A. Convergence Analysis

A.1. Convergence of Biased Gradient Descent

Consider a generalized gradient descent training algorithm with the following update rule for each iteration:

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \eta_t \boldsymbol{g}_t, \tag{12}$$

where $\boldsymbol{w}_t$ is the model parameter and η_t is the learning rate at the t-th iteration. $\boldsymbol{g}_t$ is the descent vector, while $\boldsymbol{g}_t$ is not necessarily equal to the true gradient $\nabla L(\boldsymbol{w}_t)$, with $L(\boldsymbol{w}_t) = \mathbb{E}_{\mathbf{x}\sim p_{\mathbf{x}}}[l(\mathbf{x}; \boldsymbol{w}_t)]$ as the task loss over the true data distribution $p_{\mathbf{x}}$.

We make the following smoothness assumption, which is commonly used in the convergence analysis [7].

Assumption 1 (β-smoothness)**.** $L(\boldsymbol{w}_t)$ is differentiable with respect to $\boldsymbol{w}_t$ and its gradient is β-Lipschitz: $\|\nabla L(\boldsymbol{w}_a) - \nabla L(\boldsymbol{w}_b)\| \leq \beta\|\boldsymbol{w}_a - \boldsymbol{w}_b\|$.

With this assumption, we can get the following theorem about the convergence of the generalized gradient descent.

Theorem 1. *A gradient descent algorithm with Eq. (12) as the update rule can achieve the following convergence rate with Assumption 1 and learning rate $\eta_t \leq 1/\beta$:*

$$\inf_{t\leq T} \|\nabla L(\boldsymbol{w}_t)\|^2 \leq \frac{2L(\boldsymbol{w}_1)}{\sum_{t=1}^{T}\eta_t} + \frac{\sum_{t=1}^{T}\eta_t\delta_t^2}{\sum_{t=1}^{T}\eta_t}, \tag{13}$$

where $\delta_t^2 = \|\boldsymbol{g}_t - \nabla L(\boldsymbol{w}_t)\|^2$ is the gradient shifting.

Proof. With Lipschitz smoothness in Assumption 1, we have:

$$\begin{aligned}
&L(\boldsymbol{w}_{t+1}) - L(\boldsymbol{w}_t)\\
\leq& -\eta_t\langle\nabla L(\boldsymbol{w}_t), \boldsymbol{g}_t\rangle + \frac{\beta\eta_t^2}{2}\|\boldsymbol{g}_t\|^2\\
=& -\eta_t\langle\nabla L(\boldsymbol{w}_t), \nabla L(\boldsymbol{w}_t) + \boldsymbol{g}_t - \nabla L(\boldsymbol{w}_t)\rangle\\
&+ \frac{\beta\eta_t^2}{2}\|\nabla L(\boldsymbol{w}_t) + \boldsymbol{g}_t - \nabla L(\boldsymbol{w}_t)\|^2\\
=&(-\eta_t + \frac{\beta\eta_t^2}{2})\|\nabla L(\boldsymbol{w}_t)\|^2 + \frac{\beta\eta_t^2}{2}\|\boldsymbol{g}_t - \nabla L(\boldsymbol{w}_t)\|^2\\
&+ (-\eta_t + \beta\eta_t^2)\langle\nabla L(\boldsymbol{w}_t), \boldsymbol{g}_t - \nabla L(\boldsymbol{w}_t)\rangle\\
=&(-\eta_t + \frac{\beta\eta_t^2}{2})\|\nabla L(\boldsymbol{w}_t)\|^2\\
&+ (-\eta_t + \beta\eta_t^2)\langle\nabla L(\boldsymbol{w}_t), \boldsymbol{g}_t - \nabla L(\boldsymbol{w}_t)\rangle + \frac{\beta\eta_t^2}{2}\delta_t^2.
\end{aligned}$$

