

**Outline.** Additional technical details are given in Section A. The proofs are provided in Section B. Finally, we give additional empirical results in Section C.

## A TECHNICAL DETAILS

Additional technical details are presented here.

**Label combiner parameter search.** Recall that the label combiner requires two parameters: the combining weight  $w \in [0, 1]$  and the quality score threshold  $\theta \in [0, 1]$ . We adopt a simple grid search approach to select  $w$  and  $\theta$ . More precisely, we first create a parameter candidate set  $PCS \triangleq \{w_0, w_1, w_2, \dots, w_M\} \times \{\theta_0, \theta_1, \theta_2, \dots, \theta_M\}$ , where  $w_m = \frac{m}{M}$  and  $\theta_i = \frac{m}{M}$ . Next, for each  $(w, \theta) \in PCS$ , we evaluate the performance of combining the base service and the  $k$ th service using  $(w, \theta)$ , and select the parameter that gives the highest accuracy. Note that this involves  $M^2$  number of label combinations for each  $k \in [K]$ . In practice, we have found that  $M = 10$  is sufficient to obtain a good combiner.

**$\delta$  selection in Algorithm 1.** A naive approach is to set a small constant value, say,  $\delta = 0.01$ . To obtain a more accurate strategy, we can adopt a search algorithm to select the best  $\delta$  value based on the evaluation the performance on a validation dataset. More precisely, we first create a constant set  $CS$ . Then for each  $\alpha \in CS$ , let  $\delta = \alpha \frac{\log N}{N}$ , and then solve Problem 3.2 to obtain the parameter  $\hat{p}$ , evaluate the performance on a validation dataset. Finally, we select the  $\alpha \in CS$  that achieves the highest accuracy on the validation dataset. In practice, we have found that  $CS = \{-10, -9, -8, \dots, 0, 1, 2, \dots, 10\}$  is sufficient to obtain a highly accurate solution.

## B PROOFS

For ease of notations, let us introduce  $\hat{b} \triangleq b - c_{base}$  and  $\hat{c}_k \triangleq c_k \cdot \mathbb{1}_{k \neq base}$  first. Then we can rewrite the API selection problem (Problem 3.1) as

$$\begin{aligned} \max_{\mathbf{Z} \in \mathbb{R}^{N \times K}}: & \frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \mathbf{Z}_{n,k} \hat{\mathbf{a}}_k(x_n) \\ \text{s.t.} & \frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \mathbf{Z}_{n,k} \hat{\mathbf{c}}_k \leq \hat{b} \\ & \sum_{k=1}^K \mathbf{Z}_{n,k} = 1, \forall n; \mathbf{Z}_{n,k} \in \{0, 1\}, \forall n, k \end{aligned} \quad (\text{B.1})$$

Its corresponding linear programming simply becomes

$$\begin{aligned} \max_{\mathbf{Z} \in \mathbb{R}^{N \times K}}: & \frac{1}{N} \sum_{n=1}^N \mathbf{Z}_{n,k} \hat{\mathbf{a}}_k(x_n) \\ \text{s.t.} & \frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \mathbf{Z}_{n,k} \hat{\mathbf{c}}_k \leq \hat{b} \\ & \sum_{k=1}^K \mathbf{Z}_{n,k} = 1, \forall n; \mathbf{Z}_{n,k} \in [0, 1], \forall n, k \end{aligned} \quad (\text{B.2})$$

We will analyze some useful properties for those two problems first, and then prove the desired results for the original API selection problem on top of those properties.

### B.1 HELPFUL LEMMAS

Before proving the desired results, let us also provide a few generic lemmas.

**Lemma 3.** Let  $\mathbf{A} \in \mathbb{R}^{N_1 \times N_2}$  be a fixed matrix and  $\beta \in \mathbb{R}^{N_1}$  be a random vector. If  $\beta$  is supported on  $[0, 1]^{N_1}$  with a continuous density function, then with probability 1,

$$\min_{\mathbf{x}} \|\mathbf{Ax} - \beta\|_0 \geq N_1 - N_2.$$

*Proof.* If  $N_1 \leq N_2$  then the above inequality obviously holds. Suppose  $N_1 > N_2$ . We prove this by contradiction. Assume the inequality does not hold. Then there exists some  $\mathbf{x}'$ , such that with probability larger than 0,

$$\|\mathbf{Ax}' - \beta\|_0 < N_1 - N_2.$$

That is to say, at least  $N_1 - (N_1 - N_2) + 1 = N_2 + 1$  many equations in  $\mathbf{Ax}' = \beta$  can be forced to 0. Let  $U$  be the set of those  $N_2 + 1$  indexes. Then formally we have

$$\mathbf{A}_U \mathbf{x}' = \beta_U.$$

That is to say, with probability larger than 0,  $\beta_U$  is in the subspace formed by the columns of  $\mathbf{A}_U$ .

On the other hand, we can show that for any set of indexes  $V$  with  $|V| = N_2 + 1$ ,  $\beta_V$  lies in the subspace formed by the columns of  $\mathbf{A}_V$  with probability 0, which gives a contradiction. To see this, let us start by considering a fixed set of indexes  $V$ . Let  $\Omega_V$  denote the subspace formed by  $\mathbf{A}_V$  and  $p_{\beta_V}(\cdot)$  be the density function of  $\beta_V$ . The density function of  $\beta$  is continuous and thus  $p_{\beta_V}(\cdot)$  is also continuous. The support of  $\beta$  is in  $[0, 1]^{N_1}$ , and thus the support of  $\beta_V$  is in  $[0, 1]^{N_2+1}$  (since  $|V| = N_2 + 1$  by definition). That is to say,  $p_{\beta_V}(\cdot)$  is a continuous function on a compact set. Therefore,  $p_{\beta_V}(\cdot)$  must be bounded, i.e., there exists a constant  $p^{\sup}$  such that  $p_{\beta_V}(\cdot) \leq p^{\sup}$ . Hence we have

$$\Pr[\beta_V \in \Omega_V] = \int_{\mathbf{x} \in \Omega_V} p_{\beta_V}(\mathbf{x}) d\mathbf{x} \leq \int_{\mathbf{x}_V \in \Omega_V} p^{\sup} d\mathbf{x} = p^{\sup} \int_{\mathbf{x}_V \in \Omega_V} 1 d\mathbf{x}$$

where the first equation is by definition of the random variable  $\beta_V$ , the inequality is by increasing the density function to its upper bound  $p^{\sup}$ , and the last equation simply moves the constant out of the integral. In addition,  $\Omega_V$  is a  $N_2 + 1$  dimensional space spanned by  $N_2$  vectors, which implies that its measure in  $\mathbb{R}^{N_2+1}$  is 0, i.e.,  $\int_{\mathbf{x}_V \in \Omega_V} 1 d\mathbf{x} = 0$ . Thus, we have just shown that

$$\Pr[\beta_V \in \Omega_V] \leq p^{\sup} \int_{\mathbf{x}_V \in \Omega_V} 1 d\mathbf{x} = 0$$

Probability is non-negative, and thus  $\Pr[\beta_V \in \Omega_V] = 0$  (for a fixed  $V$ ). Note that the size of  $V$  is  $N_2 + 1$  and there are in total  $N_1$  possible indexes. Thus, there are  $\binom{N_1}{N_2+1}$  many possible choices of  $V$ . Applying union bound, we have for any  $V$ ,  $\Pr[\beta_V \in \Omega_V] = 0$ . A contradiction. The assumption is incorrect, and thus we must have

$$\min_{\mathbf{x}} \|\mathbf{Ax} - \beta\|_0 \geq N_1 - N_2.$$

□

**Lemma 4.** Let  $f$  be a function defined on  $\Omega_{\mathbf{z}}$ . Assume there exists a set  $\Omega_{\mathbf{z},1} \subseteq \Omega_{\mathbf{z}}$ , such that for any  $\mathbf{z} \in \Omega_{\mathbf{z}}$ , there exists  $\mathbf{z}' \in \Omega_{\mathbf{z},1}$ , such that  $\|f(\mathbf{z}) - f(\mathbf{z}')\| \leq \Delta$ . Then we have

$$\|f(\mathbf{z}^*) - f(\mathbf{z}_1^*)\| \leq \Delta,$$

where  $\mathbf{z}^* = \arg \max_{\mathbf{z} \in \Omega_{\mathbf{z}}} f(\mathbf{z})$ ,  $\mathbf{z}_1^* = \arg \max_{\mathbf{z} \in \Omega_{\mathbf{z},1}} f(\mathbf{z})$ .

*Proof.* By assumption, there exists a  $\mathbf{z}' \in \Omega_{\mathbf{z},1}$ , such that

$$\|f(\mathbf{z}^*) - f(\mathbf{z}')\| \leq \Delta$$

which implies

$$f(\mathbf{z}') \geq f(\mathbf{z}^*) - \Delta$$

Noting that  $\mathbf{z}_1^*$  is the optimal solution on  $\Omega_{\mathbf{z},1}$  and  $\mathbf{z}'$  is a feasible solution, we have

$$f(\mathbf{z}_1^*) \geq f(\mathbf{z}')$$

Combining the above two inequalities, we have

$$f(\mathbf{z}_1^*) \geq f(\mathbf{z}^*) - \Delta$$

On the other hand, since  $\Omega_{\mathbf{z},1} \subseteq \Omega_{\mathbf{z}}$ ,  $\mathbf{z}_1^*$  is a feasible solution on  $\Omega_{\mathbf{z}}$ , and thus we have

$$f(\mathbf{z}_1^*) \leq f(\mathbf{z}^*) \leq f(\mathbf{z}^*) + \Delta$$

Combing those two inequalities we have

$$\|f(\mathbf{z}_1^*) - f(\mathbf{z}^*)\| \leq \Delta$$

which completes the proof.  $\square$

**Lemma 5.** Let  $X_1, X_2, \dots, X_{N_1}$  and  $X'_1, X'_2, \dots, X'_{N_2}$  be two i.i.d. samples from the same distribution which lies in  $[x_{\inf}, x_{\sup}]$ . Then we have with probability  $1 - \epsilon$ ,

$$\left\| \frac{1}{N_2} \sum_{n=1}^{N_2} X'_n - \frac{1}{N_1} \sum_{n=1}^{N_1} X_n \right\| \leq (x_{\sup} - x_{\inf}) \left[ \sqrt{\frac{\log 4 - \log \epsilon}{2N_2}} + \sqrt{\frac{\log 4 - \log \epsilon}{2N_1}} \right].$$

*Proof.* We can apply the Hoeffding's inequality for both sequences separately, and we can obtain with probability  $1 - \epsilon$ ,

$$\left\| \frac{1}{N_1} \sum_{n=1}^{N_1} X_n - \mathbb{E}[X_1] \right\| \leq (x_{\sup} - x_{\inf}) \sqrt{\frac{\log 2 - \log \epsilon}{2N_1}}$$

and with probability  $1 - \epsilon$

$$\left\| \frac{1}{N_2} \sum_{n=1}^{N_2} X'_n - \mathbb{E}[X_1] \right\| \leq (x_{\sup} - x_{\inf}) \sqrt{\frac{\log 2 - \log \epsilon}{2N_2}}$$

Now applying union bound, we have with probability  $1 - \epsilon$ ,

$$\left\| \frac{1}{N_1} \sum_{n=1}^{N_1} X_n - \mathbb{E}[X_1] \right\| \leq (x_{\sup} - x_{\inf}) \sqrt{\frac{\log 4 - \log \epsilon}{2N_1}}$$

and  $1 - \epsilon$

$$\left\| \frac{1}{N_2} \sum_{n=1}^{N_2} X'_n - \mathbb{E}[X_1] \right\| \leq (x_{\sup} - x_{\inf}) \sqrt{\frac{\log 4 - \log \epsilon}{2N_2}}$$

