When Robust Machine Learning Meets Noisy Supervision: Mechanism, Optimization and Generalization

Bo Han

Centre for Artificial Intelligence

University of Technology Sydney

A thesis submitted for the degree of

Doctor of Philosophy

May, 2019

CERTIFICATE OF ORIGINAL AUTHORSHIP

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of requirements for a degree except as part of the collaborative doctoral degree and/or fully acknowledged within the text.

I also certify that the thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This research is supported by an Australian Government Research Training Program.

Production Note:
Signature removed prior to publication.

Signature of Student: Bo Han
Date: 31/May/2019

I would like to dedicate this thesis to my loving parents and family.

Acknowledgements

I had a wonderful journey at University of Technology Sydney (UTS) for pursuing my PhD degree in the past four years. I am deeply grateful to all those who helped me during the writing of this thesis.

First of all, I would like to express my foremost and deepest gratitude to my advisor, Prof. Ivor W. Tsang, for his visionary, inspiring, and insightful guidance during my PhD period. Honestly speaking, I knew quite little about genuine research when I began my PhD study in UTS. He taught me how to find interesting ideas, how to develop solid algorithms, and how to write papers all from scratch. Besides his knowledge, his research taste encourages me towards the state-of-the-art algorithms, while his work ethic motivates me to become tough and strong. Inspiring by his famous saying "research is like a step function", I am always positive and full of hope in front of my research. Meanwhile, I would like to show my deepest gratitude to my co-advisor, Associate Prof. Ling Chen. She helped me revise my papers, invited me good meals, and often encouraged me, even when I felt very frustrated in my early research. Without their extraordinary support and supervision, I could not achieve what I have right now. I feel extremely grateful for their efforts and friendships.

Moreover, I am greatly indebted to Prof. Chengqi Zhang. Without his endless trust and help, I would never have this valuable opportunity to pursue my PhD research in UTS. Meanwhile, I greatly indebted to the Centre for Artificial Intelligence (CAI) directed by Prof. Jie Lu. In CAI, I got tremendous opportunities to learn from many world-famous experts; I met many colleagues with great enthusiasm for scientific research; and I was also permitted to attend many prestigious international conferences related to my research. Studying in UTS and CAI will be a fantastic memory that I will never forget.

During my PhD period, I am very fortunate to join RIKEN-AIP as a research intern in 1-year "AI Residency" Program, working with Prof. Masashi Sugiyama and Dr. Gang Niu. Prof. Masashi Sugiyama always supported my research ideas, commented all my papers carefully, and approved most of my academic travels financially. Dr. Gang Niu often

brought me excellent ideas, denoted the bar of top-tier research, and handed on many technical details. Our research styles matched very well. Meanwhile, Prof. Mingyuan Zhou from University of Texas Austin taught me Bayesian machine learning; Prof. Xiaokui Xiao from National University of Singapore taught me differential privacy; Dr. Quanming Yao from 4Paradigm Inc. taught me how to write and revise a good paper, and his work ethic won all my respects; Weihua Hu from Stanford University taught me how to evaluate a good idea, and how to follow excellence.

During my PhD period, I have been extremely fortunate to collaborate with three great minds, Jiangchao Yao, Yuangang Pan, and Xingrui Yu, who have prominent impact on my research with their outstanding expertise and profound vision. I will never forget the collaboration experiences with them for catching each deadline. Meanwhile, I must thank to all my wonderful friends, classmates and colleagues, Prof. Yi Yang, Dr. Guodong Long, Dr. Miao Xu, Dr. Liang Zheng, Dr. Zhongwen Xu, Dr. Yan Yan, Dr. Mingming Gong, Dr. Tongliang Liu, Dr. Chen Gong, Dr. Xiyu Yu, Dr. Guoliang Kang, Dr. Shaoli Huang, Dr. Liu Liu, Dr. Jiang Bian, Dr. Jing Chai, Dr. Xiaobo Shen, Dr. Haishuai Wang, Dr. Xun Yang, Dr. Guansong Pan, Dr. Peng Hao, Dr. Qinzhe Zhang, Dr. Fan Zhang, Yiliao Song, Feng Liu, Hehe Fan, Hu Zhang, Yanbing Liu, Linchao Zhu, Pingbo Pan, Zhedong Zheng, Ruiqi Hu, Donna Xu, Yaxin Shi, Bo-Jian Hou, Yao-Xiang Ding, and many others, for every enjoyable moment.

Last but not the least, my gratitude also extends to all my family members who have been consistently supporting, encouraging and caring for me all of my life! Most importantly, I feel strongly indebted to my parents Junxiao Han and Jianping Wang. I would never have gone so far without their unconditional love, accompany and support. I would also express my special thanks to my aunt Jianhua Wang, uncle Zhisheng Si and cousin Wei Si for their great help in many years.

Abstract

Modern machine learning is migrating to the era of complex models (i.e., deep neural networks), which requires a plethora of well-annotated data. Crowdsourcing is a promising tool to achieve this goal, since a plethora of labels that can be efficiently collected from crowdsourcing services at very low cost. However, existing crowdsourcing approaches barely acquire a sufficient amount of high-quality labels. This brings the *first* question: How to design the robust mechanism to improve the label quality?

Without such robust mechanism, labels annotated by crowdsourced workers are often noisy, which inevitably degrades the performance of large-scale optimizations, including the prevalent stochastic gradient descent (SGD). Specifically, these noisy labels adversely affect updates of the primal variable in conventional SGD. This bring the *second* question: How to optimize the training model robustly under noisy labels?

Without such robust optimization, it is challenging to train deep neural networks robustly with noisy labels, as the learning capacity of deep neural networks is so high that they can totally memorize and over-fit on these noisy labels. This brings the *third* question: How to acquire the robust model with good generalization under noisy labels? Therefore, in this thesis, we aim to develop a series of robust machine learning approaches, so that they can perfectly handle the difficult from noisy supervision. Our works are summarized as follows:

Chapter 2 answers the first question. Motivated by the "Guess-with-Hints" answer strategy from the Millionaire game show, we introduce the hint-guided approach into crowdsourcing to deal with this challenge. Our approach encourages workers to get help from hints when they are unsure of questions. Specifically, we propose a hybrid-stage setting, consisting of the main stage and the hint stage. When workers face any uncertain question on the main stage, they are allowed to enter the hint stage and look up hints before making any answer. A unique payment mechanism that meets two important design principles is developed. Besides, the proposed mechanism further encourages high-quality workers less using hints, which helps identify and assigns larger possible payment to them.

Experiments are performed on Amazon Mechanical Turk, which show that our approach ensures a sufficient number of high-quality labels with low expenditure and detects high-quality workers.

Chapter 3 answers the second question. We propose a robust SGD mechanism called PrOgressive STochAstic Learning (POSTAL), which naturally integrates the learning regime of curriculum learning (CL) with the update process of vanilla SGD. Our inspiration comes from the progressive learning process of CL, namely learning from "easy" tasks to "complex" tasks. Through the robust learning process of CL, POSTAL aims to yield robust updates of the primal variable on an ordered label sequence, namely from "reliable" labels to "noisy" labels. To realize POSTAL mechanism, we design a cluster of "screening losses", which sorts all labels from the reliable region to the noisy region. We derive the convergence rate of POSTAL realized by screening losses. Meanwhile, we provide the robustness analysis of representative screening losses. Experiments on benchmark datasets show that POSTAL using screening losses is more effective and robust than several existing baselines.

Chapter 4 answers the third question. Motivated by the memorization effects of deep networks, which shows networks fit clean instances first and then noisy ones, we present a new paradigm called "*Co-teaching*" combating with noisy labels. We train two networks simultaneously. First, in each mini-batch data, each network filters noisy instances based on memorization effects. Then, it teaches the remained instances as the useful knowledge to its peer network for updating the parameters. Empirical results on three benchmark datasets demonstrate that, the robustness of deep learning models trained by Co-teaching approach is much superior than that of state-of-the-art methods.

Contents

Contents vii

List of Figures x

List of Tables xiii

1 Introduction 1
- 1.1 Motivations . . . 1
- 1.2 Background . . . 3
  - 1.2.1 Mechanism . . . 3
  - 1.2.2 Optimization . . . 4
  - 1.2.3 Generalization . . . 5
  - 1.2.4 Real-world Applications . . . 5
- 1.3 Related Work . . . 6
  - 1.3.1 Robust Mechanism . . . 6
  - 1.3.2 Robust Optimization . . . 8
  - 1.3.3 Robust Generalization . . . 10
- 1.4 Contributions . . . 11
- 1.5 Thesis Structure . . . 15
- 1.6 Publications during PhD Study . . . 15

2 Millionaire: Towards Robust Mechanism 18
- 2.1 Problem Setup . . . 19
  - 2.1.1 Hint-guided Approach . . . 19
    - 2.1.1.1 Hybrid-stage Setting . . . 19
    - 2.1.1.2 Model Assumption . . . 21
    - 2.1.1.3 Payment Mechanism . . . 22
    - 2.1.1.4 Difference from Previous Approaches . . . 23
  - 2.1.2 General Rules of Hints . . . 23
  - 2.1.3 Needs of Hybrid-stage Setting . . . 24
- 2.2 Hint-guided Payment Mechanism . . . 24

2.2.1 Design Principles . . . 25
2.2.2 Proposed Payment Mechanism . . . 25
2.2.3 No Other Compatible Mechanism . . . 33
2.3 Numerical Experiments . . . 34
2.3.1 Experimental Setup . . . 34
2.3.2 Payment Parameters . . . 36
2.3.3 Experimental Results . . . 37
2.3.3.1 Label Quantity . . . 37
2.3.3.2 Label Quality . . . 38
2.3.3.3 Worker Quality Dectection . . . 38
2.3.3.4 Spammer Prevention . . . 39
2.3.3.5 Money Cost . . . 39
2.4 Summary of This Chapter . . . 40

3 POSTAL: Towards Robust Optimization 43
3.1 Mechanism of Progressive Stochastic Learning . . . 44
3.2 Realization of Progressive Stochastic Learning . . . 47
3.3 Analysis of Progressive Stochastic Learning . . . 50
3.3.1 Convergence Analysis . . . 50
3.3.2 Robustness Analysis . . . 52
3.4 Experiments . . . 55
3.4.1 Experimental Setup . . . 56
3.4.1.1 Baselines . . . 56
3.4.1.2 Parameters . . . 57
3.4.2 Empirical Study on UCI Simulated Datasets . . . 57
3.4.2.1 Effectiveness of POSTAL Mechanism . . . 57
3.4.2.2 Effectiveness of POSTAL using Screening Losses . . . 62
3.4.2.3 Robustness Improvement . . . 62
3.4.3 Real-World Application on AMT Crowdsourced Data . . . 63
3.4.3.1 Robustness Improvement . . . 63
3.5 Summary of This Chapter . . . 64

4 Co-teaching: Towards Robust Generalization 65
4.1 Co-teaching Meets Noisy Supervision . . . 66
4.1.1 Two important questions . . . 66
4.1.2 Relations to Co-training . . . 68
4.2 Experimental Setup . . . 68
4.2.1 Datasets . . . 68
4.2.2 Baselines . . . 70
4.2.3 Network Structure . . . 71
4.2.4 Noise Rates . . . 72

4.2.5 Measurements . . . 72
4.3 Empirical Results . . . 72
4.3.1 Results on *MNIST* . . . 72
4.3.2 Results on *CIFAR10* . . . 74
4.3.3 Results on *CIFAR100* . . . 75
4.3.4 Choices of $R(T)$ . . . 75
4.3.5 Choices of τ . . . 76
4.4 Full Figures . . . 77
4.4.1 *MNIST* . . . 78
4.4.2 *CIFAR10* . . . 79
4.4.3 *CIFAR100* . . . 80
4.5 Summary of This Chapter . . . 80

5 Conclusion and Future Work **81**
5.1 Thesis Summarization . . . 81
5.2 Future Work . . . 82

References **84**

List of Figures

1.1 A task that requires workers to answer the question "Which one is the Sydney Harbour Bridge?". **Top panel**: the proposed interface under the hybrid-stage setting, consists of two options ("A" and "B") and a "? & Hints" button. **Bottom panel**: when workers feel unsure of this question and click the button, the content of hints (gray) is visible, which guides workers to make a choice. 13

1.2 Comparison of error flow among MentorNet (M-Net) [Jiang et al., 2018], Decoupling [Malach and Shalev-Shwartz, 2017] and Co-teaching. Assume that the error flow comes from the biased selection of training instances, and error flow from network A or network B network is denoted by red arrows or blue arrows, respectively. **Left panel:** M-Net maintains only one network (A). **Middle panel:** Decoupling maintains two networks (A & B). The parameters of two networks are updated, when the predictions of them disagree (!=). **Right panel:** Co-teaching maintains two networks (A & B) simultaneously. In each mini-batch data, each network samples its small-loss instances as the useful knowledge, and teaches such useful instances to its peer network for the further training. Thus, the error flow in Co-teaching displays the zigzag shape. 14

1.3 The structure of this thesis. 16

2.1 Mathematical model of the decision process under our hybrid-stage setting. 21

2.2 Evaluation of label quality. Percentage (in %) of correct answers and incorrect answers on three tasks are provided. Note that, we do not plot the percentage of unlabeled questions. 41

2.3 Evaluation of the label quality. Results on *Speech Clips* are not reported, as text answers cannot be majority voted. 42

2.4 Evaluation of the spammer prevention. Average payment to each worker on all three tasks are provided. Evaluation of the money cost. . 42

3.1 **Left Panel:** Circles denote *real positive* instances such as "A". Squares represent *real negative* instances such as "B" and "C"; however, both "B" and "C" are *erroneously* annotated as positive class, which creates two noisy labels. According to their distance to the positive hyperplane, the label of "A" is reliable; while the labels of "B" and "C" are noisy. **Right Panel:** "A" (z_A) is located in the reliable region defined by $z \geq 0$; while "B" (z_B) and "C" (z_C) are located in the noisy region defined by $z \leq 0$. In Definition 3.1, $z = yf_{\mathbf{w}}(\mathbf{x})$ is formally defined as the curriculum in POSTAL. There are two points to be noted that: 1) Ramp Loss is parameterized by s^* directly; and 2) Logistic, Hinge and Ramp Losses are drawn as the baselines of two screening losses. 45

3.2 To verify the *effectiveness* of POSTAL mechanism, we compare POSTAL with vanilla SGD under the same screening losses. We provide the testing error rate (in %, TER) with the number of epochs on representative UCI noisy datasets: small-scale *A7A*, large-scale *SUSY* and high-dimensional *REAL-SIM* with 20% and 40% of noisy labels. . 58

3.3 To verify the *effectiveness* of POSTAL using screening losses, we compare POSTAL(SRamp) and POSTAL(SGomp) with stochastic methods in the second category of baselines (paragraph 1 in Section 3.4.1). We provide the testing error rate (in %, TER) with the number of epochs on representative UCI noisy datasets: small-scale *A7A*, large-scale *SUSY* and high-dimensional *REAL-SIM* with 20% and 40% of noisy labels. For PEGASOS, the number of epochs for convergence is set to $\frac{10}{\lambda}(\gg 15)$ by default. Thus, its result is not reported. 59

3.4 To verify the *robustness* of POSTAL using screening losses, we compare POSTAL(SRamp) and POSTAL(SGomp) with all baselines on all UCI noisy datasets. We provide the testing error rate (in %, TER) with varying percentages (0% to 40%) of noisy labels. Note that the color of each method is uniform. 60

3.5 To verify the *robustness* of POSTAL using screening losses, we compare POSTAL(SRamp) and POSTAL(SGomp) with all baselines on all UCI noisy datasets. We provide the variance with varying percentages (0% to 40%) of noisy labels. Note that the color of each method is uniform. 61

3.6 Sample images of four categories of dogs from the *Stanford Dogs* dataset. 63

4.1 Transition matrices of different noise types (using 5 classes as an example). 69

4.2 Test accuracy vs number of epochs on *MNIST* dataset. 73

4.3 Label precision vs number of epochs on *MNIST* dataset. 74

4.4 Results on *CIFAR10* dataset. Top: test accuracy vs number of epochs; bottom: label precision vs number of epochs. 75
4.5 Results on *CIFAR100* dataset. Top: test accuracy vs number of epochs; bottom: label precision vs number of epochs. 77
4.6 Results on *MNIST* dataset. Top: test accuracy vs number of epochs; bottom: label precision vs number of epochs. 78
4.7 Results on *CIFAR10* dataset. Top: test accuracy vs number of epochs; bottom: label precision vs number of epochs. 79
4.8 Results on *CIFAR100* dataset. Top: test accuracy vs number of epochs; bottom: label precision vs number of epochs. 80

List of Tables

2.1 Comparison of related approaches and our hint-guided approach (Chapter 2). Baseline is the approach explained above. The skip-based approach comes from [Ding and Zhou, 2017; Shah and Zhou, 2015]. Note that, the self-corrected approach [Shah and Zhou, 2016] is designed theoretically, but barely realized for real tasks (denoted as "×"), so its metrics of "label quality" and "money cost" cannot be evaluated (denoted as "-"). However, since its payment mechanism has a multiplicative form, it prevents spammers theoretically. 18

2.2 The payment parameters. The total payment is composed of FP and RP. RP is based on worker's responses to G questions. $P_h = \frac{\frac{1}{2}-\epsilon}{2T-1}P_d$ according to Algorithm 1, where we set $T = 0.75$ and $\epsilon = 0.191$ due to the Proposition 2. 36

2.3 Evaluation of the label quantity. We provide the percentage of the completion on three tasks. 38

2.4 Evaluation of the worker quality detection of the hint-guided approach. Error rate (in %) is provided for aggregating original and rescaled crowdsourced labels. 39

3.1 Comparison of different learning approaches. POSTAL integrates benefits (emphasized by color) of CL and SGD. 44

3.2 Datasets used in the UCI simulated study. Representative datasets are emphasized by color. 56

3.3 To verify the robustness of POSTAL using screening losses, we also provide the comparison of "testing error rate (in %) with standard deviation" among all methods on UCI clean datasets. Methods indicated by "-"run out of memory. 58

3.4 To verify the robustness of POSTAL using screening losses in real-world situations, we provide the comparison of "testing error rate (in %, TER) with standard deviation" on four crowdsourcing subsets. In each subset, the ratio between positive samples and negative samples is around $1:1$. For LIBLINEAR, the left TER is from LIBPrimal while the right TER is from LIBDual. 64

4.1 Summary of data sets used in the experiments. 68
4.2 Comparison of state-of-the-art techniques with our Co-teaching approach. In the first column, "large noise": can deal with a large number of class; "heavy noise": can combat the heavy noise, i.e., high noise ratio; "flexibility": need not combine with specific network architecture; "no pre-train": can be train from scratch. 70
4.3 CNN and MLP models used in our experiments on *MNIST*, *CIFAR10*, and *CIFAR100*. The slopes of all LReLU functions in the networks are set to 0.01. 71
4.4 Average test accuracy on *MNIST* over the last ten epoch. 73
4.5 Average test accuracy on *CIFAR10* over the last ten epoch. 74
4.6 Average test accuracy on *CIFAR100* over the last ten epoch. 76
4.7 Average test accuracy on *MNIST* over the last ten epoch. 76
4.8 Average test accuracy of Co-teaching with different τ on *MNIST* over the last ten epoch. 77

Chapter 1

Introduction

In this chapter, we clearly depict our motivations, which naturally bring three key questions in robust machine learning. Before addressing these questions, we introduce the background of machine learning, including mechanism, optimization and generalization. Meanwhile, we thoroughly review the related literatures. Lastly, we clearly elaborate our contributions, namely how to address three key questions, and present the organization of the entire thesis.

1.1 Motivations

Huge and complex models (i.e., deep neural networks) are popularly used in today's machine learning applications, since they can take advantage of big data to get better performance. Indeed, they have significantly boosted performance of many important tasks, such as image classification [Russakovsky et al., 2015], speech recognition [Hinton et al., 2012], dialogue systems [Sordoni et al., 2015] and autonomous driving [Bojarski et al., 2016]. More recently, they even beat a human champion by a large margin in the game Go [Silver et al., 2016]. However, a primary question arises: how can we provide a plethora of annotated data to propel complex models? The most appealing way may be the crowdsourcing technology [Russakovsky et al., 2015; Wang and Zhou, 2016; Wang et al., 2017; Zhong et al., 2015], since the process of annotations is convenient and the cost of annotations is very cheap.

While crowdsourcing techniques [Li et al., 2016a, 2017a,b] have been commonly used in many commercial platforms, such as Amazon Mechanical Turk (AMT), the quality of crowdsourced labels is not satisfactory [Ipeirotis et al., 2010]. The reasons are that workers may not be domain experts [Rodrigues et al., 2014; Vuurens et al., 2011; Yan et al., 2014]. For example, it is hard for an average person to distinguish some professional tasks, such as labeling bird images or medical data [Wais et al., 2010a]. Besides, some workers can just be spammers, who response questions with

arbitrary answers [Difallah et al., 2012; Raykar and Yu, 2012]. Such low-quality labels inevitably degenerates the performance of subsequent learning models [Han et al., 2016; Natarajan et al., 2013; Sukhbaatar et al., 2015b]. For instance, noisy labels degrade the accuracy of deep neural networks by 20% to 40% [Patrini et al., 2017; Yu et al., 2017, 2018b]. This brings the *first* question: how to design the robust mechanism to improve the label quality?

Without such robust mechanism, due to the very low reward, most crowdsourcing tasks are normally labeled by amateur workers instead of domain experts , and labels from amateur workers are often noisy [Vuurens et al., 2011; Wais et al., 2010b]. Essentially, noisy labels are corrupted from ground-truth labels, thus, they inevitably degenerate the robustness of learning models, especially for deep neural networks [Arpit et al., 2017; Zhang et al., 2017]. Unfortunately, noisy labels are ubiquitous in the real world. For instance, both online queries [Blum et al., 2003] and crowdsourcing [Yan et al., 2014; Yu et al., 2018c] yield a large amount of noisy labels across the world everyday. This issue inevitably degenerates the performance of large-scale optimizations including stochastic gradient descent (SGD).

SGD has recently become the most prevalent large-scale optimization due to its two merits. Firstly, SGD does not require to calculate the full gradient in per iteration. This merit reduces time costs greatly. Secondly, SGD processes either single point [Mitliagkas et al., 2013] or a tiny batch of points [Cotter et al., 2011] in each iteration. This merit contributes to lower storage costs vastly. As a result, many researchers are working on the area of SGD optimization [Agarwal et al., 2013; Nesterov, 2012]. Given that noisy labels degrade the performance of conventional SGD, by adversely affecting updates of the primal variable, this brings the *second* question: how to design the robust optimization under noisy labels?

Without such robust optimization, as deep neural networks have the high capacity to fit noisy labels [Zhang et al., 2017], it is challenging to train deep networks robustly with noisy labels. Current methods focus on estimating the noise transition matrix. For example, on the top of the softmax layer, Goldberger and Ben-Reuven [2017] added an additional softmax layer to model the noise transition matrix. Patrini et al. [2017] leveraged a two-step solution to estimate the noise transition matrix heuristically. However, the noise transition matrix is not easy to be estimated accurately, especially when the noise ratio is high and the number of classes is large.

To be free of estimating the noise transition matrix, a promising direction focuses on training on selected samples [Jiang et al., 2018; Malach and Shalev-Shwartz, 2017; Ren et al., 2018]. These works try to select clean instances out of the noisy ones, and then use them to update the network. Intuitively, as the training data becomes less noisy, better performance can be obtained. Among those works, the representative methods are MentorNet [Jiang et al., 2018] and Decoupling [Malach and Shalev-Shwartz, 2017]. Specifically, MentorNet pre-trains an extra network, and then uses the extra network selecting clean instances to guide the training. However, the idea

of MentorNet is similar to the self-training approach [Chapelle et al., 2009], thus inherited the same inferiority of accumulated error caused by the sample-selection bias. Decoupling trains two networks simultaneously, and then update the model only using the instances that have the different predictions from these two networks. However, noisy labels are evenly spread across the whole space of examples. Thus, the disagreement area still includes a number of noisy labels, where the decoupling approach can not handle noisy labels explicitly.

Meanwhile, one interesting observation for deep learning models is that they can learn easy instances first, then gradually adapt to hard instances as training epochs become large [Arpit et al., 2017]. When noisy labels exist, deep learning models will eventually memorize these wrongly given labels [Zhang et al., 2017], which leads to the poor generalization performance. This brings the *third* question: how to train the robust model under noisy labels?

To sum up, this thesis will aim to solve three key questions in robust machine learning from mechanism, optimization to generalization:

- How to design the robust mechanism to improve the label quality?
- How to optimize the training model robustly under noisy labels?
- How to acquire the robust model with good generalization under noisy labels ?

1.2 Background

Before delving into our contributions, we introduce the background of machine learning, from the perspectives of mechanism, optimization and generalization.

1.2.1 Mechanism

For the background of mechanism, the baseline approach consists of single-stage setting and additive payment mechanism. The single-stage setting is a special case of the hybrid-stage setting (Section 2.1.1.1), where ϵ is set to 0. Specifically, assume that each worker answers N binary-valued questions, and each question has precisely one correct answer, either "A" or "B". For every question $i \in \{1, \ldots, N\}$, the worker should be incentivized to choose answers matching his/her own belief.

- The single stage: he/she should be incentivized to select the option that he/she feels more confident, namely,

$$\text{select} \begin{cases} \text{“}A\text{”} & P_{A,i} \in [\frac{1}{2}, 1) \\ \text{“}B\text{”} & P_{A,i} \in (0, \frac{1}{2}). \end{cases}$$

Under the single-stage setting, the current state space is $\{\mathbb{D}_+, \mathbb{D}_-\}$. "$\mathbb{D}_+$" and "$\mathbb{D}_-$" denote correct answer and incorrect answer, respectively. The state evaluations of his/her responses to G questions are denoted by $a_1, \ldots, a_G \in \{\mathbb{D}_+, \mathbb{D}_-\}$. Assume that any values d_- and d_+ such that $0 \leq d_- \leq d_+$, a function $f_a : \{\mathbb{D}_+, \mathbb{D}_-\} \to \mathbb{R}_+$, where $f_a(\mathbb{D}_+) = d_+$, $f_a(\mathbb{D}_-) = d_-$. The additive payment mechanism f is:

$$f\left([a_1, \cdots, a_G]\right) = \sum_{i=1}^{G} f_a(a_i). \tag{1.1}$$

Remark 1. *Additive payment mechanism, i.e., Eq.* (1.1) *is not only additive but also incentive-compatible* [1]*. However, due to additive form, if half of the attempts in G questions are correct, the workers still acquire $\frac{(d_-+d_+)G}{2}$ payments. This payment mechanism may not effectively prevent spammers who select options randomly. Here, we prove the additive payment mechanism is incentive-compatible as follows.*

In the single-stage setting, for each question, the expected payment is "$d_+P(\mathbb{D}_+) + d_-P(\mathbb{D}_-)$", where $P(\mathbb{D}_+)$ and $P(\mathbb{D}_-)$ are the probability of correct answer and incorrect answer, respectively. For binary-value questions, $P_B = 1 - P_A$, if the worker chooses "A", which means that $P_A > \frac{1}{2} > P_B$ in his/her belief, then the expected payment for this question is equal to "Payment(A)". To verify the incentive compatibility, we compare "Payment(A)" with "Payment(B)". Due to "$d_+ \geq d_-$", we have Payment(A) $= d_+P_A + d_-P_B > d_+P_B + d_-P_A =$ Payment(B). Therefore, if the worker chooses "A" by his/her belief, the "Payment(A)" is larger than "Payment(B)", and vice versa.

1.2.2 Optimization

For the background of optimization, let $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^n$ be the training data, where $\mathbf{x}_i \in \mathbb{R}^d$ represents the ith instance, and $y_i \in \{-1, +1\}$ represents its binary label. A typical classification model is denoted by:

$$\min_{\mathbf{w}} G(\mathbf{w}) = \min_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^{n} g_i(\mathbf{w}), \tag{1.2}$$

where $\mathbf{w} \in \mathbb{R}^d$ is the primal variable. To be specific, $g_i(\mathbf{w}) = \rho_\lambda(\mathbf{w}) + r(\mathbf{w}; \{\mathbf{x}_i, y_i\})$, where $\rho_\lambda(\mathbf{w})$ is the regularizer and $r(\mathbf{w}; \{\mathbf{x}_i, y_i\})$ is the loss function.

We introduce two foundational definitions: restricted strong convexity (RSC) and restricted smoothness (RSM) [Agarwal et al., 2012; Li et al., 2016b; Loh and Wainwright, 2015]. Based on them, we provide two augmented definitions, namely

[1] A payment mechanism is incentive-compatible if it incentivizes the worker to choose the answers to all questions by his/her own belief and his/her expected payment is strictly maximized.

ARSC and ARSM [Han et al., 2016]. Note that, $\|\cdot\|$ denotes the Euclidean norm, and $B_d(\mathbf{w}^*, \gamma)$ represents the d dimensional Euclidean ball of constant radius γ centered at the optimal point $\mathbf{w}^*$.

Definition 1.1. ***(Augmented Restricted Strong Convexity (ARSC))*** *If there exists a constant $\alpha > 0$ such that for any $\mathbf{w}, \tilde{\mathbf{w}} \in B_d(\mathbf{w}^*, \gamma)$, we have*

$$G(\mathbf{w}) - G(\tilde{\mathbf{w}}) - \langle \nabla G(\tilde{\mathbf{w}}), \mathbf{w} - \tilde{\mathbf{w}} \rangle \geq \frac{\alpha}{2} \|\mathbf{w} - \tilde{\mathbf{w}}\|^2, \tag{1.3}$$

then a differentiable function G satisfies Augmented Restricted Strong Convexity.

Definition 1.2. ***(Augmented Restricted Smoothness (ARSM))*** *If there exists a constant $\beta > 0$ such that for any $i \in \{1, \cdots, n\}$ and $\mathbf{w}, \tilde{\mathbf{w}} \in B_d(\mathbf{w}^*, \gamma)$, we have*

$$g_i(\mathbf{w}) - g_i(\tilde{\mathbf{w}}) - \langle \nabla g_i(\tilde{\mathbf{w}}), \mathbf{w} - \tilde{\mathbf{w}} \rangle \leq \frac{\beta}{2} \|\mathbf{w} - \tilde{\mathbf{w}}\|^2, \tag{1.4}$$

then a differentiable function g_i satisfies Augmented Restricted Smoothness.

1.2.3 Generalization

For the background of generalization, we consider complex models (i.e., deep neural networks) and use memorization effects of deep neural networks [Arpit et al., 2017]. Namely, deep learning models can learn easy instances first, then gradually adapt to hard instances as training epochs become large. When noisy labels exist, deep learning models will eventually memorize these wrongly given labels [Zhang et al., 2017], which leads to the poor generalization performance. Besides, this phenomenon does not change with choice of regularization (e.g., dropout and batch-normalization), type of training optimizations (e.g., Resprop, adagrad and adam) or design of network architecture (e.g., MLP, Alexnet and Inception) [Jiang et al., 2018; Zhang et al., 2017].

1.2.4 Real-world Applications

Learning from noisy supervision has a lot of real-world applications, such as crowdsourcing [Yi et al., 2012], Internet analysis [Niu et al., 2015] and healthcare analysis [Xiao et al., 2018]. Therefore, robust machine learning can be utilized for solving various practical problems such as:

- **Crowdsourcing.** Crowdsourcing has become the top choice for AI companies to acquire annotated data. The reason is that, to achieve better performance, complex models (i.e., deep neural networks) used by AI companies are very hungry for a plethora of annotated data, while such data can be efficiently acquired by crowdsourcing service with the cheap price. However, annotated

data from crowdsourcing are often noisy. This motivates us to leverage game theory to design robust learning algorithms, which can be deployed on crowdsourcing platforms and improve the label quality at the stage of collection.

- **Internet analysis.** Recent years have witnessed the hyper development of the industry of Internet. There is a lot of multimedia data, such as image, text, and video data on the Internet everyday. However, annotations of such multimedia data are normally noisy. For example, in web search, if you input "Jaguar" in Google Images, the feedbacks consist of animal Jaguar and car with Jaguar brand. In implicit feedback, you may want to skip an advertisement of youtube, but mis-click a "like" tag. Therefore, robust learning algorithms can be employed to overcome the negative effects of noisy annotated data.

