Constrained Labeling for Weakly Supervised Learning

Chidubem Arachie[1] **Bert Huang**[2]

[1]Department of Computer Science, Virginia Tech, Blacksburg, Virginia, USA
[2]Department of Computer Science, Data Intensive Studies Center, Tufts University, Medford, Massachusetts, USA

Abstract

Curation of large fully supervised datasets has become one of the major roadblocks for machine learning. Weak supervision provides an alternative to supervised learning by training with cheap, noisy, and possibly correlated labeling functions from varying sources. The key challenge in weakly supervised learning is combining the different weak supervision signals while navigating misleading correlations in their errors. In this paper, we propose a simple data-free approach for combining weak supervision signals by defining a constrained space for the possible labels of the weak signals and training with a random labeling within this constrained space. Our method is efficient and stable, converging after a few iterations of gradient descent. We prove theoretical conditions under which the worst-case error of the randomized label decreases with the rank of the linear constraints. We show experimentally that our method outperforms other weak supervision methods on various text- and image-classification tasks.

1 INTRODUCTION

Recent successful demonstrations of machine learning have created an explosion of interest. The key driver of these successes is the progress in deep learning. Researchers in different fields and industries are applying deep learning to their work with varying degrees of success. Training deep learning models typically requires massive amounts of data, and in most cases this data needs to be labeled for supervised learning. The process of collecting labels for large training datasets is often expensive and can be a major bottleneck for practical machine learning.

To enable machine learning when labeled data is not available, researchers are increasingly turning to weak supervision. Weakly supervised learning involves training models using noisy labels. Using multiple sources or forms of weak supervision is common, as it provides diverse information to the model. However, each source of weak supervision has its own bias that can be transmitted to the model. Different weak supervision signals can also conflict, overlap, or—in the worst case—make dependent errors. Thus, a naive combination of these weak signals would hurt the quality of a learned model. The key problem then is how to reliably combine various sources of weak signals to train an accurate model.

To solve this problem, we propose *constrained label learning* (CLL), a method that processes various weak supervision signals and combines them to produce high-quality training labels. The idea behind CLL is that, given the weak supervision, we can define a constrained space for the labels of the unlabeled examples. The space will contain the true labels of the data, and any other label sampled from the space should be sufficient to train a model. We construct this space using the expected error of the weak supervision signals, and then we select a random vector from this space to use as training labels. Our analysis shows that, the space of labels considered by CLL improves to be tighter around the true labels as we include more information in the weak signals and that CLL is not confounded by redundant weak signals.

CLL takes as input (1) a set of unlabeled data examples, (2) multiple weak supervision signals that label a subset of data and can abstain from labeling the rest, and (3) a corresponding set of expected error rates for the weak supervision signals. While the weak supervision signals can abstain on various examples, we require that the combination of the weak signals have full coverage on the training data. The expected error rates can be estimated if the weak supervision signals have been tested on historical data or a domain expert has knowledge about their performance. In cases where the expected error rates are unavailable, they can be treated as a hyperparameter. Our experiments in Section 3.3 show that CLL is still effective when it is trained

Accepted for the 37th Conference on Uncertainty in Artificial Intelligence (UAI 2021).

with a loose estimate of the weak signals. Alternatively, we provide guidelines on how error rates can be estimated.

We implement CLL as a stable, quickly converging, convex optimization over the candidate labels. CLL thus scales much better than many other weak supervision methods. We show in Section 4 experiments that compare the performance of CLL to other weak supervision methods. On a synthetic dataset, CLL trained with a constant error rate is only a few percentage points from matching the performance of supervised learning on a test set. On real text and image classification tasks, CLL achieves superior performance over existing weak supervision methods on test data.

2 RELATED WORK

Weakly supervised learning has gained prominence in recent years due to the need to train models without access to manually labeled data. The recent success of deep learning has exacerbated the need for large-scale data annotation, which can be prohibitively expensive. One weakly supervised paradigm, data programming, allows users to define *labeling functions* that noisily label a set of unlabeled data Bach et al. [2019], Ratner et al. [2017, 2016]. Data programming then combines the noisy labels to form probabilistic labels for the data by using a generative model to estimate the accuracies and dependencies of the noisy/weak supervision signals. This approach underlies the popular software package *Snorkel* Ratner et al. [2017]. Our method is related to this approach in that we use different weak signal sources and compile them into a single (soft) labeling. However, unlike Snorkel's methods, we do not train a generative model and avoid the need for probabilistic modeling assumptions. Recently, Snorkel MeTaL was proposed for solving multi-task learning problems with hierarchical structure Ratner et al. [2018]. A user provides weak supervision for the hierarchy of tasks which is then combined in an end-to-end framework.

Another recently developed approach for weakly supervised learning is adversarial label learning (ALL) Arachie and Huang [2019b]. ALL was developed for training binary classifiers from weak supervision. ALL trains a model to perform well in the worst case for the weak supervision by simultaneously optimizing model parameters and adversarial labels for the training data in order to satisfy the constraint that the error of the weak signals on the adversarial labels be within provided error bounds. The authors also recently proposed Stoch-GALL Arachie and Huang [2019a], an extension for multi-class classification that incorporates precision bounds. Our work is related to ALL and Stoch-GALL in that we use the same error definition the authors introduced. However, the expected errors we use do not serve as upper bound constraints for the weak signals. Additionally, CLL avoids the adversarial setting that requires unstable simultaneous optimization of the estimated labels and the model parameters. Lastly, while ALL and Stoch-GALL require weak supervision signals to label every example, we allow for weak supervision signals that abstain on different data subsets.

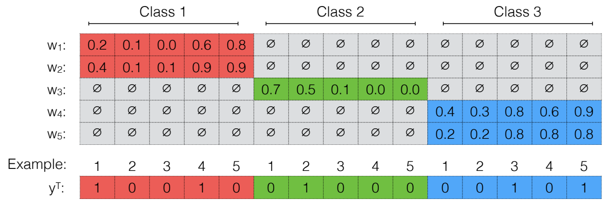
Crowdsourcing has become relevant to machine learning practitioners as it provide a means to train machine learning models using labels collected from different crowd workers Carpenter [2008], Gao et al. [2011], Karger et al. [2011], Khetan et al. [2017], Liu et al. [2012], Platanios et al. [2020], Zhou et al. [2015], Zhou and He [2016]. The key machine learning challenge when crowdsourcing is to effectively combine the different labels obtained from human annotators. Our work is similar in that we try to combine different weak labels. However, unlike most methods for crowdsourcing, we cannot assume that the labels are independent of each other. Instead, we train the model to learn while accounting for dependencies between the various weak supervision signals.