Since $\eta_t \leq 1/\beta$, we have $-\eta_t + \beta\eta_t^2/2 \leq 0$ and $-\eta_t + \beta\eta_t^2 \leq 0$. We also have

$$\begin{aligned}
&\|\boldsymbol{g}_t - \nabla L(\boldsymbol{w}_t)\|^2 = \delta_t^2\\
\Rightarrow&\langle\nabla L(\boldsymbol{w}_t), \boldsymbol{g}_t\rangle = \frac{\|\boldsymbol{g}_t\|^2 + \|\nabla L(\boldsymbol{w}_t)\|^2 - \delta_t^2}{2}\\
&\geq \frac{\|\nabla L(\boldsymbol{w}_t)\|^2 - \delta_t^2}{2}\\
\Rightarrow&\langle\nabla L(\boldsymbol{w}_t), \boldsymbol{g}_t - \nabla L(\boldsymbol{w}_t)\rangle = \langle\nabla L(\boldsymbol{w}_t), \boldsymbol{g}_t\rangle - \|\nabla L(\boldsymbol{w}_t)\|^2\\
&\geq -\frac{\|\nabla L(\boldsymbol{w}_t)\|^2 + \delta_t^2}{2}.
\end{aligned}$$

And thus

$$\begin{aligned}
&L(\boldsymbol{w}_{t+1}) - L(\boldsymbol{w}_t)\\
\leq&(-\eta_t + \frac{\beta\eta_t^2}{2})\|\nabla L(\boldsymbol{w}_t)\|^2\\
&- \frac{-\eta_t + \beta\eta_t^2}{2}(\|\nabla L(\boldsymbol{w}_t)\|^2 + \delta_t^2) + \frac{\beta\eta_t^2}{2}\delta_t^2\\
=& -\frac{\eta_t}{2}\|\nabla L(\boldsymbol{w}_t)\|^2 + \frac{\eta_t}{2}\delta_t^2.
\end{aligned}$$

Rearanging the result, we get

$$\eta_t\|\nabla L(\boldsymbol{w}_t)\|^2 \leq 2(L(\boldsymbol{w}_t) - L(\boldsymbol{w}_{t+1})) + \eta_t\delta_t^2.$$

Taking the sum over t, we get

$$\sum_{t=1}^{T}\eta_t\|\nabla L(\boldsymbol{w}_t)\|^2 \leq 2(L(\boldsymbol{w}_1) - L(\boldsymbol{w}_{T+1})) + \sum_{t=1}^{T}\eta_t\delta_t^2.$$

Without loss of generality, we assume $L(\boldsymbol{w}) \geq 0$, and we divide both sides with $\sum\eta_t$ to get:

$$\frac{1}{\sum_{t=1}^{T}\eta_t}\sum_{t=1}^{T}\eta_t\|\nabla L(\boldsymbol{w}_t)\|^2 \leq \frac{2L(\boldsymbol{w}_1)}{\sum_{t=1}^{T}\eta_t} + \frac{\sum_{t=1}^{T}\eta_t\delta_t^2}{\sum_{t=1}^{T}\eta_t},$$

which yields the final result:

$$\inf_{t\leq T}\|\nabla L(\boldsymbol{w}_t)\|^2 \leq \frac{2L(\boldsymbol{w}_1)}{\sum_{t=1}^{T}\eta_t} + \frac{\sum_{t=1}^{T}\eta_t\delta_t^2}{\sum_{t=1}^{T}\eta_t}.$$

□

A.2. Convergence Analysis of ADAMAB

Without loss of Generality, we consider the case where we generate only one data in each iteration of gradient descent. And we omit the parameter $\boldsymbol{w}_t$ in the following for simplicity. We use the following UCB algorithm to determine which class C we should augment:

$$
\begin{aligned}
&C_t = \arg\min_C \quad \hat{\delta}_t^2(C) - \frac{\alpha}{\sqrt{n_{t-1}+1}}\sqrt{\frac{1}{n_{C,t-1}}},\\
&\hat{\delta}_t^2(C) = \|\frac{1}{n_{t-1}+1}\nabla \hat{L}_C + \frac{n_{t-1}}{n_{t-1}+1}\nabla L(\mathbb{D}_{t-1}) - \nabla \hat{L}\|_2^2,\\
&\nabla \hat{L}_C = \frac{1}{n_{C,t-1}} \sum_{\boldsymbol{x}\in\mathbb{D}_{C,t-1}} \nabla l(\boldsymbol{x}),\\
&\nabla \hat{L} = \frac{1}{K}\sum_{C=1}^{K} \nabla \hat{L}_C.
\end{aligned}
\tag{14}
$$

We made the following assumption in addition to Assumption 1:

Assumption 2 (ℓ_∞-bounded gradients)**.** The gradients for any data sample $\boldsymbol{x}$ have $\|\nabla l(\boldsymbol{x})\|_\infty \le G$.

