mathematics

MDPI

Article

Bounds on Performance for Recovery of Corrupted Labels in Supervised Learning: A Finite Query-Testing Approach

Jin-Taek Seong

Graduate School of Data Science, Chonnam National University, Gwangju 61186, Republic of Korea; jtseong@jnu.ac.kr

Abstract: Label corruption leads to a significant challenge in supervised learning, particularly in deep neural networks. This paper considers recovering a small corrupted subset of data samples which are typically caused by non-expert sources, such as automatic classifiers. Our aim is to recover the corrupted data samples by exploiting a finite query-testing system as an additional expert. The task involves identifying the corrupted data samples with minimal expert queries and finding them to their true label values. The proposed query-testing system uses a random selection of a subset of data samples and utilizes finite field operations to construct combined responses. In this paper, we demonstrate an information-theoretic lower bound on the minimum number of queries required for recovering corrupted labels. The lower bound can be represented as a function of joint entropy with an imbalanced rate of data samples and mislabeled probability. In addition, we find an upper bound on the error probability using maximum a posteriori decoding.

Keywords: supervised learning; corrupted label; query testing; upper bound; lower bound

MSC: 68T07

check for updates

Citation: Seong, J.-T. Bounds on Performance for Recovery of Corrupted Labels in Supervised Learning: A Finite Query-Testing Approach. *Mathematics* **2023**, *11*, 3636. https://doi.org/10.3390/math11173636

Academic Editors: Fan Zhang, Songhe Feng, Yongsheng Zhou and Junlin Hu

Received: 12 July 2023
Revised: 15 August 2023
Accepted: 22 August 2023
Published: 23 August 2023

1. Introduction

Recently, deep neural networks using large datasets showed successful roles in a number of machine learning tasks, such as object recognition [1,2], prediction systems [3–5], and natural language processing [6–8]. This success is due to the availability of vast but well-labeled data. However, the problem with obtaining such labeled data is that it is time-consuming and expensive. Source labeling platforms, such as Amazon's Mechanical Turk, a kind of non-expert system, are now widely used for reducing costs. Nevertheless, in some cases, unreliable labels are frequently generated [9–12]. In addition, labeling data is sometimes a complex and difficult task, even for experienced domain experts [13,14]. Even adversarial attacks lead to manipulated labeling results [15]. We call unreliable labels corrupted or noisy labels because they are assigned label values that are different from the true label values. The proportion of noisy labels in the current datasets was reported from 8.0% to 38.5% [16–19]. The training of deep neural networks with noisy labels is more sensitive to overfitting [20,21]. Improved methods have been developed to better learn from these corrupted labels, including regulation techniques, the structure of deep neural networks, removing unreliable data, and improving loss functions [22]. Our finite query-testing scheme is considered in the context of a supervised learning problem, which is essentially a structure that requires labeled data. The reliability of the labels affects the overall performance with respect to the training process. In particular, incorrect labels mainly lead to overfitting. In this work, we focus on the supervised learning problem to improve the accuracy of the labels rather than semi-supervised and unsupervised learning.

Suppose there are N (i.e., $N > 1000$) data samples that are labeled, where a subset of samples is corrupted due to labeling by non-experts. The goal of this work is to find noisy data using a finite query-testing scheme in supervised learning. It is clear that label experts identifying all the data one by one is extremely costly and exhausting. We aim to find a

subset of the given noisy data from non-experts while keeping the number of queries for identifications by experts as small as possible, which is approximately greater than $\frac{H(\mathbf{s})}{\log K}$, where $\mathbf{s}$ denotes the mislabeled vector, K is the number of classification, and $H()$ is the entropy based 2. For a multiple classification problem in supervised learning, mislabeled data samples are corrected to the ground truth using the finite query-testing scheme. Each data sample is considered a finite value expressed as 0 and $K-1$ for K classifications. In this work, experts can specify K labels for N data samples using crowdsourcing approaches. Crowdsourcing, for example, asks experts: Do the two data samples belong to the same category? Or is it the same? Through such queries, experts identify whether multiple data samples are the same or different at once [23]. It then recovers the labeling based on the crowdsourced queries. This idea was used in entity resolution studies [24,25]. Since each labeling from experts takes time and money, our goal is to accurately identify data samples while minimizing the number of queries for crowdsourcing. Recently, query algorithms to solve this problem have been proposed and developed [24–26]. Experts can potentially process more than two data samples at a time. If possible, it is good to take the number of samples involved in the query to be as small as possible. Also, instead of answering 'same class' type queries, experts use a few simple functions on small input datasets.

Queries for experts are classified into adaptive and non-adaptive schemes. Adaptive and nonadaptive schemes have been proposed for the crowdsourcing problem. An adaptive query like the equal class is an approach that can exactly recover all data samples. Nonadaptive cases require queries proportional to the square of the number of samples in multiple classes. This means that adaptive algorithms can achieve better performance than nonadaptive ones in the absence of additional information. In [25], a condition on how to reduce this gap is discussed. Despite the existence of the advantages of the adaptive algorithm as mentioned above, in such a case where it performs in crowdsourcing, a nonadaptive scheme that can be processed in parallel, which is advantageous for efficient processing time, is more realistic [25]. In practice, adaptive algorithms are additionally considered slow processing times, as crowdsourcing can result in long response times by experts. In this paper, we consider a nonadaptive scheme of our problem. That is, we perform the recovery of the corrupted labels after obtaining all query responses.

In [27], the authors show that a semi-supervised clustering model is equivalent to locally encodable source coding. Then, they find bounds on the minimum number of queries by performing a labeling recovery problem using XOR and AND queries for binary classification. In addition, in [28], the authors characterize information-theoretic bounds on the optimal number of queries to reliably recover labels given a combination of queries with constant d-degrees and noise parameters in binary classification. Both [27,28] are similar to our work but differ in the following ways. The main difference is that while in the two papers [27,28], they address the recovery problem for binary classification, we extend the problem to multiple classification. In addition to this, we use finite field operations to replace the XOR and AND operations used for binary classification. Furthermore, in our work, in finding a bound for the minimum number of queries, we obtain an upper bound using the MAP (maximum a poesteriori) decoding method. With this work, we not only compare the lower and upper bounds to each other but also show bounds on the performance of recovering corrupted data.

In this paper, we consider the problem of recovering corrupted labels in supervised learning. In deep neural networks for multiple classification, a small subset of data samples with true labels are inverted due to labeling errors. So a part of the entire dataset, corrupted by the labeling of non-experts, like an automatic classifier, is used for training. Our task is to identify the corrupted data samples with a minimum number of expert queries and recover them to their true labels. To this end, we aim to recover corrupted data samples with the assistance of an additional expert, called a finite query-testing system. In this paper, the query-testing system randomly selects a subset of data samples and responds to the results of their combinations via finite field operations. And we obtain a lower bound

on the minimum number of queries to recover corrupted labels and an upper bound on the error probability with MAP decoding.

2. Related Work

Recovering unknown signals from arbitrary sets of combinations is called a constraint satisfaction problem. It has been an active research topic in computer science, statistics, and biology, and has inspired machine learning and cryptography [29]. In particular, our problem is in line with the random planted d-XOR satisfaction problem. It aims to find a small number of binary variables that satisfy a set of constraints that the XOR combination of a set of randomly and uniformly chosen variables should be equal to 0 or 1. For the pairwise queries, it has been mainly focused on the field of community detection [30]. It was shown that the information-theoretic bound for optimal sample complexity is achieved when measurements are compromised by binary symmetric noise. In [25], the authors considered the case where the number of communities could be greater than two, and proved an information-theoretic bound on the number of connections and an estimated approach went to this bound. The parity codes with greater than three degrees was considered in terms of locally encodable coding or the community recovery of hypergraphs [31]. In the case of even degrees, the authors in [30] considered the phase transition for the recovery of optimal complexity. In addition, in [31], the authors showed that the recovery of optimal complexity for parity codes with a constant degree scales as the order of degree grows, up to an order of log scales.

When the error rate in channel coding theory is not known, the code design is an important issue, which has been an intensive research field over the past half century. If data transmitted from a source through a communication channel with random loss arrive at a destination, they are modeled as a binary erasure channel with a random error rate. Unlike repeat codes that are used for TCP/IP schemes that cause long delivery delays due to a feedback channel from the destination, fountain codes called forward error correction codes have shown successful performance in delivering data reliably [32]. Suppose there is a set of input symbols, the fountain code generates rateless parity codes called output symbols. Fountain codes guarantee that input symbols can be recovered with high probability at a small overhead symbol for any set of output codes. Many efforts have been made to design practical fountain codes that achieve the reliable recovery of information bits with small overhead and low encoding and decoding complexity. Our construction of a query matrix is closely related to the fountain code design because choosing a subset for coded labels is like designing a codeword. However, the fountain codes do not take into account the ability of experts to respond to complex data labels. More specifically, it is a good idea to use the relationship between the label difficulty and the number of data samples to be combined because the higher the difficulty of the label, the more the expert must identify the labels one by one.