Now applying the triangle inequality, we have

$$\left\| \frac{1}{N_2} \sum_{n=1}^{N_2} X'_n - \frac{1}{N_1} \sum_{n=1}^{N_1} X_n \right\| \leq (x_{\sup} - x_{\inf}) \left[ \sqrt{\frac{\log 4 - \log \epsilon}{2N_2}} + \sqrt{\frac{\log 4 - \log \epsilon}{2N_1}} \right]$$

which completes the proof.  $\square$

**Lemma 6.** Let  $f_1, f_2, g_1, g_2$  be functions defined on  $\Omega_{\mathbf{z}}$ , such that  $\max_{\mathbf{z} \in \Omega_{\mathbf{z}}} |(f_1 \mathbf{z}) - f_2(\mathbf{z})| \leq \Delta_1$  and  $\max_{\mathbf{z} \in \Omega_{\mathbf{z}}} \|g_2(\mathbf{z}) - g_1(\mathbf{z})\| \leq \Delta_2$ . Suppose

$$\begin{aligned} \mathbf{z}_1^* &= \arg \max_{\mathbf{z} \in \Omega_{\mathbf{z}}} f_1(\mathbf{z}) \\ &\text{s.t. } g_1(\mathbf{z}) \leq 0 \end{aligned}$$

and

$$\begin{aligned} \mathbf{z}_2^* &= \arg \max_{\mathbf{z} \in \Omega_{\mathbf{z}}} f_2(\mathbf{z}) \\ &\text{s.t. } g_2(\mathbf{z}) \leq \Delta_2, \end{aligned}$$

then we must have

$$\begin{aligned} f_1(\mathbf{z}_2^*) &\geq f_1(\mathbf{z}_1^*) - 2\Delta_1 \\ g_1(\mathbf{z}_2^*) &\leq 2\Delta_2. \end{aligned}$$

*Proof.* Note that  $\max_{\mathbf{z} \in \Omega_{\mathbf{z}}} |(f_1(\mathbf{z}) - f_2(\mathbf{z}))| \leq \Delta_1$  implies  $f_1(\mathbf{z}) \geq f_2(\mathbf{z}) - \Delta_1$  for any  $\mathbf{z} \in \Omega_{\mathbf{z}}$ . Specifically,

$$f_1(\mathbf{z}_2^*) \geq f_2(\mathbf{z}_2^*) - \Delta_1$$

Noting  $\max_{\mathbf{z} \in \Omega_{\mathbf{z}}} \|g_2(\mathbf{z}) - g_1(\mathbf{z})\| \leq \Delta_2$ , we have  $g_2(\mathbf{z}_1^*) \leq g_1(\mathbf{z}_1^*) - \Delta_2 \leq -\Delta_2$ , where the last inequality is due to  $g_1(\mathbf{z}_1^*) \leq 0$  by definition. Since,  $\mathbf{z}_1^*$  is a feasible solution to the second optimization problem, and the optimal value must be no smaller than the value at  $\mathbf{z}_1^*$ . That is to say,

$$f_2(\mathbf{z}_2^*) \geq f_2(\mathbf{z}_1^*)$$

Hence we have

$$f_1(\mathbf{z}_2^*) \geq f_2(\mathbf{z}_2^*) - \Delta_1 \geq f_2(\mathbf{z}_1^*) - \Delta_1$$

In addition,  $\max_{\mathbf{z} \in \Omega_{\mathbf{z}}} |(f_1(\mathbf{z}) - f_2(\mathbf{z}))| \leq \Delta_1$  implies  $f_2(\mathbf{z}) \geq f_1(\mathbf{z}) - \Delta_1$  for any  $\mathbf{z} \in \Omega_{\mathbf{z}}$ . Thus, we have  $f_2(\mathbf{z}_1^*) \geq f_1(\mathbf{z}_1^*) - \Delta_1$  and thus

$$f_1(\mathbf{z}_2^*) \geq f_2(\mathbf{z}_1^*) - \Delta_1 \geq f_1(\mathbf{z}_1^*) - 2\Delta_1$$

By  $\max_{\mathbf{z} \in \Omega_{\mathbf{z}}} |g_1(\mathbf{z}) - g_2(\mathbf{z})| \leq \Delta_2$ , we must have  $g_1(\mathbf{z}_2^*) \leq g_2(\mathbf{z}_2^*) + \Delta_2 \leq 2\Delta_2$ , where the last inequality is by definition of  $\mathbf{z}'$ , which completes the proof.  $\square$

## B.2 PROOF OF THEOREM 1

*Proof.* We give a constructive proof via explicitly giving the value of  $p^*$ . In fact, let  $p^*$  and  $\mathbf{q}^*$  be the optimal solution to

$$\begin{aligned} \min_{p, \mathbf{q}} \quad & \hat{b}p + \sum_{n=1}^N \mathbf{q}_n \\ \text{s.t.} \quad & \frac{1}{N} \hat{\mathbf{c}}_k p + \mathbf{q}_n \geq \frac{1}{N} \hat{\mathbf{a}}_k(x_n) \\ & p, \mathbf{q} \geq 0 \end{aligned} \tag{B.3}$$

Then our goal is to show that for this constructed  $p^*$ ,  $s^{p^*}$  is a feasible solution to Problem 3.1 and  $r(s^{p^*}) \geq r(s^*) - \frac{1}{N}$  with probability 1 (Since probabilistic statement is only introduced in Lemma 3 whose result holds with probability 1, and we only apply it finite times, we will omit the probabilistic statement for the rest of the proof for simplicity). To achieve this, let us construct a  $N \times K$  matrix

$$\tilde{\mathbf{Z}}_{n,k}^{p^*} \triangleq \mathbb{1}_{s^{p^*}(x_n)=k}$$

It is not hard to see that  $s^{p^*}(x_n) = \arg \max_k \tilde{\mathbf{Z}}_{n,k}^{p^*}$ . By construction of  $s^{p^*}$ , feasibility of  $s^{p^*}$  to Problem 3.1 is equivalent to feasibility of  $\tilde{\mathbf{Z}}^{p^*}$  to Problem B.1. By construction of  $s^*$  and  $s^{p^*}$ ,  $r(s^{p^*}) \geq r(s^*) - \frac{1}{N}$  is equivalent to

$$\frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \tilde{\mathbf{Z}}_{n,k}^{p^*} \hat{\mathbf{a}}_k(x_n) \geq \frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \mathbf{Z}_{n,k}^* \hat{\mathbf{a}}_k(x_n) - \frac{1}{N}.$$

Therefore, our goal becomes showing the feasibility of  $\tilde{\mathbf{Z}}^{p^*}$  and the above inequality. By construction of  $\tilde{\mathbf{Z}}^{p^*}$ , the natural constraints ( $\tilde{\mathbf{Z}}_{n,k}^{p^*} \in \{0, 1\}$  and  $\sum_{k=1}^K \tilde{\mathbf{Z}}_{n,k}^{p^*} = 1, \forall n$ ) are obviously satisfied. Thus, we only need to show  $\tilde{\mathbf{Z}}^{p^*}$  satisfies the budget constraint and the above inequality. To show those two results, let us introduce another variable  $\mathbf{Z}^{*,LP}$ , which represents a sparse optimal solution to the relaxed version of Problem B.1 (i.e., Problem B.2). The proof idea is then (roughly) to show (i) that  $\tilde{\mathbf{Z}}^{p^*}$  is actually close to  $\mathbf{Z}^{*,LP}$ , (ii) that  $\mathbf{Z}^{*,LP}$  satisfies the budget constraint and gives an estimated accuracy as high as that of the optimal solution  $\mathbf{Z}^*$ , and (iii) that the difference between  $\tilde{\mathbf{Z}}^{p^*}$  and  $\mathbf{Z}^{*,LP}$  does not break the budget constraints and only decreases the estimated accuracy by  $1/N$ . Combining the three points finishes the proof. Now we formalize this idea.

Step 1: We first show that  $\tilde{\mathbf{Z}}^{p^*}$  and  $\mathbf{Z}^{*,LP}$  are close to each other.

**Lemma 7.** Let  $\mathbf{Z}^{*,LP}$  be an optimal solution to Problem B.2. Then there exists some constant  $n'$ , such that  $\tilde{\mathbf{Z}}_{n,\cdot}^{p^*} = \mathbf{Z}_{n,\cdot}^{*,LP}, \forall n \neq n'$ .

*Proof.* Note that Problem B.3 is the dual problem to Problem B.2. We can write the complementary slackness constraints as follows

$$\mathbf{Z}_{n,k}^{*,LP} \left( \frac{1}{N} \hat{\mathbf{c}}_k p^* + \mathbf{q}_n^* - \frac{1}{N} \hat{\mathbf{a}}_k(x_n) \right) = 0, \forall n, k$$

Now let us construct the matrix

$$\mathbf{A} = \begin{bmatrix} \frac{1}{N} \hat{\mathbf{c}}, & \mathbf{1}, & \mathbf{0}, & \cdots, & \mathbf{0} \\ \frac{1}{N} \hat{\mathbf{c}}, & \mathbf{0}, & \mathbf{1}, & \cdots, & \mathbf{0} \\ \vdots, & \vdots, & \vdots, & \ddots, & \vdots \\ \frac{1}{N} \hat{\mathbf{c}}, & \mathbf{0}, & \mathbf{0}, & \cdots, & \mathbf{1} \end{bmatrix} \in \mathbb{R}^{NK \times (N+1)}$$

and the vector

$$\boldsymbol{\beta} = \frac{1}{N} \begin{bmatrix} \hat{\mathbf{a}}(x_1) \\ \hat{\mathbf{a}}(x_2) \\ \vdots \\ \hat{\mathbf{a}}(x_N) \end{bmatrix} \in \mathbb{R}^{NK}.$$

Then by Lemma 3,  $\min_{\mathbf{x}} \|\mathbf{Ax} - \boldsymbol{\beta}\|_0 \geq NK - N - 1$ . Specifically, if  $\mathbf{x} = [p^*, \mathbf{q}^{*T}]^T$ , then we should have  $\|\mathbf{Ax} - \boldsymbol{\beta}\|_0 \geq NK - N - 1$ . Note that each row of  $\mathbf{Ax} - \boldsymbol{\beta}$  corresponds to  $\frac{1}{N} \hat{\mathbf{c}}_k p^* + \mathbf{q}_n^* - \frac{1}{N} \hat{\mathbf{a}}_k(x_n)$ , and thus we effectively have

$$\frac{1}{N} \hat{\mathbf{c}}_k p^* + \mathbf{q}_n^* - \frac{1}{N} \hat{\mathbf{a}}_k(x_n) \neq 0$$

for at least  $NK - N - 1$  choices of  $n, k$ . In other words, among all possible choices of  $n, k$ , at most  $N + 1$  many of them satisfies

$$\frac{1}{N} \hat{\mathbf{c}}_k p^* + \mathbf{q}_n^* - \frac{1}{N} \hat{\mathbf{a}}_k(x_n) = 0$$

Furthermore, note that the constraint  $\sum_{k=1}^K \mathbf{z}_k^{*,LP}(x_n) = 1$  ensures that for any  $n$ , there must exist at least one  $k'$  such that  $\mathbf{Z}_{n,k}^{*,LP} \neq 0$  and thus  $\frac{1}{N} \hat{\mathbf{c}}_{k'} p^* + \mathbf{q}_n^* - \frac{1}{N} \hat{\mathbf{a}}_{k'}(x_n) = 0$ . By the pigeon-hole principle, we can conclude that for all  $n$  except one (denoted by  $n'$ ), exactly one equation in  $\{\frac{1}{N} \hat{\mathbf{c}}_k p^* + \mathbf{q}_n^* - \frac{1}{N} \hat{\mathbf{a}}_k(x_n) = 0\}_k$  can be satisfied.

Now let us fix any  $n \neq n'$ . Then there exists some  $k'$ , such that  $\frac{1}{N} \hat{\mathbf{c}}_{k'} p^* + \mathbf{q}_n^* - \frac{1}{N} \hat{\mathbf{a}}_{k'}(x_n) = 0$ , and for any  $k \neq k'$ ,  $\frac{1}{N} \hat{\mathbf{c}}_k p^* + \mathbf{q}_n^* - \frac{1}{N} \hat{\mathbf{a}}_k(x_n) > 0$  (due to the natural constraint in Problem B.3). That is to say, for any  $k \neq k'$ ,

$$\frac{1}{N} \hat{\mathbf{c}}_k p^* + \mathbf{q}_n^* - \frac{1}{N} \hat{\mathbf{a}}_k(x_n) > 0 = \frac{1}{N} \hat{\mathbf{c}}_{k'} p^* + \mathbf{q}_n^* - \frac{1}{N} \hat{\mathbf{a}}_{k'}(x_n)$$

Multiplying  $N$  and rearranging the terms gives

$$\hat{\mathbf{a}}_{k'}(x_n) - \hat{\mathbf{c}}_{k'} p^* > \hat{\mathbf{a}}_k(x_n) - \hat{\mathbf{c}}_k p^*$$

That is to say,  $k'$  is the unique solution to  $\max_k \hat{\mathbf{a}}_k(x_n) - \hat{\mathbf{c}}_k p^*$ . By definition of  $\tilde{\mathbf{Z}}^{p^*}$ , we have  $\tilde{\mathbf{Z}}_{n,k'}^{p^*} = 1$  and  $\tilde{\mathbf{Z}}_{n,k}^{p^*} = 0, \forall k \neq k'$ . Meanwhile, for any  $k \neq k'$ , by the slackness constraint, since  $\frac{1}{N} \hat{\mathbf{c}}_k p^* + \mathbf{q}_n^* - \frac{1}{N} \hat{\mathbf{a}}_k(x_n) > 0$ , we must have  $\mathbf{Z}_{n,k}^{*,LP} = 0$ . The natural constraint in Problem B.2 requires  $\sum_{k=1}^K \mathbf{Z}_{n,k}^{*,LP} = 1$ . Thus, we have  $\mathbf{Z}_{n,k'}^{*,LP} = \sum_{k=1}^K \mathbf{Z}_{n,k}^{*,LP} - \sum_{k \neq k'} \mathbf{Z}_{n,k}^{*,LP} = 1$ .

That is to say, for any  $n \neq n'$ , we always have  $\tilde{\mathbf{Z}}_{n,\cdot}^{p^*} = \mathbf{Z}_{n,\cdot}^{*,LP}$ , which completes the proof.  $\square$

Step 2: Now we can show  $\frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \tilde{\mathbf{Z}}_{n,k}^{p^*} \hat{\mathbf{a}}_k(x_n) \geq \frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \mathbf{Z}_{n,k}^{*,LP} \hat{\mathbf{a}}_k(x_n) - \frac{1}{N}$ . To see this, by Lemma 7,  $\tilde{\mathbf{Z}}_{n,\cdot}^{p^*} = \mathbf{Z}_{n,\cdot}^{*,LP}, \forall n \neq n'$ , we must have

$$\frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \mathbf{Z}_{n,k}^{*,LP} \hat{\mathbf{a}}_k(x_n) - \frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \tilde{\mathbf{Z}}_{n,k}^{p^*} \hat{\mathbf{a}}_k(x_n) = \frac{1}{N} \sum_{k=1}^K \mathbf{Z}_{n',k}^{*,LP} \hat{\mathbf{a}}_k(x_{n'}) - \frac{1}{N} \sum_{k=1}^K \tilde{\mathbf{Z}}_{n',k}^{p^*} \hat{\mathbf{a}}_k(x_{n'})$$

As  $\hat{a}_k(x'_n)$  is bounded in  $[0, 1]$ , we have

$$\frac{1}{N} \sum_{k=1}^K \mathbf{Z}_{n',k}^{*,LP} \hat{\mathbf{a}}_k(x'_n) - \frac{1}{N} \sum_{k=1}^K \tilde{\mathbf{Z}}_{n',k}^{p*} \hat{\mathbf{a}}_k(x'_n) \leq \frac{1}{N} \sum_{k=1}^K \mathbf{Z}_{n',k}^{*,LP} \cdot 1 - \frac{1}{N} \sum_{k=1}^K \tilde{\mathbf{Z}}_{n',k}^{p*} \cdot 0 = \frac{1}{N} \sum_{k=1}^K \mathbf{Z}_{n',k}^{*,LP}$$

By natural constraint in Problem B.2,  $\sum_{k=1}^K \mathbf{Z}_{n',k}^{*,LP} = 1$ . Thus, we have

$$\frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \mathbf{Z}_{n,k}^{*,LP} \hat{\mathbf{a}}_k(x_n) - \frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \tilde{\mathbf{Z}}_{n,k}^{p*} \hat{\mathbf{a}}_k(x_n) \leq \frac{1}{N} \sum_{k=1}^K \mathbf{Z}_{n',k}^{*,LP} = \frac{1}{N}$$

On the other hand,  $\mathbf{Z}^{*,LP}$  is the optimal solution to Problem B.2 and  $\mathbf{Z}^*$  is a feasible solution. Thus we have

$$\frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \mathbf{Z}_{n,k}^* \hat{\mathbf{a}}_k(x_n) \leq \frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \mathbf{Z}_{n,k}^{*,LP} \hat{\mathbf{a}}_k(x_n)$$

Combining the two inequalities leads to

$$\frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \tilde{\mathbf{Z}}_{n,k}^{p*} \hat{\mathbf{a}}_k(x_n) \geq \frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \mathbf{Z}_{n,k}^* \hat{\mathbf{a}}_k(x_n) - \frac{1}{N}$$

Step 3: Finally, we are ready to show the budget constraint is satisfied. By Lemma 7,  $\tilde{\mathbf{Z}}_{n,\cdot}^{p*} = \mathbf{Z}_{n,\cdot}^{*,LP}$ ,  $\forall n \neq n'$ , we have

$$\frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \tilde{\mathbf{Z}}_{n,k}^{p*} \hat{\mathbf{c}}_k - \frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \mathbf{Z}_{n,k}^{*,LP} \hat{\mathbf{c}}_k = \frac{1}{N} \sum_{k=1}^K \tilde{\mathbf{Z}}_{n',k}^{p*} \hat{\mathbf{c}}_k - \frac{1}{N} \sum_{k=1}^K \mathbf{Z}_{n',k}^{*,LP} \hat{\mathbf{c}}_k$$

Denote  $s^{p*}(x_{n'})$  by  $k_1$ . By construction, we have

$$\sum_{k=1}^K \tilde{\mathbf{Z}}_{n',k}^{p*} \hat{\mathbf{c}}_k - \sum_{k=1}^K \mathbf{Z}_{n',k}^{*,LP} \hat{\mathbf{c}}_k = \hat{\mathbf{c}}_{k_1} - \sum_{k=1}^K \mathbf{Z}_{n',k}^{*,LP} \hat{\mathbf{c}}_k$$

Let  $S$  be the set of any  $k$  such that  $\mathbf{Z}_{n',k}^{*,LP} \neq 0$ . Then we can further write

$$\sum_{k=1}^K \tilde{\mathbf{Z}}_{n',k}^{p*} \hat{\mathbf{c}}_k - \sum_{k=1}^K \mathbf{Z}_{n',k}^{*,LP} \hat{\mathbf{c}}_k = \hat{\mathbf{c}}_{k_1} - \sum_{k \in S} \mathbf{Z}_{n',k}^{*,LP} \hat{\mathbf{c}}_k$$

Note that  $k \in S$  implies  $k \in \arg \max_k \hat{\mathbf{a}}_k(x_{n'}) - \hat{\mathbf{c}}_k p^*$  (Suppose not. Then there exists some  $k'$ , such that  $\hat{\mathbf{a}}_{k'}(x_{n'}) - \hat{\mathbf{c}}_{k'} p^* > \hat{\mathbf{a}}_k(x_{n'}) - \hat{\mathbf{c}}_k p^*$ . Multiplying both sides by  $-\frac{1}{N}$  and then adding  $\mathbf{q}_n^*$  gives  $-\frac{1}{N} \hat{\mathbf{a}}_{k'}(x_{n'}) + \frac{1}{N} \hat{\mathbf{c}}_{k'} p^* + \mathbf{q}_n^* < -\frac{1}{N} \hat{\mathbf{a}}_k(x_{n'}) + \frac{1}{N} \hat{\mathbf{c}}_k p^* + \mathbf{q}_n^*$ . By complementary slackness of Problem B.2,  $\mathbf{Z}_{n,k}^{*,LP} (-\frac{1}{N} \hat{\mathbf{a}}_k(x_{n'}) + \frac{1}{N} \hat{\mathbf{c}}_k p^* + \mathbf{q}_n^*) = 0$ .  $k \in S$  implies  $\mathbf{Z}_{n,k}^{*,LP} \neq 0$  and thus  $-\frac{1}{N} \hat{\mathbf{a}}_k(x_{n'}) + \frac{1}{N} \hat{\mathbf{c}}_k p^* + \mathbf{q}_n^* = 0$ . Thus,  $-\frac{1}{N} \hat{\mathbf{a}}_{k'}(x_{n'}) + \frac{1}{N} \hat{\mathbf{c}}_{k'} p^* + \mathbf{q}_n^* < 0$ , which contradicts with the feasibility constraint in the dual problem.). Recall that  $k_1$  is determined by  $\arg \max_k \hat{\mathbf{a}}_k(x_{n'}) - \hat{\mathbf{c}}_k p^*$  and we break ties by picking  $k$  with smallest cost. Thus, for any  $k \in S$ ,  $\hat{\mathbf{c}}_k \geq \hat{\mathbf{c}}_{k_1}$ . Therefore,

$$\sum_{k=1}^K \tilde{\mathbf{Z}}_{n',k}^{p*} \hat{\mathbf{c}}_k - \sum_{k=1}^K \mathbf{Z}_{n',k}^{*,LP} \hat{\mathbf{c}}_k \leq \hat{\mathbf{c}}_{k_1} - \sum_{k \in S} \mathbf{Z}_{n',k}^{*,LP} \hat{\mathbf{c}}_{k_1} = (1 - \sum_{k \in S} \mathbf{Z}_{n',k}^{*,LP}) \hat{\mathbf{c}}_{k_1}$$

By feasibility constraint in Problem B.2,  $\sum_{k \in S} \mathbf{Z}_{n',k}^{*,LP} = \sum_{k=1}^K \mathbf{Z}_{n',k}^{*,LP} = 1$ . Thus, the above inequality becomes  $\sum_{k=1}^K \tilde{\mathbf{Z}}_{n',k}^{p*} \hat{\mathbf{c}}_k - \sum_{k=1}^K \mathbf{Z}_{n',k}^{*,LP} \hat{\mathbf{c}}_k \leq 0$ . Thus,

$$\frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \tilde{\mathbf{Z}}_{n,k}^{p*} \hat{\mathbf{c}}_k - \frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \mathbf{Z}_{n,k}^{*,LP} \hat{\mathbf{c}}_k = \frac{1}{N} \sum_{k=1}^K \tilde{\mathbf{Z}}_{n',k}^{p*} \hat{\mathbf{c}}_k - \frac{1}{N} \sum_{k=1}^K \mathbf{Z}_{n',k}^{*,LP} \hat{\mathbf{c}}_k \leq 0$$

$\mathbf{Z}^{*,LP}$  is a feasible solution to Problem B.2, so it must satisfy the budget constraint and thus  $\frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \mathbf{Z}_{n,k}^{*,LP} \hat{\mathbf{c}}_k \leq b$ . Hence, we must have

$$\frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \tilde{\mathbf{Z}}_{n,k}^{p^*} \hat{\mathbf{c}}_k \leq \frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \mathbf{Z}_{n,k}^{*,LP} \hat{\mathbf{c}}_k \leq b \leq b$$

i.e.,  $\tilde{\mathbf{Z}}^{p^*}$  satisfies the budget constraint in Problem B.1.