Before we prove the convergence result, we first prove the following lemma that will be useful in our analysis:

Lemma 1 (Vector Hoeffding's Inequality)**.** *For a random vector $\mathbf{x} \in \mathbb{R}^d$ with n I.I.D. data samples $\{\hat{\boldsymbol{x}}_1, \hat{\boldsymbol{x}}_2, \cdots, \hat{\boldsymbol{x}}_n\}$, if $\|\mathbf{x}\|_\infty \le G$, for any positive constant T, we have*

$$
\begin{aligned}
&\Pr\left(\left\|\frac{1}{n}\sum_{i=1}^{n}\hat{\boldsymbol{x}}_i - \mathbb{E}[\mathbf{x}]\right\|_2 \ge \epsilon\right) \le \frac{1}{T^4},\\
&\textit{where} \quad \epsilon = \sqrt{\frac{2dG^2\log(2T^4 d)}{n}}.
\end{aligned}
\tag{15}
$$

Proof. Since

$$
\left\|\frac{1}{n}\sum_{i=1}^{n}\hat{\boldsymbol{x}}_i - \mathbb{E}[\mathbf{x}]\right\|_2 \ge \epsilon \Rightarrow \left\|\frac{1}{n}\sum_{i=1}^{n}\hat{\boldsymbol{x}}_i - \mathbb{E}[\mathbf{x}]\right\|_\infty \ge \epsilon/\sqrt{d}
$$

We have

$$
\begin{aligned}
&\Pr\left(\left\|\frac{1}{n}\sum_{i=1}^{n}\hat{\boldsymbol{x}}_i - \mathbb{E}[\mathbf{x}]\right\|_2 \ge \epsilon\right)\\
\le &\Pr\left(\left\|\frac{1}{n}\sum_{i=1}^{n}\hat{\boldsymbol{x}}_i - \mathbb{E}[\mathbf{x}]\right\|_\infty \ge \epsilon/\sqrt{d}\right)\\
= &\Pr\left(\bigcup_{j=1}^{d}\left|\frac{1}{n}\sum_{i=1}^{n}\hat{\boldsymbol{x}}_i^{(j)} - \mathbb{E}[\mathbf{x}^{(j)}]\right| \ge \epsilon/\sqrt{d}\right)\\
\le &\sum_{j=1}^{d}\Pr\left(\left|\frac{1}{n}\sum_{i=1}^{n}\hat{\boldsymbol{x}}_i^{(j)} - \mathbb{E}[\mathbf{x}^{(j)}]\right| \ge \epsilon/\sqrt{d}\right).
\end{aligned}
$$

With Hoeffding's Inequality, we have

$$
\Pr\left(\left|\frac{1}{n}\sum_{i=1}^{n}\hat{\boldsymbol{x}}_i^{(j)} - \mathbb{E}[\mathbf{x}^{(j)}]\right| \ge \epsilon/\sqrt{d}\right) \le 2e^{-\frac{\epsilon^2 n}{2G^2 d}} \le \frac{1}{dT^4}.
$$

Thus, we get

$$
\Pr\left(\left\|\frac{1}{n}\sum_{i=1}^{n}\hat{\boldsymbol{x}}_i - \mathbb{E}[\mathbf{x}]\right\|_2 \ge \epsilon\right) \le \frac{1}{T^4}.
$$

□

Now we can prove the following convergence result:

Theorem 2 (Convergence of **ADAMAB**)**.** *Assuming Assumption 1 and 2, the gradient descent with **ADAMAB** given in Eq. (14) can achieve the following convergence with a constant learning rate $\eta_t = \eta \le 1/\beta$ and some properly selected positive constant α:*

$$
\begin{aligned}
&\inf_{t\le T}\mathbb{E}\|\nabla L(\boldsymbol{w}_t)\|^2\\
\le&\mathcal{O}\left(\frac{1}{T}\right) + \mathcal{O}\left(\sqrt{\frac{\log(T)}{T}}\right) + \sup_t \inf_C \delta_t^2(C).
\end{aligned}
\tag{16}
$$

