Contextual Active Online Model Selection with Expert Advice

Xuefeng Liu XUEFENG@UCHICAGO.EDU
University of Chicago

Fangfang Xia FANGFANG@ANL.GOV
Argonne National Laboratory

Rick L. Stevens RSTEVENS@UCHICAGO.EDU
University of Chicago

Yuxin Chen CHENYUXIN@UCHICAGO.EDU
University of Chicago

Abstract

How can we collect the most useful labels to learn a model selection policy, when presented with arbitrary heterogeneous data streams? In this paper, we formulate this task as a *contextual active model selection* problem, where at each round the learner receives an unlabeled data point along with a context. The goal is to output the best model for any given context without obtaining an excessive amount of labels. In particular, we focus on the task of selecting pre-trained classifiers, and propose a contextual active model selection algorithm (CAMS), which relies on a novel uncertainty sampling query criterion defined on a given policy class for adaptive model selection. In comparison to prior art, our algorithm does not assume a globally optimal model. We provide rigorous theoretical analysis for the regret and query complexity under both adversarial and stochastic settings. Our experiments on several benchmark classification datasets demonstrate the algorithm's effectiveness in terms of both regret and query complexity. Notably, to achieve the same accuracy, CAMS incurs less than 10% of the label cost when compared to the best online model selection baselines on CIFAR10.

1. Introduction

With the rise of pre-trained models in many real-world machine learning tasks (e.g., BERT (Devlin et al., 2018), GPT-3 (Brown et al., 2020)), there is a growing demand for *label-efficient* approaches for model selection, especially when facing varying data distributions and contexts at run time. Often times, no single pre-trained model achieves the best performance for every context, and a proper approach is to identify an optimal policy for selecting *data-adaptive* models (Luo et al., 2020) for specific contexts (e.g. in airline ancillary pricing (Shukla et al., 2019), ecology (Cade, 2015), etc.). In many of these applications, collecting labels (e.g., querying the optimal pricing model for online pricing, evaluating conservation strategies in ecology) is expensive. Furthermore, one may not gain access to the pool of data instances all at once, but rather receive a stream of data points—possibly in an arbitrary order—as the learning and model selection process unfolds. This calls for cost-effective and robust online algorithms that can identify the best model selection policy under limited labeling resources.

Motivated by the above challenges, we focus on a novel *contextual active model selection* problem, where a learner adaptively selects among a collection pre-trained models when presented with a stream of unlabeled data points. Specifically, at each round, the learner receives a data point with a context (e.g., geographical location, user profile, environment conditions, etc) and decides whether to query its label. The goal is to output the best data-adaptive model without obtaining an excessive amount of labels. In contrast to existing work in *active learning* (Dagan and Engelson, 1995; Tosh and Dasgupta, 2018; Beygelzimer et al., 2009, 2011a), *contextua bandit* (Auer et al., 2002b; Beygelzimer et al., 2011b; Neu, 2015), *online learning with full information* (Freund and Schapire, 1997; Cesa-Bianchi and Lugosi, 2006; Shalev-Shwartz et al., 2011), and *model selection* (Foster et al., 2019; Zhang et al., 2020; Cutkosky et al., 2021), our work takes a unique stand by capturing the key challenges from these relevant domains in a unified framework (see Appendix A.1 for a detailed comparison). Our key contributions are highlighted below.

- We proposed CAMS—a novel contextual active online model selection framework—by leveraging the context of data to adaptively select the best models when presented with an *arbitrary* data steam. Inspired by Karimi et al. (2021) which aim to actively select a *single best model* under the context-free setting, our algorithm comprises two key novel technical components: (1) a contextual online model selection policy and (2) a novel active uncertainty sampling strategy.
- We provide rigorous theoretical analysis on the regret and query complexity of the proposed algorithm, and provide upper bounds on each term for both stochastic (§4) and adversarial (Appendix E) data streams.
- Empirically, we demonstrate the effectiveness of our approach on a variety of online model selection tasks spanning different application domains, task scales, data modalities and labels types. Our experiments show remarkable performance of CAMS: For the tasks evaluated, CAMS outperforms all competing baselines by a significant margin (to achieve the same level of prediction accuracy, CAMS incurs less than 10% of the label cost of the best competing baselines on CIFAR10 (10K examples), and 68% the cost on VERTEBRAL); furthermore, with our query strategy, we observe an improved query complexity when evaluated against prior art (Karimi et al., 2021) under the special case of context-free setting (see §5, Fig. 2).

2. Problem Statement

Notations. Let $\mathcal{X}$ be the input domain and $\mathcal{Y} := \{0, \dots, c-1\}$ be the set of c possible class labels for each input instance. Let $\mathcal{F} = \{f_1, \dots, f_k\}$ be a set of k pre-trained classifiers over $\mathcal{X} \times \mathcal{Y}$. A model selection policy $\pi : \mathcal{X} \to \Delta^{k-1}$ maps any input instance $\boldsymbol{x} \in \mathcal{X}$ to a distribution over the pre-trained classifiers $\mathcal{F}$, specifying the probability $\pi(\boldsymbol{x})$ of selecting each classifier under input $\boldsymbol{x}$. Here, Δ^{k-1} denotes the k-dimensional probability simplex. One can interpret a policy π as an "expert" that suggests which model to select for a given *context* $\boldsymbol{x}$. Let Π be a collection of model selection policies, and $\Pi^* := \Pi \cup \{\pi_1^{\text{const}}, \dots, \pi_k^{\text{const}}\}$ (where $\pi_j^{\text{const}}(\cdot) := \boldsymbol{e}_j$)[1] be the extended policy set including constant policies always suggest a fixed model. Unless otherwise specified, we assume Π is finite with $|\Pi| = n$, and $|\Pi^*| \leq n + k$.

1. $\boldsymbol{e}_j \in \Delta^{k-1}$ denotes the canonical basis vector with $e_j = 1$.

The contextual active model selection protocol. Assume that the learner knows the set of classifiers $\mathcal{F}$ as well as the set of model selection policies Π. At round t, the learner receives a data instance $\boldsymbol{x}_t \in \mathcal{X}$ as the context for the current round, and computes the predicted label $\hat{y}_{t,j} = f_j(\boldsymbol{x}_t)$ for each pre-trained classifier indexed by $j \in [k]$. Denote the vector of predicted labels by all k models by $\hat{\boldsymbol{y}}_t := [\hat{y}_{t,1}, \ldots, \hat{y}_{t,k}]^\top$. Based on previous observations, the learner identifies a model /classifier f_{j_t} and makes a prediction $\hat{y}_{t,j_t}$ for the instance $\boldsymbol{x}_t$. Meanwhile, the learner can obtain the true label y_t *only if* it decides to query $\boldsymbol{x}_t$. Upon observing y_t, the learner incurs a *query cost*, and receives a (full) loss vector $\boldsymbol{\ell}_t = \mathbb{I}_{\{\hat{\boldsymbol{y}}_t \neq y_t\}}$, where the jth entry $\ell_{t,j} := \mathbb{I}_{\{\hat{y}_{t,j} \neq y_t\}}$ corresponds to the 0-1 loss for model $j \in [k]$ at round t. The learner can then use the queried labels to adjust its model selection criterion for future rounds.

Performance metric. Note that if $\boldsymbol{x}_t$ is misclassified by the model j_t selected by learner at round t, i.e. $\hat{y}_{t,j_t} \neq y_t$, it will be counted towards the *cumulative loss* of the learner, regardless of the learner making a query. Otherwise, no loss will be incurred for that round. For a learning algorithm $\mathcal{A}$, its cumulative loss over T rounds is defined as $L_T^{\mathcal{A}} := \sum_{t=1}^T \ell_{t,j_t}$.

For stochastic data streams, we assume that each policy i recommends the *most probable* model[2] w.r.t. $\pi_i(\boldsymbol{x}_t)$ for context $\boldsymbol{x}_t$. We use $\mathrm{maxind}(\boldsymbol{w}) := \arg\max_{j:w_j \in \boldsymbol{w}} w_j$ to denote the index of the maximal-value entry[3] of $\boldsymbol{w}$. Since $(\boldsymbol{x}, y)$ are drawn i.i.d., we define $\mu_i = \frac{1}{T}\sum_{t=1}^T \mathbb{E}_{\boldsymbol{x}_t, y_t}\left[\ell_{t,\mathrm{maxind}(\pi_i(\boldsymbol{x}_t))}\right]$. This leads to the pseudo-regret for the stochastic setting over T rounds, defined as

$$\overline{\mathcal{R}}_T(\mathcal{A}) = \mathbb{E}[L_T^{\mathcal{A}}] - T \min_{i \in [|\Pi^*|]} \mu_i. \tag{1}$$

In an adversarial setting, since the data stream (and hence the loss vector) is determined by an adversary, we consider the reference best policy to be the one that minimizes the loss on the adversarial data stream, and the expected regret is defined as $\mathcal{R}_T(\mathcal{A}) = \mathbb{E}[L_T^{\mathcal{A}}] - \min_{i \in [|\Pi^*|]} \sum_{t=1}^T \tilde{\ell}_{t,i}$ (2), where $\tilde{\ell}_{t,i} := \langle \pi_i(\boldsymbol{x}_t), \boldsymbol{\ell}_t \rangle$ denotes the expected loss if the learner commits to policy π_i, randomizes and selects $j_t \sim \pi_i(\boldsymbol{x}_t)$ (and receives loss ℓ_{t,j_t}) at round t.

3. Contextual Active Model Selection with Expert Advice

Contextual model selection. Our key insight underlying the contextual model selection strategy extends from the *online learning with expert advice* framework (Freund and Schapire, 1997; Burtini et al., 2015). Pseudocode relevant to the model selection steps is provided in Line 4-8 in Fig. 1. At each round, CAMS maintains a probability distribution over the (extended) policy set Π^*, and updates those according to the observed loss for each policy. We use $\boldsymbol{q}_t := (q_{t,i})_{i \in |\Pi^*|}$ to denote the probability distribution over Π^* at t. Specifically, the probability $q_{t,i}$ is computed based on the exponentially weighted cumulative loss, i.e. $q_{t,i} \propto \exp\left(-\eta_t \tilde{L}_{t-1,i}\right)$ where $\tilde{L}_{t,i} := \sum_{\tau=1}^t \tilde{\ell}_{\tau,i}$ denotes the cumulative loss of policy i.

Under the stochastic setting, CAMS adopts a weighted majority strategy (Littlestone and Warmuth, 1994) when selecting models. The vector of weighted votes each model receives from the policies are computed as $\boldsymbol{w}_t = \sum_{i \in |\Pi^*|} q_{t,i} \pi_i(\boldsymbol{x}_t)$, which can be interpreted as a distribution induced by the weighted policy. Then, the most probable model $j_t = \mathrm{maxind}(\boldsymbol{w}_t)$

2. Our choice of the most probable selection strategy is based on superior empirical performance (see §5).
3. Assume ties are broken randomly.

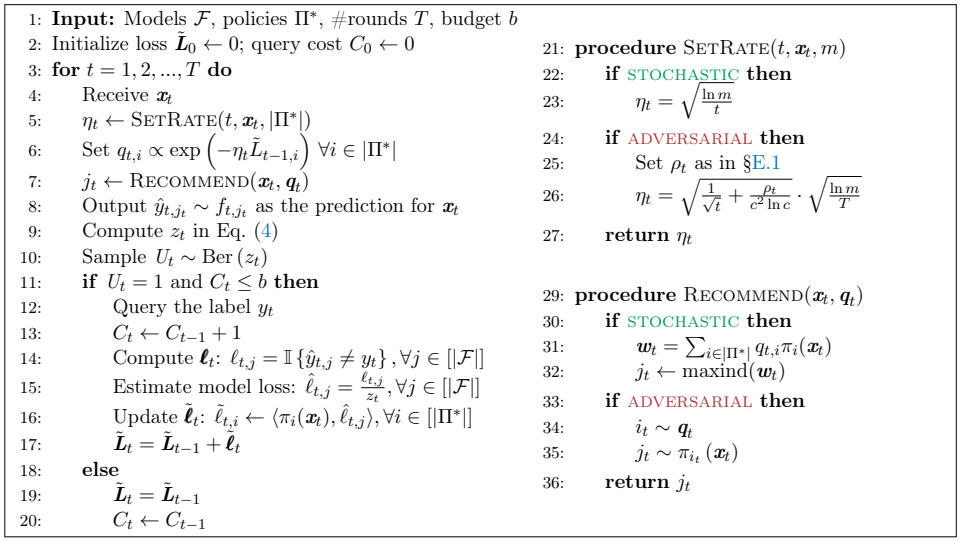


Figure 1: The CAMS Algorithm

is recommended as the target model at round t. This amounts to a deterministic model selection strategy as is commonly used in stochastic online optimization (Hazan, 2019). For adversarial data streams, it is natural for both the online learner and the model selection policies to randomize their actions to avoid linear regret (Hazan, 2019). Following this insight, CAMS randomly samples a policy $i_t \sim \boldsymbol{q}_t$, and—based on the current context $\boldsymbol{x}_t$—samples a classifier $j_t \sim \pi_{i_t}(\boldsymbol{x}_t)$ to recommend at round t.

Active queries. We intend to query the labels of those instances that exhibit significant disagreement among the pre-trained models $\mathcal{F}$. Given context $\boldsymbol{x}_t$, model predictions $\hat{\boldsymbol{y}}_t$ and model distribution $\boldsymbol{w}_t$, we denote by $\bar{\ell}_t^y := \langle \boldsymbol{w}_t, \mathbb{I}\{\hat{\boldsymbol{y}}_t \neq y\} \rangle$ as the expected loss if the true label is y. We characterize the model disagreement as $\mathfrak{E}(\hat{\boldsymbol{y}}_t, \boldsymbol{w}_t) := \frac{1}{c} \sum_{y \in \mathcal{Y}, \bar{\ell}_t^y \in (0,1)} \bar{\ell}_t^y \log_c \frac{1}{\bar{\ell}_t^y}$ (3). Intuitively, when $\bar{\ell}_t^y$ is close to 0 or 1, there is little disagreement among the models in labeling $\boldsymbol{x}_t$ as y, otherwise there is significant disagreement. Note that $\mathfrak{E}$ takes a similar algebraic form to the entropy function, although it does not inherit the information-theoretic interpretation. We consider an adaptive query probability[4]

$$z_t = \max\left\{\delta_0^t, \mathfrak{E}(\hat{\boldsymbol{y}}_t, \boldsymbol{w}_t)\right\}, \tag{4}$$

where $\delta_0^t = \frac{1}{\sqrt{t}} \in (0, 1]$ is an adaptive lower bound on the query probability to encourage exploration at an early stage. The query strategy is summarized in Line 9-13 in Fig. 1.

Model updates. Now define $U_t \sim \text{Ber}(z_t)$ as a binary query indicator that is sampled from a Bernoulli distribution parametrized by z_t. Upon querying the label y_t, one can calculate the loss for each model $f_j \in \mathcal{F}$ as $\ell_{t,j} = \mathbb{I}\{\hat{y}_{t,j} \neq y_t\}$. Since CAMS does not query all the i.i.d. examples, we introduce an unbiased loss estimator for the models, defined as $\hat{\ell}_{t,j} = \frac{\ell_{t,j}}{z_t} U_t$. The unbiased loss of policy $\pi_i \in \Pi^*$ can then be computed as $\tilde{\ell}_{t,i} = \langle \pi_i(\boldsymbol{x}_t), \hat{\ell}_{t,j} \rangle$. In the end, CAMS computes the (unbiased) cumulative loss of policy π_i as $\tilde{L}_{T,i} = \sum_{t=1}^T \tilde{\ell}_{t,i}$. Pseudocode for the model update steps is summarized in Line 14-20 in Fig. 1.

4. For convenience of discussion, we assume that those rounds where all policies in Π^* select the same models or all models $\mathcal{F}$ make the same predictions are removed as a precondition.

