APPENDIX

A. DERIVATIONS OF EQUATIONS 8 AND 9

We first show step by step how the isrmed gradient difference between two instances can be bounded by the last layer of the neural network.

Specifically, considering a T-layer perception, we define $\varphi^{(t)}(.)$ as the Lipschitz continuous activation function for layer t, and $\theta^{(t)}$ is the weight matrix for layer t. We simply define $p(l|x_i)$ as p_1 and define $p(l|x_j)$ as p_2. To compute the gradient of the loss function f with respect to the weights $\theta^{(T)}$ in the final layer, we use backpropagation as follows:

i).**Output of the neural network**:

- Let $o_i^{(T-1)}$ be the input to the final layer T.
- The output of the final layer before activation is $z_i^{(T)} = \theta^{(T)} \cdot o_i^{(T-1)}$.
- The activated output is $o_i^{(T)} = \varphi^{(T)}(z_i^{(T)})$.

ii). **Gradient with respect to θ**: By the chain rule, we have:

$$\begin{aligned} p_1 \nabla f_i(\theta) = p_1 \frac{\mathrm{d}f_i}{\mathrm{d}\theta} &= p_1 \frac{\mathrm{d}f_i}{\mathrm{d}o_i^{(T)}} \cdot \frac{\mathrm{d}o_i^{(T)}}{\mathrm{d}z_i^{(T)}} \cdot \frac{\mathrm{d}z_i^{(T)}}{\mathrm{d}\theta} \\ &= p_1 \frac{\mathrm{d}f_i}{\mathrm{d}o_i^{(T)}} \cdot \frac{\mathrm{d}o_i^{(T)}}{\mathrm{d}z_i^{(T)}} \cdot \frac{\mathrm{d}z_i^{(T)}}{\mathrm{d}z_i^{(T-1)}} \cdot \frac{\mathrm{d}z_i^{(T-1)}}{\mathrm{d}z_i^{(T-2)}} \cdot \ldots \cdot \frac{\mathrm{d}z_i^{(2)}}{\mathrm{d}z_i^{(1)}} \cdot \frac{\mathrm{d}z_i^{(1)}}{\mathrm{d}\theta} \\ &= p_1 \nabla f_i^{(T)}(\theta) \cdot \varphi'^{(T)}(z_i^{(T)}) \cdot [\theta^{(T)} \cdot \varphi'^{(T)}(z_i^{(T-1)})] \\ &\cdot [\theta^{(T-1)} \cdot \varphi'^{(T-1)}(z_i^{(T-2)})] \cdot \ldots \cdot [\theta^{(2)} \cdot \varphi'^{(2)}(z_i^{(1)})] \cdot (o_i^{(0)})^\intercal \\ &= p_1 \nabla f_i^{(T)}(\theta) \cdot \varphi'^{(T)}(z_i^{(T)}) \cdot \Omega_i^{(T)} \cdot (o_i^{(0)})^\intercal \end{aligned} \tag{1}$$

where $\Omega_i^{(T)}$ is denoted by $[\theta^{(T)} \cdot \varphi'^{(T)}(z_i^{(T-1)})] \cdot [\theta^{(T-1)} \cdot \varphi'^{(T-1)}(z_i^{(T-2)})] \cdot \ldots \cdot [\theta^{(2)} \cdot \varphi'^{(2)}(z_i^{(1)})]$.

iii). **Upper bound of approximation error**: The maximum distance between two gradients in the whole parameter space Θ could be expressed as:

$$\begin{aligned} &\|p_1 \nabla f_i(\theta) - p_2 \nabla f_j(\theta)\| \\ &= \|p_1 \Omega_i^{(T)} \varphi'^{(T)}(z_i^{(T)}) \nabla f_i^{(T)}(\theta)(o_i^{(0)})^\intercal - p_2 \Omega_j^{(T)} \varphi'^{(T)}(z_j^{(T)}) \nabla f_j^{(T)}(\theta)(o_j^{(0)})^\intercal\| \\ &= \|p_1 \Omega_i^{(T)} \varphi'^{(T)}(z_i^{(T)}) \nabla f_i^{(T)}(\theta)(o_i^{(0)})^\intercal - p_1 \Omega_i^{(T)} \varphi'^{(T)}(z_j^{(T)}) \nabla f_j^{(T)}(\theta)(o_i^{(0)})^\intercal \\ &+ p_1 \Omega_i^{(T)} \varphi'^{(T)}(z_j^{(T)}) \nabla f_j^{(T)}(\theta)(o_i^{(0)})^\intercal - p_2 \Omega_j^{(T)} \varphi'^{(T)}(z_j^{(T)}) \nabla f_j^{(T)}(\theta)(o_j^{(0)})^\intercal\| \\ &\leq \|p_1 \Omega_i^{(T)}\| \cdot \|o_i^{(0)}\| \cdot \|\varphi'^{(T)}(z_i^{(T)}) \nabla f_i^{(T)}(\theta) - \varphi'^{(T)}(z_j^{(T)}) \nabla f_j^{(T)}(\theta)\| \\ &+ \|\varphi'^{(T)}(z_j^{(T)}) \nabla f_j^{(T)}(\theta)\| \cdot \|p_1 \Omega_i^{(T)} o_i^{(0)} - p_2 \Omega_j^{(T)} o_j^{(0)}\| \\ &\leq \|p_1 \Omega_i^{(T)}\| \cdot \|o_i^{(0)}\| \cdot \|\varphi'^{(T)}(z_i^{(T)}) \nabla f_i^{(T)}(\theta) - \varphi'^{(T)}(z_j^{(T)}) \nabla f_j^{(T)}(\theta)\| \\ &+ \|\varphi'^{(T)}(z_j^{(T)}) \nabla f_j^{(T)}(\theta)\| \cdot \|p_1 \Omega_i^{(T)} o_i^{(0)} - p_2 \Omega_i^{(T)} o_i^{(0)}\| \\ &+ \|\varphi'^{(T)}(z_j^{(T)}) \nabla f_j^{(T)}(\theta)\| \cdot \|p_2 \Omega_i^{(T)} o_i^{(0)} - p_2 \Omega_j^{(T)} o_j^{(0)}\| \\ &\leq p_1 \cdot \|\Omega_i^{(T)}\| \cdot \|o_i^{(0)}\| \cdot \|\varphi'^{(T)}(z_i^{(T)}) \nabla f_i^{(T)}(\theta) - \varphi'^{(T)}(z_j^{(T)}) \nabla f_j^{(T)}(\theta)\| \\ &+ \|\varphi'^{(T)}(z_j^{(T)}) \nabla f_j^{(T)}(\theta)\| \cdot (\|p_1 - p_2\| \cdot \|\Omega_i^{(T)} o_i^{(0)}\| + p_2 \cdot \|\Omega_i^{(T)} o_i^{(0)} - \Omega_j^{(T)} o_j^{(0)}\|) \\ &\leq S_{ij} = p_1 \cdot n_1 \cdot \|\varphi'^{(T)}(z_i^{(T)}) \nabla f_i^{(T)}(\theta) - \varphi'^{(T)}(z_j^{(T)}) \nabla f_j^{(T)}(\theta)\| \\ &+ \|\varphi'^{(T)}(z_j^{(T)}) \nabla f_j^{(T)}(\theta)\| \cdot (\|p_1 - p_2\| \cdot n_2 + p_2 \cdot n_3) \end{aligned} \tag{2}$$

where $n_1 = \max\limits_{T,i}(\|\Omega_i^{(T)}\| \cdot \|o_i^{(0)}\|)$, $n_2 = \max\limits_{T,i}(\|\Omega_i^{(T)} o_i^{(0)}\|)$ and $n_3 = \max\limits_{T,i,j}(\|\Omega_i^{(T)} o_i^{(0)} - \Omega_j^{(T)} o_j^{(0)}\|)$ are constants.

B. PROOF OF THEOREM 1 AND THEOREM 2

Next, we will show that solving Equation 7 is an NP-hard problem with submodular property. This allows us to design a greedy algorithm that effectively solves this problem with an approximate ratio.

Theorem 1. *The problem of subset selection under uncertainty is NP-hard.*

Proof. Consider a scenario where every instance in D is assigned to a hard label (with a probability of 1). Then the problem simplifies to $C = \arg\min\limits_{C \subseteq D} \sum_{i=1}^{N} \min_{c_j \in C} \|\nabla f_i(\theta) - \nabla f_j(\theta)\|, |C| \leq K$. Naturally, the K-medoid problem [1] can be reduced to the special case. Therefore, our problem is also NP-hard.

Theorem 2. *The problem of subset selection under uncertainty shows the submodular property.*

Proof. We define the utility function as $B(C, \theta) = \sum_{i=1}^{N} \max_{c_j \in C} u_{ij}$, where $u_{ij} = 1-$ normalized$(\max\limits_{\theta \in \Theta} \|p(l|x_i)\nabla f_i(\theta) - p(l|x_j)\nabla f_j(\theta)\|)$. The subset selection under uncertainty problem is equivalent to maximizing utility B. If B has the submodular property, for any $C \subseteq C^* \subseteq D$ and $o_i \in D \setminus C^*$, we have to prove (1)B is monotonous, *i.e.*, $B(C \cup \{o_i\}, \theta) \geq B(C, \theta)$, and (2) B has the diminishing marginal returns property, *i.e.*, $B(C \cup \{o_i\}, \theta) - B(C, \theta) \geq B(C^* \cup \{o_i\}, \theta) - B(C^*, \theta)$. For simplicity, we use $B(o_i|C, \theta)$ to denote $B(C \cup \{o_i\}, \theta) - B(C, \theta)$ in the following parts of the paper. The proof starts with considering u_{ij} to be known, which will be computed in Section **??**. In this situation, each instance in D will be assigned to an instance of the subset that maximizes the utility.