Proof. We first derive the confidence bound of estimating $\delta_t(C)$ for any class $C \in \{1, 2, ..., K\}$. We notice that

$$
\begin{aligned}
&|\hat{\delta}_t(C) - \delta_t(C)|\\
=&\Big|\|\frac{1}{n_{t-1}+1}\nabla \hat{L}_C + \frac{n_{t-1}}{n_{t-1}+1}\nabla L(\mathbb{D}_{t-1}) - \nabla \hat{L}\|_2\\
&- \|\frac{1}{n_{t-1}+1}\nabla L_C + \frac{n_{t-1}}{n_{t-1}+1}\nabla L(\mathbb{D}_{t-1}) - \nabla L\|_2\Big|\\
\le&\|\frac{1}{n_{t-1}+1}(\nabla \hat{L}_c - \nabla L_c) - (\nabla \hat{L} - \nabla L)\|_2\\
\le&\frac{1}{n_{t-1}+1}\|\nabla \hat{L}_c - \nabla L_c\|_2 + \|\nabla \hat{L} - \nabla L\|_2.
\end{aligned}
$$

And thus we have

$$
\begin{aligned}
&\Pr(\|\nabla \hat{L}_c - \nabla L_c\|_2 \le (1+n_{t-1})\epsilon_1(C)\\
&\qquad \cap \|\nabla \hat{L} - \nabla L\|_2 \le \epsilon_2)\\
&\quad \le \Pr(|\hat{\delta}_t(C) - \delta_t(C)| \le \epsilon_1(C) + \epsilon_2)\\
&\Rightarrow \Pr(|\hat{\delta}_t(C) - \delta_t(C)| \ge \epsilon_1(C) + \epsilon_2)\\
&\quad \le \Pr(\|\nabla \hat{L}_c - \nabla L_c\|_2 \ge (1+n_{t-1})\epsilon_1(C)\\
&\qquad \cup \|\nabla \hat{L} - \nabla L\|_2 \ge \epsilon_2)\\
&\quad \le \Pr(\|\nabla \hat{L}_c - \nabla L_c\|_2 \ge (1+n_{t-1})\epsilon_1(C))\\
&\qquad + \Pr(\|\nabla \hat{L} - \nabla L\|_2 \ge \epsilon_2)
\end{aligned}
\tag{17}
$$

For the first term in Eq. (17), with Lemma 1, we have:

$$
\begin{aligned}
&\Pr(\|\nabla \hat{L}_c - \nabla L_c\|_2 \ge (n_{t-1}+1)\epsilon_1(C))\\
&\le \Pr(\|\nabla \hat{L}_c - \nabla L_c\|_2 \ge \sqrt{n_{t-1}+1}\epsilon_1(C)) \le \frac{1}{T^4},\\
&\text{where} \quad \epsilon_1(C) = \frac{1}{\sqrt{n_{t-1}+1}}\sqrt{\frac{2dG^2\log(2T^4 d)}{n_{C,t-1}}}.
\end{aligned}
\tag{18}
$$

And for the second term, we have

$$\|\nabla\hat{L}-\nabla L\|_2=\|\frac{1}{K}\sum_{C=1}^{K}(\nabla\hat{L}_C-\nabla L_C)\|_2 \\ \leq\frac{1}{K}\sum_{C=1}^{K}\|\nabla\hat{L}_C-\nabla L_C\|_2.$$

And thus,

$$\forall C,\|\nabla\hat{L}_C-\nabla L_C\|_2\leq\epsilon_C \\ \Rightarrow\|\nabla\hat{L}-\nabla L\|_2\leq\frac{1}{K}\sum_{C=1}^{K}\epsilon_C.$$

This leads to

$$\begin{aligned}&\Pr(\bigcap_{C=1}^{K}\|\nabla\hat{L}_C-\nabla L_C\|_2\leq\epsilon_C)\\ &\leq\Pr(\|\nabla\hat{L}-\nabla L\|_2\leq\frac{1}{K}\sum_{C=1}^{K}\epsilon_C)\\ \Rightarrow&\Pr(\|\nabla\hat{L}-\nabla L\|_2\geq\frac{1}{K}\sum_{C=1}^{K}\epsilon_C)\\ &\leq\Pr(\bigcup_{C=1}^{K}\|\nabla\hat{L}_C-\nabla L_C\|_2\geq\epsilon_C)\\ &\leq\sum_{C=1}^{K}\Pr(\|\nabla\hat{L}_C-\nabla L_C\|_2\geq\epsilon_C).\end{aligned}$$