Finally, combining step 2 and step 3 finishes the proof.  $\square$

### B.3 PROOF OF THEOREM 2

*Proof.* Let us first establish a few lemmas consisting of the main components of the proof.

**Lemma 8.** Suppose  $\delta \geq \frac{\|\mathbf{c}\|_\infty}{b} \left[ \sqrt{\frac{\log 4 - \log \epsilon}{N}} + \sqrt{\frac{\log 4 - \log \epsilon}{N^{Tr}}} \right]$ . Then with probability at least  $1 - \epsilon$ ,  $s^{\hat{p}}$  is a feasible solution to Problem 3.1.

*Proof.* We first note that Problem 3.2 is a linear programming, and its dual problem is

$$\begin{aligned} \max_{\mathbf{Z} \in \mathbb{R}^{N \times K}}: & \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} \mathbf{Z}_{n,k} \hat{\mathbf{a}}_k(x_n^{Tr}) \\ \text{s.t.} & \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} \sum_{k=1}^K \mathbf{Z}_{n,k} \hat{\mathbf{c}}_k \leq (1 - \delta) \hat{b} \\ & \sum_{k=1}^K \mathbf{Z}_{n,k} = 1, \mathbf{Z}_{n,k} \in [0, 1], \forall n, k \end{aligned} \quad (\text{B.4})$$

Note that this is in the same form of Problem B.2 except that the data become  $\{x_n^{Tr}\}_{n=1}^{N^{Tr}}$  instead of  $\{x_n\}_{n=1}^N$ . Using a similar argument in the proof for Theorem 1,  $s^{\hat{p}}(x_n^{Tr})$  is a feasible solution to

$$\begin{aligned} \max & \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} r^{s^{\hat{p}}}(x_n^{Tr}) \\ \text{s.t.} & \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} \eta^{[s^{\hat{p}}]}(x_n^{Tr}, \mathbf{c}) \leq (1 - \delta) b, \end{aligned}$$

and thus we have

$$\frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} \eta^{[s^{\hat{p}}]}(x_n^{Tr}, \mathbf{c}) \leq (1 - \delta) b$$

Note that training data  $x_n^{Tr}$  are i.i.d samples from the true distribution and  $0 \leq \eta^{[s]}(x_n^{Tr}, \mathbf{c}) \leq \|\mathbf{c}\|_\infty$ . Thus, by Hoeffding's inequality, with probability  $1 - \epsilon$ , we have

$$\left\| \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} \eta^{[s^{\hat{p}}]}(x_n^{Tr}, \mathbf{c}) - \mathbb{E} [\eta^{[s^{\hat{p}}]}(x, \mathbf{c})] \right\| \leq \|\mathbf{c}\|_\infty \sqrt{\frac{\log 2 - \log \epsilon}{2N^{Tr}}}$$

The data stream  $x_n$  is also from the same distribution, and thus we also have with probability  $1 - \epsilon$ ,

$$\left\| \frac{1}{N} \sum_{n=1}^N \eta^{[s^{\hat{p}}]}(x_n, \mathbf{c}) - \mathbb{E} [\eta^{[s^{\hat{p}}]}(x, \mathbf{c})] \right\| \leq \|\mathbf{c}\|_\infty \sqrt{\frac{\log 2 - \log \epsilon}{2N}}$$

Applying union bound, we have with probability  $1 - \epsilon$ ,

$$\left\| \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} \eta^{[s^p]}(x_n^{Tr}, \mathbf{c}) - \mathbb{E} [\eta^{[s^p]}(x, \mathbf{c})] \right\| \leq \|\mathbf{c}\|_\infty \sqrt{\frac{\log 4 - \log \epsilon}{2N^{Tr}}}$$

and

$$\left\| \frac{1}{N} \sum_{n=1}^N \eta^{[s^p]}(x_n, \mathbf{c}) - \mathbb{E} [\eta^{[s^p]}(x, \mathbf{c})] \right\| \leq \|\mathbf{c}\|_\infty \sqrt{\frac{\log 4 - \log \epsilon}{2N}}$$

Using triangle inequality, we have with probability  $1 - \epsilon$ ,

$$\left\| \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} \eta^{[s^p]}(x_n^{Tr}, \mathbf{c}) - \frac{1}{N} \sum_{n=1}^N \eta^{[s^p]}(x_n, \mathbf{c}) \right\| \leq \|\mathbf{c}\|_\infty \sqrt{\frac{\log 4 - \log \epsilon}{2N}} + \|\mathbf{c}\|_\infty \sqrt{\frac{\log 4 - \log \epsilon}{2N^{Tr}}}.$$

Thus we have

$$\begin{aligned} \frac{1}{N} \sum_{n=1}^N \eta^{[s^p]}(x_n, \mathbf{c}) &\leq \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} \eta^{[s^p]}(x_n^{Tr}, \mathbf{c}) + \|\mathbf{c}\|_\infty \sqrt{\frac{\log 4 - \log \epsilon}{2N}} + \|\mathbf{c}\|_\infty \sqrt{\frac{\log 4 - \log \epsilon}{2N^{Tr}}} \\ &\leq (1 - \delta)b + \|\mathbf{c}\|_\infty \sqrt{\frac{\log 4 - \log \epsilon}{2N}} + \|\mathbf{c}\|_\infty \sqrt{\frac{\log 4 - \log \epsilon}{2N^{Tr}}} \leq b \end{aligned}$$

where the last inequality is due to the assumption on  $\delta$ . That is to say, with probability  $1 - \epsilon$ ,  $s^{\hat{p}}$  is a feasible solution to Problem 3.1, which completes the proof.  $\square$

**Lemma 9.** Construct the set  $\Omega_M \triangleq \{0, \frac{1}{(M-1) \min_{\mathbf{c}_k \neq 0} \mathbf{c}_k}, \frac{2}{(M-1) \min_{\mathbf{c}_k \neq 0} \mathbf{c}_k}, \dots, \frac{1}{\min_{\mathbf{c}_k \neq 0} \mathbf{c}_k}\}$  and

$$\begin{aligned} \hat{p}(\Omega_M) &\triangleq \arg \max_{p \in \Omega_M} \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} r^{s^p}(x_n^{Tr}) \\ &\text{s.t. } \frac{1}{N^{Tr}} \eta^{[s^p]}(x_n, \mathbf{c}) \leq (1 - \delta)b. \end{aligned}$$

Then with probability  $1 - \epsilon$ ,

$$\left\| \frac{1}{N} \sum_{n=1}^N r^{s^{\hat{p}}}(x_n) - \frac{1}{N} \sum_{n=1}^N r^{s^{\hat{p}(\Omega_M)}}(x_n) \right\| \leq O\left(\sqrt{\frac{\log N + \log 8 - \log \epsilon}{2N}} + \sqrt{\frac{\log N^{Tr} + \log 8 - \log \epsilon}{2N^{Tr}}}\right).$$

*Proof.* Note that  $\Omega_M \subseteq \mathbb{R}$ . Consider an element  $p \in \mathbb{R}$ .

(i)  $p \geq \frac{1}{\min_{\mathbf{c}_k \neq 0} \mathbf{c}_k}$ : This effectively means the API with the smallest cost is always selected. In other words, we always have

$$bs = \arg \max \hat{\mathbf{a}}_k(x) - p\hat{\mathbf{c}}_k$$

To see this, simply note that for any other  $k_1$ , we have

$$\begin{aligned} \hat{\mathbf{a}}_{bs}(x) - p\hat{\mathbf{c}}_{bs} - (\hat{\mathbf{a}}_{k_1}(x) - p\hat{\mathbf{c}}_{k_1}) &= \hat{\mathbf{a}}_{bs}(x) - \hat{\mathbf{a}}_{k_1}(x) + p(\hat{\mathbf{c}}_{k_1} - \hat{\mathbf{c}}_{bs}) = \hat{\mathbf{a}}_{bs}(x) - \hat{\mathbf{a}}_{k_1}(x) + p\hat{\mathbf{c}}_{k_1} \\ &\geq 0 - 1 + p\hat{\mathbf{c}}_{k_1} \geq -1 + \hat{\mathbf{c}}_{k_1} \cdot \frac{1}{\min_{\mathbf{c}_k \neq 0} \mathbf{c}_k} \geq 0 \end{aligned}$$

Thus, for such  $p$ , the objective value is the same as that for  $\frac{1}{\min_{\mathbf{c}_k \neq 0} \mathbf{c}_k} \in \Omega_M$ .

(ii)  $0 \leq p \leq \frac{1}{\min_{\mathbf{c}_k \neq 0} \mathbf{c}_k}$ : By construction of  $\Omega_M$ , there exists some  $m$ , such that  $\frac{m}{(M-1) \min_{\mathbf{c}_k \neq 0} \mathbf{c}_k} \leq p \leq \frac{m+1}{(M-1) \min_{\mathbf{c}_k \neq 0} \mathbf{c}_k}$ . Let  $p_j \triangleq \frac{j}{(M-1) \min_{\mathbf{c}_k \neq 0} \mathbf{c}_k}$  for ease of notations. Clearly, we have  $p_m \in \Omega_M$ .

Now let us partition the space of  $\hat{\mathbf{a}}(x)$  into  $M$  regions, denoted by  $A_1, A_2, \dots, A_M$ . Abusing the notation a little bit, let  $\phi(p, x) \triangleq \arg \max \hat{\mathbf{a}}_k(x) - p\hat{\mathbf{c}}_k$ .  $A_1$  is the set of all  $\hat{\mathbf{a}}(x)$  such that  $\phi(p, x)$  is a constant.  $A_2$  is the set of all  $\hat{\mathbf{a}}(x)$  such that  $\phi(p, x)$  is a constant for  $p$  larger than  $p_1$ . Generally,  $A_j$  is the set of all  $\hat{\mathbf{a}}(x)$  such that  $\phi(p, x)$  is a constant for  $p$  larger than  $p_{j-1}$  subtracting  $A_{j-1}$ . Formally,

$$A_j = \begin{cases} \{\hat{\mathbf{a}}(x) : \phi(p, x) \text{ is a constant}\}, & j = 1 \\ \{\hat{\mathbf{a}}(x) : \phi(p, x) \text{ is a constant if } p \geq p_{j-1}\} - A_j, & j > 1 \end{cases}$$



One can easily verify that  $\{A_j\}$  form a partition of the space of the estimated accuracy, and further more,  $\|A_j\| \leq \frac{\|\mathbf{c}\|_1}{M \min_{\mathbf{c}_k \neq 0} \mathbf{c}_k}$ . By the assumption of the distribution, there exists some constant  $u$ , such that  $\Pr(A) \leq u\|A\|$ , for any  $A$  in the probability space. Thus, we must have

$$\Pr[\hat{\mathbf{a}}(x) \in A_j] \leq \|A_j\|u = \frac{u\|\mathbf{c}\|_1}{M \min_{\mathbf{c}_k \neq 0} \mathbf{c}_k^2}$$