- **Healthcare analysis.** Healthcare domain has entered the AI-driven era. For example, deep learning has been widely used for analyzing electronic health records (EHR) data. However, the quality of EHR data may be not satisfactory, and even extremely noisy. The reason mainly comes from two causes. First, the sensor used for data collection owns some device deviations physically; second, the people involving in annotations have their subjective biases. Therefore, when healthcare data is extremely noisy, our robust learning algorithms can be leveraged to train models robustly under these noisy data.

1.3 Related Work

Due to the extensive applications and solid mathematical foundations, learning from noisy labels has been investigated by many researchers and various robust methodologies have been developed as a result. The existing robust methodologies can be divided into three orthogonal directions: mechanism, optimization and generalization. This section will review the related literatures according to this taxonomy.

1.3.1 Robust Mechanism

Robust mechanism mainly focuses on pre-processed approach to improve label quality, which is very related to post-processed approach and worker quality control as follows.

Post-Processed Approach: The statistical inference (post-processed) approach is popularly used to improve the quality of labels [Jin et al., 2018; Zhang et al., 2016; Zheng et al., 2015a, 2016, 2017]. Such approach tries to find the correct label for each question only after noisy labels being collected from the platform. Many methods have been developed under this approach. Specifically, the statistical inference consists of discriminative and generative approaches.

Among discriminative approaches, Majority Voting is widely used in practice, since it is not only simple to operate, but also scalable to large-scale crowdsourced labels. Specifically, Majority Voting identifies true labels by simple aggregation rules, without considering worker expertise and sample difficulty. Due to simple aggregations, Majority Voting hardly ensures the aggregation quality of heavily noisy labels. To improve the quality of labels, Li and Yu [2014] proposed weighted Majority Voting , which considers worker expertise by assigning different weights to define worker importance. Tian and Zhu [2015] extended weighted Majority Voting by the max-margin principle, which provides a geometric interpretation of crowdsourcing margin. However, with both methods, inferring weights from heavily noisy labels is non-trivial and time-consuming.

Among generative approaches, the classical Dawid-Skene model works well [Dawid and Skene, 1979]. Each worker is linked to a probabilistic confusion matrix that generates his labels. Raykar et al. [2010a] presented a two-coin probabilistic model, which assumes that each worker's labels are generated by flipping the ground-truth labels according to a certain probability. Yan et al. [2010] extended this two-coin model by setting the dynamic flipping probability associated with samples. Kajino et al. [2012] formulated a probabilistic multi-task model, where each worker is considered as a task. Liu et al. [2012] first introduced variational inference as a way to solve crowdsourcing problems. This method is similar to the message-passing approach [Karger et al., 2011]. Zhou et al. [2012a] proposed a minimax entropy model and extended it to aggregate crowdsourced ordinal labels [Zhou et al., 2014]. Bi et al. [2014] employed a mixture probabilistic model for worker annotations, which learns a prediction model directly. However, as labels are intrinsically noisy, it is hard to obtain a sufficient among of correct labels using statistical inference.

Pre-Processed Approach: While previous efforts have extensively focused on several statistical inferences, pre-processing approach has been recently developed as an alternative way to improve label quality. Namely, the crowdsourced setting is coupled with the payment mechanism, which incentivizes workers to provide more reliable labels at the stage of label collection. Thus, unlike post-processed approach, pre-processed approach can directly reduce the noise in obtained labels. Moreover, post-processed approach can be used to further reduce the noise in labels after they are obtained by the pre-processed approach.

In this thesis, we target the pre-processed approach from the perspective of machine learning [Buhrmester et al., 2011; Ding and Zhou, 2017; Goel et al., 2014; Ho et al., 2015; Lambert et al., 2015; Shah and Zhou, 2015; Singla and Krause, 2013]. The most related works are the skip-based [Ding and Zhou, 2017; Shah and Zhou, 2015] and self-corrected approaches [Shah and Zhou, 2016]. In the skip-based approach, workers are allowed to select a skip option based on their confidence for each question.

However, this in turn leads to insufficient label quantity. A two-stage setting is used in the self-corrected approach. Workers firstly answer all questions in the first stage, and then they are allowed to correct their first-stage answers after looking at a reference in the second stage. However, references consisting of responses from other workers are noisy, which may mislead workers to providing incorrect labels. Besides, as a reference needs to be set for each task, such a setting is not supported by the AMT platform and only simulation results are reported in Shah and Zhou [2016]. Finally, neither the skip-based nor self-corrected approaches can identify worker quality as our approach.

The pre-processed approach was also considered in the database area, but the focus is different. Normally, their research is to dynamically assign the optimal K ($\leq N$) problems to each worker by his/her work quality, where N is the total number of problems to be annotated [Fan et al., 2015; Hu et al., 2016; Zheng et al., 2015b]. Thus, worker quality control plays an fundamental role in the quality of crowdsourcing from the viewpoint of database.

Worker Quality Control: As workers' quality has huge impact on the obtained labels, many researchers tried to improve label quality by offering better control over workers' quality. For example, Raykar and Yu [2012] considered detecting spammers or adversarial behavior, and tried to eliminate them in the following iterations or phases. However, this method does not consider how to detect high-quality workers. Then, Joglekar et al. [2013] devised techniques to generate confidence intervals for worker error rate estimates, thereby enabling a better evaluation of worker quality. However, this method is too complex to be deployed in practice. For our hybrid-stage setting, the less number of times workers enter the hint stage, the higher quality they are estimated to be.

1.3.2 Robust Optimization

Robust optimization can optimize the training model robustly under noisy labels. Robust optimization focuses on the optimization level (e.g., changing update mechanism in SGD). This topic is often related to stochastic optimization, due to the large data volume in the real world. This direction is also related to a robust learning policy called curriculum learning, robust losses and noisy labels as follows.

Stochastic Optimization: The research of this direction is mainly related to SGD optimization [Tao et al., 2014]. For example, Bottou [2010] leveraged the vanilla SGD to optimize complex models such as deep neural networks [Le et al., 2011]. Xu [2011] proposed the averaged SGD (ASGD) to reduce the error rate of vanilla SGD [Bottou, 2010]. Ghadimi and Lan [2013] introduced a randomized stochastic method for non-

convex problems. Meanwhile, they generalized the accelerated gradient method to improve the convergence rate of stochastic optimization for non-convex problems [Ghadimi and Lan, 2016]. Shalev-Shwartz et al. [2011] developed a variant of SGD called PEGASOS for scalable text classification. Nevertheless, all their works have a strong assumption that the data is free of noise. This assumption restricts their applicabilities to the problem of noisy labels. Our POSTAL focuses on learning with noisy labels.

Curriculum Learning: This direction is also related to curriculum learning (CL) and self-paced learning [Gong et al., 2016a; Jiang et al., 2014]. Bengio et al. [2009] proposed a learning framework named curriculum learning , and Kumar et al. [2010] presented the similar learning regime named self-paced learning. The idea shared by these two studies is to learn easier tasks first, and gradually learn more difficult tasks to result in a robust model. However, their mechanisms have not been applied into the update process of SGD for noisy labels. To the best of our knowledge, our POSTAL is the first attempt to integrate the learning regime of CL with the update process of SGD for noisy labels.

Robust Losses: Bounded non-convex losses [Rakotomamonjy et al., 2016] for robust classification are related to our study. For instance, Collobert et al. [2006] developed a kind of bounded loss named "ramp loss" to deal with support vector machine (SVM) classification problems . Wang et al. [2008] smoothed the turning points of ramp loss to suppress outliers . Their non-convex losses are designed for robust SVMs, while our screening losses are tailor-made for POSTAL to provide an ordered label sequence and alleviate adverse effects led by noisy labels.

Noisy Labels: Lastly, this direction deals with noisy labels [Li and Fu, 2016; Liu and Tao, 2016a]. Natarajan et al. [2013] proposed the methods of unbiased estimators and weighted loss functions for noisy labels . However, their work has not been applied to reduce the performance degeneration of SGD caused by noisy labels. Patrini et al. [2016] adapted SGD to deal with asymmetric label noise, but their work has not been verified on real-world noisy datasets. Although Raykar et al. [2010b] described a probabilistic approach to handle noisy labels, their model cannot be scalable to large datasets due to the high cost of computations. It is worth noting that the 0-1 loss function is also designed for suppressing noisy labels. However, the 0-1 loss is non-convex and non-differentiable, which may not be tractable easily in practical situations. Although the surrogates of 0-1 loss are convex [Bartlett et al., 2006], they are still sensitive to noisy labels [Natarajan et al., 2017].

1.3.3 Robust Generalization

Robust generalization is to acquire the robust model with good generalization under noisy labels. robust generalization focuses on the model level (e.g., providing robust training policy). Robust generalization is mainly related to statistical learning, when the training model is linear and simple. However, then the training model becomes very complex, such as deep neural networks, this direction will be highly related to deep learning. Meanwhile, the recent topic in Learning-to-teach will boost the robustness further.

Statistical Learning: Statistical learning contributed a lot to the problem of noisy labels, especially in theoretical aspects. The approach can be categorized into three strands: surrogate loss, noise rate estimation and probabilistic modeling. For example, in the surrogate losses, Natarajan et al. [2013] proposed an unbiased estimator to provide the noise corrected loss approach. Masnadi-Shirazi and Vasconcelos [2009] presented a robust non-convex loss, which is the special case in a family of robust losses Han et al. [2016]. In noise rate estimation, both Menon et al. [2015] and Liu and Tao [2016b] proposed a class-probability estimator using order statistics on the range of scores. Sanderson and Scott [2014] presented the same estimator using the slope of the ROC curve. In probabilistic modeling, Raykar et al. [2010b] proposed a two-coin model to handle noisy labels from multiple annotators. Yan et al. [2014] extended this two-coin model by setting the dynamic flipping probability associated with instances.

Deep Learning: Let us take a close look at how deep learning deals with the label noise problem. In the early stage, Reed et al. [2015] augmented the prediction objective with a notion of consistency. Based on the consistency principle, they developed "reconstruction error" and "hard/soft bootstrapping" approaches. Azadi et al. [2016] proposed an auxiliary image regularization technique, which exploits the mutual information among training samples to select reliable samples for further training.

In the recent stage, state-of-the-art methodologies mainly focus on estimating the noise transition matrix in an end-to-end framework. For instance, Sukhbaatar et al. [2015a] proposed to add a constrained linear layer on the top of the softmax layer, which explicitly matches the noise transition. Goldberger and Ben-Reuven [2017] presented to add an additional softmax layer to model the noise transition. Patrini et al. [2017] leveraged a two-step solution to estimate the transition matrix heuristically, and proposed the strategy of forward/backward loss correction given such estimated matrix. However, when the number of classes becomes very large, the transition matrix is not easy to be estimated exactly.

Free of estimating the transition matrix, Jiang et al. [2018] presented a MentorNet to regularize deep networks in the data dimension inspired by curriculum learning.

However, the idea of MentorNet is similar to the self-training approach, which may introduce the accumulated error due to the biased selection. Besides, Malach and Shalev-Shwartz [2017] proposed a decoupling approach to decouple "when to update" from "how to update" in training deep networks. Specifically, they maintain two networks, and update parameters of two networks only when the predictions of them disagree. However, the decoupling approach may not combat with massive noisy labels, since the disagreement area still overlap with the noise area.

In addition, there are some other solutions to deal with noisy labels from computer vision community. For example, Li et al. [2017c] proposed a unified framework to distill the knowledge from clean labels and knowledge graph, which can be exploited to learn a better model from noisy labels. Veit et al. [2017] trained a label cleaning network by a small set of clean labels, and used this network to reduce the noise in large-scale noisy labels. Tanaka et al. [2018] presented a joint optimization framework to learn parameters and estimate true labels simultaneously. Ren et al. [2018] leveraged an additional validation set to adaptively assign weights to training examples in every iteration. Rodrigues and Pereira [2018] added a crowd layer after the output layer for noisy labels from multiple annotators. However, all their methods require either extra resources or more complex networks.

Learn to Teach: Learning-to-teach is also a hot topic. Inspired by [Hinton et al., 2015], these methods is made up by one teacher and student networks. The duty of teacher network is to select more informative instances for better training of student networks. Recently, such idea is applied for learning a proper curriculum for the training data [Fan et al., 2018] and deal with multi-labels [Gong et al., 2016b]. However, these works does not consider noisy labels, and MentorNet [Jiang et al., 2018] introduced this idea into such area.

1.4 Contributions

To address three key questions in robust machine learning, we proposed three robust methodologies in this thesis, which are "Guess-with-Hints" (robust mechanism), "PrOgressive STochAstic Learning (POSTAL)" (robust optimization), and "Co-teaching" (robust generalization), respectively.

In Guess-with-Hints, we are inspired by the "Guess-with-Hints" answer strategy from the Millionaire game show [1], where a challenger has opportunities to request hints from the show host when he/she feels unsure of the questions. By this strategy, we introduce a hint-guided approach to improve the quality of crowdsourced labels.

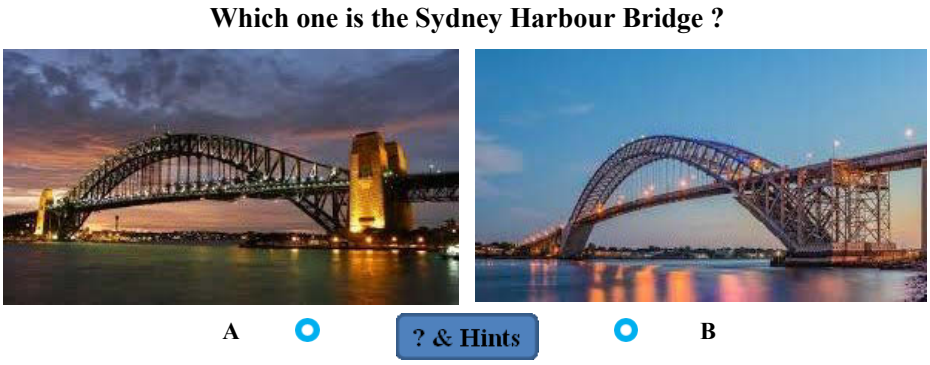
[1] https://en.wikipedia.org/wiki/Who_Wants_to_Be_a_Millionaire_(U.S._game_show)

This approach encourages workers to get help from auxiliary hints when they answer questions that they are unsure of. To be specific, we introduce a hybrid-stage setting, which consists of the main stage and the hint stage. In the main stage, for each question, workers answer it directly when they feel confident or jump into the hint stage when they feel uncertain. Once they enter the hint stage, they are allowed to look up hints before making any answer to this unsure question. The less number of times workers enter the hint stage, the higher quality they are estimated to be. To realize this setting, we provide an explicit "? & Hints" button (the bottom panel in Figure 1.1) for each question. For example, when the worker is unsure of the question in Figure 1.1, he/she can click this button and answer the question under the help of hints (the gray sentence).

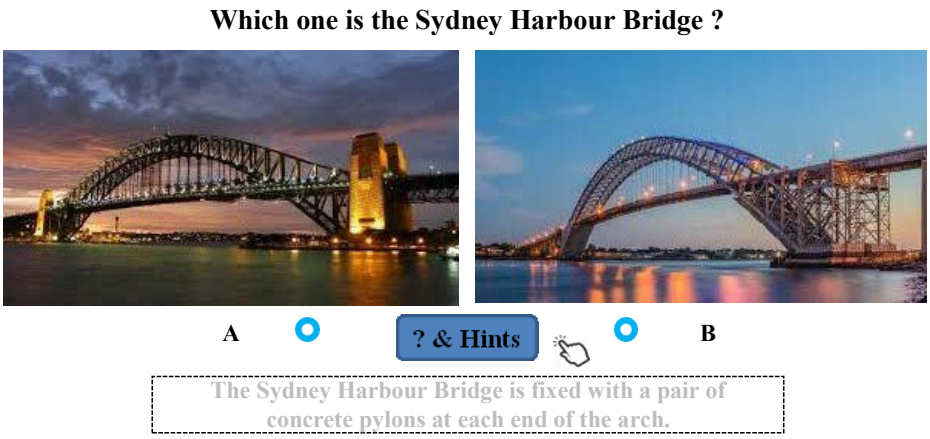
Nevertheless, only hybrid-stage setting is not enough to address all issues. For example, if hints are freely available in the hint stage, even high-quality workers may abuse free hints for higher accuracy and rewards. This issue causes failure in the detection of high-quality workers. Under the hybrid-stage setting, we develop a hint-guided payment mechanism, which aims to incentivize workers to use the hints properly. Specifically, our mechanism penalizes workers who use the hints. Therefore, high-quality workers will answer most of the questions directly (without hints) for higher rewards. Then, our mechanism assists our setting to detect the high-quality workers effectively. Moreover, we prove that our mechanism is unique under the hybrid-stage setting. Since our mechanism has a multiplicative form, it prevents spammers as well.

In POSTAL, we introduce a robust SGD mechanism that integrates the progressive learning regime - curriculum learning (CL) with the update process of vanilla SGD. Our inspiration springs from the learning process of CL, namely learning from "easy" tasks to "complex" tasks, which is often used for training robust models [Bengio et al., 2009; Kumar et al., 2010]. Through this type of learning process, POSTAL aims to yield robust updates of the primal variable on an ordered label sequence, namely from "reliable" labels to "noisy" labels. To provide this ordered label sequence, we design a cluster of screening losses, which serves as a guidance to sort all labels from the reliable region to the noisy region. Moreover, screening losses assist POSTAL to reduce the performance degeneration of SGD yielded by noisy labels. To sum up, in the first epoch, POSTAL using screening losses ensures the update of the primal variable on reliable labels, which creates a robust model from the outset. In subsequent epochs, updates of the primal variable occur on noisy labels gradually until convergence.

In Co-teaching, we present a simple but effective learning paradigm, which allows us to train deep networks robustly even with extremely noisy labels (e.g., 45% of noisy labels occur in the fine-grained classification with multiple classes [Deng et al., 2013]). Our idea stems from the Co-training approach [Blum and Mitchell, 1998]. Similarly to Decoupling, our Co-teaching also maintains two networks simultaneously. However, it is worth noting that, in each mini-batch data, each network views its small-loss



(a) Main stage.



(b) Hint stage.

Figure 1.1: A task that requires workers to answer the question "Which one is the Sydney Harbour Bridge?". **Top panel**: the proposed interface under the hybrid-stage setting, consists of two options ("A" and "B") and a "? & Hints" button. **Bottom panel**: when workers feel unsure of this question and click the button, the content of hints (gray) is visible, which guides workers to make a choice.

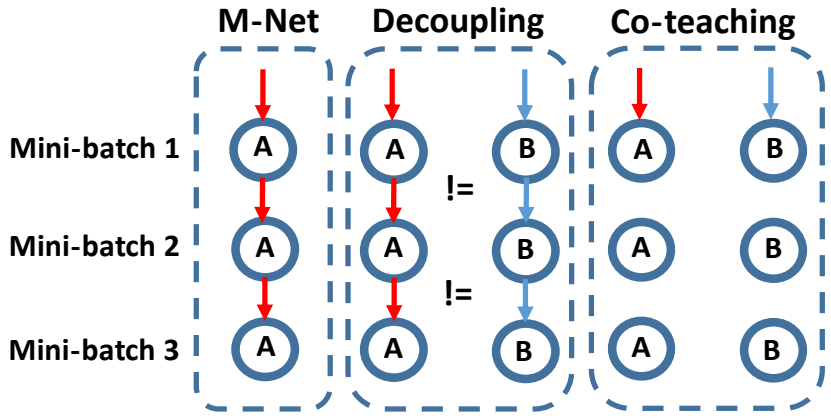


Figure 1.2: Comparison of error flow among MentorNet (M-Net) [Jiang et al., 2018], Decoupling [Malach and Shalev-Shwartz, 2017] and Co-teaching. Assume that the error flow comes from the biased selection of training instances, and error flow from network A or network B network is denoted by red arrows or blue arrows, respectively. **Left panel:** M-Net maintains only one network (A). **Middle panel:** Decoupling maintains two networks (A & B). The parameters of two networks are updated, when the predictions of them disagree (!=). **Right panel:** Co-teaching maintains two networks (A & B) simultaneously. In each mini-batch data, each network samples its small-loss instances as the useful knowledge, and teaches such useful instances to its peer network for the further training. Thus, the error flow in Co-teaching displays the zigzag shape.

instances as the useful knowledge, and teaches such useful instances to its peer network for updating the parameters. The intuition why Co-teaching is more robust can be briefly explained in Figure 1.2.

Assume that the error flow comes from the biased selection of training instances in the first mini-batch data. In MentorNet or Decoupling, the error from one network will be directly transferred back itself in the second mini-batch data, and the error should be increasingly accumulated. However, in Co-teaching, since two networks have different learning abilities, they can filter different types of error introduced by noisy labels. In this exchange procedure, the error flows can be reduced by peer networks mutually. Moreover, we train deep networks using stochastic optimization with momentum, and nonlinear deep networks can memorize clean data first to become robust [Arpit et al., 2017]. When the error from noisy data flows into the peer network, it will attenuate this error due to its robustness.

We conduct experiments on simulated noisy *MNIST*, *CIFAR10*, and *CIFAR100* datasets. Empirical results demonstrate that, under extremely noisy labels, the robustness of deep learning models trained by Co-teaching approach is much superior than that of all state-of-the-art baselines. Under low-level noisy labels (i.e., 20% of

noisy labels), the robustness of deep learning models trained by Co-teaching approach is still superior than that of most baselines.

In summary, the main contributions of this thesis lie in the following three aspects:

1. We propose to improve the quality of labels by auxiliary hints. We introduce a hybrid-stage setting. Under this setting, we propose a hint-guided payment mechanism, which incentivizes workers to use hints properly instead of abusing them. Moreover, we prove the uniqueness of our mechanism under the proposed setting.

2. We introduce a robust SGD mechanism called PrOgressive STochAstic Learning (POSTAL) to handle noisy labels. For the label noise problem, POSTAL is the first attempt to leverage the learning regime of CL to ensure robust updates of the primal variable in vanilla SGD. We design a cluster of screening losses, which serves as a guidance to provide an ordered label sequence for POSTAL. Moreover, it assists POSTAL to reduce the performance degeneration of SGD yield by noisy labels.

3. We present a new paradigm called Co-teaching combating with noisy labels. We train two networks simultaneously. First, in each mini-batch data, each network filters noisy instances based on memorization effects. Then, it teaches the remained instances to its peer network for updating the parameters.

1.5 Thesis Structure

To achieve robust methodologies, this thesis proposes three orthogonal methods Guess-with-Hints, POSTAL and Co-teaching based on different motivations. The remaining parts of this thesis are organized as follows:

Chapter 2 introduces the robust mechanism for achieving the high-quality labels.
Chapter 3 introduces the robust optimization under noisy labels.
Chapter 4 introduces the robust generalization under extremely noisy labels.
The structure of the entire thesis is illustrate in Fig. 1.3.

1.6 Publications during PhD Study

1. **Bo Han**, Ivor W. Tsang, Ling Chen. On the Convergence of a Family of Robust Losses for Stochastic Gradient Descent. European Conference on Machine Learning (ECML), 2016. (oral)

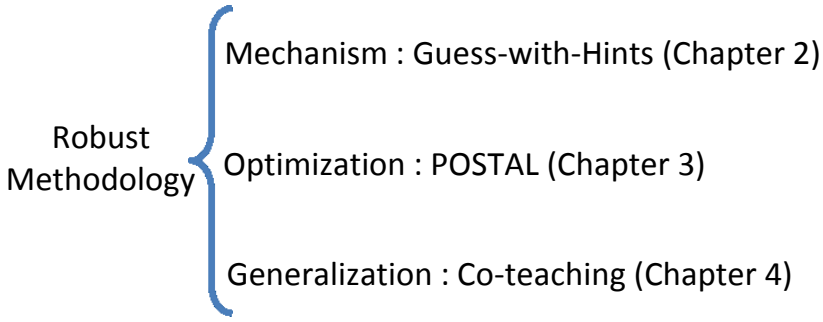


Figure 1.3: The structure of this thesis.

2. **Bo Han**, Ivor W. Tsang, Ling Chen, Celina Yu, Sai-Fu Fung. Progressive Stochastic Learning for Noisy Labels. IEEE Transactions on Neural Networks and Learning Systems (TNNLS), 2018, 99: 1-13.

3. **Bo Han**, Yuangang Pan, Ivor W. Tsang. Robust Plackett-–Luce Model for k-ary Crowdsourced Preferences. Machine Learning (MLJ), 2018, 107(4): 675-702.

4. **Bo Han**, Quanming Yao, Yuangang Pan, Ivor W. Tsang, Xiaokui Xiao, Yang Qiang, Masashi Sugiyama. Millionaire: A Hint-guided Approach for Crowdsourcing. Machine Learning (MLJ), 2018, 1-29.

5. **Bo Han**, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor W. Tsang, Masashi Sugiyama. Co-teaching: Robust Training of Deep Neural Networks with Extremely Noisy Labels. Advances in Neural Information Processing Systems (NeurIPS), 2018. (poster)

6. **Bo Han**, Jiangchao Yao, Gang Niu, Mingyuan Zhou, Ivor W. Tsang, Zhang Ya, Masashi Sugiyama. Masking: A New Perspective of Noisy Supervision. Advances in Neural Information Processing Systems (NeurIPS), 2018. (poster)

7. **Bo Han**, Ivor W. Tsang, Ling Chen, Joey Tianyi Zhou, Celina Yu. Beyond Majority Voting: A Coarse-to-Fine Label Filtration for Heavily Noisy Labels. IEEE Transactions on Neural Networks and Learning Systems (TNNLS), 2019, 1-14.

8. **Bo Han**, Ivor W. Tsang, Xiaokui Xiao, Ling Chen, Sai-Fu Fung, Celina Yu. Privacy-preserving Stochastic Gradual Learning. IEEE Transactions on Knowledge and Data Engineering (TKDE), 2019. (major revision)

9. Yuangang Pan, **Bo Han**, Ivor W. Tsang. Stagewise Learning for Noisy k-ary Preferences. Machine Learning (MLJ), 2018, 1-29.

10. Xingrui Yu, **Bo Han**, Jiangchao Yao, Gang Niu, Ivor W. Tsang, Masashi Sugiyama. How does Disagreement Help Generalization against Label Corruption? International Conference on Machine Learning (ICML), 2019. (long oral)

11. Jingfeng Zhang, **Bo Han**, Laura Wynter, Kian Hsiang Low, Mohan Kankanhalli. Towards Robust ResNet: A Small Step but A Giant Leap. International Joint Conference on Artificial Intelligence (IJCAI), 2019.

12. Quanming Yao, James T. Kwok, **Bo Han**. Efficient Nonconvex Regularized Tensor Completion with Structure-aware Proximal Iterations. International Conference on Machine Learning (ICML), 2019. (oral)

Chapter 2

Millionaire: Towards Robust Mechanism

This chapter answers the first question, namely "how to design the robust mechanism to improve the label quality?". From the high level, we want to design the robust mechanism based on the fundamental assumptions from game theory [Nisan et al., 2007]. Namely, our robust mechanism is provably effective.

Table 2.1: Comparison of related approaches and our hint-guided approach (Chapter 2). Baseline is the approach explained above. The skip-based approach comes from [Ding and Zhou, 2017; Shah and Zhou, 2015]. Note that, the self-corrected approach [Shah and Zhou, 2016] is designed theoretically, but barely realized for real tasks (denoted as "×"), so its metrics of "label quality" and "money cost" cannot be evaluated (denoted as "-"). However, since its payment mechanism has a multiplicative form, it prevents spammers theoretically.

Perspective	Metric	Baseline	Skip-based	Self-corrected	**Hint-guided** (ours)
requester	large label quantity	✓	×	✓	✓
	high label quality	×	✓	-	✓
worker	quality detection	×	×	×	✓
	spammer prevention	×	✓	✓	✓
platform	low money cost	×	✓	-	✓
	realization	✓	✓	×	✓

Specifically, there are some problems in improving the label quality (Table 2.1). For example, skip-based approach allows workers to select a skip option when they feel unsure of the questions. However, high-quality labels are at the expense of the insufficient label quantity, and this issue is very critical to the complex models training. Self-correct approach allows workers to correct their first-stage answer after looking at a reference in the second stage. However, references consisting of responses from other workers are noisy, which may mislead workers to providing incorrect labels.

Besides, this approach is designed theoretically, but barely realized in the real-world platform.

Motivated by the "Guess-with-Hints" answer strategy from the Millionaire game show, we introduce the hint-guided approach into crowdsourcing to deal with this challenge. Our approach encourages workers to get help from hints when they are unsure of questions. Specifically, we propose a hybrid-stage setting, consisting of the main stage and the hint stage. When workers face any uncertain question on the main stage, they are allowed to enter the hint stage and look up hints before making any answer. Following this thought, we design a physical interface (Figure 1.1). More importantly, a unique payment mechanism that meets two important design principles is developed. Besides, the proposed mechanism further encourages high-quality workers less using hints, which helps identify and assigns larger possible payment to them. Experiments are performed on Amazon Mechanical Turk, which show that our approach ensures a sufficient number of high-quality labels with low expenditure and detects high-quality workers.

The remainder of this chapter is organized as follows. Section 2.1 introduces the novel setup in crowdsourcing, namely the hybrid-stage setting. In Section 2.2, we propose a hint-guided payment mechanism under this setting. In Section 2.3, we provide the experiment setup and empirical results related to three real-world tasks. The conclusions are given in Section 2.4.

2.1 Problem Setup

Inspired by the "Guess-with-Hints" answer strategy, we introduce the hint-guided approach to improve the quality of crowdsourced labels and detect the high-quality workers at the same time. This approach encourages workers to get help from the useful hints when they answer uncertain questions (Figure 1.1). Specifically, we realize this approach in Section 2.1.1, including the hybrid-stage setting and the payment mechanism. Then, easy usage of hints is discussed in Section 2.1.2. Finally, the rationality of our design is discussed in Section 2.1.3.

2.1.1 Hint-guided Approach

Here, we describe our hint-guided approach from the following four aspects.

2.1.1.1 Hybrid-stage Setting

We first set up definitions for the hybrid-stage setting that consists of the main stage and the hint stage. To model our setting, let us consider a simple example: each worker answers N binary-valued (objective) questions, and each question has precisely one

correct answer, either "A" or "B". Therefore, for every question $i \in \{1, \ldots, N\}$, a worker chooses an answer matching his/her own belief under the following hybrid-stage setting.

- The main stage (Figure 1.1(a)): For question i, he/she should be incentivized to select the option that he/she feels confident. When he/she feels unsure and clicks the "? & Hints" button, he/she jumps into the hint stage formalized by the "H" option, namely,

$$\text{select} \begin{cases} \text{“}A\text{”} & \text{if } P_{A,i} \in [\frac{1}{2} + \epsilon, 1), \\ \text{“}B\text{”} & \text{if } P_{A,i} \in (0, \frac{1}{2} - \epsilon], \\ \text{“}H\text{”} & \text{otherwise,} \end{cases}$$

  where $\epsilon \in [0, \frac{1}{2})$ models the worker's uncertainty degree in this stage, $P_{A,i}$ is the probability of the worker's belief that the answer to the ith question is "A" (i.e., the probability that the worker believes "A" is the correct answer for the ith question).

- The hint stage (Figure 1.1(b)): When he/she feels unsure of the question, the worker clicks the "? & Hints" button. This means that he/she enters the hint stage. Then, the worker picks up "A" or "B" according to

$$\text{select} \begin{cases} \text{“}A\text{”} & \text{if } P_{A|H,i} \in [T, 1), \\ \text{“}B\text{”} & \text{if } P_{B|H,i} \in [T, 1), \end{cases}$$

  where $T \in (\frac{1}{2}, 1)$ is the predefined threshold value of the worker's belief in the hint stage, $P_{A|H,i}$ is the probability of the worker's belief that the answer to the ith question is "A" given hints, and $P_{B|H,i}$ is the probability of the worker's belief that the answer to the ith question is "B" given hints ($P_{B|H,i} = 1 - P_{A|H,i}$).

The above modeling of the decision process is also summarized in Figure 2.1. As we can see, ϵ controls the decision in the main stage and the hint stage depends on T. When ϵ is large, i.e., $\epsilon \to \frac{1}{2}$, more workers need hints to make their decision for each question. When ϵ is smaller, i.e., $\epsilon \to 0$, fewer workers need hints to make their decision for each question. Once the worker enters the hint stage, when T is set to a large value, i.e., $T \to 1$, he/she will become more confident to make his/her final decision for each question. When T is set to a small value, i.e., $T \to \frac{1}{2}$, he/she will be less confident to make his/her final decision for each question.