For (1), when o_i is added into C, if no instance in D will be assigned to o_i, then $B(C \cup \{o_i\}, \theta) = B(C, \theta)$. If one or more instances in D are assigned to o_i, clearly $B(C \cup \{o_i\}, \theta) > B(C, \theta)$. Hence, B is monotonous.

For (2), we can see that $B(C, \theta)$ is the sum of different terms w.r.t. different instances, and they are computed independently. Therefore, if there is only a single instance and the diminishing marginal returns property satisfies, then B has the property. Suppose that the instance is denoted by o_*. Given C, C^* and $\{o_i\}$, we prove the diminishing marginal returns for o_* in all possible three cases of $c_* = \arg \max\limits_{c_j \in C \cup (C^* \setminus C) \cup \{o_i\}} u_{*j}$.

[Case 1: $c_* \in C$**]** In this case, obviously, $B(C \cup \{o_i\}, \theta) - B(C, \theta) = B(C^* \cup \{o_i\}, \theta) - B(C^*, \theta) = 0$ because o_* will not change its assignment when $C^* \setminus C$ and o_i are added.

[Case 2: $c_* \in C^* \setminus C$**]** Apparently, $B(C^* \cup \{o_i\}, \theta) - B(C^*, \theta) = 0$, which must be smaller than $B(C \cup \{o_i\}, \theta) - B(C, \theta)$.

[Case 3: $c_* = o_i$**]** There are two cases here.

1. If $\max\limits_{c_j \in C} u_{*j} \geq \max\limits_{c_j \in C^* \setminus C} u_{*j}$, $B(C \cup \{o_i\}, \theta) - B(C, \theta) = B(C^* \cup \{o_i\}, \theta) - B(C^*, \theta) > 0$.
2. If $\max\limits_{c_j \in C} u_{*j} < \max\limits_{c_j \in C^* \setminus C} u_{*j}$, $B(C \cup \{o_i\}, \theta) - B(C, \theta) > B(C^* \cup \{o_i\}, \theta) - B(C^*, \theta) > 0$. The reason is that when $C^* \setminus C$ is added to C, o_* will change its assignment and thus the utility is increased. Afterwards, o_i is added, and the utility is further increased.

Table 1: Epochs of All the Methods for Each Dataset.

	HERDING	GradMatch	DEEPFOOL	M-DYR-H	Co-teaching	*k*NN	Cleanlab	Cleanlab-S	MisDetect	Direct-Training	ActiveClean	Deem
CoverType	62	68	78	195	190	135	145	80	153	196	89	75
IMDB-Large	86	95	103	202	193	125	136	76	147	197	112	95
SVHN	87	93	105	204	200	136	144	86	150	204	102	100
MNIST	39	40	53	101	86	60	63	50	75	103	58	42
CIFAR-10	50	53	65	147	117	101	98	65	106	150	73	57
Clothing1M	76	79	87	210	157	130	126	89	146	208	104	83

Table 2: Proportion of Cleaning Time to Model Training Time for Cleanlab and Cleanlab-S.

	CoverType	IMDB-Large	SVHN	MNIST	CIFAR-10	Clothing1M
Cleanlab	10.2m(6.4%)	186.2m(7.4%)	2.2m(7.2%)	1.8m(16.2%)	2.1m(9.5%)	119.8m(7.9%)
Cleanlab-S	10.2m(31.5%)	186.2m(19.6%)	2.2m(19.1%)	1.8m(29.0%)	2.1m(25.6%)	119.8m(21.7%)

C. NUMBER OF EPOCHS OF EACH BASELINE

The number of epochs of all the methods that have been trained on each dataset are listed in Table 1.

D. NUMBER OF EPOCHS OF EACH BASELINE

For Cleanlab and Cleanlab-S, we have detailed the cleaning time as a percentage of the model training duration in Table 2.

E. DECOMPOSED EXECUTION TIME OF DEEM

The execution time of Deem consists of (i) initial model training for a few epochs, (ii) subset select/update and (iii) training for a few more epochs after the subset selection. We measure the execution time of each individual component and calculate the ratio of execution time of each component relative to the overall runtime. The decomposed execution time of each component is listed in Table 3 respectively.

Table 3: The Decomposed Execution Time of Each Component of Deem.

	CoverType	IMDB-Large	SVHN	MNIST	CIFAR-10	Clothing1M
(i)	4.0m(17.5%)	93.1m(15.1%)	1.3m(15.8%)	0.9m(16.1%)	1.2m(15.8%)	65.0m(14.1%)
(ii)	1.7m(12.4%)	75.3m(12.2%)	0.9m(11.0%)	1.0m(17.9%)	1.1m(14.5%)	73.4m(16.0%)
(iii)	16.1m(70.1%)	450.2m(72.7%)	6.0m(73.2%)	3.7m(66%)	5.3m(69.7%)	321.7m(69.9%)

REFERENCES

[1] M. R. Garey and David S. Johnson. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman.