta \sum_{t=1}^{T} \ell_t + T\eta^2/8\right)$.

Survival of the Optimal Hypothesis: Let $c^* = \arg\min_{h \in \mathcal{C}} L_T(h)$ be the optimal hypothesis in hindsight, and let $v^* = (c^*(x_1), \ldots, c^*(x_T))$ be its corresponding sequence of predictions. By definition, their cumulative losses are identical: $L_T(v^*) = L_T(c^*)$. Because $c^* \in \mathcal{C}$, the sequence v^* is strictly realizable and will never be pruned by the weak consistency oracle. Furthermore, because $\mathcal{A}$ guarantees $\le M$ mistakes on realizable data, v^* never exceeds the mistake budget. Thus, $v^* \in V_T$. Since $W_T(v^*) = \binom{0}{0} = 1$, we can strictly lower-bound the final potential by the weight of this single surviving sequence: $Z_T \ge W_T(v^*) \exp(-\eta L_T(v^*)) = \exp(-\eta L_T(c^*))$. Comparing this lower bound with the upper bound on Z_t from the previous step, noting the initial state is bounded $Z_0 \le (eT/M)^M$, rearrangement with optimally tuned η yields the $\mathcal{O}(\sqrt{T \cdot M \log T})$ expected regret.

Active Tree Width Is Bounded via Sauer's Lemma: Finally, we bound the per-round complexity. At the start of round t, the active set V_{t-1} contains only pseudo-label prefixes that are strictly realizable over the previously observed instances $x_{1:t-1}$. By Sauer's Lemma, the number of such dichotomies is bounded by $|V_{t-1}| \le \sum_{i=0}^{d_{\mathrm{VC}}} \binom{t-1}{i} = \mathcal{O}(t^{d_{\mathrm{VC}}})$. In Step 2 of the algorithm, we query the base learner once per active parent, and propose exactly 2 binary extensions, executing exactly $2|V_{t-1}|$ weak consistency queries. Summing these directly caps the total per-round query complexity at $\mathcal{O}(t^{d_{\mathrm{VC}}} \cdot (1 + Q_{\mathcal{A}}(t)))$, completing the proof. ■

4.1 Tradeoffs between Oracle Calls and Regret

While the above result achieves $\sqrt{T}$-type regret (for constant Littlestone dimension) using $O(T^{d_{\mathrm{VC}}})$ ERM queries, it is possible to further reduce the query complexity at the cost of worse, yet still sublinear, regret.

Theorem 4.5 (Oracle-Regret Tradeoff) *Let $\mathcal{C}$ be a concept class with Littlestone dimension d_{LD} and VC dimension d_{VC}, and let M be the mistake bound for the realizable base learner of* ADEPT*. For every $c \in (0, 1)$, there exists a randomized online learner such that, against any adaptive adversary over T rounds, $\mathbb{E}[\mathrm{Reg}_T] \le \mathcal{O}\left(T^{1-c/2}\sqrt{d_{\mathrm{LD}} + (M+1)\log(eT)}\right)$, and the learner makes $\mathcal{O}\left(T^{1+cd_{\mathrm{VC}}}\right)$ weak-consistency oracle calls in total.*

This gives a continuous interpolation between the full-query $\tilde{\mathcal{O}}(\sqrt{T})$ regime and lower-query algorithms with larger, but still sublinear, regret exponents. The idea to prove it is based on the adversarial uniform laws of large numbers [ABED+21], from which one can (informally) show that given an algorithm that has average regret α when restricted to a subsample of $\frac{1}{\epsilon^2}$ indices will have average regret $\alpha + O(\epsilon)$ on the entire sequence. Details can be found in Section D.

5 Lower Bounds for Tradeoffs between Oracle Calls and Regret

We next state a query-budget lower bound showing any learner must incur $\omega(\sqrt{T})$ regret.

Theorem 5.1 (Lower Bound for Bounded Query Algorithms with an ERM oracle) *For every integer $d \ge 1$, there exists a family $\mathcal{F}_d$ of concept classes with VC dimension and Littlestone dimension equal to d such that the following holds. For any horizon T and query budget Q, against an adaptive adversary, every randomized online learner that makes at most Q total queries to an ERM oracle (Definition 2.1) suffers expected regret at least $\frac{T}{2}\min\left\{1, \frac{d}{Q+1}\right\} - \frac{Q}{2}$. Consequently, whenever $d \le Q + 1$ and $Q = o(\sqrt{dT})$, the learner suffers expected regret $\Omega\left(\frac{dT}{Q}\right)$.*

In particular, for $d = 1$ and for any $\epsilon \in (0, \frac{1}{2})$, a learner that uses $Q \le T^{1/2-\epsilon}$ ERM queries incurs expected regret $\Omega(T^{1/2+\epsilon})$. If the number of queries is bounded by an absolute constant independent of T, then the regret is linear. This lower bound is complementary to [AHR25]. Their result gives a dimension-dependent lower bound that applies even to learners making an arbitrarily large finite

number of ERM queries: some classes require regret $\Omega(\sqrt{T2^{d_{\mathrm{LD}}}})$. By contrast, Theorem 5.1 is a query-budget lower bound: already for classes with $d_{\mathrm{LD}} = 1$, any learner using $Q = o(\sqrt{T})$ ERM queries suffers regret $\Omega(T/Q) = \omega(\sqrt{T})$. Thus, our bound is stronger in the low-query regime, whereas [AHR25] applies beyond this regime.

Hazan and Koren [HK16] prove a related lower bound in the finite N-expert optimizable-oracle model: vanishing regret requires $\widetilde{\Omega}(\sqrt{N})$ total computation in the worst case. Their result does not directly subsume ours, since it is a total-runtime lower bound, whereas our lower bound isolates the number of ERM-oracle calls and leaves computation between calls uncharged. Thus the two results address related computational barriers under different complexity measures.

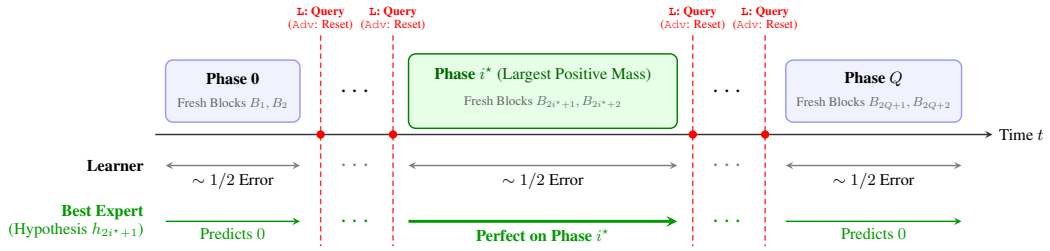

Figure 3: Illustration of the adaptive reset lower bound. The adversary uses two fresh hidden blocks in each phase, chosen from a set whose partition is unexposed by earlier queries and responses, and starts a new phase after any round containing an ERM query. On rounds with no query before prediction, the learner incurs expected loss $1/2$. In hindsight, the best singleton selects the phase with the most positive examples and predicts 0 elsewhere, saving at least a $1/(Q+1)$ fraction of the total positive mass.

Proof Sketch of Theorem 5.1 For simplicity, we outline the proof for $d = 1$; the extension to general d appears in Section E. It suffices to consider the following subfamily of classes. For a countable partition $\mathcal{P}$ of $\mathcal{X}$ into infinite blocks, let

$$\mathcal{C}_{\mathcal{P}} = \{\mathbf{1}_B : B \in \mathcal{P}\}.$$

These block-singleton classes have VC dimension and Littlestone dimension equal to 1: one point can be shattered, but after observing a positive label, the target block is fixed.

To formalize the hidden blocks, first partition $\mathcal{X}$ into countably many infinite sets U_j, and independently split each U_j into two blocks using independent fair membership bits. Fix an ERM tie-breaking rule independent of these bits. A finite query and its returned concept can expose the partitions of only finitely many sets U_j, while the remaining assignments stay independent.

The adversary runs phases. In each phase it chooses a set U_j that has neither been exposed by earlier queries and responses nor used in an earlier phase. Denote its two hidden blocks by B_{2i+1}, B_{2i+2}. The adversary presents fresh points from U_j in a fixed order, labels points in B_{2i+1} by 1 and points in B_{2i+2} by 0, and starts a new phase after any round on which the learner queries. Since there are infinitely many sets U_j, a fresh unexposed pair is always available.

On every round with no oracle query before prediction, the fresh membership bit remains unbiased given the learner's information, so the learner's expected loss is $1/2$. There are at most Q rounds with a query before prediction, and hence $\mathbb{E}[L_{\mathrm{alg}}] \geq \frac{T-Q}{2}$. Here the expectation also includes the initial random partition.

Let P_i be the number of positive examples in phase i, let $P = \sum_i P_i$, and let $m \leq Q+1$ be the number of phases. The best singleton can choose the positive block of the phase $i^\star$ with largest P_i, so its loss is at most $P - \max_i P_i \leq P - \frac{P}{m} \leq \frac{Q}{Q+1}P$. Since $\mathbb{E}[P] = T/2$, taking the difference gives $\mathbb{E}[\mathrm{Reg}_T] \geq \frac{T}{2(Q+1)} - \frac{Q}{2}$. Finally, averaging over the initial partition gives a fixed concept class and a valid ERM oracle for which the same expected-regret lower bound holds. ■

Acknowledgments

IA is supported by the National Science Foundation under Grant ECCS-2217023, through the Institute for Data, Econometrics, Algorithms, and Learning (IDEAL).

References

[AAD+23] Angelos Assos, Idan Attias, Yuval Dagan, Constantinos Daskalakis, and Maxwell K Fishelson. Online learning and solving infinite games with an erm oracle. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 274–324. PMLR, 2023.

[ABED+21] Noga Alon, Omri Ben-Eliezer, Yuval Dagan, Shay Moran, Moni Naor, and Eylon Yogev. Adversarial laws of large numbers and optimal regret in online classification. In *Proceedings of the 53rd annual ACM SIGACT symposium on theory of computing*, pages 447–455, 2021.

[ACBG02] Peter Auer, Nicolo Cesa-Bianchi, and Claudio Gentile. Adaptive and self-confident on-line learning algorithms. *Journal of Computer and System Sciences*, 64(1):48–75, 2002.

[AHK+14] Alekh Agarwal, Daniel Hsu, Satyen Kale, John Langford, Lihong Li, and Robert Schapire. Taming the monster: A fast and simple algorithm for contextual bandits. In *International conference on machine learning*, pages 1638–1646. PMLR, 2014.

[AHR25] Idan Attias, Steve Hanneke, and Arvind Ramaswami. Tradeoffs between mistakes and erm oracle calls in online and transductive online learning. *Advances in Neural Information Processing Systems*, 39, 2025.

[BDPSS09] Shai Ben-David, Dávid Pál, and Shai Shalev-Shwartz. Agnostic online learning. In *COLT*, volume 3, page 1, 2009.

[BRS24] Adam Block, Alexander Rakhlin, and Abhishek Shetty. On the performance of empirical risk minimization with smoothed data. In *The Thirty Seventh Annual Conference on Learning Theory*, pages 596–629. PMLR, 2024.

[CBL06] Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

[CBMS07] Nicolo Cesa-Bianchi, Yishay Mansour, and Gilles Stoltz. Improved second-order bounds for prediction with expert advice. *Machine Learning*, 66(2):321–352, 2007.

[DG24] Constantinos Daskalakis and Noah Golowich. Is efficient pac learning possible with an oracle that responds. In *The Thirty Seventh Annual Conference on Learning Theory*, pages 1263–1307. PMLR, 2024.

[DHK+11] Miroslav Dudik, Daniel Hsu, Satyen Kale, Nikos Karampatziakis, John Langford, Lev Reyzin, and Tong Zhang. Efficient optimal learning for contextual bandits. *arXiv preprint arXiv:1106.2369*, 2011.

[DHL+20] Miroslav Dudík, Nika Haghtalab, Haipeng Luo, Robert E Schapire, Vasilis Syrgkanis, and Jennifer Wortman Vaughan. Oracle-efficient online learning and auction design. *Journal of the ACM (JACM)*, 67(5):1–57, 2020.

[FHMM23] Yuval Filmus, Steve Hanneke, Idan Mehalel, and Shay Moran. Optimal prediction using expert advice and randomized littlestone dimension. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 773–836. PMLR, 2023.

[FL98] Moti Frances and Ami Litman. Optimal mistake bound learning is hard. *Information and Computation*, 144(1):66–82, 1998.

[FR20] Dylan Foster and Alexander Rakhlin. Beyond ucb: Optimal and efficient contextual bandits with regression oracles. In *International conference on machine learning*, pages 3199–3210. PMLR, 2020.

[HBD23] Niki Hasrati and Shai Ben-David. On computable online learning. In *International Conference on Algorithmic Learning Theory*, pages 707–725. PMLR, 2023.

[HHSY22] Nika Haghtalab, Yanjun Han, Abhishek Shetty, and Kunhe Yang. Oracle-efficient online learning for beyond worst-case adversaries. *arXiv preprint arXiv:2202.08549*, 2022.

[HK16] Elad Hazan and Tomer Koren. The computational power of optimization in online learning. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 128–141, 2016.

[HRS24] Nika Haghtalab, Tim Roughgarden, and Abhishek Shetty. Smoothed analysis with adaptive adversaries. *Journal of the ACM*, 71(3):1–34, 2024.

[KS24] Alexander Kozachinskiy and Tomasz Steifer. Simple online learning with consistent oracle. In *The Thirty Seventh Annual Conference on Learning Theory*, pages 3241–3256. PMLR, 2024.

[KV05] Adam Kalai and Santosh Vempala. Efficient algorithms for online decision problems. *Journal of Computer and System Sciences*, 71(3):291–307, 2005.

[Lit88] Nick Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine learning*, 2:285–318, 1988.

[Man22] Pasin Manurangsi. Improved inapproximability of vc dimension and littlestone's dimension via (unbalanced) biclique. *arXiv preprint arXiv:2211.01443*, 2022.

[MR17] Pasin Manurangsi and Aviad Rubinstein. Inapproximability of vc dimension and littlestone's dimension. In *Conference on Learning Theory*, pages 1432–1460. PMLR, 2017.

[MRT12] Mehryar Mohri, A Rostamizadeh, and A Talwalkar. Foundations of machine learning, ser. adaptive computation and machine learning, 2012.

[SKS16] Vasilis Syrgkanis, Akshay Krishnamurthy, and Robert Schapire. Efficient algorithms for adversarial contextual learning. In *International Conference on Machine Learning*, pages 2159–2168. PMLR, 2016.

[SLX22] David Simchi-Levi and Yunzong Xu. Bypassing the monster: A faster and simpler optimal algorithm for contextual bandits under realizability. *Mathematics of Operations Research*, 47(3):1904–1931, 2022.

[SS+12] Shai Shalev-Shwartz et al. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194, 2012.

[SSBD14] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[VC71] VN Vapnik and A Ya Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications*, 16(2):264, 1971.

A Additional Related Work

Oracle-Efficient Online Learning. Oracle-efficient methods establish a powerful framework for addressing the computational challenges of online learning. This provides an alternative to the fact that standard online learning can be computationally intractable [FL98, HBD23]. Assos et al. [AAD+23] and Kozachinskiy et al. [KS24] studied the online learnability of concept classes with finite Littlestone dimension using the ERM oracle. In addition, the partial-information setting, specifically contextual bandits, has been explored using regression oracles [FR20, SLX22] and the ERM oracle [SKS16].

Theoretical limitations of oracle-efficient online learning have also been investigated. Hazan and Koren [HK16] considered prediction with a finite set of N experts in an oracle model where a black-box optimizer returns the empirical leader for any history. They gave a vanishing-regret algorithm with total running time $\widetilde{O}(\sqrt{N})$, and proved a matching lower bound up to logarithmic factors on the total running time in this oracle model. Thus their result is best viewed as a runtime separation between offline optimization and adversarial online learning for finite expert classes, rather than as a lower bound on the number of ERM-oracle calls. For finite classes whose size N is exponential in a relevant dimension parameter, this corresponds to an exponential dependence on that parameter; however, their lower bound is formulated in terms of N and total runtime.

Recent work of Attias et al. [AHR25] is closer to the oracle-access model considered here. They study online and transductive online learning when the learner interacts with the concept class only through ERM or weak-consistency oracles, and evaluate both mistakes/regret and oracle calls. In the standard online setting with ERM access, they show that finite oracle access cannot avoid exponential-in-d_{LD} mistake/regret bounds, including $\Omega(2^{d_{\mathrm{LD}}})$ mistakes in the realizable case and $\Omega(\sqrt{T\,2^{d_{\mathrm{LD}}}})$ regret in the agnostic case. Our lower bounds are complementary: they are stated directly as query-budget tradeoffs, and apply even to classes with very small Littlestone dimension. Altogether, these findings underscore that offline optimization oracles do not, by themselves, remove the sequential or oracle-query complexity of fully adversarial online learning.

A separate line of work develops FTPL-style oracle-efficient algorithms for structured online decision problems. Kalai and Vempala [KV05], Dudík et al. [DHL+20], and Syrgkanis et al. [SKS16] use offline optimization oracles together with problem-specific structure. These oracles are stronger than the weak consistency access considered here: in contextual learning, for example, the offline oracle returns a policy minimizing cumulative cost-sensitive loss on an arbitrary history, rather than merely deciding whether a labeled sample is realizable.

Syrgkanis et al. [SKS16] studied oracle-efficient algorithms for adversarial contextual learning. Their FTPL-style algorithms access the policy class through an offline policy-optimization oracle and obtain sublinear regret for finite policy classes under additional structure: either in the transductive setting, where the learner knows the context set in advance, or in a non-transductive setting with a small separator. They also extend the transductive contextual-experts guarantee to infinite classes of bounded Natarajan dimension, using a Sauer–Shelah-type finite-projection argument. In the binary case, this gives a transductive upper bound for VC classes. This finite-projection idea is related to the transductive weak-oracle results of Attias et al. [AHR25], but the oracle models differ: Syrgkanis et al. assume a more powerful optimization oracle over policies and losses, while Attias et al. show that, in the transductive binary setting, the induced labelings can be recovered using weak consistency queries. Our work uses the same weak-consistency perspective in the fully online agnostic setting.

This VC/Natarajan guarantee is inherently transductive. They show that in fully online adaptive settings, finite VC dimension alone is insufficient: even a VC-dimension-one threshold class can force linear regret. Our work is complementary. We assume finite Littlestone dimension, the relevant condition for fully online learnability, and use VC dimension only to bound the number of oracle-verified realizable prefixes maintained by the algorithm.

Complementary work obtains oracle efficiency under beyond-worst-case assumptions on the adversary. Haghtalab et al. [HHSY22] gave oracle-efficient algorithms for smoothed online learning, where the adversary's instance distributions have bounded density relative to a base measure, and obtained regret bounds depending on the VC/pseudo-dimension and the smoothness parameter. Related work of Block et al. [BRS24] shows that repeated ERM succeeds for well-specified smooth data, with error controlled by the usual statistical complexity of the class. These results show that smoothness can make VC-type complexity sufficient for efficient online learning. Our work is orthogonal: we allow

fully adversarial instance sequences, assume finite Littlestone dimension for online learnability, and study the oracle-query complexity of achieving agnostic regret guarantees.

Weak Consistency Oracles. Daskalakis et al. [DG24] introduced the *weak consistency oracle* as a weaker alternative to ERM in distributional learning: given a labeled sample, the oracle only reports whether the sample is realizable by the class. They showed that this oracle suffices for realizable PAC learning of any binary VC class with polynomial sample and oracle complexity, and extended the framework to agnostic learning, multiclass classification, regression, and partial concept classes.

Weak consistency access has also been studied in online learning. Attias et al. [AHR25] showed that realizable online learning algorithms based on ERM access can be simulated using weak consistency queries, and studied related query/mistake tradeoffs in online and transductive online models. Our work continues this line in the fully online agnostic setting, using weak consistency queries to obtain no-regret guarantees whose oracle complexity is controlled by combinatorial parameters of the class.

B Additional Preliminaries

Definition B.1 (Littlestone Dimension [Lit88]) Given a concept class $\mathcal{C}$, a d-depth Littlestone tree is a set of $\{x_{\mathbf{y}} : \mathbf{y} \in \mathcal{Y}^t, t \in \{0, d-1\}\} \subset \mathcal{X}$ (interpreting $\mathcal{Y}^0 = \{()\}$, where for all $y_1, y_2, \ldots, y_d \in \mathcal{Y}$, there's a $c \in \mathcal{C}$ such that $(c(x_{()}), c(x_{y_1}), c(x_{y_{1:2}}), \ldots, c(x_{y_{1:(d-1)}})) = (y_1, y_2, \ldots, y_d)$. The Littlestone dimension of $\mathcal{C}$ is the maximum n such that there exists a Littlestone tree of depth n.

Definition B.2 (VC Dimension [VC71]) Given a concept class $\mathcal{C}$, a set of n points $x_1, \ldots, x_n \subset \mathcal{X}$ is shattered if $\{(c(x_1), \ldots, c(x_n)) : c \in \mathcal{C}\} = \{0,1\}^n$. The VC dimension of $\mathcal{C}$ is the largest n such that there exist n shattered points in $\mathcal{X}$.

C Proofs for Section 4

The proof of Theorem 4.1 proceeds by decomposing the analysis into four lemmas. First, we establish that the combinatorial future capacity (the weight assigned by Algorithm 1 [BDPSS09]) telescopes exactly over valid extensions (Lemma C.1). Next, we prove that dynamically pruning unrealizable paths strictly improves the standard Hedge potential function (Lemma C.2). We then demonstrate that the true optimal hypothesis survives all pruning steps (Lemma C.3), yielding the near-optimal regret bound. Finally, we bound the memory and computational complexity via VC theory (Lemma C.4). We conclude the section by combining these lemmas to prove the main theorem.

Lemma C.1 (Weight Monotonicity under Pruning) *For any active prefix $v \in V_{t-1}$, adopt the convention that $W_t(v \circ b) = 0$ whenever the extension $v \circ b$ is inactive (for example, because it is pruned by the weak consistency oracle or exceeds the mistake budget). Then the total combinatorial future weight of the surviving extensions is at most the weight of v:*

$$W_{t-1}(v) \geq W_t(v \circ 0) + W_t(v \circ 1).$$

Proof Let $p_t = \mathcal{A}(v, x_t) \in \{0,1\}$ be the base learner's prediction. First ignore realizability pruning and consider the two formal extensions $b = p_t$ and $b = 1 - p_t$. For the extension matching the base learner ($b = p_t$), no mistake is charged, so $k(v \circ p_t) = k(v)$. For the extension differing from the base learner ($b = 1 - p_t$), exactly one mistake is charged, so $k(v \circ (1 - p_t)) = k(v) + 1$.

Assume first that $k(v) < M$, and write $\widetilde{W}_t$ for the unpruned formal branch weights. Summing the combinatorial weights of these two formal branches yields:

$$\widetilde{W}_t(v \circ p_t) + \widetilde{W}_t(v \circ (1 - p_t)) = \sum_{j=0}^{M-k(v)} \binom{T-t}{j} + \sum_{j=0}^{M-k(v)-1} \binom{T-t}{j}.$$

To apply Pascal's Identity $\left[\binom{n}{j} + \binom{n}{j-1} = \binom{n+1}{j}\right]$, we extract the $j = 0$ term from the first summation and re-index the second summation:

$$= \binom{T-t}{0} + \sum_{j=1}^{M-k(v)} \binom{T-t}{j} + \sum_{j=1}^{M-k(v)} \binom{T-t}{j-1}$$

$$= \binom{T-t}{0} + \sum_{j=1}^{M-k(v)} \left[\binom{T-t}{j} + \binom{T-t}{j-1}\right]$$

$$= \binom{T-t+1}{0} + \sum_{j=1}^{M-k(v)} \binom{T-t+1}{j} = \sum_{j=0}^{M-k(v)} \binom{T-t+1}{j}.$$

This final expression is exactly $W_{t-1}(v)$. For the boundary case where $k(v) = M$, the mistaken formal branch $1 - p_t$ has $k = M + 1$ and contributes weight 0, while the matching branch p_t contributes $\binom{T-t}{0} = 1$, again matching $W_{t-1}(v) = \binom{T-t+1}{0} = 1$.

Thus, Pascal's identity gives equality for the two formal, unpruned extensions. In the actual active tree, however, an extension $v \circ b$ might be inactive and therefore assigned weight 0; this can happen, for example, if the weak consistency oracle prunes it. Since making an extension inactive only removes a nonnegative term from the formal sum, the surviving active extensions satisfy

$$W_{t-1}(v) \geq W_t(v \circ 0) + W_t(v \circ 1).$$

■

Lemma C.2 (Regret Analysis via the Hedge Potential) *Let $Z_t = \sum_{u \in V_t} W_t(u) \exp(-\eta L_t(u))$ be the potential at round t. Let $\ell_t = \mathbb{P}(\hat{y}_t \neq y_t)$ be the algorithm's expected loss at round t. Then $Z_T \leq Z_0 \exp\left(-\eta \sum_{t=1}^{T} \ell_t + \frac{T\eta^2}{8}\right)$.*

Proof Let $Z_t = \sum_{v \in V_t} W_t(v) \exp(-\eta L_t(v))$. In round t, before the true label y_t is revealed, the normalizer of the surviving active set V_t is:

$$Z'_t = \sum_{u \in V_t} W_t(u) \exp(-\eta L_{t-1}(u_{1:t-1})).$$

Since V_t contains only the strictly realizable subset of all possible binary extensions from V_{t-1}, and weights are non-negative, Lemma C.1 guarantees:

$$\begin{aligned} Z'_t &\leq \sum_{v \in V_{t-1}} (W_t(v \circ 0) + W_t(v \circ 1)) \exp(-\eta L_{t-1}(v)) \\ &\leq \sum_{v \in V_{t-1}} W_{t-1}(v) \exp(-\eta L_{t-1}(v)) \\ &= Z_{t-1}. \end{aligned}$$

Thus, pruning can only strictly decrease the normalizer: $Z'_t \leq Z_{t-1}$.

The algorithm predicts $\hat{y}_t$ using the probability distribution P_t defined over V_t, proportional to $w_t(u) = W_t(u) \exp(-\eta L_{t-1}(u_{1:t-1}))$. Notice that $Z'_t = \sum_{u \in V_t} w_t(u)$ is exactly the normalizer of this distribution. After observing y_t, the loss of branch u is $\ell_t(u) = \mathbb{I}[u_t \neq y_t] \in \{0, 1\}$. The expected loss of the algorithm is $\ell_t = \sum_{u \in V_t} \frac{w_t(u)}{Z'_t} \ell_t(u)$. Viewing the branch loss $\ell_t(u)$ as a random variable drawn from P_t, Hoeffding's Lemma for distributions bounded in $[0, 1]$ guarantees:

$$\sum_{u \in V_t} \frac{w_t(u)}{Z'_t} \exp(-\eta \ell_t(u)) = \mathbb{E}_{u \sim P_t}\left[\exp(-\eta \ell_t(u))\right] \leq \exp\left(-\eta \ell_t + \frac{\eta^2}{8}\right).$$

Multiplying by Z'_t and noting that $w_t(u) \exp(-\eta \ell_t(u)) = W_t(u) \exp(-\eta L_t(u)) = Z_t$ gives the updated potential Z_t:

$$Z_t \leq Z'_t \exp\left(-\eta \ell_t + \frac{\eta^2}{8}\right) \leq Z_{t-1} \exp\left(-\eta \ell_t + \frac{\eta^2}{8}\right).$$

Telescoping this inequality iteratively from $t = 1$ to T yields $Z_T \leq Z_0 \exp\left(-\eta \sum_{t=1}^{T} \ell_t + \frac{T\eta^2}{8}\right)$. ■

Lemma C.3 (Survival of the Optimal Expert) *The algorithm achieves an expected agnostic regret bound of $\mathcal{O}\left(\sqrt{T \cdot M \log T}\right)$ against any hypothesis in $\mathcal{C}$.*

Proof Let $c^* = \arg\min_{h \in \mathcal{C}} L_T(h)$ be the true optimal hypothesis in hindsight. The optimal comparison pseudo-label sequence is defined as $v^* = (c^*(x_1), \ldots, c^*(x_T))$. Since c^* belongs to the concept class $\mathcal{C}$, the sequence v^* is exactly the labeling produced by a valid hypothesis on the data sequence. Consequently, at any round t, the prefix $v^*_{1:t}$ is strictly realizable on $x_{1:t}$, and the weak consistency oracle will never return `UNREALIZABLE` when queried on $v^*_{1:t}$. Furthermore, because $\mathcal{A}$ guarantees at most M mistakes on any realizable sequence, the number of disagreements between $\mathcal{A}$ and c^* is bounded by M. Thus, $k(v^*_{1:t}) \leq M$ for all t.