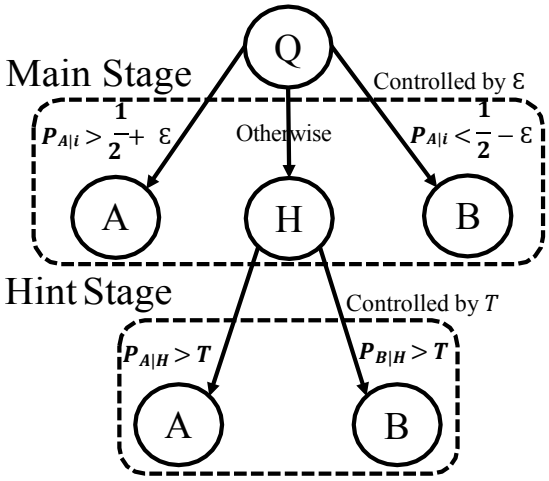


Figure 2.1: Mathematical model of the decision process under our hybrid-stage setting.

Note that, ϵ is decided by T according to Proposition 2 in Section 2.2.2, and T is controlled by a mechanism designer. The choice of T is based on different applications and given to us. In the experiments, we empirically choose $T = 0.75$ due to the qualitative psychology [Smith, 2007].

2.1.1.2 Model Assumption

Based on the hybrid-stage setting, we will introduce the corresponding payment mechanism, where it is rooted in the following assumption.

Assumption 1. *(A). There are G "gold standard" questions ($1 \leq G \leq N$), of which answers are known to the requester, uniformly distributed at random positions among all N questions;*

(B). Each worker aims to maximize his/her expected payment for N questions;

Assumption 1 is a standard one in analyzing pre-processed approaches for crowdsourcing [Shah and Zhou, 2015, 2016; Zhang et al., 2016]. Specifically, as answers to the "gold standard" questions are known to the requester in advanced, workers' responses to them can be used to evaluate workers' performance and decide payment for workers. This is the functionality of Assumption 1 (A). Then, Assumption 1 (B) is a must for analyzing workers' performance. It originates from game theory [Nisan et al., 2007], and means that each work wants to maximize its revenue. Besides, Assumption 1 (B) would not allow for malicious or adversarial workers.

Next, we make the following Assumption 2, which specifies our usage of hints here. It is motivated by the educational psychology [Koedinger and Aleven, 2007], and means that the hints are useful enough to guide workers to making final decisions.

Assumption 2. *Workers have enough confidence to make a final decision after acquiring useful hints, i.e.,* $T \in (\frac{5}{8}, 1)$ *in the hint state.*

Note that the confidence of a random guess is $T = \frac{1}{2}$, thus $T > \frac{5}{8}$ means that the worker's confidence to pick up an answer is high after looking at the hint. This value (5/8) is related to the proof of Corollary 1. As an illustration, let us see Figure 1.1(a). Workers outside Australia may not know which one is the Sydney Harbour Bridge. However, after reading the hints (grey) in Figure 1.1(b), workers should have enough confidence to make a final decision "A" as the pylons structure is very obvious. When T approaches to 1, the beliefs from the hint are maximal, or equivalently, the hint provides the worker with a certain answer.

2.1.1.3 Payment Mechanism

According to the model assumption, we are ready to introduce our payment mechanism based on the hybrid-stage setting. Specifically, after the worker answers all N questions in the hybrid-stage setting, his/her performance is evaluated by his/her responses to G ($\leq N$) questions. Namely, his/her choice for each question in the gold standard gets evaluated to one of four states, denoted by $\{\mathbb{D}_+, \mathbb{D}_-, \mathbb{H}_+, \mathbb{H}_-\}$. We define the four states as follows.

- $\mathbb{D}_+$: answer in the main stage and correct;
- $\mathbb{D}_-$: answer in the main stage and incorrect;
- $\mathbb{H}_+$: answer in the hint stage and correct;
- $\mathbb{H}_-$: answer in the hint stage and incorrect.

Note that "answer in the main stage" means that he/she feels confident in the main stage and answers directly; "answer in the hint stage" means that he/she feels unsure in the main stage and answers with hints in the hint stage. "correct" or "incorrect" denotes whether the worker's selection matched with the standard answer in G questions or not.

Therefore, under the hybrid-stage setting, we can formulate any payment mechanism as function

$$f : \{\mathbb{D}_+, \mathbb{D}_-, \mathbb{H}_+, \mathbb{H}_-\}^G \to [\mu_{\min}, \mu_{\max}], \tag{2.1}$$

where $\min f(\cdot) = \mu_{\min}$ and $\max f(\cdot) = \mu_{\max}$. We reserve the rights to set $\mu_{\min}$ and $\mu_{\max}$, where $0 \leq \mu_{\min} \leq \mu_{\max}$. In this chapter, the goal is to design f such that its expected payment for each worker is strictly maximized under the above setting.

2.1.1.4 Difference from Previous Approaches

The most related approach to ours is the self-corrected approach [Shah and Zhou, 2016], since both of us have two phases in the setting. However, they are totally different in probabilistic modeling. The self-corrected approach builds up the two-stage setting, and workers are necessarily required to enter the second stage to check the reference answer of every question, whereas, our approach builds up the hybrid-stage setting, and workers are not necessary to enter the hint stage related to confident questions. Besides, since each payment mechanism is customized based on some designed goals (examples are in Section 2.2.1) under its corresponding setting, our hint-guided payment mechanism is also different from the one used in the self-corrected approach.

It would be also interesting to discuss the advantages of the proposed approach over active learning [Yan et al., 2011] for crowdsourcing. There are two points to be highlighted. First, compared with active learning, hints in our approach may not be as strong as querying the ground-truth label; the hint only guides the worker to make a choice. Second, active learning is constrained to query which data sample should be labeled next and which annotator should be queried to benefit the learning model. However, our approach is free of these restrictions.

2.1.2 General Rules of Hints

Motivated by instructional hints in the educational psychology [Koedinger and Aleven, 2007], to make the hints useful and reduce interface designers' workloads, we offer three general rules here:

(A). The hints should be easily accessible to interface designers;

(B). The hints should be discriminative and concise for workers; and

(C). The hints should be irrelevant to the number of annotated samples in each task.

We adopt the three rules in designing our hints in experiments. We take three practical datasets in our experimental setup (Section 2.3.1) to justify that these requirements are reasonable in the real world. First, for *Sydney Bridge*, as an interface designer, we easily acquire the content of hints from Wikipedia, which includes discriminative and concise phrases, such as "concrete pylons" and "around Sydney Opera House". Second, for *Stanford Dogs*, we build a lookup table as hints, which includes the characteristics of four breeds of dogs, such as "prick ears" for Norwich Terrier. It means that, the hints in this dataset should be irrelevant to the number of annotated samples, but relevant to the number of classes. Third, for *Speech Clips*, the tool is freely available online to roughly recognize each speech clip and save the concise keywords (≤ 4) as the hints.

2.1.3 Needs of Hybrid-stage Setting

It is also noted that, in designing the pre-processing mechanism, the high-quality worker detection is very important for collecting a sufficient number of high-quality labels. If the tasks can be assigned to each worker by his/her work quality, the annotation quality will be increased accordingly. Also, if we can detect the high-quality workers and give more weights on his/her annotations, we can acquire the better label aggregation. Here, we show that it may not be achieved by a single-stage setting with hints (i.e., only Figure 1.1(b) and no Figure 1.1(a)). Later, we also empirically demonstrate this point in Section 2.3.3.3.

Specifically, by our Assumption 2, if we want to collect more correct labels, it is more naturally to directly assign visible hints for every single question. This removes the necessity to have a hybird-stage setting as we design here. However, high-quality workers are always preferred by crowdsourcing platforms, thus they should be identified and more paid. Such a fundamental goal may not be achieved by a simple single-stage setting with visible hints. The reason is explained as follows. Under the single-stage setting, both high-quality and low-quality workers can easily read the visible hints to answer questions. Thus, we cannot make a difference between them. However, under the hybrid-stage setting, high-quality workers may not read the hints frequently. Namely, the less number of times workers enter the hint stage, the higher quality they are estimated to be. Thus, we can track the high-quality workers by our setting.

Note that, only this setting may encounter a problem: if the hints are freely available in the hint stage, by Assumption 1 (B), even high-quality workers may abuse free hints for higher accuracy and rewards. This issue causes failure in the detection of high-quality workers by the hybrid-stage setting. Therefore, under the hybrid-stage setting, we hope to develop a payment mechanism (Section 2.2), which incentivizes workers to use the hints properly. Specifically, this mechanism penalizes workers who use the hints. Then, high-quality workers will answer most of the questions directly for higher rewards. As a result, this mechanism helps our setting to detect the high-quality workers effectively.

2.2 Hint-guided Payment Mechanism

In Section 2.2.1, we first give two important definitions which help us to design a payment function. Then, the designed payment function is given in Section 2.2.2. Furthermore, we prove that our incentive-compatible payment mechanism is also unique under the hybrid-stage setting. Finally, in Section 2.2.3, we clarify that more restrictive designing goals cannot be realized here.

2.2.1 Design Principles

Incentive compatibility (Definition 2.1) and mild no-free-lunch axiom (Definition 2.2) are important to design a payment mechanism for pre-processed approaches, which are also popularly used by previous works [Shah and Zhou, 2015, 2016].

Definition 2.1 (Incentive Compatibility)**.** *A payment mechanism f is incentive-compatible only if the following two conditions are satisfied: (i) f gives an incentive to a worker to choose all answers by his/her belief; (ii) The expected payment, from the worker's belief, is strictly maximized in both the main stage and the hint stage.*

Definition 2.1, which is adapted from the standard game theoretical assumption [Nisan et al., 2007], describes incentive compatibility. Basically, it means that f should encourage a worker to select the option he/she believes most likely to be correct.

Definition 2.2 (Mild No-free-lunch Axiom)**.** *If all answers attempted by a worker in "gold standard" questions are either wrong or based on hints, then the payment for the worker should be zero, unless all answers attempted by the worker are correct. More formally, $f(\mathbf{a}) = 0, \forall \mathbf{a} \in \{\mathbb{D}_-, \mathbb{H}_+, \mathbb{H}_-\}^G \backslash \{\mathbb{H}_+\}^G$.*

Definition 2.2 is a variant of the no-free-lunch axioms for our hybrid-stage setting. It requires that f should not pay a worker who has bad performance on "gold standard" questions. This helps to reject spammers and keep high-quality workers, since answers to these questions are known to the platform and spammers are likely to give wrong answers while high-quality workers are not.

Our aim is to design the payment mechanism f, which is defined in Eq. (2.1), simultaneously satisfies the above definitions.

2.2.2 Proposed Payment Mechanism

In order to design a payment mechanism, we first consider the easiest case, i.e., for a single question, how the worker should get paid under our hybrid-stage setting. This helps us to find specific rules under Definition 2.1 for our setting under Assumption 1, such rules are given in Proposition 1 below with its proof.

Proposition 1. *Let $f : \{\mathbb{D}_+, \mathbb{D}_-, \mathbb{H}_+, \mathbb{H}_-\} \to [0, \mu_{max}]$, $d_+ = f(\mathbb{D}_+)$, $d_- = f(\mathbb{D}_-)$, $h_+ = f(\mathbb{H}_+)$ and $h_- = f(\mathbb{H}_-)$. When $N = G = 1$, f satisfies Definition 2.1 if it meets the following pricing constraints:*

(A). $d_+ > d_-, h_+ > h_-, d_+ > h_+$.

(B). $\frac{d_+ - d_-}{1-2\epsilon} \geq \frac{h_+ - h_-}{2\epsilon}$.

(C). $d_+ - d_- \leq \frac{2T-1}{1/2-\epsilon}(h_+ - h_-)$.

Proof. **The justification of the first pricing constraint:** under the hybrid-stage setting, the reasonable payment mechanism should consider two facts: 1) the payment for correct answer should be much higher than the payment for incorrect answer. Namely, $d_+ > d_-, h_+ > h_-$. 2) If the answer is correct, the payment to the worker who answers directly should be higher than the payment to the worker who answers using hints. Namely, $d_+ > h_+$. Why this condition penalizes workers who use the hints? The reason is that: $d_+ > h_+$ ensures that high-quality workers will answer most of questions directly for higher rewards. Under the proposed setting, whether using hints can be taken as a criterion to detect the high-quality workers. Thus, $d_+ > h_+$ assists the hybrid-stage setting to detect the high-quality workers.

The justification of the second pricing constraint: for each question, $d_+ - d_-$ is the income gap for a worker who answers directly, and $h_+ - h_-$ is the income gap for a worker who answers with hints. In order to encourage the worker to use the hints properly, we consider bridge the per unit of income gap $\frac{d_+ - d_-}{1-2\epsilon}$ in the main stage and $\frac{h_+ - h_-}{2\epsilon}$ in the hint stage together. Namely, we impose the condition $\frac{d_+ - d_-}{1-2\epsilon} \geq \frac{h_+ - h_-}{2\epsilon}$ into the incentive-compatible payment mechanism, which encourages his/her to directly answer questions that he/she feels confident about. In other words, the worker should not abuse the hints.

The justification of the third pricing constraint: the third condition is the complementary condition for the second one. It should incentivize the worker to leverage the hints before answering questions that he/she feels unsure of. When $P_A > \frac{1}{2} - \epsilon$ while $P_{A|H} > T$, we prefer to choose "A". If the worker selects "A" by his/her own belief, then his/her expected payment is

$$Payment(A) = (\frac{1}{2} - \epsilon)(d_+ P_A + d_- P_B) + 2\epsilon(h_+ P_{A|H} + h_- P_{B|H}).$$

When $P_A < \frac{1}{2} + \epsilon$ while $P_{B|H} > T$, we prefer to choose "B". If the worker selects "B" by his/her own belief, then his/her expected payment is

$$Payment(B) = (\frac{1}{2} - \epsilon)(d_+ P_B + d_- P_A) + 2\epsilon(h_+ P_{B|H} + h_- P_{A|H}).$$

If a payment mechanism is incentive-compatible, the worker is incentivized to choose the answer by his/her own belief while the according payment is strictly maximized. In this case, when $P_A > \frac{1}{2} - \epsilon$ while $P_{A|H} > T$, $Payment(A) > Payment(B)$. When $P_A < \frac{1}{2} + \epsilon$ while $P_{B|H} > T$, $Payment(A) < Payment(B)$. Let us infer the case of $Payment(A) > Payment(B)$, the other case gives the same result by symmetry.

$$\begin{aligned}(\frac{1}{2} - \epsilon)(d_+ P_A + d_- P_B) + 2\epsilon(h_+ P_{A|H} + h_- P_{B|H}) \\ > (\frac{1}{2} - \epsilon)(d_+ P_B + d_- P_A) + 2\epsilon(h_+ P_{B|H} + h_- P_{A|H}).\end{aligned}$$

Due to the facts that $P_A = 1 - P_B$ and $P_{A|H} = 1 - P_{B|H}$,

$$(\frac{1}{2} - \epsilon)(d_+(2P_A - 1) + d_-(1 - 2P_A)) + 2\epsilon(h_+(2P_{A|H} - 1) + h_-(1 - 2P_{A|H})) > 0.$$
$$(\frac{1}{2} - \epsilon)(2P_A - 1)(d_+ - d_-) + 2\epsilon(2P_{A|H} - 1)(h_+ - h_-) > 0. \tag{2.2}$$

Due to $P_A > \frac{1}{2} - \epsilon$ and $P_{A|H} > T$, then we have $2P_A - 1 > -2\epsilon$ and $2P_{A|H} - 1 > 2T - 1$. According to Eq. (2.2), we have,

$$(\frac{1}{2} - \epsilon)(2P_A - 1)(d_+ - d_-) > -2\epsilon(2P_{A|H} - 1)(h_+ - h_-). \tag{2.3}$$

For Eq. (2.3), the term in the left hand side should always be larger than the term in the right hand side for the same ϵ. Hence, the "Infimum" value of the left hand side should be always larger than the "Supremum" value of the right hand side.

$$\inf_{P_A}\{(\frac{1}{2} - \epsilon)(2P_A - 1)(d_+ - d_-)\} \geq \sup_{P_{A|H}}\{-2\epsilon(2P_{A|H} - 1)(h_+ - h_-)\}.$$

Therefore, we have

$$(\frac{1}{2} - \epsilon)(-2\epsilon)(d_+ - d_-) \geq -2\epsilon(2T - 1)(h_+ - h_-). \tag{2.4}$$

Due to $-2\epsilon < 0$, therefore, we eliminate the same negative parameter in both ends of the Eq. (2.4), then we have,

$$(\frac{1}{2} - \epsilon)(d_+ - d_-) \leq (2T - 1)(h_+ - h_-).$$

Finally, due to $\epsilon \in [0, \frac{1}{2})$, we have the condition as follows,

$$(d_+ - d_-) \leq \frac{2T - 1}{1/2 - \epsilon}(h_+ - h_-).$$

□

Condition (A) highlights that, for each question, the payment h_+ from an indirect correct answer (after reading hints) should be less than d_+ from a direct correct answer. Condition (B) bridges the per unit income gap $\frac{d_+ - d_-}{1-2\epsilon}$ in the main stage and $\frac{h_+ - h_-}{2\epsilon}$ in the hint stage together, and the inequality encourages a worker to directly answer questions that he/she feels confident about in the main-stage. Condition (C) incentivizes his/her to leverage the hints before answering questions that he/she is unsure of. Thus, conditions (A) and (B) encourage workers to directly answer questions without hints if they are confident enough; and when a worker has really low confidence, condition (C) encourages him/her to use hints.

Remark 2. *In condition (A), we cannot set $d_+ = h_+$. If $d_+ = h_+$, even high-quality workers may abuse hints for higher accuracy and rewards. This issue fails the detection of high-quality workers by the hybrid-stage setting. Therefore, $d_+ > h_+$ ensures that high-quality workers will answer most of the questions directly for higher rewards. Under the hybrid-stage setting, whether hints are used can be taken as a criterion to detect the high-quality workers. Then, $d_+ > h_+$ assists our setting to detect the high-quality workers, which has been verified in experiments in Section 2.3.3.3.*

From Proposition 1, we can see that f relies on workers' uncertainty degree ϵ in the main stage and their confidence T in the hint stage. When ϵ is set to a large value, more workers need hints to make their decision for each question. The disadvantage of large ϵ is that the overall payments for workers may be low due to leveraging too many hints. When ϵ is set to a small value, fewer workers need hints to make their decision for each question. The disadvantage of small ϵ is that the quality of crowdsourced labels may be poor since more workers avoid hints for higher payments. Thus, we need to find ϵ to achieve a good tradeoff such that most workers are balanced, neither too cautious nor too careless.

However, Proposition 1 only makes use of Assumption 1 to find rules for f and does not specify the relationship between ϵ and T. Below Proposition 2 helps to connect ϵ and T, and shows a lower-bound of ϵ.

Proposition 2. *Under Assumption 1, f satisfies both Definitions 2.1 and 2.2 if $\epsilon \in [\epsilon_{\min}, 1/2)$ where $\epsilon_{\min} = T - \sqrt{T^2 - 1/4}$.*

Proof. When we consider the last two pricing constraints in Proposition 1 under the "mild no-free-lunch Axiom" in Definition 2.2, we can see that $N = G = 1$ and $d_- = h_- = 0$. Therefore, we have,

$$\frac{1-2\epsilon}{2\epsilon} \leq \frac{d_+}{h_+} \leq \frac{2T-1}{\frac{1}{2}-\epsilon}.$$

as $\epsilon \in [0, \frac{1}{2})$, then we have

$$2(\frac{1}{2}-\epsilon) \leq (2T-1)2\epsilon.$$

$$\epsilon^2 - 2T\epsilon + \frac{1}{4} \leq 0.$$

$$\left(\epsilon - \frac{2T+\sqrt{4T^2-1}}{2}\right)\left(\epsilon - \frac{2T-\sqrt{4T^2-1}}{2}\right) \leq 0.$$

Thus the feasible region for ϵ is

$$[0, \frac{1}{2}) \cap \left[T - \sqrt{T^2 - \frac{1}{4}},\ T + \sqrt{T^2 - \frac{1}{4}}\right].$$

In addition, due to $T \in (\frac{1}{2}, 1)$ and $T + \sqrt{T^2 - \frac{1}{4}}$ increases monotonically with T, then

$$\min_T \left(T + \sqrt{T^2 - \frac{1}{4}} \right) > \left(T + \sqrt{T^2 - \frac{1}{4}} \right) |_{T=\frac{1}{2}} = \frac{1}{2}.$$

Similarly, $T - \sqrt{T^2 - \frac{1}{4}} = \frac{\frac{1}{4}}{T + \sqrt{T^2 - \frac{1}{4}}}$ decreases monotonically with T, then

$$\max_T \left(T - \sqrt{T^2 - \frac{1}{4}} \right) < \left(T - \sqrt{T^2 - \frac{1}{4}} \right) |_{T=\frac{1}{2}} = \frac{1}{2}.$$

and

$$\min_T \left(T - \sqrt{T^2 - \frac{1}{4}} \right) > \left(T - \sqrt{T^2 - \frac{1}{4}} \right) |_{T=1} = 1 - \frac{\sqrt{3}}{2} > 0.$$

To sum up, ϵ satisfies

$$\epsilon \in \left[T - \sqrt{T^2 - \frac{1}{4}},\ \frac{1}{2} \right).$$

Therefore, for a fixed $T \in (\frac{1}{2}, 1)$, the minimum ϵ for a incentive-compatible payment mechanism should be $\epsilon_{\min} = T - \sqrt{T^2 - \frac{1}{4}}$. For the case of $1 \leq G \leq N$, we have the same result because, for a random question to be a gold standard question, we get the minimum value $\epsilon_{\min}$, which is the lower bound of ϵ. Due to previous assumptions that each question is independent and all questions share the same ϵ in system design, $\epsilon_{\min}$ will be the most suitable value to cover all cases. This completes the proof. □

Moreover, based on above Proposition, we can derived following Corollary which is based on Assumption 2.

Corollary 1. *Under Assumption 2,* $(1/2 - \epsilon_{min}) < (2T - 1)$.

Proof. Under Assumption 2, we have $T \in (5/8, 1)$. Therefore, we can build up the above inequality,

$$(8T - 5)(T - 2/4) > 0.$$

Namely,

$$8T^2 - 9T + 10/4 > 0.$$

Then we have,

$$(T - 1/2)(T + 1/2) < (3T - 3/2)^2,$$

which equals to

$$3/2 + \sqrt{T^2 - 1/4} < 3T.$$

Therefore, we have

$$1/2 - T + \sqrt{T^2 - 1/4} < 2T - 1.$$

According to Proposition 2 ($\epsilon_{min} = T - \sqrt{T^2 - 1/4}$), we have $(1/2 - \epsilon_{min}) < (2T - 1)$. □

Finally, we show when $\epsilon = \epsilon_{\min}$, i.e., the boundary condition in Proposition 2 is achieved, a hint-guided payment mechanism f can be designed (Algorithm 1). The function g, which sets how a single question should be paid, is defined at step 1 in Algorithm 1. Note that $g(\mathbb{H}_+) < g(\mathbb{D}_+) = 1$ due to Corollary 1, which is also in consistent with condition (a) in Proposition 1. Responses from workers on "gold standard" questions are collected in step 2, and the budget is set in step 3. A multiplicative form of g is adopted in step 4, which is inspired by Shah and Zhou [2015]. It incentivizes workers to use hints properly and also helps to make the smallest payment to spammers. The reasons are highlighted in Remark 3.

Algorithm 1 Hint-guided Payment Mechanism

Inputs:

1. Define a function $g : \{\mathbb{D}_+, \mathbb{D}_-, \mathbb{H}_+, \mathbb{H}_-\} \rightarrow \mathbb{R}_+$, which sets how a single question should be paid, and , where $g(\mathbb{D}_+) = 1$, $g(\mathbb{D}_-) = 0$, $g(\mathbb{H}_+) = \frac{1/2-\epsilon_{min}}{(2T-1)}$ and $g(\mathbb{H}_-) = 0$;

2. $a_1, \ldots, a_G \in \{\mathbb{D}_+, \mathbb{D}_-, \mathbb{H}_+, \mathbb{H}_-\}$ are the state evaluations of the answers to the G gold standard questions;

3. Set the minimum payment $\mu_{\min}$ and maximum payment $\mu_{\max}$ properly;

The payment is:

4. $f([a_1, \ldots, a_G]) = \beta \prod_{i=1}^{G} g(a_i) + \mu_{\min}$ where $\beta = \mu_{\max} - \mu_{\min}$.

Remark 3. *The benefits of using the multiplicative form is detailed as follows. For example, a spammer will respond to a question with an arbitrary answer, thus he/she will get the minimum payment once any answers in "gold standard" are wrong. Then, for a normal worker, if he/she tries to get the highest payment, he/she is encouraged to use hints as less as possible. The reason is that the payment for a correct answer after using hints is $g(\mathbb{H}_+)$ which is smaller than 1, i.e., $g(\mathbb{D}_+)$ (Corollary 1). Thus, more hints are used, the maximum payment for a worker will get smaller. Besides, such a multiplicative form also helps us to identify and pay more for high-quality workers, as those workers will naturally user less hints.*

The design of Algorithm 1 is further supported by the following Theorem 2.3. Thus, our algorithm is the unique one to satisfy both Definitions 2.1 and 2.2, and $\epsilon = \epsilon_{\min}$ is also a must choice here. Note that, in practice, the algorithm makes the minimum payment $\mu_{\min}$ instead of 0 in Definition 2.2, if one or more attempted answers in the gold standard are wrong. This operation is without any loss of generality.

Theorem 2.3. *Under Assumption 1 and 2, f in Algorithm 1 satisfies both Definitions 2.1 and 2.2 if and only if $\epsilon = \epsilon_{\min}$.*

Proof. To prove "if and only if", the standard way is to prove the existence first, and then prove the uniqueness.

Existence: To consider the case of $N = G = 1$, when $\epsilon = \epsilon_{\min}$, the proposed payment mechanism meets three pricing constraints, therefore, the proposed payment mechanism is incentive-compatible. For the case of $1 \leq G \leq N$, $a_1, \cdots, a_G \in \{\mathbb{D}_+, \mathbb{D}_-, \mathbb{H}_+, \mathbb{H}_-\}$ are the evaluations of the answers to the G gold standard questions. The final payment is $\beta \prod_{i=1}^{G} f(a_i) + \mu_{\min}$. Due to the assumption that each $a_i, i \in \{1, \cdots, G\}$ is independent, the overall expected payment $Payment(a_1, \cdots, a_G)$ equals to the product (scaled by β; shifted by $\mu_{\min}$) of the expected values $f(a_i)$ for each $a_i, i \in \{1, \cdots, G\}$. As $f(a_i)$ is maximized when the worker answers the question by his/her own belief, the overall expected payment is then maximized when all workers answer the questions by their own beliefs. This proves the existence.

Uniqueness: To consider the case of $N = G = 1$, according to the "mild no-free-lunch" axiom, both d_- and h_- are equal to 0. Furthermore, according to three pricing constraints, when $\epsilon = \epsilon_{\min}$, we get an equality case of 2.2.2, namely, $\frac{1-2\epsilon}{2\epsilon} = \frac{d_+}{h_+} = \frac{2T-1}{\frac{1}{2}-\epsilon}$, hence, the relation between d_+ and h_+ is fixed. The derived mechanism is identical to the hint-guided payment mechanism. Therefore, we further consider whether the derived mechanism is identical to the hint-guided payment mechanism for the general case of $1 \leq G \leq N$.

The base case of the induction hypothesis is that the derived mechanism is identical to the hint-guided payment mechanism whenever $\mathbf{y} \in \{\mathbb{H}_+, \mathbb{H}_-\}^G \backslash \{\mathbb{H}_+\}^G$. For G gold standard questions, we assume that the worker answers at least $G-r-1$ questions with hints, namely, $\sum_{i=1}^{G} \mathbf{1}\{y_i \in \{\mathbb{H}_+, \mathbb{H}_-\}\} \geq G - r - 1$ [1]. We suppose that the induction hypothesis is true whenever $\mathbf{y} \in \{\mathbb{D}_+, \mathbb{D}_-, \mathbb{H}_+, \mathbb{H}_-\}^G \backslash \{\mathbb{D}_+, \mathbb{H}_+\}^G$. Now we prove the induction hypothesis keeps true whenever $\mathbf{y} \in \{\mathbb{D}_+, \mathbb{D}_-, \mathbb{H}_+, \mathbb{H}_-\}^G \backslash \{\mathbb{D}_+, \mathbb{H}_+\}^G$ and $\sum_{i=1}^{G} \mathbf{1}\{y_i \in \{\mathbb{H}_+, \mathbb{H}_-\}\} = G - r$.

Let y_i denote the state evaluation of his/her answer to the question $i \in \{1, \cdots, G\}$. Assume that $y_1, \cdots, y_{r-1} \in \{\mathbb{D}_+, \mathbb{D}_-\}$ and $y_{r+1}, \cdots, y_G \in \{\mathbb{H}_+, \mathbb{H}_-\}$. For N questions, suppose for $i \in \{1, \cdots, r-1\}$, we have $P_A > \frac{1}{2} + \epsilon$; for $i \in \{r+1, \cdots, N\}$, we have $\frac{1}{2} - \epsilon < P_A < \frac{1}{2} + \epsilon$ while $P_{A|H} > T$. Thus, he/she will select "A" for

[1] $\mathbf{1}\{x\}$ is an indicator function, and $\mathbf{1}\{x = true\} = 1$ while $\mathbf{1}\{x = false\} = 0$.

all questions $\{1, \cdots, N\} \backslash \{r\}$. Moreover, the worker holds a belief that questions $1, \cdots, r-1$ can be selected directly; while questions $r+1, \cdots, G$ will be answered with hints.

Assume that $q : \{\mathbb{D}_+, \mathbb{D}_-, \mathbb{H}_+, \mathbb{H}_-\} \to [0, \mu_{\max}]$ be a function defined as follows. $q(y_r)$ is an expected payment conditioned on the rth question, which is composed of a convex combination of two parts. The first part is the payment that rth question is in the gold standard question; the second part is the payment that rth question is not in the gold standard question. Hence, $q(y_r) = \phi q^*(y_r) + (1-\phi)c$, where $\phi \in (0,1), c \geq 0$. q^* denote the first part payment, which is dependent on $q(y_r)$.

Assume that the function q^* is a convex combination of the payment function f evaluated at various points. Due to "mild no-free-lunch" axiom, we have $q^*(y_r) = 0$ when $y_r \in \{\mathbb{H}_-\}$. Let $P_B = 1 - P_A$; $P_{B|H} = 1 - P_{A|H}$ be the worker's confidence for the question r, if the payment mechanism incentivizes the worker to select the answer for question r properly, then the result should be:

If $P_A > \frac{1}{2} - \epsilon$ and $P_{A|H} > T$, then the answer to question r is A, hence, the expected payment by A should be larger than the expected payment by B.

$$\begin{aligned}(\frac{1}{2} - \epsilon)(P_A q^*(\mathbb{D}_+) + P_B q^*(\mathbb{D}_-)) + 2\epsilon(P_{A|H} q^*(\mathbb{H}_+) + P_{B|H} q^*(\mathbb{H}_-)) > \\ (\frac{1}{2} - \epsilon)(P_B q^*(\mathbb{D}_+) + P_A q^*(\mathbb{D}_-)) + 2\epsilon(P_{B|H} q^*(\mathbb{H}_+) + P_{A|H} q^*(\mathbb{H}_-)).\end{aligned}$$

In turn, if $P_A < \frac{1}{2} + \epsilon$ and $P_{B|H} > T$, then the answer to question r is B, hence, the expected payment by B should be larger than the expected payment by A.

$$\begin{aligned}(\frac{1}{2} - \epsilon)(P_A q^*(\mathbb{D}_+) + P_B q^*(\mathbb{D}_-)) + 2\epsilon(P_{A|H} q^*(\mathbb{H}_+) + P_{B|H} q^*(\mathbb{H}_-)) < \\ (\frac{1}{2} - \epsilon)(P_B q^*(\mathbb{D}_+) + P_A q^*(\mathbb{D}_-)) + 2\epsilon(P_{B|H} q^*(\mathbb{H}_+) + P_{A|H} q^*(\mathbb{H}_-)).\end{aligned}$$

Let $P_A = \frac{1}{2} + \epsilon$ and $P_{B|H} = T$, then we have,

$$\begin{aligned}(\frac{1}{2} - \epsilon)(\frac{1}{2} + \epsilon) q^*(\mathbb{D}_+) + (\frac{1}{2} - \epsilon)^2 q^*(\mathbb{D}_-) + 2\epsilon(1-T) q^*(\mathbb{H}_+) + 2\epsilon T q^*(\mathbb{H}_-) \leq \\ (\frac{1}{2} - \epsilon)^2 q^*(\mathbb{D}_+) + (\frac{1}{2} - \epsilon)(\frac{1}{2} + \epsilon) q^*(\mathbb{D}_-) + 2\epsilon T q^*(\mathbb{H}_+) + 2\epsilon(1-T) q^*(\mathbb{H}_-).\end{aligned}$$

Due to $q^*(\mathbb{H}_-) = 0$, we have,

$$\begin{aligned}(\frac{1}{2} - \epsilon)(\frac{1}{2} + \epsilon) q^*(\mathbb{D}_+) + (\frac{1}{2} - \epsilon)^2 q^*(\mathbb{D}_-) + 2\epsilon(1-T) q^*(\mathbb{H}_+) \leq \\ (\frac{1}{2} - \epsilon)^2 q^*(\mathbb{D}_+) + (\frac{1}{2} - \epsilon)(\frac{1}{2} + \epsilon) q^*(\mathbb{D}_-) + 2\epsilon T q^*(\mathbb{H}_+).\end{aligned}$$

After simplifying, we have,

$$(\frac{1}{2}-\epsilon)2\epsilon q^*(\mathbb{D}_+)-(\frac{1}{2}-\epsilon)2\epsilon q^*(\mathbb{D}_-)\leq 2\epsilon(2T-1)q^*(\mathbb{H}_+).$$
$$q^*(\mathbb{D}_+)-q^*(\mathbb{D}_-)\leq \frac{2T-1}{\frac{1}{2}-\epsilon}q^*(\mathbb{H}_+).$$

When designing a payment mechanism, we assume that $q^*(\mathbb{D}_+)$ is fixed, and $q^*(\mathbb{H}_+)$ can be derived by the relation with $q^*(\mathbb{D}_+)$. We hope to set $q^*(\mathbb{H}_+)$ to the boundary value. Specifically, as we want to penalize the use of hints, we set $q^*(\mathbb{H}_+)$ as small as possible while meeting the inequality. Therefore, we set $q^*(\mathbb{D}_+)$ to the lower bound. Namely, $q^*(\mathbb{H}_+)=\frac{\frac{1}{2}-\epsilon}{2T-1}(q^*(\mathbb{D}_+)-q^*(\mathbb{D}_-))$. Due to "mild no-free-lunch" axiom, $q^*(\mathbb{H}_+)=\frac{\frac{1}{2}-\epsilon}{2T-1}q^*(\mathbb{D}_+)$.

Since q^* is a convex combination of the payment function f evaluated at various points. Therefore, we have,

$$(2T-1)f(\mathbb{H}_+,y_2,\cdots,y_G)=(\frac{1}{2}-\epsilon)f(\mathbb{D}_+,y_2,\cdots,y_G),$$

where the augments above hold for any permutation of the G gold standard questions.

$$(2T-1)f(y_1,y_2,\cdots,\mathbb{H}_+^i,\cdots,y_G)=(\frac{1}{2}-\epsilon)f(y_1,y_2,\cdots,\mathbb{D}_+^i,\cdots,y_G),$$

where the notation $\mathbb{H}_+^i$ denotes the state evaluation of y_i is $\mathbb{H}_+$. The same is true for $\mathbb{D}_+^i$. Then, according to the recursive induction, we have

$$f(\overbrace{\mathbb{D}_+,\cdots,\mathbb{D}_+}^{G})=\left(\frac{2T-1}{\frac{1}{2}-\epsilon}\right)^G f(\overbrace{\mathbb{H}_+,\cdots,\mathbb{H}_+}^{G}). \tag{2.5}$$

Based on Eq. (2.5) and using the fact that all arguments apply to any permutation of the G gold standard questions, we can see that f should be identical to the hint-guided payment mechanism. This proves the uniqueness. □

2.2.3 No Other Compatible Mechanism

Definition 2.1 is a must to design a payment mechanism. However, under our hybird-setting here, there exists another popular "harsh no-free-lunch" axiom (Definition 2.4), which is adapted from Definition 2 in Shah and Zhou [2016].

Definition 2.4 (Harsh No-free-lunch Axiom)**.** *If all answers attempted by the worker in "gold standard" questions are either wrong or based on hints, then the payment for the worker should be zero. More formally,* $f(\mathbf{a})=0$, $\mathbf{a}\in\{\mathbb{D}_-,\mathbb{H}_+,\mathbb{H}_-\}^G$.

Compared to the "mild no-free-lunch" axiom, Definition 2.4 encourages the worker to answer without hints no matter whether he/she is unsure. Thus, it is stronger than the "mild no-free-lunch" axiom and can be used to replace Definition 2.2. We wonder whether we can find another payment function which satisfies this more restrictive condition. However, below Theorem 2.5 shows a contradiction to Definition 2.4.

Theorem 2.5. *Under Assumption 1 and 2, there is no mechanism that satisfies both Definitions 2.1 and 2.4.*

Proof. Under the hybrid-stage setting, the incentive-compatible mechanism encourages the worker to use hints when he/she is unsure of the questions. Since Definition 2.1 is contradictory to Definition 2.4, the "Harsh No-Free-Lunch Axiom" is too strong for the existence of any incentive-compatible payment mechanism. □

Therefore, the "harsh no-free-lunch" axiom is too strong for the existence of any incentive-compatible payment mechanism here. This further illustrates the uniqueness of the proposed payment mechanism.

2.3 Numerical Experiments

We conduct real-world experiments on Amazon Mechanical Turk [1], which is the leading platform to collect crowdsourced labels. We compare our hint-guided approach with: (1) Baseline approach : a single-stage setting with an additive payment mechanism. (2) Skip-based approach [Ding and Zhou, 2017; Shah and Zhou, 2015]: a skip-stage setting with a skip-based payment mechanism. Note that the skip-based payment mechanism is multiplicative. For the self-corrected approach [Shah and Zhou, 2016], it has not been verified on AMT tasks, since there is no criteria how to set references. Therefore, we do not include it in our comparison. Note that additive and multiplicative payment mechanisms are respectively denoted as "+" and "×" for subsequent use in Section 2.3.3.4.

2.3.1 Experimental Setup

All these datasets are collected by us on Amazon MTurk, where hints are easily designed according to the criteria in Section 2.1.2. We conducted three real tasks as follows.

- *Sydney Bridge* (binary-choice questions): we collect 30 images of various bridges. Each image contains one bridge. The task is to identify whether the bridge in each image is the Sydney Bridge. The content of hints includes

[1] https://www.mturk.com/mturk/welcome

discriminative phrases, such as "concrete pylons" and "around Sydney Opera House".

- *Stanford Dogs* (multiple-choice questions): we collect 100 images of four breeds of dogs. The task is to identify the breed of dogs in each image. We build a lookup table as hints, which includes the characteristics of four breeds of dogs, such as "prick ears" for Norwich Terrier.

- *Speech Clips* (subjective questions): we collect 10 speech clips. Each speech clip consists of 1 or 2 short sentences (15 words). The task is to recognize each speech clip and write down the corresponding sentence. We leverage the open tool [1] to roughly recognize each speech clip and save the key words (≤ 4) as the hints.

We verify the effectiveness of our hint-guided approach from three perspectives (Table 2.1), and each perspective includes one to two metrics in brackets: requester ("label quantity" and "label quality"), worker ("worker quality detection" and "spammer prevention") and platform ("money cost"). Except "worker quality detection", other metrics have been popularly used by previous works [Shah and Zhou, 2015, 2016]. They are detailed as follows.

- Label quantity: we evaluate the label quantity by the percentage of the completion of three tasks. In the skip-stage setting, worker yields unlabeled (uncompleted) data by skipping unsure questions. In the single-stage and the hybrid-stage settings, for objective questions, worker yields (few) unlabeled data because he/she forgets or ignores few questions. For subjective questions, worker yields (more) unlabeled data by inputting invalid answers. For example, they write sentences, such as "I do not know" in the answer box.

- Label quality: we evaluate the label quality from two aspects: (i) the percentage of correct answers and incorrect answers on three tasks; and (ii) the error of aggregated labels [Shah and Zhou, 2015]. For the ith question where $i \in \{1, \ldots, n\}$, if there are m_i options after majority voting (the tie situation), and the ground-truth label is one of m_i options, then we consider that $\frac{1}{m_i}$ of the ith question is correct. Therefore, the error of aggregated labels is $1 - (\sum_{i=1}^{n} 1/m_i)/n$. Since text answers cannot be majority voted on *Speech Clips*, we do not report the error of aggregated labels on *Speech Clips*.

- Worker quality detection: we evaluate the worker quality detection of the hint-guided approach implicitly, by the error rate (in %) of aggregating original and rescaled crowdsourced labels. For example, *Sydney Bridge* (origin) means the

[1] https://speech-to-text-demo.mybluemix.net/

original labels collected by our approach. For *Sydney Bridge* (rescale), we rank the worker quality from high to low by the usage frequency of the hints in the collection of original labels. Then, we rescale original labels by adaptive weights. Labels from top 20% (bottom 20%) workers have been empirically rescaled by 1.8 (0.2). The remaining labels keep unchanged. If the error rate on rescaled dataset decreases, then we speculate that our hint-guided approach indeed detects the worker quality. Namely, the less usage of hints indicates the higher quality of the worker.

- Spammer prevention and money cost: we evaluate the spammer prevention and the money cost by the average payment to each worker. Note that the payment consists of two parts: fixed payment and reward payment. Reward payment is based on a worker's responses to G gold standard questions. All payment parameters are in Appendix.

2.3.2 Payment Parameters

Here, we provide the details of parameter setting for different payment mechanisms. The payment is often composed of two parts: a fixed (minimum) payment and a reward (bonus) payment. The fixed payment is paid for each worker who undertakes all tasks, which avoids all multiplicative payment mechanisms too harsh for workers. The reward payment is based on his/her responses to G gold standard questions. For each task, the fixed payment and reward payment are denoted as FP and RP, respectively.

For additive mechanism, k_1 denotes the unit reward for each correct answer. For skip-based mechanism, RP starts from k_2, increasing by P_s for each correct answer. Note that the RP remains the same for skip option, but becomes zero for any incorrect answer. For hint-guided mechanism, RP starts from k_3, increasing by P_d and P_h for each correct answer in the main stage and the hint stage, respectively. However, RP will become zero for any incorrect answer. All payment parameters of this chapter are in Table 2.2. Here, we will explain how we set these parameters as follows.

Table 2.2: The payment parameters. The total payment is composed of FP and RP. RP is based on worker's responses to G questions. $P_h = \frac{\frac{1}{2}-\epsilon}{2T-1}P_d$ according to Algorithm 1, where we set $T = 0.75$ and $\epsilon = 0.191$ due to the Proposition 2.

Data set	FP	G	Baseline	Skip-based		Hint-guided		
			k_1	k_2	P_s	k_3	P_d	P_h
Sydney Bridge	5	3	8.5	9.15	50%	9.15	50%	30%
Stanford Dogs	7	10	9	1.02	60%	1.02	60%	37%
Speech Clips	5	2	20	12.25	100%	12.25	100%	62%

According to the suggestion on Amazon MTurk, the reward per question(denotes as r_a) should be set according to the minimum wage, e.g., a 30 seconds question that

pays 5 cents is a 6 dollars hourly wage. Therefore, r_a are set to 1 cent for each image annotation question and 4.5 cents for each speech recognition question. Moreover, the fixed payment FP are set to 5 cents, 7 cents, 5 cents for Sydney Bridge, Stanford Dogs, and Speech Clips respectively. For different payment mechanisms, the other parameters can be decided as follows:

- Additive payment mechanism: the payment for a strong worker who answers all the G gold questions correctly should be the same as the total payment, thus k_1 satisfies the constrain $FP + G * k_1 = N * r_a$.
- Skip-based payment mechanism (multiplicative): To incentive the worker to provide high-quality labels, the reward per question with the multiplicative payment mechanism (denotes as r_m) should be higher than r_a. Therefore, in our experiments, r_m are set to 1.2 cents for each image annotation question and 5.4 cents for each speech recognition question. Furthermore, the amount payment for a strong worker who answers all the G gold questions correctly under the Skip-based payment mechanism should be the same as the total payment, thus k_2 satisfies the constrain $FP + k_2 * (1 + P_s)^G = N * p_m$, where P_s are set to 50%, 60% and 100% for Sydney Bridge, Stanford Dogs, and Speech Clips respectively.
- Hints-guided payment mechanism (multiplicative): For a strong worker who answers all the G gold questions correctly in the main stage, the reward should be the same as that under the Skip-based payment mechanism, thus parameters k_3 and P_d satisfy $k_3 = k_2$ and $P_d = P_s$ for each task specifically. Moreover, for the reward in the hint stage, we decide P_h according to $P_h = \frac{\frac{1}{2}-\epsilon}{2T-1} P_d$ in Hint-guided payment mechanism, where we set $T = 0.75$ and $\epsilon = 0.191$ according to the equation $\epsilon_{\min} = T - \sqrt{T^2 - \frac{1}{4}}$.

2.3.3 Experimental Results

We demonstrate the effectiveness of our hint-guided approach from the following five aspects. Specifically, Section 2.3.3.1 verifies whether our approach provides a sufficient number of labels. Section 2.3.3.2 displays whether our approach provides high-quality labels. Section 2.3.3.3 denotes whether our approach can detect worker quality. Section 2.3.3.4 indicates whether our approach prevents spammers. Section 2.3.3.5 demonstrates whether our approach saves money.

2.3.3.1 Label Quantity

Table 3.2 denotes the percentage of the completion of three tasks. The first two tasks (*Sydney Bridge* and *Stanford Dogs*) belong to objective questions, while the last task

(*Speech Clips*) belongs to subjective questions. Objective questions can be answered by the random guess. Therefore, the percentage of the completion for objective questions is much higher than that for subjective questions. In addition, the hint-guided approach has a high percentage of the completion of both objective and subjective questions. Our approach inspires workers to finish the questions, ensuring the quantity of crowdsourced labels.

Table 2.3: Evaluation of the label quantity. We provide the percentage of the completion on three tasks.

Data set	Baseline	Skip-based	**Hint-guided**
Sydney Bridge	**100.00%**	74.00%	99.11%
Stanford Dogs	99.72%	58.18%	**99.91%**
Speech Clips	58.33%	30.00%	**75.00%**

2.3.3.2 Label Quality

Figure 2.2 plots the percentage of correct answers and incorrect answers on three tasks. First, on all tasks, the percentage of correct answers in the hint-guided approach is higher than that in the baseline and skip-based approaches. Second, on *Speech Clips*, the percentage of incorrect answers is extremely low in the skip-based approach. The reason is that most people skip difficult speech clips, and answer several easy ones. Third, compared with other approaches, our hint-guided approach ensures a sufficient number of high-quality labels.

Figures 2.3(a) and 2.3(b) plot the error of aggregated labels on the *Sydney Bridge* and *Standford Dogs* tasks. The number of workers (abbreviated as *n_workers*) is set to $\{5, 6, 7, 8, 9, 10\}$, since the error of aggregated labels comes from majority voting among multiple workers [Shah and Zhou, 2015], and the number of multiple workers depends on varying situations. For each of combinations between tasks and *n_workers*, we perform the following actions 200 times repeatedly. In each time, for all questions, we randomly select *n_workers* workers and perform the majority voting on their responses to yield the aggregated labels. The plotted error of aggregated labels is averaged across 200 results. We observe that the hint-guided approach consistently outperforms the baseline and the skip-based approaches, and the performance gap between the baseline and the hint-guided approaches is extremely obvious on *Stanford Dogs*.

2.3.3.3 Worker Quality Dectection

Table 2.4 denotes the error of aggregating original and rescaled crowdsourced labels. For rescaled crowdsourced labels, labels from estimated high-quality workers are

adaptively given more weights, and vice versa. From Table 2.4, we can see the error of aggregating rescaled labels is lower than the error of aggregating original labels. It demonstrates that our hint-guided approach can detect the high-quality workers effectively. Then, the error decreases significantly on *Sydney Bridge*, since the size of *Sydney Bridge* is relatively small (30 questions) compared to *Stanford Dogs* (100 questions). We believe that, the informative hints for *Stanford Dogs* may guide the low-quality workers to make more accurate decisions. Then, the performance gap between high-quality and low-quality workers is insignificant. Therefore, the effect of label rescaling is marginal on this dataset.

Table 2.4: Evaluation of the worker quality detection of the hint-guided approach. Error rate (in %) is provided for aggregating original and rescaled crowdsourced labels.

Number of Workers		5	10
Sydney Bridge	origin	38.33%	16.67%
	rescale	**30.00%**	**11.67%**
Stanford Dogs	origin	12.50%	4.50%
	rescale	**12.00%**	**4.00%**

2.3.3.4 Spammer Prevention

The baseline and hint-guided approaches are represented as Single($+$) and Hybrid($\times$), respectively. We provide one extra interaction: the single-stage setting with the "$\times$" mechanism (Single($\times$)), and all parameters are consistent. Figure 2.4(a) explores how our approach prevents spammers. It plots the average payment to each worker under three approaches. We have one observation: the payments of Single($\times$) and Hybrid($\times$) are lower than that of Single($+$), since an answer in G questions is incorrect, and thus the reward of the "$\times$" mechanism becomes zero. Since spammers answer each question randomly, the "$\times$" mechanism used by our approach makes the smallest payment to them. Thus, our approach prevents spammers.

2.3.3.5 Money Cost

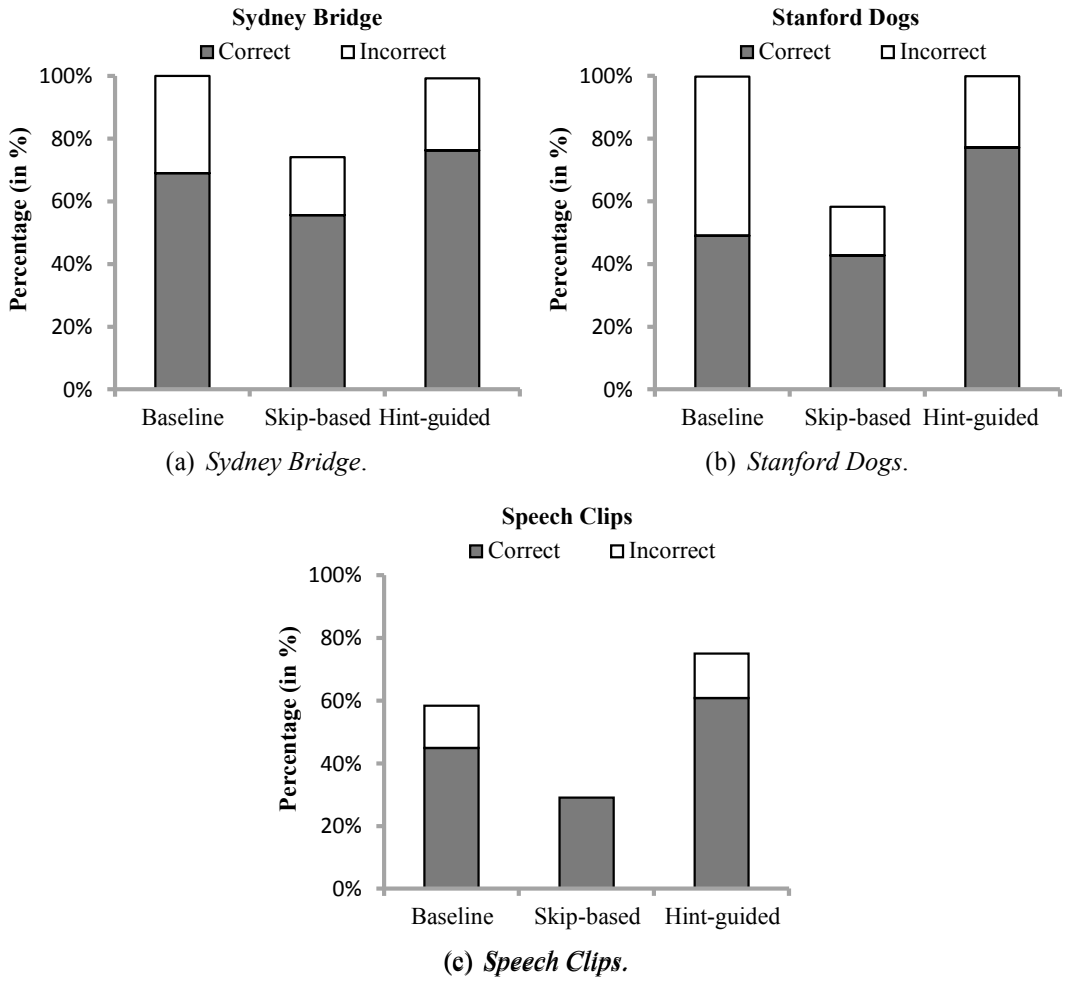
Figure 2.4(b) plots the average payment to each worker under the three approaches. The higher the payment is, the worse the economy of the approach. The payment is calculated as the average of the payments across 200 random selections of G questions. This process mitigates the distortion of results caused by the randomness in the choice of G questions. We can see that, the payments of the skip-based and hint-guided approaches are comparable but less than the payment of the baseline approach, especially in the *Stanford Dogs* task, since both the skip-based and hint-guided approaches use the multiplicative mechanism but the baseline approach use

the additive mechanism. Thus, from the perspective of saving money, we should not employ the baseline approach. Note that, on the *Sydney Bridge* and *Stanford Dogs* tasks, although the payment in the skip-based approach is slightly lower than that in the hint-guided approach, the number of high-quality labels from the hint-guided approach is obviously higher than that from the skip-based approach (Figure 2.3).

2.4 Summary of This Chapter

This chapter has proposed a hint-guided approach that encourages workers to use hints when they answer unsure questions. Our approach consists of the hybrid-stage setting and the hint-guided payment mechanism. We proved the incentive compatibility and uniqueness of our mechanism. Besides, our approach can detect the high-quality workers for more accurate result aggregation. Comprehensive experiments conducted on Amazon MTurk revealed the effectiveness of our approach and validated the simple and practical deployment of our approach.

To sum up, our hints-guided approach provides practitioners an effective way to obtain adequate quantity of high-quality labels within limited budgets, which are critical for the success of many machine learning applications in practice.

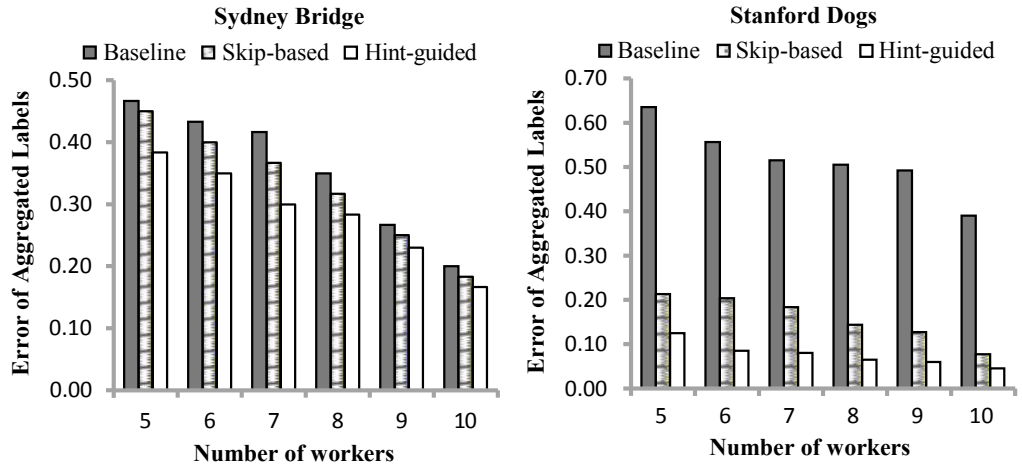


(a) *Sydney Bridge*.

(b) *Stanford Dogs*.

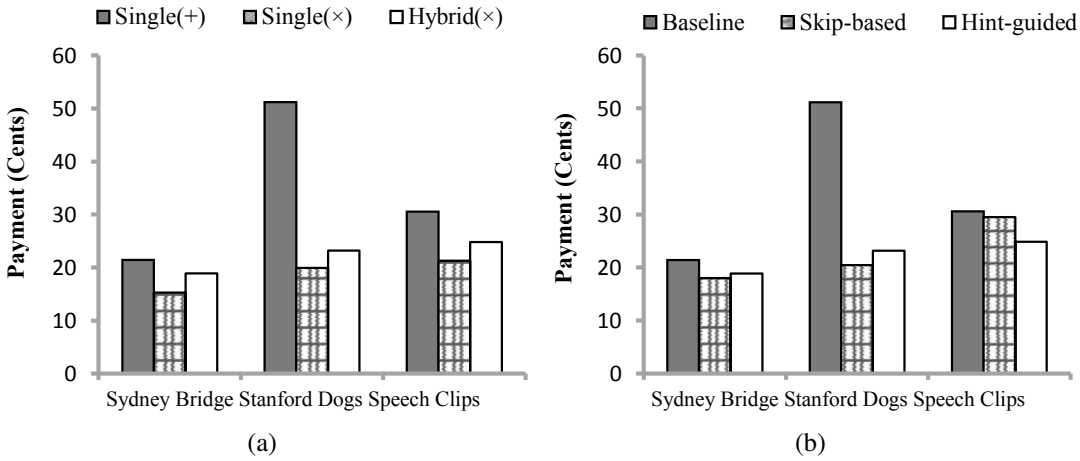
(c) *Speech Clips*.

Figure 2.2: Evaluation of label quality. Percentage (in %) of correct answers and incorrect answers on three tasks are provided. Note that, we do not plot the percentage of unlabeled questions.



(a) *Sydney Bridge*. (b) *Stanford Dogs*.

Figure 2.3: Evaluation of the label quality. Results on *Speech Clips* are not reported, as text answers cannot be majority voted.



(a) (b)

Figure 2.4: Evaluation of the spammer prevention. Average payment to each worker on all three tasks are provided. Evaluation of the money cost.

Chapter 3

POSTAL: Towards Robust Optimization

This chapter answers the second question, namely "how to design the robust optimization under noisy labels?". From the high level, we hope to bridge curriculum learning and stochastic optimization, in order to progressively learn from clean labels to noisy labels.

As mentioned in the Chapter 1, all state-of-the-art stochastic optimizations have a strong assumption that the data is free of noise. This assumption may simplify the theoretical analysis (i.e., convergence rate). However, this assumption restricts their applicabilities to the real-world problem, since the industrial-level datasets are not always perfect. Therefore, how to consider the robustness in stochastic optimization? This question motivates our below research.

We propose a robust SGD mechanism called PrOgressive STochAstic Learning (POSTAL), which naturally integrates the learning regime of curriculum learning (CL) with the update process of vanilla SGD. Our inspiration comes from the progressive learning process of CL, namely learning from "easy" tasks to "complex" tasks. Through the robust learning process of CL, POSTAL aims to yield robust updates of the primal variable on an ordered label sequence, namely from "reliable" labels to "noisy" labels. To realize POSTAL mechanism, we design a cluster of "screening losses", which sorts all labels from the reliable region to the noisy region. We derive the convergence rate of POSTAL realized by screening losses. Meanwhile, we provide the robustness analysis of representative screening losses. Experiments on benchmark datasets show that POSTAL using screening losses is more effective and robust than several existing baselines.

The remainder of this chapter is organized as follows. We first present our proposed progressive stochastic learning (POSTAL) from the mechanism (Section 3.1), the realization by screening losses (Section 3.2), and the theoretical analysis (Section 3.3). Then, in Section 3.4, we conduct sufficient experiments on six UCI simulated datasets

Table 3.1: Comparison of different learning approaches. POSTAL integrates benefits (emphasized by color) of CL and SGD.

Methods	CL	SGD	POSTAL
Learning Process	"Easy" to "Complex"	Random	"Reliable" to "Noisy"
Iterative Training	Heuristic	Gradient-based	Gradient-based

and one AMT crowdsourcing dataset. The conclusive remarks are given in Section 3.5.

3.1 Mechanism of Progressive Stochastic Learning

Real-world applications, such as crowdsourced data, are full of noisy labels. This issue inevitably degenerates the performance of conventional SGD, by adversely affecting updates of the primal variable $\mathbf{w}$ in SGD. Therefore, we hope to explore a mechanism to ensure robust updates of the primal variable $\mathbf{w}$ under noisy settings. Our idea is motivated by curriculum learning (CL), which learns easier tasks first, and learns more difficult tasks gradually to ensure a robust model. This learning strategy is similar to training an infant through to adulthood. In human learning, knowledge is ordered to allow for gradual learning. Infants are provided with easy knowledge first. As they grow and are able to handle more complex concepts, more difficult knowledge will be provided.

Based on the above inspiration, we introduce a robust mechanism called progressive stochastic learning (POSTAL) for noisy labels, which incorporates the mechanism of CL with the update process of vanilla SGD. Since CL provides the ordered learning to learn from easy tasks first then to hard tasks until convergence. Through such a learning paradigm, POSTAL aims to yield robust updates of the primal variable in SGD on an ordered label sequence, namely from "reliable" labels to "noisy" labels. Table 3.1 shows a key comparison of different learning approaches, and POSTAL integrates benefits of CL and SGD together. The following example explains the central idea of POSTAL.

From the left panel of Figure 3.1, we observe that $\mathbf{x}_A$ (point "A") is correctly labeled with $y_A = +1$, which corresponds to its predicted label value (+1) (predicted label value = $\begin{cases} +1 & f_{\mathbf{w}}(\mathbf{x}) \geq 0 \\ -1 & f_{\mathbf{w}}(\mathbf{x}) < 0 \end{cases}$, where $f_{\mathbf{w}}$ denotes the current classifier). Therefore, label y_A of instance $\mathbf{x}_A$ can be regarded as a reliable label. Conversely, instances $\mathbf{x}_B$ and $\mathbf{x}_C$ (i.e., data points "B" and "C") are incorrectly annotated with label $y_B = +1, y_C = +1$. These labels are the opposite of their predicted label value (-1). Therefore, label y_B

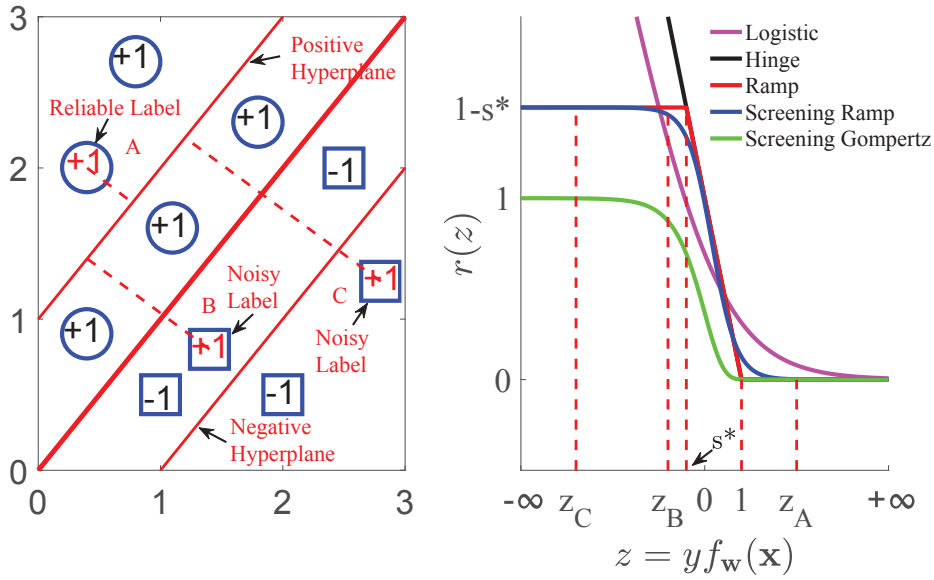


Figure 3.1: **Left Panel:** Circles denote *real positive* instances such as "A". Squares represent *real negative* instances such as "B" and "C"; however, both "B" and "C" are *erroneously* annotated as positive class, which creates two noisy labels. According to their distance to the positive hyperplane, the label of "A" is reliable; while the labels of "B" and "C" are noisy. **Right Panel:** "A" (z_A) is located in the reliable region defined by $z \geq 0$; while "B" (z_B) and "C" (z_C) are located in the noisy region defined by $z \leq 0$. In Definition 3.1, $z = yf_{\mathbf{w}}(\mathbf{x})$ is formally defined as the curriculum in POSTAL. There are two points to be noted that: 1) Ramp Loss is parameterized by s^* directly; and 2) Logistic, Hinge and Ramp Losses are drawn as the baselines of two screening losses.

Algorithm 2 **PrOgressive STochAstic Learning (POSTAL)**

Input: $\lambda \geq 0$, b, the max number of epochs T_{max}, the initial learning rate η_0, the step size μ, the loss function $r(\mathbf{w}; \{\mathbf{x}_i, y_i\})$, the regularizer $\rho_\lambda(\mathbf{w}) = \frac{\lambda}{2}\|\mathbf{w}\|^2$, and the training set $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^n$

Initialize: $t = 0$, $\tilde{\mathbf{w}}^{(0)}$ randomly, the dynamic threshold $D_{th} = 1$ by the max-margin principle
for $T = 1, 2, \ldots, T_{max}$ **do**
  Preprocess: $\mathbf{w}^{(tmp)} = \tilde{\mathbf{w}}^{(epoch-1)}$ and shuffle n data points in $\mathcal{D}$ stochastically
  for $k = 1, \ldots, n$ **do**
    Select: $\{\mathbf{x}_{it}, y_{it}\}$ from $\mathcal{D}$, $it \in \{1, ..., n\}$
    Curriculum: $z_{it}(\mathbf{w}^{(tmp)}) = (\langle \mathbf{w}^{(tmp)}, \mathbf{x}_{it} \rangle + b) y_{it}$
    If $z_{it}(\mathbf{w}^{(tmp)}) \geq D_{th}$:
      Update: $t = t + 1$ and $\eta = \eta_0(1 + \lambda\eta_0 t)^{-1}$
      Compute: $Q = \partial_{\mathbf{w}} r(\mathbf{w}^{(tmp)}; \{\mathbf{x}_{it}, y_{it}\})$
      Update: $\mathbf{w}^{(new)} = \mathbf{w}^{(tmp)} - \eta(\lambda \mathbf{w}^{(tmp)} + Q)$
      Assign: $\mathbf{w}^{(tmp)} = \mathbf{w}^{(new)}$
  end
  Assign: $\tilde{\mathbf{w}}^{(epoch)} = \mathbf{w}^{(tmp)}$
  Update: $D_{th} = D_{th} - \mu\sqrt{T}$
end
Output: $\tilde{\mathbf{w}}^{(T_{max})}$

of instance $\mathbf{x}_B$ and label y_C of instance $\mathbf{x}_C$ can be regarded as noisy labels. Among them, "B" is closer than "C" to the positive hyperplane, which means that "C" is more unreliable. More importantly, both labels y_B and y_C would negatively affect the update of the primal variable. To remedy this negative effect, POSTAL attempts to leverage the robust learning regime of CL. First, we define "the curriculum" in POSTAL as follows.

Definition 3.1. ***(The Curriculum in POSTAL)*** *Given training data $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^n$, if $f_{\mathbf{w}}$ denotes the current classifier, then the curriculum z in POSTAL can be calculated as the product of the annotated label y and the predicted label of $\mathbf{x}$, namely $z = y f_{\mathbf{w}}(\mathbf{x})$.*

Remark 4. *In POSTAL, $z_i > z_j$ represents that x_i is more reliable than x_j, which further means that x_i should be learned earlier than x_j. In conclusion, POSTAL learns from "reliable" labels to "noisy" labels.*

According to Definition 3.1, given reliable data $\{\mathbf{x}_A, y_A\}$, the noisy data $\{\mathbf{x}_B, y_B\}$ and $\{\mathbf{x}_C, y_C\}$ in the left panel of Figure 3.1, we have the curriculum $z_A = y_A f_{\mathbf{w}}(\mathbf{x}_A) > 0$, the curriculums $z_C = y_C f_{\mathbf{w}}(\mathbf{x}_C) < z_B = y_B f_{\mathbf{w}}(\mathbf{x}_B) < 0$. Therefore, in the right

panel of Figure 3.1, the curriculum $z = 0$ can be employed to separate the x-axis into the reliable region ($z \geq 0$) and the noisy region ($z \leq 0$) respectively. Furthermore, the update region for POSTAL ($z \geq D_{th}$) is controlled by a dynamic threshold D_{th}, and D_{th} is initialized to 1 on the x-axis by the max-margin principle. Through the robust learning process of CL, in the initial epoch, the update of the primal variable $\mathbf{w}$ is limited to "reliable" labels ($z \geq D_{th} = 1$) to establish a robust model. In the following epochs, we gradually reduce the dynamic threshold D_{th}. Correspondingly, updates occur on "noisy" labels incrementally until convergence. In conclusion, POSTAL learns from "reliable" label regions to "noisy" label regions, and details are shown in Algorithm 2.

Remark 5. *In Algorithm 2, POSTAL consists of two key components: curriculum calculation (line 6) and variable update (line 10). Specifically,*

1. *POSTAL computes the curriculum $z_{it}(\mathbf{w}^{(tmp)})$ for any stochastic sample $\{\mathbf{x}_{it}, y_{it}\}$ in line 6. If the corresponding curriculum locates within the current update region controlled by D_{th} (line 7), then POSTAL updates its primal variable $\mathbf{w}^{(tmp)}$ on this random sample in line 10, and vice versa.*

2. *With the decrease of D_{th} (line 14), POSTAL updates its primal variable from "reliable" samples to "noisy" samples; while vanilla SGD updates its primal variable on a stochastic sample sequence.*

3. *For simplicity, the curriculum $z_{it}(\mathbf{w}^{(tmp)})$ is realized by the linear mapping function f_w, namely $f_{\mathbf{w}^{(tmp)}}(\mathbf{x}_{it}) = \langle \mathbf{w}^{(tmp)}, \mathbf{x}_{it} \rangle + b$ in line 6. It means that we apply POSTAL to SVM classification model.*

4. *Since both SVM and Deep Learning are under the same empirical risk minimization (ERM) principle (1.2), the POSTAL mechanism can be leveraged by Deep Learning model if we represent f_w by deep neural networks. Namely, if we realize f_w using the nonlinear mapping function (deep neural networks) in line 6, then POSTAL could be applied to Deep Learning model.*

5. *Our theoretical analysis (Theorems 3.3 and 3.4) is based on ERM principle, therefore, this analysis may also hold for Deep Learning model with additional efforts (e.g., considering model capacity and optimization landscape).*

3.2 Realization of Progressive Stochastic Learning

To realize the POSTAL mechanism, we design a cluster of bounded losses called "screening losses". The screening losses are desirable for two reasons. First, screening losses sort all the labels from the reliable region ($z \geq 0$) to the noisy region ($z \leq 0$),

where $z = yf_{\mathbf{w}}(\mathbf{x})$ (the right panel of Figure 3.1). In other words, a cluster of screening losses serves as a guidance to provide an ordered label sequence for updates of the primal variable in POSTAL. Second, since screening losses are bounded, if the update of POSTAL enters the noisy region (i.e., around z_B and z_C), the stochastic gradients of non-convex screening losses are approximately zero in this region. Therefore, screening losses assist POSTAL in minimizing the adverse effects caused by noisy labels. Based on Definitions 1.1 and 1.2, we propose a cluster of screening losses $r(z)$ for POSTAL, where z is the curriculum variable of loss function (x-axis) in the left panel of Figure 3.1.

Definition 3.2. *(Definition 3 in [Han et al., 2016]) For POSTAL, a loss function $r(z)$ belongs to screening losses if it meets the following conditions simultaneously:*

(a). Upper bound condition - it should be bounded such that $\lim_{z\to-\infty} \frac{\mathrm{d}r(z)}{\mathrm{d}z} = 0$.

(b). Locally λ-strongly convex condition - it should be locally λ-strongly convex if there exists a constant $\lambda > 0$ such that $r(z) - \frac{\lambda}{2}\|z\|^2$ is convex when $z \in B_1(z^*, \gamma)$, where $B_1(z^*, \gamma)$ denotes the 1 dimensional Euclidean ball of radius $\gamma > 0$ centered at a local minimum z^*.

(c). Smoothly decreasing condition - it should be decreasing monotonically and differentiable continuously.

Remark 6. *Here, we illustrate above conditions in details.*

1. *The upper bound is regarded as the threshold to suppress the adverse effects led by noisy labels.*

2. *The loss function should be locally λ-strongly convex. If $r(\mathbf{w}; \{\mathbf{x}_i, y_i\})$ is locally λ-strongly convex and $\rho_\lambda(\mathbf{w})$ is globally λ-strongly convex (e.g., $\frac{\lambda}{2}\|\mathbf{w}\|^2$), the $G(\mathbf{w})$ is locally strongly-convex. Then, $G(\mathbf{w})$ meets the ARSC.*

3. *If $r(\mathbf{w}; \{\mathbf{x}_i, y_i\})$ is decreasing monotonically, we assume that $G(\mathbf{w})$ is non-increasing around some local minima. If $r(\mathbf{w}; \{\mathbf{x}_i, y_i\})$ is differentiable at every point, $g_i(\mathbf{w})$ meets the ARSM when $\frac{\lambda}{2}\|\mathbf{w}\|^2$ is used as the regularizer.*

In this chapter, we present two screening losses that satisfy the three conditions in Definition 3.2. One case is the screening ramp loss (3.1), which can be viewed as a smooth version of ramp loss (smoothing around s^* and 1). Here, we represent the screening ramp loss by a reversed sigmoid function:

$$r(s^*, z) = \frac{1 - s^*}{1 + e^{\alpha_{s^*}(z+\beta_{s^*})}}. \tag{3.1}$$

When parameter s^* of ramp loss is set, parameters α_{s^*} and β_{s^*} of screening ramp loss are decided by minimizing the difference between ramp loss and screening ramp loss.

The other case is the screening Gompertz loss. We represent the screening Gompertz loss by a reversed Gompertz function:

$$r(c^*, z) = e^{-e^{c^* \cdot z}}, \tag{3.2}$$

where parameter c^* controls the curve of this loss. In summary, the screening losses not only serve as a guidance to provide an ordered label sequence for updates of $\mathbf{w}$ in POSTAL, but also reduce the adverse effects caused by noisy labels on updates of $\mathbf{w}$. Moreover, screening losses are tailor-made to make POSTAL converge fast (Theorem 3.3 and Section 3.4.2.2). The realization of POSTAL (Algorithm 2) by screening losses is shown in Algorithm 3.

Algorithm 3 **Pr**ogressive **ST**och**A**stic **L**earning using **S**creening **L**osses

Input: $\lambda \geq 0$, b, the max number of epochs T_{max}, the initial learning rate η_0, the step size μ, the parameters s^* and c^* in loss function, the regularizer $\rho_\lambda(\mathbf{w}) = \frac{\lambda}{2}\|\mathbf{w}\|^2$, and the training set $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^n$

Initialize: $t = 0$, $\tilde{\mathbf{w}}^{(0)}$ randomly, the dynamic threshold $D_{th} = 1$ by the max-margin principle

Set: $\begin{cases} I : f(\alpha_{s^*}, \beta_{s^*}, z) = e^{\alpha_{s^*}(z+\beta_{s^*})} \\ II : f(c^*, z) = c^* z - e^{c^* z} \end{cases}$

for $T = 1, 2, \ldots, T_{max}$ **do**

  Preprocess: $\mathbf{w}^{(tmp)} = \tilde{\mathbf{w}}^{(epoch-1)}$ and shuffle n data points in $\mathcal{D}$ stochastically

  for $k = 1, \ldots, n$ **do**

    Select: $\{\mathbf{x}_{it}, y_{it}\}$ from $\mathcal{D}$, $it \in \{1, ..., n\}$

    Curriculum: $z_{it}(\mathbf{w}^{(tmp)}) = (\langle \mathbf{w}^{(tmp)}, \mathbf{x}_{it} \rangle + b) y_{it}$

    If $z_{it}(\mathbf{w}^{(tmp)}) \geq D_{th}$:

      Update: $t = t + 1$ and $\eta = \eta_0(1 + \lambda\eta_0 t)^{-1}$

      Compute: $\begin{cases} Q(I) : (1 - s^*)\alpha_{s^*}\mathbf{x}_{it} y_{it} \dfrac{f(\alpha_{s^*}, \beta_{s^*}, z_{it}(\mathbf{w}^{(tmp)}))}{(1 + f(\alpha_{s^*}, \beta_{s^*}, z_{it}(\mathbf{w}^{(tmp)})))^2} \\ Q(II) : c^* \mathbf{x}_{it} y_{it} e^{f(c^*, z_{it}(\mathbf{w}^{(tmp)}))} \end{cases}$

      Update: $\mathbf{w}^{(new)} = \begin{cases} I : \mathbf{w}^{(tmp)} - \eta^{(tmp)}\big[\lambda \mathbf{w}^{(tmp)} - Q(I)\big] \\ II : \mathbf{w}^{(tmp)} - \eta^{(tmp)}\big[\lambda \mathbf{w}^{(tmp)} - Q(II)\big] \end{cases}$

      Assign: $\mathbf{w}^{(tmp)} = \mathbf{w}^{(new)}$

  end

  Assign: $\tilde{\mathbf{w}}^{(epoch)} = \mathbf{w}^{(tmp)}$

  Update: $D_{th} = D_{th} - \mu\sqrt{T}$

end

Output: $\tilde{\mathbf{w}}^{(T_{max})}$

3.3 Analysis of Progressive Stochastic Learning

3.3.1 Convergence Analysis

If we apply POSTAL to the classification model (1.2) with screening losses, it converges to a local minimum. According to the conditions in Section 3.2, $G(\mathbf{w})$ meets ARSC and $g_i(\mathbf{w})$ meets ARSM, respectively. Then, we derive the convergence rate of POSTAL realized by screening losses. The details of convergence rate are shown in Theorem 3.3. Note that, $\mathbb{E}\big[\,\cdot\,\big]$ denotes the expectation.

Theorem 3.3. *For POSTAL using screening losses, consider that $G(\mathbf{w})$ and $g_i(\mathbf{w})$ satisfy Augmented Restricted Strong Convexity (ARSC) and Augmented Restricted Smoothness (ARSM), respectively. Let $\mathbf{w}^*$ be the local minimum and β be the parameter of ARSM. Assume that the learning rate η is sufficient to let $G(\mathbf{w}^{(t)})$ be a non-increasing update. After T actual updates, we have*

$$G(\mathbf{w}^{(T)}) - G(\mathbf{w}^*) \leq \frac{\mathbb{E}\big[\|\mathbf{w}^{(0)} - \mathbf{w}^*\|^2\big]}{(2\eta - 12\eta^2\beta)\cdot T}.$$

Remark 7. *When $T = \frac{d}{\eta\cdot\epsilon}$ and $d = \frac{\mathbb{E}\left[\|\mathbf{w}^{(0)}-\mathbf{w}^*\|^2\right]}{(2-12\eta\beta)}$, POSTAL realized by screening losses has ϵ-solution [1] and the convergence rate is $\mathcal{O}(1/T)$. To reach a ϵ-solution, the complexity of Algorithm 2 realized by screening losses, namely Algorithm 3, is $\mathcal{O}(\frac{n\cdot d}{\eta\cdot\epsilon})$.*

Proof. According to the update rule of POSTAL, $\mathbf{w}^{(t)} = \mathbf{w}^{(t-1)} - \eta\nabla g_{it}(\mathbf{w}^{(t-1)})$, where t is the current number of actual updates and varies from $1\cdots T$. The random number it belongs to the set $\{1, ..., n\}$ while $z_{it}(\mathbf{w}^{(t-1)}) \geq D_{th}$. Assume the number of training sample n is very large, due to (1.2), $|\mathbb{E}\big[\nabla g_{it}(\mathbf{w}^{(t-1)})\big] - \nabla G(\mathbf{w}^{(t-1)})| \leq \varepsilon$. With the increase of actual updates, the decrease of dynamic threshold D_{th} makes ε monotonically decrease towards 0. Therefore, we have the following inequality:

$$\begin{aligned}
&\mathbb{E}\big[\|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2\big] \\
&= \mathbb{E}\big[\|\mathbf{w}^{(t-1)} - \mathbf{w}^*\|^2\big] + \eta^2\mathbb{E}\big[\|\nabla g_{it}(\mathbf{w}^{(t-1)})\|^2\big] \\
&\quad - 2\eta\langle\nabla G(\mathbf{w}^{(t-1)}), \mathbf{w}^{(t-1)} - \mathbf{w}^*\rangle \\
&\leq \mathbb{E}\big[\|\mathbf{w}^{(t-1)} - \mathbf{w}^*\|^2\big] + \eta^2\mathbb{E}\big[\|\nabla g_{it}(\mathbf{w}^{(t-1)})\|^2\big] \\
&\quad - 2\eta\big[G(\mathbf{w}^{(t-1)}) - G(\mathbf{w}^*)\big],
\end{aligned} \tag{3.3}$$

where the inequality employs the ARSC. Specifically,

$$\langle\nabla G(\mathbf{w}^{(t-1)}), \mathbf{w}^* - \mathbf{w}^{(t-1)}\rangle \leq G(\mathbf{w}^*) - G(\mathbf{w}^{(t-1)}). \tag{3.4}$$

[1] Please refer to the page 47/315 in KDD15 tutorial: https://homepage.cs.uiowa.edu/~tyng/kdd15-tutorial.pdf for a good visualization of ϵ.

If we extract minus sign and then multiply 2η at the LHS and the RHS of (3.4) simultaneously, we have:

$$-2\eta\langle\nabla G(\mathbf{w}^{(t-1)}), \mathbf{w}^{(t-1)} - \mathbf{w}^*\rangle \leq -2\eta\big[G(\mathbf{w}^{(t-1)}) - G(\mathbf{w}^*)\big]. \tag{3.5}$$

Therefore, (3.3) has been guaranteed. Then we construct the Bregman divergence generator $\varphi_i(\mathbf{w})$ as an auxiliary function:

$$\varphi_i(\mathbf{w}) = g_i(\mathbf{w}) - g_i(\mathbf{w}^*) - \langle\nabla g_i(\mathbf{w}^*), \mathbf{w} - \mathbf{w}^*\rangle. \tag{3.6}$$

And it is obvious that:

$$\varphi_i(\mathbf{w}^*) = g_i(\mathbf{w}^*) - g_i(\mathbf{w}^*) = 0, \tag{3.7a}$$

$$\nabla\varphi_i(\mathbf{w}) = \nabla g_i(\mathbf{w}) - \nabla g_i(\mathbf{w}^*), \tag{3.7b}$$

$$\nabla\varphi_i(\mathbf{w}^*) = \nabla g_i(\mathbf{w}^*) - \nabla g_i(\mathbf{w}^*) = 0. \tag{3.7c}$$

Thus, $\mathbf{w}^*$ is a local minimum of $\varphi_i(\mathbf{w})$ by (3.7c) and we construct the following inequality from (3.6):

$$\begin{aligned} 0 = \varphi_i(\mathbf{w}^*) &\leq \min \varphi_i(\mathbf{w} - \gamma\nabla\varphi_i(\mathbf{w})) \\ &\leq \min \varphi_i(\mathbf{w}) + \frac{\beta\gamma^2}{2}\|\nabla\varphi_i(\mathbf{w}))\|^2 - \gamma\|\nabla\varphi_i(\mathbf{w}))\|^2 \\ &= \varphi_i(\mathbf{w}) - \frac{1}{2\beta}\|\nabla\varphi_i(\mathbf{w})\|^2, \end{aligned} \tag{3.8}$$

where the last inequality satisfies the ARSM and the function is minimized at the parameter $\gamma = \frac{1}{\beta}$. We construct the following inequality based on (3.6), (3.7b) and (3.8):

$$\begin{aligned} &\|\nabla g_i(\mathbf{w}) - \nabla g_i(\mathbf{w}^*)\|^2 \\ &\leq 2\beta\big[g_i(\mathbf{w}) - g_i(\mathbf{w}^*) - \langle\nabla g_i(\mathbf{w}^*), \mathbf{w} - \mathbf{w}^*\rangle\big]. \end{aligned} \tag{3.9}$$

Therefore, we have:

$$\begin{aligned} &\mathbb{E}\big[\|\nabla g_i(\mathbf{w}) - \nabla g_i(\mathbf{w}^*)\|^2\big] \\ &\leq 2\beta\big[G(\mathbf{w}) - G(\mathbf{w}^*) - \langle\nabla G(\mathbf{w}^*), \mathbf{w} - \mathbf{w}^*\rangle\big] \\ &\leq 4\beta\big[G(\mathbf{w}) - G(\mathbf{w}^*)\big], \end{aligned} \tag{3.10}$$

where the second last inequality satisfies the ARSC. Since $\|A + B + C\|^2 \leq 3\|A\|^2 + 3\|B\|^2 + 3\|C\|^2$ and $\mathbf{w}^*$ is a local minimum, we have the following inequality with $\nabla G(\mathbf{w}^*) = 0$ and (3.10):

$$\begin{aligned} &\mathbb{E}\big[\|\nabla g_{it}(\mathbf{w}^{(t-1)})\|^2\big] \\ &\leq 3\mathbb{E}\big[\|\nabla g_{it}(\mathbf{w}^{(t-1)}) - \nabla g_{it}(\mathbf{w}^*)\|^2\big] \\ &\quad + 3\mathbb{E}\big[\|\nabla g_{it}(\mathbf{w}^*) - \nabla G(\mathbf{w}^*)\|^2\big] + 3\mathbb{E}\big[\|\nabla G(\mathbf{w}^*)\|^2\big] \\ &\leq 12\beta\big[G(\mathbf{w}^{(t-1)}) - G(\mathbf{w}^*)\big]. \end{aligned} \tag{3.11}$$

Therefore, (3.3) equals the following inequality:

$$\begin{aligned}
&\mathbb{E}\big[\|\mathbf{w}^{(t)} - \mathbf{w}^*\|^2\big] \\
&\leq \mathbb{E}\big[\|\mathbf{w}^{(t-1)} - \mathbf{w}^*\|^2\big] + \eta^2\mathbb{E}\big[\|\nabla g_{it}(\mathbf{w}^{(t-1)})\|^2\big] \\
&\quad - 2\eta\big[G(\mathbf{w}^{(t-1)}) - G(\mathbf{w}^*)\big] \\
&\leq \mathbb{E}\big[\|\mathbf{w}^{(t-1)} - \mathbf{w}^*\|^2\big] \\
&\quad + (12\eta^2\beta - 2\eta)\big[G(\mathbf{w}^{(t-1)}) - G(\mathbf{w}^*)\big].
\end{aligned} \tag{3.12}$$

Based on (3.12), when t varies from $1 \cdots T$, we get T inequalities respectively, and then simultaneously add the LHS and RHS of T inequalities to get:

$$\begin{aligned}
\mathbb{E}\big[\|\mathbf{w}^{(T)} - \mathbf{w}^*\|^2\big] &\leq \mathbb{E}\big[\|\mathbf{w}^{(0)} - \mathbf{w}^*\|^2\big] \\
&+ (12\eta^2\beta - 2\eta)\big[\sum_{t=1}^{T} G(\mathbf{w}^{(t-1)}) - T \cdot G(\mathbf{w}^*)\big].
\end{aligned} \tag{3.13}$$

On the assumption of a non-increasing update, we have the following inequality:

$$\begin{aligned}
&(2\eta - 12\eta^2\beta)\big[T \cdot G(\mathbf{w}^{(T)}) - T \cdot G(\mathbf{w}^*)\big] \\
&\leq (2\eta - 12\eta^2\beta)\big[\sum_{t=1}^{T} G(\mathbf{w}^{(t-1)}) - T \cdot G(\mathbf{w}^*)\big] \\
&\leq \mathbb{E}\big[\|\mathbf{w}^{(0)} - \mathbf{w}^*\|^2\big] - \mathbb{E}\big[\|\mathbf{w}^{(T)} - \mathbf{w}^*\|^2\big].
\end{aligned} \tag{3.14}$$

We thus obtain:

$$\begin{aligned}
G(\mathbf{w}^{(T)}) - G(\mathbf{w}^*) &\leq \frac{\mathbb{E}\big[\|\mathbf{w}^{(0)} - \mathbf{w}^*\|^2\big] - \mathbb{E}\big[\|\mathbf{w}^{(T)} - \mathbf{w}^*\|^2\big]}{(2\eta - 12\eta^2\beta) \cdot T} \\
&\leq \frac{\mathbb{E}\big[\|\mathbf{w}^{(0)} - \mathbf{w}^*\|^2\big]}{(2\eta - 12\eta^2\beta) \cdot T} = \frac{d}{\eta \cdot T} = \epsilon,
\end{aligned} \tag{3.15}$$

where $d = \frac{\mathbb{E}\left[\|\mathbf{w}^{(0)} - \mathbf{w}^*\|^2\right]}{(2-12\eta\beta)}$. Therefore we conclude that when $T = \frac{d}{\eta \cdot \epsilon}$, POSTAL realized by screening losses has ϵ-solution and its convergence rate is $\mathcal{O}(1/T)$. To reach a ϵ-solution, the complexity of POSTAL algorithm realized by screening losses, namely Algorithm 3, is $\mathcal{O}(\frac{n \cdot d}{\eta \cdot \epsilon})$. □

3.3.2 Robustness Analysis

For the POSTAL mechanism, in the initial epoch, the update is limited to the reliable labels to establish a robust model. In following epochs, the updates gradually occur on noisy labels. Therefore, this mechanism is empirically robust due to the nature of

curriculum learning [Bengio et al., 2009]. Here, we explore the robustness of screening losses, which assist POSTAL to reduce adverse effects led by noisy labels. The details of the robustness analysis are shown in Theorem 3.4.

Theorem 3.4. *(Theorem 2 in [Han et al., 2016]) Assume that $\mathbf{x}_i$ is annotated with a noisy label y_i, namely $y_i(\mathcal{K}_i^T\alpha + b) < 0$, where $\mathcal{K}_i$ is the i-th component of mercer kernel. Then its corresponding weighted coefficient ϕ_i for screening ramp loss with $(s^*, \alpha_{s^*}, \beta_{s^*})$ is*

$$\phi_i = \frac{(1-s^*)\alpha_{s^*}\delta e^{\alpha_{s^*}(y_i\mathcal{K}_i^T\alpha + y_i b)}}{(1-(y_i\mathcal{K}_i^T\alpha + y_i b))(1+\delta e^{\alpha_{s^*}(y_i\mathcal{K}_i^T\alpha + y_i b)})^2};$$

and the coefficient ϕ_i for screening Gompertz loss with c^* is

$$\phi_i = \frac{c^* e^{c^*(y_i\mathcal{K}_i^T\alpha + y_i b) - e^{c^*(y_i\mathcal{K}_i^T\alpha + y_i b)}}}{1-(y_i\mathcal{K}_i^T\alpha + y_i b)}.$$

When $|f_w(\mathbf{x}_i)| = |(\mathcal{K}_i^T\alpha + b)|$ increases, it means that $\mathbf{x}_i$ with the noisy label y_i becomes more unreliable, then ϕ_i will decrease. This phenomenon demonstrates that screening losses do suppress the adverse effects caused by noisy labels.

Remark 8. *If $\{\mathbf{x}_i, y_i\}$ is an instance with a noisy label, then the mislabeled instance becomes more unreliable when $|f_{\mathbf{w}}(\mathbf{x}_i)| = |(\mathcal{K}_i^T\alpha + b)|$ increases. This means that the mislabeled instance is far away from the hyperplane. The coefficient ϕ_i for both screening losses will then decrease because $1-(y_i\mathcal{K}_i^T\alpha + y_i b)$ will increase, while $\frac{e^{\alpha_{s^*}(y_i\mathcal{K}_i^T\alpha + y_i b)}}{(1+\delta e^{\alpha_{s^*}(y_i\mathcal{K}_i^T\alpha + y_i b)})^2}$ and $e^{c^*(y_i\mathcal{K}_i^T\alpha + y_i b) - e^{c^*(y_i\mathcal{K}_i^T\alpha + y_i b)}}$ will decrease. This denotes that ϕ_i will decrease with the increase of $|f_{\mathbf{w}}(\mathbf{x}_i)|$ for unreliable instance $\mathbf{x}_i$ and does not play an important role in the update of dual variable. Thus, screening ramp loss and screening Gompertz loss can suppress adverse effects led by noisy labels.*

Proof. Assume that $\{\mathbf{x}_i, y_i\}_{i=1}^k$ is a random subset of the training data $\mathcal{D}$ and $f_{\mathbf{w}}$ is the decision function, according to the representer theorem, $z_i = y_i f_{\mathbf{w}}(\mathbf{x}_i) = y_i(\sum_{j=1}^k \mathcal{K}(\mathbf{x}_j, \mathbf{x}_i)\alpha_j + b) = y_i\mathcal{K}_i^T\alpha + y_i b$, where $\alpha = (\alpha_1, \alpha_2, ..., \alpha_k)'$, $\mathcal{K} = (\mathcal{K}_1, \mathcal{K}_2, ..., \mathcal{K}_k)'$ and $\mathcal{K}_i = (\mathcal{K}(\mathbf{x}_1, \mathbf{x}_i), \mathcal{K}(\mathbf{x}_2, \mathbf{x}_i), ..., \mathcal{K}(\mathbf{x}_k, \mathbf{x}_i))'$. $\lambda > 0$ is a regularizer parameter, $\mathcal{K}$ is a mercer kernel and $\mathcal{H}_{\mathcal{K}}$ is a reproducing kernel hilbert space (RKHS). For a cluster of screening losses $r(z)$, we define two functions $q(z)$ and $\varrho(z)$ such that $r(z) = q(1-z)$ and $\varrho(z) = \frac{q'(z)}{z}$. Therefore, our robust model is

presented as:

$$
\begin{aligned}
f^*_{\mathbf{w}} &= \arg\min_{f_{\mathbf{w}}} \frac{1}{k}\sum_{i=1}^{k} r(z_i) + \frac{\lambda}{2}\|f_{\mathbf{w}}\|^2 \\
&= \arg\min_{f_{\mathbf{w}}} \frac{1}{k}\sum_{i=1}^{k} r(f_{\mathbf{w}}(\mathbf{x}_i)\cdot y_i) + \frac{\lambda}{2} f_{\mathbf{w}}^T f_{\mathbf{w}} \\
&= \arg\min_{\alpha,b} \frac{1}{k}\sum_{i=1}^{k} q(1 - y_i\mathcal{K}_i^T\alpha - y_i b) + \frac{\lambda}{2}\alpha^T\mathcal{K}\alpha.
\end{aligned}
\tag{3.16}
$$

The last equation satisfies the second condition of screening losses $r(z) = q(1-z)$. Due to $\varrho(z) = \frac{q'(z)}{z}$, we define coefficient $\phi_i = \varrho(1 - y_i\mathcal{K}_i^T\alpha - y_i b)$, then

$$
q'(1 - y_i\mathcal{K}_i^T\alpha - y_i b) = (1 - y_i\mathcal{K}_i^T\alpha - y_i b)\phi_i. \tag{3.17}
$$

Because our proposed loss is non-convex, we assume that $(\hat{\alpha}, \hat{b})$ is one of the critical points for above minimization problem (3.16). Let us set $Q(\alpha, b) = \frac{1}{k}\sum_{i=1}^{k} q(1 - y_i\mathcal{K}_i^T\alpha - y_i b) + \frac{\lambda}{2}\alpha^T\mathcal{K}\alpha$, therefore: $\frac{\partial Q(\hat{\alpha},\hat{b})}{\partial\alpha} = 0$ and $\frac{\partial Q(\hat{\alpha},\hat{b})}{\partial b} = 0$. Then, we have two equations below:

$$
\frac{1}{k}\sum_{i=1}^{k}(1 - y_i\mathcal{K}_i^T\hat{\alpha} - y_i\hat{b})(y_i\mathcal{K}_i)\phi_i - \lambda\mathcal{K}^T\hat{\alpha} = 0. \tag{3.18}
$$

$$
\frac{1}{k}\sum_{i=1}^{k}(1 - y_i\mathcal{K}_i^T\hat{\alpha} - y_i\hat{b})y_i\phi_i = 0. \tag{3.19}
$$

The solution $(\hat{\alpha}, \hat{b})$ of (3.18) and (3.19) can be achieved by solving the following L2-SVM

$$
\min_{\alpha,b} \frac{1}{k}\sum_{i=1}^{k}(y_i - \mathcal{K}_i^T\alpha - b)^2\phi_i + \frac{\lambda}{2}\alpha^T\mathcal{K}\alpha. \tag{3.20}
$$

When $k = 1$, (3.20) is solved by streaming SGD. If $k > 1$, (3.20) can be solved by mini-batch SGD. Currently, we consider ϕ_i as an important coefficient that affects the update of the stochastic dual variable α, and therefore, we analyze the robust statistics briefly from coefficient ϕ_i view.