Ensemble methods such as boosting Schapire et al. [2002] combine different weak learners (low-cost, low-powered classifiers) to create classifiers that outperform the various weak learners. These weak learners are not weak in the same sense as weak supervision. These strategies are defined for fully supervised settings. Although recent work has proposed leveraging unlabeled data to improve the accuracies of boosting methods Balsubramani and Freund [2015], our settings differs since we do not expect to have access to labeled data.

A growing set of weakly supervised applications includes web knowledge extraction Bunescu and Mooney [2007], Hoffmann et al. [2011], Mintz et al. [2009], Riedel et al. [2010], Yao et al. [2010], visual image segmentation Chen et al. [2014], Xu et al. [2014], and tagging of medical conditions from health records Halpern et al. [2016]. As better weakly supervised methods are developed, this set will expand to include other important applications.

We will show an estimation method that is connected to those developed to estimate the error of classifiers without labeled data Dawid and Skene [1979], Jaffe et al. [2016], Madani et al. [2005], Platanios et al. [2014, 2016], Steinhardt and Liang [2016]. These methods rely on statistical relationships between the error rates of different classifiers or weak signals. Unlike these methods, we show in our experiments that we can train models even when we do not learn the error rates of classifiers. We show that using a maximum error estimate of the weak signals, CLL learns to accurately classify.

Like our approach, many other methods incorporate human knowledge or side information into a learning objective. These methods, including posterior regularization Druck et al. [2008] and generalized expectation (GE) criteria and its variants Mann and McCallum [2008, 2010], can be used



	Class 1					Class 2					Class 3				
w1:	0.2	0.1	0.0	0.6	0.8	∅	∅	∅	∅	∅	∅	∅	∅	∅	∅
w2:	0.4	0.1	0.1	0.9	0.9	∅	∅	∅	∅	∅	∅	∅	∅	∅	∅
w3:	∅	∅	∅	∅	∅	0.7	0.5	0.1	0.0	0.0	∅	∅	∅	∅	∅
w4:	∅	∅	∅	∅	∅	∅	∅	∅	∅	∅	0.4	0.3	0.8	0.6	0.9
w5:	∅	∅	∅	∅	∅	∅	∅	∅	∅	∅	0.2	0.2	0.8	0.8	0.8
Example:	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
yT:	1	0	0	1	0	0	1	0	0	0	0	0	1	0	1

Figure 1: Illustration of weak signals and label vectorized structure. For multi-class problems, we arrange the label vector so that it contains indicators for each example belonging to each class. The weak signals use the same indexing scheme. In this illustration, weak signals $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ estimate the probability of each example belonging to class 1 and abstain on estimating membership in all other classes.

for semi- and weakly supervised learning. They work by providing parameter estimates as constraints to the objective function of the model so that the label distribution of the trained model tries to match the constraints. In our approach, we incorporate human knowledge as error estimates into our algorithm. However, we do not use the constraints for model training. Instead, we use them to generate training labels that satisfy the constraints, and these labels can then be used downstream to train any model.

3 CONSTRAINED LABEL LEARNING

The goal of *constrained label learning* (CLL) is to return accurate training labels for the data given the weak supervision signals. The estimation of these labels should be aware of the correlation among the weak supervision signals and should not be confounded by it. Toward this goal, we use the weak signals' expected error to define a constrained space of possible labelings for the data. Any vector sampled from this space can then be used as training labels. We consider the setting in which the learner has access to a training set of unlabeled examples, and a set of weak supervision signals from various sources that provide approximate indicators of the target classification for the data. Along with the weak supervision signals, we are provided estimates of the expected error rates of the weak signals. Formally, let the data be $X = [x_1, \ldots, x_n]$. These examples have corresponding labels $\boldsymbol{y} = [y_1, \ldots, y_n] \in \{0, 1\}^n$. For multi-label classification, where each example may be labeled as a member of K classes, we expand the label vector to include an entry for each example-class combination, i.e., $\boldsymbol{y} = [y_{(1,1)}, \ldots, y_{(n,1)}, y_{(1,2)}, \ldots, y_{(n-1,K)}, y_{(n,K)}]$, where y_{ij} is the indicator of whether the ith example is in class j.[1] See Fig. 1 for an illustration of this arrangement.

With weak supervision, the training labels $\boldsymbol{y}$ are unavailable. Instead, we have access to m weak supervision signals $\{\boldsymbol{w}_1, \ldots, \boldsymbol{w}_m\}$, where each weak signal $\boldsymbol{w} \in [\emptyset, 0, 1]^n$ is represented as a vector of indicators that each example is in each class. The weak signals can choose to abstain on some examples. In that case, they assign a null value $\emptyset$ to that example's entry. In practice, weak signals for multi-class problems typically only label one class at a time, such as a one-versus-rest classification rule, so they effectively abstain on all out-of-class entries. The weak signals can be soft labels (probabilities) or hard labels (class assignments) of the data. In conjunction with the weak signals, the learner also receives the expected error rates of the weak signals $\boldsymbol{\epsilon} = [\epsilon_1, \ldots, \epsilon_m]$. In practice, the error rates of the weak signals are estimated or treated as a hyperparameter. The expected empirical error of a weak signal $\boldsymbol{w}_i$ is

$$\begin{aligned}\epsilon_i &= \frac{1}{n_i}\left(\mathbf{1}_{(\boldsymbol{w}\neq\emptyset)}\boldsymbol{w}_i^\top(1-\boldsymbol{y}_k) + \mathbf{1}_{(\boldsymbol{w}\neq\emptyset)}(1-\boldsymbol{w}_i)^\top\boldsymbol{y}_k\right)\\ &= \frac{1}{n_i}\left(\mathbf{1}_{(\boldsymbol{w}\neq\emptyset)}(1-2\boldsymbol{w}_i)^\top\boldsymbol{y}_k + \boldsymbol{w}_i^\top\mathbf{1}_{(\boldsymbol{w}\neq\emptyset)}\right),\end{aligned} \tag{1}$$

where $\boldsymbol{y}_k$ is the true label for the class k that the weak signal $\boldsymbol{w}_i$ labels, $n_i = \sum \mathbf{1}_{(\boldsymbol{w}_i\neq\emptyset)}$ and $\mathbf{1}_{(\boldsymbol{w}_i\neq\emptyset)}$ is an indicator function that returns 1 on examples the weak signals label (i.e., do not abstain on). Hence, we only calculate the error of the weak signals on the examples they label.