Semi-supervised learning is a technique that allows you to train a model without having to label some of the data. There are two main ways to train a model with noisy data. The first is to train a robust model that performs well under noisy labels. This can be done by using regularization or loss functions to avoid overfitting to noisy labels. The second is to detect and remove data that are likely to be a noisy label, and then train the model with the remaining data. The co-teaching method [33] helps detect noisy data by having two networks train together, with one network selecting data to use in training the other. In [34], O2U-NET repeated learning from underfitting to overfitting several times by changing the learning rate as a parameter. It detected noisy labels by using the average of losses for each datum and improved the reliability of the performance. In addition, in [35], the authors computed the average of the losses using an exponential moving average to determine the noisy labels more reliably than a simple average.

3. Finite Query-Testing Scheme

3.1. Problem Statement

In this subsection, we define a finite query-testing scheme for the reconstruction of corrupted labels in supervised learning. First, we consider a random vector $\mathbf{x}$ of size N, $\mathbf{x} = (x_1, x_2, \ldots, x_N)^T$. The value of each element of $\mathbf{x}$ is to represent K classification. For $i \in \{1, 2, \ldots, N\}$, $x_i = 0$ means that the i-th data are the first classification, and when the value is $K-1$, $x_i = k-1$, it corresponds to the K-th classification. In the end, $\mathbf{x}$ is represented by a random vector of size N with K values. In this work, we consider labeling imbalanced data. Imbalanced data are a dataset in which the labeling distribution of K categories is such that the majority of data belong to a particular category, and the minority of data belong to the other categories. For example, classifying a disease from a patient's diagnostic image or identifying a defect from observing the condition of a product are examples of imbalanced data. This is when the majority of the classification labeling is normal and a small percentage of the data is abnormal. For such imbalanced data, we assume that each element of the random vector has the following probability distribution. We assume a probabilistic model, where the first classification appears with the greatest probability and the others occur evenly:

$$\Pr(x_i = \theta) = \begin{cases} 1-p & \text{if } \theta = 0, \\ \frac{p}{K-1} & \text{if } \theta \neq 0, \end{cases} \tag{1}$$

where p is the imbalance rate of classification, $1-p$ is the probability of classifying as normal, and θ is a dummy variable. We assume that the imbalance ratio is less than 0.5, $p < 0.5$. In other words, we restrict ourselves to classification problems, where the probability of being a certain classification (e.g., normal) is greater than or equal to the majority. We define data with K classes, and now we aim to describe how an expert oracle (labeler or worker) can use a finite field to combine labeling tasks and collect data.

Now, we consider a situation where the oracle has mislabeled the data contained in a given query. The main benefit of our query-testing system is that we can find a true label vector $\mathbf{x}$ with the correction of data labeling, even if the oracle responds with a mislabeled answer. The reason we consider this situation is that the oracle's skill and expertise can lead to mislabeling. For example, in medical imaging, the same data can be labeled differently by experts with different levels of experience and proficiency. Let $\mathbf{c}$ be a corrupted label vector that contains the error corresponding to $\mathbf{x}$, where $\mathbf{c} = (c_1, c_2, \ldots, c_N)^T$. We represent the transition from a true label vector $\mathbf{x}$ to a corrupted label vector $\mathbf{c}$ due to mislabeling using the following conditional probability with independently identical distribution for each element:

$$\Pr(c_i | x_i) = \begin{cases} 1-\gamma & \text{if } c_i = x_i, \\ \frac{\gamma}{K-1} & \text{if } c_i \neq x_i, \end{cases} \tag{2}$$

where γ is the mislabeling probability that the oracle classifies the given data differently in the K categories. In fact, this mislabeling probability is very small. In our work, it is possible to perfectly recover the original true vector $\mathbf{x}$ by considering this mislabeling probability. Let $\mathbf{s}$ be a sparse vector represented by addition on a finite field between the corrupted label vector $\mathbf{c}$ and the true vector $\mathbf{x}$, as $\mathbf{s} = \mathbf{c} \oplus \mathbf{x}$, where the symbol $\oplus$ is the addition on finite fields. Note that the vector $\mathbf{c}$ is labeled with small mislabeled probability, then, using the following relationship, $\mathbf{s} \oplus \mathbf{c} = \mathbf{x}$, we identify the original true vector $\mathbf{x}$. Herein, the vector $\mathbf{s}$ is sparse because the mislabeling probability γ is very small. In other words, $\mathbf{s}$ is a sparse vector with mostly zero elements. Therefore, we see that finding a sparse vector is equivalent to recovering a true label vector.

In our query-testing scheme, an oracle responds to labeling information by querying the data. A query can be thought of as a random subset of data that is composed of one or more data. For example, a subset of a sparse vector consisting of data 1, data 5, and data 7 is requested to the oracle as a single query (e.g., a query containing three data on a single screen on the web). For each data (data 1, 5, and 7), the oracle performs labeling

for each of the K categories. The oracle then combines the results of each of the three data to answer the query once. Here, the query and response are performed once, but the oracle performs the labeling on three data. We assume that there exist M queries in our query-testing scheme, $j \in \{1, 2, \ldots, M\}$. Let us vectorize all the y_js and call $\mathbf{y}$ the response vector, as $\mathbf{y} = (y_1, y_2, \ldots, y_M)^T$. We define the j-th response y_j to the query labeled by the oracle using a finite field with size K as follows:

$$y_j = \sum_{i \in \mathcal{R}_j} s_i, \tag{3}$$

where we let $\mathcal{R}_j$ be the data selection set to be included in the j-th query, which is the set of indices of the data to be included in the j-th query that tells us which of the N data of the sparse vector $\mathbf{s}$ to include to combine with a single query. For the previous example, it would be $\mathcal{R}_j = \{1, 5, 7\}$. And the addition and multiplication operations in (3) are performed on finite fields. Note that the value of the response y_j does not imply K classification but rather addition on a finite field. And actual data agents do not collect the element s_i; they collect y_j. If the data to be labeled are sensitive and need to be protected, the oracle does not pass s_i directly to the data agent. Instead, it passes y_j, which is the result of combinations in a finite field on the labeled values of the data in the query. This has a privacy benefit because it essentially does not store the original data s_i on the data agent.

The next consideration is to understand how to generate $\mathcal{R}_j$. In fact, the data selection set for queries has a significant impact on the recovery performance of corrupted data. A well-crafted data selection set can achieve the condition that the original true data can be fully recovered with a minimum number of queries. Similarly, problems, such as the sensing matrix design in compressed sensing [36], parity check matrix design in error correction codes [37], and group matrix design in group testing problems [38,39], are analogous to the generation of a data selection set. According to [36], the role of matrix design holds paramount significance within the domain of compressed sensing, owing to its direct influence on the efficacy of signal recovery from sparse measurements. A well-crafted matrix design underpins the capability to achieve precise signal reconstruction utilizing a limited number of measurements, thereby optimizing the efficiency of data acquisition processes. Conversely, the selection of an inadequately structured matrix can considerably degrade the quality of signal reconstruction, compelling the requirement for an increased number of measurements, thereby negating the inherent advantages of compressed sensing in terms of efficiency and resource optimization. Let Q_{ji} be whether or not the i-th sparse data participate in the j-th query. To define our data selection set, we assume the following probability distribution for each Q_{ji} for whether the i-th data are included in the j-th query:

$$\Pr(Q_{ji} = \theta) = \begin{cases} 1 - \delta & \text{if } \theta = 0, \\ \delta & \text{if } \theta = 1, \end{cases} \tag{4}$$

where δ is the probability that the i-th data are included in the j-th query. A large δ means that a lot of data are included in the same query and the oracle is being asked to label it. Conversely, a small δ indicates that you are asking the oracle to label a query that consists of a small number of data.

For the corrupted labels, our query-testing system is represented by the following formula based on finite field operations:

$$\mathbf{y} = \mathbf{Q}\mathbf{s}, \tag{5}$$

where $\mathbf{Q} \in \{0, 1\}^{M \times N}$ is a query matrix with binary numbers of size $M \times N$, $\mathbf{s} \in \mathbb{F}_K^{N \times 1}$ is a sparse vector of finite size K and length N, and $\mathbf{y} \in \mathbb{F}_K^{M \times 1}$ is a response vector of finite size K and length M. And the addition and multiplication in (5) follow the operations defined in a finite field of size K. In this paper, we assume that K is either a prime or a power of a

prime number. So far, we defined the problem for the query-testing system considered in this work. We formulate it as the inverse problem of finding $\mathbf{s}$ given $\mathbf{Q}$ and $\mathbf{y}$ as in (5).

3.2. Decoding

In this paper, to find a true label vector $\mathbf{x}$, we consider using the MAP (maximum a posteriori) decoding method. The MAP method is a process of selecting a possible candidate random vector that maximizes the posterior probability. In this work, we find $\mathbf{x}$ such that the conditional probability is maximized by the following argument:

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} \Pr(\mathbf{x}|\mathbf{y}, \mathbf{Q}). \tag{6}$$

We rewrite (6) as follows:

$$\begin{aligned} \Pr(\mathbf{x}|\mathbf{y}, \mathbf{Q}) &= \frac{P(\mathbf{x}, \mathbf{y}, \mathbf{Q})}{P(\mathbf{y}, \mathbf{Q})} \\ &\propto P(\mathbf{x}, \mathbf{y}, \mathbf{Q}) \\ &= \sum_{\mathbf{s}} P(\mathbf{x}, \mathbf{y}, \mathbf{Q}, \mathbf{s}) \\ &= \sum_{\mathbf{s}} P(\mathbf{x}) P(\mathbf{s}|\mathbf{x}) P(\mathbf{Q}) P(\mathbf{y}|\mathbf{Q}, \mathbf{s}) \\ &\propto \sum_{\mathbf{s}} P(\mathbf{x}) P(\mathbf{s}|\mathbf{x}) \mathbb{1}_{\mathbf{y}=\mathbf{Q}\mathbf{s}}. \end{aligned} \tag{7}$$

The fourth line of (7) is due to an independent condition. In addition, the conditional probability $P(\mathbf{y}|\mathbf{Q}, \mathbf{s})$ is an indicator function $\mathbb{1}_{\mathbf{y}=\mathbf{Q}\mathbf{s}}$ that satisfies the following condition, $\mathbf{y} = \mathbf{Q}\mathbf{s}$:

$$P(\mathbf{y}|\mathbf{Q}, \mathbf{s}) = \begin{cases} 1 & \text{if } \mathbf{y} = \mathbf{Q}\mathbf{s}, \\ 0 & \text{if } \mathbf{y} \neq \mathbf{Q}\mathbf{s}. \end{cases} \tag{8}$$

For $\hat{\mathbf{x}}$ obtained from the MAP decoding in (6), we define an error event if it differs from realization $\mathbf{x}$. In this paper, the error probability P_E is for the error event when the result obtained from decoding differs from the original true value as $P_E = \Pr\{\hat{\mathbf{x}} \neq \mathbf{x}\}$.

3.3. Labeling Error

Our query-testing system is characterized by the ability to correct errors in the traditional labeling process. In traditional labeling, data are selected and labeled once per query, so the true value of the labeling is inverted according to the oracle's skill and expertise, and used as a dataset for supervised learning. The data are then used as a dataset for supervised learning without any further correction or modification of the data labeling. In the previous section, we used a probabilistic model to represent the errors that oracles make when labeling data. The next step is to define the probability distribution of the response vector of the query-testing system due to these labeling errors. Using (1) and (2), we lead to the following probability model for the element of the sparse vector:

$$P(s_i = \theta) = \sum_{\beta \in \mathbb{F}_K} P(x_i = \beta) P(c_i = \theta | x_i = \beta) = \begin{cases} 1 - q & \text{if } \theta = 0, \\ \frac{q}{K-1} & \text{if } \theta \neq 0, \end{cases} \tag{9}$$

where $1 - q = (1 - p)(1 - \gamma) + \frac{p\gamma}{K-1}$.

Next, based on the query matrix, the oracle selects a subset of the data to be included in a single query. This is performed by generating a subset of the data using the query matrix by (4) defined earlier. If there is only one instance of data in the subset, $|\mathcal{R}_j| = 1$, then the corresponding response result y_j is equal to the probability distribution in (9). Let us go a step further and see what happens to the response result y_j after the subset contains two data entries. For example, in our query-testing system, the probability of a response result involving the first and second data is $P(s_1 + s_2)$ for $s_1 \neq 0, s_2 \neq 0$ as $|\mathcal{R}_j| = 2$. In the same way, the probability of the response result involving the second and third data

is $P(s_2 + s_3)$ for $s_2 \neq 0, s_3 \neq 0$. At this point, we can see that the probability distributions of the two response results above are identical, $P(s_1 + s_2) = P(s_2 + s_3)$. The probability of one response result from the combination of the three data is also $P(s_1 + s_2 + s_3)$ for $s_1 \neq 0, s_2 \neq 0, s_3 \neq 0$, as $|\mathcal{R}_j| = 3$, and, eventually, the probabilities of all the possible response results from selecting three data from N data are the same.

As a more general case, consider a sparse vector $\mathbf{s}$ with the first d nonzero elements, $||\mathbf{s}||_0 = d$, and let P_d be the probability that $P_d := \Pr\left(\sum_{i=1}^{d} Q_{ji}s_i = 0\right)$, where $s_i \neq 0$ due to operations of a finite field. Given the probability distribution of Q_{ji} in (4), we can write the probability P_d in a recursive form as follows:

$$\begin{aligned} P_d &= \Pr\left(\sum_{i=1}^{d-1} Q_{ji}s_i = 0\right) \Pr\left(Q_{jd}s_d = 0\right) + \sum_{\theta \in \mathbb{F}_K \setminus \{0\}} \Pr\left\{\sum_{i=1}^{d-1} Q_{ji}s_i = \theta\right\} \Pr\left\{Q_{id}s_d = -\theta\right\} \\ &= P_{d-1}(1-\delta) + (1-P_{d-1})\frac{\delta}{K-1}. \end{aligned} \tag{10}$$

Define $H_d := P_d - K^{-1}$. The recursive expression is obtained as follows:

$$H_d = H_{d-1}\left(1 - \frac{\delta}{1-K^{-1}}\right). \tag{11}$$

Solving (11), we find the probability P_d

$$P_d = K^{-1} + \left(1-K^{-1}\right)\left(1 - \frac{\delta}{1-K^{-1}}\right)^d. \tag{12}$$

So far we defined the problem of the query-testing system we are considering. Next, we investigate from an information-theoretic point of view how many queries are needed to perfectly recover the true data $\mathbf{x}$ as a function of the percentage of imbalanced data. We also use the MAP decoding method to obtain an upper bound on the error probability that recovery will fail.

4. Bounds on Performance for Recovery of Corrupted Labels

4.1. Lower Bound

Now consider the minimum number of queries to require the recovery of a corrupted label vector in the finite query-testing scheme. To handle this, Fano's inequality theorem [40] is used, which was introduced by information theory. In this section, we exploit Fano's inequality theorem to derive a lower bound on the error probability for the recovery of the corrupted labels in our finite query-testing scheme.

Theorem 1. *For any recovery scheme, a true label vector defined in* (1)*, and transition probability in* (2)*, a lower bound on the error probability P_E such that*

$$P_E \geq \frac{H(\mathbf{s}) - M\log K - 1}{N\log K}, \tag{13}$$

where $H(\cdot)$ is the entropy function and the base of log *is 2.*

Proof of Theorem 1. Let $\hat{\mathbf{x}}$ be the estimated true label vector of $\mathbf{x}$ by any recovery scheme. Using Markov chain, we have $\mathbf{x} \to \mathbf{s} \to (\mathbf{Q}, \mathbf{y}) \to \hat{\mathbf{x}}$. From this processing, we consider the two following inequalities:

$$H(\mathbf{x}|\mathbf{s}) \leq H(\mathbf{x}|\hat{\mathbf{x}}), \tag{14}$$

and

$$H(\mathbf{s}|\mathbf{Q}, \mathbf{y}) \leq H(\mathbf{x}|\hat{\mathbf{x}}). \tag{15}$$

Using Fano's inequality, the following bound is obtained:

$$H(\mathbf{x}|\hat{\mathbf{x}}) \leq 1 + P_E \log\Big(K^N - 1\Big). \tag{16}$$

From (14) and (16), one lower bound on the error probability is

$$P_E \geq \frac{H(\mathbf{x}|\mathbf{s}) - 1}{N \log K}, \tag{17}$$

where once we know the corrupted labels of both entropies are the same, $\mathbf{s} = \mathbf{c} \oplus \mathbf{x}$, there is no randomness. Therefore, (17) is ignored with respect to finding the bound on the error probability. In addition, we have other bounds of P_E using (15) and (16) as follows:

$$P_E \geq \frac{H(\mathbf{s}|\mathbf{Q}, \mathbf{y}) - 1}{N \log K}. \tag{18}$$

The conditional entropy $H(\mathbf{s}|\mathbf{Q}, \mathbf{y})$ of (18) can be rewritten as

$$\begin{aligned} H(\mathbf{s}|\mathbf{Q}, \mathbf{y}) &= H(\mathbf{s}) - I(\mathbf{s}; \mathbf{Q}, \mathbf{y}) \\ &= H(\mathbf{s}) - (I(\mathbf{s}; \mathbf{Q}) + I(\mathbf{s}; \mathbf{y}|\mathbf{Q})) \\ &\overset{(a)}{=} H(\mathbf{s}) - (H(\mathbf{y}|\mathbf{Q}) - H(\mathbf{y}|\mathbf{Q}, \mathbf{s})) \\ &\overset{(b)}{\geq} H(\mathbf{s}) - M \log K, \end{aligned} \tag{19}$$

where $I(\cdot)$ denotes mutual information, and (a) is due to the independence of both terms $\mathbf{s}$ and $\mathbf{Q}$. The last line of (19) is that if we know $\mathbf{Q}, \mathbf{s}$, the randomness of $\mathbf{y}$ vanishes, so the conditional entropy $H(\mathbf{y}|\mathbf{Q}, \mathbf{s}) = 0$. And the following inequality is obtained:

$$H(\mathbf{y}|\mathbf{Q}) \leq H(\mathbf{y}) = M \log K. \tag{20}$$

Therefore, the lower bound on P_E is

$$P_E \geq \frac{H(\mathbf{s}) - M \log K - 1}{N \log K}. \tag{21}$$

This is complete for the proof of Theorem 1. □

Considering two bounds (18) and (21), we obtain the lower bound as if $H(\mathbf{x}) - H(\mathbf{s}|\mathbf{x}) \geq 0$, then (16) holds; otherwise, (21) is the lower bound on P_E. The two parameters p and γ from (1) and (2) decide the lower bound as shown by entropy $H(\mathbf{s})$ in Theorem 1. At the other extreme, we can see that the closer the two parameters p and γ are to zero, the more the error rate converges to zero.

4.2. Upper Bound

In this subsection, we consider an upper bound on the error probability for the MAP decoding in the finite query-testing scheme. For the proof of the upper bound, we divide into two parts the definition of the error event and formulate the probability of decoding the error. Note that $\mathbf{x}$ is the true labels, $\mathbf{s}$ is the corrupted labels, and the response results of their combinations are collected as $\mathbf{y} = \mathbf{Q}\mathbf{s}$. We rewrite the a posteriori probability

$$P(\mathbf{x}|\mathbf{Q}, \mathbf{y}) \propto \sum_{\mathbf{z}} P(\mathbf{x}) P(\mathbf{z}|\mathbf{x}) \mathbb{1}_{\mathbf{y}=\mathbf{Q}\mathbf{z}}, \tag{22}$$

where $\mathbf{z} \in \mathbb{F}_K^N$ is a dummy vector that satisfies the condition on $\mathbf{y} = \mathbf{Q}\mathbf{z}$. In this work, the decoder knows $\mathbf{Q}$ and $\mathbf{y}$. Using MAP decoding, we estimate the true labels from (22),

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} \sum_{\mathbf{z}} P(\mathbf{x}) P(\mathbf{z}|\mathbf{x}) \mathbb{1}_{\mathbf{y}=\mathbf{Q}\mathbf{z}}. \tag{23}$$

An error event occurs if there exists any feasible vector $\mathbf{x} \neq \bar{\mathbf{x}}$ such that

$$\sum_{\mathbf{z}} P(\mathbf{x}, \mathbf{z}) \mathbb{1}_{\mathbf{Qs}=\mathbf{Qz}} \leq \sum_{\mathbf{z}} P(\bar{\mathbf{x}}, \mathbf{z}) \mathbb{1}_{\mathbf{Qs}=\mathbf{Qz}}. \tag{24}$$

So far, we define the error event and now derive an upper bound on the error probability. With a given $\mathbf{x}$ and $\mathbf{s}$, let $P(\mathcal{E}|\mathbf{x}, \mathbf{s})$ be the conditional error probability. We have an average error probability as follows:

$$P_E = \sum_{\mathbf{x}} \sum_{\mathbf{s}} P(\mathbf{x}, \mathbf{s}) P(\mathcal{E}|\mathbf{x}, \mathbf{s}). \tag{25}$$

We use the two typical sets in [40]. For any $\epsilon > 0$, let $\mathcal{T}^N_{[\mathbf{x}]\epsilon}$ be the typical set for $\mathbf{x}$ with respect to $P(\mathbf{x})$ and $\mathcal{T}^N_{[\mathbf{s}|\mathbf{x}]\epsilon}$ be the conditional typical set for $\mathbf{s}$ distributed with $P(\mathbf{s}|\mathbf{x})$. For any sufficiently large number N, both typical sets are respectively defined as

$$\mathcal{T}^N_{[\mathbf{x}]\epsilon} = \left\{ \mathbf{x} \in \mathbb{F}_K^N : \left| -\frac{1}{N} \log P(\mathbf{x}) - H(\mathbf{x}) \right| \leq \epsilon \right\}, \tag{26}$$

and

$$\mathcal{T}^N_{[\mathbf{s}|\mathbf{x}]\epsilon} = \left\{ \mathbf{s} \in \mathbb{F}_K^N : \left| -\frac{1}{N} \log P(\mathbf{s}|\mathbf{x}) - H(\mathbf{s}|\mathbf{x}) \right| \leq \epsilon \right\}. \tag{27}$$

Note that $H(\mathbf{x}, \mathbf{s}) = H(\mathbf{x}) + H(\mathbf{s}|\mathbf{x})$ if $\mathbf{x} \in \mathcal{T}^N_{[\mathbf{x}]\epsilon}$ and $\mathbf{s} \in \mathcal{T}^N_{[\mathbf{s}|\mathbf{x}]\epsilon}$. We denote the joint typical set as $\mathcal{T}^N_{[\mathbf{x},\mathbf{s}]2\epsilon}$ and the set of pairs $(\mathbf{x}, \mathbf{s}) \in \mathbb{F}_K^N \times \mathbb{F}_K^N$ such that

$$\left| -\frac{1}{N} \log P(\mathbf{x}, \mathbf{s}) - H(\mathbf{x}, \mathbf{s}) \right| \leq 2\epsilon. \tag{28}$$

As a result, we have two bounds by using the Shannon–McMillan–Breiman theorem [40], $P\left(\mathbf{s} \in \mathcal{T}^N_{[\mathbf{s}|\mathbf{x}]\epsilon}\right) \geq 1 - \epsilon$ and $P\left((\mathbf{x}, \mathbf{s}) \in \mathcal{T}^N_{[\mathbf{x},\mathbf{s}]2\epsilon}\right) \geq 1 - 2\epsilon$. And the cardinality of the joint typical set is $\left|\mathcal{T}^N_{[\mathbf{x},\mathbf{s}]2\epsilon}\right| \leq 2^{N(H(\mathbf{x},\mathbf{s})+2\epsilon)}$. To handle the error probability, we decompose into two parts, and find bounds on the probability of decoding error as

$$\begin{aligned} P_E &= \sum_{(\mathbf{x},\mathbf{s}) \in \mathcal{T}^N_{[\mathbf{x},\mathbf{s}]2\epsilon}} P(\mathbf{x}, \mathbf{s}) P(\mathcal{E}|\mathbf{x}, \mathbf{s}) + \sum_{(\mathbf{x},\mathbf{s}) \notin \mathcal{T}^N_{[\mathbf{x},\mathbf{s}]2\epsilon}} P(\mathbf{x}, \mathbf{s}) P(\mathcal{E}|\mathbf{x}, \mathbf{s}) \\ &\overset{(a)}{\leq} \sum_{(\mathbf{x},\mathbf{s}) \in \mathcal{T}^N_{[\mathbf{x},\mathbf{s}]2\epsilon}} P(\mathbf{x}, \mathbf{s}) P(\mathcal{E}|\mathbf{x}, \mathbf{s}) + \sum_{(\mathbf{x},\mathbf{s}) \notin \mathcal{T}^N_{[\mathbf{x},\mathbf{s}]2\epsilon}} P(\mathbf{x}, \mathbf{s}) \\ &\overset{(b)}{\leq} \sum_{(\mathbf{x},\mathbf{s}) \in \mathcal{T}^N_{[\mathbf{x},\mathbf{s}]2\epsilon}} P(\mathbf{x}, \mathbf{s}) P(\mathcal{E}|\mathbf{x}, \mathbf{s}) + 2\epsilon, \end{aligned} \tag{29}$$

where (a) is due to $P(\mathcal{E}|\mathbf{x}, \mathbf{s}) \leq 1$ and (b) comes from the definition of the joint typical set. For all the average query matrices, we consider overall query matrices in (29),

$$P_E \leq \sum_{\mathbf{Q}} P(\mathbf{Q}) \sum_{\mathbf{x} \in \mathcal{T}^N_{[\mathbf{x}]\epsilon}} \sum_{\mathbf{s} \in \mathcal{T}^N_{[\mathbf{s}|\mathbf{x}]\epsilon}} P(\mathbf{x}, \mathbf{s}) P(\mathcal{E}|\mathbf{x}, \mathbf{s}, \mathbf{Q}) + 2\epsilon. \tag{30}$$

where the conditional error probability of (30) is an indicator function as

$$P(\mathcal{E}|\mathbf{x}, \mathbf{s}, \mathbf{Q}) = \begin{cases} 1 & \text{if (29) holds,} \\ 0 & \text{otherwise.} \end{cases} \tag{31}$$

Using a part of Gallager's error exponents [41], the conditional error probability is bounded by

$$P(\mathcal{E}|\mathbf{x},\mathbf{s},\mathbf{Q}) \leq \sum_{\bar{\mathbf{x}}\in\mathbb{F}_K^N\setminus\mathbf{x}} \frac{\sum_{\mathbf{z_1}\in\mathbb{F}_K^N} P(\bar{\mathbf{x}},\mathbf{z_1})\mathbb{1}_{\mathbf{Qs}=\mathbf{Qz_1}}}{\sum_{\mathbf{z_2}\in\mathbb{F}_K^N} P(\mathbf{x},\mathbf{z_2})\mathbb{1}_{\mathbf{Qs}=\mathbf{Qz_2}}}.$$

Then, (30) can be rewritten by

$$P_E \leq \sum_{\substack{\mathbf{Q}\\ \mathbf{x}\in\mathcal{T}_{[\mathbf{x}]\epsilon}^N}} P(\mathbf{Q}) \sum_{\mathbf{s}\in\mathcal{T}_{[\mathbf{s}|\mathbf{x}]\epsilon}^N} P(\mathbf{x},\mathbf{s}) \sum_{\bar{\mathbf{x}}\in\mathbb{F}_K^N\setminus\mathbf{x}} \frac{\sum_{\mathbf{z_1}\in\mathbb{F}_K^N} P(\bar{\mathbf{x}},\mathbf{z_1})\mathbb{1}_{\mathbf{Qs}=\mathbf{Qz_1}}}{\sum_{\mathbf{z_2}\in\mathbb{F}_K^N} P(\mathbf{x},\mathbf{z_2})\mathbb{1}_{\mathbf{Qs}=\mathbf{Qz_2}}} + 2\epsilon. \tag{32}$$

Considering the right-hand side of (30) for a sufficiently large N, we have the following bound:

$$\frac{\sum_{\mathbf{s}\in\mathcal{T}_{[\mathbf{s}|\mathbf{x}]\epsilon}^N} P(\mathbf{x},\mathbf{s})\mathbb{1}_{\mathbf{y}=\mathbf{Qs}}}{\sum_{\mathbf{z_2}\in\mathbb{F}_K^N} P(\mathbf{x},\mathbf{z_2})\mathbb{1}_{\mathbf{y}=\mathbf{Qz_2}}} \leq 1. \tag{33}$$

Using the bound (33), (32) is further bounded as

$$\begin{aligned} P_E &\leq \sum_{\substack{\mathbf{Q}\\ \mathbf{x}\in\mathcal{T}_{[\mathbf{x}]\epsilon}^N}} P(\mathbf{Q}) \sum_{\mathbf{s}\in\mathcal{T}_{[\mathbf{s}|\mathbf{x}]\epsilon}^N} \sum_{\substack{\bar{\mathbf{x}}\in\mathbb{F}_K^N\setminus\mathbf{x}\\ \mathbf{z_1}\in\mathbb{F}_K^N}} P(\bar{\mathbf{x}},\mathbf{z_1})\mathbb{1}_{\mathbf{Qs}=\mathbf{Qz_1}} + 2\epsilon \\ &\leq \sum_{\substack{\mathbf{x}\in\mathcal{T}_{[\mathbf{x}]\epsilon}^N\\ \bar{\mathbf{x}}\in\mathbb{F}_K^N\setminus\mathbf{x}}} \sum_{\substack{\mathbf{s}\in\mathcal{T}_{[\mathbf{s}|\mathbf{x}]\epsilon}^N\\ \mathbf{z_1}\in\mathbb{F}_K^N}} P(\bar{\mathbf{x}},\mathbf{z_1}) \sum_{\mathbf{Q}} P(\mathbf{Q})\mathbb{1}_{\mathbf{Qs}=\mathbf{Qz_1}} + 2\epsilon, \end{aligned} \tag{34}$$

where $\sum_{\mathbf{Q}} P(\mathbf{Q})\mathbb{1}_{\mathbf{Qs}=\mathbf{Qz_1}} = P(\mathbf{Qs}=\mathbf{Qz_1})$. We can rewrite (34) as

$$P_E \leq \sum_{\substack{\mathbf{x}\in\mathcal{T}_{[\mathbf{x}]\epsilon}^N\\ \bar{\mathbf{x}}\in\mathbb{F}_K^N\setminus\mathbf{x}}} \sum_{\substack{\mathbf{s}\in\mathcal{T}_{[\mathbf{s}|\mathbf{x}]\epsilon}^N\\ \mathbf{z_1}\in\mathbb{F}_K^N}} P(\bar{\mathbf{x}},\mathbf{z_1})P(\mathbf{Qs}=\mathbf{Qz_1}) + 2\epsilon. \tag{35}$$

We denote $\|\mathbf{s}-\mathbf{z_1}\|_0 = d$. For example, $d=0$, then $P(\mathbf{Qs}=\mathbf{Qz_1})=1$. We decompose (35) into two parts when $d=0$ or not:

$$P_E \leq P_{E1} + P_{E2} + 2\epsilon, \tag{36}$$

where

$$P_{E1} := \sum_{\substack{\mathbf{x}\in\mathcal{T}_{[\mathbf{x}]\epsilon}^N\\ \bar{\mathbf{x}}\in\mathbb{F}_K^N\setminus\mathbf{x}}} \sum_{\mathbf{z_1}\in\mathbb{F}_K^N} P(\bar{\mathbf{x}},\mathbf{z_1}),$$

and

$$P_{E2} := \sum_{\substack{\mathbf{x}\in\mathcal{T}_{[\mathbf{x}]\epsilon}^N\\ \bar{\mathbf{x}}\in\mathbb{F}_K^N\setminus\mathbf{x}}} \sum_{\substack{\mathbf{s}\in\mathcal{T}_{[\mathbf{s}|\mathbf{x}]\epsilon}^N\\ \mathbf{z_1}\in\mathbb{F}_K^N\setminus\mathbf{s}}} P(\bar{\mathbf{x}},\mathbf{z_1})P(\mathbf{Qs}=\mathbf{Qz_1}).$$

Theorem 2. *For our MAP decoding, a sufficiently large number N, vanishing the error probability P_E, the upper bound on the number of queries M is greater than*

$$\frac{M}{N} > \frac{H(\mathbf{x},\mathbf{s})+2\epsilon}{\log\left(K^{-1}+(1-K^{-1})\left(1-\frac{\delta}{1-K^{-1}}\right)^{\lceil\alpha N\rceil}\right)^{-1}}, \tag{37}$$

where $0<\alpha<0.5$.

4.3. Proof of Theorem 2

For P_{E1}, we have the following under $\mathbf{x} \neq \bar{\mathbf{x}}$:

$$P_{E1} = \sum_{\bar{\mathbf{x}} \in \mathbb{F}_K^N \setminus \mathbf{x}} P(\bar{\mathbf{x}}) \sum_{\substack{\mathbf{x} \in \mathcal{T}_{[\mathbf{x}]\epsilon}^N \\ \mathbf{z_1} \in \mathbb{F}_K^N}} P(\mathbf{z_1}|\bar{\mathbf{x}}), \tag{38}$$

where (38) can be divided into two parts as $P_{E1} = P_{E11} + P_{E12}$ by the typical set $\mathcal{T}_{[\mathbf{x}]\epsilon}^N$. Due to the assumption $\mathbf{x} \neq \bar{\mathbf{x}}$, we have

$$\begin{aligned} P_{E11} &= \sum_{\bar{\mathbf{x}} \in \mathbb{F}_K^N} P(\bar{\mathbf{x}}) \sum_{\substack{\mathbf{x} \in \mathcal{T}_{[\mathbf{x}]\epsilon}^N \setminus \bar{\mathbf{x}} \\ \mathbf{z_1} \in \mathbb{F}_K^N}} P(\mathbf{z_1}|\bar{\mathbf{x}}) \\ &\leq \sum_{\bar{\mathbf{x}} \in \mathbb{F}_K^N} P(\bar{\mathbf{x}}) \sum_{\mathbf{z_1} \in \mathbb{F}_K^N \setminus \mathcal{T}_{[\mathbf{x}]\epsilon}^N} P(\mathbf{z_1}|\bar{\mathbf{x}}) \\ &\leq \epsilon, \end{aligned}$$

and

$$\begin{aligned} P_{E12} &= \sum_{\bar{\mathbf{x}} \in \mathbb{F}_K^N \setminus \mathcal{T}_{[\mathbf{x}]\epsilon}^N} P(\bar{\mathbf{x}}) \sum_{\substack{\mathbf{x} \in \mathcal{T}_{[\mathbf{x}]\epsilon}^N \\ \mathbf{z_1} \in \mathbb{F}_K^N}} P(\mathbf{z_1}|\bar{\mathbf{x}}) \\ &\leq \sum_{\bar{\mathbf{x}} \in \mathbb{F}_K^N \setminus \mathcal{T}_{[\mathbf{x}]\epsilon}^N} P(\bar{\mathbf{x}}) \\ &\leq \epsilon. \end{aligned}$$

Then, we consider the second part P_{E2} and enumerate the same probability with the Hamming weight, $\|\mathbf{s} - \mathbf{z_1}\|_0 = d$:

$$P_{E2} = \sum_{d=1}^{N} \sum_{\substack{\mathbf{x} \in \mathcal{T}_{[\mathbf{x}]\epsilon}^N \\ \bar{\mathbf{x}} \in \mathbb{F}_K^N \setminus \mathbf{x}\, \mathbf{z_1} \in \mathbb{F}_K^N : \|\mathbf{s} - \mathbf{z_1}\|_0 = d}} \sum_{\mathbf{s} \in \mathcal{T}_{[\mathbf{s}|\mathbf{x}]\epsilon}^N} P\Big(\bar{\mathbf{x}}, \mathbf{z_1}\Big) P\Big(\mathbf{Qs} = \mathbf{Qz_1}\Big| \|\mathbf{s} - \mathbf{z_1}\|_0 = d\Big). \tag{39}$$

Note that $P\big(\mathbf{Qs} = \mathbf{Qz_1}\big|\|\mathbf{s} - \mathbf{z_1}\|_0 = d\big)$ comes from (12):

$$\begin{aligned} P\Big(\mathbf{Qs} = \mathbf{Qz_1}\Big| \|\mathbf{s} - \mathbf{z_1}\|_0 = d\Big)^M &= \left(K^{-1} + \left(1 - K^{-1}\right)\left(1 - \frac{\delta}{1 - K^{-1}}\right)^d\right)^M \\ &\leq \left(K^{-1} + \left(1 - K^{-1}\right)\left(1 - \frac{\delta}{1 - K^{-1}}\right)\right)^M \\ &= \left(1 - \delta\right)^M, \end{aligned} \tag{40}$$

where the second inequality is due to $P\big(\mathbf{Qs} = \mathbf{Qz_1}\big|\|\mathbf{s} - \mathbf{z_1}\|_0 = d\big)$ being a monotonically decreasing function as d increases.

Next, for $0 < \alpha < 0.5$, (39) can be decomposed into two probabilities as follows $P_{E2} = P_{E21} + P_{E22}$:

$$P_{E21} = \sum_{d=1}^{\lfloor \alpha N \rfloor} \sum_{\substack{\mathbf{x} \in \mathcal{T}_{[\mathbf{x}]\epsilon}^N \\ \bar{\mathbf{x}} \in \mathbb{F}_K^N \setminus \mathbf{x}\, \mathbf{z_1} \in \mathbb{F}_K^N : \|\mathbf{s} - \mathbf{z_1}\|_0 = d}} \sum_{\mathbf{s} \in \mathcal{T}_{[\mathbf{s}|\mathbf{x}]\epsilon}^N} P\Big(\bar{\mathbf{x}}, \mathbf{z_1}\Big) P\Big(\mathbf{Qs} = \mathbf{Qz_1}\Big| \|\mathbf{s} - \mathbf{z_1}\|_0 = d\Big), \tag{41}$$

and

$$P_{E22} = \sum_{d=\lceil \alpha N \rceil}^{N} \sum_{\substack{\mathbf{x}\in\mathcal{T}^N_{[\mathbf{x}]\epsilon} \\ \bar{\mathbf{x}}\in\mathbb{F}_K^N\setminus\mathbf{x}\ \mathbf{z_1}\in\mathbb{F}_K^N:\|\mathbf{s}-\mathbf{z_1}\|_0=d}} \sum_{\mathbf{s}\in\mathcal{T}^N_{[\mathbf{s}|\mathbf{x}]\epsilon}} P\Big(\bar{\mathbf{x}},\mathbf{z_1}\Big) P\Big(\mathbf{Qs}=\mathbf{Qz_1}\Big|\|\mathbf{s}-\mathbf{z_1}\|_0=d\Big). \tag{42}$$

First, we consider the probability P_{E21},

$$\begin{aligned} P_{E21} &= \sum_{d=1}^{\lfloor \alpha N \rfloor} \sum_{\substack{\mathbf{x}\in\mathcal{T}^N_{[\mathbf{x}]\epsilon} \\ \bar{\mathbf{x}}\in\mathbb{F}_K^N\setminus\mathbf{x}\ \mathbf{z_1}\in\mathbb{F}_K^N:\|\mathbf{s}-\mathbf{z_1}\|_0=d}} \sum_{\mathbf{s}\in\mathcal{T}^N_{[\mathbf{s}|\mathbf{x}]\epsilon}} P\Big(\bar{\mathbf{x}},\mathbf{z_1}\Big) P\Big(\mathbf{Qs}=\mathbf{Qz_1}\Big|\|\mathbf{s}-\mathbf{z_1}\|_0=d\Big) \\ &\le \sum_{d=1}^{\lfloor \alpha N \rfloor} \sum_{\substack{\mathbf{x}\in\mathcal{T}^N_{[\mathbf{x}]\epsilon} \\ \bar{\mathbf{x}}\in\mathbb{F}_K^N\setminus\mathbf{x}\ \mathbf{z_1}\in\mathbb{F}_K^N:\|\mathbf{s}-\mathbf{z_1}\|_0=d}} \sum_{\mathbf{s}\in\mathcal{T}^N_{[\mathbf{s}|\mathbf{x}]\epsilon}} P\Big(\bar{\mathbf{x}},\mathbf{z_1}\Big) P\Big(\mathbf{Qs}=\mathbf{Qz_1}\Big|\|\mathbf{s}-\mathbf{z_1}\|_0=1\Big) \\ &= \Big(1-\delta\Big)^M \sum_{d=1}^{\lfloor \alpha N \rfloor} \sum_{\substack{\mathbf{x}\in\mathcal{T}^N_{[\mathbf{x}]\epsilon} \\ \bar{\mathbf{x}}\in\mathbb{F}_K^N\setminus\mathbf{x}\ \mathbf{z_1}\in\mathbb{F}_K^N:\|\mathbf{s}-\mathbf{z_1}\|_0=d}} \sum_{\mathbf{s}\in\mathcal{T}^N_{[\mathbf{s}|\mathbf{x}]\epsilon}} P\Big(\bar{\mathbf{x}},\mathbf{z_1}\Big). \end{aligned} \tag{43}$$

In (43), we change the order of summation and take the result in (40). And we further bound

$$\begin{aligned} P_{E21} &\le \Big(1-\delta\Big)^M \sum_{d=1}^{\lfloor \alpha N \rfloor} \sum_{\substack{\mathbf{x}\in\mathcal{T}^N_{[\mathbf{x}]\epsilon} \\ \bar{\mathbf{x}}\in\mathbb{F}_K^N\setminus\mathbf{x}}} \sum_{\mathbf{s}\in\mathcal{T}^N_{[\mathbf{s}|\mathbf{x}]\epsilon}} P\Big(\bar{\mathbf{x}},\mathbf{z_1}\Big) \sum_{\mathbf{z_1}\in\mathbb{F}_K^N:\|\mathbf{s}-\mathbf{z_1}\|_0=d} \mathbb{1} \\ &\le \Big(1-\delta\Big)^M \sum_{d=1}^{\lfloor \alpha N \rfloor} \sum_{\substack{\mathbf{x}\in\mathcal{T}^N_{[\mathbf{x}]\epsilon} \\ \mathbf{s}\in\mathcal{T}^N_{[\mathbf{s}|\mathbf{x}]\epsilon}}} \sum_{\bar{\mathbf{x}}\in\mathbb{F}_K^N\setminus\mathbf{x}} P\Big(\bar{\mathbf{x}}\Big) \sum_{\mathbf{z_1}\in\mathbb{F}_K^N:\|\mathbf{s}-\mathbf{z_1}\|_0=d} \mathbb{1} \\ &\le \Big(1-\delta\Big)^M \sum_{d=1}^{\lfloor \alpha N \rfloor} \sum_{\substack{\mathbf{x}\in\mathcal{T}^N_{[\mathbf{x}]\epsilon} \\ \mathbf{s}\in\mathcal{T}^N_{[\mathbf{s}|\mathbf{x}]\epsilon}}} \sum_{\bar{\mathbf{x}}\in\mathbb{F}_K^N\setminus\mathbf{x}} P\Big(\bar{\mathbf{x}}\Big) \Big|\Big\{\mathbf{z_1}\in\mathbb{F}_K^N:\|\mathbf{s}-\mathbf{z_1}\|_0=d\Big\}\Big|, \end{aligned} \tag{44}$$

where $\mathbb{1}$ denotes the indicator function. For the cardinality of the conditional set in the last line of (44), we find its bound as,

$$\begin{aligned} \Big|\Big\{\mathbf{z_1}\in\mathbb{F}_K^N:\|\mathbf{s}-\mathbf{z_1}\|_0=d\Big\}\Big| &= \binom{N}{d}(K-1)^d \\ &\le 2^{NH(d/N)}(K-1)^d. \end{aligned}$$

Because the entropy $H(d/N)$ is a monotonically increasing function up to $d/N < 1/2$ and the size of the typical set comes from [40], we obtain the exponent terms as follows:

$$\begin{aligned} P_{E21} &\le \Big(1-\delta\Big)^M \sum_{d=1}^{\lfloor \alpha N \rfloor} \sum_{\substack{\mathbf{x}\in\mathcal{T}^N_{[\mathbf{x}]\epsilon} \\ \mathbf{s}\in\mathcal{T}^N_{[\mathbf{s}|\mathbf{x}]\epsilon}}} \sum_{\bar{\mathbf{x}}\in\mathbb{F}_K^N\setminus\mathbf{x}} P\Big(\bar{\mathbf{x}}\Big) \Big|\Big\{\mathbf{z_1}\in\mathbb{F}_K^N:\|\mathbf{s}-\mathbf{z_1}\|_0=d\Big\}\Big| \\ &\le \Big(1-\delta\Big)^M \sum_{\substack{\mathbf{x}\in\mathcal{T}^N_{[\mathbf{x}]\epsilon} \\ \mathbf{s}\in\mathcal{T}^N_{[\mathbf{s}|\mathbf{x}]\epsilon}}} \sum_{d=1}^{\lfloor \alpha N \rfloor} 2^{NH(d/N)(K-1)^d} \\ &\le \Big(1-\delta\Big)^M 2^{N(H(\mathbf{x},\mathbf{s})+2\epsilon)} \alpha N 2^{NH(\alpha)} \Big(K-1\Big)^{\alpha N}. \end{aligned} \tag{45}$$

The exponent term of the last line of (45) is

$$-N\left(-\frac{M}{N}\log(1-\delta)-H(\mathbf{x},\mathbf{s})-2\epsilon-H(\alpha)-\frac{\log(\alpha N)}{N}-\alpha\log(K-1)\right). \tag{46}$$

In order to vanish the error probability P_{E21} such that the inner term of (46) is greater than 0, taking $N\to\infty$, the following exponent has the bound of M/N:

$$\frac{M}{N}>\frac{H(\mathbf{x},\mathbf{s})+H(\alpha)+\alpha\log(K-1)+\log(\alpha N)/N+2\epsilon}{\log(1-\delta)^{-1}}. \tag{47}$$

Next, for P_{E22}, we find the bound using the same way for P_{E21}:

$$\begin{aligned}
P_{E22} &= \sum_{d=\lceil\alpha N\rceil}^{N}\sum_{\substack{\mathbf{x}\in\mathcal{T}^N_{[\mathbf{x}]\epsilon}\\ \mathbf{s}\in\mathcal{T}^N_{[\mathbf{s}|\mathbf{x}]\epsilon}}}\sum_{\substack{\bar{\mathbf{x}}\in\mathbb{F}_K^N\setminus\mathbf{x}\\ \mathbf{z_1}\in\mathbb{F}_K^N:\|\mathbf{s}-\mathbf{z_1}\|_0=d}} P\left(\bar{\mathbf{x}},\mathbf{z_1}\right)P\left(\mathbf{Qs}=\mathbf{Qz_1}\middle|\|\mathbf{s}-\mathbf{z_1}\|_0=d\right)\\
&\le \sum_{d=\lceil\alpha N\rceil}^{N}\sum_{\substack{\mathbf{x}\in\mathcal{T}^N_{[\mathbf{x}]\epsilon}\\ \mathbf{s}\in\mathcal{T}^N_{[\mathbf{s}|\mathbf{x}]\epsilon}}}\sum_{\substack{\bar{\mathbf{x}}\in\mathbb{F}_K^N\setminus\mathbf{x}\\ \mathbf{z_1}\in\mathbb{F}_K^N:\|\mathbf{s}-\mathbf{z_1}\|_0=d}} P\left(\bar{\mathbf{x}},\mathbf{z_1}\right)P\left(\mathbf{Qs}=\mathbf{Qz_1}\middle|\|\mathbf{s}-\mathbf{z_1}\|_0=\lceil\alpha N\rceil\right)\\
&\le \sum_{\substack{\mathbf{x}\in\mathcal{T}^N_{[\mathbf{x}]\epsilon}\\ \mathbf{s}\in\mathcal{T}^N_{[\mathbf{s}|\mathbf{x}]\epsilon}}}\left(\sum_{\bar{\mathbf{x}}\in\mathbb{F}_K^N\setminus\mathbf{x}}P\left(\bar{\mathbf{x}}\right)\right)\left(\sum_{\mathbf{z_1}\in\mathbb{F}_K^N}P\left(\mathbf{z_1}\right)\right)P\left(\mathbf{Qs}=\mathbf{Qz_1}\middle|\|\mathbf{s}-\mathbf{z_1}\|_0=\lceil\alpha N\rceil\right)\\
&\le \sum_{\substack{\mathbf{x}\in\mathcal{T}^N_{[\mathbf{x}]\epsilon}\\ \mathbf{s}\in\mathcal{T}^N_{[\mathbf{s}|\mathbf{x}]\epsilon}}}P\left(\mathbf{Qs}=\mathbf{Qz_1}\middle|\|\mathbf{s}-\mathbf{z_1}\|_0=\lceil\alpha N\rceil\right)\\
&\le 2^{N(H(\mathbf{x},\mathbf{s})+2\epsilon)}\left(K^{-1}+\left(1-K^{-1}\right)\left(1-\frac{\delta}{1-K^{-1}}\right)^{\lceil\alpha N\rceil}\right)^{M},
\end{aligned} \tag{48}$$

where $\|\mathbf{s}-\mathbf{z_1}\|_0=d$ is ignored and the bound of the typical set is used. Finally, we find the exponent of the right-hand side of (48). For $N\to\infty$, vanishing the error probability, the exponent of (48) is bounded as

$$\frac{M}{N}>\frac{H(\mathbf{x},\mathbf{s})+2\epsilon}{\log\left(K^{-1}+(1-K^{-1})\left(1-\frac{\delta}{1-K^{-1}}\right)^{\lceil\alpha N\rceil}\right)^{-1}}. \tag{49}$$

4.4. Discussion

In this subsection, we discuss our findings from several perspectives. The first of these is that the upper bound on the error probability obtained by the MAP decoding is as shown in (37), where P_{E1} is negligible if ϵ is trivially small. On the other hand, P_{E2} is expressed as the sum of two probabilities, $P_{E2}=P_{E21}+P_{E22}$. Let us see which of these is the more significant factor. From (40), the denominator of the right-hand side of (49) satisfies the following inequality:

$$\log(1-\delta)^{-1}\le\log\left(K^{-1}+\left(1-K^{-1}\right)\left(1-\frac{\delta}{1-K^{-1}}\right)^{\lceil\alpha N\rceil}\right)^{-1}$$

So we conclude an upper bound on the error probability as in Theorem 2.

Theorems 1 and 2 state the lower and upper bounds on the error probability for recovering corrupted data, respectively. Combining the two conditions, we see that the upper bound should be larger than the lower bound. In other words, the range for α

mentioned in Theorem 2 is determined from two conditions. Using the vanishing of the error probability as $N \to \infty$, the ratio of the number of queries in M/N for Theorem 1 to hold is simply bounded as follows:

$$\frac{M}{N} > \frac{H(\mathbf{s})}{\log K} \tag{50}$$

From (37) and (50), we see that the upper bound requires a large number of queries to recover corrupted data. This leads to the conclusion that the upper bound is expressed as a function of the joint entropy $H(\mathbf{x}, \mathbf{s})$. As mentioned earlier, the joint entropy is represented as the conditional entropy $H(\mathbf{s}|\mathbf{x})$ and the entropy $H(\mathbf{x})$ that both the transition and generation of the true label vector are a form of compression. In addition, consider the case of $K = 2$ as a binary classification, where our lower bound is a necessary condition for the number of queries that matches the result of Theorem 5 in [27].

5. Conclusions

We considered the problem of recovering corrupted labels in supervised learning. However, a subset of the data samples was replaced with incorrect labels due to the non-expert's lack of skill and experience. In this paper, we proposed the finite query-testing scheme. Then, we aimed to recover corrupted data samples with additional expert labeling tasks. Our testing system worked on identifying corrupted data samples with minimal expert queries and recovering them to their true label values. In this work, we considered a probabilistic model for generating queries based on combinations on finite fields. Our finite query-testing scheme utilized queries that combined random subsets of data samples. Our goal was to recover corrupted data samples with a minimum number of queries. To this end, we obtained lower bounds using Fano's inequality. In addition, we obtained upper bounds on the probability of error using the MAP decoding method. We showed the number of queries required for recovery, not only for binary classification but also for corrupted data in multiple classification problems. In this work, we were not able to verify the applicability of our scheme in an unsupervised learning environment, but we suggest further research to extend it to semi-supervised and unsupervised in future studies, where our proposed finite query-testing scheme may work.

Funding: This paper was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (NRF-2020R1I1A3071739).

Data Availability Statement: Not applicable.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet Classification with Deep Convolutional Neural Networks. In Proceedings of the Neural Information Processing System (NIPS), Harrahs and Harveys, Lake Tahoe, NV, USA, 3–8 December 2012; Volume 2, pp. 1097–1105.
2. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
3. Zhang, W.; Du, T.; Wang, J. Deep Learning over Multi-Field Categorical Data. In *European Conference on Information Retrieval*; Springer: Cham, Switzerland, 2016; pp. 45–57.
4. Chen, M.; Zhou, X. DeepRank: Learning to rank with neural networks for recommendation. *Knowl. Based Syst.* **2020**, *209*, 106478. [CrossRef]
5. Onal, K.D.; Zhang, Y.; Altingovde, I.S.; Rahman, M.M.; Karagoz, P.; Braylan, A.; Dang, B.; Chang, H.-L.; Kim, H.; McNamara, Q.; et al. Neural information retrieval: At the end of the early years. *Inf. Retr. J.* **2018**, *21*, 111–182. [CrossRef]
6. Howard, J.; Ruder, S. Universal Language Model Fine-tuning for Text Classification. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia, 15–20 July 2018; Volume 1, pp. 328–339. [CrossRef]
7. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2018**, arXiv:1810.04805.

8. Severyn, A.; Moschitti, A. Twitter sentiment analysis with deep convolutional neural networks. In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, 9–13 August 2015; pp. 959–962. [CrossRef]
9. Paolacci, G.; Chandler, J.; Ipeirotis, P.G. Running experiments on Amazon Mechanical Turk. *Judgm. Decis. Mak.* **2010**, *5*, 411–419. [CrossRef]
10. Cothey, V. Web-crawling reliability. *J. Am. Soc. Inf. Sci. Technol.* **2004**, *14*, 1228–1238. [CrossRef]
11. Mason, W.; Suri, S. Conducting behavioral research on Amazon's mechanical turk. *Behav. Res. Methods* **2012**, *44*, 1–23. [CrossRef]
12. Scott, C.; Blanchard, G.; Handy, G. Classification with asymmetric label noise: Consistency and maximal denoising. In Proceedings of the 26th Annual Conference on Learning Theory, Princeton, NJ, USA, 12–14 June 2013; Volume 30, pp. 489–511.
13. Frenay, B.; Verleysen, M. Classification in the presence of label Noise: A survey. *IEEE Trans. Neural Netw. Learn. Syst.* **2013**, *25*, 845–869. [CrossRef]
14. Lloyd, R.V.; Erickson, L.A.; Casey, M.B.; Lam, K.Y.; Lohse, C.M.; Asa, S.L.; Chan, J.K.; DeLellis, R.A.; Harach, H.R.; Kakudo, K.; et al. Observer variation in the diagnosis of follicular variant of papillary thyroid carcinoma. *Am. J. Surg. Pathol.* **2004**, *28*, 1336–1340. [CrossRef]
15. Xiao, H.; Xiao, H.; Eckert, C. Adversarial Label Flips Attack on Support Vector Machines. In Proceedings of the ECAI, Montpellier, France, 27–31 August 2012; pp. 870–875.
16. Tong, X.; Tian, X.; Yi, Y.; Chang, H.; Wang, X. Learning from massive noisy labeled data for image classification. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015. [CrossRef]
17. Li, W.; Wang, L.; Li, W.; Agustsson, E.; Gool, L.V. WebVision Database: Visual Learning and Understanding from Web Data. *arXiv* **2017**, arXiv:1708.02862.
18. Lee, K.H.; He, X.; Zhang, L.; Yang, L. CleanNet: Transfer Learning for Scalable Image Classifier Training with Label Noise. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–21 June 2018. [CrossRef]
19. Song, H.; Kim, M.; Lee, J.G. SELFIE: Refurbishing unclean samples for robust deep learning. In Proceedings of the 36th International Conference on Machine Learning (ICML), Long Beach, CA, USA, 10–15 June 2019; pp. 5907–5915.
20. Krause, J.; Sapp, B.; Howard, A.; Zhou, H.; Toshev, A.; Duerig, T.; Philbin, J.; Fei-Fei, L. The Unreasonable Effectiveness of Noisy Data for Fine-Grained Recognition. In Proceedings of the 14th European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Volume 9907, pp. 301–320. [CrossRef]
21. Arpit, D.; Jastrzebski, S.; Ballas, N.; Krueger, D.; Bengio, E.; Kanwal, M.S.; Maharaj, T.; Fischer, A.; Courville, A.; Bengio, Y.; et al. A closer look at memorization in deep networks. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 7–9 August 2017.
22. Song, H.; Kim, M.; Park, D.; Shin, Y.; Lee, J.-G. Learning from noisy labels with Deep Neural Networks: A survey. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, 1–19. *early access*. [CrossRef]
23. Ashtiani, H.; Kushagra, S.; Ben-David, S. Clustering with Same-Cluster Queries. In *Advances in Neural Information Processing Systems*; Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2016; Volume 29, pp. 3216–3224.
24. Firmani, D.; Saha, B.; Srivastava, D. Online entity resolution using an Oracle. *Proc. VLDB Endow.* **2016**, *9*, 384–395. [CrossRef]
25. Mazumdar, A.; Saha, B. Clustering with Noisy Queries. In *Advances in Neural Information Processing Systems*; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2017; Volume 30, pp. 5788–5799.
26. Wang, J.; Kraska, T.; Franklin, M.J.; Feng, J. CrowdER: Crowdsourcing Entity Resolution. *Proc. VLDB Endow.* **2012**, *5*, 1483–1494. [CrossRef]
27. Mazumdar, A.; Pal, S. Semisupervised Clustering by Queries and Locally Encodable Source Coding. *IEEE Trans. Inf. Theory* **2021**, *67*, 1141–1155. [CrossRef]
28. Kim, D.; Chung, H.W. Binary Classification with XOR Queries: Fundamental Limits and an Efficient Algorithm. *IEEE Trans. Inf. Theory* **2021**, *67*, 4588–4612. [CrossRef]
29. Haanpaa, H.; Jarvisalo, M.; Kaski, P.; Niemela, I. Hard satisfiable clause sets for benchmarking equivalence reasoning techniques. *J. Satisf. Boolean Model. Comput.* **2006**, *2*, 27–46. [CrossRef]
30. Abbe, E.; Bandeira, A.S.; Bracher, A.; Singer, A. Decoding binary node labels from censored edge measurements: Phase transition and efficient recovery. *IEEE Trans. Netw. Sci. Eng.* **2014**, *1*, 10–22. [CrossRef]
31. Ahn, K.; Lee, K.; Suh, C. Community recovery in hypergraphs. *IEEE Trans. Inf. Theory* **2019**, *65*, 6561–6579. [CrossRef]
32. MacKay, D.J.C. Fountain codes. *IEEE Proc. Commun.* **2004**, *152*, 1062–1068. [CrossRef]
33. Cook, L.; Friend, M. Co-Teaching: Guidelines for Creating Effect Practices. *Focus Except. Child.* **1995**, *28*, 1–16. [CrossRef]
34. Huang, J.C.; Qu, L.; Jia, R.F.; Zhao, B.Q. O2U-Net: A Simple Noisy Label Detection Approach for Deep Neural Networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 3325–3333. [CrossRef]
35. Zhou, T.; Wang, S.; Bilmes, J. Robust curriculum learning: From clean label detection to noisy label self-correction. In Proceedings of the International Conference on Learning Representations, Vienna, Austria, 4 May 2021.
36. Donoho, D.L. Compressed Sensing. *IEEE Trans. Inf. Theory* **2006**, *52*, 1289–1306. [CrossRef]

37. MacKay, D.J.C. Good error-correcting codes based on very sparse matrices. *IEEE Trans. Inf. Theory* **1999**, *45*, 399–431. [CrossRef]
38. Seong, J.-T. Theoretical Bounds on Performance in Threshold Group Testing. *Mathematics* **2020**, *8*, 637. [CrossRef]
39. Seong, J.-T. Theoretical Bounds on the Number of Tests in Noisy Threshold Group Testing Frameworks. *Mathematics* **2022**, *10*, 2508. [CrossRef]
40. Cover, T.M.; Thomas, J.A. *Elements of Information Theory*; Wiley: Hoboken, NJ, USA, 2009.
41. Gallager, R. *Information Theory and Reliable Communication*; John Wiley and Sons: Hoboken, NJ, USA, 1968.