Therefore, the prefix v^* never triggers the realizability or the mistake-bound pruning conditions, ensuring $v^* \in V_T$. Since v^* survives, we can lower bound the final potential. At the final round ($t = T$), the remaining future capacity of the tree is zero ($T - T = 0$). By the definition of the binomial coefficient, $\binom{0}{0} = 1$ and $\binom{0}{j} = 0$ for all $j \geq 1$. Since the optimal sequence never exceeds the mistake budget ($k(v^*) \leq M$), the summation safely includes the $j = 0$ term and perfectly collapses:

$$W_T(v^*) = \sum_{j=0}^{M-k(v^*)} \binom{0}{j} = \binom{0}{0} + 0 + \cdots = 1.$$

Furthermore, because v^* is exactly the sequence of predictions generated by the true optimal hypothesis c^*, its empirical cumulative loss is identically $L_T(v^*) = L_T(c^*)$. Since v^* is guaranteed to survive pruning and all terms in the potential Z_T are non-negative, we can cleanly lower-bound the final potential by the weight of this single surviving sequence:

$$Z_T = \sum_{u \in V_T} W_T(u) \exp(-\eta L_T(u)) \geq W_T(v^*) \exp(-\eta L_T(v^*)) = 1 \cdot \exp(-\eta L_T(c^*)).$$

The initial normalizer is the sum of all combinatorial paths bounded by M mistakes:

$$Z_0 = \sum_{j=0}^{M} \binom{T}{j} \leq \left(\frac{eT}{M}\right)^M.$$

Let $L_{\text{algo}} = \sum_{t=1}^{T} \ell_t$. Combining the upper bound from on Z_t from Lemma C.2 with the lower bound yields:

$$\exp(-\eta L_T(c^*)) \leq \left(\frac{eT}{M}\right)^M \exp\left(-\eta L_{\text{algo}} + \frac{T\eta^2}{8}\right).$$

Taking the logarithm and rearranging terms:

$$L_{\text{algo}} - L_T(c^*) \leq \frac{M \ln(eT/M)}{\eta} + \frac{\eta T}{8}.$$

Substituting the optimal learning rate $\eta = \sqrt{\frac{8M \ln(eT/M)}{T}}$ yields the expected regret bound:

$$\mathbb{E}[\text{Regret}_T] \leq \sqrt{\frac{1}{2} T M \ln\left(\frac{eT}{M}\right)} = \mathcal{O}\left(\sqrt{T \cdot M \log T}\right).$$

■

Lemma C.4 (Active Tree Width Is Bounded via Sauer's Lemma) *At round t, the active set size $|V_{t-1}|$ is bounded by $\mathcal{O}(t^{d_{\text{VC}}})$, and the algorithm issues a total of $\mathcal{O}(t^{d_{\text{VC}}})$ queries to the weak consistency oracle for realizability pruning.*

Proof At the beginning of round t, the active set V_{t-1} contains only pseudo-label sequences v of length $t-1$. By the construction of the algorithm, a sequence v is only added to V_{t-1} if the oracle verified it was strictly realizable on the data sequence $x_{1:t-1}$. Therefore, every element $v \in V_{t-1}$ corresponds to a distinct dichotomy over $x_{1:t-1}$ that can be realized by at least one hypothesis in $\mathcal{C}$. Thus, V_{t-1} is precisely a subset of the projection of $\mathcal{C}$ onto the points $(x_1, \ldots, x_{t-1})$.

By Sauer's Lemma, the maximum number of realizable dichotomies on any sequence of $t-1$ points by a class with VC dimension d_{VC} is strictly bounded by:

$$|V_{t-1}| \leq \sum_{i=0}^{d_{\mathrm{VC}}} \binom{t-1}{i} = \mathcal{O}(t^{d_{\mathrm{VC}}}).$$

This structurally bounds the number of experts maintained in memory to $\mathcal{O}(t^{d_{\mathrm{VC}}})$. In round t, for each prefix in V_{t-1}, the reduction proposes exactly 2 extensions ($b \in \{0, 1\}$) and queries the weak consistency oracle once for each. Thus, the total number of oracle calls incurred by the adaptive reduction is exactly $2|V_{t-1}| \leq \mathcal{O}(t^{d_{\mathrm{VC}}})$. When incorporating the base learner $\mathcal{A}$ making $Q_{\mathcal{A}}(t)$ queries per state, the total global bound on oracle queries becomes $\mathcal{O}(t^{d_{\mathrm{VC}}} \cdot (1 + Q_{\mathcal{A}}(t)))$. ■

Proof of Theorem 4.1 The theoretical guarantees of the reduction follow immediately from combining the preceding lemmas. By Lemma C.3, the algorithm's expected regret against any hypothesis in $\mathcal{C}$ is bounded by $\mathcal{O}(\sqrt{T \cdot M \log T})$, which provides the optimal statistical bound. Simultaneously, Lemma C.4 ensures that the geometric pruning successfully bounds the space complexity and the per-round oracle query complexity to $\mathcal{O}(t^{d_{\mathrm{VC}}} \cdot (1 + Q_{\mathcal{A}}(t)))$. Combining these statistical and structural guarantees establishes the complete statement of Theorem 4.1. ■

Proof of Theorem 4.2 The regret bound follows by substituting the realizable mistake bound $M = 2^{\mathcal{O}(d_{\mathrm{LD}})}$ into the regret guarantee of Theorem 4.1.

We first briefly describe the realizable base learner of [AAD+23] and its weak-consistency implementation from [AHR25]. The learner of [AAD+23] proceeds in phases and maintains a pool of hypotheses, whose predictions it aggregates using lazy multiplicative weights. At the end of each phase, it adds a hypothesis consistent with all examples observed so far. [AAD+23] shows that the number of phases, and hence the number of hypotheses in the pool, is at most $2^{\mathcal{O}(d_{\mathrm{LD}})}$.

Theorem 3.3 of [AHR25] implements this learner using a weak-consistency oracle. The simulation represents each hypothesis by its labeling of the points observed so far. To evaluate such a hypothesis on x_t, it queries the current labeling extended by $(x_t, 0)$; if this extension is unrealizable, it assigns label 1 to x_t. Thus, at most one weak-consistency call is needed per hypothesis, and a single Base Learner Step at timestep t makes at most $2^{\mathcal{O}(d_{\mathrm{LD}})}$ weak-consistency calls.

We now prove the query bound. A direct application of the query guarantee of Theorem 4.1 is insufficient to obtain the claimed bound. Indeed, the weak-consistency implementation described above may make $2^{\mathcal{O}(d_{\mathrm{LD}})}$ queries per round for each copy of the base learner. Therefore, directly summing the per-round guarantee from Theorem 4.1 would give

$$\sum_{t=1}^{T} \mathcal{O}\left(t^{d_{\mathrm{VC}}}\left(1 + 2^{\mathcal{O}(d_{\mathrm{LD}})}\right)\right) = \mathcal{O}\left(T^{d_{\mathrm{VC}}+1} 2^{\mathcal{O}(d_{\mathrm{LD}})}\right).$$

However, the same input to the weak-consistency oracle may arise in many different copies of the base learner. Repeated calls to the same oracle input can be replaced by the previously obtained answer. It therefore suffices to bound the number of distinct inputs to the oracle. We now show that, across all copies of the base learner and the realizability-pruning step, there are only $\mathcal{O}\left(t^{d_{\mathrm{VC}}}\right)$ distinct oracle inputs on round t.

In the base learner described above, the weak-consistency oracle on round t is queried only on sequences of the form

$$(x_1, z_1), (x_2, z_2), \ldots, (x_{t-1}, z_{t-1}), (x_t, 0),$$

where

$$(x_1, z_1), (x_2, z_2), \ldots, (x_{t-1}, z_{t-1})$$

is realizable.

Furthermore, the realizability-pruning step of Algorithm 2 only queries sequences of the form

$$(x_1, v_1), (x_2, v_2), \ldots, (x_{t-1}, v_{t-1}), (x_t, b), \qquad b \in \{0, 1\},$$

where

$$(x_1, v_1), (x_2, v_2), \ldots, (x_{t-1}, v_{t-1})$$

is realizable.

Thus, every distinct oracle input on round t is obtained by extending a realizable labeling of $x_1, \ldots, x_{t-1}$ with one of the two possible labels for x_t. By Sauer's lemma, the number of such inputs is at most

$$2 \sum_{i=0}^{d_{\mathrm{VC}}} \binom{t-1}{i} = \mathcal{O}\left(t^{d_{\mathrm{VC}}}\right).$$

Therefore, the number of oracle calls on round t is $\mathcal{O}\left(t^{d_{\mathrm{VC}}}\right)$. Summing over all rounds gives

$$\sum_{t=1}^{T} \mathcal{O}\left(t^{d_{\mathrm{VC}}}\right) = \mathcal{O}\left(T^{d_{\mathrm{VC}}+1}\right),$$

as claimed. ■

Proof of Corollary 4.3 The Standard Optimal Algorithm (SOA) [Lit88] guarantees a realizable mistake bound of $M = d_{\mathrm{LD}}$. Substituting $M = d_{\mathrm{LD}}$ into the statistical bound of Theorem 4.1 yields an expected agnostic regret of $\mathcal{O}(\sqrt{T \cdot d_{\mathrm{LD}} \log T}) = \tilde{\mathcal{O}}(\sqrt{T \cdot d_{\mathrm{LD}}})$. The spatial bounding mechanism of Theorem 4.1 depends only on the VC dimension d_{VC} and not on M, strictly preserving the $\mathcal{O}(t^{d_{\mathrm{VC}}})$ dynamic memory footprint. ■

Proof of Corollary 4.4 The classical reduction of Algorithm 1 constructs an explicit ensemble of abstract experts. Each expert E represents a fixed mistake schedule for the internal base learner over the T rounds. Since the base learner has a realizable mistake bound of M, the total number of valid experts is bounded by:

$$N = \sum_{j=0}^{M} \binom{T}{j} \leq \left(\frac{eT}{M}\right)^{M}.$$

Let $\mathcal{E}$ denote this abstract set of N experts.