Now note that, when  $p_m = \frac{m}{(M-1) \min_{\mathbf{c}_k \neq 0} \mathbf{c}_k} \leq p \leq \frac{m+1}{(M-1) \min_{\mathbf{c}_k \neq 0} \mathbf{c}_k} = p_{m+1}$ , only elements in  $A_m$  may affect the reward. More precisely, we have

$$\frac{1}{N} \sum_{n=1}^N r^{s^p}(x_n) - \frac{1}{N} \sum_{n=1}^N r^{s^{p_m}}(x_n) = \frac{1}{N} \sum_{x_n \in A_m} r^{s^p}(x_n) - \frac{1}{N} \sum_{x_n \in A_m} r^{s^{p_m}}(x_n)$$

Note that each estimated accuracy is an i.i.d sample from the true distribution, and its value is from  $[0, 1]$ , by Hoeffding's inequality, with probability  $1 - \epsilon$ , we have

$$\left\| \frac{1}{N} \sum_{n=1}^N \mathbb{1}_{x_n \in A_j} - \Pr[x_n \in A_j] \right\| \leq \sqrt{\frac{\log 2 - \log \epsilon}{2N}}$$

Applying the union bound, we have for any  $j$ , with probability  $1 - \epsilon$ ,

$$\left\| \frac{1}{N} \sum_{n=1}^N \mathbb{1}_{x_n \in A_j} - \Pr[x_n \in A_j] \right\| \leq \sqrt{\frac{\log M + \log 2 - \log \epsilon}{2N}}$$

Therefore, we have with probability  $1 - \epsilon$ ,

$$\begin{aligned} \frac{1}{N} \sum_{n=1}^N r^{s^p}(x_n) - \frac{1}{N} \sum_{n=1}^N r^{s^{p_m}}(x_n) &= \frac{1}{N} \sum_{x_n \in A_m} r^{s^p}(x_n) - \frac{1}{N} \sum_{x_n \in A_m} r^{s^{p_m}}(x_n) \\ &\geq \sum_{x_n \in A_m} 0 - \frac{1}{N} \sum_{x_n \in A_m} 1 = \frac{1}{N} \sum_{n=1}^N \mathbb{1}_{x_n \in A_m} \\ &\geq \Pr[x_n \in A_m] - \sqrt{\frac{\log M + \log 2 - \log \epsilon}{2N}} \\ &\geq -\sqrt{\frac{\log M + \log 2 - \log \epsilon}{2N}} \end{aligned}$$

and similarly

$$\begin{aligned} \frac{1}{N} \sum_{n=1}^N r^{s^p}(x_n) - \frac{1}{N} \sum_{n=1}^N r^{s^{p_m}}(x_n) &= \frac{1}{N} \sum_{x_n \in A_m} r^{s^p}(x_n) - \frac{1}{N} \sum_{x_n \in A_m} r^{s^{p_m}}(x_n) \\ &\leq \sum_{x_n \in A_m} 1 - \frac{1}{N} \sum_{x_n \in A_m} 0 = \frac{1}{N} \sum_{n=1}^N \mathbb{1}_{x_n \in A_m} \\ &\leq \Pr[x_n \in A_m] + \sqrt{\frac{\log M + \log 2 - \log \epsilon}{2N}} \\ &\leq \frac{u\|\mathbf{c}\|_1}{M \min_{\mathbf{c}_k \neq 0} \mathbf{c}_k} + \sqrt{\frac{\log M + \log 2 - \log \epsilon}{2N}} \end{aligned}$$

That is to say,

$$\left\| \frac{1}{N} \sum_{n=1}^N r^{s^p}(x_n) - \frac{1}{N} \sum_{n=1}^N r^{s^{p_m}}(x_n) \right\| \leq \frac{u\|\mathbf{c}\|_1}{M \min_{\mathbf{c}_k \neq 0} \mathbf{c}_k} + \sqrt{\frac{\log M + \log 2 - \log \epsilon}{2N}} \quad (\text{B.5})$$

Similarly, for the training dataset, we can also get, with probability  $1 - \epsilon$ ,

$$\left\| \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} r^{s^p}(x_n^{Tr}) - \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} r^{s^{p_m}}(x_n^{Tr}) \right\| \leq \frac{u\|\mathbf{c}\|_1}{M \min_{\mathbf{c}_k \neq 0} \mathbf{c}_k} + \sqrt{\frac{\log M + \log 2 - \log \epsilon}{2N}}$$

Combining case (i) and case (ii), we have just shown that for any  $p \in \mathbb{R}$ , there exists another  $p' \in \Omega_M$ , such that

$$\left\| \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} r^{s^p}(x_n^{Tr}) - \frac{1}{N^{Tr}} \sum_{n=1}^N r^{s^{p'}}(x_n^{Tr}) \right\| \leq \frac{u\|\mathbf{c}\|_1}{M \min_{\mathbf{c}_k \neq 0} \mathbf{c}_k} + \sqrt{\frac{\log M + \log 2 - \log \epsilon}{2N}} \quad (\text{B.6})$$

Thus, applying Lemma 4, we have with probability  $1 - \epsilon$ ,

$$\left\| \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} r^{s^{\hat{p}}}(x_n^{Tr}) - \frac{1}{N^{Tr}} \sum_{n=1}^N r^{s^{\hat{p}(\Omega_M)}}(x_n^{Tr}) \right\| \leq \frac{u\|\mathbf{c}\|_1}{M \min_{\mathbf{c}_k \neq 0} \mathbf{c}_k} + \sqrt{\frac{\log M + \log 2 - \log \epsilon}{2N}}$$

Now by Lemma 5, for each fixed  $j$ , we have with probability  $1 - \epsilon$ ,

$$\left\| \frac{1}{N} \sum_{n=1}^N r^{s^{p_j}}(x_n) - \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} r^{s^{p_j}}(x_n^{Tr}) \right\| \leq \left[ \sqrt{\frac{\log 4 - \log \epsilon}{2N}} + \sqrt{\frac{\log 4 - \log \epsilon}{2N^{Tr}}} \right]$$

Applying union bound, with probability  $1 - \epsilon$ ,

$$\left\| \frac{1}{N} \sum_{n=1}^N r^{s^{p_j}}(x_n) - \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} r^{s^{p_j}}(x_n^{Tr}) \right\| \leq \left[ \sqrt{\frac{\log M + \log 4 - \log \epsilon}{2N}} + \sqrt{\frac{\log M + \log 4 - \log \epsilon}{2N^{Tr}}} \right]$$

for all  $j$ . Specifically, we have

$$\left\| \frac{1}{N} \sum_{n=1}^N r^{s^{\hat{p}(\Omega_M)}}(x_n) - \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} r^{s^{\hat{p}(\Omega_M)}}(x_n^{Tr}) \right\| \leq \left[ \sqrt{\frac{\log M + \log 4 - \log \epsilon}{2N}} + \sqrt{\frac{\log M + \log 4 - \log \epsilon}{2N^{Tr}}} \right] \quad (\text{B.7})$$

and

$$\left\| \frac{1}{N} \sum_{n=1}^N r^{s^{p'}}(x_n) - \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} r^{s^{p'}}(x_n^{Tr}) \right\| \leq \left[ \sqrt{\frac{\log M + \log 4 - \log \epsilon}{2N}} + \sqrt{\frac{\log M + \log 4 - \log \epsilon}{2N^{Tr}}} \right] \quad (\text{B.8})$$

Now combining equations B.5, B.6, B.7, and B.8 with triangle inequality, we have with probability  $1 - \epsilon$ ,

$$\begin{aligned} & \left\| \frac{1}{N} \sum_{n=1}^N r^{s^{\hat{p}}}(x_n) - \frac{1}{N} \sum_{n=1}^N r^{s^{\hat{p}(\Omega_M)}}(x_n) \right\| \\ & \leq \frac{u\|\mathbf{c}\|_1}{4M \min_{\mathbf{c}_k \neq 0} \mathbf{c}_k} + 4\sqrt{\frac{\log M + \log 8 - \log \epsilon}{2N}} + 2\sqrt{\frac{\log M + \log 8 - \log \epsilon}{2N^{Tr}}} \end{aligned}$$

Setting  $M = \min\{N^{Tr}, N\}$ , we have

$$\left\| \frac{1}{N} \sum_{n=1}^N r^{s^{\hat{p}}}(x_n) - \frac{1}{N} \sum_{n=1}^N r^{s^{\hat{p}(\Omega_M)}}(x_n) \right\| \leq O\left(\sqrt{\frac{\log N + \log 8 - \log \epsilon}{2N}} + \sqrt{\frac{\log N^{Tr} + \log 8 - \log \epsilon}{2N^{Tr}}}\right)$$

which completes the proof.  $\square$

**Lemma 10.** *Let*

$$\begin{aligned} p(\Omega_M) & \triangleq \arg \max_{p \in \Omega_M} \frac{1}{N} \sum_{n=1}^N r^{s^p}(x_n) \\ \text{s.t. } & \frac{1}{N} \eta^{[s^p]}(x_n, \mathbf{c}) \leq (1 - \delta)b - \|\mathbf{c}\|_\infty \left[ \sqrt{\frac{\log 8 - \log \epsilon}{2N}} + \sqrt{\frac{\log 8 - \log \epsilon}{2N^{Tr}}} \right] \end{aligned}$$

Then with probability  $1 - \epsilon$ ,

$$\frac{1}{N} \sum_{n=1}^N r^{s^{\hat{p}(\Omega_M)}}(x_n) \geq \frac{1}{N} \sum_{n=1}^N r^{s^{p(\Omega_M)}}(x_n) - \sqrt{\frac{\log 8 - \log \epsilon}{2N}} - \sqrt{\frac{\log 8 - \log \epsilon}{2N^{Tr}}}$$

and

$$\frac{1}{N} \sum_{n=1}^N \eta^{[s^{\hat{p}(\Omega_M)}]}(x_n, \mathbf{c}) \leq (1 - \delta)b + 2\|\mathbf{c}\|_\infty \left[ \sqrt{\frac{\log 8 - \log \epsilon}{2N}} + \sqrt{\frac{\log 8 - \log \epsilon}{2N^{Tr}}} \right].$$

*Proof.* By Lemma 5, with probability  $1 - \epsilon$ , we have

$$\left\| \frac{1}{N} \sum_{n=1}^N r^{s^p}(x_n) - \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} r^{s^p}(x_n^{Tr}) \right\| \leq \sqrt{\frac{\log 4 - \log \epsilon}{2N}} + \sqrt{\frac{\log 4 - \log \epsilon}{2N^{Tr}}}.$$

and similarly, with probability  $1 - \epsilon$ ,

$$\left\| \frac{1}{N} \sum_{n=1}^N \eta^{[s^p]}(x_n, \mathbf{c}) - \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} \eta^{[s^p]}(x_n^{Tr}, \mathbf{c}) \right\| \leq \|\mathbf{c}\|_\infty \left[ \sqrt{\frac{\log 4 - \log \epsilon}{2N}} + \sqrt{\frac{\log 4 - \log \epsilon}{2N^{Tr}}} \right].$$

which is the same as

$$\left\| \frac{1}{N} \sum_{n=1}^N \eta^{[s^p]}(x_n, \mathbf{c}) - (1 - \delta)b - \left[ \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} \eta^{[s^p]}(x_n^{Tr}, \mathbf{c}) - (1 - \delta)b \right] \right\| \leq \|\mathbf{c}\|_\infty \left[ \sqrt{\frac{\log 4 - \log \epsilon}{2N}} + \sqrt{\frac{\log 4 - \log \epsilon}{2N^{Tr}}} \right].$$

Now applying union bound, we have with probability  $1 - \epsilon$ ,

$$\left\| \frac{1}{N} \sum_{n=1}^N r^{s^p}(x_n) - \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} r^{s^p}(x_n^{Tr}) \right\| \leq \sqrt{\frac{\log 8 - \log \epsilon}{2N}} + \sqrt{\frac{\log 8 - \log \epsilon}{2N^{Tr}}}.$$

and

$$\left\| \frac{1}{N} \sum_{n=1}^N \eta^{[s^p]}(x_n, \mathbf{c}) - (1 - \delta)b - \left[ \frac{1}{N^{Tr}} \sum_{n=1}^{N^{Tr}} \eta^{[s^p]}(x_n^{Tr}, \mathbf{c}) - (1 - \delta)b \right] \right\| \leq \|\mathbf{c}\|_\infty \left[ \sqrt{\frac{\log 8 - \log \epsilon}{2N}} + \sqrt{\frac{\log 8 - \log \epsilon}{2N^{Tr}}} \right].$$

both hold. By Lemma 6, we can conclude that

$$\frac{1}{N} \sum_{n=1}^N r^{s^{\hat{p}(\Omega_M)}}(x_n) \geq \frac{1}{N} \sum_{n=1}^N r^{s^p(\Omega_M)} - \sqrt{\frac{\log 8 - \log \epsilon}{2N}} - \sqrt{\frac{\log 8 - \log \epsilon}{2N^{Tr}}}.$$

and

$$\frac{1}{N} \sum_{n=1}^N \eta^{[s^{\hat{p}(\Omega_M)}]}(x_n, \mathbf{c}) - (1 - \delta)b \leq 2\|\mathbf{c}\|_\infty \left[ \sqrt{\frac{\log 8 - \log \epsilon}{2N}} + \sqrt{\frac{\log 8 - \log \epsilon}{2N^{Tr}}} \right].$$

with probability  $1 - \epsilon$ , which completes the proof.  $\square$

**Lemma 11.** For  $\delta = \Omega\left(\sqrt{\frac{\log 4 - \log \epsilon}{N}} + \sqrt{\frac{\log 4 - \log \epsilon}{N^{Tr}}}\right)$ , we have with probability  $1 - \epsilon$ ,

$$\frac{1}{N} \sum_{n=1}^N r^{s^p(\Omega_M)}(x_n) - \frac{1}{N} \sum_{n=1}^N r^{s^{p^*}}(x_n) \geq -O\left(\sqrt{\frac{\log N - \log \epsilon}{2N}} + \sqrt{\frac{\log N - \log \epsilon}{2N^{Tr}}}\right).$$

*Proof.* Let  $\tilde{p}(\Delta)$  be the optimal solution to the following problem

$$\max_{p \in \mathbb{R}} \frac{1}{N} \sum_{n=1}^N r^{s^p}(x_n) \text{ s.t. } \frac{1}{N} \eta^{[s^p]}(x_n, \mathbf{c}) \leq b - \Delta$$

On one hand,  $p^*$  apparently is a feasible solution to the above problem with  $\Delta = 0$ , so we must have

$$\frac{1}{N} \sum_{n=1}^N r^{s^{p^*}}(x_n) \leq \frac{1}{N} \sum_{n=1}^N r^{s^{\tilde{p}(0)}}(x_n) \quad (\text{B.9})$$

Let  $\Delta' = \delta + \|\mathbf{c}\|_\infty \left[ \sqrt{\frac{\log 8 - \log \epsilon}{2N}} + \sqrt{\frac{\log 8 - \log \epsilon}{2N^{Tr}}} \right]$ . Then  $\tilde{p}(\Delta')$  corresponds to the following problem

$$\begin{aligned} \max_{p \in \mathbb{R}} \frac{1}{N} \sum_{n=1}^N r^{s^p}(x_n) \\ \text{s.t. } \frac{1}{N} \eta^{[s^p]}(x_n, \mathbf{c}) \leq (1 - \delta)b - \|\mathbf{c}\|_\infty \left[ \sqrt{\frac{\log 8 - \log \epsilon}{2N}} + \sqrt{\frac{\log 8 - \log \epsilon}{2N^{Tr}}} \right] \end{aligned}$$

Then using the same argument in the proof of Lemma 9, we have with probability  $1 - \epsilon$ ,

$$\left\| \frac{1}{N} \sum_{n=1}^N r^{s^{\tilde{p}(\Delta')}}(x_n) - \frac{1}{N} \sum_{n=1}^N r^{s^{p(\Omega_M)}}(x_n) \right\| \leq \sqrt{\frac{\log N + \log 8 + \log \epsilon}{2N}} + \sqrt{\frac{\log N + \log 8 - \log \epsilon}{2N^{Tr}}}. \quad (\text{B.10})$$

Furthermore, it is clear that  $\tilde{p}(\Delta)$  is decreasingly-monotone with respect to  $\Delta$ . In fact, removing the budget by  $\Delta_1$ , at most  $\frac{\Delta_1}{\min_{\mathbf{c}_k > \mathbf{c}_j} \mathbf{c}_k - \mathbf{c}_j}$  data's APIs need to be changed, and thus incurs at most  $\frac{\Delta_1}{\min_{\mathbf{c}_k > \mathbf{c}_j} \mathbf{c}_k - \mathbf{c}_j}$  accuracy decrease. That is to say, we must have

$$\left\| \frac{1}{N} \sum_{n=1}^N r^{s^{\tilde{p}(\Delta')}}(x_n) - \frac{1}{N} \sum_{n=1}^N r^{s^{\tilde{p}(0)}}(x_n) \right\| \leq \frac{\Delta_1}{\min_{\mathbf{c}_k > \mathbf{c}_j} \mathbf{c}_k - \mathbf{c}_j} \quad (\text{B.11})$$

Now combining equations B.9, B.10, B.11, we can obtain

$$\begin{aligned} & \frac{1}{N} \sum_{n=1}^N r^{s^{p(\Omega_M)}}(x_n) - \frac{1}{N} \sum_{n=1}^N r^{s^{p^*}}(x_n) \\ &= \frac{1}{N} \sum_{n=1}^N r^{s^{p(\Omega_M)}}(x_n) - \frac{1}{N} \sum_{n=1}^N r^{s^{\tilde{p}(\Delta')}}(x_n) \\ &+ \frac{1}{N} \sum_{n=1}^N r^{s^{\tilde{p}(\Delta')}}(x_n) - \frac{1}{N} \sum_{n=1}^N r^{s^{\tilde{p}(0)}}(x_n) + \frac{1}{N} \sum_{n=1}^N r^{s^{\tilde{p}(0)}}(x_n) - \frac{1}{N} \sum_{n=1}^N r^{s^{p^*}}(x_n) \\ &\geq -\sqrt{\frac{\log N + \log 8 + \log \epsilon}{2N}} - \sqrt{\frac{\log N^{Tr} + \log 8 - \log \epsilon}{2N^{Tr}}} - \frac{\Delta_1}{\min_{\mathbf{c}_k > \mathbf{c}_j} \mathbf{c}_k - \mathbf{c}_j} - 0 \end{aligned}$$

When  $\delta = \Omega\left(\sqrt{\frac{\log 4 - \log \epsilon}{N}} + \sqrt{\frac{\log 4 - \log \epsilon}{N^{Tr}}}\right)$ , we have

$$\begin{aligned} & \frac{1}{N} \sum_{n=1}^N r^{s^{p(\Omega_M)}}(x_n) - \frac{1}{N} \sum_{n=1}^N r^{s^{p^*}}(x_n) \\ &\geq -O\left(\sqrt{\frac{\log N - \log \epsilon}{2N}} + \sqrt{\frac{\log N^{Tr} - \log \epsilon}{2N^{Tr}}}\right) \end{aligned}$$

which completes the proof.  $\square$

Now we are ready to prove the main theorem. We start by showing the bound on the reward. Suppose  $\delta = \Theta\left(\sqrt{\frac{\log N - \log \epsilon}{N}} + \sqrt{\frac{\log N^{Tr} - \log \epsilon}{N^{Tr}}}\right)$ . By union bound, with probability  $1 - \epsilon$ , Lemma 9, Lemma 10, and Lemma 11 all hold, and we have

$$\begin{aligned}
& \frac{1}{N} \sum_{n=1}^N r^{s^{\hat{p}}}(x_n) - \frac{1}{N} \sum_{n=1}^N r^{s^{\hat{p}^*}}(x_n) \\
&= \frac{1}{N} \sum_{n=1}^N r^{s^{\hat{p}}}(x_n) - \frac{1}{N} \sum_{n=1}^N r^{s^{\hat{p}(\Omega_M)}}(x_n) + \frac{1}{N} \sum_{n=1}^N r^{s^{\hat{p}(\Omega_M)}}(x_n) \\
&+ \frac{1}{N} \sum_{n=1}^N r^{s^{\hat{p}(\Omega_M)}}(x_n) - \frac{1}{N} \sum_{n=1}^N r^{s^{\hat{p}^*}}(x_n) \\
&\geq -O\left(\sqrt{\frac{\log N - \log \epsilon}{N}} + \sqrt{\frac{\log N^{Tr} - \log \epsilon}{N^{Tr}}}\right)
\end{aligned}$$

Now note that by Theorem 1, we have with probability 1,

$$\frac{1}{N} \sum_{n=1}^N r^{s^{\hat{p}^*}}(x_n) \geq \frac{1}{N} \sum_{n=1}^N r^{s^*}(x_n) - \frac{1}{N}$$

Combing the above two inequalities, we have

$$\frac{1}{N} \sum_{n=1}^N r^{s^{\hat{p}}}(x_n) - \frac{1}{N} \sum_{n=1}^N r^{s^*}(x_n) \geq -O\left(\sqrt{\frac{\log N - \log \epsilon}{N}} + \sqrt{\frac{\log N^{Tr} - \log \epsilon}{N^{Tr}}}\right)$$

Next we consider the feasibility requirement. By Lemma 8, with probability  $1 - \epsilon$ ,  $s^{\hat{p}}$  is a feasible solution to Problem 3.1. That is to say,  $s^{\hat{p}}$  with probability  $1 - \epsilon$  is a feasible solution. Applying union bound completes the proof.  $\square$

## C EXPERIMENTAL DETAILS

We provide missing experimental details in this part.

**Experimental setup** All experiments were run on a machine with 8 Intel Xeon Platinum 2.5 GHz cores, 32 GB RAM, and 500GB disk with Ubuntu 16.04 LTS as the OS. Our code is implemented in Python 3.7. Each experiments, except the case study, were run for five times to mitigate the randomness introduced by training-testing splitting.

**ML tasks and services** Recall that We focus on three multi-label classification tasks, multi-label image classification (*MIC*), scene text recognition (*STR*), and named entity recognition (*NER*).

*MIC* is a computer vision task, where the goal is to assign a set of labels to a given image. For *MIC*, we use 3 different commercial ML cloud services, Google Vision (Goo), Microsoft Vision (Mic), and Everyapixel(Eve). We also use a single shot detector model (SSD) pretrained on OpenImageV4 (Kuznetsova et al., 2020), which is freely available from GitHub (SSD). All of those APIs produce labels from a large (and unknown) set, but the datasets we consider have bounded number of labels. For example, there are only 80 distinct labels in COCO dataset. Thus, we remove the predicted labels which are not in the full label set. For example, if Google API gives label  $\{person, car, man\}$  for an image in COCO, but *man* is not in the full label set of COCO, then we will use  $\{person, car\}$  as the label set produced from Google.

*STR* is a computer vision task, where the goal is to predict all texts in a natural scene image. In the context of multi-label classification, we view each predicted word as a label, and all possible words as the label set. For *STR*, the ML services used in the experiments are Google Vision (Goo), iFLYTEK API (Ifi), and Tencent API (Ten). We also use PP-OCR (Pad), an open source model from GitHub.