With Lemma 1, we have

$$\Pr(\|\nabla\hat{L}_C-\nabla L_C\|_2\geq\epsilon_C)\leq\frac{1}{KT^4}, \\ \text{with}\quad\epsilon_C=\sqrt{\frac{2dG^2\log(2dKT^4)}{n_C}}.$$

Thus if we choose

$$\epsilon_2=\frac{1}{K}\sum_{C=1}^{K}\sqrt{\frac{2dG^2\log(2dKT^4)}{n_C}},$$

we achieve

$$\Pr(\|\nabla\hat{L}-\nabla L\|_2\geq\epsilon_2)\leq\frac{1}{T^4}. \tag{19}$$

Plugging Eq. (18) and Eq. (19) into Eq. (17), we get the confidence bound for $\hat{\delta}_t$:

$$\Pr(|\hat{\delta}_t(C)-\delta_t(C)|\geq\epsilon_1(C)+\epsilon_2)\leq\frac{2}{T^4}$$

with

$$\epsilon_1(C)=\frac{1}{\sqrt{n_{t-1}+1}}\sqrt{\frac{2dG^2\log(2T^4d)}{n_{C,t-1}}}, \\ \epsilon_2=\frac{1}{K}\sum_{C=1}^{K}\sqrt{\frac{2dG^2\log(2dKT^4)}{n_{C,t-1}}}. \tag{20}$$

We can easily extend this result to get the confidence bound of $\hat{\delta}_t^2$, since

$$|\hat{\delta}_t^2-\delta_t^2|=(\hat{\delta}_t+\delta_t)|\hat{\delta}_t-\delta_t|.$$

And we also have

$$\begin{aligned}&\hat{\delta}_t\\ =&\|\frac{1}{n_{t-1}+1}\nabla\hat{L}_C+\frac{n_{t-1}}{n_{t-1}+1}\nabla L(\mathbb{D}_{t-1})-\nabla\hat{L}\|_2\\ \leq&\|\frac{1}{n_{t-1}+1}\nabla\hat{L}_C\|_2+\|\frac{n_{t-1}}{n_{t-1}+1}\nabla L(\mathbb{D}_{t-1})\|_2+\|\nabla\hat{L}\|_2\\ \leq&2\sqrt{d}G,\end{aligned} \tag{21}$$

and similarly for δ_t. Accordingly, we have

$$\Pr(|\hat{\delta}_t^2-\delta_t^2|\leq4\sqrt{d}G(\epsilon_1(C)+\epsilon_2)) \\ \geq\Pr(|\hat{\delta}_t-\delta_t|\leq\epsilon_1(C)+\epsilon_2))\geq1-\frac{2}{T^4}.$$

With the confidence bound, we can prove the instantaneous regret at each iteration as follows. Assuming that the optimal class selection is

$$C^*=\arg\min_C\delta_t^2(C).$$

Let $\epsilon_t(C)=4\sqrt{d}G(\epsilon_1(C)+\epsilon_2)$. Notice that in Eq. (14), the selection algorithm is the same as we select with $\hat{\delta}_t^2(C)-\epsilon_t(C)$ as the objective function when $\alpha=4dG^2\sqrt{2\log(2T^4d)}$. We select C instead of C^* meaning that:

$$\hat{\delta}_t^2(C)-\epsilon_t(C)\leq\hat{\delta}_t^2(C^*)-\epsilon_t(C^*) \\ \Rightarrow\hat{\delta}_t^2(C)-\hat{\delta}_t^2(C^*)\leq\epsilon_t(C)-\epsilon_t(C^*).$$

Accordingly, we have the following instantaneous regret:

$$\begin{aligned}&\mathbb{E}[\delta_t^2(C)-\delta_t^2(C^*)]\\ =&\mathbb{E}[\delta_t^2(C)-\hat{\delta}_t^2(C)+\hat{\delta}_t^2(C)-\hat{\delta}_t^2(C^*)+\hat{\delta}_t^2(C^*)-\delta_t^2(C^*)]\\ \leq&\epsilon_t(C)+\epsilon_t(C)-\epsilon_t(C^*)+\epsilon_t(C^*)\\ =&2\epsilon_t(C).\end{aligned} \tag{22}$$