In the adaptive exponential-weights form used here, the unnormalized weight of an expert $E \in \mathcal{E}$ before predicting at time t is exactly $\exp(-\eta_t L_{t-1}(E))$, where $L_{t-1}(E)$ is the cumulative prediction loss of the expert against the true labels $y_{1:t-1}$. In our active tree, a child sequence $u \in V_t$ represents a massive bundle of full, T-round experts that share the exact same sequence of pseudo-label predictions up to time t. Thus, every full expert E extending the prefix u has the exact same empirical loss on the past history: $L_{t-1}(E) = L_{t-1}(u_{1:t-1})$.

Consequently, the total probability mass over the bundle of experts extending u factors out algebraically:

$$\sum_{E \text{ extending } u} \exp\big(-\eta_t L_{t-1}(E)\big) = \exp\big(-\eta_t L_{t-1}(u_{1:t-1})\big) \sum_{E \text{ extending } u} 1.$$

The combinatorial sum $\sum_{E \text{ extending } u} 1$ is exactly the number of valid future mistake schedules remaining for this specific path. This is precisely the weight (representing the future capacity) computed natively by our algorithm: $W_t(u) = \sum_{j=0}^{M-k(u)} \binom{T-t}{j}$, where $k(u)$ is the number of mistakes the internal base learner has made when tracking the sequence u.

Because the combinatorial weight $W_t(u)$ is independent of the learning rate η_t, updating the learning rate dynamically at each round perfectly preserves the exact marginalization. Therefore, our geometrically pruned tree outputs the same predictive distribution as the explicit adaptive Exponential Weights algorithm over $\mathcal{E}$, with experts deleted once their induced pseudo-label prefix is pruned.

By standard analysis of adaptive Exponential Weights tuned to the empirical best loss [ACBG02, CBMS07], setting $\eta_t = \min\left\{1/2, \sqrt{\frac{\ln N}{L^*_{t-1}+1}}\right\}$ guarantees an expected regret over the finite expert class $\mathcal{E}$ bounded by:

$$\mathbb{E}[\text{Regret}_T] \leq \mathcal{O}\left(\sqrt{L^* \ln N} + \ln N\right),$$

where L^* is the loss of the optimal expert $E^* \in \mathcal{E}$. Because dynamically pruning unrealizable branches strictly removes invalid experts, it only decreases the Hedge potential function (as formally shown in Lemma C.2), which strictly improves the standard potential-based upper bound on the algorithm's loss, so the above regret bound holds here too. As proven in Lemma C.3, there exists an expert in $\mathcal{E}$ that perfectly tracks the optimal hypothesis $c^* \in \mathcal{C}$, meaning L^* is bounded by the true loss of c^*.

Substituting $\ln N \leq M \ln(eT/M) = \mathcal{O}(M \log T)$ directly yields the claimed statistical bound. ■

D Proof of Theorem 4.5

To prove the theorem, we use tools from [ABED+21] about adversarial laws of large numbers to analyze how an algorithm run on a subsample still performs well on the entire sequence, and we state their results here for completeness. The setting is as follows:

Given a domain X, an adversary and sampler spend T rounds as follows:

For $t = 1 \dots T$,

- The adversary presents a point $x_t \in X$.
- The sampler decides whether or not the point should be in the sample.
- The sampler's decision is revealed to the adversary.

We also adopt the definition of an ϵ-approximation from [ABED+21], as follows:

Definition D.1 (ϵ-approximation) Let $\bar{z} = (z_1, \dots, z_T)$ be a sequence in $\mathcal{Z}$, let $I \subseteq [T]$ be a sampled set of indices, and let $\bar{s} = (z_t)_{t \in I}$ be the corresponding subsequence. For a set system $\mathcal{E} \subseteq 2^{\mathcal{Z}}$, we say that $\bar{s}$ is an ϵ-approximation of $\bar{z}$ with respect to $\mathcal{E}$ if

$$\sup_{E \in \mathcal{E}} \left| \frac{1}{|I|} \sum_{t \in I} \mathbb{1}[z_t \in E] - \frac{1}{T} \sum_{t=1}^{T} \mathbb{1}[z_t \in E] \right| \leq \epsilon.$$

Equivalently, the empirical frequency of every set $E \in \mathcal{E}$ on the sample uniformly approximates its frequency on the full adversarial stream.

For a set system $\mathcal{E} \subseteq 2^{\mathcal{Z}}$, we define $\text{Ldim}(\mathcal{E})$ to be the Littlestone dimension of the associated binary class

$$\{\mathbb{1}_E : E \in \mathcal{E}\} \subseteq \{0,1\}^{\mathcal{Z}}.$$

We use the following theorem of [ABED+21].

Lemma D.2 (Adversarial ULLN, [ABED+21]) *Let $\mathcal{E} \subseteq 2^{\mathcal{Z}}$ have Littlestone dimension d. There is a universal constant $C > 0$ such that, for every $\epsilon, \delta \in (0,1)$, a uniform sample of size*

$$k \geq C \frac{d + \log(1/\delta)}{\epsilon^2}$$

is an ϵ-approximation to any adversarially generated stream $\bar{z} = (z_1, \dots, z_T)$ with probability at least $1 - \delta$.

We will now apply this to our setting, where the "adversary" corresponds to the instances x_t, the predictions of the learner $\hat{y}_t$ and the true labels y_t, and the "sampler" corresponds to the timesteps where the learner changes their state. The idea is that predicting on T points (where some of them are in a sample), and only updating the state of the learner if it was in the sample, will allow the regret guarantees from a smaller sample to extend to the entire sequence. This is formalized via the following definition:

Definition D.3 (Sample-Blind Lazy Simulation) An online learner A is a *sample-blind lazy simulation* of a k-round learner B if it uses a random sampled set $I \subseteq [T]$, $|I| = k$, and satisfies:

1. Before predicting at time t, A does not inspect the current or future sampling decisions; it may use past sampling decisions.
2. A's internal copy of B is updated only after rounds $t \in I$.
3. On the sampled rounds, the internal state and predictions coincide with running B on the sampled subsequence $(x_t, y_t)_{t \in I}$ in chronological order.

Below, we describe the precise lazy version of ADEPT used in the subsampling argument. A subtlety is that ADEPT uses oracle calls while constructing its prediction: before predicting, it tentatively extends every active prefix, queries the oracle to remove unrealizable extensions, computes the corresponding mistake counters and weights, and only then predicts from the resulting weighted active set. Thus, in the lazy simulation, these computations must be performed speculatively on every round. If the round is sampled, we commit the tentative state; otherwise, we discard it and revert to the previous committed state. The rollback affects only the learner's internal state; oracle calls made during the speculative prediction are still counted.

Lazy rollback version of ADEPT. Let $K \leq T$ be the desired number of sampled updates. We run ADEPT internally as a K-round algorithm. For this internal execution, use the mistake budget $\overline{M} = \min\{M, K\}$ in place of M throughout. If $\overline{M} = 0$, set $\eta = 1$.Thus the Algorithm 1 future-capacity weights and the learning rate are computed with horizon K, not horizon T. Let

$$I \subseteq [T], \qquad |I| = K,$$

be a uniformly random sampled set. The prediction rule is not allowed to inspect whether $t \in I$ before predicting.

The learner maintains a committed ADEPT state after the sampled rounds observed so far. Let

$$r_t = |I \cap \{1, \ldots, t-1\}|$$

be the number of committed sampled updates before round t. The committed state consists of the active set V_{r_t}, the mistake counters $k(v)$ for $v \in V_{r_t}$, and the true-loss counters $L_{r_t}(v)$.

If $r_t = K$, the learner predicts 0, makes no oracle calls, and leaves its committed state unchanged. Since no further round is sampled, this does not affect the internal K-round execution. The following steps apply only when $r_t < K$.

At full-horizon round t:

1. The learner receives x_t.
2. Starting from the committed state at internal time r_t, the learner performs a *speculative ADEPT prediction step* on x_t. That is, it runs Steps 1–3 of Algorithm 2 as if x_t were the next internal ADEPT round, namely round $r_t + 1$ of a K-round execution:
   (a) initialize a tentative next active set $V^{\mathrm{tmp}}_{r_t+1} = \emptyset$;
   (b) for every active prefix $v \in V_{r_t}$ and every proposed extension bit $b \in \{0, 1\}$, query the weak consistency oracle on the labeled sequence consisting of the committed sampled history encoded by v, together with the proposed pair (x_t, b);
   (c) discard unrealizable extensions;
   (d) for surviving extensions $u = v \circ b$, compute the speculative mistake counter
   $$k^{\mathrm{tmp}}(u) = k(v) + \mathbb{1}[b \neq A(v, x_t)];$$
   discard u if $k^{\mathrm{tmp}}(u) > M$;
   (e) assign the speculative Algorithm 1 weight
   $$W_{r_t+1}(u) = \sum_{j=0}^{M-k^{\mathrm{tmp}}(u)} \binom{K-(r_t+1)}{j},$$
   and the speculative Hedge weight
   $$w_{r_t+1}(u) = W_{r_t+1}(u) \exp(-\eta L_{r_t}(v)).$$

3. The learner predicts using the same distribution as ADEPT would use at internal round $r_t + 1$:
$$\mathbb{P}(\hat{y}_t = 1) = \frac{\sum_{u \in V^{\text{tmp}}_{r_t+1} : u_{r_t+1}=1} w_{r_t+1}(u)}{\sum_{u \in V^{\text{tmp}}_{r_t+1}} w_{r_t+1}(u)}.$$
This prediction is made before the learner observes whether $t \in I$.
4. After the prediction, the learner observes y_t and the sampling bit $\mathbb{1}[t \in I]$.
5. If $t \in I$, the learner commits the tentative ADEPT transition by performing Step 4 of Algorithm 2:
$$L_{r_t+1}(u) = L_{r_t}(\text{parent}(u)) + \mathbb{1}[u_{r_t+1} \neq y_t] \qquad \text{for all } u \in V^{\text{tmp}}_{r_t+1}.$$
The committed state is set to
$$V_{r_t+1} = V^{\text{tmp}}_{r_t+1}.$$
6. If $t \notin I$, the learner discards $V^{\text{tmp}}_{r_t+1}$, all speculative mistake counters, all speculative weights, and any speculative base-learner computations. The committed state remains exactly the previous state at internal time r_t.

It's clear that the above algorithm is a sample-blind lazy simulation, since it predicts (steps 2 and 3) before determining if x_t was in the sample. Furthermore, if x_t was not in the sample, it reverts its state to what it originally was (step 6). Furthermore, the state of the algorithm on the K steps corresponding to I exactly matches what the state would be if $T = K$ and I is equal to all the instances.

To adapt AULLN [ABED+21] to a sample-blind lazy simulation, for each round, consider the augmented item $z_t = (x_t, y_t, \hat{y}_t) \in \mathcal{X} \times \{0,1\} \times \{0,1\}$. Define the augmented loss class as $\mathcal{E}_A = \{E_h : h \in \mathcal{C}\} \cup \{E_{\text{alg}}\}$ where
$$E_h = \{(x, y, \hat{y}) : h(x) \neq y\}, E_{\text{alg}} = \{(x, y, \hat{y}) : \hat{y} \neq y\}$$

Lemma D.4 (Augmented AULLN) *Given a class $\mathcal{C}$ with Littlestone dimension d_{LD}, let A be a sample-blind lazy simulation of a learner with*
$$k = \mathcal{O}\left(\frac{d_{\text{LD}} + \log(1/\delta)}{\epsilon^2}\right).$$
rounds. Then, with probability at least $1 - \delta$,
$$\sup_{E \in \mathcal{E}_A} \left| \frac{1}{k} \sum_{t \in S} \mathbb{1}[z_t \in E] - \frac{1}{T} \sum_{t=1}^{T} \mathbb{1}[z_t \in E] \right| \leq \epsilon.$$

Proof First, we show that $\mathcal{E}_A$ has Littlestone dimension at most $d_{\text{LD}} + 1$. First, the subclass $\{E_h : h \in \mathcal{C}\}$ has Littlestone dimension equal to that of $\mathcal{C}$ – the y-coordinate only flips the branch labels at a node, and the $\hat{y}$-coordinate is ignored. Adding the single concept E_{alg} can only increase the Littlestone dimension by 1 (otherwise, there's a shattered tree with depth $d_{\text{LD}} + 2$, implying that $\mathcal{C}$ has a shattered tree with depth $d_{\text{LD}} + 1$).

Directly applying Lemma D.2 gives that an ϵ-approximation of $\mathcal{E}_A$ holds with probability at least $1 - \delta$, so the result follows. ■

Lemma D.5 (Regret Transfer) *Let A be a sample-blind lazy simulation of a k-round learner B, and let $I \subseteq [T]$ be the sampled index set. Suppose I is an ϵ-approximation for the augmented class $\mathcal{E}_A$ from Lemma D.4. Then*
$$\frac{1}{T}\text{Reg}_T(A) \leq \frac{1}{k}\text{Reg}_I(A) + 2\epsilon$$
where
$$\text{Reg}_T(A) = \sum_{t=1}^{T} \mathbb{1}[\hat{y}_t \neq y_t] - \inf_{h \in \mathcal{C}} \sum_{t=1}^{T} \mathbb{1}[h(x_t) \neq y_t], \text{ and}$$
$$\text{Reg}_I(A) = \sum_{t \in I} \mathbb{1}[\hat{y}_t \neq y_t] - \inf_{h \in \mathcal{C}} \sum_{t \in I} \mathbb{1}[h(x_t) \neq y_t].$$

Proof First define $L_T(E) = \sum_{t=1}^{T} \mathbb{1}[z_t \in E], L_I(E) = \sum_{t \in I} \mathbb{1}[z_t \in E]$

Then, it holds that

$$\begin{aligned}
\frac{1}{T}\mathrm{Reg}_T(A) &= \frac{1}{T}\sum_{t=1}^{T} \mathbb{1}[\hat{y}_t \neq y_t] - \frac{1}{T}\inf_{h \in \mathcal{C}} \sum_{t=1}^{T} \mathbb{1}[h(x_t) \neq y_t] \\
&= \frac{1}{T} L_T(E_{\mathrm{alg}}) - \inf_{h \in \mathcal{C}} \frac{1}{T} L_T(E_h) \\
&\leq \left(\frac{1}{k} L_I(E_{\mathrm{alg}}) + \epsilon\right) - \inf_{h \in \mathcal{C}} \left(\frac{1}{k} L_I(E_h) - \epsilon\right) \\
&= \frac{1}{k} L_I(E_{\mathrm{alg}}) - \inf_{h \in \mathcal{C}} \frac{1}{k} L_I(E_h) + 2\epsilon \\
&= \frac{1}{k}\mathrm{Reg}_I(A) + 2\epsilon.
\end{aligned}$$

■

Proof of Theorem 4.5 Fix $c \in (0, 1)$ and set

$$K = \lfloor T^c \rfloor.$$

Run the sample-blind lazy rollback simulation described above, using a uniformly random sampled set $I \subseteq [T]$ of size K, and use the K-round version of ADEPT as the internal learner. Let M denote the realizable mistake bound of the base learner used inside ADEPT. Since the simulation is sample-blind, Lemma D.4 applies to the augmented stream

$$z_t = (x_t, y_t, \hat{y}_t)$$

even against an adaptive adversary.

Choose $\delta = (eT)^{-1}$ and

$$\epsilon = C_0 \sqrt{\frac{d_{\mathrm{LD}} + \log(eT)}{K}},$$

for a sufficiently large universal constant C_0. If $\epsilon \geq 1$, then the desired regret bound follows from the trivial bound $\mathrm{Reg}_T(A) \leq T$, after increasing the implicit constant. Hence assume $\epsilon < 1$.

Let G be the event that the sampled subsequence is an ϵ-approximation for the augmented class $\mathcal{E}_A$. By Lemma D.4, $\mathbb{P}(G) \geq 1 - \delta$. On G, Lemma D.5 gives

$$\mathrm{Reg}_T(A) \leq \frac{T}{K}\mathrm{Reg}_I(A) + 2\epsilon T.$$

On the sampled rounds, the lazy simulation coincides exactly with running ADEPT on the sampled subsequence. Therefore, by Theorem 4.1 with horizon K,

$$\mathbb{E}\big[\mathrm{Reg}_I(A)\big] \leq \mathcal{O}\left(\sqrt{KM\log(eK)}\right).$$

Using the preceding display on G, the trivial upper bound $\mathrm{Reg}_T(A) \leq T$ on G^c, and $|\mathrm{Reg}_I(A)| \leq K$, we get

$$\begin{aligned}
\mathbb{E}[\mathrm{Reg}_T(A)] &\leq \frac{T}{K}\mathbb{E}\big[\mathrm{Reg}_I(A)\big] + 2\epsilon T + \mathcal{O}(\delta T) \\
&\leq \mathcal{O}\left(T\sqrt{\frac{M\log(eK)}{K}}\right) + \mathcal{O}\left(T\sqrt{\frac{d_{\mathrm{LD}} + \log(eT)}{K}}\right) + \mathcal{O}(\delta T) \\
&\leq \mathcal{O}\left(T\sqrt{\frac{d_{\mathrm{LD}} + (M+1)\log(eT)}{K}}\right) \\
&\leq \mathcal{O}\left(T^{1-c/2}\sqrt{d_{\mathrm{LD}} + (M+1)\log(eT)}\right),
\end{aligned}$$

where we used $K \asymp T^c$, $K \leq T$, and $\delta T \leq 1$.

It remains to bound the number of weak-consistency oracle calls. At any external round, the committed internal history has length at most K. Hence the active set used in the speculative ADEPT step consists only of realizable labelings of at most K points, and Sauer's Lemma gives

$$|V| \le \sum_{i=0}^{d_{\mathrm{VC}}} \binom{K}{i} = \mathcal{O}\left(K^{d_{\mathrm{VC}}}\right).$$

Use the base learner and weak-consistency implementation from Theorem 4.2, retaining the committed base-learner states and caching repeated oracle inputs across all copies and pruning calls within each external round. As in that proof, every such query consists of a realizable labeling of the committed sampled instances, extended by (x_t, b) for some $b \in \{0, 1\}$. Thus, on a round with $r_t < K$, the number of distinct queries is at most

$$2\sum_{i=0}^{d_{\mathrm{VC}}} \binom{r_t}{i} \le 2\sum_{i=0}^{d_{\mathrm{VC}}} \binom{K}{i} = \mathcal{O}\left(K^{d_{\mathrm{VC}}}\right).$$

Rounds with $r_t = K$ make no oracle calls. Therefore, the total number of calls is

$$\mathcal{O}\left(TK^{d_{\mathrm{VC}}}\right) \le \mathcal{O}\left(T(T^c)^{d_{\mathrm{VC}}}\right) = \mathcal{O}\left(T^{1+cd_{\mathrm{VC}}}\right).$$

This proves the claimed oracle–regret tradeoff.