*NER* is a natural language processing task where the goal is to extract all possible entities from a given text. For example, for the sentence *ICML was held in Long Beach in 2019*, *ICML* should be extracted as an organization, and *Long Beach* should be identified as a location. In this paper, we consider three common types of entities, *person*, *location*, and *organization*. For any given text, each possible entity is viewed as a label, and the label set is the number of unique entities in the entire dataset. For *NER*, we use three common APIs: Amazon Comprehend (Ama), Google NLP (GoN), and IBM natural language API (IBM). a multi-task convolutional neural network model(Spa) from GitHub is also used.

Table 3: Dataset Statistics.

Task	Dataset	Size	# Labels	Dist Labels
MIC	PASCAL (Everingham et al., 2015)	11540	16682	20
	MIR (Huiskes & Lew, 2008)	25000	92909	24
	COCO (Lin et al., 2014)	123287	357662	80
STR	MTWI (He et al., 2018)	9742	867727	4404
	ReCTS (Zhang et al., 2019)	20000	555286	4134
	LSVT (Sun et al., 2019)	30000	1878682	4852
NER	CONLL (Sang & Meulder, 2003)	10898	43968	9910
	ZHNER (ZHN)	16915	147164	4375
	GMB (Bos, 2013)	47830	116225	14376

**Datasets.** The experiments were conducted on 9 datasets. For *MIC*, we use three popular datasets including PASCAL (Everingham et al., 2015), MIR (Huiskes & Lew, 2008) and COCO (Lin et al., 2014). PASCAL is a standard object recognition dataset with 20 distinct labels, and COCO is another one with 80 unique labels. PASCAL’s label set contains 20 common objects: *person, bird, cat, cow, dog, horse, sheep, aeroplane, bicycle, boat, bus, car, motorbike, train, bottle, chair, dining*

*table, potted plant, sofa, tv/monitor*. The 80 objects in COCO include: *person, bicycle, car, motorcycle, airplane, bus, train, truck, boat, traffic light, fire hydrant, stop sign, parking meter, bench, bird, cat, dog, horse, sheep, cow, elephant, bear, zebra, giraffe, backpack, umbrella, handbag, tie, suitcase, frisbee, skis, snowboard, sports ball, kite, baseball bat, baseball glove, skateboard, surfboard, tennis racket, bottle, wine glass, cup, fork, knife, spoon, bowl, banana, apple, sandwich, orange, broccoli, carrot, hot dog, pizza, donut, cake, chair, couch, potted plant, bed, dining table, toilet, tv, laptop, mouse, remote, keyboard, cell phone, microwave, oven, toaster, sink, refrigerator, book, clock, vase, scissors, teddy bear, hair drier, toothbrush*. For those two datasets, we use their original associated labels as the label set. MIR is a dataset designed for image retrieval. There are originally 25 labels: *animals, baby, bird, car, clouds, dog, female, flower, food, indoor, lake, male, night, people, plant\_life, portrait, river, sea, sky, structures, sunset, transport, tree, water*. We remove the label *night* since it is not in the label set of any of the APIs or the GitHub model. On average, there are 1.44 labels per image for PASCAL, 3.71 labels per image for MIR, and 2.91 labels per image for COCO. The dataset statistic is summarized in Table 3. Most of the datasets are open and under Creative Commons license (e.g., the dataset COCO (Lin et al., 2014)). The details can be found in their corresponding paper and repository. As those datasets are actually open, they do not require an in-person consent from the authors/developers. The datasets themselves may contain personal information (e.g., there are personal images in COCO). Though, they have been rendered anonymous. For the purpose of deciding which API to call, we also do not use personally identifiable information.

For *STR*, we use three large scale Chinese text recognition datasets, MTWI (He et al., 2018), ReCTS (Zhang et al., 2019) and LSVT (Sun et al., 2019). The label set contains all possible Chinese characters as well as digits (0-9). MTWI contains images from the internet mainly targeting at advertisements. Thus, most of its images have dense texts. ReCTS includes photos taken on sign boards and thus has relatively fewer words. The images from LSVT are typically street view images and hence have medium number of words. All images in MTWI and ReCTS are fully annotated and used in our experiments. LSVT contains both fully and partially annotated images, and we only use the subset with full annotations.

The other datasets, CONLL (Sang & Meulder, 2003), ZHNER (ZHN) and GMB (Bos, 2013), are used for *NER* task. CONLL contains English sentences from newspapers, and texts from GMB are also English and from a wider range of sources. On the other hand, ZHNER is a Chinese text dataset. We consider four common types of entities: organization, person, and location. In this paper, we focus on three common types of entities that all datasets contain: *persons, locations, and organizations*. Each sentence from those datasets is extracted as a data point, and the associated label set is simply all entities in this sentence. An entity is considered correctly extracted if and only if it is labeled as an entity and its entity type is correct.

**GitHub model cost** We evaluate the inference time of all GitHub models on an Amazon EC2 p2.x instance, which is \$0.90 per hour. For multi-label image classification, the GitHub model (SSD) takes 6s to classify each image, resulting in an equivalent cost of \$0.0015 per image. For the named entity task, the GitHub model (Spa) can extract the entities from a sentence in 0.015s, leading to \$ 0.00000375 per sentence. The GitHub model (Pad) with the mobile version 3.0 text detector and recognizer requires 1.5s on average to extract text from an image, causing a cost of \$ 0.000375 per image. Compared to the commercial APIs, this cost is much cheaper.

**Training cost of FrugalMCT** Both dollar cost and computation time of training are often much smaller than ML APIs inference cost. This is because (i) training is a one-time cost and (ii) FrugalMCT requires a small number of label annotations (a few thousands see Figure 5). Consider the image tagging task as an example: the dollar cost of calling all APIs is  $\$0.0006 + \$0.001 + \$0.0015 = \$0.0031$  per image. Labeling for (say) five thousands images takes  $\$0.0031 \times 5000 = \$15.5$ . Training a FrugalMCT strategy on half of the COCO dataset takes 59.5s on the experiment machine. This is much cheaper than calling the selected APIs after at large scale (e.g., millions of images)

**Case study on COCO** Now we provide more details about the case study on the multi-label image classification dataset, COCO. There are in total 123,287 images containing labels from 80 different categories in COCO. Figure 6 gives the label distribution. First note that the label distribution is quite skewed. overall, the label person is the most frequent: more than 50% of the images contain the person label. Among others, car, chair, and dining tables are also quite common labels

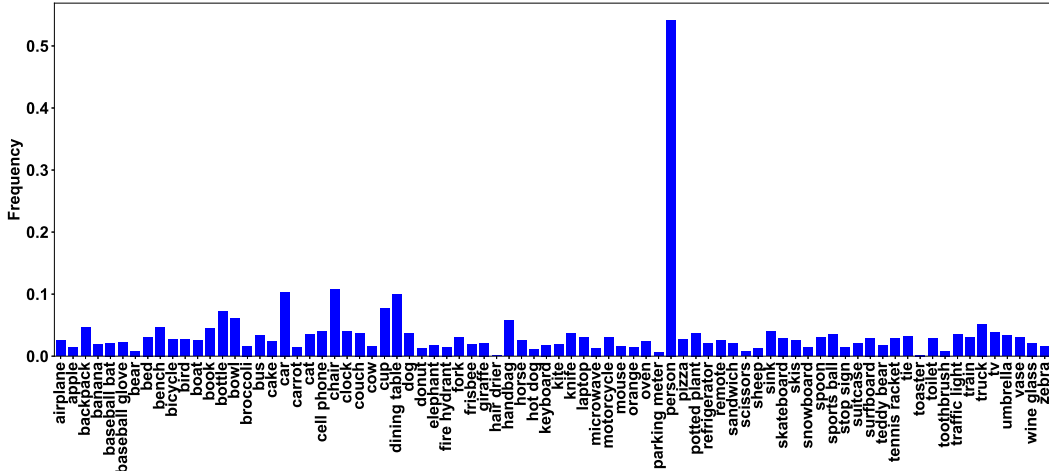


Figure 6: Label Distribution on COCO.

in this dataset with more than 10% occurrence. On the other hand, there are also quite some rare labels. For example, half driver and toaster appear in less than 1% of the images. Such imbalance between different labels imposes a high data and computational complexity to directly apply previous approach that learns a decision rule per label, and thus verifies the necessity of the proposed framework, FrugalMCT.

To further understand when and why FrugalMCT gives a better performance than single API, we present the precision and recall per class for each API, majority vote, and FrugalMCT in Figure 7 and Figure 8. We first note that there is no API universally better than other APIs for each label. For example, GitHub and Microsoft APIs can hardly correctly predict the label “toaster”, but Everypixel and Google APIs have a relatively high accuracy on label “toaster”. On the other hand, Everypixel has a low accuracy on label “kite” and “knife”, while Microsoft, Google, GitHub APIs can usually predict those labels with higher accuracy. This implies that combining different APIs may produce an accuracy better than any single one of them. There are also some easy labels on which all APIs give a high accuracy. For example, on the label “zebra”, all APIs give a 90% precision and recall. This actually suggests that it is not always necessary to use all API. For example, if GitHub predicts an image has the label “zebra”, and we know there is no other labels in this image, then probably there is no need to call any other APIs.

Another interesting observation is that FrugalMCT improves the precision and recall for almost every label compared to any single API. This is primarily because FrugalMCT appropriately utilizes the predicted label information from GitHub model to infer which API is better on certain input, and combine its performance with the base API aptly. Yet, the precision and recall difference can be quite different for different APIs. For example, as shown in Figure 8(c), the recall for “airplane” is much higher than its precision, but banana’s precision is much higher than its recall. For applications that have specific precision and recall requirements, we may adopt different accuracy metrics in FrugalMCT. Another interesting observation is that the precision and recall for some labels is extremely. For example, “hair drier” cannot be predicted by FrugalMCT, which is due to that no API actually predicts this label correctly. How to extend FrugalMCT to recognize unseen labels remains an open question.

**Ensemble method comparison** For comparison, we compare FrugalMCT against FrugalML as well as two ensemble methods, majority vote and weighted majority vote. In majority vote, for each label, we accept it if at least half of the APIs predict it. In weighted majority vote, we assign each API’s accuracy as its weight. Next, for each label, we compute a label score, which is equal to the sum of each API’s confidence score on this label weighted by its corresponding weight. If an API does not predict a label, then its confidence score is viewed as 0. Finally, we only accept the label if its label score is larger than a threshold. We pick a threshold that maximizes the overall accuracy by grid search.



Table 4: Performance of FrugalMCT’s accuracy predictor. Root mean square error (RMSE) quantifies the standard deviation of the differences between predicted and true accuracy.

Data	RMSE	Data	RMSE	Data	RMSE
PASCAL	0.28	MIR	0.22	COCO	0.24
MTWI	0.17	ReCTS	0.22	LSVT	0.19
CONLL	0.29	ZHNER	0.31	GMB	0.28

Table 5: Comparison of ensemble methods as well as cost-aware approaches. For FrugalMCT and FrugalML, we pick their corresponding strategies that minimize the cost while ensures that the accuracy reaches the highest possible.