4. Theoretical Analysis

We now present the theoretical bounds on the regret and the query complexity of CAMS. Here, we focus on the stochastic setting, and defer the discussion of the adversarial setting to the Appendix E. Let $i^* = \arg\min_{i \in [|\Pi^*|]} \mu_i$ be the index of the best policy (μ_i denotes the expected loss of policy i, as defined in §2. Define $\Delta := \min_{i \neq i^*}(\mu_i - \mu_{i^*})$ as the minimal sub-optimality gap[5] in terms of the expected loss against the best policy i^*. Furthermore, let $\boldsymbol{w}_{i^*}^t := \pi_{i^*}(\boldsymbol{x}_t)$ be probability distribution over $\mathcal{F}$ induced by policy i^* at round t. We define $\gamma := \min_{\boldsymbol{x}_t}\{\max_{w_j \in \boldsymbol{w}_{i^*}^t} w_j - \max_{w_j \in \boldsymbol{w}_{i^*}^t, j \neq \text{maxind}(\boldsymbol{w}_{i^*}^t)} w_j\}$ (5) as the minimal probability gap between the most probable model and the rest (assuming no ties) induced by the best policy i^*. We bound the expected regret as follows.

Theorem 4.1. *(Regret) Consider the stochastic setting. With probability at least* $1-\delta$, CAMS *achieves expected pseudo regret (Eq. (1))* $\overline{\mathcal{R}}_T(\text{CAMS}) \leq \left(\frac{\ln \frac{|\Pi^*|}{\gamma} + \sqrt{\ln |\Pi^*| \cdot 2 \ln \frac{2}{\delta}}}{\sqrt{\ln |\Pi^*|}\Delta}\right)^2$.

The proof for Theorem 4.1 is deferred to Appendix D.1. The following theorem (proved in Appendix D.2) provides an upper bound on the query complexity in the stochastic setting.

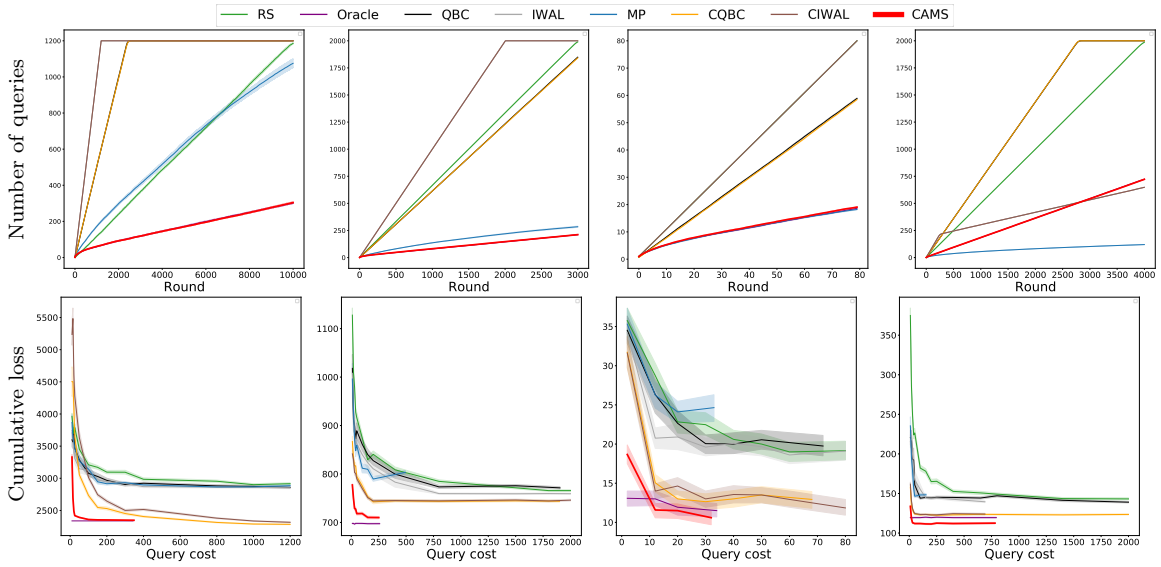
Theorem 4.2. *(Query Complexity, informal)*[6]*. For c-class classification problems, w.h.p. the expected number of queries made by* CAMS *over* T *rounds is* $\frac{\ln T}{c \ln c}\left(\left(\ln \frac{|\Pi^*|}{\gamma} / \left(\sqrt{|\Pi^*|}\Delta\right)\right)^2 + T\mu_{i^*}\right)$.

5. Experiments

Datasets, policies and pre-trained models. We consider 4 datasets: {CIFAR10 (Krizhevsky et al., 2009), DRIFT (Vergara et al., 2012), VERTEBRAL (Asuncion and Newman, 2007), HIV (Wu et al., 2018)}. We train a set of models on different subsamples from each dataset. Then we construct policies mixed with *malicious*, *normal*, *random*, and *biased* policy types for each dataset based on different models and features. In total, we create 80, 10, 6, 4 classifiers and 85, 11, 17, 20 policies for data-sets in list above respectively (see Appendix B for details).

Baselines. We use 4 *non-contextual* baselines: (1) Random Query (RS) queries the instance label with a fixed probability $\frac{b}{T}$; (2) Model Picker (MP) (Karimi et al., 2021) uses variance-based active sampling with coin-flip query probability; (3) Query by Committee (QBC) is committee-based sampling (Dagan and Engelson, 1995) with vote-entropy query probability; (4) Importance Weighted Active Learning (IWAL) (Beygelzimer et al., 2009) uses query probability calculated based on labeling disagreements of surviving classifiers. In addition, we consider 3 *contextual* baselines. Since no such algorithm is available yet, we create the contextual versions of QBC, IWAL as (5) CQBC, (6) CIWAL. Both extensions adopt their respective original query strategy but use an exponential-weight algorithm for model selection. For model selection, CAMS, MP, CQBC, and CIWAL recommend the classifier with the highest probability. The rest of the baselines use Follow-the-Leader, which greedily recommends the model with the minimum cumulative loss for past queried instances. Finally, we add (7) Oracle as the best policy with the same query strategy as CAMS.

5. w.l.o.g. assume there is a single best policy, and thus $\Delta > 0$.
6. Assume $T\mu_{i^*}$ is a constant value given by oracle, then the query-complexity bound is in sub-linear.



(a) CIFAR10 (b) DRIFT (c) VERTEBRAL (d) HIV

Figure 2: **(Top)** #queries vs #rounds. **(Bottom)** cumulative loss vs #queries, for a fixed number of rounds T (where $T = 10000, 3000, 80, 4000$ from left to right) with maximal query cost B (where $B = 1200, 2000, 80, 2000$ from left to right). Shades correspond to 90% confident interval.

Experimental results. *Query complexity:* A sub-linear and low increase in query cost indicate that the learner is actively (not passive or greedily) querying. Notably, by comparing to variance-based strategy (Karimi et al., 2021) and evaluating based on the same model selection strategy, Appendix C.2 indicates that CAMS's query strategy requests 6%, 14%, and 71% fewer queries for VERTEBRAL, DRIFT, and CIFAR10 datasets, respectively, while achieving even less cumulative loss. Fig. 2 **(Top)** demonstrate the compelling effectiveness of CAMS's query strategy outperforming all baselines (excluding Oracle) in terms of query cost in VERTEBRAL, DRIFT, and CIFAR10 benchmarks, which is consistent with our theoretical result in Theorem 4.2. *Cost effectiveness:* Fig. 2 **(Bottom)** illustrates the cost effectiveness (as the rate of change in cumulative loss compared to query cost changes) of each algorithm. CAMS outperforms all baselines (other than Oracle) across all datasets by querying fewer labels. CAMS not only achieves the lowest cumulative loss but also has the sharpest cumulative loss decreasing rate to converge to the optimal status by only increasing a few query cost on all benchmarks. Moreover, it takes CAMS fewer than 10 and 20 queries, respectively, to outperform Oracle on VERTEBRAL and HIV benchmarks. In particular, on the VERTEBRAL benchmark, CAMS has a 20% margin over the best baseline in query cost, and it achieves this despite 11 of the 17 experts giving malicious or random advice.

6. Conclusion

We introduced CAMS, a contextual active online model selection framework based on a novel model selection and active query strategy. We have provided rigorous theoretical guarantees on the regret and query complexity for both stochastic and adversarial settings, as well as extensive empirical study on several online model selection tasks. Our results show remarkable performance of CAMS on a diverse range of datasets.

Acknowledgements

We thank Varun Gupta and Rad Niazadeh for the helpful discussions; we also thank Renyu Zhang and the anonymous reviewers for their valuable feedback. This work is supported in part by the RadBio-AI project (DE-AC02-06CH11357), U.S. Department of Energy Office of Science, Office of Biological and Environment Research; the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration.

References

Alnur Ali, Rich Caruana, and Ashish Kapoor. Active learning with model selection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 28, 2014.

Arthur Asuncion and David Newman. Uci machine learning repository, 2007.

Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2):235–256, 2002a.

Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1):48–77, 2002b.

Alina Beygelzimer, Sanjoy Dasgupta, and John Langford. Importance weighted active learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 49–56, 2009.

Alina Beygelzimer, Daniel Hsu, Nikos Karampatziakis, John Langford, and Tong Zhang. Efficient active learning. In *ICML 2011 Workshop on On-line Trading of Exploration and Exploitation*. Citeseer, 2011a.

Alina Beygelzimer, John Langford, Lihong Li, Lev Reyzin, and Robert Schapire. Contextual bandit algorithms with supervised learning guarantees. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 19–26. JMLR Workshop and Conference Proceedings, 2011b.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Sébastien Bubeck, Nicolo Cesa-Bianchi, et al. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5(1):1–122, 2012.

Giuseppe Burtini, Jason Loeppky, and Ramon Lawrence. A survey of online experiment design with the stochastic multi-armed bandit. *arXiv preprint arXiv:1510.00757*, 2015.

Brian S Cade. Model averaging and muddled multimodel inferences. *Ecology*, 96(9):2370–2382, 2015.

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

Nicolo Cesa-Bianchi, Yoav Freund, David Haussler, David P Helmbold, Robert E Schapire, and Manfred K Warmuth. How to use expert advice. *Journal of the ACM (JACM)*, 44(3): 427–485, 1997.

Nicolo Cesa-Bianchi, Gábor Lugosi, and Gilles Stoltz. Minimizing regret with label efficient prediction. *IEEE Transactions on Information Theory*, 51(6):2152–2162, 2005.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3): 273–297, 1995.

Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27, 1967.

Jan Salomon Cramer. The origins of logistic regression. 2002.

Ashok Cutkosky, Christoph Dann, Abhimanyu Das, Claudio Gentile, Aldo Pacchiano, and Manish Purohit. Dynamic balancing for model selection in bandits and rl. In *International Conference on Machine Learning*, pages 2276–2285. PMLR, 2021.

Ido Dagan and Sean P Engelson. Committee-based sampling for training probabilistic classifiers. In *Machine Learning Proceedings 1995*, pages 150–157. Elsevier, 1995.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Joseph L Durant, Burton A Leland, Douglas R Henry, and James G Nourse. Reoptimization of mdl keys for use in drug discovery. *Journal of chemical information and computer sciences*, 42(6):1273–1280, 2002.

Ronald A Fisher. The statistical utilization of multiple measurements. *Annals of eugenics*, 8 (4):376–386, 1938.

Dylan J Foster, Akshay Krishnamurthy, and Haipeng Luo. Model selection for contextual bandits. *arXiv preprint arXiv:1906.00531*, 2019.

Yoav Freund and Robert E Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139, 1997.

Yoav Freund, Robert Schapire, and Naoki Abe. A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, 14(771-780):1612, 1999.

Jacob R Gardner, Gustavo Malkomes, Roman Garnett, Kilian Q Weinberger, Dennis Barbour, and John P Cunningham. Bayesian active model selection with an application to automated audiometry. In *Proceedings of the 28th International Conference on Neural Information Processing Systems-Volume 2*, pages 2386–2394, 2015.

Pierre Geurts, Damien Ernst, and Louis Wehenkel. Extremely randomized trees. *Machine learning*, 63(1):3–42, 2006.

David J Hand and Keming Yu. Idiot's bayes—not so stupid after all? *International statistical review*, 69(3):385–398, 2001.

Elad Hazan. Introduction to online convex optimization. *arXiv preprint arXiv:1909.05207*, 2019.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Shen-Shyang Ho and Harry Wechsler. Query by transduction. *IEEE transactions on pattern analysis and machine intelligence*, 30(9):1557–1571, 2008.

Steven CH Hoi, Doyen Sahoo, Jing Lu, and Peilin Zhao. Online learning: A comprehensive survey. *Neurocomputing*, 459:249–289, 2021.

Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

Hanzhang Hu, Wen Sun, Arun Venkatraman, Martial Hebert, and Andrew Bagnell. Gradient boosting on stochastic data streams. In *Artificial Intelligence and Statistics*, pages 595–603. PMLR, 2017.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

Mohammad Reza Karimi, Nezihe Merve Gürel, Bojan Karlaš, Johannes Rausch, Ce Zhang, and Andreas Krause. Online active model selection for pre-trained classifiers. In *International Conference on Artificial Intelligence and Statistics*, pages 307–315. PMLR, 2021.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

Rui Leite and Pavel Brazdil. Active testing strategy to predict the best classification algorithm via sampling and metalearning. In *ECAI*, pages 309–314, 2010.

Nick Littlestone and Manfred K Warmuth. The weighted majority algorithm. *Information and computation*, 108(2):212–261, 1994.

Chen Change Loy, Timothy M Hospedales, Tao Xiang, and Shaogang Gong. Stream-based joint exploration-exploitation active learning. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1560–1567. IEEE, 2012.

Mi Luo, Fei Chen, Pengxiang Cheng, Zhenhua Dong, Xiuqiang He, Jiashi Feng, and Zhenguo Li. Metaselector: Meta-learning for recommendation with user-level adaptive model selection. In *Proceedings of The Web Conference 2020*, pages 2507–2513, 2020.

Omid Madani, Daniel J Lizotte, and Russell Greiner. Active model selection. *arXiv preprint arXiv:1207.4138*, 2012.

Jaouad Mourtada and Stéphane Gaïffas. On the optimality of the hedge algorithm in the stochastic regime. *Journal of Machine Learning Research*, 20:1–28, 2019.

Mohamad T Musavi, Wahid Ahmed, Khue Hiang Chan, Kathleen B Faris, and Donald M Hummels. On the training of radial basis function classifiers. *Neural networks*, 5(4): 595–603, 1992.

Gergely Neu. Explore no more: Improved high-probability regret bounds for non-stochastic bandits. *arXiv preprint arXiv:1506.03271*, 2015.

Nikunj C Oza and Stuart J Russell. Online bagging and boosting. In *International Workshop on Artificial Intelligence and Statistics*, pages 229–236. PMLR, 2001.

J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986.

Carl Edward Rasmussen. Gaussian processes in machine learning. In *Summer school on machine learning*, pages 63–71. Springer, 2003.

Ryan M Rifkin and Ross A Lippert. Notes on regularized least squares. 2007.

David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of chemical information and modeling*, 50(5):742–754, 2010.

Marlesson RO Santana, Luckeciano C Melo, Fernando HF Camargo, Bruno Brandão, Anderson Soares, Renan M Oliveira, and Sandor Caetano. Contextual meta-bandit for recommender systems selection. In *Fourteenth ACM Conference on Recommender Systems*, pages 444–449, 2020.

Christoph Sawade, Niels Landwehr, Steffen Bickel, and Tobias Scheffer. Active risk estimation. In *ICML*, 2010.

Christoph Sawade, Niels Landwehr, and Tobias Scheffer. Active comparison of prediction models. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

Nick Schneider, Florian Piewak, Christoph Stiller, and Uwe Franke. Regnet: Multimodal sensor registration using deep neural networks. In *2017 IEEE intelligent vehicles symposium (IV)*, pages 1803–1810. IEEE, 2017.

Yevgeny Seldin and Gábor Lugosi. A lower bound for multi-armed bandits with expert advice. In *13th European Workshop on Reinforcement Learning (EWRL)*, 2016.

Burr Settles. Active learning literature survey. 2009.

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

Shai Shalev-Shwartz et al. Online learning and online convex optimization. *Foundations and trends in Machine Learning*, 4(2):107–194, 2011.

Xiaotong Shen and Hsin-Cheng Huang. Optimal model assessment, selection, and combination. *Journal of the American Statistical Association*, 101(474):554–568, 2006.

Naman Shukla, Arinbjörn Kolbeinsson, Lavanya Marla, and Kartik Yellepeddi. Adaptive model selection framework: An application to airline pricing. *arXiv preprint arXiv:1905.08874*, 2019.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Masashi Sugiyama and Neil Rubens. A batch ensemble approach to active learning with model selection. *Neural Networks*, 21(9):1278–1286, 2008.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

Christopher Tosh and Sanjoy Dasgupta. Interactive structure learning with structural query-by-committee. *Advances in Neural Information Processing Systems*, 31, 2018.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

Alexander Vergara, Shankar Vembu, Tuba Ayhan, Margaret A Ryan, Margie L Homer, and Ramón Huerta. Chemical gas sensor drift compensation using classifier ensembles. *Sensors and Actuators B: Chemical*, 166:320–329, 2012.