■

E Proof of Theorem 5.1

It is useful to interpret Theorem 5.1 as evidence for a separation between the dimensions governing computation and regret. The lower bound construction has $d_{\mathrm{VC}} = d_{\mathrm{LD}} = d$, but the parameter that appears on the query-complexity side is naturally viewed as the VC dimension: in the regime $d_{\mathrm{VC}} \le Q + 1$ and $Q = o(\sqrt{d_{\mathrm{VC}} T})$, the theorem gives

$$\mathbb{E}[\mathrm{Reg}_T] \ge \Omega\left(\frac{d_{\mathrm{LD}} T}{Q}\right).$$

Thus, in the low-query regime, the regret forced by a restricted oracle budget grows with the number of VC degrees of freedom that the learner must distinguish through queries. This is consistent with our upper bounds: the regret term is controlled by the realizable mistake bound, and in particular by the Littlestone dimension when the base learner is SOA, yielding $\widetilde{O}(\sqrt{T\, d_{\mathrm{LD}}})$ regret; whereas the number of oracle-verified realizable prefixes maintained by ADEPT is controlled by Sauer's lemma and hence by d_{VC}, leading to oracle complexity polynomial in T with exponent depending on d_{VC}. Informally, Littlestone dimension governs the statistical price of adversarial online prediction, while VC dimension governs the oracle/computational price.

Proof We formally prove the lower bound for $d = 1$, namely,

$$\mathbb{E}[\mathrm{Reg}_T] \ge \frac{T}{2(Q+1)} - \frac{Q}{2}.$$

Let $\mathcal{X} = \mathbb{N} \times \mathbb{N}$, partitioned into the countably infinite sets

$$U_j = \{j\} \times \mathbb{N}, \qquad j \ge 1.$$

Before the interaction, assign independent fair bits $\xi_x \in \{0, 1\}$ to all $x \in \mathcal{X}$, and define

$$D_{j,b} = \{x \in U_j : \xi_x = b\}, \qquad j \ge 1, \quad b \in \{0, 1\}.$$

Almost surely, these are disjoint infinite blocks partitioning $\mathcal{X}$. Let

$$\mathcal{C} = \{\mathbf{1}_{D_{j,b}} : j \ge 1,\ b \in \{0, 1\}\}.$$

This class has VC dimension and Littlestone dimension equal to 1: one point can be shattered, but a positive label fixes the target block. Throughout the argument, expectations include the initial random partition as well as the learner's randomness. We fix the partition by averaging at the end.

Fix an enumeration of the block indices (j, b), independently of the hidden bits, and let the ERM oracle return the first empirical minimizer in this enumeration. For any finite query, the empirical errors of all concepts are determined by the bits of the queried points: every block containing no queried point predicts zero throughout the query. Consequently, the index of the returned block depends only on these bits. The returned concept itself depends only on the partition of its own U_j.

For the information argument, augment the transcript as follows. For each oracle query, first expose the entire partition of every U_j containing a queried point. The index of the oracle's response is then determined; additionally expose the partition of the U_j containing the returned block. Each query exposes only finitely many sets U_j. By independence of the initial assignments, the remaining unexposed bits, other than those already revealed through observed labels, stay independent and fair.

The adversary proceeds in phases. At the beginning of phase i, it chooses the first unexposed set U_j that has not been used in an earlier phase. Such a set always exists, since there are infinitely many sets U_j and only finitely many have been exposed or used. For the remainder of the proof, denote its two blocks by

$$B_{2i+1} = D_{j,1}, \qquad B_{2i+2} = D_{j,0},$$

and write $h_{2i+1} = \mathbf{1}_{B_{2i+1}} \in \mathcal{C}$. These are merely names for the selected blocks; the partition, concept class, and oracle were fixed before the interaction. On each round of phase i, the adversary presents the next unused point x_t of U_j in a fixed ordering and sets $y_t = \xi_{x_t}$. Thus all examples in phase i are labeled according to h_{2i+1}.

The phase changes only when the learner makes an oracle query. More precisely, the adversary starts in phase 0. Whenever the learner makes at least one oracle query on a round, the adversary ends the current phase after that round and moves to the next phase. Since the learner makes at most Q queries, the adversary uses at most $Q + 1$ phases.

Before any oracle query on the current round, the bit ξ_{x_t} is fair conditional on the preceding augmented transcript and x_t. Indeed, the selected set was unexposed at the start of the phase, no earlier round of the same phase contained a query, and x_t has not appeared previously. The decision whether to make a first oracle query before predicting is made without access to this bit. Therefore, on a round with no oracle query before prediction, the learner's prediction is independent of ξ_{x_t} and its expected loss is $1/2$.

Let R be the number of rounds on which the learner makes at least one oracle query before predicting. Since the total number of queries is at most Q, we have $R \le Q$. Ignoring any loss on these rounds gives

$$\mathbb{E}[L_{\mathrm{alg}}] \ge \frac{T - \mathbb{E}[R]}{2} \ge \frac{T - Q}{2}.$$

Queries made after prediction do not affect this calculation, although they still end the current phase after that round.

It remains to upper bound the loss of the best expert in hindsight. For each phase $i \in \{0, \dots, Q\}$, let P_i denote the number of positive examples in phase i, namely, the number of examples lying in B_{2i+1}. Set $P_i = 0$ for any phase that is not reached, and let

$$P = \sum_{i=0}^{Q} P_i$$

be the total number of positive examples over the entire sequence. Since the label selected at the beginning of each round is a fresh fair bit, regardless of the adaptively chosen phase lengths,

$$\mathbb{E}[P] = \frac{T}{2}.$$

Now fix a phase i that is reached. The expert h_{2i+1} is correct on every example in phase i, because the labels in that phase are generated exactly according to h_{2i+1}. On every other phase, all examples lie outside B_{2i+1}, so h_{2i+1} predicts 0. Therefore, outside phase i, this expert makes mistakes exactly on the positive examples. Its total loss is

$$P - P_i.$$

Thus the loss of the best expert in hindsight is at most

$$P - \max_{i \in \{0,\dots,Q\}} P_i.$$

Using the deterministic inequality

$$\max_{i\in\{0,\dots,Q\}} P_i \geq \frac{1}{Q+1}\sum_{i=0}^{Q} P_i = \frac{P}{Q+1},$$

we get

$$\inf_{h\in\mathcal{C}} \sum_{t=1}^{T} \mathbf{1}[h(x_t)\neq y_t] \leq P - \frac{P}{Q+1} = \frac{Q}{Q+1}P.$$

Taking expectations gives

$$\mathbb{E}\left[\inf_{h\in\mathcal{C}} \sum_{t=1}^{T} \mathbf{1}[h(x_t)\neq y_t]\right] \leq \frac{Q}{Q+1}\mathbb{E}[P] = \frac{Q}{Q+1}\cdot\frac{T}{2}.$$

Combining the learner lower bound with the comparator upper bound,

$$\begin{aligned}\mathbb{E}[\mathrm{Reg}_T] &= \mathbb{E}[L_{\mathrm{alg}}] - \mathbb{E}\left[\inf_{h\in\mathcal{C}} \sum_{t=1}^{T} \mathbf{1}[h(x_t)\neq y_t]\right] \\ &\geq \frac{T-Q}{2} - \frac{Q}{Q+1}\cdot\frac{T}{2} \\ &= \frac{T}{2(Q+1)} - \frac{Q}{2},\end{aligned}$$

as desired.

To extend the argument to general d, replace the block-singleton class by the class of unions of at most d blocks:

$$\mathcal{C}_d = \left\{ x \mapsto \mathbf{1}\left[x \in \bigcup_{(j,b)\in S} D_{j,b}\right] : S \subseteq \mathbb{N}\times\{0,1\},\ |S|\leq d \right\}.$$

This class has VC dimension and Littlestone dimension exactly d. Indeed, points from d distinct blocks are shattered. Conversely, when the partition is known, predicting 1 on every block that has previously received a positive label and 0 elsewhere makes at most d mistakes on any realizable sequence. Hence its Littlestone dimension is at most d.

Fix an enumeration of the subsets of block indices of size at most d, independently of the hidden bits, and let the ERM oracle return the first empirical minimizer in this enumeration. As before, the selected subset depends only on the bits of the queried points. The returned concept is a union of at most d blocks and therefore depends only on the partitions of at most d sets U_j. Exposing these sets, together with those containing queried points, still exposes only finitely many sets per query.

The adversary uses the same phase construction as above, always choosing a previously unused, unexposed set U_j. The same independence argument therefore gives

$$\mathbb{E}[L_{\mathrm{alg}}] \geq \frac{T-Q}{2}, \qquad \mathbb{E}[P] = \frac{T}{2}.$$

The only change is the hindsight comparator: a concept in $\mathcal{C}_d$ can now select the positive blocks from the d phases with the largest positive mass. If $m \leq Q+1$ is the number of phases, let Top_d denote the indices of the $\min\{d,m\}$ largest values of P_i. Then

$$\sum_{i\in\mathrm{Top}_d} P_i \geq \min\left\{1, \frac{d}{m}\right\} P \geq \min\left\{1, \frac{d}{Q+1}\right\} P.$$

Hence the best comparator has expected loss at most

$$\left(1 - \min\left\{1, \frac{d}{Q+1}\right\}\right)\frac{T}{2}.$$

Combining this with the learner lower bound gives

$$\mathbb{E}[\mathrm{Reg}_T] \geq \frac{T}{2} \min\left\{1, \frac{d}{Q+1}\right\} - \frac{Q}{2}.$$

Finally, for each d, let $\mathcal{F}_d$ contain all concept classes arising from the corresponding construction as the partition varies over partitions with infinite blocks. This family does not depend on the learner, T, or Q. For any fixed randomized learner, the preceding bound holds after averaging over the initial random partition. Consequently, there exists a fixed partition, and hence a fixed concept class in $\mathcal{F}_d$ and a valid ERM oracle, for which the same lower bound holds in expectation over the learner's randomness against the adaptive phase adversary. This proves the theorem. ■

F Oracle-Efficient Learning in the k-Realizable Setting

We briefly discuss how the k-realizable viewpoint of [FHMM23] can be combined with the oracle-efficient realizable learner used in our work. Recall that a sequence is k-realizable by $\mathcal{C}$ if there exists a concept $h \in \mathcal{C}$ that makes at most k mistakes on the sequence. Let $L_k = L_k(\mathcal{C})$ denote the k-Littlestone dimension of $\mathcal{C}$, namely, the largest depth of a complete binary tree for which every root-to-leaf path is k-realizable by $\mathcal{C}$. As shown by [FHMM23], $L_k = \mathcal{O}\left(d_{\mathrm{LD}} + k\right)$.