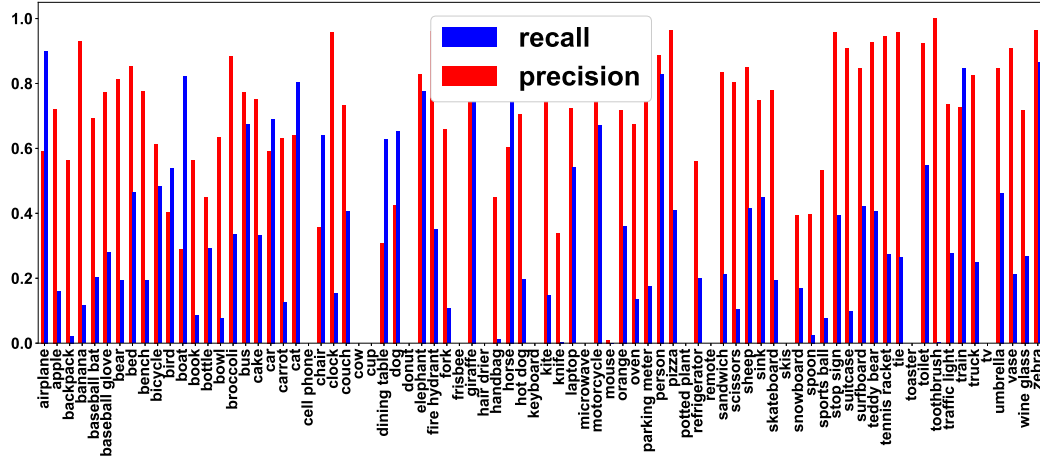
	best single API		FrugalML		FrugalMCT		majority vote		weighted maj vote	
	acc	cost	acc	cost	acc	cost	acc	cost	acc	cost
PASCAL	74.8	10	76.9	11	78.5	8	77.8	31.01	77.8	31.01
MIR	41.2	10	43.8	8	49.2	14	41.4	31.01	48.7	31.01
COCO	47.5	10	49.3	8	54	12	50.1	31.01	52.8	31.01
MTWI	67.9	210	68.1	213	71.1	208	75.4	275.01	75.4	275.01
ReCTS	61.3	210	63.4	213	64.7	208	70.2	275.01	70.2	275.01
LSVT	53.8	210	56.2	213	57.2	208	62.8	275.01	62.8	275.01
CONLL	52.6	3	55.7	32	56.8	36.8	58.5	43.01	58.5	43.01
ZHNER	61.3	30	67.4	31.2	71.8	36.8	66	43.01	66	43.01
GMB	50.1	30	52.6	30.1	53.1	20.5	51.3	43.01	51.5	43.01

The results are summarized in Table 5. Overall, we observe that FrugalMCT and ensemble methods have similar performance across different tasks and datasets, but with a much lower cost. In fact, for datasets including COCO and ZHNER, FrugalMCT can achieve an accuracy even higher than ensemble methods.

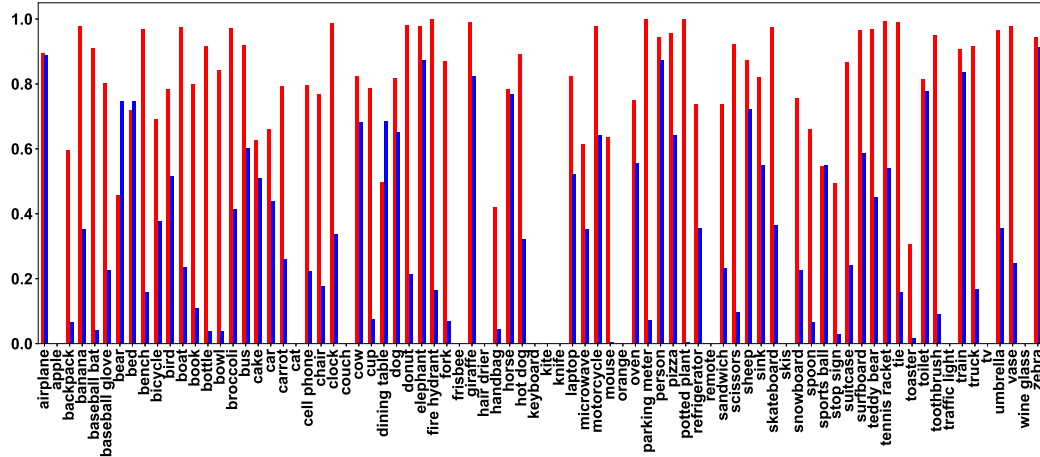
**Accuracy predictor performance** Note that FrugalMCT’s performance highly depends on its accuracy predictors’ performance. To obtain a quantitative sense of the accuracy predictors, we evaluate the accuracy predictors’ performance in Table 5. RMSE measures the standard deviation of the difference between accuracy predictor’s output and the corresponding true accuracy. PCC stands for Pearson correlation coefficient, which roughly measures the linear correlation between the true accuracy and the predicted value from the accuracy predictors. Overall, FrugalMCT’s random forest predictors enjoy a much smaller RMSE and higher PCC than DAP (the dummy accuracy predictors), which matches the fact that FrugalMCT gives a higher end to end performance than using the DAP.

Table 6: Accuracy predictor performance. RMSE and PCC stand for root mean square error and Pearsons correlation coefficient.

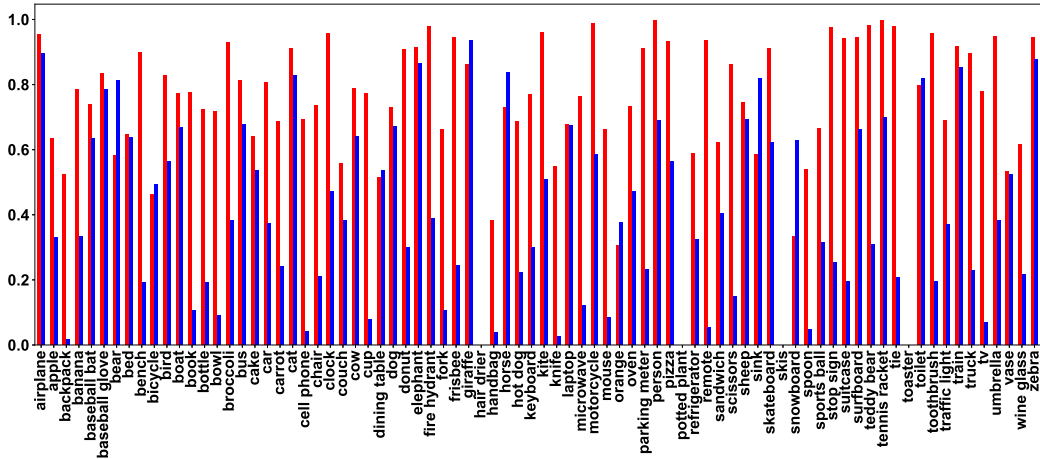
Data	RMSE		PCC	
	FrugalMCT	DAP	FrugalMCT	DAP
PASCAL	0.28	0.35	0.55	0.012
MIR	0.22	0.31	0.55	-0.013
COCO	0.24	0.31	0.63	0.001
MTWI	0.17	0.21	0.57	0.004
ReCTS	0.22	0.27	0.57	0.001
LSVT	0.19	0.24	0.61	-0.003
CONLL	0.29	0.41	0.72	-0.003
ZHNER	0.31	0.36	0.48	-0.005
GMB	0.28	0.40	0.69	-0.006



(a) GitHub

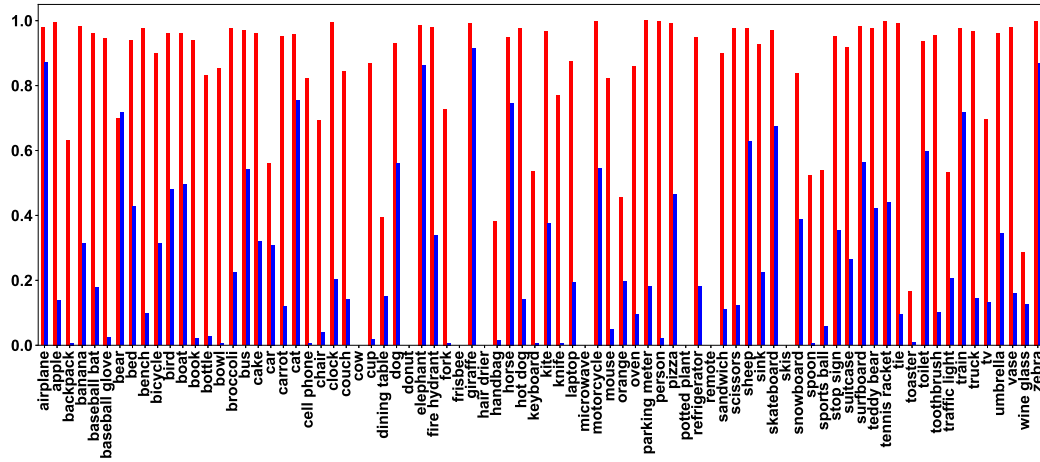


(b) Everypixel

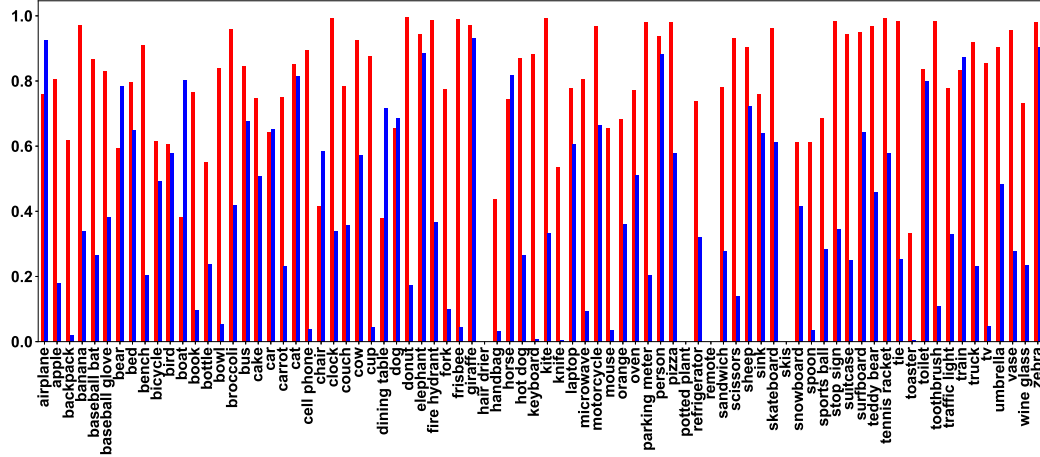


(c) Microsoft

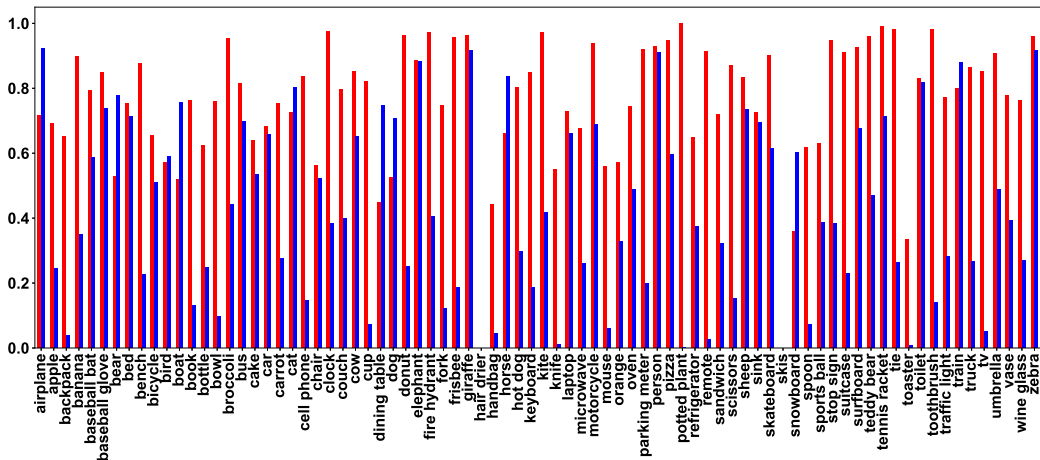
Figure 7: The per class precision and recall of different APIs .



(a) Google



(b) Majority Vote



(c) FrugalMCT

Figure 8: The per class precision and recall of different APIs .