If an instance $\mathbf{x}_i$ is annotated with a noisy label y_i, it means that $y_i f_{\mathbf{w}}(\mathbf{x}_i) < 0$. By the representer theorem, we can easily find $y_i(\mathcal{K}_i^T\alpha + b) < 0$ for this instance. We consider $|(\mathcal{K}_i^T\alpha + b)|$ as the degree where this instance is far away from the hyperplane.

So we define $\phi_i = \varrho(1 - y_i\mathcal{K}_i^T\alpha - y_i b)$. To analyze the robustness of $r(z)$, we employ two concrete cases: screening ramp loss and screening Gompertz loss.

First, we analyze the robustness of screening ramp loss. We define $\delta = e^{\alpha_{s^*}\beta_{s^*}}$. According to our inference

$$\begin{aligned}\phi_i &= \varrho(1 - y_i\mathcal{K}_i^T\alpha - y_i b) \\ &= \frac{(1-s^*)\alpha_{s^*}\delta e^{\alpha_{s^*}(y_i\mathcal{K}_i^T\alpha + y_i b)}}{(1-(y_i\mathcal{K}_i^T\alpha + y_i b))(1+\delta e^{\alpha_{s^*}(y_i\mathcal{K}_i^T\alpha + y_i b)})^2}.\end{aligned} \tag{3.21}$$

If $\{\mathbf{x}_i, y_i\}$ is an instance with a noisy label ($y_i(\mathcal{K}_i^T\alpha + b) < 0$), then the mislabeled instance becomes more unreliable when $|f_{\mathbf{w}}(\mathbf{x}_i)| = |(\mathcal{K}_i^T\alpha + b)|$ increases. This means that the mislabeled instance is far away from the hyperplane. The coefficient ϕ_i will then decrease because $1 - (y_i\mathcal{K}_i^T\alpha + y_i b)$ will increase, while $\frac{e^{\alpha_{s^*}(y_i\mathcal{K}_i^T\alpha + y_i b)}}{(1+\delta e^{\alpha_{s^*}(y_i\mathcal{K}_i^T\alpha + y_i b)})^2}$ will decrease. This indicates that the parameter ϕ_i will decrease with the increase of $|f_{\mathbf{w}}(\mathbf{x}_i)|$ for instance $\mathbf{x}_i$ and does not play a significant role in the update of the dual variable. Thus, screening ramp loss can suppress the adverse effects introduced by noisy labels.

Second, we analyze the robustness of screening Gompertz loss ($c^* > 0$). According to our inference:

$$\begin{aligned}\phi_i &= \varrho(1 - y_i\mathcal{K}_i^T\alpha - y_i b) \\ &= \frac{c^* e^{c^*(y_i\mathcal{K}_i^T\alpha + y_i b) - e^{c^*(y_i\mathcal{K}_i^T\alpha + y_i b)}}}{1-(y_i\mathcal{K}_i^T\alpha + y_i b)}.\end{aligned} \tag{3.22}$$

If $\{\mathbf{x}_i, y_i\}$ is an instance with a noisy label ($y_i(\mathcal{K}_i^T\alpha + b) < 0$), then the mislabeled instance becomes more unreliable when $|f_{\mathbf{w}}(\mathbf{x}_i)| = |(\mathcal{K}_i^T\alpha + b)|$ increases. This means that the mislabeled instance is far away from the hyperplane. The parameter ϕ_i will then decrease because $1 - (y_i\mathcal{K}_i^T\alpha + y_i b)$ will increase and $e^{c^*(y_i\mathcal{K}_i^T\alpha + y_i b) - e^{c^*(y_i\mathcal{K}_i^T\alpha + y_i b)}}$ will decrease. This indicates that the coefficient ϕ_i will decrease with the increase of $|f_{\mathbf{w}}(\mathbf{x}_i)|$ for instance $\mathbf{x}_i$ and play a trivial role in the update of the dual variable. Thus, screening Gompertz loss can suppress the adverse effects introduced by noisy labels. □

3.4 Experiments

In this section, on noisy datasets, we conduct experiments to verify the effectiveness of POSTAL mechanism first, then verify the effectiveness and robustness of POSTAL using (two devised) screening losses. The datasets include UCI simulated and AMT crowdsourcing datasets. For convenience, we abbreviate POSTAL using screening

Table 3.2: Datasets used in the UCI simulated study. Representative datasets are emphasized by color.

DATA SET	TRAINING PTS.	TESTING PTS.	FEATURES.
A7A	16,100	16,461	123
IJCNN1	49,990	91,701	22
REAL-SIM	57,847	14,462	20,985
COVTYPE	464,810	116,202	54
MNIST38	450,000	97,570	784
SUSY	4,000,000	1,000,000	18

ramp loss to POSTAL(SRamp) and POSTAL using screening Gompertz loss to POSTAL(SGomp).

3.4.1 Experimental Setup

3.4.1.1 Baselines

There are three categories of baselines. The first category consists of vanilla SGD using proposed screening losses (e.g., screening ramp and screening Gompertz losses). When we compare them with POSTAL using screening losses, we can verify the effectiveness of POSTAL mechanism under noisy settings. The second category consists of vanilla SGD using basic losses (e.g., logistic, hinge and ramp losses), ASGD [Xu, 2011] using logistic loss, PEGASOS [Shalev-Shwartz et al., 2011] and "Library for Large Linear Classification" (LIBLINEAR) [Fan et al., 2008], which can verify the effectiveness and robustness of POSTAL using screening losses under noisy settings. The third category consists of μSGD with different noise rates [Patrini et al., 2016], which can verify the robustness of POSTAL using screening losses in real-world situations.

There are three points to be noted that: (1) We use LIBLINEAR because it has become popular for handling large-scale datasets. We abbreviate L2-regularized L2-loss primal solver as "LIBPrimal" and dual solver as "LIBDual". (2) We do not compare the variance-reduction techniques such as SVRG and SAGA for the following reason. In this chapter, the aim of POSTAL is to improve the robustness of vanilla SGD against noisy labels. According to the idea of control variable, the fair comparison should be between SGD and the curriculum version of SGD, i.e., POSTAL and SGD instead of POSTAL and SVRG or SAGA. (3) Experiments are implemented on a laptop with a 3.20GHz CPU and 8GB memory.

3.4.1.2 Parameters

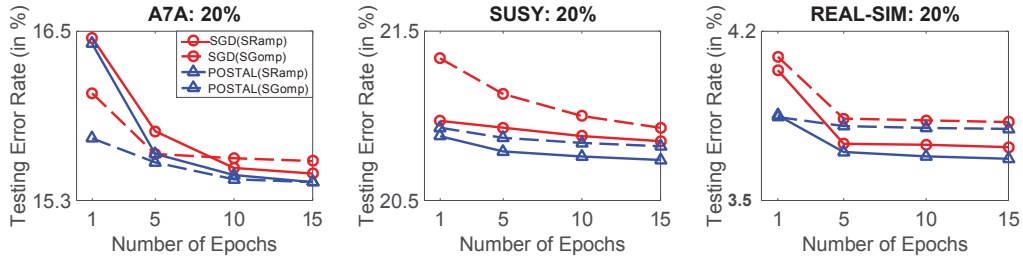
The parameter λ is selected using 10-fold cross validation (CV) in the range of $\{10^{-6}, 10^{-5}, \cdots, 10\}$ for all methods. For all stochastic learning methods, the maximum number of epochs T_{max} is set to 15, and the primal variable $\mathbf{w}$ is initialized randomly. It is noted that, for the convergence of PEGASOS, T_{max} is set to $\frac{10}{\lambda}$ by default. For LIBLINEAR, we set the bias b to 1 and the stopping tolerance ϵ to 10^{-2} for primal solver and 10^{-1} for dual solver by default. For μSGD, we set the noise rates $(p_+, p_-) = (0.1, 0.1)$ and $(0.2, 0.1)$ followed by [Patrini et al., 2016], and other parameters by default. For convenience, we abbreviate μSGD with noise rates $(0.1, 0.1)$ and $(0.2, 0.1)$ as μSGD$(0.1, 0.1)$ and μSGD$(0.2, 0.1)$, respectively. By 10-fold CV, for POSTAL(SRamp), the parameter s^* is chosen in the range of $[-2, 0]$. For POSTAL(SGomp), the parameter c^* is set to 2. For both of them, we set η_0 by the standard method [Bottou, 2010] and step size μ to 1. We repeat all experiments 20 times. Then each plot is averaged across 20 results. Methods not reported in Figure 3.4 and Figure 3.5 are because they either run out of memory or take very long time in training phase.

3.4.2 Empirical Study on UCI Simulated Datasets

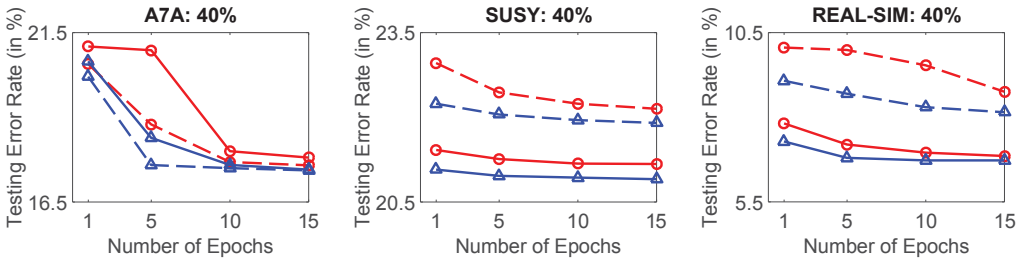
UCI simulated datasets are from [Chang and Lin, 2011]. The statistics for six datasets are summarized in Table 3.2. From the above datasets, we denote *A7A*, *SUSY* and *REAL-SIM* as representative datasets, where *SUSY* and *REAL-SIM* represent the large-scale dataset and the high-dimensional dataset, respectively. To yield the noisy datasets, we follow the settings from [Natarajan et al., 2013], and flip the data labels proportionally. For instance, we randomly flip 20% of data labels from -1 to 1 or 1 to -1, and assume that the data owns 20% of noisy labels. We repeat the same procedure to yield 40% of noisy labels. In this chapter, we believe both 20% and 40% of noisy labels are reasonable. Specifically, 20% of noisy labels are very normal in real applications [Zhao et al., 2011]. We select 40% of noisy labels for two reasons. First, it displays the superiority of POSTAL in extreme conditions. Second, 40% of noisy labels has been leveraged by previous researchers [Yang et al., 2012]. Moreover, spammers annotate binary-value tasks randomly and 50% of noisy labels are even introduced [Shah and Zhou, 2015], which is beyond 40% of noisy labels.

3.4.2.1 Effectiveness of POSTAL Mechanism

To verify the effectiveness of POSTAL mechanism, in Figure 3.2, we provide the performance (the testing error rate (in %, TER) with the number of epochs) comparison between POSTAL and vanilla SGD on representative UCI noisy datasets: small-scale *A7A*, large-scale *SUSY* and high-dimensional *REAL-SIM* with 20% and 40%



(a) TER(%) with the number of epochs on *A7A*, *SUSY* and *REAL-SIM* with 20% of noisy labels.

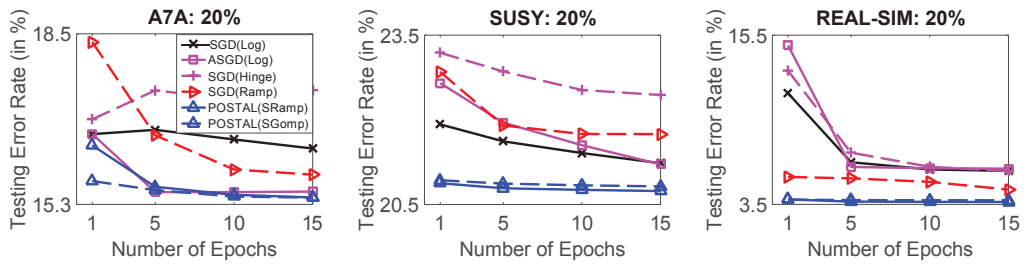


(b) TER(%) with the number of epochs on *A7A*, *SUSY* and *REAL-SIM* with 40% noisy labels.

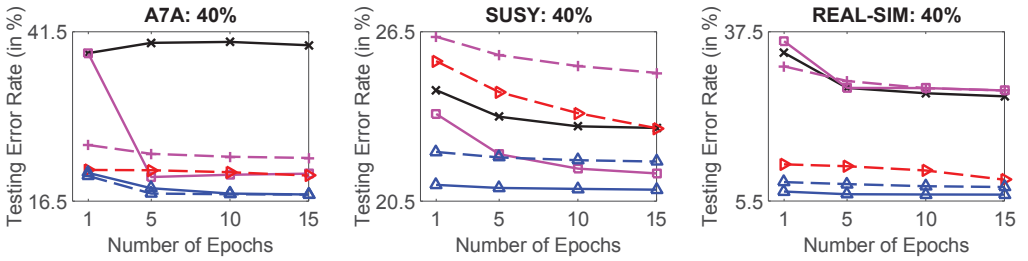
Figure 3.2: To verify the *effectiveness* of POSTAL mechanism, we compare POSTAL with vanilla SGD under the same screening losses. We provide the testing error rate (in %, TER) with the number of epochs on representative UCI noisy datasets: small-scale *A7A*, large-scale *SUSY* and high-dimensional *REAL-SIM* with 20% and 40% of noisy labels.

Table 3.3: To verify the robustness of POSTAL using screening losses, we also provide the comparison of "testing error rate (in %) with standard deviation" among all methods on UCI clean datasets. Methods indicated by "-"run out of memory.

METHODS	*A7A*	*IJCNN1*	*REAL-SIM*	*COVTYPE*	*MNIST38*	*SUSY*
LIBPRIMAL	**14.99**	8.25	2.57	24.35	5.71	21.34
LIBDUAL	15.02	8.20	2.67	24.25	6.09	35.32
PEGASOS	17.62±1.56	8.50±0.19	3.32±0.06	26.36±1.99	-	-
SGD(Log)	15.16±0.06	9.08±0.48	2.62±0.03	25.07±0.28	5.73±0.09	20.93±0.01
ASGD(Log)	14.99±0.14	8.04±0.04	2.54±0.01	24.38±0.01	5.54±0.01	20.83±0.09
SGD(Hinge)	15.45±0.09	8.40±0.22	2.69±0.13	24.62±0.54	5.77±0.16	20.89±0.08
SGD(Ramp)	15.54±0.54	8.50±0.03	4.02±0.02	24.22±0.10	6.04±0.08	21.36±0.05
POSTAL(SRamp)	15.08±0.02	**6.04±0.05**	2.54±0.01	23.63±0.01	5.73±0.01	**20.7±0.01**
POSTAL(SGomp)	15.03±0.02	6.30±0.02	**2.38±0.01**	**23.18±0.01**	**5.49±0.01**	20.72±0.01

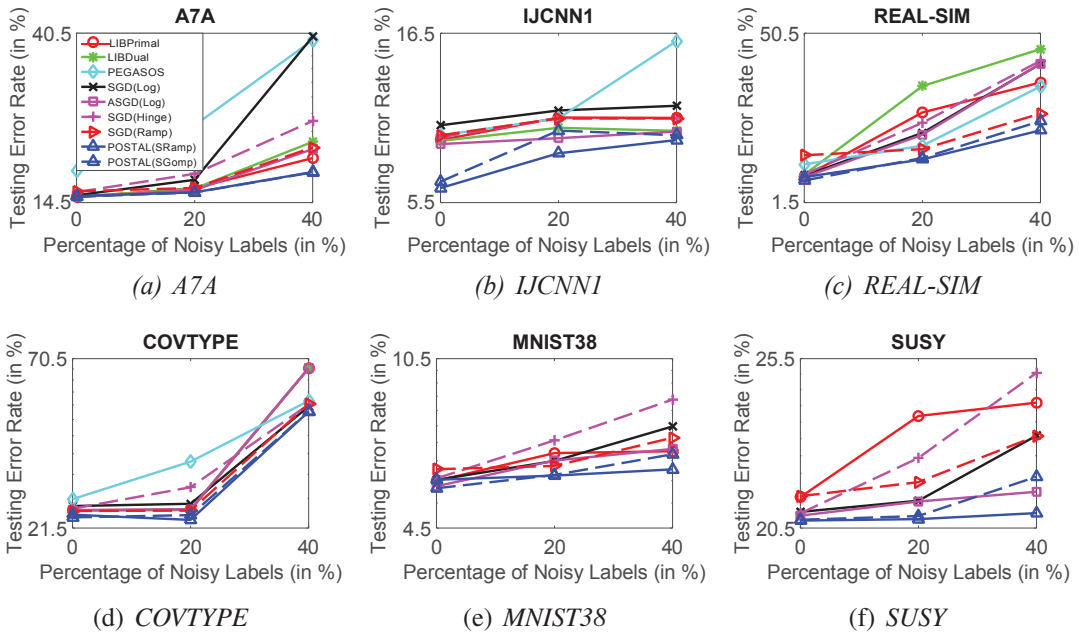


(a) TER(%) with the number of epochs on *A7A*, *SUSY* and *REAL-SIM* with 20% of noisy labels.



(b) TER(%) with the number of epochs on *A7A*, *SUSY* and *REAL-SIM* with 40% of noisy labels.

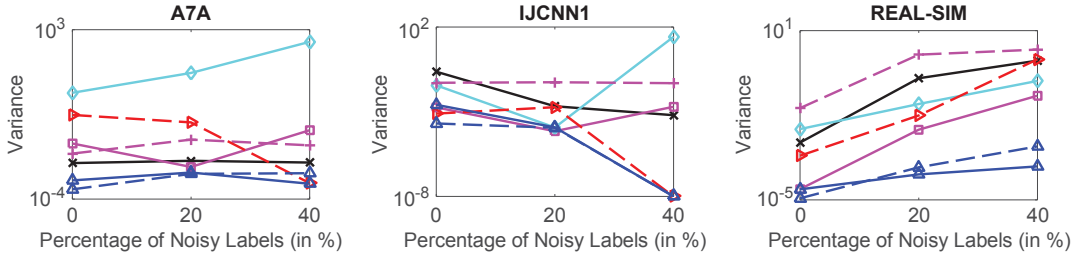
Figure 3.3: To verify the *effectiveness* of POSTAL using screening losses, we compare POSTAL(SRamp) and POSTAL(SGomp) with stochastic methods in the second category of baselines (paragraph 1 in Section 3.4.1). We provide the testing error rate (in %, TER) with the number of epochs on representative UCI noisy datasets: small-scale *A7A*, large-scale *SUSY* and high-dimensional *REAL-SIM* with 20% and 40% of noisy labels. For PEGASOS, the number of epochs for convergence is set to $\frac{10}{\lambda}(\gg 15)$ by default. Thus, its result is not reported.



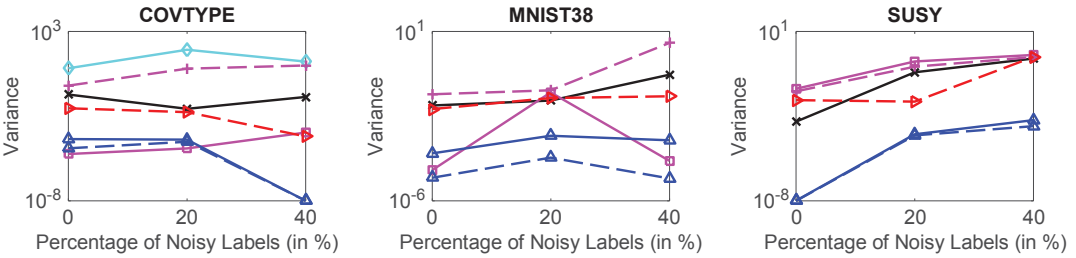
(a) A7A *(b) IJCNN1* *(c) REAL-SIM*

(d) *COVTYPE* (e) *MNIST38* (f) *SUSY*

Figure 3.4: To verify the *robustness* of POSTAL using screening losses, we compare POSTAL(SRamp) and POSTAL(SGomp) with all baselines on all UCI noisy datasets. We provide the testing error rate (in %, TER) with varying percentages (0% to 40%) of noisy labels. Note that the color of each method is uniform.



(a) *A7A* (b) *IJCNN1* (c) *REAL-SIM*



(d) *COVTYPE* (e) *MNIST38* (f) *SUSY*

Figure 3.5: To verify the *robustness* of POSTAL using screening losses, we compare POSTAL(SRamp) and POSTAL(SGomp) with all baselines on all UCI noisy datasets. We provide the variance with varying percentages (0% to 40%) of noisy labels. Note that the color of each method is uniform.

of noisy labels. Inspired by the idea of control variable, we leverage the same screening losses to realize POSTAL and SGD. We observe, using the same screening losses, POSTAL converges faster than vanilla SGD. Meanwhile, using fewer number of epochs, POSTAL achieves the lower TERs than vanilla SGD, which verifies the effectiveness of POSTAL mechanism.

3.4.2.2 Effectiveness of POSTAL using Screening Losses

We further verify the effectiveness of POSTAL using screening losses on representative UCI noisy datasets. We provide the TER with the number of epochs on small-scale *A7A*, large-scale *SUSY* and high-dimensional *REAL-SIM* with 20% and 40% of noisy labels. We observe that in Figure 3.3, with an increase in the number of epochs, the TERs of POSTAL(SRamp) and POSTAL(SGomp) not only decrease faster than those of other baseline methods but also stay relative stable in most cases. Our methods takes within 5 epochs to reach their lowest TERs, whereas SGD(Hinge) spends around 15 epochs to reach its lowest TER in presence of 20% and 40% of noisy labels. Even worse, in presence of *A7A* with 40% of noisy labels, SGD(Log) diverges with an increase in the number of epochs.

3.4.2.3 Robustness Improvement

We verify the robustness of POSTAL using screening losses on all UCI noisy datasets. Figure 3.4 and Figure 3.5 respectively present the testing error rate (TER) and the variance with varying percentages of noisy labels on six datasets (e.g., from small to large-scale and high-dimensional datasets). According to the results, we derive the following conclusions. (a) For all datasets, POSTAL(SRamp) outperforms another baseline methods in TER beyond 20% of noisy labels. From 0% to 20%, both POSTAL(SRamp) and POSTAL(SGomp) still have comparative advantages. It is noted that, for high-dimensional REAL-SIM, the advantage of POSTAL(SRamp) and POSTAL(SGomp) is significantly obvious along the entire x-axis. (b) We observe that the variance of TER for baseline methods (e.g., PEGASOS) gradually increases with the growing percentage of noisy labels, while the variance of TER for POSTAL(SRamp) and POSTAL(SGomp) keeps at the lowest level in most cases. By our analysis, the good results of POSTAL on very high percentage of noisy labels (40%) may be due to two reasons. First, the mechanism of curriculum learning (CL) learns a reliable model from the ordered label sequence. Second, the upper bound of screening losses suppresses the adverse effects led by highly noisy labels.

Given the above observations, POSTAL(SRamp) and POSTAL(SGomp) outperform other baselines on noisy datasets. Furthermore, both of them outperform other baselines on clean datasets. For instance, Table 3.3 showcases that according to the TER with the standard deviation, POSTAL(SRamp) and POSTAL(SGomp) surpass

Figure 3.6: Sample images of four categories of dogs from the *Stanford Dogs* dataset.

other baselines on *IJCNN1*, *REAL-SIM*, *COVTYPE*, *MNIST38* and *SUSY* without noisy labels.

3.4.3 Real-World Application on AMT Crowdsourced Data

Since crowdsourced data is often noisy, it can be leveraged to test whether an algorithm is robust in real-world situations. Here, we employ one crowdsourcing dataset [Zhou et al., 2012b] from Amazon Mechanical Turk (AMT) [1] to verify the robustness of POSTAL using screening losses. The selected dataset includes annotated images of four categories of dogs from the *Stanford Dogs* dataset. This image dataset contains 7354 labels provided by 52 actual workers on the AMT platform. The four categories of dogs consist of Norwich Terrier (NWT), Norfolk Terrier (NFT), Scottish Deerhound (SDH) and Irish Wolfhound (IWH) (Figure 3.6). Based on this dataset, we construct four crowdsourcing subsets for binary classification denoted by *Dog(NWT)*, *Dog(NFT)*, *Dog(SDH)* and *Dog(IWH)*. The positive category in each subset is listed in the parentheses, images sampled from the other three categories are considered to be negative. The ratio between the positive and negative samples is around $1:1$. For each subset, we aggregate multiple crowdsourced labels of each image to yield single noise label by the simplest majority voting, then we verify the robustness of POSTAL using screening losses and other baseline methods on these noisy labels.

3.4.3.1 Robustness Improvement

Table 3.4 represents that according to the TER with standard deviation, POSTAL(SRamp) and POSTAL(SGomp) surpass other baselines around 2% on *Dog(NWT)*, *Dog(NFT)*, *Dog(SDH)* and *Dog(IWH)* datasets. One underlying reason is that, in the AMT

[1] https://www.mturk.com/mturk/welcome

Table 3.4: To verify the robustness of POSTAL using screening losses in real-world situations, we provide the comparison of "testing error rate (in %, TER) with standard deviation" on four crowdsourcing subsets. In each subset, the ratio between positive samples and negative samples is around $1:1$. For LIBLINEAR, the left TER is from LIBPrimal while the right TER is from LIBDual.

METHODS	**LIBLINEAR**	**PEGASOS**	**SGD(Log)**	**ASGD(Log)**	**SGD(Hinge)**
Dog(NWT)	22.22 / 23.23	22.81±1.29	23.51±0.57	22.90±0.36	22.94±0.86
Dog(NFT)	19.70 / 16.65	14.85±1.49	14.90±0.27	13.70±0.18	15.41±0.81
Dog(SDH)	18.18 / 19.70	19.02±1.05	20.46±1.18	19.70±1.01	20.58±1.26
Dog(IWH)	23.74 / 24.24	23.61±1.04	24.12±0.50	22.07±0.24	27.28±1.53
METHODS	**SGD(Ramp)**	**μSGD(0.1,0.1)**	**μSGD(0.2,0.1)**	**POSTAL(SRamp)**	**POSTAL(SGomp)**
Dog(NWT)	22.05±0.69	22.72±0.26	23.50±0.29	**21.58±0.24**	21.89±0.36
Dog(NFT)	15.15±0.41	14.50±0.31	16.35±0.43	**13.23±0.23**	13.89±0.29
Dog(SDH)	18.47±0.77	18.50±0.25	19.20±0.61	16.85±0.62	**16.79±0.31**
Dog(IWH)	22.10±0.86	23.35±0.72	23.50±0.52	**19.92±0.40**	20.88±0.25

crowdsourcing dataset, the average and best accuracy of workers is 70.60% and 88.24% respectively, which introduces low percentage of noisy labels. In other words, the quality of labels is good enough for updates of the primal variable in SGD. However, we conjecture that, if four breeds of dogs are too similar to be distinguished easily (e.g., Norfolk Terrier, Norwich Terrier, Irish Terrier and Silky Terrier), and non-professional workers are hired to annotate these dogs, POSTAL(SRamp) and POSTAL(SGomp) may outperform other baseline methods higher than 2% since both of them perform well with high percentage of noisy labels.

3.5 Summary of This Chapter

This chapter studies a robust SGD mechanism called progressive stochastic learning (POSTAL) for the label noise problem. POSTAL incorporates the progressive learning paradigm of curriculum learning (CL) with the update process of vanilla SGD. Through the learning process of CL, POSTAL yields robust updates of the primal variable on an ordered label sequence, namely from "reliable" labels to "noisy" labels. To provide this ordered label sequence, we design a cluster of screening losses, where all labels are sorted from the reliable region to the noisy region. Our analysis derives the convergence rate of POSTAL realized by screening losses, while demonstrates the robustness of screening losses. Comprehensive experiments on UCI simulated and AMT real-world datasets show that, proposed method is sufficiently effective and robust to reduce adverse effects caused by noisy labels.

Chapter 4

Co-teaching: Towards Robust Generalization

This chapter answers the third question, namely "how to train the robust model under noisy labels?". From the high level, we hope to introduce the insights from "peer review" to robustly training deep neural networks with extremely noisy labels.

As mentioned in the Chapter 1, a promising direction focuses on training on selected samples. These works try to select clean instances out of the noisy ones, and then use them to update the network. Intuitively, as the training data becomes less noisy, better performance can be obtained. Among these works, the representative methods are MentorNet [Jiang et al., 2018] and Decoupling [Malach and Shalev-Shwartz, 2017].

Specifically, MentorNet pre-trains an extra network, and then uses the extra network for selecting clean instances to guide the training. When the clean validation data is not available, MentorNet has to use a predefined curriculum (e.g., self-paced curriculum). Nevertheless, the idea of self-paced MentorNet is similar to the self-training approach, which may introduce the accumulated error due to the biased selection. Besides, Malach and Shalev-Shwartz [2017] proposed a decoupling approach to decouple "when to update" from "how to update" in training deep networks. Specifically, they maintain two networks, and update parameters of two networks only when the predictions of them disagree. However, the decoupling approach may not combat with massive noisy labels, since the disagreement area still overlap with the noise area. Although MentorNet and Decoupling are representative approaches in this promising direction, there still exist the above discussed issues, which naturally motivates us to improve them in our research.

Recent studies on the *memorization effects* of deep neural networks show that they would first memorize training data of clean labels and then those of noisy labels. Therefore in this chapter, we propose a new deep learning paradigm called "*Co-teaching*" for combating with noisy labels. Namely, we train two deep neural networks

simultaneously, and let them *teach each other* given every mini-batch: firstly, each network feeds forward all data and selects some data of possibly clean labels; secondly, two networks communicate with each other what data in this mini-batch should be used for training; finally, each network back propagates the data selected by its peer network and updates itself. Empirical results on noisy versions of *MNIST*, *CIFAR-10* and *CIFAR-100* demonstrate that Co-teaching is much superior to the state-of-the-art methods in the robustness of trained deep models.

The remainder of this chapter is organized as follows. Section 4.1.1 discusses two important questions related to our Co-teaching algorithm. Section 4.1.2 claims the key difference between our work and Co-training. Then, we prepare the experimental setup in Section 4.2, and conduct sufficient experiments on noisy *MNIST*, *CIFAR-10* and *CIFAR-100*. We analyze empirical results in Section 4.3. Besides, we display full Y-axis figures in Section 4.4. The conclusive remarks are given in Section 4.5.

4.1 Co-teaching Meets Noisy Supervision

Our idea is to train two deep networks simultaneously. As in Figure 1.2, in each mini-batch data, each network samples its small-loss instances as the useful knowledge, and teaches such useful instances to its peer network for the further training. Therefore, the proposed algorithm is named *Co-teaching* (Algorithm 4). As all deep learning training methods are based on stochastic gradient descent, our Co-teaching works in a mini-batch manner. Specifically, we maintain two networks f (with parameter w_f) and g (with parameter w_g). When a mini-batch $\bar{\mathcal{D}}$ is formed (step 3), we first let f (resp. g) select a small proportion of instances in this mini-batch $\bar{\mathcal{D}}_f$ (resp. $\bar{\mathcal{D}}_g$) that have small training loss (steps 4 and 5). The number of instances is controlled by $R(T)$, and f (resp. g) only samples $R(T)$ percentage of instances out of the mini-batch. Then, the selected instances are fed into its peer network as the useful knowledge for parameter updates (steps 6 and 7).

4.1.1 Two Important Questions

There are two important points for designing above Algorithm 4:

Q1. Why can sampling small-loss instances based on dynamic $R(T)$ help us find clean instances?

Q2. Why do we need two networks and cross-update the parameters?

To answer the *first question*, we need first clarify the connect between small losses and clean instances. Intuitively, when labels are correct, small-loss instances are more likely to be the ones which are correctly classified. Thus, when the classifier has the

Algorithm 4 Co-teaching Algorithm.

1: **Input** w_f and w_g, learning rate η, fixed τ, epoch T_k and $T_{\max}$, iteration $N_{\max}$;
for $T = 1, 2, \ldots, T_{\max}$ **do**
  2: **Shuffle** training set $\mathcal{D}$; //noisy dataset
  for $N = 1, \ldots, N_{\max}$ **do**
    3: **Fetch** mini-batch $\bar{\mathcal{D}}$ from $\mathcal{D}$;
    4: **Obtain** $\bar{\mathcal{D}}_f = \arg\min_{\mathcal{D}':|\mathcal{D}'|\geq R(T)|\bar{\mathcal{D}}|} \ell(f, \mathcal{D}')$; //sample $R(T)\%$ small-loss instances
    5: **Obtain** $\bar{\mathcal{D}}_g = \arg\min_{\mathcal{D}':|\mathcal{D}'|\geq R(T)|\bar{\mathcal{D}}|} \ell(g, \mathcal{D}')$; //sample $R(T)\%$ small-loss instances
    6: **Update** $w_f = w_f - \eta\nabla\ell(f, \bar{\mathcal{D}}_g)$; //update w_f by $\bar{\mathcal{D}}_g$;
    7: **Update** $w_g = w_g - \eta\nabla\ell(g, \bar{\mathcal{D}}_f)$; //update w_g by $\bar{\mathcal{D}}_f$;
  end
  8: **Update** $R(T) = 1 - \min\left\{\frac{T}{T_k}\tau, \tau\right\}$;
end
9: **Output** w_f **and** w_g**.**

ability to predict, if we drop large-loss instances in each mini-batch data, and only sample small-loss instances to train a single network, this single network should be resistant to the light noise.

However, the above requires us have a good classifier in advance. The "memorization" effect of deep networks [Arpit et al., 2017] can exactly help us address this problem. Namely, on noisy data sets, even with the existence of noisy labels, deep networks will learn clean and easy pattern first [Arpit et al., 2017; Zhang et al., 2017]. So, they have the ability to predict at the beginning, and the problem is that when the number of epochs goes large, they will eventually overfit on noisy labels. Thus, we want to keep more instances in the mini-batch at the start, i.e., $R(T)$ is large. Then, we gradually increase the drop rate, i.e., $R(T)$ becomes smaller, so that we can keep clean instances and drop those noisy ones (details of $R(T)$ will be discussed in Section 4.3.4).

Based on this idea, we can just use one network in Algorithm 4, and let the classifier evolves by itself. This process is similar to boosting [Freund and Schapire, 1995] and active learning [Cohn et al., 1996]. However, it is commonly known that boosting and active learning are sensitive to outliers and noise, a few wrongly selected instances can deteriorate the learning performance of the whole model [Balcan et al., 2009; Freund et al., 1999]. This connects with our *second question*, where two classifiers can help.

Intuitively, different classifiers can generate different decision boundaries and then have different abilities to learn. Thus, when training on noisy labels, we also hope that they can have different abilities to filter out the label noise. This motivates us to

exchange the selected instances, i.e., update parameters in f (resp. g) using mini-batch instances selected from g (resp. f). This process is similar to Co-training [Blum and Mitchell, 1998], and these two networks will adaptively correct the training error by the peer network if the selected instances are not fully clean. Take "peer-review" as a supportive example. When students checks their own exam papers, it is hard for them to find any errors or bugs because they have some personal bias for the answers. Luckily, they can ask peer classmates to review their papers, then it becomes much easier for them to find their potential faults. To sum up, as the error from one network will not be directly transferred back itself, we can expect that our Co-teaching method can deal with heavier noise compared with the self-evolving one.

4.1.2 Relations to Co-training

Although Co-teaching is motivated by Co-training, the only similarity is that two classifiers are trained. There are fundamental differences between them. (i). Co-training needs two views (two independent sets of features), while Co-teaching needs a single view. (ii) Co-training does not exploit the memorization of deep neural networks, while Co-teaching does. (iii) Co-training is designed for *semi-supervised learning* (SSL), and Co-teaching is for *learning with noisy labels* (LNL); as LNL is not a special case of SSL, we cannot simply translate Co-training for one problem setting to another problem setting.

4.2 Experimental Setup

4.2.1 Datasets

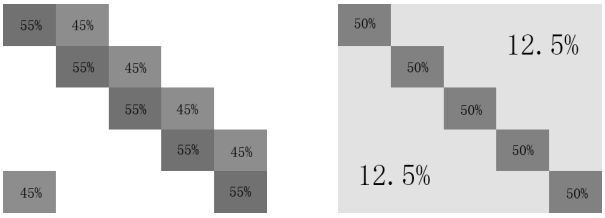
We verify the effectiveness of our approach on three benchmark datasets. *MNIST*, *CIFAR10* and *CIFAR100* are used here (Table 4.1), because these data sets are popularly used for evaluation of noisy labels in the literature [Goldberger and Ben-Reuven, 2017; Patrini et al., 2017; Reed et al., 2015].

Table 4.1: Summary of data sets used in the experiments.

	# of training	# of testing	# of class	image size
MNIST	60,000	10,000	10	28×28
CIFAR10	50,000	10,000	10	32×32
CIFAR100	50,000	10,000	100	32×32

Since all datasets are clean, following [Patrini et al., 2017; Reed et al., 2015], we need to corrupt these datasets manually by the noise transition matrix Q, where $Q_{ij} = \Pr(\tilde{y} = j|y = i)$ given that noisy $\tilde{y}$ is flipped from clean y. Assume that the matrix Q has two representative structures (Figure 4.1): (1) Symmetry flipping [van Rooyen

et al., 2015]; (2) Pair flipping: a real-world application is the fine-grained classification, where you may make mistake only within very similar classes in the adjunct positions.



(a) pair ($\epsilon = 45\%$). (b) symmetry ($\epsilon = 50\%$).

Figure 4.1: Transition matrices of different noise types (using 5 classes as an example).

Since this chapter mainly verifies the robustness of our Co-teaching on *extremely* noisy supervision, the noise rate ϵ is chosen from $\{0.45, 0.5\}$. Intuitively, this means almost half of the instances have noisy labels. Note that, the noise rate $> 50\%$ for pair flipping means over half of the training data have wrong labels that cannot be learned without additional assumptions. As a side product, we also verify the robustness of Co-teaching on *low-level* noisy supervision, where ϵ is set to 0.2. Note that pair case is much harder than symmetry case. In Figure 4.1(a), the true class only has 10% more correct instances over wrong ones. However, the true has 37.5% more correct instances in Figure 4.1(b).

The definition of transition matrix Q is as follow. n is number of the class.

$$\text{Pair flipping:} \quad Q = \begin{bmatrix} 1-\epsilon & \epsilon & 0 & \dots & 0 \\ 0 & 1-\epsilon & \epsilon & & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & & & 1-\epsilon & \epsilon \\ \epsilon & 0 & \dots & 0 & 1-\epsilon \end{bmatrix},$$

$$\text{Symmetry flipping:} \quad Q = \begin{bmatrix} 1-\epsilon & \frac{\epsilon}{n-1} & \dots & \frac{\epsilon}{n-1} & \frac{\epsilon}{n-1} \\ \frac{\epsilon}{n-1} & 1-\epsilon & \frac{\epsilon}{n-1} & \dots & \frac{\epsilon}{n-1} \\ \vdots & & \ddots & & \vdots \\ \frac{\epsilon}{n-1} & \dots & \frac{\epsilon}{n-1} & 1-\epsilon & \frac{\epsilon}{n-1} \\ \frac{\epsilon}{n-1} & \frac{\epsilon}{n-1} & \dots & \frac{\epsilon}{n-1} & 1-\epsilon \end{bmatrix}.$$

4.2.2 Baselines

Table 4.2: Comparison of state-of-the-art techniques with our Co-teaching approach. In the first column, "large noise": can deal with a large number of class; "heavy noise": can combat the heavy noise, i.e., high noise ratio; "flexibility": need not combine with specific network architecture; "no pre-train": can be train from scratch.

	Bootstrap	S-model	F-correction	Decoupling	MentorNet	Co-teaching
large class	×	×	×	✓	✓	✓
heavy noise	×	×	×	×	✓	✓
flexibility	×	×	✓	✓	✓	✓
no pre-train	✓	×	×	×	✓	✓

We compare the Co-teaching (Algorithm 4) with following state-of-art approaches: (i). Bootstrap [Reed et al., 2015]. It uses weighted combination of predicted and original labels as the correct labels, and then do back-propagate, hard labels are used as they yield better performance; (ii). S-model [Goldberger and Ben-Reuven, 2017]. It uses an additional softmax layer to model the noise transition matrix; (iii). F-correction [Patrini et al., 2017]. It corrects the prediction by the noise transition matrix. As suggested by the authors, we first train a normal network to estimate the transition matrix; (iv). Decoupling [Malach and Shalev-Shwartz, 2017]. It updates the parameters only using the samples of which have different prediction from two classifiers; and (v). MentorNet [Jiang et al., 2018]. An extra teacher network is pre-train, which is used to help sample clean instances during the training of the student network. Then, student network is used for classification. We used self-paced MentorNet in this chapter. (vi). As a baseline, we compare Co-teaching with the normal deep networks (abbreviated as Normal). As can be seen, our Co-teaching method does not rely on any specific network architectures, which can also deal with a large number of classes and is more robust to noise. Besides, it can be trained from scratch. These make our Co-teaching more appealing for practical usage. The availability of codes for compared approaches are as follows:

- Bootstrap [Reed et al., 2015].

  - https://github.com/emalach/UpdateByDisagreement

- S-model [Goldberger and Ben-Reuven, 2017].

  - https://github.com/udibr/noisy_labels

- F-correction [Patrini et al., 2017].

  - https://github.com/giorgiop/loss-correction

- Decoupling [Malach and Shalev-Shwartz, 2017].

  - https://github.com/emalach/UpdateByDisagreement

- MentorNet [Jiang et al., 2018]

  As the code is not available, we implement the code based on authors' description.

Note that all methods are not implemented in PyTorch and the network structure in all methods are not the same as our 9-layer CNN here, we change implementations based on authors' give codes.

4.2.3 Network Structure

For the fair comparison, we implement all methods with default parameters by PyTorch, and conduct all the experiments on a NIVIDIA K80 GPU. Standard CNN is used with LReLU active function, the detailed architecture is in Table 4.3. Namely, the 9-layer CNN architecture in this chapter follows "Temporal Ensembling" [Laine and Aila, 2017] and "Virtual Adversarial Training" [Miyato et al., 2016], since the network structure we used here are standard test bed for weakly-supervised learning.

Table 4.3: CNN and MLP models used in our experiments on *MNIST*, *CIFAR10*, and *CIFAR100*. The slopes of all LReLU functions in the networks are set to 0.01.

CNN on *MNIST*	CNN on *CIFAR10*	CNN on *CIFAR100*
28×28 Gray Image	32×32 RGB Image	32×32 RGB Image
3×3 conv, 128 LReLU 3×3 conv, 128 LReLU 3×3 conv, 128 LReLU		
2×2 max-pool, stride 2 dropout, $p = 0.25$		
3×3 conv, 256 LReLU 3×3 conv, 256 LReLU 3×3 conv, 256 LReLU		
2×2 max-pool, stride 2 dropout, $p = 0.25$		
3×3 conv, 512 LReLU 3×3 conv, 256 LReLU 3×3 conv, 128 LReLU		
avg-pool		
dense 128→10	dense 128→10	dense 128→100

For all datasets, Adam optimizer (momentum=0.9) with an initial learning rate of 0.001, the batch size is set to 128 and runs for 200 epoch. Besides, dropout and batch-normalization are also used. As deep networks are highly nonconvex, even with the

same network and optimization method, different initializations can lead to different local optimal. Thus, following [Malach and Shalev-Shwartz, 2017], we also take two networks with the same architecture but different initializations as two classifiers.

4.2.4 Noise Rates

Here, we assume the noise level ϵ is known and set $R(T) = 1 - \tau \cdot \min(T/T_k, 1)$ with $T_k = 10$ and $\tau = \epsilon$. If ϵ is not known in advanced, ϵ can be inferred using validation sets [Liu and Tao, 2016b; Yu et al., 2018a]. The choices of $R(T)$ and τ are analyzed in Section 4.3.4. Note that $R(T)$ only depends on the memorization effect of deep networks but not any specific datasets.

4.2.5 Measurements

As for performance measurements, first, we use the test accuracy, i.e., *test Accuracy = (# of correct predictions) / (# of test dataset)*. Besides, we also use the label precision in each mini-batch, i.e., *label Precision = (# of clean labels) / (# of all selected labels)*. Specifically, we sample $R(T)$ of small-loss instances in each mini-batch, and then calculate the ratio of clean labels in the small-loss instances. Intuitively, higher label precision means less noisy instances in the mini-batch after sample selection; and the algorithm with higher label precision is also more robust to the label noise. All experiments are repeated five times. The error bar for STD in each figure has been highlighted as a shade. Besides, the full Y-axis versions for all figures are in Section 4.4.

4.3 Empirical Results

4.3.1 Results on *MNIST*

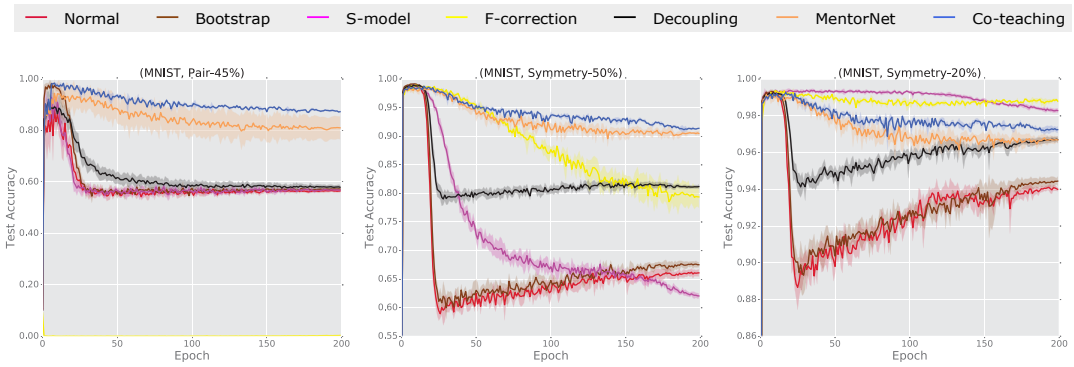
Table 4.4 report the accuracy on the testing set. As can be seen, on the symmetry case with 20% noisy rate, which is also the easiest case. All methods works well, even Normal can achieve 94.05% test set accuracy. Then, when noisy rate raises to 50%, Normal, Bootstrap, S-model and F-correction fail, their accuracy decrease lower than 80%. Methods based on "selected instances", i.e., Decoupling, MentorNet and Co-teaching, are better. Among them, Co-teaching is the best. Finally, in the hardest case, i.e., pair case with 45% noisy rate, Normal, Bootstrap and S-Model cannot learn anything, their testing accuracy keep the same as the percentage of clean instances in the training dataset. F-correct fails totally, it heavily relies on the correct estimation of the underneath transition matrix. Thus, when Normal works, it can worker better than Normal; then, when Normal fails, it works much worse than Normal. In this case, our

Co-teaching is again the best, which is also much better than the second method, i.e. 87.53% for Co-teaching vs 80.88% for MentorNet.

Table 4.4: Average test accuracy on *MNIST* over the last ten epoch.

Flipping-Rate	Normal	Bootstrap	S-model	F-correction	Decoupling	MentorNet	Co-teaching
Pair-45%	56.52% ±0.55%	57.23% ±0.73%	56.88% ±0.32%	0.24% ±0.03%	58.03% ±0.07%	80.88% ±4.45%	**87.63%** ±0.21%
Symmetry-50%	66.05% ±0.61%	67.55% ±0.53%	62.29% ±0.46%	79.61% ±1.96%	81.15% ±0.03%	90.05% ±0.30%	**91.32%** ±0.06%
Symmetry-20%	94.05% ±0.16%	94.40% ±0.26%	98.31% ±0.11%	**98.80%** ±0.12%	95.70% ±0.02%	96.70% ±0.22%	97.25% ±0.03%

In Figure 4.2 , we show test accuracy vs number of epochs. In all three plots, we can clearly see the memorization effects of networks, i.e., test accuracy of Normal first reaches a very high level and then gradually decreases. Thus, a good robust training method should stop or alleviate the decreasing processing. On this point, all method except Bootstrap works well in the easiest Symmetry-20% case. However, only MentorNet and our Co-teaching can combat with the other two harder cases, i.e., Pair-45% and Symmetry-50%. Besides, our Co-teaching consistently achieves higher accuracy MentorNet, and is the best method in these two cases.

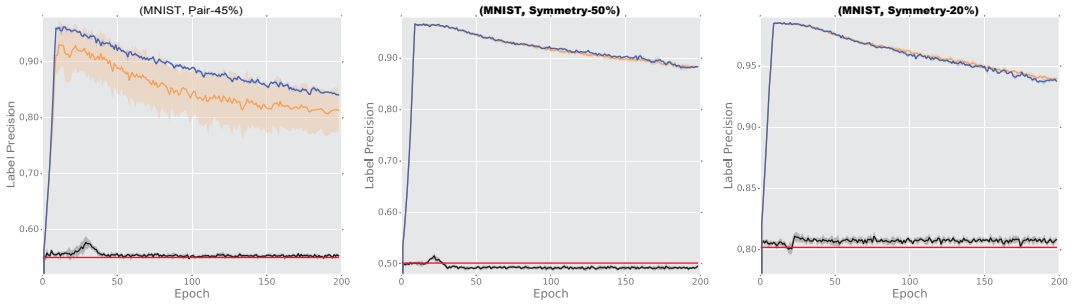


(a) Pair-45%. (b) Symmetry-50%. (c) Symmetry-20%.

Figure 4.2: Test accuracy vs number of epochs on *MNIST* dataset.

To explain such good performance, we plot label precision vs number of epochs in Figure 4.3. Only, MentorNet, Decoupling and Co-teaching are considered here, as they are methods do instance selecting during training. First, we can see Decoupling fails to pick up clean instances, its label precision is the same as Normal which does not compact with noisy label at all. The reason is that Decoupling does not utilize the memorization effects during the training. Then, we can see Co-teaching and MentorNet can successfully pick clean instances out. These two methods ties on the easier Symmetry-50% and Symmetry-20%, and our Co-teaching achieve higher

precision on the hardest Pair-45% case. This shows our approach is better at finding clean instances.



(a) Pair-45%. (b) Symmetry-50%. (c) Symmetry-20%.

Figure 4.3: Label precision vs number of epochs on *MNIST* dataset.

Finally, note that while in Figure 4.3(b) and (c), MentorNet and Co-teaching tie together. Co-teaching still gets higher testing accuracy (Table 4.4). Recall that MentorNet is a self-evolving method, which only uses one classifier, while Co-Teaching use two. The better accuracy comes from the fact Co-Teaching further takes the advantage of different learning abilities from two classifiers.

4.3.2 Results on *CIFAR10*

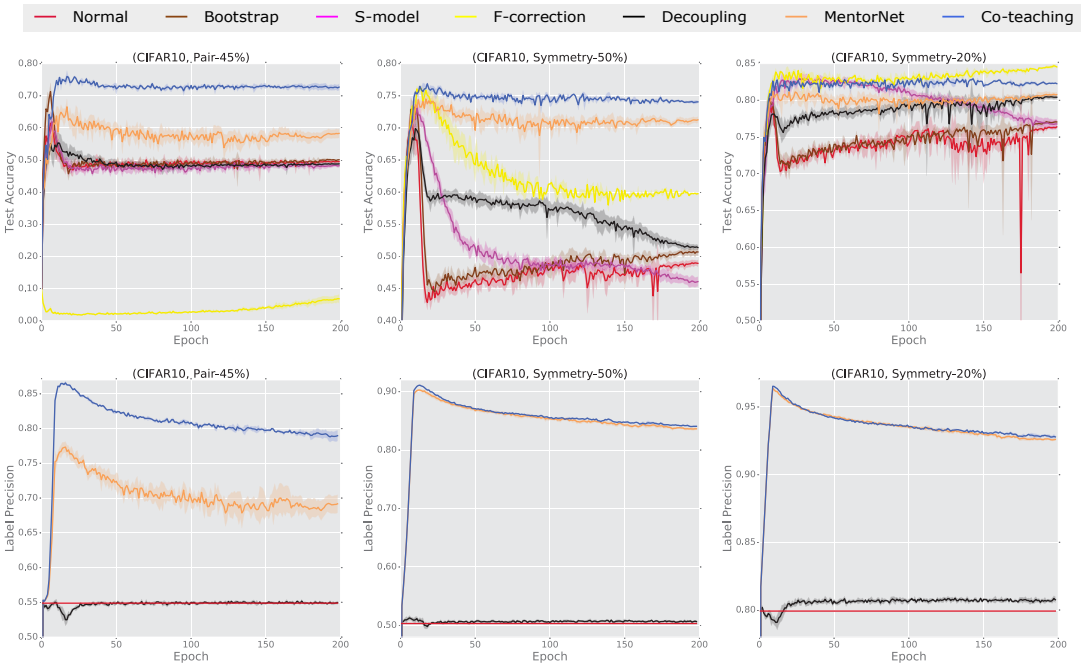
Test accuracy is shown in Table 4.5. As we can see, the observations here are consistently the same as these for *MNIST* dataset. In the easiest Symmetry-20% case, all methods works well. F-correction is the best, and our Co-teaching is comparable with F-correction. Then, all methods, except MentorNet and Co-teaching, fail on harder, i.e., Pair-45% and Symmetry-50% cases. Among these two, Co-teaching is the best. In the extreme Pair-45% case, Co-teaching is more than 14% higher than MentorNet in test accuracy.

Table 4.5: Average test accuracy on *CIFAR10* over the last ten epoch.

Flipping,Rate	Normal	Bootstrap	S-model	F-correction	Decoupling	MentorNet	Co-teaching
Pair-45%	49.50% ±0.42%	50.05% ±0.30%	48.21% ±0.55%	6.61% ±1.12%	48.80% ±0.04%	58.14% ±0.38%	**72.62%** ±0.15%
Symmetry-50%	48.87% ±0.52%	50.66% ±0.56%	46.15% ±0.76%	59.83% ±0.17%	51.49% ±0.08%	71.10% ±0.48%	**74.02%** ±0.04%
Symmetry-20%	76.25% ±0.28%	77.01% ±0.29%	76.84% ±0.66%	**84.55%** ±0.16%	80.44% ±0.05%	80.76% ±0.36%	82.32% ±0.07%

Figure 4.4 shows test accuracy and label precision vs number of epochs. Again, on test accuracy, we can see Co-teaching strongly stops the memorization effects

of networks. Thus, it works much better on the harder Pair-45% and Symmetry-50% cases. On label precision, while Decoupling fails to find clean instances, both MentorNet and Co-teaching can do this. However, due to the usage of two classifiers, Co-teaching is stronger.



(a) Pair-45%. (b) Symmetry-50%. (c) Symmetry-20%.

Figure 4.4: Results on *CIFAR10* dataset. Top: test accuracy vs number of epochs; bottom: label precision vs number of epochs.

4.3.3 Results on *CIFAR100*

Finally, we show our results on *CIFAR100*. The test accuracy is in Table 4.6, test accuracy and label precision vs number of epochs are in Figure 4.5. Note that, there are only 10 classes in *MNIST* and *CIFAR10* datasets. Thus, overall the accuracy is much lower than previous ones in Table 4.4 and 4.5. However, the observations are the same as previous datasets. We can clearly see our Co-teaching is the best on harder and noisy cases.

4.3.4 Choices of $R(T)$

Since deep networks initially fit clean (easy) instances, and then fit noisy (hard) instances progressively. Thus, intuitively $R(T)$ should meet following requirements:

Table 4.6: Average test accuracy on *CIFAR100* over the last ten epoch.

Flipping,Rate	Normal	Bootstrap	S-model	F-correction	Decoupling	MentorNet	Co-teaching
Pair-45%	31.99% ±0.64%	32.07% ±0.30%	21.79% ±0.86%	1.60% ±0.04%	26.05% ±0.03%	31.60% ±0.51%	**34.81%** ±0.07%
Symmetry-50%	25.21% ±0.64%	21.98% ±6.36%	18.93% ±0.39%	41.04% ±0.07%	25.80% ±0.04%	39.00% ±1.00%	**41.37%** ±0.08%
Symmetry-20%	47.55% ±0.47%	47.00% ±0.54%	41.51% ±0.60%	**61.87%** ±0.21%	44.52% ±0.04%	52.13% ±0.40%	54.23% ±0.08%

(i). $R(T) \in [\tau, 1]$, where τ depends on the noise rate ϵ; (ii). $R(1) = 1$, which means we do not need to drop any instances at the beginning. At the initial learning epochs, we can safely update the parameters of deep neural networks using entire noisy data, because the networks will not memorize the noisy data at the early stage [Arpit et al., 2017]; (iii). $R(T)$ should be a non-increasing function on T, which means that we need to drop more instances when the number of epochs gets large. This is because as the learning proceeds, the networks will eventually try to fit noisy data (which tends to have large losses compared to clean data). Thus, we need to ignore them by not updating the networks parameters using large loss instances [Arpit et al., 2017]. The MNIST dataset is used in the sequel.

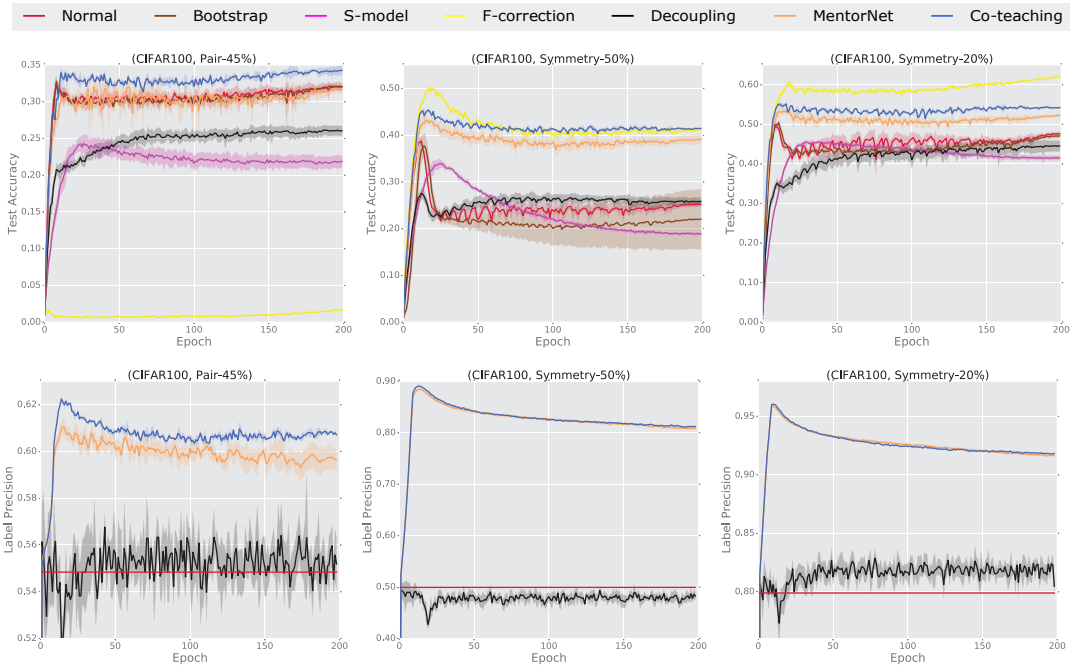
Based on above principles, to show how the decay of $R(T)$ affects Co-teaching, first, we let $R(T) = 1 - \tau \cdot \min\{T^c/T_k, 1\}$ with $\tau = \epsilon$, where three choices of c should be considered, i.e., $c = \{0.5, 1, 2\}$. Then, three values of T_k are considered, i.e., $T_k = \{5, 10, 15\}$. Results are in Table 4.7. As can be seen, the test accuracy is stable on the choices of T_k and c here. The previous setup ($c = 1$ and $T_k = 10$) works well but does not lead to the best performance.

Table 4.7: Average test accuracy on *MNIST* over the last ten epoch.

		$c = 0.5$	$c = 1$	$c = 2$
Pair-45%	$T_k = 5$	75.56%±0.33%	87.59%±0.26%	87.54%±0.23%
	$T_k = 10$	**88.43%±0.25%**	87.56%±0.12%	87.93%±0.21%
	$T_k = 15$	**88.37%±0.09%**	87.29%±0.15%	**88.09%±0.17%**
Symmetry-50%	$T_k = 5$	91.75%±0.13%	91.75%±0.12%	**92.20%±0.14%**
	$T_k = 10$	91.70%±0.21%	91.55%±0.08%	91.27%±0.13%
	$T_k = 15$	91.74%±0.14%	91.20%±0.11%	91.38%±0.08%
Symmetry-20%	$T_k = 5$	97.05%±0.06%	97.10%±0.06%	97.41%±0.08%
	$T_k = 10$	97.33%±0.05%	96.97%±0.07%	**97.48%±0.08%**
	$T_k = 15$	97.41%±0.06%	97.25%±0.09%	**97.51%±0.05%**

4.3.5 Choices of τ

Finally, to show the impact of τ, we vary $\tau = \{0.5, 0.75, 1, 1.25, 1.5\}\epsilon$. Note that, τ cannot be zero. In this case, no gradient will be back-propagated and the optimization will stop. Test accuracy is in Table 4.8. We can see, with more dropped instances, the



(a) Pair-45%. (b) Symmetry-50%. (c) Symmetry-20%.

Figure 4.5: Results on *CIFAR100* dataset. Top: test accuracy vs number of epochs; bottom: label precision vs number of epochs.

performance can be improved. However, if too many instances are dropped, networks may not get sufficiently trained and the performance can deteriorate. We set $\tau = \epsilon$ in Section 4.3.1, it works well but not necessarily leads to the best performance.

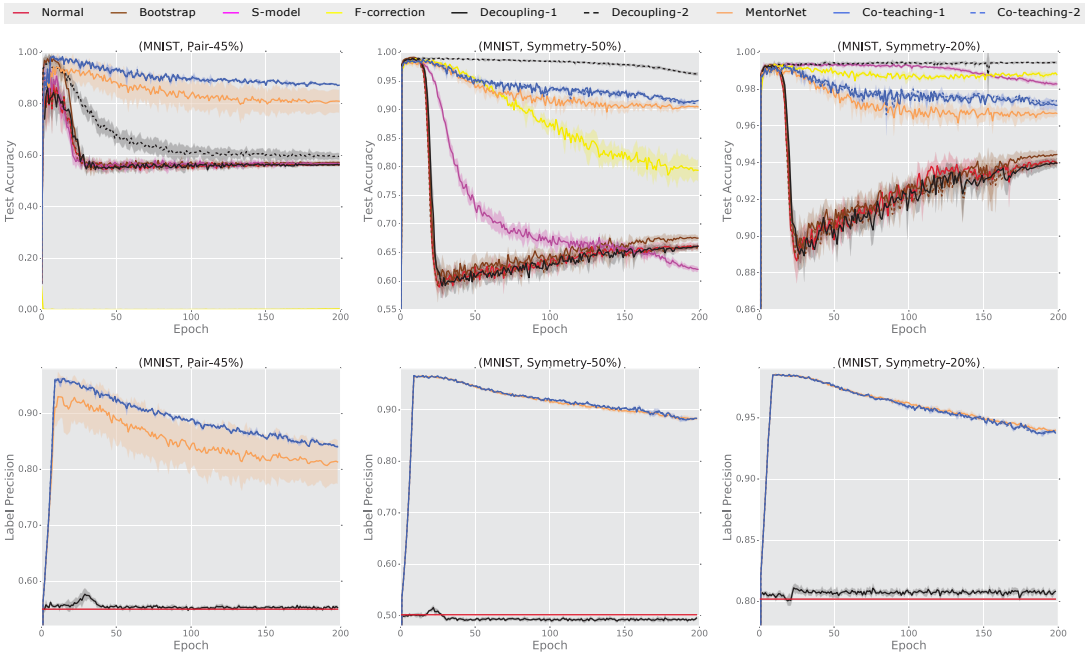
Table 4.8: Average test accuracy of Co-teaching with different τ on *MNIST* over the last ten epoch.

Flipping,Rate	0.5ϵ	0.75ϵ	ϵ	1.25ϵ	1.5ϵ
Pair-45%	66.74%±0.28%	77.86%±0.47%	87.63%±0.21%	**97.89%±0.06%**	69.47%±0.02%
Symmetry-50%	75.89%±0.21%	82.00%±0.28%	91.32%±0.06%	**98.62%±0.05%**	79.43%±0.02%
Symmetry-20%	94.94%±0.09%	96.25%±0.06%	97.25%±0.03%	98.90%±0.03%	**99.39%±0.02%**

4.4 Full Figures

Since Decoupling and Co-teaching are related to training two networks, we display the performance of both networks in this section (e.g., Co-teaching-1 and Co-teaching-2).

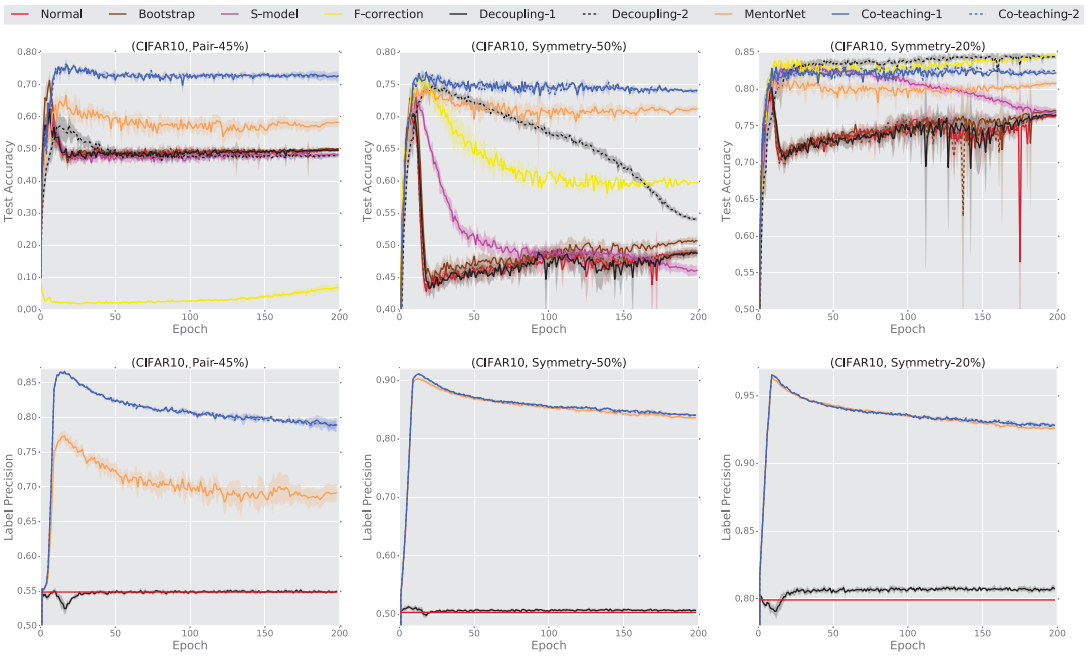
4.4.1 *MNIST*



(a) Pair-45%. (b) Symmetirc-50%. (c) Symmetirc-20%.

Figure 4.6: Results on *MNIST* dataset. Top: test accuracy vs number of epochs; bottom: label precision vs number of epochs.

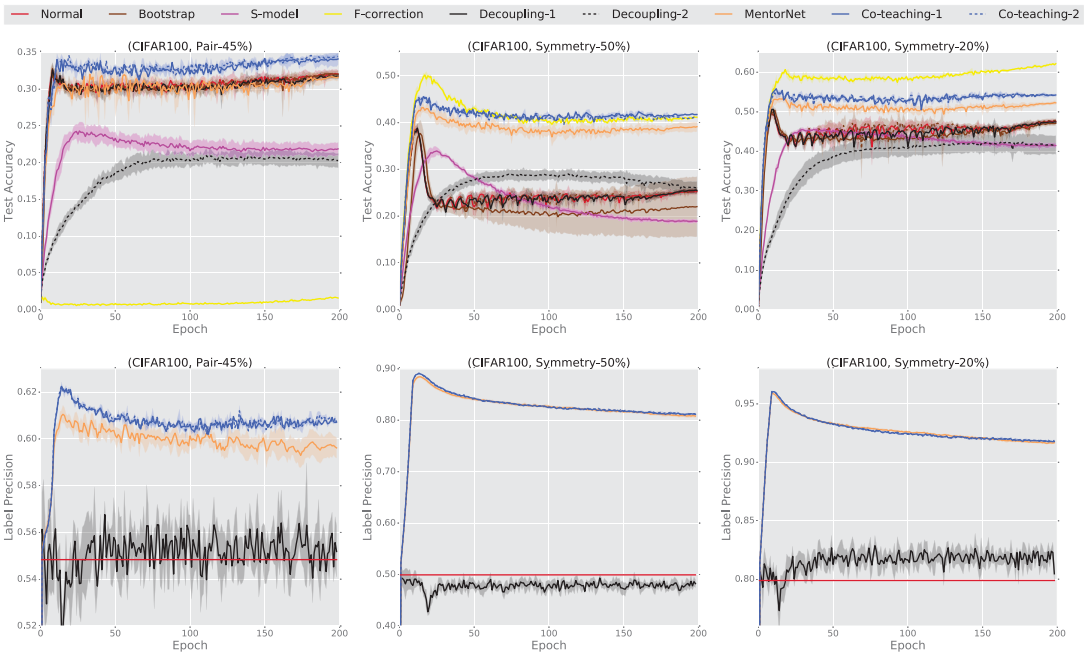
4.4.2 *CIFAR10*



(a) Pair-45%. (b) Symmetirc-50%. (c) Symmetirc-20%.

Figure 4.7: Results on *CIFAR10* dataset. Top: test accuracy vs number of epochs; bottom: label precision vs number of epochs.

4.4.3 *CIFAR100*



(a) Pair-45%. (b) Symmetirc-50%. (c) Symmetirc-20%.

Figure 4.8: Results on *CIFAR100* dataset. Top: test accuracy vs number of epochs; bottom: label precision vs number of epochs.

4.5 Summary of This Chapter

This chapter presentd a simple but effective learning paradigm called Co-teaching, which trains deep neural networks robustly. Our key idea is to maintain two networks simultaneously, and cross-trains on instances screened by the "small loss" criteria. We conduct simulated experiments to demonstrate that, our proposed Co-teaching can train deep models robustly with the extremely noisy supervision.

Chapter 5

Conclusion and Future Work

In this section, we first conclude the entire thesis, and then elaborate the possible trends for future research.

5.1 Thesis Summarization

This thesis addresses the problem of learning from noisy supervision. Real-world data is often noisy, which inevitably degrades the performance of learning models. Therefore, we designed various robust machine learning methodologies, which can reduce the negative effects of the noisy data from three orthogonal aspects. Namely, robust mechanism, optimization and generalization were proposed.

In robust mechanism (Chapter 2), to improve the label quality, we proposed a hint-guided approach that encourages workers to use hints when they answer unsure questions. Our approach consists of the hybrid-stage setting and the hint-guided payment mechanism. We proved the incentive compatibility and uniqueness of our mechanism. Besides, our approach can detect the high-quality workers for more accurate result aggregation. These merits are critical for the success of many machine learning applications in practice.

In robust optimization (Chapter 3), we studied a robust SGD mechanism called progressive stochastic learning (POSTAL) for the label noise problem. POSTAL incorporates the progressive learning paradigm of curriculum learning (CL) with the update process of vanilla SGD. Through the learning process of CL, POSTAL yields robust updates of the primal variable on an ordered label sequence, namely from "reliable" labels to "noisy" labels. To provide this ordered label sequence, we design a cluster of screening losses, where all labels are sorted from the reliable region to the noisy region. Our analysis derives the convergence rate of POSTAL realized by screening losses, while demonstrates the robustness of screening losses.

In robust generalization (Chapter 4), we presented a simple but effective learning paradigm called Co-teaching, which trains deep neural networks robustly. Our key idea is to maintain two networks simultaneously, and cross-trains on instances screened by the "small loss" criteria.

5.2 Future Work

Although the algorithms proposed in this thesis have achieved encouraging results to some extent, some issues still remain open and should be further investigated.

For robust mechanism, we summarize two potential research as follows.

- **Coarse-to-fine hints.** We consider to provide hints from different levels for all questions. Specifically, we will provide the hints from coarse to fine, which corresponds the different expected payments. When the worker is a little bit unsure of some questions, she is encouraged to select the coarse hints, which corresponds to the higher payment. When the worker is moderately unsure of some questions, she is encouraged to select the fine hints, which corresponds to the lower payment.

- **Mixture of hints and unsure options.** Moreover, some workers may still be very confused even with hints, we may mix up the unsure option in the hint stage to further improve the label quality further. For each question, when the worker is unsure of some questions, she is encouraged to select the hints; when the worker is extremely unsure of some questions, she is encouraged to skip these questions. The payment of the mixture situation is consistent with the payment of coarse to fine hints.

For robust optimization, we summarize two potential research as follows.

- **Extending POSTAL to deep learning models.** Since both SVM and Deep Learning are under the same empirical risk minimization (ERM) principle (1.2), the POSTAL mechanism can be leveraged by Deep Learning model if we represent f_w by deep neural networks. Namely, if we realize f_w using the nonlinear mapping function (deep neural networks) in line 6, then POSTAL could be applied to Deep Learning model.

- **Faster convergence of POSTAL.** Another key problem in POSTAL mechanism is how to speed up the convergence rate. We can leverage the classical Nesterov's accelerated strategy [Nesterov, 2007], or the strategy of iterative machine teaching [Liu et al., 2017, 2018]. Empirical results will help us decide which strategy can balance both the convergence rate and the generalization most.

For robust generalization, we summarize two potential research as follows.

- **Adapting Co-teaching to another weakly-supervised settings.** We can adapt Co-teaching paradigm to train deep models under another weak supervision, e.g., positive and unlabeled data [Kiryo et al., 2017].

- **Theoretical guarantee for Co-teaching.** We would investigate the theoretical guarantees for Co-teaching. Previous theories for Co-training are very hard to transfer into Co-teaching, since our setting is fundamentally different. Besides, there is no analysis for generalization performance on deep learning with noisy labels. Thus, we leave the generalization analysis as a future work.

All developed methods in this thesis focus on handling noisy single-label data. However, in many real-world scenario, each data can simultaneously own multiple labels, i.e., one scene image with multiple annotations. Consequently, our algorithms should be adapted to handle noisy multi-label data by considering the structure among multiple labels. Moreover, the next challenge in label-noise learning should be how to handle instance-dependent label noise without strong assumptions. Thus, we should adapt our algorithms to handle such practical noise. Last but not least, the target of our research is to apply our fundamental algorithms into practical applications. For example, the quality of electronic health records (EHR) data is often not satisfactory, and even extremely noisy. Therefore, we can adapt our Co-teaching algorithms for EHR data, which can train robust prediction models.

References

A. Agarwal, S. Negahban, and M. Wainwright. Fast global convergence of gradient methods for high-dimensional statistical recovery. *The Annals of Statistics*, 40(5): 2452–2482, 2012. 4

A. Agarwal, D. Foster, D. Hsu, S. Kakade, and A. Rakhlin. Stochastic convex optimization with bandit feedback. *SIAM Journal on Optimization*, 23(1):213, 2013. 2

D. Arpit, S. Jastrzkbski, N. Ballas, D. Krueger, E. Bengio, M. Kanwal, T. Maharaj, A. Fischer, A. Courville, and Y. Bengio. A closer look at memorization in deep networks. In *International Conference on Machine Learning*, 2017. 2, 3, 5, 14, 67, 76

S. Azadi, J. Feng, S. Jegelka, and T. Darrell. Auxiliary image regularization for deep cnns with noisy labels. In *International Conference on Learning Representations*, 2016. 10

M. Balcan, A. Beygelzimer, and J. Langford. Agnostic active learning. *Journal of Computer and System Sciences*, 75(1):78–89, 2009. 67

P. Bartlett, M. Jordan, and J. McAuliffe. Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101(473):138–156, 2006. 9

Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In *International Conference on Machine Learning*, pages 41–48, 2009. 9, 12, 53

W. Bi, L. Wang, J. Kwok, and Z. Tu. Learning to predict from crowdsourced data. In *Conference on Uncertainty in Artificial Intelligence*, pages 82–91, 2014. 7

A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Annual Conference on Computational Learning Theory*, pages 92–100, 1998. 12, 68

A. Blum, A. Kalai, and H. Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model. *Journal of the ACM*, 50(4):506–519, 2003. 2

M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. Jackel, M. Monfort, U. Muller, and J. Zhang. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016. 1

L. Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010. 8, 57

M. Buhrmester, T. Kwang, and S. Gosling. Amazon's Mechanical Turk a new source of inexpensive, yet high-quality, data? *Perspectives on Psychological Science*, 6(1): 3–5, 2011. 7

C. Chang and C. Lin. Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):27, 2011. 57

O. Chapelle, B. Scholkopf, and A. Zien. Semi-supervised learning. *IEEE Transactions on Neural Networks*, 20(3):542–542, 2009. 3

D. Cohn, Z. Ghahramani, and M. Jordan. Active learning with statistical models. *Journal of Artificial Intelligence Research*, 4:129–145, 1996. 67

R. Collobert, F. Sinz, J. Weston, and L. Bottou. Trading convexity for scalability. In *International Conference on Machine Learning*, pages 201–208, 2006. 9

A. Cotter, O. Shamir, N. Srebro, and K. Sridharan. Better mini-batch algorithms via accelerated gradient methods. In *Advances in Neural Information Processing Systems*, pages 1647–1655, 2011. 2

A. Dawid and A. Skene. Maximum likelihood estimation of observer error-rates using the em algorithm. *Applied Statistics*, pages 20–28, 1979. 7

J. Deng, J. Krause, and L. Fei-Fei. Fine-grained crowdsourcing for fine-grained recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587, 2013. 12

D. Difallah, G. Demartini, and P. Cudre-Mauroux. Mechanical cheat: Spamming schemes and adversarial techniques on crowdsourcing platforms. In *CrowdSearch*, pages 26–30, 2012. 2

Y.-X. Ding and Z.-H. Zhou. Crowdsourcing with unsure option. *Machine Learning*, pages 1–18, 2017. xiii, 7, 18, 34

J. Fan, G. Li, B. Ooi, K. Tan, and J. Feng. iCrowd: An adaptive crowdsourcing framework. In *ACM Special Interest Group on Management of Data*, pages 1015–1030, 2015. 8

R. Fan, K. Chang, C. Hsieh, X. Wang, and C. Lin. Liblinear: A library for large linear classification. *Journal of Machine Learning Research*, 9(Aug):1871–1874, 2008. 56

Y. Fan, F. Tian, T. Qin, J. Bian, and T. Liu. Learning to teach. In *International Conference on Learning Representations*, 2018. 11

Y. Freund and R. Schapire. A desicion-theoretic generalization of on-line learning and an application to boosting. In *European Conference on Computational Learning Theory*, pages 23–37, 1995. 67

Y. Freund, R. Schapire, and N. Abe. A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, 14(771-780):1612, 1999. 67

S. Ghadimi and G. Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013. 8

S. Ghadimi and G. Lan. Accelerated gradient methods for nonconvex nonlinear and stochastic programming. *Mathematical Programming*, 156(1-2):59–99, 2016. 9

G. Goel, A. Nikzad, and A. Singla. Mechanism design for crowdsourcing markets with heterogeneous tasks. In *AAAI Conference on Human Computation and Crowdsourcing*, 2014. 7

J. Goldberger and E. Ben-Reuven. Training deep neural-networks using a noise adaptation layer. In *International Conference on Learning Representations*, 2017. 2, 10, 68, 70

C. Gong, D. Tao, S. Maybank, W. Liu, G. Kang, and J. Yang. Multi-modal curriculum learning for semi-supervised image classification. *IEEE Transactions on Image Processing*, 25(7):3249–3260, 2016a. 9

C. Gong, D. Tao, J. Yang, and W. Liu. Teaching-to-learn and learning-to-teach for multi-label propagation. In *Association for the Advancement of Artificial Intelligence*, pages 1610–1616, 2016b. 11

B. Han, I. Tsang, and L. Chen. On the convergence of a family of robust losses for stochastic gradient descent. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 665–680, 2016. 2, 5, 10, 48, 53

G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, and T. Sainath. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97, 2012. 1

G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 11

C. Ho, A. Slivkins, S. Suri, and J. Vaughan. Incentivizing high quality crowdwork. In *World Wide Web*, pages 419–429, 2015. 7

H. Hu, Y. Zheng, Z. Bao, G. Li, J. Feng, and R. Cheng. Crowdsourced poi labelling: Location-aware result inference and task assignment. In *International Conference on Data Engineering*, pages 61–72, 2016. 8

P. Ipeirotis, F. Provost, and J. Wang. Quality management on Amazon Mechanical Turk. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining workshop*, pages 64–67, 2010. 1

L. Jiang, D. Meng, S. Yu, Z. Lan, S. Shan, and A. Hauptmann. Self-paced learning with diversity. In *Advances in Neural Information Processing Systems*, pages 2078–2086, 2014. 9

L. Jiang, Z. Zhou, T. Leung, L. Li, and L. Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *International Conference on Machine Learning*, pages 2309–2318, 2018. x, 2, 5, 10, 11, 14, 65, 70, 71

Y. Jin, M. Carman, Y. Zhu, and W. Buntine. Distinguishing question subjectivity from difficulty for improved crowdsourcing. In *Asian Conference on Machine Learning*, pages 192–207, 2018. 6

M. Joglekar, H. Garcia-Molina, and A. Parameswaran. Evaluating the crowd with confidence. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 686–694, 2013. 8

H. Kajino, Y. Tsuboi, and H. Kashima. A convex formulation for learning from crowds. *Transactions of the Japanese Society for Artificial Intelligence*, 27(3), 2012. 7

D. Karger, S. Oh, and D. Shah. Iterative learning for reliable crowdsourcing systems. In *Advances in Neural Information Processing Systems*, pages 1953–1961, 2011. 7

R. Kiryo, G. Niu, M. Du Plessis, and M. Sugiyama. Positive-unlabeled learning with non-negative risk estimator. In *Advances in Neural Information Processing Systems*, pages 1675–1685, 2017. 83

K. Koedinger and V. Aleven. Exploring the assistance dilemma in experiments with cognitive tutors. *Educational Psychology Review*, 19(3):239–264, 2007. 21, 23

M. Kumar, B. Packer, and D. Koller. Self-paced learning for latent variable models. In *Advances in Neural Information Processing Systems*, pages 1189–1197, 2010. 9, 12

S. Laine and T. Aila. Temporal ensembling for semi-supervised learning. In *International Conference on Learning Representations*, 2017. 71

N. Lambert, J. Langford, J. Vaughan, Y. Chen, D. Reeves, Y. Shoham, and D. Pennock. An axiomatic characterization of wagering mechanisms. *Journal of Economic Theory*, 156:389–416, 2015. 7

Q. Le, J. Ngiam, A. Coates, A. Lahiri, B. Prochnow, and A. Ng. On optimization methods for deep learning. In *International Conference on Machine Learning*, pages 265–272, 2011. 8

G. Li, J. Wang, Y. Zheng, and M. Franklin. Crowdsourced data management: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 28(9):2296–2319, 2016a. 1

G. Li, C. Chai, J. Fan, X. Weng, J. Li, Y. Zheng, Y. Li, X. Yu, X. Zhang, and H. Yuan. Cdb: Optimizing queries with crowd-based selections and joins. In *ACM SIGMOD International Conference on Management of Data*, pages 1463–1478, 2017a. 1

G. Li, Y. Zheng, J. Fan, J. Wang, and R. Cheng. Crowdsourced data management: Overview and challenges. In *ACM SIGMOD International Conference on Management of Data*, pages 1711–1716, 2017b. 1

H. Li and B. Yu. Error rate bounds and iterative weighted majority voting for crowdsourcing. *arXiv preprint arXiv:1411.4086*, 2014. 7

S. Li and Y. Fu. Learning robust and discriminative subspace with low-rank constraints. *IEEE Transactions on Neural Networks and Learning Systems*, 27(11):2160–2173, 2016. 9

X. Li, T. Zhao, R. Arora, H. Liu, and J. Haupt. Stochastic variance reduced optimization for nonconvex sparse learning. In *International Conference on Machine Learning*, pages 917–925, 2016b. 4

Y. Li, J. Yang, Y. Song, L. Cao, J. Luo, and J. Li. Learning from noisy labels with distillation. In *International Conference on Computer Vision*, pages 1928–1936, 2017c. 11

Q. Liu, J. Peng, and A. Ihler. Variational inference for crowdsourcing. In *Advances in Neural Information Processing Systems*, pages 692–700, 2012. 7

T. Liu and D. Tao. Classification with noisy labels by importance reweighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(3):447–461, 2016a. 9

T. Liu and D. Tao. Classification with noisy labels by importance reweighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(3):447–461, 2016b. 10, 72

W. Liu, B. Dai, A. Humayun, C. Tay, Y. Chen, L. Smith, J. Rehg, and L. Song. Iterative machine teaching. In *International Conference on Machine Learning*, pages 2149–2158, 2017. 82

W. Liu, B. Dai, X. Li, J. Rehg, and L. Song. Towards black-box iterative machine teaching. In *International Conference on Machine Learning*, 2018. 82

P. Loh and M. Wainwright. Regularized m-estimators with nonconvexity: statistical and algorithmic theory for local optima. *Journal of Machine Learning Research*, 16: 559–616, 2015. 4

E. Malach and S. Shalev-Shwartz. Decoupling "when to update" from "how to update". In *Advances in Neural Information Processing Systems*, pages 960–970, 2017. x, 2, 11, 14, 65, 70, 71, 72

H. Masnadi-Shirazi and N. Vasconcelos. On the design of loss functions for classification: theory, robustness to outliers, and savageboost. In *Advances in Neural Information Processing Systems*, pages 1049–1056, 2009. 10

A. Menon, B. Van Rooyen, C. Ong, and B. Williamson. Learning from corrupted binary labels via class-probability estimation. In *International Conference on Machine Learning*, pages 125–134, 2015. 10

I. Mitliagkas, C. Caramanis, and P. Jain. Memory limited, streaming pca. In *Advances in Neural Information Processing Systems*, pages 2886–2894, 2013. 2

T. Miyato, A. Dai, and I. Goodfellow. Virtual adversarial training for semi-supervised text classification. In *International Conference on Learning Representations*, 2016. 71

N. Natarajan, I. Dhillon, P. Ravikumar, and A. Tewari. Learning with noisy labels. In *Advances in Neural Information Processing Systems*, pages 1196–1204, 2013. 2, 9, 10, 57

N. Natarajan, I. Dhillon, P. Ravikumar, and A. Tewari. Cost-sensitive learning with noisy labels. *Journal of Machine Learning Research*, 18:155–1, 2017. 9

Y. Nesterov. Gradient methods for minimizing composite objective function, 2007. 82

Y. Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2012. 2

N. Nisan, T. Roughgarden, E. Tardos, and V. Vazirani. *Algorithmic Game Theory*, volume 1. Cambridge University Press Cambridge, 2007. 18, 21, 25

L. Niu, W. Li, and D Xu. Visual recognition by learning from web data: A weakly supervised domain generalization approach. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2774–2783, 2015. 5

G. Patrini, F. Nielsen, R. Nock, and M. Carioni. Loss factorization, weakly supervised learning and label noise robustness. In *International Conference on Machine Learning*, pages 708–717, 2016. 9, 56, 57

G. Patrini, A. Rozza, A. Menon, R. Nock, and L. Qu. Making deep neural networks robust to label noise: a loss correction approach. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2, 10, 68, 70

A. Rakotomamonjy, R. Flamary, and G. Gasso. Dc proximal newton for nonconvex optimization problems. *IEEE Transactions on Neural Networks and Learning Systems*, 27(3):636–647, 2016. 9

V. Raykar and S. Yu. Eliminating spammers and ranking annotators for crowdsourced labeling tasks. *Journal of Machine Learning Research*, 13(Feb):491–518, 2012. 2, 8

V. Raykar, S. Yu, L. Zhao, G. Valadez, C. Florin, L. Bogoni, and L. Moy. Learning from crowds. *Journal of Machine Learning Research*, 11:1297–1322, 2010a. 7

V. Raykar, S. Yu, L. Zhao, G. Valadez, C. Florin, L. Bogoni, and L. Moy. Learning from crowds. *Journal of Machine Learning Research*, 11(Apr):1297–1322, 2010b. 9, 10

S. Reed, H. Lee, D. Anguelov, C. Szegedy, D. Erhan, and A. Rabinovich. Training deep neural networks on noisy labels with bootstrapping. In *International Conference on Learning Representations*, 2015. 10, 68, 70

M. Ren, W. Zeng, B. Yang, and R. Urtasun. Learning to reweight examples for robust deep learning. In *International Conference on Machine Learning*, 2018. 2, 11

F. Rodrigues and F. Pereira. Deep learning from crowds. In *Association for the Advancement of Artificial Intelligence*, 2018. 11

F. Rodrigues, F. Pereira, and B. Ribeiro. Sequence labeling with multiple annotators. *Machine learning*, 95(2):165–181, 2014. 1

O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, and F. F. Li. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 1

T. Sanderson and C. Scott. Class proportion estimation with application to multiclass anomaly rejection. In *Artificial Intelligence and Statistics*, pages 850–858, 2014. 10

N. Shah and D. Zhou. Double or nothing: Multiplicative incentive mechanisms for crowdsourcing. In *Advances in Neural Information Processing Systems*, pages 1–9, 2015. xiii, 7, 18, 21, 25, 30, 34, 35, 38, 57

N. Shah and D. Zhou. No Oops, You Won't Do it again: Mechanisms for self-correction in crowdsourcing. In *International Conference on Machine Learning*, 2016. xiii, 7, 8, 18, 21, 23, 25, 33, 34, 35

S. Shalev-Shwartz, Y. Singer, N. Srebro, and A. Cotter. Pegasos: Primal estimated sub-gradient solver for svm. *Mathematical programming*, 127(1):3–30, 2011. 9, 56

D. Silver, A. Huang, C. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, and M. Lanctot. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016. 1

A. Singla and A. Krause. Truthful incentives in crowdsourcing tasks using regret minimization mechanisms. In *World Wide Web*, pages 1167–1178, 2013. 7

J. Smith. *Qualitative psychology: A practical guide to research methods*. Sage, 2007. 21

A. Sordoni, M. Galley, M. Auli, C. Brockett, Y. Ji, M. Mitchell, J. Nie, J. Gao, and B. Dolan. A neural network approach to context-sensitive generation of conversational responses. In *Conference of the Association for Computational Linguistics*, pages 196–205, 2015. 1

S. Sukhbaatar, J. Bruna, M. Paluri, L. Bourdev, and R. Fergus. Training convolutional networks with noisy labels. In *International Conference on Learning Representations Workshop*, 2015a. 10

S. Sukhbaatar, J. Bruna, M. Paluri, L. Bourdev, and R. Fergus. Training convolutional networks with noisy labels. In *International Conference on Learning Representations workshop*, 2015b. 2

D. Tanaka, D. Ikami, T. Yamasaki, and K. Aizawa. Joint optimization framework for learning with noisy labels. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 11

Q. Tao, Q. Gao, D. Chu, and G. Wu. Stochastic learning via optimizing the variational inequalities. *IEEE Transactions Neural Networks and Learning Systems*, 25(10): 1769–1778, 2014. 8

T. Tian and J. Zhu. Max-margin majority voting for learning from crowds. In *Advances in Neural Information Processing Systems*, pages 1621–1629, 2015. 7

B. van Rooyen, A. Menon, and B. Williamson. Learning with symmetric label noise: The importance of being unhinged. In *Advances in Neural Information Processing Systems*, pages 10–18, 2015. 68

A. Veit, N. Alldrin, G. Chechik, I. Krasin, A. Gupta, and S. Belongie. Learning from noisy large-scale datasets with minimal supervision. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 6575–6583, 2017. 11

J. Vuurens, A. Vries, and C. Eickhoff. How much spam can you take? an analysis of crowdsourcing results to increase accuracy. In *ACM Special Interest Group on Information Retrieval workshop*, pages 21–26, 2011. 1, 2

P. Wais, S. Lingamneni, D. Cook, J. Fennell, B. Goldenberg, D. Lubarov, D. Marin, and H. Simons. Towards building a high-quality workforce with mechanical turk. In *Advances in Neural Information Processing Systems workshop*, 2010a. 1

P. Wais, S. Lingamneni, D. Cook, J. Fennell, B. Goldenberg, D. Lubarov, D. Marin, and H. Simons. Towards building a high-quality workforce with mechanical turk. *NIPS Workshop on Computational Social Science and The Wisdom of Crowds*, pages 1–5, 2010b. 2

L. Wang and Z.-H. Zhou. Cost-ssving effect of crowdsourcing learning. In *International Joint Conference on Artificial Intelligence*, pages 2111–2117, 2016. 1

L. Wang, H. Jia, and J. Li. Training robust support vector machine with smooth ramp loss in the primal space. *Neurocomputing*, 71(13-15):3020–3025, 2008. 9

W. Wang, X.-Y. Guo, S.-Y. Li, Y. Jiang, and Z.-H. Zhou. Obtaining high-quality label by distinguishing between easy and hard items in crowdsourcing. In *International Joint Conference on Artificial Intelligence*, pages 2964–2970, 2017. 1

C. Xiao, E. Choi, and J. Sun. Opportunities and challenges in developing deep learning models using electronic health records data: a systematic review. *Journal of the American Medical Informatics Association*, 25(10):1419–1428, 2018. 5

W. Xu. Towards optimal one pass large scale learning with averaged stochastic gradient descent. *arXiv preprint arXiv:1107.2490*, 2011. 8, 56

Y. Yan, R. Rosales, G. Fung, M. Schmidt, G. Hermosillo, L. Moy, and J. Dy. Modeling annotator expertise: Learning when everybody knows a bit of something. In *International Conference on Artificial Intelligence and Statistics*, pages 932–939, 2010. 7

Y. Yan, R. Rosales, G. Fung, and J. Dy. Active learning from crowds. In *International Conference on Machine Learning*, volume 11, pages 1161–1168, 2011. 23

Y. Yan, R. Rosales, G. Fung, R. Subramanian, and J. Dy. Learning from multiple annotators with varying expertise. *Machine learning*, 95(3):291–327, 2014. 1, 2, 10

T. Yang, M. Mahdavi, R. Jin, L. Zhang, and Y. Zhou. Multiple kernel learning from noisy labels by stochastic programming. In *International Conference on Machine Learning*, 2012. 57

J. Yi, R. Jin, S. Jain, T. Yang, and A. Jain. Semi-crowdsourced clustering: Generalizing crowd labeling by robust distance metric learning. In *Advances in Neural Information Processing Systems*, pages 1772–1780, 2012. 5

X. Yu, T. Liu, M. Gong, K. Zhang, and D. Tao. Transfer learning with label noise. *arXiv preprint arXiv:1707.09724*, 2017. 2

X. Yu, T. Liu, M. Gong, K. Batmanghelich, and D. Tao. An efficient and provable approach for mixture proportion estimation using linear independence assumption. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4480–4489, 2018a. 72

X. Yu, T. Liu, M. Gong, and D. Tao. Learning with biased complementary labels. In *European Conference on Computer Vision*, 2018b. 2

X. Yu, T. Liu, M. Gong, and D. Tao. Learning with biased complementary labels. In *European Conference on Computer Vision*, 2018c. 2

C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations*, 2017. 2, 3, 5, 67

J. Zhang, X.-D. Wu, and V. Sheng. Learning from crowdsourced labeled data: A survey. *Artificial Intelligence Review*, 46(4):543–576, 2016. 6, 21

L. Zhao, G. Sukthankar, and R. Sukthankar. Incremental relabeling for active learning with noisy crowdsourced annotations. In *SocialCom/PASSAT*, pages 728–733, 2011. 57

Y. Zheng, R. Cheng, S. Maniu, and L. Mo. On optimality of jury selection in crowdsourcing. In *International Conference on Extending Database Technology*, 2015a. 6

Y. Zheng, J. Wang, G. Li, R. Cheng, and J. Feng. QASCA: A quality-aware task assignment system for crowdsourcing applications. In *ACM Special Interest Group on Management of Data*, pages 1031–1046, 2015b. 8

Y. Zheng, G. Li, and R. Cheng. Docs: A domain-aware crowdsourcing system using knowledge bases. *Very Large Data Base Endowment*, 10(4):361–372, 2016. 6

Y. Zheng, G. Li, Y. Li, C. Shan, and R. Cheng. Truth inference in crowdsourcing: is the problem solved? *Very Large Data Base Endowment*, 10(5):541–552, 2017. 6

J.-H. Zhong, K. Tang, and Z.-H. Zhou. Active learning from crowds with unsure option. In *International Joint Conference on Artificial Intelligence*, pages 1061–1068, 2015. 1

D. Zhou, S. Basu, Y. Mao, and J. Platt. Learning from the wisdom of crowds by minimax entropy. In *Advances in Neural Information Processing Systems*, pages 2195–2203, 2012a. 7

D. Zhou, S. Basu, Y. Mao, and J. Platt. Learning from the wisdom of crowds by minimax entropy. In *Advances in Neural Information Processing Systems*, pages 2195–2203, 2012b. 63

D. Zhou, Q. Liu, J. Platt, and C. Meek. Aggregating ordinal labels from crowds by minimax conditional entropy. In *International Conference on Machine Learning*, pages 262–270, 2014. 7