The following result shows that the k-realizable setting can be reduced to realizable learning using oracle-efficient algorithms.

Theorem F.1 (Oracle-Efficient k-Realizable Learning) *Let $\mathcal{C} \subseteq \{0,1\}^{\mathcal{X}}$ be a concept class. There exists a deterministic online learner that makes at most $2^{\mathcal{O}(L_k)}$ mistakes on any k-realizable sequence. Furthermore, this learner can be implemented using:*

- $\mathcal{O}\left(T^{k+1} 2^{\mathcal{O}(L_k)}\right)$ *weak-consistency oracle calls;*
- $T 2^{\mathcal{O}(L_k)}$ *weak ERM oracle calls;*
- $2^{\mathcal{O}(L_k)}$ *agnostic ERM oracle calls.*

In particular, the mistake bound is at most $2^{\mathcal{O}(d_{\mathrm{LD}}+k)}$.

Proof We use a time-indexed reduction. Consider the domain

$$\widetilde{\mathcal{X}} = \mathcal{X} \times \mathbb{N}.$$

For every $h \in \mathcal{C}$, every $S \subseteq \mathbb{N}$ with $|S| \leq k$, and every collection $Z = (z_i)_{i \in S}$ of points in $\mathcal{X}$, define

$$h_{S,Z}(x,i) = \begin{cases} 1 - h(x), & \text{if } i \in S \text{ and } x = z_i, \\ h(x), & \text{otherwise.} \end{cases}$$

Let

$$\widetilde{\mathcal{C}}_k = \left\{h_{S,Z} : h \in \mathcal{C},\ S \subseteq \mathbb{N},\ |S| \leq k,\ Z = (z_i)_{i \in S}\right\}.$$

The set S records the at most k rounds on which the label is allowed to differ from h, while z_i records the corresponding instance on round i.

We claim that

$$\mathrm{Ldim}(\widetilde{\mathcal{C}}_k) \leq L_k.$$

To see this, consider a depth-d tree shattered by $\widetilde{\mathcal{C}}_k$, and fix any root-to-leaf path, realized by some $h_{S,Z}$. The same time-indexed point cannot occur twice along a path in a shattered tree: otherwise, at its second occurrence, one of the two outgoing edges would contradict the label chosen at its first occurrence. Since $h_{S,Z}$ differs from its underlying $h \in \mathcal{C}$ on at most $|S| \leq k$ time-indexed points, the labels along the path disagree with h on at most k nodes. Therefore, projecting each node (x,i) to x gives a depth-d tree in which every path is k-realizable by $\mathcal{C}$. Hence $d \leq L_k$, proving the claim.

The k-realizable problem can now be reduced to realizable learning over $\widetilde{\mathcal{C}}_k$. Indeed, suppose that $h \in \mathcal{C}$ makes mistakes on a set $S \subseteq [T]$, where $|S| \leq k$, and set $z_t = x_t$ for every $t \in S$. Then

$$h_{S,Z}(x_t, t) = y_t$$

for every $t \in [T]$. Thus, the time-indexed sequence

$$((x_1, 1), y_1), \ldots, ((x_T, T), y_T)$$

is realizable by $\widetilde{\mathcal{C}}_k$. Applying the oracle-efficient realizable learner [AAD+23, AHR25] therefore gives a mistake bound of

$$2^{\mathcal{O}\left(\mathrm{Ldim}(\widetilde{\mathcal{C}}_k)\right)} \leq 2^{\mathcal{O}(L_k)},$$

and hence a $2^{\mathcal{O}(d_{\mathrm{LD}}+k)}$ mistake bound.

It remains to discuss the oracle complexity. For the queries arising in this simulation, the time coordinates are distinct. A labeled sample of m time-indexed instances is realizable by $\widetilde{\mathcal{C}}_k$ if and only if there is a set of at most k labels whose reversal makes the corresponding sample realizable by $\mathcal{C}$. Thus, one weak-consistency query to $\widetilde{\mathcal{C}}_k$ can be simulated using at most

$$\sum_{j=0}^{k} \binom{m}{j}$$

weak-consistency queries to $\mathcal{C}$. Since the realizable learner makes $T2^{\mathcal{O}(L_k)}$ weak-consistency queries, this gives

$$T\left(\sum_{j=0}^{k} \binom{T}{j}\right) 2^{\mathcal{O}(L_k)} = \mathcal{O}\left(T^{k+1}2^{\mathcal{O}(L_k)}\right)$$

weak-consistency queries to $\mathcal{C}$.

This T^k overhead can be avoided using stronger oracles. The weak ERM oracle of [DG24] returns the minimum empirical error attained by a concept in $\mathcal{C}$, but not a minimizing concept. Therefore, a "weak consistency" query to $\widetilde{\mathcal{C}}_k$ can be answered with one weak ERM query: the time-indexed sample is realizable if and only if the minimum number of errors on the corresponding sample over $\mathcal{X}$ is at most k. This gives $T2^{\mathcal{O}(L_k)}$ weak ERM queries.

With the agnostic ERM oracle of [AHR25], which additionally returns a minimizing concept, the remaining factor of T can also be removed. Given a realizable-ERM query to $\widetilde{\mathcal{C}}_k$, call the agnostic ERM oracle on the corresponding labeled sample over $\mathcal{X}$. If the returned $h \in \mathcal{C}$ makes at most k errors, its error locations directly determine S and Z, and the resulting $h_{S,Z}$ is consistent with the time-indexed sample. Thus, each realizable-ERM query to $\widetilde{\mathcal{C}}_k$ can be simulated using one agnostic ERM query to $\mathcal{C}$, giving $2^{\mathcal{O}(L_k)}$ oracle calls in total. ■

Comparison with the first-order guarantee. A complementary guarantee follows from Corollary 4.4. For any realizable base learner with mistake bound M, this result gives

$$\mathbb{E}[\mathrm{Reg}_T] = \mathcal{O}\left(\sqrt{L^* M \log T} + M \log T\right),$$

where L^* is the loss of the best concept in hindsight. On a k-realizable sequence, $L^* \leq k$. If the base learner is SOA, so that $M = d_{\mathrm{LD}}$, this gives

$$\widetilde{\mathcal{O}}\left(\sqrt{k d_{\mathrm{LD}}} + d_{\mathrm{LD}}\right),$$

matching the dependence on k and d_{LD} of [FHMM23] up to logarithmic factors.

More relevant to our oracle model, using the oracle-efficient realizable learner with $M = 2^{\mathcal{O}(d_{\mathrm{LD}})}$ gives

$$\mathcal{O}\left(\sqrt{k\, 2^{\mathcal{O}(d_{\mathrm{LD}})} \log T} + 2^{\mathcal{O}(d_{\mathrm{LD}})} \log T\right)$$

expected regret, while retaining the same $\mathcal{O}\left(T^{d_{\mathrm{VC}}+1}\right)$ total weak-consistency query bound.

Thus, these two approaches provide complementary guarantees: the first-order result has a milder dependence on k and preserves the $\mathcal{O}\left(T^{d_{\mathrm{VC}}+1}\right)$ weak-consistency query bound, whereas the reduction above gives a horizon-independent $2^{\mathcal{O}(L_k)}$ mistake bound, together with $2^{\mathcal{O}(L_k)}$ oracle calls under agnostic ERM access.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [N/A].
- [N/A] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for [N/A]).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will also be asked to include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a proper justification is given (e.g., error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering [No] or [N/A] is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: In the abstract and the introduction, we give a detailed description of the claims and results we prove in the paper as a technical overview

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

Justification: The paper discusses the remaining unresolved regimes in Figure 1 and the accompanying text. The yellow entries mark combinations of query budget and regret target for which our results do not establish either a matching upper or lower bound. The paper also clarifies that the main complexity measure is the number of ERM-oracle calls, rather than total implementation runtime.

Guidelines:

- The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: In every theoretical statement, we characterize the assumptions and conditions of the statement and then we give a proof.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [N/A]

Justification: This is a completely theoretical paper so there are no experiments.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [N/A]

   Justification: This is a completely theoretical paper so there are no experiments.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.

- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [N/A]

   Justification: This is a completely theoretical paper so there are no experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [N/A]

   Justification: This is a completely theoretical paper so there are no experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [N/A]

   Justification: This is a completely theoretical paper so there are no experiments.

   Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research in this paper conforms to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: Since this is a learning theory paper focused on characterizing learnability and complexity of learning problems, we do not see any immediate negative social impact.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: This is a completely theoretical paper so there are no datasets or experimental models.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [N/A]

Justification: This is a completely theoretical paper so we don't have any code, data, or models.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: This is a completely theoretical paper so we don't have any code, data, or models.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: We do not conduct any crowdsourcing experiments or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: We do not conduct any crowdsourcing experiments or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The paper does not use any large language models as part of its research methodology or core methods.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.