We assert that $\forall C,C'\in\{1,2,...,K\}$ and $\forall t\leq T$, there exists a positive constant M such that $n_{C,t-1}/n_{C',t-1}\leq$

M. Assume that there exist two classes C and C' with $n_{C,t-1} = Mn_{C',t-1}$. To select C instead of C' according to Eq. (14), we must have

$$\begin{aligned}
&\hat{\delta}_t^2(C) - \frac{\alpha}{\sqrt{n_{t-1}+1}}\sqrt{\frac{1}{n_{C,t-1}}} \\
&\le \hat{\delta}_t^2(C') - \frac{\alpha}{\sqrt{n_{t-1}+1}}\sqrt{\frac{1}{n_{C',t-1}}} \\
\Rightarrow &\hat{\delta}_t^2(C') - \hat{\delta}_t^2(C) \\
&\ge \frac{\alpha}{\sqrt{n_{t-1}+1}}\sqrt{\frac{1}{n_{C',t-1}}} - \frac{\alpha}{\sqrt{n_{t-1}+1}}\sqrt{\frac{1}{n_{C,t-1}}} \\
&= \frac{\alpha}{\sqrt{(n_{t-1}+1)n_{C',t-1}}}(1 - \sqrt{\frac{1}{M}}) \\
&> \frac{\alpha\sqrt{M+1}}{n_{t-1}+1}(1 - \sqrt{\frac{1}{M}})
\end{aligned} \tag{23}$$

Eq. (23) comes from the fact that $n_{C,t-1} + n_{C',t-1} = (1 + M)n_{C',t-1} \le n_{t-1} < n_{t-1} + 1$. For the left-hand side, we have

$$\begin{aligned}
&\hat{\delta}_t^2(C') - \hat{\delta}_t^2(C) \\
\le &(\hat{\delta}_t(C') + \hat{\delta}_t(C))|\hat{\delta}_t(C') - \hat{\delta}_t(C)| \\
\le &\frac{4\sqrt{d}G}{n_{t-1}+1}\|\nabla\hat{L}_C - \nabla\hat{L}_{C'}\| \qquad (24)\\
\le &\frac{8dG^2}{n_{t-1}+1}
\end{aligned}$$

where Eq. (24) comes from Eq. (21) and triangle inequality, while the last inequality comes from Assumption 2. Accordingly, if M satisfies

$$8dG^2 \le \alpha\sqrt{M+1}(1 - \sqrt{\frac{1}{M}}),$$

we can guarantee that $n_{C,t-1}/n_{C',t-1} \le M$. We can simplify this condition as

$$M \ge \left(1 + \sqrt{\frac{2}{\log(2T^4 d)}}\right)^2. \tag{25}$$

We further set $M = (1 + \sqrt{2/\log(2d)})^2$ to get rid of T, then we can get that

$$\sum_{C=1}^{K} n_{C,t-1} = t - 1 + n_0 \Rightarrow \forall C, n_{C,t-1} \ge \frac{t-1+n_0}{M(K-1)+1}. \tag{26}$$

With Eq. (20), Eq. (22) and Eq. (26), we get the instantaneous regret as

$$\mathbb{E}[\delta_t^2(C) - \delta_t^2(C^*)] \le 2\epsilon_t(C) \le \mathcal{O}(\sqrt{\frac{\log(T)}{t}}), \tag{27}$$

Taking Expectation over the result of Theorem 1 and plugging Eq. (27) in, we complete the proof with the fact that

$$\frac{\sum_{t=1}^{T}\eta_t\delta_t^2(C_t^*)}{\sum_{t=1}^{T}\eta_t} \le \sup_t \delta_t^2(C_t^*) = \sup_t \inf_C \delta_t^2(C).$$

□

Remark 1. It is noteworthy that we relax the confidence bound $\epsilon_1(C)$ in Eq. (18), whose tight confidence bound should be

$$\epsilon_1^*(C) = \frac{1}{n_{t-1}+1}\sqrt{\frac{2dG^2\log(2T^4 d)}{n_{C,t-1}}}.$$

We notice that the tight confidence bound $\epsilon_1^*(C)$ decreases so fast that some classes cannot be guaranteed to be sufficiently explored to ensure a constant M as given in Eq. (25). In fact, the M needs to be proportional to t with $\epsilon_1^*(C)$, which makes $\epsilon_t(C)$ not able to decrease along t as in Eq. (27), and thus the convergence cannot be achieved.