Zhenqin Wu, Bharath Ramsundar, Evan N Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S Pappu, Karl Leswing, and Vijay Pande. Moleculenet: a benchmark for molecular machine learning. *Chemical science*, 9(2):513–530, 2018.

Zhaoping Xiong, Dingyan Wang, Xiaohong Liu, Feisheng Zhong, Xiaozhe Wan, Xutong Li, Zhaojun Li, Xiaomin Luo, Kaixian Chen, Hualiang Jiang, et al. Pushing the boundaries of molecular representation for drug discovery with the graph attention mechanism. *Journal of medicinal chemistry*, 63(16):8749–8760, 2019.

Chicheng Zhang and Kamalika Chaudhuri. Beyond disagreement-based agnostic active learning. *Advances in Neural Information Processing Systems*, 27:442–450, 2014.

Yao Zhang, Daniel Jarrett, and Mihaela van der Schaar. Stepwise model selection for sequence prediction via deep kernel learning. *arXiv preprint arXiv:2001.03898*, 2020.

Appendix A. Comparison against Related Work

A.1 Summary of related work

Contextual bandits. Classical bandit algorithms—such as EXP3 (Auer et al., 2002b) or UCB (Auer et al., 2002a)— aim to find the best action(s) achieving the minimal regret through a sequence of actions. When side information (e.g. user profile for recommender systems or environmental context for experimental design) is available at each round, many bandit algorithms can be lifted to the contextual setting: For example, EXP4 and its variants (Auer et al., 2002b; Beygelzimer et al., 2011b; Neu, 2015) consider the bandit setting with expert advice: At each round, experts announce their predictions of which actions are the most promising for the given context, and the goal is to construct a expect selection policy that competes with the best expert from hindsight. Note that in bandit problems, the learner only gets to observe the reward for each action taken. In contrast, for the online model selection problem considered in this work—where an action corresponds to choosing a model to make prediction on an incoming data point—we get to see the loss/reward of *all* models on the labeled data point. In this regard, our work aligns more closely with online learning with *full information* setting, where the learner has access to the loss of all the arms at each round (e.g. as considered by the Hedge algorithm (Freund and Schapire, 1997; Burtini et al., 2015; Cesa-Bianchi and Lugosi, 2006; Hoi et al., 2021)).

Online learning with full information. A clear distinction between our work and online learning is that we assume the labels of the online data stream are not readily available but can be acquired at each round with a cost. In addition, the learner only observes the loss incurred by all models on a data point when it decides to query its label. In contrast, in the canonical online learning setting, labels arrive with the data and one gets to observe the loss of all candidate models at each round. Similar setting also applies to other online learning problems, such as online boosting or bagging[7] (Oza and Russell, 2001; Hu et al., 2017). A related work to ours is online learning with label-efficient prediction (Cesa-Bianchi et al., 2005), which proposes an online learning algorithm with matching upper and lower bounds on the regret. However, they consider a fixed query probability that leads to a linear query complexity. Our algorithm, inspired by uncertainty sampling in active learning, achieves an improved query complexity with the adaptive query strategy while maintaining a comparable regret.

Active learning. The goal of active learning is to achieve a target learning performance with fewer training examples (Settles, 2009). In the context of active model selection, we aim to collect the most useful labels to differentiate the candidate models while maintaining a low query cost. The active learning framework closest to our setting is query-by-committee (QBC) (Seung et al., 1992), in particular under the stream-based setting (Loy et al., 2012; Ho and Wechsler, 2008). QBC maintains a committee of hypotheses; each committee member votes on the label of an instance, and the instances with the maximal disagreement among the committee are considered the most informative labels. Note that existing stream-based QBC algorithms are designed and analyzed assuming i.i.d. data streams. In comparison,

7. Additionally, these online ensemble learning problems also differ from online (contextual) model selection in that they aim to build a composite model by aggregating the strength of different models (Shen and Huang, 2006), rather than selecting the best model (for a given context).

our work uses a different query strategy as well as a novel model recommendation strategy, which also applies to the adversarial setting.

Active model selection. Active model selection captures a broad class of problems where model evaluations are expensive, either due to (1) the cost of evaluating (or "probing") a model, or (2) the cost of annotating a training example. Existing works under the former setting (Madani et al., 2012; Cutkosky et al., 2021; Shukla et al., 2019; Santana et al., 2020) often ignore context information and data annotation cost, and only consider *partial* feedback on the models being evaluated/ probed on i.i.d. data. The goal is to identify the best model with as few model probes as possible. For example, Cutkosky et al. (2021) propose a model selection framework, which balances the regret among a set of well-specified active learners, and Shukla et al. (2019) propose a (context-free) Thompson-sampling-based adaptive model selection framework for selecting the next model(s) to evaluate. This is quite different from our problem setting which considers the full information setting as well as non-negligible data annotation cost. For the later, most existing works assume a pool-based setting where the learner can choose among the pool of unlabeled data (Sugiyama and Rubens, 2008; Madani et al., 2012; Sawade et al., 2012, 2010; Ali et al., 2014; Gardner et al., 2015; Zhang and Chaudhuri, 2014; Leite and Brazdil, 2010), and the goal is to identify the best model with a minimal set of labels. Recently, Karimi et al. (2021) investigate active model selection under the stream-based setting, which aims to select a single best model for arbitrary data steams. This is closely related to our work; the key difference being the prior work does not model context information which could be vital for heterogeneous data streams.

A.2 Comparison against related work: Problem setup

For better positioning of this work, we compare our setting against a few related works in this domain, and highlight the key differences in the problem setup in Table 1.

Algorithm	**Online bagging** (Oza and Russell, 2001)	**Hedge** (Freund and Schapire, 1997)	**EXP3** (Auer et al., 2002b)	**EXP4** (Auer et al., 2002b)	**QBC** (Seung et al., 1992)	**ModelPicker** (Karimi et al., 2021)	**CAMS** (ours)
criterion	combination	selection	selection	selection	combination	selection	selection
full-information	yes	yes	no	no	yes	yes	yes
active	no	no	no	no	yes	yes	yes
contextual	no	no	no	yes	no	no	yes

Table 1: Algorithm comparison: problem setup

A.3 Comparison against related work: Theoretical guarantees

We summarize our key theoretical contributions of regret and query complexity bound [8] together with the bounds of other related algorithms in the table of this section.

Algorithm	**Regret**	**Query Complexity**
Exp3 (Lattimore and Szepesvári, 2020)	$2\sqrt{Tk\log k}$	–
Exp3.p (Bubeck et al., 2012)	$5.15\sqrt{nT\log\frac{n}{\delta}}$	–
Exp4 (Lattimore and Szepesvári, 2020)	$\sqrt{2Tk\log n}$	–
Exp4.p (Beygelzimer et al., 2011b)	$6\sqrt{kT\ln\frac{n}{\delta}}$	–
Model Picker$_{\text{stochastic}}$ (Karimi et al., 2021)	$62\max_i\Delta_i k/\left(\lambda^2\log k\right)$ $\lambda=\min_{j\in[k]\setminus\{i^*\}}\Delta_j^2/\theta_j$	$\sqrt{2T\log k}(1+4\frac{c}{\Delta})$
Model Picker$_{\text{adversarial}}$ (Karimi et al., 2021)	$2\sqrt{2T\log k}$	$5\sqrt{T\log k}+2L_{T,*}$
CAMS$_{\text{stochastic}}$ (§D.1, §D.2)	$\left(\left(\ln\frac{\lvert\Pi^*\rvert}{\gamma}+\sqrt{\ln\lvert\Pi^*\rvert\cdot 2\ln\frac{2}{\delta}}\right)/\left(\sqrt{\ln\lvert\Pi^*\rvert}\Delta\right)\right)^2$	$O\left(\frac{\ln T}{c\ln c}\left(\frac{\ln T}{\Delta^2}+\left(\frac{\ln\frac{\lvert\Pi^*\rvert}{\gamma}}{\sqrt{\ln\lvert\Pi^*\rvert}\Delta}\right)^2+T\mu_{i^*}\right)\right)$
CAMS$_{\text{adversarial}}$ (§E.2, §E.3)	$2c\sqrt{\ln c/\rho_T}\cdot\sqrt{T\log\lvert\Pi^*\rvert}$	$O\left(\frac{\ln T}{c\ln c}\left(\sqrt{\frac{T\log(\lvert\Pi^*\rvert)}{\rho_T}}+\tilde{L}_{T,*}\right)\right)$

Table 2: Regret and query complexity bound

8. Let us assume $T\mu_{i^*}, \tilde{L}_{T,*}$ is a constant value given by oracle in stochastic and adversarial setting respectively, then we consider query-complexity bound is sub-linear.
i^* is the model with the highest expected accuracy.
$\theta_j=\mathbb{P}\left[\ell_{\cdot,j}\neq\ell_{\cdot,i^*}\right]$ to be the probability that exactly one of j and i^* correctly classify a sample.
γ is defined in (5).
ρ_T is defined in (11).

Appendix B. Supplemental Materials on Experimental Setup

B.1 Details on datasets

CIFAR10: The CIFAR10 dataset contains 60,000 32x32 color images in 10 different classes. Each class has 6000 images. We randomly split the dataset into three subsets: the training set contains 45000 images, the validation set contains 5000 images, and we keep 10000 images for the online testing set.

DRIFT: The DRIFT dataset measures gas using data from chemical sensors exposed to different gases at various concentration levels. DRIFT contains 13910 measurements from 6 types of gases and 16 chemical sensors, forming a 128-dimensional feature vector. We randomly split the dataset into a training, validation, and test set with 9737, 1113 and 3060 records, respectively.

VERTEBRAL: The VERTEBRAL dataset is a biomedical dataset that classifies patients into three categories: Normal (100 patients), Spondylolisthesis (150 patients), or Disk Hernia (60 patients). Each patient is represented by six biomechanical attributes: pelvic incidence, pelvic tilt, lumbar lordosis angle, sacral slope, pelvic radius, and spondylolisthesis grade. VERTEBRAL contains 310 instances. We randomly selected 155 instances as the training set, 25 as validation sets, and 130 as testing sets.

HIV: The HIV dataset measures the ability to inhibit HIV replication for over 40,000 compounds with binary labels (active and inactive). We randomly draw 32901 records as the training set, 4113 records as a validation set, and 4113 records as the test set.

B.2 Details on policies and classifiers

Policy: At each round, the *malicious* policy gives opposite advice; the *random* policy gives random advice; the *biased* policy gives biased advice through training on a biased distribution over classifying specific classes. The *normal* policy gives reasonable advice by training under a standard process of the training set.

CIFAR10: We have constructed 80 diversified classifiers based on VGG (Simonyan and Zisserman, 2014), ResNet (He et al., 2016), DenseNet (Huang et al., 2017), GoogLeNet (Szegedy et al., 2015). We have also used EfficientNet (Tan and Le, 2019), MobileNets (Howard et al., 2017), RegNet (Schneider et al., 2017), and ResNet to construct 85 diversified policies.

DRIFT: We have constructed ten classifiers using Decision Tree (Quinlan, 1986), SVM (Cortes and Vapnik, 1995), AdaBoost (Freund et al., 1999), Logistic Regression (Cramer, 2002), KNN (Cover and Hart, 1967) models. We have also created 8 diversified policies with multilayer perceptron (MLP) models of different layer configurations: (128, 30, 10); (128, 60, 30, 10); (128, 120, 30, 10); (128, 240, 120, 30, 10).

VERTEBRAL: We have built six classifiers using Random Forest (Breiman, 2001), Gaussian Process (Rasmussen, 2003), linear discriminant analysis (Fisher, 1938), Naive Bayes (Hand and Yu, 2001) algorithms. We have constructed policies by using standard scikit-learn built-in models including Random Forest Classifier, Extra Trees Classifier (Geurts et al.,

2006), Decision Tree Classifier, Radius Neighbors Classifier (Musavi et al., 1992), Ridge Classifier (Rifkin and Lippert, 2007) and K-Nearest-Neighbor classifiers.

HIV: We have used graph convolutional networks (GCN) (Kipf and Welling, 2016), Graph Attention Networks (GAT) (Veličković et al., 2017), AttentiveFP (Xiong et al., 2019), and Random Forest to construct 4 classifiers. We have also used various feature representations of molecules such as MACCS key (Durant et al., 2002), ECFP2, ECFP4, and ECFP6 (Rogers and Hahn, 2010) molecular fingerprints to build 6 MLP-based policies, respectively.

B.3 Implementation details

For the experiments, we build our contextual online active learning platform on top of prior non-contextual work (Karimi et al., 2021) around the four benchmark datasets. The context $\boldsymbol{x}_t$ is the raw context of the data (e.g., the 32x32 image for CIFAR10). The predictions $\hat{\boldsymbol{y}}_t$ contain the predicted label vector of all the classifiers' predictions according to the online context x_t. The oracle file contains the true label y_t of $\boldsymbol{x}_t$. The advice matrix file contains the matrix data and each row represents a matrix of all policies' probability distribution λ over all the classifiers on context x_t. To adapt to an online setting, we sequentially draw random T i.i.d. instances from the testing pool set and define it as a realization. For a fair comparison, all algorithms receive data instances in the same order within the same realization.

B.4 Regularized policy

As discussed in §E.1, we wish to ensure that the probability a policy selecting any model is bounded away from 0 so that the regret bound in Theorem E.1 is non vacuous. In our experiments, we achieve this goal by applying a regularized policy $\overline{\pi}$ as shown in Algorithm 1.

Algorithm 1 Regularized policy $\overline{\pi}(\boldsymbol{x}_t)$

1: **Input:** context $\boldsymbol{x}_t$, Models $\mathcal{F}$, policy $\pi \in \Pi^*$
2: $\gamma = \sum_{j=1}^{|\mathcal{F}|} \left([\pi(x_t)]_j - \frac{1}{|\mathcal{F}|} \right)^2$
3: **return** $\frac{\pi_i(x_t)+\gamma}{1+|\mathcal{F}|\cdot\gamma}$

B.5 Summary of datasets and models

We summarize the attributes of datasets, the models, and the model selection policies as follows.

dataset	classification	total instances	test set	stream size	classifier	policy
CIFAR10	10	60000	10000	10000	80	85
DRIFT	6	13910	3060	3000	10	11
VERTEBRAL	3	310	127	80	6	17
HIV	2	40000	4113	4000	4	20

Table 3: Attributes of benchmark datasets

B.6 Details on baseline algorithms

Model Picker (MP) Model Picker (Karimi et al., 2021) is a context-free online active model selection method inspired by EXP3. Model Picker aims to find the best classifier in hindsight while making a small number of queries. For query strategy, it uses a variance-based active learning sampling method to select the most informative label to query to differentiate a pool of models, where the variance is defined as $v\left(\hat{\boldsymbol{y}}_t, \boldsymbol{w}_t\right) = \max_{y \in \mathcal{Y}} \bar{\ell}_t^y \left(1 - \bar{\ell}_t^y\right)$. The coin-flip query probability is defined as $\max\left\{v\left(\hat{\boldsymbol{y}}_t, \boldsymbol{w}_t\right), \eta_t\right\}$ when $v\left(\hat{\boldsymbol{y}}_t, \boldsymbol{w}_t\right) \neq 0$, or 0 otherwise. For model recommendation, it uses an exponential weight algorithm to recommend the model with minimal exponential cumulative loss based on the past queried labels at each round.

Query by Committee (QBC) For query strategy, we have adapted the method of (Dagan and Engelson, 1995) as a disagreement-based selective sampling query strategy for online streaming data. We treat each classifier as a committee member and compute the query probability by measuring disagreement between models for each instance. The query function is coin-flip by vote entropy probability $-\frac{1}{\log \min(k, |C|)} \sum_c \frac{V(c,x)}{k} \log \frac{V(c,x)}{k}$, where $V(c, x)$ stands for the number of committee members assigning a class c for input context x and k is the number of committee. For the model recommendation part, we use the method of Follow-the-Leader (FTL) (Lattimore and Szepesvári, 2020), which greedily recommends the model with the minimum cumulative loss for past queried instances.

Importance Weighted Active Learning (IWAL) We have implemented (Beygelzimer et al., 2009) as the IWAL baseline. For the query strategy part, IWAL computes an adaptive rejection threshold for each instance and assigns an importance weight to each classifier in the hypothesis space $\mathcal{H}_t$. IWAL retains the classifiers in the hypothesis space according to their weighted error versus the current best classifier's weighted error at round t. The query probability is calculated based on labeling disagreements of surviving classifiers through function $\max_{i,j \in \mathcal{H}_t, y \in [c]} \ell_{t,i}^{(y)} - \ell_{t,j}^{(y)}$. For model recommendation, we also adopt the Follow-the-Leader (FTL) strategy.

Random Query Strategy (RS) The RS method queries the label of incoming instances by the coin-flip fixed probability $\frac{b}{T}$. It also uses the FTL strategy based on queried instances for model recommendation.

Contextual Query by Committee (CQBC) We have created a contextual variant of QBC termed CQBC, which has the same entropy query strategy as the original QBC. For model recommendation, we combine two model selection strategies. The first strategy calculates the cumulative reward of each classifier based on past queries and normalizes it as a probability simplex vector. We also adopt Exp4's arm recommending vector to use contextual information. Finally, we compute the element-wise product of the two vectors and normalize it to be CQBC's model recommendation vector. At each round, CQBC would recommend the top model based on the classifiers' historical performance on queried instances and the online advice matrix for streaming data.

Contextual Importance Weighted Active Learning (CIWAL) We have created a variant version of importance-weighted active learning. Similar to CQBC, CIWAL adopts the query strategy from IWAL and converts the model selection strategy to be contextual.

For model selection, we incorporate Exp4's arm recommendation strategy based on the side-information advice matrix and each classifier's historical performance according to queried instances. We compute the element-wise product of the two vectors as the model selection vector of CIWAL and normalize it as a weighted vector. Finally, CIWAL recommends the classifier with the highest weight.

Oracle: Among all the given policies, oracle represents the best single policy that achieves the minimum cumulative loss, and it has the same query strategy as CAMS.

B.7 Hyperparameters

We performed our experiments on a Linux server with 80 Intel(R) Xeon(R) Gold 6148 CPU @ 2.40GHz and total 528 Gigabyte memory.

By considering the resource of server, We set 100 realizations and 3000 stream-size for DRIFT, 20 realizations and 10000 stream-size for CIFAR10, 200 realizations and 4000 stream size for HIV, 300 realization and 80 stream-size for VERTEBRAL. In each realization, we randomly selected steam-size aligned data from testing-set and make it as online streaming data which is the input of each algorithm. Thus, we got independent result for each realization.

A small realization number would increase the variance of the results due to the randomness of stream order. A large realization number would make the result be more stable but at the cost of increasing computational cost (time, memory, etc.). We chose the realization number by balancing both aspects.

Appendix C. Additional Experiments

In addition to the main results reported in Fig. 2, our empirical results demonstrate the remarkable performance of CAMS as follows:

(1) In a *mixture of experts* environment, CAMS converge to the best policy and outperform all others (Appendix C.3).

(2) In a *complete malicious experts* environment, CAMS can efficiently recover from malicious advice and approach the performance of the best classifier (Appendix C.5). CAMS has its guarantee to outperform any algorithms chasing a global optimal classifier (model) (Appendix C.1).

(3) In a *non-contextual* (no experts) environment, CAMS has approximately equal performance as Model Picker to reach the best classifier effectively (Appendix C.4).

(4) CAMS achieves the notable performances while making limited queries and maintains its robustness on the low-data benchmark (Appendix C.1, Fig. 2).

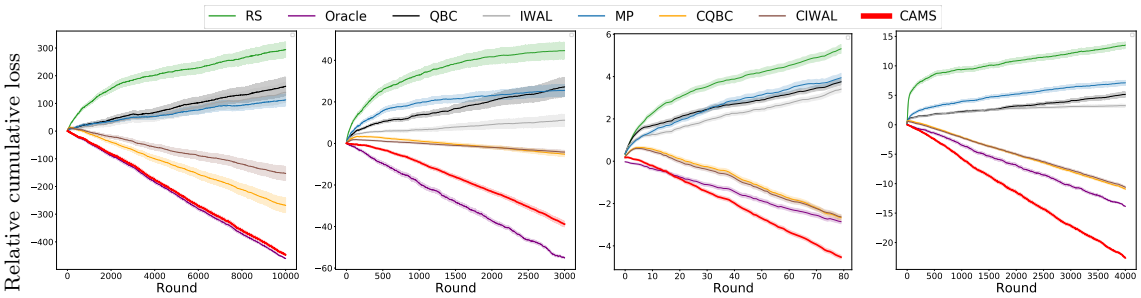
(5) CAMS has low variance for all benchmarks, which is desirable in sequential decision making (Appendix C.1, Fig. 2).

(6) In a *complete sub-optimal expert* environment, a variant of the CAMS algorithm, namely CAMS-MAX, which deterministically picks the most probable policy and selects the most probable model, outperforms CAMS-Random-Policy, which randomly samples a policy and selects the most probable model (Appendix C.7 & Appendix C.8). However, CAMS-MAX at most approaches the performance of the best policy. In contrast, perhaps surprisingly, CAMS is able to outperform the best policy in both VERTEBRAL and HIV

benchmarks (Appendix C.6).

C.1 Relative cumulative loss

To demonstrate that CAMS could outperform any algorithms chasing a global optimal classifier (model), we use the relative cumulative loss to compare the algorithm's cumulative loss with the best classifier. At round t, RCL is defined as $L_{t,j_i} - L_{t,j^*}$, where L_{t,j^*} stands for the cumulative loss (CL) of the policy always selecting the best classifier, and L_{t,j_i} stands for the CL of any policy i. The RCL under the same query cost for all baselines is shown in Fig. 3. The loss trajectory demonstrates that CAMS efficiently adapts to the best policy after only a few rounds and outperforms all baselines (excluding the Oracle performance on CIFAR10 and DRIFT) in all experiments. The result also demonstrates that CAMS can achieve negative RCL on all benchmarks, which means it outperforms any algorithms that chase the best classifier. This empirical result aligns with our Theorem 4.1 that, in the worst scenario, if the best classifier is the best policy, CAMS will achieve its performance. Otherwise, CAMS will outperform and reach a better policy under the no regret guarantee.



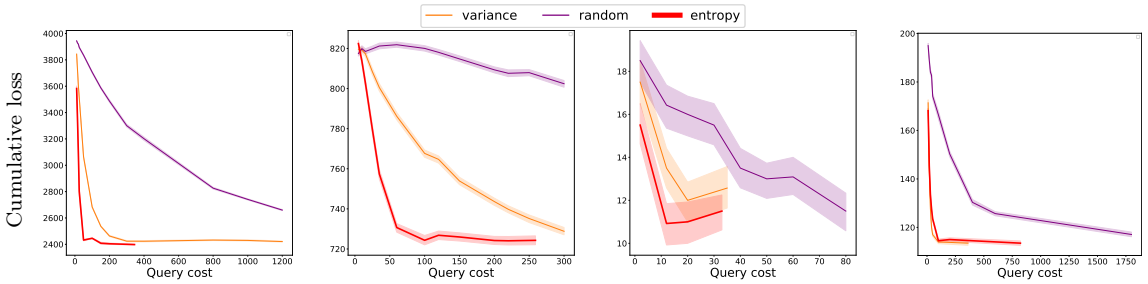
(a) CIFAR10 (b) DRIFT (c) VERTEBRAL (d) HIV

Figure 3: Comparing CAMS with 7 model selection baselines on 4 diverse benchmarks in terms of loss trajectory. CAMS outperforms all baselines (excluding Oracle). Performance measured by relative cumulative loss (i.e. loss against the best classifier) under a fixed query cost B (where $B = 200, 400, 30, 400$ from left to right). **Algorithms**: 4 contextual {Oracle, CQBC, CIWAL, CAMS} and 4 non-contextual baselines {RS, QBC, IWAL, MP} are included (see Section). 90% confident interval are indicated in shades.

C.2 Query strategies ablation comparison

Using the same CAMS model recommendation section, we compare three query strategies: the adaptive model-disagreement-based query strategy in Line 9-13 of Fig. 1 (referred to as *entropy* in the following), the variance-based query strategy from Model Picker (Karimi et al., 2021) (referred to as *variance*), and a random query strategy. Fig. 4 shows that CAMS's adaptive query strategy has the sharpest converge rate on cumulative loss, which demonstrates the effectiveness of the queried labels. Moreover, *entropy* achieves the minimum cumulative loss for CIFAR10, DRIFT, and VERTEBRAL under the same query cost. For

the HIV dataset, there is no clear winner between *entropy* and *variance* since the mean of their performance lie within the error bar of each other for the most part.

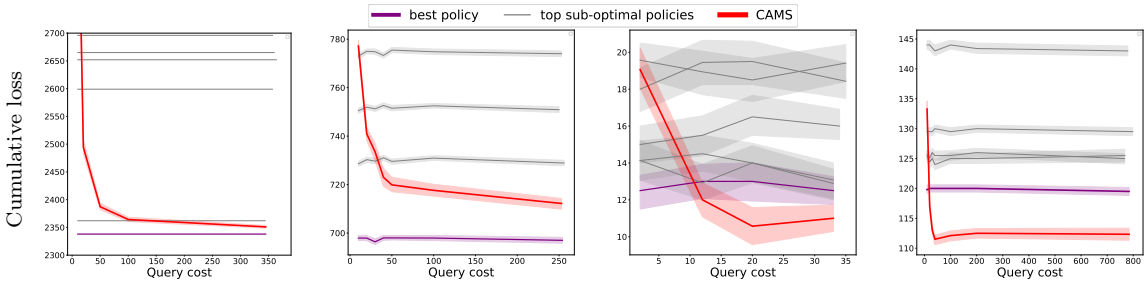


(a) CIFAR10 (b) DRIFT (c) VERTEBRAL (d) HIV

Figure 4: Ablation study of three query strategies (*entropy*, *variance*, *random*) for 4 diverse benchmarks based on the same model recommendation strategy. Under the same query cost constraint, CAMS's query strategy (*entropy*) exceeds the performance of the other two strategies on non-binary benchmarks in terms of query cost and cumulative lost. 90% confident intervals are indicated in shades.

C.3 Comparing CAMS with each individual expert

We evaluate CAMS by comparing it with all the policies available in various benchmarks. The policies in each benchmark are summarized in Appendix B.2 and Table B.5. The empirical results in Fig. 5 demonstrate that CAMS could efficiently outperform all policies and converge to the performance of the best policy with only slight increase in query cost in all benchmarks. In particular, on the VERTEBRAL and HIV benchmarks, CAMS even outperforms the best policy.



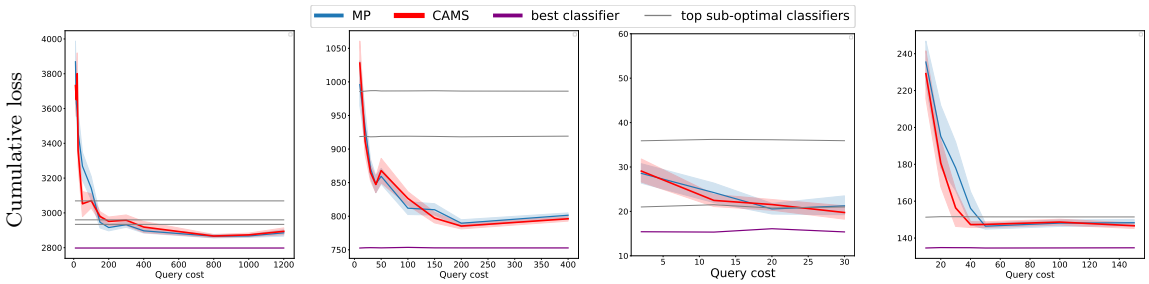
(a) CIFAR10 (b) DRIFT (c) VERTEBRAL (d) HIV

Figure 5: Comparing CAMS with every single policy (only plotted top performance policies in Figure). CAMS could approach the best expert and exceed all others with limited queries. In particular, on VERTEBRAL and HIV Benchmarks, CAMS outperforms the best expert. 90% confident intervals are indicated in shades.

C.4 Comparing CAMS and Model Picker in a context-free environment

CAMS with its own active query strategy component outperforms Model Picker in Fig. 2. When no policy (context) is available and if CAMS's query strategy component uses the same

variance-based query strategy as in Model Picker (Karimi et al., 2021), CAMS degenerates to the Model Picker algorithm in a context-free environment. In a context-free environment, $\Pi = \{\varnothing\}$, so $\Pi^* := \left\{\pi_1^{\text{const}}, \ldots, \pi_k^{\text{const}}\right\}$, where $\pi_j^{\text{const}}(\cdot) := \boldsymbol{e}_j$ represents a policy that only recommends a fixed model. In this case, selecting the best policy to CAMS equals selecting the best single model. Fig. 6 demonstrates that the mean of CAMS and Model Picker lies in the shades of each other, which means CAMS has approximately the same performance as model picker considering the randomness on all benchmarks.



(a) CIFAR10 (b) DRIFT (c) VERTEBRAL (d) HIV

Figure 6: Comparing the model selection strategy of CAMS and Model Picker baseline based on the same variance-based query strategy in a context-free environment. CAMS has approximately the same performance as Model Picker on all the benchmarks. 90% confident intervals are indicated in shades.

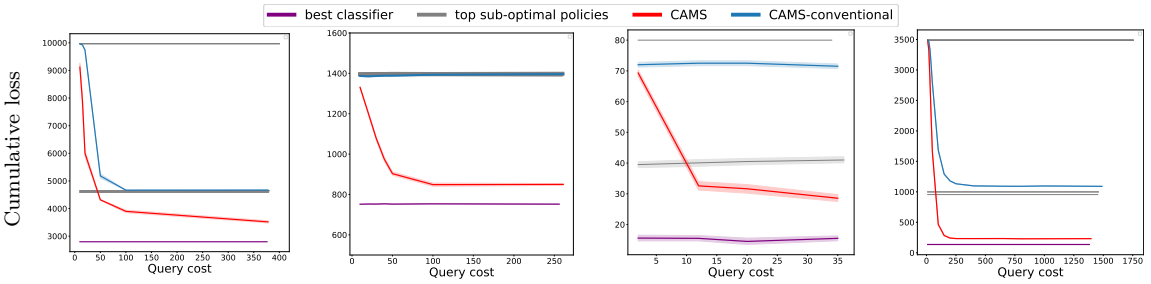
C.5 Robustness against malicious experts in adversarial environments

When given only malicious and random advice policies, the conventional contextual online learning from experts advice framework will be trapped in the malicious or random advice. In contrast, CAMS could efficiently identify these policies and avoid taking advice from them. Meanwhile, it also successfully identifies the best classifier to learn to reach its best performance.

The *novelty* in CAMS that enables this robustness is that we add the constant policies $\left\{\pi_1^{\text{const}}, \ldots, \pi_k^{\text{const}}\right\}$ into the policy set Π to form the new set as Π^*. To illustrate the performance difference, we have created a variant of CAMS by adapting to the conventional approach (named CAMS-conventional). Fig. 7 demonstrates that CAMS could outperform all the malicious and random policies and converge to the performance of the best classifier.
CAMS-*conventional:* We create the CAMS-conventional algorithm as the CAMS using policy set Π, not Π^*.

C.6 Outperformance over the best policy/expert

We also observe that CAMS does not stop at approaching the best policy or classifier performance. Sometimes, it even outperforms all the policies and classifiers, and Fig. 8 demonstrates such a case. To demonstrate the advantage of CAMS, we create two variant versions of CAMS: (1) CAMS-MAX (Appendix C.7), (2) CAMS-Random-Policy (Appendix C.8). CAMS-MAX and CAMS-Random-Policy use the same algorithm as CAMS in



(a) CIFAR10 (b) DRIFT (c) VERTEBRAL (d) HIV

Figure 7: Evaluating the robustness of CAMS compared to the conventional learning from experts' advice (CAMS-conventional) in a complete malicious and random policies environment. When no good policy is available, CAMS could recover from malicious advice and successfully approach the performance of the best classifier. In contrast, the conventional approach will be trapped in malicious advice. 90% confident intervals are indicated in shades.

adversarial settings but have different model selection strategies for ablation study in the stochastic settings.

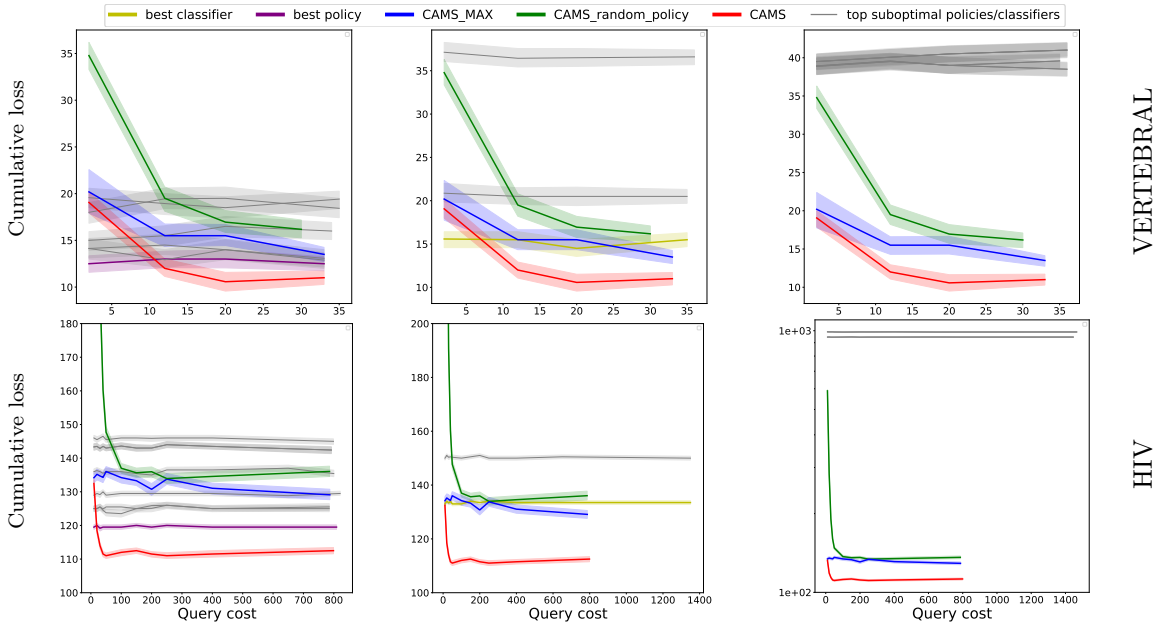
We evaluate the three algorithms on VERTEBRAL and HIV benchmarks in terms of (a) *normal policies* (Fig. 8 Left), (b) *classifiers* (Fig. 8 Middle), and (c) *malicious and random policies* (Fig. 8 Right). In the normal policies column, we only compare the policies with regular policies giving helpful advice. In the classifier column, we compare them with the performance of classifiers only. In the malicious and random policies column, we compare them with unreasonable policies only.

Fig. 8 demonstrates that all three algorithms could outperform the malicious/random policies. However, CAMS-Random-Policy does not outperform the best classifier while both CAMS and CAMS-MAX can on both benchmarks. CAMS-MAX approaches the performance of the best policy but does not outperform the best policy on both benchmarks. Finally, perhaps surprisingly, CAMS outperforms the best policy (Oracle) on both benchmarks and continues to approach the hypothetical, optimal policy (with 0 cumulative loss).

This surprising factor is contributed by the adaptive weighted policy of CAMS, which adaptively creates a better policy by combining the advantage of each sub-optimal policy and classifier to reach the performance of the hypothetical, optimal policy (defined as $\sum_{t=1}^{T} \min_{i\in[|\Pi^*|]} \widetilde{\ell}_{t,i}$). The second reason could be that the benchmark we created, or any real-world cases, will not be strictly in a stochastic setting (in which a single policy outperforms all others or has lower μ in every round). The weight policy strategy can make a better combination of advice for this case.

C.7 The CAMS-MAX algorithm

CAMS-MAX is a variant of CAMS. In an adversarial setting, they share the same algorithm. However, in a stochastic setting, CAMS-MAX gets the index i^* of max value in the probability distribution of policy $\boldsymbol{q}$, and selects the model with the max value in $\pi_{i^*}(\boldsymbol{x}_t)$ to recommendation. The difference is marked in blue in Fig. 9.



(a) normal policies (b) classifiers (c) malicious and random policies

Figure 8: Comparing CAMS, CAMS-MAX and CAMS-RANDOM-POLICY with top policies and classifiers in the VERTEBRA and HIV benchmarks. They outperform all the malicious/random policies. Moreover, CAMS and CAMS-MAX outperform the best classifier. Finally, only CAMS outperforms the best policy (Oracle) in both benchmarks and continues approaching the hypothetical, optimal policy (0 cumulative loss). 90% confident intervals are indicated in shades.

Input: Models $\mathcal{F}$, policies Π^*, #rounds T, budget b
Initialize loss $\tilde{\boldsymbol{L}}_0 \leftarrow 0$; query cost $C_0 \leftarrow 0$
for $t = 1, 2, ..., T$ **do**
  Receive $\boldsymbol{x}_t$
  $\eta_t \leftarrow \textsc{SetRate}(t, \boldsymbol{x}_t, |\Pi^*|)$
  Set $q_{t,i} \propto \exp\left(-\eta_t \tilde{L}_{t-1,i}\right)\ \forall i \in |\Pi^*|$
  $j_t \leftarrow \textsc{Recommend}(\boldsymbol{x}_t, \boldsymbol{q}_t)$
  Output $\hat{y}_{t,j_t} \sim f_{t,j_t}$ as the prediction for $\boldsymbol{x}_t$
  Compute z_t in Eq. (4)
  Sample $U_t \sim \text{Ber}(z_t)$
  if $U_t = 1$ and $C_t \leq b$ **then**
    Query the label y_t
    $C_t \leftarrow C_{t-1} + 1$
    Compute $\boldsymbol{\ell}_t$: $\ell_{t,j} = \mathbb{I}\{\hat{y}_{t,j} \neq y_t\}, \forall j \in [|\mathcal{F}|]$
    Estimate model loss: $\hat{\ell}_{t,j} = \frac{\ell_{t,j}}{z_t}, \forall j \in [|\mathcal{F}|]$
    $\tilde{\boldsymbol{\ell}}_t$: $\tilde{\ell}_{t,i} \leftarrow \langle \pi_i(\boldsymbol{x}_t), \hat{\ell}_{t,j} \rangle, \forall i \in [|\Pi^*|]$
    $\tilde{\boldsymbol{L}}_t = \tilde{\boldsymbol{L}}_{t-1} + \tilde{\boldsymbol{\ell}}_t$
  else
    $\tilde{\boldsymbol{L}}_t = \tilde{\boldsymbol{L}}_{t-1}$
    $C_t \leftarrow C_{t-1}$

procedure $\textsc{SetRate}(t, \boldsymbol{x}_t, m)$
  if STOCHASTIC **then**
    $\eta_t = \sqrt{\frac{\ln m}{t}}$
  if ADVERSARIAL **then**
    Set ρ_t as in §E.1
    $\eta_t = \sqrt{\frac{1}{\sqrt{t}} + \frac{\rho_t}{c^2 \ln c}} \cdot \sqrt{\frac{\ln m}{T}}$
  return η_t

procedure $\textsc{Recommend}(\boldsymbol{x}_t, \boldsymbol{q}_t)$
  if STOCHASTIC **then**
    $i_t \leftarrow \text{maxind}(\boldsymbol{q}_t)$
    $j_t \leftarrow \text{maxind}(\pi_{i_t}(\boldsymbol{x}_t))$
  if ADVERSARIAL **then**
    $i_t \sim \boldsymbol{q}_t$
    $j_t \sim \pi_{i_t}(\boldsymbol{x}_t)$
  return j_t

Figure 9: The CAMS-MAX Algorithm

C.8 The CAMS-Random-Policy algorithm

CAMS-Random-Policy is a variant of CAMS. It shares the same framework with CAMS in an adversarial environment. However, it uses a random sampling policy method in a stochastic setting. It randomly samples the policy from the probability distribution of policy $\boldsymbol{q}$, and selects the model with max value in $\pi_{i^*}(\boldsymbol{x}_t)$ to recommendation. The difference is marked in blue in Fig. 10.

Input: Models $\mathcal{F}$, policies Π^*, #rounds T, budget b
Initialize loss $\tilde{\boldsymbol{L}}_0 \leftarrow 0$; query cost $C_0 \leftarrow 0$
for $t = 1, 2, ..., T$ **do**
  Receive $\boldsymbol{x}_t$
  $\eta_t \leftarrow \textsc{SetRate}(t, \boldsymbol{x}_t, |\Pi^*|)$
  Set $q_{t,i} \propto \exp\left(-\eta_t \tilde{L}_{t-1,i}\right)\ \forall i \in |\Pi^*|$
  $j_t \leftarrow \textsc{Recommend}(\boldsymbol{x}_t, \boldsymbol{q}_t)$
  Output $\hat{y}_{t,j_t} \sim f_{t,j_t}$ as the prediction for $\boldsymbol{x}_t$
  Compute z_t in Eq. (4)
  Sample $U_t \sim \text{Ber}(z_t)$
  if $U_t = 1$ and $C_t \leq b$ **then**
    Query the label y_t
    $C_t \leftarrow C_{t-1} + 1$
    Compute $\boldsymbol{\ell}_t$: $\ell_{t,j} = \mathbb{I}\{\hat{y}_{t,j} \neq y_t\}, \forall j \in [|\mathcal{F}|]$
    Estimate model loss: $\hat{\ell}_{t,j} = \frac{\ell_{t,j}}{z_t}, \forall j \in [|\mathcal{F}|]$
    $\tilde{\boldsymbol{\ell}}_t$: $\tilde{\ell}_{t,i} \leftarrow \langle \pi_i(\boldsymbol{x}_t), \hat{\ell}_{t,j} \rangle, \forall i \in [|\Pi^*|]$
    $\tilde{\boldsymbol{L}}_t = \tilde{\boldsymbol{L}}_{t-1} + \tilde{\boldsymbol{\ell}}_t$
  else
    $\tilde{\boldsymbol{L}}_t = \tilde{\boldsymbol{L}}_{t-1}$
    $C_t \leftarrow C_{t-1}$

procedure $\textsc{SetRate}(t, \boldsymbol{x}_t, m)$
  if STOCHASTIC **then**
    $\eta_t = \sqrt{\frac{\ln m}{t}}$
  if ADVERSARIAL **then**
    Set ρ_t as in §E.1
    $\eta_t = \sqrt{\frac{1}{\sqrt{t}} + \frac{\rho_t}{c^2 \ln c}} \cdot \sqrt{\frac{\ln m}{T}}$
  return η_t

procedure $\textsc{Recommend}(\boldsymbol{x}_t, \boldsymbol{q}_t)$
  if STOCHASTIC **then**
    $i_t \sim \boldsymbol{q}_t$
    $j_t \leftarrow \text{maxind}(\pi_{i_t}(\boldsymbol{x}_t))$
  if ADVERSARIAL **then**
    $i_t \sim \boldsymbol{q}_t$
    $j_t \sim \pi_{i_t}(\boldsymbol{x}_t)$
  return j_t

Figure 10: The CAMS-Random-Policy Algorithm

C.9 Maximal queries from experiments

Table 4 in this section summarizes the maximum query cost under a fixed number of realizations with its associated cumulative loss for all baselines (exclude oracle) on all benchmarks in §5. The result in table is slightly different from the query complexity curves of Fig. 2 (Top). The curve in Fig. 2 (Top) takes the average value, while the table takes the maximal value from a fixed number of simulations. CAMS wins all baselines (other than oracle) in terms of query cost on CIFAR10, DRIFT, and VERTEBRAL benchmarks. CAMS outperforms all baselines in terms of cumulative loss on DRIFT, VERTEBRAL, and HIV benchmarks. In particular, CAMS outperforms both cumulative loss and query cost on the DRIFT and VERTEBRAL benchmarks.

Algorithm	**CIFAR10**	**DRIFT**	**VERTEBRAL**	**HIV**
Max queries, Cumulative loss	*1200, 10000*	*2000, 3000*	*80, 80*	*2000, 4000*
RS	1200, 2916	2000, 766	80, 19	2000, 143
QBC	1200, 2857	1904, 771	72, 20	2000, 139
IWAL	1200, 2854	2000, 760	80, 19	690, 140
MP	1200, 2885	493, 803	33, 25	153, 148
CQBC	1200, 2284	1900, 744	68, 13	2000, 124
CIWAL	1200, 2316	2000, 746	80, 12	690, 124
CAMS	**348**, 2348	**251, 710**	**32, 11**	782, **112**

Table 4: Maximal queries from experiments

Appendix D. Proofs for the Stochastic Setting

In this section, we focus on the stochastic setting. We first prove the regret bound presented in Theorem 4.1 and then prove the query complexity presented in Theorem 4.2 for Algorithm 1.

For Theorem 4.1, note that in the stochastic setting, a lower bound of $\Omega\left(\frac{\log \Pi^*}{\Delta}\right)$ was shown in (Mourtada and Gaïffas, 2019) for online learning problems with expert advice under the full information setting (i.e. assuming labels are given for all data points in the stochastic stream). Our regret upper bounds in stochastic setting match (up to constant factors) the existing lower bounds for online learning problems with expert advice under the full information setting.

To establish the proof of Theorem 4.1, we consider a novel procedure to connect the weighted policy by CAMS to the best policy π_{i^*}. Conceptually, we would like to show that, after a *constant* number of rounds τ_{const}, with high probability, the model selected by CAMS (Line 32) will be the same as the one selected by the best policy i^*. In that way, the expected pseudo regret will be dominated by the maximal cumulative loss up to τ_{const}. Toward this goal, we first bound the weight of the best policy w_{t,i^*} as a function of t, by choosing a proper learning rate η_t (CAMS, Line 23). Then, we identify a constant threshold τ_{const}, beyond which CAMS exhibits the same behavior as π_{i^*} with high probability. Finally, we obtain the regret bound by inspecting the regret at the two stages separately. The formal statement of Theorem 4.1 and the detailed proof are deferred to Appendix D.1.

D.1 Proof of Theorem 4.1

Before providing the proof of Theorem 4.1, we first introduce the following lemma.

Lemma 1. *Fix $\tau \in (0,1)$. Let q_{t,i^*} be the probability of the optimal policy i^* maintained by Algorithm 1 at t. When $t \geq \left(\frac{\ln \frac{|\Pi^*|\tau}{1-\tau}}{\sqrt{\ln|\Pi^*|}\left(\Delta-\sqrt{\frac{2}{t}\ln\frac{2}{\delta}}\right)}\right)^2$, with probability at least $1-\delta$, it holds that $q_{t,i^*} \geq \tau$.*

Proof of Lemma 1. W.l.o.g, we assume $\mu_1 \leq \mu_2 \leq \ldots \mu_{n+k}$. Since we define $\Delta = \min_{i \neq i^*} \Delta_i = \mu_2 - \mu_1 = \frac{\mathbb{E}\left[\widetilde{L}_{t,2} - \widetilde{L}_{t,1}\right]}{t}$, we also have $q_{t,i^*} = q_{t,1} = \frac{\exp\left(-\eta_t \widetilde{L}_{t-1,1}\right)}{\sum_{i=1}^{|\Pi^*|}\exp\left(-\eta_t \widetilde{L}_{t-1,i}\right)}$ as the weight of optimal

expert at round t. Therefore

$$
\begin{aligned}
q_{t,i^*} = q_{t,1} &= \frac{\exp\left(-\eta_t \widetilde{L}_{t-1,1}\right)}{\sum_{i=1}^{|\Pi^*|} \exp\left(-\eta_t \widetilde{L}_{t-1,i}\right)} \\
&\overset{(a)}{=} \frac{\exp\left(-\eta_t \widetilde{L}_{t-1,1} + \eta_t \widetilde{L}_{t-1,2}\right)}{\sum_{i=1}^{|\Pi^*|} \exp\left(-\eta_t \widetilde{L}_{t-1,i} + \eta_t \widetilde{L}_{t-1,2}\right)} \\
&\overset{(b)}{=} \frac{\exp\left(\eta_t \sum_{s=1}^{t} \delta_s\right)}{\exp\left(\eta_t \sum_{s=1}^{t} \delta_s\right) + 1 + \sum_{i=3}^{|\Pi^*|} \exp\left(-\eta_t \widetilde{L}_{t-1,i} + \eta_t \widetilde{L}_{t-1,2}\right)} \\
&\geq \frac{\exp\left(\eta_t \sum_{s=1}^{t} \delta_s\right)}{\exp\left(\eta_t \sum_{s=1}^{t} \delta_s\right) + |\Pi^*|}
\end{aligned}
\tag{6}
$$

where step (a) is by dividing the cumulative loss of sub-optimal policy π_2 and step (b) is by defining $\delta_t \triangleq \widetilde{\ell}_{t-1,2} - \widetilde{\ell}_{t-1,1}$.

Let $\tau \in (0,1)$, such that $q_{t,i^*} \geq \frac{\exp\left(\eta_t \sum_{s=1}^{t} \delta_s\right)}{\exp\left(\eta_t \sum_{s=1}^{t} \delta_s\right) + |\Pi^*|} \geq \tau$. Plugging in $\eta_t = \sqrt{\frac{\ln|\Pi^*|}{t}}$ and define $\overline{\delta_t} = \frac{1}{t}\sum_{s=1}^{t} \delta_s$, we get

$$
\frac{\exp\left(\sqrt{\ln|\Pi^*|}\sqrt{t}\cdot\overline{\delta_t}\right)}{\exp\left(\sqrt{\ln|\Pi^*|}\sqrt{t}\cdot\overline{\delta_t}\right) + |\Pi^*|} \geq \tau
$$

Therefore, we obtain $\exp\left(\sqrt{\ln|\Pi^*|}\sqrt{t}\cdot\overline{\delta_t}\right) \geq \frac{|\Pi^*|\tau}{1-\tau}$. Rearranging the terms, we get

$$
t \geq \left(\frac{\ln\frac{|\Pi^*|\tau}{1-\tau}}{\sqrt{\ln|\Pi^*|}\cdot\overline{\delta_t}}\right)^2
$$

Now by Hoeffding's inequality, we know $\mathbb{P}\left[|\overline{\delta}_t - \Delta| \geq \epsilon\right] \leq 2e^{-\frac{t\epsilon^2}{2}}$. Let $2e^{-\frac{t\epsilon^2}{2}} = \delta$. Therefore, when $t \geq \left(\frac{\ln\frac{|\Pi^*|\tau}{1-\tau}}{\sqrt{\ln|\Pi^*|}(\Delta-\epsilon)}\right)^2 = \left(\frac{\ln\frac{|\Pi^*|\tau}{1-\tau}}{\sqrt{\ln|\Pi^*|}\left(\Delta-\sqrt{\frac{2}{t}\ln\frac{2}{\delta}}\right)}\right)^2$, it holds that $q_{t,i^*} \geq \tau$ with probability at least $1-\delta$.

□

Lemma 2. *At round t, when $t \geq \left(\frac{\ln\frac{|\Pi^*|}{\gamma} + \sqrt{\ln|\Pi^*|\cdot 2\ln\frac{2}{\delta}}}{\sqrt{\ln|\Pi^*|}\Delta}\right)^2$, it holds that the arm chosen by the best policy i^* will be the arm chosen by Algorithm 1 with probability at least $1-\delta$. That is,* $\arg\max\left\{\sum_{i\in[|\Pi^*|]} q_{t,i}\pi_i(\mathbf{x}_t)\right\} = \arg\max\left\{\pi_{i^*}(\mathbf{x}_t)\right\}$.

Proof of Lemma 2. At round t, for Algorithm 1, we have loss

$$\sum_{j=1}^{k} \mathbb{I}\left\{ j = \arg\max \left\{ \sum_{i\in[|\Pi^*|]} q_{t,i}\pi_i(\boldsymbol{x}_t) \right\} \right\} \widehat{\ell}_{t,j}.$$

Let $q_{t,i^*} \geq \tau$. At round t, the best policy i^*'s top weight arm j_{t,i^*}'s probability $\max\{\pi_{i^*}(\boldsymbol{x}_t)\}$ is at least $\frac{1}{k}$. The second rank probability of $\pi_{i^*}(\boldsymbol{x}_t)$ is $\max_j [\pi_{i^*}(\boldsymbol{x}_t)]_{j\neq \text{maxind}(\pi_{i^*}(\boldsymbol{x}_t))}$. Let us define

$$\gamma := \min_{\boldsymbol{x}_t} \left\{ \max_{w_j\in \boldsymbol{w}^t_{i^*}} w_j - \max_{w_j\in \boldsymbol{w}^t_{i^*}, j\neq \text{maxind}(\boldsymbol{w}^t_{i^*})} w_j \right\} \tag{7}$$

$$= \max\{\pi_{i^*}(\boldsymbol{x}_t)\} - \max_j \left\{ [\pi_{i^*}(\boldsymbol{x}_t)]_{j\neq \text{maxind}(\pi_{i^*}(\boldsymbol{x}_t))} \right\}$$

as minimal gap in model distribution space of best policy. The arm recommended by the best policy i^* of CAMS will dominate CAMS's selection, when we have

$$q_{t,i^*} \cdot \max\{\pi_{i^*}(\boldsymbol{x}_t)\} \geq (1 - q_{t,i^*}) + q_{t,i^*} \left(\max_j [\pi_{i^*}(\boldsymbol{x}_t)]_{j\neq \text{maxind}(\pi_{i^*}(\boldsymbol{x}_t))} \right)$$

Rearranging the terms, and by

$$q_{t,i^*} \cdot \gamma \overset{\text{Eq. (7)}}{=} q_{t,i^*} \left(\max\{\pi_{i^*}(\boldsymbol{x}_t)\} - \max_j [\pi_{i^*}(\boldsymbol{x}_t)]_{j\neq \text{maxind}(\pi_{i^*}(\boldsymbol{x}_t))} \right) \geq (1 - q_{t,i^*})$$

Therefore, we get $\tau \cdot \gamma \geq (1 - \tau)$, and thus $\tau \geq \frac{1}{\gamma+1}$.

Set $\tau \geq \frac{1}{\gamma+1}$. By Lemma 1, we get

$$\begin{aligned} t &\geq \left(\frac{\ln \frac{|\Pi^*|\tau}{1-\tau}}{\sqrt{\ln|\Pi^*|}\,(\Delta-\epsilon)} \right)^2 \\ &\geq \left(\frac{\ln\left(\frac{|\Pi^*|}{\gamma}\right)}{\sqrt{\ln|\Pi^*|}\,(\Delta-\epsilon)} \right)^2 \\ &\overset{(c)}{\geq} \left(\frac{\ln \frac{|\Pi^*|}{\gamma}}{\sqrt{\ln|\Pi^*|}\Delta - \sqrt{\ln|\Pi^*| \cdot \frac{2}{t}\ln\frac{2}{\delta}}} \right)^2 \end{aligned}$$

where the last step is by applying $2e^{-\frac{t\epsilon^2}{2}} = \delta$, thus, $\epsilon = \sqrt{\frac{2}{t}\ln\frac{2}{\delta}}$. Dividing both sides by t,

$$
\begin{aligned}
1 &\overset{(d)}{\geq} \left(\frac{\ln\frac{|\Pi^*|}{\gamma}}{\sqrt{\ln|\Pi^*|\cdot t}\Delta - \sqrt{\ln|\Pi^*|\cdot 2\ln\frac{2}{\delta}}}\right)^2 \\
\ln\frac{|\Pi^*|}{\gamma} &\leq \sqrt{t}\sqrt{\ln\left(|\Pi^*|\right)}\Delta - \sqrt{\ln\left(|\Pi^*|\right)\cdot 2\ln\frac{2}{\delta}} \\
t &\geq \left(\frac{\ln\frac{|\Pi^*|}{\gamma} + \sqrt{\ln|\Pi^*|\cdot 2\ln\frac{2}{\delta}}}{\sqrt{\ln|\Pi^*|}\Delta}\right)^2 .
\end{aligned}
$$

So, when $t \geq \left(\frac{\ln\frac{|\Pi^*|}{\gamma}+\sqrt{\ln|\Pi^*|\cdot 2\ln\frac{2}{\delta}}}{\sqrt{\ln|\Pi^*|}\Delta}\right)^2$, it holds that

$$
\arg\max\left\{\sum_{i\in[|\Pi^*|]} q_{t,i}\pi_i(\boldsymbol{x}_t)\right\} = \arg\max\left\{\pi_{i^*}(\boldsymbol{x}_t)\right\}.
$$

□

Proof of Theorem 4.1. Therefore, with probability at least $1-\delta$, we get constant regret $\left(\frac{\ln\frac{|\Pi^*|}{\gamma}+\sqrt{\ln|\Pi^*|\cdot 2\ln\frac{2}{\delta}}}{\sqrt{\ln|\Pi^*|}\Delta}\right)^2$.

Furthermore, with probability at most δ, the regret is upper bounded by T. Thus, we have

$$
\begin{aligned}
\overline{\mathcal{R}}(T) &\leq (1-\delta)\left(\frac{\ln\frac{|\Pi^*|}{\gamma} + \sqrt{\ln|\Pi^*|\cdot 2\ln\frac{2}{\delta}}}{\sqrt{\ln|\Pi^*|}\Delta}\right)^2 + \delta T \\
&\overset{(a)}{\leq} \left(1-\frac{1}{T}\right)\left(\frac{\ln\frac{|\Pi^*|}{\gamma} + \sqrt{\ln|\Pi^*|\cdot(2\ln T + 2\ln 2)}}{\sqrt{\ln|\Pi^*|}\Delta}\right)^2 + 1 \\
&= O\left(\frac{\ln T}{\Delta^2} + \left(\frac{\ln\frac{|\Pi^*|}{\gamma}}{\sqrt{\ln|\Pi^*|}\Delta}\right)^2\right),
\end{aligned}
$$

where step (a) by set $\delta = \frac{1}{T}$, and where γ in Eq. (7) is the min gap. □

D.2 Proof of Theorem 4.2

In this section, we analyze the query complexity of CAMS in the stochastic setting. Our main idea is to derive from query indicator U_t and query probability z_t. We first used Lemma 3 to bound the expected number of queries $\sum_{t=1}^{T} U_t$ by the sum of query probability as $\sum_{t=1}^{T}\delta_0^t + \sum_{t=1}^{T}\mathfrak{E}(\hat{\boldsymbol{y}}_t, \boldsymbol{w}_t)$. Then we used Lemma 4 to bound the first item (which

corresponds to the lower bound of query probability over T rounds) and applied Lemma 5 to bound the second term (which characterizes the model disagreement). Finally, we combined the upper bounds on the two parts to reach the desired result.

Lemma 3. *The query complexity of Algorithm 1 is upper bounded by*

$$\mathbb{E}\left[\sum_{t=1}^{T}\left(\frac{1}{\sqrt{t}}+\frac{\sum_{y\in\mathcal{Y}}\langle \mathbf{w}_t,\boldsymbol{\ell}_t^y\rangle \log_{|\mathcal{Y}|}\frac{1}{\langle \mathbf{w}_t,\boldsymbol{\ell}_t^y\rangle}}{|\mathcal{Y}|}\right)\right]. \tag{8}$$

Proof. Now we have model disagreement defined in Eq. (3), the query probability defined in Eq. (4), and the query indicator U. Let us assume, at each round, we have query probability $z_t > 0$, which indicates we will not process the instance that all the models' prediction are the same.

At round t, from query probability Eq. (4), we have

$$\begin{aligned} z_t =& \max\left\{\delta_0^t, \mathfrak{E}\left(\hat{\boldsymbol{y}}_t, \boldsymbol{w}_t\right)\right\} \\ \leq& \delta_0^t + \mathfrak{E}\left(\hat{\boldsymbol{y}}_t, \boldsymbol{w}_t\right), \end{aligned}$$

where the inequality is by applying that $\forall A, B \geq 0, \max\{A, B\} \leq A + B$.

Thus, in total round T, we could get the following equation as the cumulative query cost,

$$\mathbb{E}\left[\sum_{t=1}^{T} U_t\right] \leq \mathbb{E}\left[\sum_{t=1}^{T}\left(\frac{1}{\sqrt{t}}+\frac{\sum_{c\in\mathcal{Y}}\langle \boldsymbol{w}_t,\boldsymbol{\ell}_t^y\rangle \log_{|\mathcal{Y}|}\frac{1}{\langle \boldsymbol{w}_t,\boldsymbol{\ell}_t^y\rangle}}{|\mathcal{Y}|}\right)\right], \tag{9}$$

where the inequality is by inputting $\delta_0^t = \frac{1}{\sqrt{t}}$ and Eq. (3). □

Lemma 4. $\sum_{t=1}^{T}\frac{1}{\sqrt{t}} \leq 2\sqrt{T}$.

Proof. We can bound the LHS as follows:

$$\begin{aligned} \sum_{t=1}^{T}\frac{1}{\sqrt{t}} &= \sum_{t=1}^{\lfloor\sqrt{T}\rfloor}\frac{1}{\sqrt{t}} + \sum_{t=\lfloor\sqrt{T}\rfloor+1}^{T}\frac{1}{\sqrt{t}} \\ &\leq \sqrt{T} + \sum_{t=\lfloor\sqrt{T}\rfloor+1}^{T}\frac{1}{\sqrt{T}} \\ &= \sqrt{T} + \left(T - \sqrt{T}\right)\frac{1}{\sqrt{T}} \\ &\leq 2\sqrt{T}. \end{aligned}$$

□

Lemma 5. *Denote the true label at round t by y_t, and define $p_{t,y} := \sum_{j\in[k]}\mathbb{I}\{\hat{y}_{t,j} = y\}\, w_j$. Further define $R_t := \sum_t 1 - p_{t,y_t}$ as the expected cumulative loss of Algorithm 1 at t . Then*

$$\sum_{t=1}^{T}\frac{\sum_{y\in\mathcal{Y}}\langle \mathbf{w}_t,\boldsymbol{\ell}_t^y\rangle \log_{|\mathcal{Y}|}\frac{1}{\langle \mathbf{w}_t,\boldsymbol{\ell}_t^y\rangle}}{|\mathcal{Y}|} \leq \frac{R_T\cdot\left(\log_{|\mathcal{Y}|}\frac{T^2(|\mathcal{Y}|-1)}{R^2}\right)}{|\mathcal{Y}|}.$$

Proof of Lemma 5. Suppose at round t, the true label is y_t. $\sum_{y\neq y_t} p_{t,y} = 1 - p_{t,y_t} = 1 - \left\langle \sum_{i\in|\Pi^*|} q_{t,i}\pi_i(\boldsymbol{x}_t), \boldsymbol{\ell}_t \right\rangle = r_t$,

$$\begin{aligned}
\frac{\sum_{y\in\mathcal{Y}}\langle \boldsymbol{w}_t, \boldsymbol{\ell}_t^y\rangle \log_{|\mathcal{Y}|}\frac{1}{\langle \boldsymbol{w}_t, \boldsymbol{\ell}_t^y\rangle}}{|\mathcal{Y}|} &= \frac{(1-p_{t,y_t})\log_{|\mathcal{Y}|}\frac{1}{1-p_{t,y_t}}}{|\mathcal{Y}|} + \frac{\sum_{y\neq y_t}(1-p_{t,y})\log_{|\mathcal{Y}|}\frac{1}{1-p_{t,y}}}{|\mathcal{Y}|} \\
&\overset{(a)}{\leq} \frac{(1-p_{t,y_t})\log_{|\mathcal{Y}|}\frac{1}{1-p_{t,y_t}}}{|\mathcal{Y}|} + (|\mathcal{Y}|-1)\frac{\frac{(1-p_{t,y_t})}{|\mathcal{Y}|-1}\log_{|\mathcal{Y}|}\frac{|\mathcal{Y}|-1}{1-p_{t,y_t}}}{|\mathcal{Y}|} \\
&\leq \frac{(1-p_{t,y_t})\log_{|\mathcal{Y}|}\frac{1}{1-p_{t,y_t}}}{|\mathcal{Y}|} + \frac{(1-p_{t,y_t})\log_{|\mathcal{Y}|}\frac{|\mathcal{Y}|-1}{1-p_{t,y_t}}}{|\mathcal{Y}|} \\
&= \frac{(1-p_{t,y_t})\log_{|\mathcal{Y}|}\frac{|\mathcal{Y}|-1}{(1-p_{t,y_t})^2}}{|\mathcal{Y}|} \\
&\overset{(b)}{\leq} \frac{r_t\log_{|\mathcal{Y}|}\frac{|\mathcal{Y}|-1}{r_t^2}}{|\mathcal{Y}|},
\end{aligned}$$

where step (a) is by applying Jensen's inequality and using $1 - p_{t,y} = \frac{1-p_{t,y_t}}{|\mathcal{Y}|-1}$, and step (b) is by replacing $1 - p_{t,y_t}$ by r_t.

Since when $r_t \in [0,1]$, $\frac{r_t\log_{|\mathcal{Y}|}\frac{|\mathcal{Y}|-1}{r_t^2}}{|\mathcal{Y}|}$ is concave, let $R_T = \sum_{t=1}^T r_t = \sum r_t$, we get

$$\sum_{t=1}^T \frac{\sum_{y\in\mathcal{Y}}\langle \boldsymbol{w}_t, \boldsymbol{\ell}_t^y\rangle \log_{|\mathcal{Y}|}\frac{1}{\langle \boldsymbol{w}_t, \boldsymbol{\ell}_t^y\rangle}}{|\mathcal{Y}|} \leq \frac{T\left(\frac{\sum r_t}{T}\right)\left(\log_{|\mathcal{Y}|}\frac{|\mathcal{Y}|-1}{\frac{\sum r_t}{T}\frac{\sum r_t}{T}}\right)}{|\mathcal{Y}|} = \frac{R\left(\log_{|\mathcal{Y}|}\frac{T^2(|\mathcal{Y}|-1)}{R^2}\right)}{|\mathcal{Y}|}. \quad (10)$$

Since R is the cumulative loss up to round T, T's incremental rate is no less than R's incremental rate. Thus, $R \leq T$ and $\frac{T_t}{R_t} \leq \frac{T_{t+1}}{R_{t+1}}$. So we get Eq. (10). □

Now we are ready to prove Theorem 4.2.

Proof of Theorem 4.2. From Lemma 3, we get the following equation as the cumulative query cost

$$\mathbb{E}\left[\sum_{t=1}^T U_t\right] \leq \mathbb{E}\left[\sum_{t=1}^T\left(\frac{1}{\sqrt{t}} + \frac{\sum_{y\in\mathcal{Y}}\langle \boldsymbol{w}_t, \boldsymbol{\ell}_t^y\rangle \log_{|\mathcal{Y}|}\frac{1}{\langle \boldsymbol{w}_t, \boldsymbol{\ell}_t^y\rangle}}{|\mathcal{Y}|}\right)\right].$$

Let us assume the expected total loss of best policy is $T\mu_{i^*}$, thus from Theorem 4.1, we get

$$\mathbb{E}[R] = \mathbb{E}\left[\sum_{t=1}^T r_t\right] \leq O\left(\frac{\ln T}{\Delta^2} + \left(\frac{\ln\frac{|\Pi^*|}{\gamma}}{\sqrt{\ln|\Pi^*|}\Delta}\right)^2\right) + T\mu_{i^*}.$$

We get our regret bound $\overline{\mathcal{R}}_T$ (CAMS) proved in Theorem 4.1 and plug theorem bound into the query complexity bound given by Lemma 4 and Lemma 5, we have

$$
\begin{aligned}
\mathbb{E}\left[\sum_{t=1}^{T} U_t\right] &\leq 2\sqrt{T} + \frac{\left(O\left(\frac{\ln T}{\Delta^2} + \left(\frac{\ln \frac{|\Pi^*|}{\gamma}}{\sqrt{\ln |\Pi^*|}\Delta}\right)^2\right) + T\mu_{i^*}\right)}{|\mathcal{Y}|} \\
&\quad \cdot \left(\log_{|\mathcal{Y}|} \frac{T^2\left(|\mathcal{Y}|-1\right)}{\left(O\left(\frac{\ln T}{\Delta^2} + \left(\frac{\ln \frac{|\Pi^*|}{\gamma}}{\sqrt{\ln |\Pi^*|}\Delta}\right)^2\right) + T\mu_{i^*}\right)^2}\right) \\
&\leq \frac{\left(O\left(\frac{\ln T}{\Delta^2} + \left(\frac{\ln \frac{|\Pi^*|}{\gamma}}{\sqrt{\ln |\Pi^*|}\Delta}\right)^2\right) + T\mu_{i^*}\right)\left(\log_{|\mathcal{Y}|}\left(T|\mathcal{Y}|\right)\right)}{|\mathcal{Y}|} \\
&= O\left(\frac{\left(\frac{\ln T}{\Delta^2} + \left(\frac{\ln \frac{|\Pi^*|}{\gamma}}{\sqrt{\ln |\Pi^*|}\Delta}\right)^2 + T\mu_{i^*}\right)\left(\ln\left(T\right)\right)}{|\mathcal{Y}| \ln |\mathcal{Y}|}\right),
\end{aligned}
$$

where γ is defined as Eq. (7). □

Appendix E. Proofs for the Adversarial Setting

E.1 Adversarial setting

Under the adversarial setting, we assume that the loss is chosen by an adversary (i.e. by determining what $\boldsymbol{x}_t$ the leaner should receive) before each round. To avoid a linear regret, CAMS randomizes its choice of model selection policies according to the policy distribution $\boldsymbol{q}_t$ it maintains at round t.

Let $\tilde{L}_{T,*} := \min_{i\in[|\Pi^*|]} \sum_{t=1}^{T} \tilde{\ell}_{t,i}$ be the cumulative loss of the best policy. The expected regret (Eq. (2)) for CAMS equals to $\mathcal{R}_T\,(\text{CAMS}) = \mathbb{E}\big[\sum_{t=1}^{T}\langle \boldsymbol{q}_t, \tilde{\boldsymbol{\ell}}_t\rangle\big] - \tilde{L}_{T,*}$. We show that under the adversarial setting, CAMS achieves sub-linear regret in T without accessing all labels.

Theorem E.1. *(Regret) Let c be the number of classes and ρ_t be specified as Line 25-26 in the* SetRate *procedure. Under the adversarial setting, the expected regret of* CAMS *is bounded by* $2c\sqrt{\ln c/\rho_T}\cdot\sqrt{T\log|\Pi^*|}$.

The proof is provided in Appendix E.2. Assuming ρ_t as constant, our regret upper bound in Theorem E.1 matches (up to constants) the lower bound of $\Omega\left(\sqrt{T\ln|\Pi^*|}\right)$ for online learning problems with expert advice under the full information setting (Cesa-Bianchi et al., 1997; Seldin and Lugosi, 2016) (i.e. assuming labels are given for all data points). Hereby, the decaying learning rate η_t as specified in Line 26 is based on two parameters, where $1/\sqrt{t}$ corresponds to the lower bound δ_0^t on the query probability (see §3), and $\rho_t \triangleq 1 - \max_{\tau\in[t]}\langle \boldsymbol{w}_t, \mathbb{I}\{\hat{\boldsymbol{y}}_t = y\}\rangle$ (11) is a (data-dependent) term that is chosen to reduce the impact of the randomized query strategy on the regret bound (especially when t is large). Intuitively, ρ_t relates to the skewness of the policy where the max term corresponds to the maximal probability of most probable mispredicted label over t rounds. Note that in theory ρ_t can be small (e.g. CAMS may choose a constant policy $\pi_i^{\text{const}} \in \Pi^*$ that mispredict the label for $\boldsymbol{x}_t$, which leads to $\rho_t = 0$); therefore, for practical applications, we consider to "regularize" the policies (Appendix B.4) to ensure that probability a policy selecting any model is bounded away from 0.

Finally, the following theorem, as proved in Appendix E.3, establishes a bound on the query complexity of CAMS.

Theorem E.2. *(Query Complexity, informal). Under the adversarial setting, the expected query complexity over T rounds is* $O\left(\frac{\ln T}{c\ln c}\left(\sqrt{\frac{T\log(|\Pi^*|)}{\rho_T}} + \tilde{L}_{T,*}\right)\right)$.

In this section, we first prove the regret bound presented in Theorem E.1 and then prove the query complexity bound presented in Theorem E.2 for Algorithm 1 in the adversarial setting. Lemma 6 builds upon the proof of the hedge algorithm (Freund and Schapire, 1997), but with an *adaptive* learning rate.

E.2 Proof of Theorem E.1

Lemma 6. *Consider the setting of Algorithm 1, Let us define* $h_{t,i} = \exp\left(-\eta_t \tilde{L}_{t-1,i}\right)\ \forall i \in |\Pi^*|$ *as exponential cumulative loss of policy i, η_t is the adaptive learning rate and $\mathbf{q}_t$ is the*

probability distribution of policies, then

$$\log \frac{\sum_{i\in[|\Pi^*|]} h_{T+1,i}}{\sum_{i\in[|\Pi^*|]} h_{1,i}} \leq -\sum_{t=1}^{T} \eta_t \sum_{i=1}^{|\Pi^*|} q_{t,i} \widetilde{\ell}_{t,i} + \sum_{t=1}^{T} \frac{\eta_t^2}{2} \sum_{i=1}^{|\Pi^*|} q_{t,i} \left(\widetilde{\ell}_{t,i}\right)^2.$$

Proof. We first bound the following term

$$\begin{aligned}
\frac{\sum_{i\in[|\Pi^*|]} h_{t+1,i}}{\sum_{i\in[|\Pi^*|]} h_{t,i}} &= \sum_{i=1}^{|\Pi^*|} \frac{h_{t+1,i}}{\sum_{i\in[|\Pi^*|]} h_{t,i}} \\
&= \sum_{i=1}^{|\Pi^*|} q_{t,i} \exp\left(-\eta_t \widetilde{\ell}_{t,i}\right) \\
&\leq \sum_{i=1}^{|\Pi^*|} q_{t,i} \left(1 - \eta_t \widetilde{\ell}_{t,i} + \frac{\eta_t^2 \left(\widetilde{\ell}_{t,i}\right)^2}{2}\right) \\
&= 1 - \eta_t \sum_{i=1}^{|\Pi^*|} q_{t,i} \widetilde{\ell}_{t,i} + \frac{\eta_t^2}{2} \sum_{i=1}^{|\Pi^*|} q_{t,i} \left(\widetilde{\ell}_{t,i}\right)^2,
\end{aligned}$$

where the inequality is by applying that for $x \leq 0$, we have $e^x \leq 1 + x + \frac{x^2}{2}$.
By taking log on both side, we get

$$\begin{aligned}
\log \frac{\sum_{i\in[|\Pi^*|]} h_{t+1,i}}{\sum_{i\in[|\Pi^*|]} h_{t,i}} &\leq \log\left(1 - \eta_t \sum_{i=1}^{|\Pi^*|} q_{t,i} \widetilde{\ell}_{t,i} + \frac{\eta_t^2}{2} \sum_{i=1}^{|\Pi^*|} q_{t,i} \left(\widetilde{\ell}_{t,i}\right)^2\right) \\
&\overset{(a)}{\leq} -\eta_t \sum_{i=1}^{|\Pi^*|} q_{t,i} \widetilde{\ell}_{t,i} + \frac{\eta_t^2}{2} \sum_{i=1}^{|\Pi^*|} q_{t,i} \left(\widetilde{\ell}_{t,i}\right)^2,
\end{aligned}$$

where step (a) is by applying that $\log(1 + x) \leq x$, when $x \geq -1$.
Now summing over $t = 1 : T$ yields:

$$\begin{aligned}
\log \frac{\sum_{i\in[|\Pi^*|]} h_{T+1,i}}{\sum_{i\in[|\Pi^*|]} h_{1,i}} &= \sum_{t=1}^{T} \log \frac{\sum_{i\in[|\Pi^*|]} h_{t+1,i}}{\sum_{i\in[|\Pi^*|]} h_{t,i}} \\
&\leq -\sum_{t=1}^{T} \eta_t \sum_{i=1}^{|\Pi^*|} q_{t,i} \widetilde{\ell}_{t,i} + \sum_{t=1}^{T} \frac{\eta_t^2}{2} \sum_{i=1}^{|\Pi^*|} q_{t,i} \left(\widetilde{\ell}_{t,i}\right)^2.
\end{aligned}$$

□

Lemma 7. *Consider the setting of Algorithm 1, z_t is query probability defined as* $\max\{\delta_0^t, \mathfrak{E}(\hat{\mathbf{y}}_t, \mathbf{w}_t)\}$, *where* $\delta_0^t = \frac{1}{\sqrt{t}}$ *and* $\mathfrak{E}(\hat{\mathbf{y}}_t, \mathbf{w}_t)$ *is defined in Eq. (3),* $p_{t,y} := \sum_{j\in[k]} \mathbb{I}\{\hat{y}_{t,j} = y\} w_j$, *then*

$$z_t \geq \frac{1}{|\mathcal{Y}| \ln |\mathcal{Y}|} \left(p_{t,y_t}(1 - p_{t,y_t}) + p_{t,y}(1 - p_{t,y})\right), \forall c \neq y_t.$$

Proof. We first bound the query probability term

$$\begin{aligned}
z_t &= \max\left\{\delta_0^t, \mathfrak{E}\left(\hat{\boldsymbol{y}}_t, \boldsymbol{w}_t\right)\right\} \\
&= \max\{\delta_0^t, \frac{1}{|\mathcal{Y}|}\sum_{y\in\mathcal{Y}}\langle \boldsymbol{w}_t, \boldsymbol{\ell}_t^y\rangle \log_{|\mathcal{Y}|}\frac{1}{\langle \boldsymbol{w}_t, \boldsymbol{\ell}_t^y\rangle}\} \\
&= \max\{\delta_0^t, \frac{1}{|\mathcal{Y}|}\sum_{y\in\mathcal{Y}}(1-p_{t,y})\cdot \ln\frac{1}{1-p_{t,y}}\frac{1}{\ln|\mathcal{Y}|}\} \\
&\overset{(a)}{\geq} \max\{\delta_0^t, \frac{1}{|\mathcal{Y}|}\sum_{y\in\mathcal{Y}}(1-p_{t,y})\cdot p_{t,y}\cdot\frac{1}{\ln|\mathcal{Y}|}\} \\
&= \max\{\delta_0^t, \frac{1}{|\mathcal{Y}|\ln|\mathcal{Y}|}\sum_{y\in\mathcal{Y}}(1-p_{t,y})\cdot p_{t,y}\} \\
&\overset{(b)}{\geq} \frac{1}{|\mathcal{Y}|\ln|\mathcal{Y}|}\left(p_{t,y_t}\left(1-p_{t,y_t}\right)+p_{t,y}\left(1-p_{t,y}\right)\right), \forall y\neq y_t,
\end{aligned}$$

where step (a) is by applying $\ln(1+x)\geq\frac{x}{1+x}$ for $x>-1$,

$$\ln\frac{1}{1-p_{t,y}} = \ln\left(1+\frac{p_{t,y}}{1-p_{t,y}}\right) \geq \frac{\frac{p_{t,y}}{1-p_{t,y}}}{\frac{1}{1-p_{t,y}}} = p_{t,y},$$

and step (b) is by applying $\forall a,b\in\mathbb{R}, \max\{a,b\}\geq a$.

□

Proof of Theorem E.1. By applying Lemma 6, we got

$$\log\frac{\sum_{i\in[|\Pi^*|]}h_{T+1,i}}{\sum_{i\in[|\Pi^*|]}h_{1,i}} \leq -\sum_{t=1}^{T}\eta_t\sum_{i=1}^{|\Pi^*|}q_{t,i}\widetilde{\ell}_{t,i}+\sum_{t=1}^{T}\frac{\eta_t^2}{2}\sum_{i=1}^{|\Pi^*|}q_{t,i}\left(\widetilde{\ell}_{t,i}\right)^2.$$

For any policy s, we have a lower bound

$$\begin{aligned}
\log\frac{\sum_{i\in[|\Pi^*|]}h_{T+1,i}}{\sum_{i\in[|\Pi^*|]}h_{1,i}} &\geq \log\frac{h_{T+1,s}}{\sum_{i\in[|\Pi^*|]}h_{1,i}} \\
&\overset{(a)}{=} \log\frac{h_{T+1,s}}{|\Pi^*|} \\
&= -\log(n+k) - \eta_T\sum_{t=1}^{T}\widetilde{\ell}_{t,s}, \qquad (12)
\end{aligned}$$

where step (a) in Eq. (12) is by initializing $\widetilde{\boldsymbol{L}}_0=0$, $e^0=1$, and $\sum_{i\in[|\Pi^*|]}\boldsymbol{h}_1 = e^{\left(-\eta_t\widetilde{\boldsymbol{L}}_0\right)} = |\Pi^*|$.

Thus, we have

$$
\begin{aligned}
-\sum_{t=1}^{T}\eta_t\sum_{i=1}^{|\Pi^*|}q_{t,i}\widetilde{\ell}_{t,i}+\sum_{t=1}^{T}\frac{\eta_t^2}{2}\sum_{i=1}^{|\Pi^*|}q_{t,i}\left(\widetilde{\ell}_{t,i}\right)^2 &\geq -\log\left(n+k\right)-\eta_T\sum_{t=1}^{T}\widetilde{\ell}_{t,s}\\
\sum_{t=1}^{T}\eta_t\sum_{i=1}^{|\Pi^*|}q_{t,i}\widetilde{\ell}_{t,i}-\eta_T\sum_{t=1}^{T}\widetilde{\ell}_{t,s} &\leq \log\left(n+k\right)+\sum_{t=1}^{T}\frac{\eta_t^2}{2}\sum_{i=1}^{|\Pi^*|}q_{t,i}\left(\widetilde{\ell}_{t,i}\right)^2\\
\eta_T\sum_{t=1}^{T}\sum_{i=1}^{|\Pi^*|}q_{t,i}\widetilde{\ell}_{t,i}-\eta_T\sum_{t=1}^{T}\widetilde{\ell}_{t,s} &\overset{(b)}{\leq} \log\left(n+k\right)+\sum_{t=1}^{T}\frac{\eta_t^2}{2}\sum_{i=1}^{|\Pi^*|}q_{t,i}\left(\widetilde{\ell}_{t,i}\right)^2\\
\sum_{t=1}^{T}\sum_{i=1}^{|\Pi^*|}q_{t,i}\widetilde{\ell}_{t,i}-\sum_{t=1}^{T}\widetilde{\ell}_{t,s} &\overset{(c)}{\leq} \frac{\log|\Pi^*|}{\eta_T}+\frac{1}{\eta_T}\sum_{t=1}^{T}\frac{\eta_t^2}{2}\sum_{i=1}^{|\Pi^*|}q_{t,i}\left(\widetilde{\ell}_{t,i}\right)^2,
\end{aligned}
$$

where step (b) is by applying

$$
\eta_T\sum_{t=1}^{T}\sum_{i=1}^{|\Pi^*|}q_{t,i}\widetilde{\ell}_{t,i}-\eta_T\sum_{t=1}^{T}\widetilde{\ell}_{t,s}\leq\sum_{t=1}^{T}\eta_t\sum_{i=1}^{|\Pi^*|}q_{t,i}\widetilde{\ell}_{t,i}-\eta_T\sum_{t=1}^{T}\widetilde{\ell}_{t,s},
$$

and step (c) is by dividing η_T on both side.

Because we have

$$
\begin{aligned}
\mathbb{E}_T\left[q_{t,i}\left(\widetilde{\ell}_{t,i}\right)^2\right] &= q_{t,i}\mathbb{E}_T\left[\left(\pi_i\left(\boldsymbol{x}_t\right)\cdot\widehat{\boldsymbol{\ell}}_t\right)^2\right]\\
&= q_{t,i}\left(P\left(U_t=1\right)\left(\pi_i\left(\boldsymbol{x}_t\right)\cdot\frac{\boldsymbol{\ell}_t}{z_t}\right)^2+P\left(U_t=0\right)\cdot 0\right)\\
&= q_{t,i}\left(z_t\left(\pi_i\left(\boldsymbol{x}_t\right)\cdot\frac{\boldsymbol{\ell}_t}{z_t}\right)^2\right)\\
&= \frac{q_{t,i}}{z_t}\left(\pi_i\left(\boldsymbol{x}_t\right)\cdot\boldsymbol{\ell}_t\right)^2\\
&\leq \frac{q_{t,i}}{z_t}\pi_i\left(\boldsymbol{x}_t\right)\cdot\boldsymbol{\ell}_t\\
&= \frac{q_{t,i}}{z_t}\langle\pi_i\left(\boldsymbol{x}_t\right),\boldsymbol{\ell}_t\rangle,
\end{aligned}
$$

it leads to

$$
\begin{aligned}
\sum_{t=1}^{T}\sum_{i=1}^{|\Pi^*|}q_{t,i}\widetilde{\ell}_{t,i}-\sum_{t=1}^{T}\widetilde{\ell}_{t,s} &\leq \frac{\log|\Pi^*|}{\eta_T}+\frac{1}{\eta_T}\sum_{t=1}^{T}\frac{\eta_t^2}{2}\sum_{i=1}^{|\Pi^*|}\frac{q_{t,i}}{z_t}\langle\pi_i\left(\boldsymbol{x}_t\right),\boldsymbol{\ell}_t\rangle\\
&\overset{(d)}{\leq} \frac{\log|\Pi^*|}{\eta_T}+\frac{1}{\eta_T}\sum_{t=1}^{T}\frac{\eta_t^2}{2}\frac{\langle\boldsymbol{w}_t,\boldsymbol{\ell}_t\rangle}{z_t},
\end{aligned}
$$

where step (d) is by applying $\sum_{i=1}^{|\Pi^*|} q_{t,i}\langle \pi_i(\boldsymbol{x}_t), \boldsymbol{\ell}_t\rangle = \langle \boldsymbol{w}_t, \boldsymbol{\ell}_t\rangle$.

So we have,

$$\begin{aligned}
\sum_{t=1}^{T}\sum_{i=1}^{|\Pi^*|} q_{t,i}\widetilde{\ell}_{t,i} - \sum_{t=1}^{T}\widetilde{\ell}_{t,s} &\leq \frac{\log|\Pi^*|}{\eta_T} + \frac{1}{\eta_T}\sum_{t=1}^{T}\frac{\eta_t^2}{2}\frac{\langle \boldsymbol{w}_t, \boldsymbol{\ell}_t\rangle}{z_t} \\
&\overset{(e)}{\leq} \frac{\log|\Pi^*|}{\eta_T} + \frac{1}{\eta_T}\sum_{t=1}^{T}\frac{\eta_t^2}{2}\frac{1-p_{t,y_t}}{z_t} \\
&\overset{(f)}{\leq} \frac{\log|\Pi^*|}{\eta_T} + \frac{1}{\eta_T}\sum_{t=1}^{T}\frac{\eta_t^2}{2}\frac{1-p_{t,y_t}}{\mathcal{Y}_0\left(\left(1-p_{t,y_t}\right)p_{t,y_t} + \left(1-p_{t,y}\right)p_{t,y}\right)} \\
&\leq \frac{\log|\Pi^*|}{\eta_T} + \frac{1}{\eta_T}\sum_{t=1}^{T}\frac{\eta_t^2}{2}\frac{1}{\mathcal{Y}_0\left(p_{t,y_t} + \frac{1-p_{t,y}}{1-p_{t,y_t}}p_{t,y}\right)},
\end{aligned}$$

where step (e) is by using $\langle \boldsymbol{w}_t, \boldsymbol{\ell}_t\rangle = 1 - p_{t,y_t}$ and step (f) by using Lemma 7 and get lower bound of z_t as $\frac{1}{|\mathcal{Y}|\ln|\mathcal{Y}|}\left(p_{t,y_t}\left(1-p_{t,y_t}\right) + p_{t,y}\left(1-p_{t,y}\right)\right)$ and applying $\frac{1}{|\mathcal{Y}|\ln|\mathcal{Y}|} = \mathcal{Y}_0$.

If $p_{t,y_t} \geq \frac{1}{|\mathcal{Y}|}$,

$$p_{t,y_t} + \frac{1-p_{t,y}}{1-p_{t,y_t}}p_{t,y} \geq \frac{1}{|\mathcal{Y}|}.$$

If $p_{t,y_t} < \frac{1}{|\mathcal{Y}|}$, $\exists y, p_{t,y} \to 1$, $\delta_1^t = 1 - \max_{y,\tau\in[t]} p_{\tau,y}$. Let $p_{t,\hat{y}} = \max_y p_{t,y}$. Thus, we have $w_{\hat{y}} > \frac{1}{|\mathcal{Y}|}$ and

$$p_{t,y_t} + \frac{1-p_{t,y}}{1-p_{t,y_t}}p_{t,y} \geq p_{t,y_t} + w_{\hat{y}}\frac{\delta_1^t}{1-p_{t,y_t}} \geq 0 + \frac{1}{|\mathcal{Y}|}\frac{\delta_1^t}{1} = \frac{\delta_1^t}{|\mathcal{Y}|}.$$

Therefore

$$\max\{p_{t,y_t} + p_{t,y}\frac{1-p_{t,y}}{1-p_{t,y_t}}\} = \begin{cases} \frac{1}{|\mathcal{Y}|} & \text{if } p_{t,y_t} \geq \frac{1}{|\mathcal{Y}|}, \\ \frac{\delta_1^t}{|\mathcal{Y}|} & \text{if } p_{t,y_t} < \frac{1}{|\mathcal{Y}|}. \end{cases}$$

So we have

$$\begin{aligned}
\sum_{t=1}^{T}\sum_{i=1}^{|\Pi^*|} q_{t,i}\widetilde{\ell}_{t,i} - \sum_{t=1}^{T}\widetilde{\ell}_{t,s} &\le \frac{\log|\Pi^*|}{\eta_T} + \frac{1}{\eta_T}\sum_{t=1}^{T}\frac{\eta_t^2}{2}\frac{1}{\mathcal{Y}_0\left(p_{t,y_t} + \frac{1-\boldsymbol{w}_y}{1-p_{t,y_t}}p_{t,y}\right)} \\
&\overset{(g)}{\le} \frac{\log|\Pi^*|}{\eta_T} + \frac{1}{\eta_T}\sum_{t=1}^{T}\frac{\eta_t^2}{2}\frac{1}{\max\{\mathcal{Y}_0\frac{\delta_1^t}{|\mathcal{Y}|}, \delta_0^t\}} \\
&= \frac{\log|\Pi^*|}{\eta_T} + \frac{1}{\eta_T}\sum_{t=1}^{T}\frac{\eta_t^2}{2}\frac{|\mathcal{Y}|^2\ln|\mathcal{Y}|}{\max\{\delta_1^t, \delta_0^t|\mathcal{Y}|^2\ln|\mathcal{Y}|\}} \\
&\overset{(h)}{\le} \frac{\log|\Pi^*|}{\eta_T} + \frac{1}{\eta_T}\sum_{t=1}^{T}\frac{\eta_t^2}{2}\cdot\frac{|\mathcal{Y}|^2\ln|\mathcal{Y}|}{\frac{\delta_1^t+\delta_0^t|\mathcal{Y}|^2\ln|\mathcal{Y}|}{2}} \\
&= \frac{\log|\Pi^*|}{\eta_T} + \frac{1}{\eta_T}\sum_{t=1}^{T}\eta_t^2\frac{1}{\delta_1^t+\delta_0^t|\mathcal{Y}|^2\ln|\mathcal{Y}|}\cdot|\mathcal{Y}|^2\ln|\mathcal{Y}|,
\end{aligned}$$

where step (g) is by getting the lower bound of z_t as $\frac{\delta_1^t}{|\mathcal{Y}|} \le \frac{1}{|\mathcal{Y}|}$, $\delta_0^t \le \frac{\delta_0^t}{1-p_{t,y_t}}$ and step (h) is by applying $\max\{A,B\} \ge \frac{A+B}{2}$.

Let us define $\rho_t \triangleq \min_{\tau\in[t]}\delta_1^\tau = 1 - \max_{c,\tau\in[t]} p_{t,y}^\tau$. We get

$$\mathbb{E}_T[R_T] \le \frac{\log|\Pi^*|}{\eta_T} + \frac{1}{\eta_T}\sum_{t=1}^{T}\log|\Pi^*|\cdot\frac{1}{T} \le \frac{2\log|\Pi^*|}{\eta_T}$$

Let $\eta_t = \sqrt{\frac{\rho_t+\delta_0^t|\mathcal{Y}|^2\ln|\mathcal{Y}|}{|\mathcal{Y}|^2\ln|\mathcal{Y}|}}\cdot\sqrt{\frac{\log|\Pi^*|}{T}}$, we obtain

$$\begin{aligned}
\mathbb{E}_T[R_T] &\le \frac{2\sqrt{\log|\Pi^*|}\cdot\sqrt{T}\cdot\sqrt{|\mathcal{Y}|^2\ln|\mathcal{Y}|}}{\sqrt{\rho_T+\delta_0^T|\mathcal{Y}|^2\ln|\mathcal{Y}|}} \\
&\le 2|\mathcal{Y}|\sqrt{\frac{T\ln|\mathcal{Y}|\log|\Pi^*|}{\rho_T}}
\end{aligned}$$

□

E.3 Proof of Theorem E.2

Proof of Theorem E.2. From Lemma 3, we get the following equation as the cumulative query cost

$$\mathbb{E}\left[\sum_{t=1}^{T}U_t\right] \le \mathbb{E}\left[\sum_{t=1}^{T}\left(\frac{1}{\sqrt{t}} + \frac{\sum_{y\in\mathcal{Y}}\langle\boldsymbol{w}_t, \boldsymbol{\ell}_t^y\rangle\log_{|\mathcal{Y}|}\frac{1}{\langle\boldsymbol{w}_t,\boldsymbol{\ell}_t^y\rangle}}{|\mathcal{Y}|}\right)\right].$$

Let us assume the expected total loss of best policy is $\tilde{L}_{T,*}$. Thus, from *Theorem E.1*, we get

$$\mathbb{E}\left[R\right] = \mathbb{E}\left[\sum_{t=1}^{T} r_t\right] \leq 2|\mathcal{Y}|\sqrt{\frac{T\ln|\mathcal{Y}|\log|\Pi^*|}{\rho_T}} + \tilde{L}_{T,*}.$$

We get our regret bound R proved in Theorem E.1 and plug the regret bound into the query complexity bound given by Lemma 5, we have

$$\begin{aligned}
\sum_{t=1}^{T} \frac{\sum_{y\in\mathcal{Y}}\langle \boldsymbol{w}_t, \boldsymbol{\ell}_t^y\rangle \log_{|\mathcal{Y}|} \frac{1}{\langle \boldsymbol{w}_t, \boldsymbol{\ell}_t^y\rangle}}{|\mathcal{Y}|} &\leq \frac{\left(2|\mathcal{Y}|\sqrt{\frac{T\ln|\mathcal{Y}|\log|\Pi^*|}{\rho_T}} + \tilde{L}_{T,*}\right)\left(\log_{|\mathcal{Y}|} \frac{T^2(|\mathcal{Y}|-1)}{\left(2|\mathcal{Y}|\sqrt{\frac{T\ln|\mathcal{Y}|\log|\Pi^*|}{\rho_T}}+\tilde{L}_{T,*}\right)^2}\right)}{|\mathcal{Y}|} \\
&\leq \frac{\left(2|\mathcal{Y}|\sqrt{\frac{T\ln|\mathcal{Y}|\log|\Pi^*|}{\rho_T}} + \tilde{L}_{T,*}\right)\left(\log_{|\mathcal{Y}|} T|\mathcal{Y}|\right)}{|\mathcal{Y}|} \\
&= \frac{\left(2|\mathcal{Y}|\sqrt{\frac{T\ln|\mathcal{Y}|\log|\Pi^*|}{\rho_T}} + \tilde{L}_{T,*}\right)\left(\log_{|\mathcal{Y}|} T + 1\right)}{|\mathcal{Y}|}.
\end{aligned}$$

Finally, by applying query complexity upper bound of Lemma 4, we get

$$\mathbb{E}\left[\sum_{t=1}^{T} U_t\right] \leq 2\sqrt{T} + \frac{\left(2|\mathcal{Y}|\sqrt{\frac{T\ln|\mathcal{Y}|\log|\Pi^*|}{\rho_T}} + \tilde{L}_{T,*}\right)\left(\log_{|\mathcal{Y}|} T + 1\right)}{|\mathcal{Y}|}.$$

Since the second term on the RHS dominates the upper bound, we have

$$O\left(\mathbb{E}\left[\sum_{t=1}^{T} U_t\right]\right) = O\left(\frac{\left(\sqrt{\frac{T\log|\Pi^*|}{\rho_T}} + \tilde{L}_{T,*}\right)(\ln T)}{|\mathcal{Y}|\ln(|\mathcal{Y}|)}\right).$$

□