B. Experimental Settings

All the experiments are completed on a single MacBook Pro with a single M4 Max chip and 36GB memory.

B.1. Training Hyperparameters

In all the calibration methods, we adopt an Adam Optimizer [29] with initial learning rate $\eta_0 = 0.005$ and a weight decay $\lambda = 0.0001$. We use a Cosine Annealing learning rate scheduler with $\eta_T = 0.5\eta_0$. For the other hyperparameters, we summarize them in Tab. 4. "# Training Rounds" means the total number of training epochs we conducted, and "# Aug. Rounds" denotes the number of augmentation epochs, i.e., we only generate data in the first "# Aug. Rounds", and then we do not generate but only train the model until "# Training Rounds".

We conduct experiments on a wide range of hyperparaters and select the best hyperparameters with the highest accuracy. Specifically, we tried initial learning rate from $\{0.01, 0.005, 0.001, 0.0005\}$ and batch size from $\{16, 32, 64, 128, 256, 512\}$. We also tried different α from $\{0, 1, 3, 10, 30, 100\}$ and different number of augmentation rounds from $\{0, K, 2K, 3K\}$ where K is the number of classes. We report the results with different α and number of augmentation rounds in Sec. 4.3.

B.2. Prompt Templates

We use one text dataset and one image dataset as examples to demonstrate how to construct prompt templates for in-context learning, reranking, embedding, and data augmentation, respectively. Notice that for the few-shot setting, we

Table 4. The hyperparameters for the results in Tab. 2 and Tab. 3.

Dataset	MultiWD	FQS	TREC	OxfordPets	Flowers102	CUB200
Batch Size	32	32	64	64	128	256
# Training Rounds	30	50	100	100	300	500
Δn	5	5	5	3	2	2
# Aug. Rounds	18	39	60	74	102	400
α	100	100	30	100	30	100

will add all the examples in the initial training set to the description of each label, while no examples will be included for the zero-shot setting.

Forbidden Question Set (FQS):

```
in_context_learning_prompt: |
   You are a Responsible AI system checking if a
       user query complies with certain safety
       guidelines and application-specific rules.
        Each rule will be provided with a
       detailed description and some example
       queries that violate the rule. You need to
        judge whether a query provided by the
       user violates any of the rules and which
       one it is most likely to violate.

   Given the following rules with their
       corresponding indices:

   1. {label1}: {description1}
   2. {label2}: {description2}
   ...
   K. {labelK}: {descriptionK}

   Evaluate the following user query carefully
       against the rules:

   User Query: {query}

   Please make a judgment which rule this query
       violates. Output your judgment as a single
        integer: violated_rule_index. Do not
       include anything else except for
       violated_rule_index in your response.
   If none of the rules are violated, return -1
       as the violated_rule_index.

reranker_query_prompt: |
   Please find the rule that the following query
       violates:
   {query}

reranker_document_prompt: |
   Rule:
   {label}: {description}

embedding_query_prompt: |
   {query}

embedding_label_prompt: |
   Rule:
   {label}: {description}

augmentation_prompt: |
   Here is a rule that defines the topics of the
       queries that should not be answered by an
       LLM agent:
   {label}: {description}

   Please generate {num_generate} **diverse**
       example queries that violate this rule.
   The examples should be as **diverse** as
       possible so that the agent can learn from
       the examples if a query violates the rule.
        In addition, do not generate examples
       that are similar to the following existing
        examples:
   {existing_queries}

   In your response, only list the examples with
       one example in each line, without any
       numbering or bullets.
```

Listing 1. FQS prompts.

OxfordIIITPet (OxfordPets):

```
in_context_learning_prompt: |
   You will be given an image and asked to
       identify the species of the pet in the
       image. A predefined set of pet species
       will be provided, and you need to select
       one species from them for the pet in the
       given image.

   Given the following pet species with their
       corresponding indices:

   1. {label1}
   2. {label2}
   ...
   K. {labelK}

   Check the image carefully to determine which
       species the pet belongs to.

   Output your decision as a single integer:
       species_index. Do not include anything
       else except for species_index in your
       response.
   If none of the species apply, return -1 as the
        species_index.

reranker_query_prompt: |
   {query}

reranker_document_prompt: |
   A picture of a pet belonging to {label}
       species.

embedding_query_prompt: |
   {query}

embedding_label_prompt: |
   A picture of a pet belonging to {label}
       species.

augmentation_prompt: |
   Generate pictures of a pet belonging to {label
       } species in different scenarios.
```

Listing 2. OxfordPets prompts.

Table 5. Accuracy comparison (%) between open- and closed-source generators for text and image datasets.

Generator	TREC	FQ	multiWD
Qwen/Qwen3-32B	59.30%	87.10%	57.27%
Llama-3.1-70B-Instruct	61.19%	87.69%	58.59%
gpt-4o-mini	**61.63%**	**89.85%**	**59.91%**
Generator	**OxfordPets**	**Flower102**	**CUB200**
Qwen-Image (20B)	92.13	91.76	61.53
GPT-Image-1-mini	**92.20**	**93.17**	**68.60**

Table 6. Sample Generation time cos per sample (in seconds) comparison between ADAMAB and random data generation .

Embedder / Dataset Generator	TREC GPT-4o-mini	OxfordPets GPT-Image-1-mini	TREC Llama-3.1-70B-Instruct	OxfordPets Qwen-Image (20B)
Random	0.28	4.26	0.23	6.15
ADAMAB	0.30	4.28	0.24	6.19

Table 7. Accuracy (%) for different calibrating networks.

Network	TREC	FQ	multiWD	OxfordPets	Flower102	CUB200
Linear Probe	59.74%	88.00%	53.30%	88.36%	91.77%	66.40%
LWNSN	**61.63%**	**89.85%**	**59.91%**	**93.20%**	**93.17%**	**68.80%**

Table 8. Accuracy of ADAMAB v.s. other acquisition strategies.

Acquisition	TREC	FQ	multiWD	OxfordPets	Flower102	CUB200
Entropy[20]	59.01%	87.38%	52.86%	91.23%	90.51%	59.08%
Cross-Entropy	57.99%	88.62%	53.74%	90.78%	90.61%	58.05%
ADAMAB	**61.63%**	**89.85%**	**59.91%**	**93.20%**	**93.17%**	**68.80%**

C. Additional Experiments

C.1. Comparison between Different Generators

We evaluate ADAMAB with both closed-source and open-source generators (Table 5). The results show that ADAMAB achieves comparable performance even when paired with smaller or older open-source generators, without noticeable degradation in accuracy. This suggests that the effectiveness of our approach is largely robust to the choice of generator.

C.2. Sample Generation Latency

We have compared the generation latency of ADAMAB's adaptive data augmentation against random generation using different generators (see Table 6). The results show only a marginal increase in generation time, indicating that the adaptive strategy incurs minimal additional overhead.

C.3. Comparison of Different Calibrators

We compare our Light-weight Neural Similarity Networks (LWNSN) with linear probing networks (see Table 7). The results show that our LWNSN consistently achieves higher accuracy while maintaining a lightweight architecture, demonstrating the effectiveness of the proposed design.

C.4. Comparison with Other Acquisition Strategies

We further compare our approach with several non-bandit adaptive strategies (see Table 8). The acquisition functions are given below:

1. Entropy [20]:

$$a_{\text{Entropy}}(C; \boldsymbol{w}_t, \mathbb{D}_{t-1}) = \frac{\sum_{\boldsymbol{x}\in\mathbb{D}_{C,t-1}} H(p(\mathrm{y}|\boldsymbol{x}, \boldsymbol{w}_t))}{n_{C,t-1}} \tag{28}$$

2. Cross-Entropy:

$$a_{\text{CE}}(C; \boldsymbol{w}_t, \mathbb{D}_{t-1}) = \frac{\sum_{\boldsymbol{x}\in\mathbb{D}_{C,t-1}} l_{\text{CE}}(p(\mathrm{y}|\boldsymbol{x}, \boldsymbol{w}_t), C)}{n_{C,t-1}} \tag{29}$$

The results demonstrate that combining LWNSN with our bandit-based adaptive strategy leads to higher accuracy, highlighting the advantage of our adaptive data augmentation mechanism.