Analogously to Eq. (1), we can express the expected error of all weak signals for the label vector as a system of linear equations in the form $\boldsymbol{A}\boldsymbol{y} = \boldsymbol{c}$. To do this, we define each row in $\boldsymbol{A}$ as

$$\boldsymbol{A}_i = \mathbf{1}_{(\boldsymbol{w}_i\neq\emptyset)}(1-2\boldsymbol{w}_i), \tag{2}$$

a linear transformation of a weak signal $\boldsymbol{w}$. Each entry in the vector $\boldsymbol{c}$ is the difference between the expected error of the weak signal and the sum of the weak signal, i.e.,

$$\boldsymbol{c_i} = n_i\epsilon_i - \boldsymbol{w}_i^\top\mathbf{1}_{(\boldsymbol{w}\neq\emptyset)}. \tag{3}$$

Valid label vectors then must be in the space

$$\{\tilde{\boldsymbol{y}}|\boldsymbol{A}\tilde{\boldsymbol{y}} = \boldsymbol{c} \wedge \tilde{\boldsymbol{y}} \in [0,1]^n\}\ . \tag{4}$$

The true label $\boldsymbol{y}$ is not known. Thus, we want to find training labels $\tilde{\boldsymbol{y}}$ that satisfy the system of linear equations.

3.1 ALGORITHM

Having defined the space of possible labelings for the data given the weak signals, we explain here how we efficiently sample a vector of training labels from the space. First, we initialize a random $\tilde{\boldsymbol{y}}$ from a uniform distribution $\tilde{\boldsymbol{Y}} \sim U(0,1)^n$. Then we minimize a quadratic penalty on violations of the constraints defining the space. The objective function is

$$\min_{\tilde{\boldsymbol{y}}\in[0,1]^n} \ \|\boldsymbol{A}\tilde{\boldsymbol{y}} - \boldsymbol{c}\|_2^2\ . \tag{5}$$

[1] We represent the labels as a vector for later notational convenience, even though it may be more naturally arranged as a matrix.

Algorithm 1 Constrained Label Learning

```
Require: Weak signals $[\boldsymbol{w}_1, \ldots, \boldsymbol{w}_m]$, and expected error $\boldsymbol{\epsilon} = [\epsilon_1, \ldots, \epsilon_m]$ for the signals.
Define $\boldsymbol{A}$ from Eq. (2) and $\boldsymbol{c}$ from Eq. (3) using the weak signals and expected errors.
Initialize $\tilde{\boldsymbol{y}}$ as $\tilde{\boldsymbol{y}} \sim U(0,1)^n$
while not converged do
    Update $\tilde{\boldsymbol{y}}$ with its gradient from Eq. (5)
    Clip $\tilde{\boldsymbol{y}}$ to $[0,1]^n$
end while
return estimated labels $\tilde{\boldsymbol{y}}$
```

The solution to this quadratic objective function gives us feasible labels for the training data. In our experiments, we estimate the error rates $\boldsymbol{\epsilon}$ of the weak signals. In cases where the error estimates make an infeasible space, this quadratic penalty acts as a squared slack. We solve Eq. (5) iteratively using projected Adagrad Duchi et al. [2011], clipping $\tilde{\boldsymbol{y}}$ values to $[0,1]^n$ between gradient updates. This approach is fast and efficient, even for large datasets. Our algorithm is a simple quadratic convex optimization that converges to a unique optimum for each initialization of $\tilde{\boldsymbol{y}}$. In our experiments, it converges after only a few iterations of gradient descent. We run the algorithm 3 times with random initialization of $\tilde{\boldsymbol{y}}$ and take the mean of the $\tilde{\boldsymbol{y}}$s as the estimated label. We observed that the labels returned from the different runs are very similar. We fix the number of iterations of gradient descent for each run to 200 for all our experiments. The full algorithm is summarized in Algorithm 1.

3.2 ANALYSIS

We start by analyzing the case where we have the true error $\boldsymbol{\epsilon}$, in which case the true label vector $\boldsymbol{y}$ for CLL is a solution in the feasible space. Although the true error rates are not available in practice, this ideal setting is the motivating case for the CLL approach. To begin the analysis, consider an extreme case: if $\boldsymbol{A}$ is a square matrix with full rank, then the only valid label $\tilde{\boldsymbol{y}}$ in the space is the true label, $\tilde{\boldsymbol{y}} = \boldsymbol{y}$. Normally, $\boldsymbol{A}$ is usually underdetermined, which means we have more data examples than weak signals. In this case, there are many solutions for $\tilde{\boldsymbol{y}}$, so we can analyze this space to understand how distant any feasible vector is from the vector of all incorrect labels. Since label vectors are constrained to be in the unit box, the farthest possible label vector from the true labels is $(1 - \boldsymbol{y})$. The result of our analysis is the following theorem, which addresses the binary classification case with non-abstaining weak signals.

Theorem 1. *For any $\tilde{\boldsymbol{y}} \in [0,1]^n$ such that $\boldsymbol{A}\tilde{\boldsymbol{y}} = \boldsymbol{c}$, its Euclidean distance from the negated label vector $(1 - \boldsymbol{y}) \in \{0,1\}^n$ is bounded below by*

$$||\tilde{\boldsymbol{y}} - (1 - \boldsymbol{y})|| \geq n||\boldsymbol{A}^+(1 - 2\boldsymbol{\epsilon})||, \tag{6}$$

where $\boldsymbol{A}^+$ is the Moore-Penrose pseudoinverse of $\boldsymbol{A}$.

Proof. We first relax the constrained space by removing the $[0,1]^n$ box constraints. We can then analyze the projection onto the feasible space:

$$\min_{\tilde{\boldsymbol{y}}} ||(1 - \boldsymbol{y}) - \tilde{\boldsymbol{y}}|| \text{ s.t. } \boldsymbol{A}\tilde{\boldsymbol{y}} = \boldsymbol{c}. \tag{7}$$

Define a vector $\boldsymbol{z} := \tilde{\boldsymbol{y}} - \boldsymbol{y}$. We can rewrite the distance as

$$\min_{\boldsymbol{z}} ||(1 - 2\boldsymbol{y}) - \boldsymbol{z}|| \text{ s.t. } \boldsymbol{A}\boldsymbol{z} = 0. \tag{8}$$

The minimization is a projection of $(1 - 2y)$ onto the null space of $\boldsymbol{A}$. Since the null and row spaces of a matrix are complementary, $(1 - 2\boldsymbol{y})$ decomposes into

$$(1 - 2\boldsymbol{y}) = \mathbb{P}_{\text{row}}(1 - 2\boldsymbol{y}) + \mathbb{P}_{\text{null}}(1 - 2\boldsymbol{y}),$$

where $\mathbb{P}_{\text{row}}$ and $\mathbb{P}_{\text{null}}$ are orthogonal projections into the row and null spaces of $\boldsymbol{A}$, respectively. We can use this decomposition to rewrite the distance of interest:

$$\begin{aligned} &||(1 - 2\boldsymbol{y}) - \mathbb{P}_{\text{null}}(1 - 2\boldsymbol{y})|| \\ &= ||(1 - 2\boldsymbol{y}) - ((1 - 2\boldsymbol{y}) - \mathbb{P}_{\text{row}}(1 - 2\boldsymbol{y}))|| \\ &= ||\,\mathbb{P}_{\text{row}}(1 - 2\boldsymbol{y})||. \end{aligned} \tag{9}$$

For any vector $\boldsymbol{v}$, its projection into the row space of matrix $\boldsymbol{A}$ is $\boldsymbol{A}^+\boldsymbol{A}\boldsymbol{v}$, where $\boldsymbol{A}^+$ is the Moore-Penrose pseudoinverse of $\boldsymbol{A}$. The distance of interest is thus $||\boldsymbol{A}^+\boldsymbol{A}(1-2\boldsymbol{y})||$. We can use the definition of $\boldsymbol{A}$ to further simplify. Let $\boldsymbol{W}$ be the matrix of weak signals $\boldsymbol{W} = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_m]^\top$. Then the distance is

$$\begin{aligned} &||A^+(1 - 2\boldsymbol{W})(1 - 2\boldsymbol{y})|| \\ &= ||A^+((1 - 2\boldsymbol{W})\vec{1}_n - 2(1 - 2\boldsymbol{W})\boldsymbol{y})|| \\ &= ||A^+(n - 2\boldsymbol{W}\vec{1}_n - 2\boldsymbol{A}\boldsymbol{y})||. \end{aligned} \tag{10}$$

Because $\boldsymbol{A}\boldsymbol{y} = \boldsymbol{c} = n\boldsymbol{\epsilon} - \boldsymbol{W}\vec{1}_n$, terms cancel, yielding the bound in the theorem:

$$\begin{aligned} &||A^+(n - 2\boldsymbol{W}\vec{1}_n - 2n\boldsymbol{\epsilon} + 2\boldsymbol{W}\vec{1}_n)|| \\ &= ||A^+(n - 2n\boldsymbol{\epsilon})|| = n||A^+(1 - 2\boldsymbol{\epsilon})||. \end{aligned} \tag{11}$$

□

This bound provides a quantity that is computable in practice. However, to gain an intuition about what factors affect its value, the distance formula can be further analyzed by using the singular-value decomposition (SVD) formula for the pseudoinverse. Consider SVD $\boldsymbol{A} = \boldsymbol{U}\Sigma\boldsymbol{V}^\top$. Then $\boldsymbol{A}^+ = \boldsymbol{V}\Sigma^+\boldsymbol{U}^\top$, where the pseudoinverse Σ^+ contains the reciprocal of all nonzero singular values along the diagonal (and zeros elsewhere). The distance simplifies to

$$n||\boldsymbol{V}\Sigma^+\boldsymbol{U}^\top(1 - 2\boldsymbol{\epsilon})|| = n||\Sigma^+\boldsymbol{U}^\top(1 - 2\boldsymbol{\epsilon})||, \tag{12}$$

since $\boldsymbol{V}$ is orthonormal. Furthermore, let $\boldsymbol{p} = \boldsymbol{U}^\top(1 - 2\boldsymbol{\epsilon})$, i.e., $\boldsymbol{p}$ is a rotation of the centered error rates of the weak

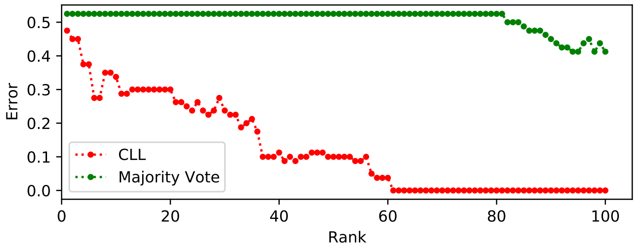


Figure 2: Error of CLL estimated labels compared to majority vote as we increase the rank of $\boldsymbol{A}$ by replacing redundant weak signals with linearly independent weak signals.

signals with the same norm as $(1 - 2\boldsymbol{\epsilon})$. From this change of variables, we can decompose the distance into

$$n||\Sigma^{+}\boldsymbol{p}|| = n\sqrt{\sigma_1^2 p_1^2 \ + \ldots + \ \sigma_m^2 p_m^2}, \qquad (13)$$

where σ_j is the jth singular value of $\boldsymbol{A}^{+}$.

As this distance grows toward $\sqrt{n}$, the space of possible labelings shrinks toward zero, at which point the only feasible label vectors are close to the true labels $\boldsymbol{y}$. Equation (13) indicates that the distance increases roughly as the rank of $\boldsymbol{A}$ increases, in which case the number of non-zero singular values in Σ^{+} increases, irrespective of how many actual weak signals are given. Thus, redundancy in the weak supervision does not affect the performance of CLL. The other key factor in the distance is how far from 0.5 the errors $\boldsymbol{\epsilon}$ are. These quantities can be interpreted as the diversity and number of the weak signals (corresponding to the rank) and their accuracies (the magnitude of $\boldsymbol{p}$).

Though the analysis is for length-n label vectors, it is straightforwardly extended to multi-label settings with length-(nK). And with careful indexing and tracking of the abstaining indicators, the same form of analysis can apply for abstaining weak signals.

Figure 2 shows an empirical validation of Theorem 1 on a synthetic experiment. We plot the error of the labels returned by CLL and majority voting as we change the rank of $\boldsymbol{A}$. We use a synthetic data for a binary classification task with 100 randomly generated examples containing 20 binary features. The weak signals are random binary predictions for the labels where each weak signal error rate is calculated using the true labels of the data. We start with 100 redundant weak signals by generating a matrix $\boldsymbol{A}$ whose 100 columns contain copies of the same weak signal, giving it a rank of 1. We then iteratively increase the rank of $\boldsymbol{A}$ by replacing copies of the weak signal with random vectors from the uniform distribution. The error of CLL labels approaches zero as the rank of the matrix increases while the majority vote error does not improve significantly.

3.3 ERROR ESTIMATION

In our analysis, we assume that the expected error rates of the weak signals are available. This may be the case if the weak signals have been evaluated on historical data or if an expert provides the error rates. In practice, users typically define weak supervision signals whose error rates are unknown. In this section, we discuss two approaches to handle such situations. We test these estimation techniques on real and synthetic data in our experiments, finding that CLL with these strategies forms a powerful weakly supervised approach.

3.3.1 Agreement Rate Method

Estimating the error rates of binary classifiers using their agreement rates was first proposed by Platanios et al. [2014]. They propose two different objective functions for solving the error rates of classifiers using their agreement rates as constraints. Similar to MeTaL Ratner et al. [2018], we solve a matrix-completion problem to find a low-rank factorization for the weak signal accuracies. We assume that if the weak signals are conditionally independent, we can relate the disagreement rates to the weak signal accuracies. We implemented this method and report its performance in our synthetic experiment (see Section 4). The one-vs-all form of the weak signals on our real datasets violates the assumption that each weak signal makes prediction on all the classes, so we cannot use the agreement rate method on our real data.

3.3.2 Uniform Error Rate

The idea of using uniform error rates of the weak signals was first proposed in ALL Arachie and Huang [2019b]. Their experiments showed that ALL can learn as effectively as when using true error rates by using a constant for the error rates of all the weak signals on their binary classification datasets. We use this approach in our experiments and extend it to weak supervision signals that abstain and also on multi-class datasets. Figure 3 plots the accuracy of generated labels as we increase the error-rate parameter. On the binary-class SST-2 dataset, the label accuracy remains similar if the error rate is set between 0 and 0.5 and drops for values at least 0.5. On the multiclass Fashion-MNIST data, we notice similar behavior where the label accuracies are similar between 0.05 and 0.1 and drop with larger values. We surmise that this behavior mirrors the type of weak supervision signals we use in our experiments. The weak signals in our real experiments are one-vs-all signals; hence a baseline signal (guessing 0 on all examples) will have an error rate of $\frac{1}{K}$. Performance deteriorates when the error rate is worse than this baseline rate.

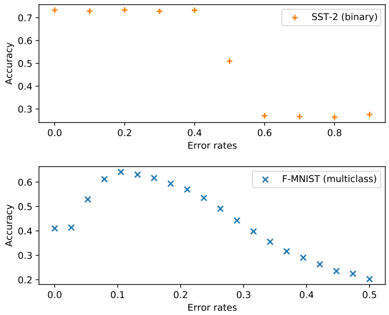


Figure 3: Accuracy of constrained label learning as we increase the error rates from 0 to 1 on binary and 0 to 0.5 on multiclass datasets (SST-2 and Fashion-MNIST).

Method	Test Accuracy
CLL (Agr. rate ϵ)	**0.668**± 0.005
CLL (Constant ϵ)	0.630± 0.009
Data Programming	0.504± 0.000
Majority Vote	0.504± 0.000
CLL (True ϵ)	0.675 ± 0.024
Supervised Learning	0.997± 0.001

Table 1: Classification accuracies of the different methods on synthetic data using dependent weak signals. We report the mean and standard deviation over three trials.

4 EXPERIMENTS

We test constrained label learning on a variety of tasks on text and image classification. First, we measure the test accuracy of CLL on a synthetic dataset and compare its performance to that of supervised learning and other baselines. Second, we validate our approach on real datasets.

For all our experiments, we compare CLL to other weakly supervised methods: data programming (DP) Ratner et al. [2016] and majority-vote (MV) or averaging (AVG). Additionally, on our real datasets we show comparison to regularized minimax conditional entropy for crowdsourcing (MMCE) Zhou et al. [2015]. For reference, we include the performance of supervised learning baseline. On the image datasets, we show comparison of CLL to Stoch-GALL, a multiclass extension of adversarial label learning. It is worth noting that DP was developed for binary classification, thus to compare its performance on our multiclass datasets, we run DP on the weak signals that label each class in the datasets. All the weak signals on the real datasets are one-vs-all signals meaning they only label a single class and abstain on other classes.

Method	Test Accuracy
CLL (Agr. rate ϵ)	**0.984**± 0.003
CLL (Constant ϵ)	0.978± 0.004
Data Programming	0.978± 0.003
Majority Vote	0.925± 0.009
CLL (True ϵ)	0.985± 0.0004
Supervised Learning	0.997± 0.001

Table 2: Classification accuracies of the different methods on synthetic data using independent weak signals. We report the mean and standard deviation over three trials

4.1 SYNTHETIC EXPERIMENT

We construct a toy dataset for a binary classification task where the data has 200 randomly generated binary features and 20,000 examples, 16,000 for training and 4,000 for testing. Each feature vector has between 50% to 70% correlation with the true label. We define two scenarios for our synthetic experiments. We run the methods using (1) dependent weak signals and, (2) independent weak signals. In both experiments, we use 10 weak signals that have at most 30% coverage on the data and conflicts on their label assignments. The dependent weak signals were constructed by generating one weak signal that is copied noisily 9 times (randomly flipping 20% of the labels). The original weak signal labeled 30% of the data points and had an accuracy in $[0.5, 0.6]$. So, on average, we expect to perturb 6% of its labels on the copies. The independent weak signals are randomly generated to have accuracies in the range $[0.6, 0.7]$.

We report in Table 1 and Table 2 the label and test accuracy from running CLL using true error rates for the weak signals, error rates estimated via agreement rate described in Section 3.3.1, and error rates using a maximum error rate constant set to 0.4 as the expected error for all the weak signals. CLL trained using the true ϵ obtains the highest test accuracy compared to the other baselines, and its performance almost matches that of supervised learning in Table 2. With the true bounds, CLL slightly outperforms CLL trained using estimated and constant ϵ. More interestingly, the results in Table 1 show that our method outperforms other baselines that are strongly affected by the dependence in the weak signals. The generative model of data programming assumes that the weak signals are independent given the true labels, but this is not the case in this setup as the weak signals are strongly dependent. Thus the conditional independence violation hurts its performance and essentially reduces it to performing a majority vote on the labels.

Since our evaluation in Fig. 3 demonstrated that CLL is not very sensitive to the choice of error rate, we set the error rates $\epsilon = 0.01$ on the text datasets and $\epsilon = \frac{1}{K}$ on the image datasets. We choose these values because our weak signals in the text dataset tend to label few examples and

Datasets	CLL	MMCE	DP	MV
IMDB	**0.736**± 0.0005	0.573	0.693	0.702
SST-2	**0.678**± 0.0004	0.677	0.666	0.666
YELP-2	0.765± 0.0002	0.685	0.770	**0.775**
TREC-6	0.842± 0.004	0.833	**0.898**	0.273

Table 3: Label accuracies of CLL compared to other weak supervision methods on different text classification datasets. We report the mean and standard deviation over three trials. CLL is trained using $\epsilon = 0.01$ on the text classification datasets.

Datasets	CLL	MMCE	DP	MV	Supervised
IMDB	**0.740**± 0.005	0.551	0.623± 0.007	0.724±0.004	0.820±0.003
SST-2	**0.729**± 0.001	0.727	0.720± 0.001	0.720± 0.0009	0.792± 0.001
YELP-2	**0.840**± 0.0007	0.68	0.760± 0.005	0.798± 0.007	0.879± 0.001
TREC-6	**0.641**± 0.022	0.64	0.627± 0.014	0.605± 0.006	0.700± 0.024

Table 4: Test accuracies of CLL compared to other weak supervision methods on different text classification datasets. We report the mean and standard deviation over three trials. CLL is trained using $\epsilon = 0.01$ on the text classification datasets

have low error rates thus we prefer not to under-constrain the optimization by using high error rates values for the one-vs-all weak-signals. In contrast, our human labeled weak signals on the image datasets have high error rates hence we set the error rate value to the baseline value for one-vs-all signals.

4.2 REAL EXPERIMENTS

The data sets for our real experiments and their weak signal generation process are described below. Table 7 summarizes the key statistics about these datasets. Our code and datasets are provided here.[2]

IMDB The IMDB dataset Maas et al. [2011] is used for sentiment analysis. The data contains reviews of different movies, and the task is to classify user reviews as either positive or negative in sentiment. We provide weak supervision by measuring mentions of specific words in the movie reviews. We created a set of positive words that weakly indicate positive sentiment and negative words that weakly indicate negative sentiment. We chose these keywords by looking at samples of the reviews and selecting popular words used in them. Many reviews could contain both positive and negative keywords, and in these cases, the weak signals will conflict on their labels. We split the dataset into training and testing subsets, where any example that contains one of our keywords is placed in the training set. Thus, *the test set consists of reviews that are not labeled by any weak signal*, making it important for the weakly supervised learning to generalize beyond the weak signals. The dataset contains 50,000 reviews, of which 29,182 are used for training and 20,392 are test examples.

SST-2 The Stanford Sentiment Treebank (SST-2) is another sentiment analysis dataset Socher et al. [2013] containing movie reviews. Like the IMDB dataset, the goal is to classify reviews from users as having either positive or negative sentiment. We use similar keyword-based weak supervision but with different keywords. We use the standard train-test split provided by the original dataset. While the original training data contained 6,920 reviews, our weak signals only cover 3,998 examples. Thus, we used the reduced data size to train our model. We use the full test set of 1,821 reviews.

YELP-2 We used the Yelp review dataset containing user reviews of businesses from the Yelp Dataset Challenge in 2015. Like the IMDB and SST-2 dataset, the goal is to classify reviews from users as having either positive or negative sentiment. We converted the star ratings in the dataset by considering reviews above 3 stars rating as positive and negative otherwise. We used similar weak supervision generating process as in SST-2. We sampled 50,000 reviews for training and 10,000 for testing from the original data set. Our weak signals only cover 45,370 data points, thus, we used the reduced data size to train our model.

TREC-6 TREC is a question classification dataset consisting of fact-based questions divided into different categories Li and Roth [2002]. The task is to classify questions to predict what category the question belongs to. We use the six-class version (TREC-6) from which we use 4,988 examples for training and 500 for testing. The weak supervision we use combines word mentions with other heuristics we defined to analyze patterns of the question and assign a class label based on certain patterns.

SVHN The Street View House Numbers (SVHN) Netzer et al. [2018] dataset represents the task of recognizing digits

[2] https://github.com/VTCSML/Constrained-Labeling-for-Weakly-Supervised-Learning

Datasets	CLL	MMCE	DP	AVG	Stoch-GALL
SVHN	**0.575**± 0.001	0.1	0.42	0.444	0.196± 0.025
Fashion-MNIST	**0.658**± 0.001	0.147	0.65	0.649	0.488± 0.002

Table 5: Label accuracies of CLL compared to other weak supervision methods on image datasets. We report the mean and standard deviation over three trials. CLL is trained using $\epsilon = \frac{1}{K}$ on the datasets and it outperforms other baseline approaches.

Datasets	CLL	MMCE	DP	AVG	Stoch-GALL	Supervised
SVHN	**0.670**± 0.031	0.1	0.265± 0.004	0.432± 0.001	0.366± 0.003	0.851± 0.002
Fashion-MNIST	**0.695**± 0.002	0.151	0.635± 0.0004	0.666± 0.002	0.598± 0.002	0.852± 0.003

Table 6: Test accuracies of CLL compared to other weak supervision methods on image datasets. We report the mean and standard deviation over three trials. CLL is trained using $\epsilon = \frac{1}{K}$ on the datasets.

Dataset	No. classes	No. weak signals	Train Size	Test Size
IMDB	2	10	29,182	20,392
SST-2	2	14	3,998	1,821
YELP-2	2	14	45,370	10,000
TREC-6	6	18	4,988	500
SVHN	10	50	73,257	26,032
Fashion-MNIST	10	50	60,000	10,000

Table 7: Summary of datasets, including the number of weak signals used for training.

on real images of house numbers taken by Google Street View. Each image is a 32 × 32 RGB vector. The dataset has 10 classes and has 73,257 training images and 26,032 test images. We define 50 weak signals for this dataset. For this image classification dataset, we augment 40 other human-annotated weak signals (four per class) with ten pseudolabel predictions of each class from a model trained on 1% of the training data. The human-annotated weak signals are nearest-neighbor classifiers where a human annotator is asked to mark distinguishing features about an exemplar image belonging to a specific class. We then calculate pairwise Euclidean distances between the pixels in the marked region across images. We convert the Euclidean scores to probabilities (soft labels for the examples) via a logistic transform. Through this process, an annotator is guiding the design of a simple one-versus-rest classifier, where images most similar to the reference image are more likely to belong to its class.

Fashion-MNIST The Fashion-MNIST dataset Xiao et al. [2017] represents the task of recognizing articles of clothing where each example is a 28 × 28 grayscale image. The images are categorized into 10 classes of clothing types where each class contains 6,000 training examples and 1,000 test examples. We used the same format of weak supervision signals as in the SVHN dataset (pseudolabels and human-annotated nearest-neighbor classifiers).

Models For the text analysis tasks, we use 300-dimensional GloVe vectors Pennington et al. [2014] as features for the text classification tasks. Then we train a simple two-layer neural network with 512 hidden units and ReLU activation in its hidden layer. The model for the image classification tasks is a six-layer convolutional neural network model with a 3×3 filter and 32 channels at each layer. We use a sigmoid function as the output layer for both models in our experiment. Thus we train using binary cross-entropy loss with the soft labels returned by CLL, which represent the probability of examples belonging to classes.

Results Tables 3 and 4 list the performance of the various weakly supervised methods on text classification datasets, while Tables 5 and 6 list the performance of various weakly supervised methods on image classification datasets. Considering both types of accuracy, CLL is able to output labels for the training data that train high-quality models for the test set. CLL outperforms all competing methods on test accuracy on the datasets. Interestingly, on Yelp and Trec-6 datasets, CLL label accuracy is lower than that of competing baselines yet CLL still achieves superior test accuracy. We surmise that CLL label accuracy is lower than competing methods on some datasets because of the inaccuracy in the error estimates. Generally, CLL is able to learn robust labels from the weak signals, and it seems to pass this information to the learning algorithm to help it generalize on unseen examples. For example, on the IMDB dataset, we used keyword-based weak signals that only occur on the training data. The model trained using CLL labels performs better on the test set than models trained with labels learned from data programming or majority vote. CLL outperforms all competing methods on the image classification tasks. On the digit recognition task (SVHN), CLL outperforms the best compared method (average) by over 13 percentage points for the label accuracy and 23 percentage points on the test data. CLL is able to better synthesize information from the low-quality human-annotated signals combined with the higher-quality pseudolabel signals.

5 CONCLUSION

We introduced constrained label learning (CLL), a weakly supervised learning method that combines different weak supervision signals to produce probabilistic training labels for the data. CLL defines a constrained space for the labels of the training data by requiring that the errors of the weak signals agree with the provided error estimates. CLL is fast and converges after a few iterations of gradient descent. Our theoretical analysis shows that the accuracy of our estimated labels increases as we add more linearly independent weak signals. This analysis is consistent with the intuition that the constrained-space interpretation of weak supervision avoids overcounting evidence when multiple redundant weak signals provide the same information, since they are linearly dependent. Our experiments compare CLL against other weak supervision approaches on different text and image classification tasks. The results demonstrate that CLL outperforms these methods on most tasks. Interestingly, we are able to perform well when we train CLL using a worst case uniform error estimate for the weak signals. This shows that CLL is robust and not too sensitive to inaccuracy in the error estimates. In future work, we aim to theoretically analyze the behavior of this approach in such settings where the error rates are unreliable, with the hope that theoretical understanding will suggest new approaches that are even more robust.

Acknowledgements

We like to thank Sai Sindhura Tipirneni for her help in setting up the MMCE code. Arachie and Huang were both supported by a grant from the U.S. Department of Transportation, University Transportation Centers Program to the Safety through Disruption University Transportation Center (69A3551747115).

References

Chidubem Arachie and Bert Huang. Stochastic generalized adversarial label learning. *arXiv preprint arXiv:1906.00512*, 2019a.

Chidubem Arachie and Bert Huang. Adversarial label learning. In *Proc. of the AAAI Conf. on Artif. Intelligence*, pages 3183–3190, 2019b.

Stephen H. Bach, Daniel Rodriguez, Yintao Liu, Chong Luo, Haidong Shao, Cassandra Xia, Souvik Sen, Alex Ratner, Braden Hancock, and Houman Alborzi. Snorkel DryBell: A case study in deploying weak supervision at industrial scale. In *Intl. Conf. on Manag. of Data*, pages 362–375, 2019.

Akshay Balsubramani and Yoav Freund. Scalable semi-supervised aggregation of classifiers. In *Advances in Neural Information Processing Systems*, pages 1351–1359, 2015.

Razvan C. Bunescu and Raymond Mooney. Learning to extract relations from the web using minimal supervision. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, volume 45, pages 576–583, 2007.

Bob Carpenter. Multilevel Bayesian models of categorical data annotation. *Unpublished manuscript*, 17(122):45–50, 2008.

Liang-Chieh Chen, Sanja Fidler, Alan L. Yuille, and Raquel Urtasun. Beat the mturkers: Automatic image labeling from weak 3d supervision. In *Proc. of the IEEE Conf. on Comp. Vis. and Pattern Recognition*, pages 3198–3205, 2014.

Alexander Philip Dawid and Allan M. Skene. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Applied Statistics*, pages 20–28, 1979.

Gregory Druck, Gideon Mann, and Andrew McCallum. Learning from labeled features using generalized expectation criteria. In *Proceedings of the 31st Annual Intl. ACM SIGIR Conf. on Research and Dev. in Information Retrieval*, pages 595–602, 2008.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12 (Jul):2121–2159, 2011.

Huiji Gao, Geoffrey Barbier, and Rebecca Goolsby. Harnessing the crowdsourcing power of social media for disaster relief. *IEEE Intelligent Systems*, 26(3):10–14, 2011.

Yoni Halpern, Steven Horng, and David Sontag. Clinical tagging with joint probabilistic models. In *Proceedings of the Conference on Machine Learning for Healthcare*, pages 209–225, 2016.

Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke Zettlemoyer, and Daniel S. Weld. Knowledge-based weak supervision for information extraction of overlapping relations. In *Proc. of the Annual Meeting of the Assoc. for Comp. Linguistics: Human Language Tech.*, pages 541–550, 2011.

Ariel Jaffe, Ethan Fetaya, Boaz Nadler, Tingting Jiang, and Yuval Kluger. Unsupervised ensemble learning with dependent classifiers. In *Artificial Intelligence and Statistics*, pages 351–360, 2016.

David R. Karger, Sewoong Oh, and Devavrat Shah. Iterative learning for reliable crowdsourcing systems. In *Advances in Neural Information Processing Systems*, pages 1953–1961, 2011.

Ashish Khetan, Zachary C. Lipton, and Anima Anandkumar. Learning from noisy singly-labeled data. *arXiv preprint arXiv:1712.04577*, 2017.

Xin Li and Dan Roth. Learning question classifiers. In *Proceedings of the 19th International Conference on Computational Linguistics-Volume 1*, pages 1–7. Association for Computational Linguistics, 2002.

Qiang Liu, Jian Peng, and Alexander T. Ihler. Variational inference for crowdsourcing. In *Advances in Neural Information Processing Systems*, pages 692–700, 2012.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 142–150. Association for Computational Linguistics, 2011.

Omid Madani, David M. Pennock, and Gary W. Flake. Co-validation: Using model disagreement on unlabeled data to validate classification algorithms. In *Advances in Neural Information Processing Systems*, pages 873–880, 2005.

Gideon S. Mann and Andrew McCallum. Generalized expectation criteria for semi-supervised learning of conditional random fields. *Proc. of the Annual Meeting of the Assoc. for Comp. Linguistics: Human Language Tech.*, pages 870–878, 2008.

Gideon S. Mann and Andrew McCallum. Generalized expectation criteria for semi-supervised learning with weakly labeled data. *Journal of Machine Learning Research*, 11: 955–984, 2010.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. Distant supervision for relation extraction without labeled data. In *Proc. of the Annual Meeting of the Assoc. for Comp. Ling.*, 2009.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and A Ng. The street view house numbers (SVHN) dataset. Technical report, Accessed 2016-08-01.[Online]., 2018.

Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, 2014.

Emmanouil Antonios Platanios, Avrim Blum, and Tom Mitchell. Estimating accuracy from unlabeled data. In *Proceedings of the Thirtieth Conf. on Uncertainty in Artificial Intelligence*, pages 682–691, 2014.

Emmanouil Antonios Platanios, Avinava Dubey, and Tom Mitchell. Estimating accuracy from unlabeled data: A Bayesian approach. In *International Conference on Machine Learning*, pages 1416–1425, 2016.

Emmanouil Antonios Platanios, Maruan Al-Shedivat, Eric Xing, and Tom Mitchell. Learning from imperfect annotations. *arXiv preprint arXiv:2004.03473*, 2020.

Alex Ratner, Braden Hancock, Jared Dunnmon, Roger Goldman, and Christopher Ré. Snorkel metal: Weak supervision for multi-task learning. In *Proceedings of the Second Workshop on Data Management for End-To-End Machine Learning*, pages 1–4, 2018.

Alexander J. Ratner, Christopher M. De Sa, Sen Wu, Daniel Selsam, and Christopher Ré. Data programming: Creating large training sets, quickly. In *Advances in Neural Info. Proc. Sys.*, pages 3567–3575, 2016.

Alexander J. Ratner, Stephen H. Bach, Henry R. Ehrenberg, and Chris Ré. Snorkel: Fast training set generation for information extraction. In *Proceedings of the 2017 ACM Intl. Conf. on Management of Data*, pages 1683–1686. ACM, 2017.

Sebastian Riedel, Limin Yao, and Andrew McCallum. Modeling relations and their mentions without labeled text. In *Joint Euro. Conf. on Mach. Learn. and Knowledge Disc. in Databases*, 2010.

Robert E. Schapire, Marie Rochery, Mazin Rahim, and Narendra Gupta. Incorporating prior knowledge into boosting. In *Intl. Conf. on Machine Learning*, volume 2, pages 538–545, 2002.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, 2013.

Jacob Steinhardt and Percy S Liang. Unsupervised risk estimation using only conditional independence structure. In *Adv. in Neural Information Processing Systems*, pages 3657–3665, 2016.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Jia Xu, Alexander G. Schwing, and Raquel Urtasun. Tell me what you see and I will show you where it is. In *Proc. of the IEEE Conf. on Computer Vis. and Pattern Recog.*, pages 3190–3197, 2014.

Limin Yao, Sebastian Riedel, and Andrew McCallum. Collective cross-document relation extraction without labelled data. In *Proc. of the Conf. on Empirical Methods in Natural Language Processing*, pages 1013–1023, 2010.

Dengyong Zhou, Qiang Liu, John C. Platt, Christopher Meek, and Nihar B. Shah. Regularized minimax conditional entropy for crowdsourcing. *arXiv preprint arXiv:1503.07240*, 2015.

Yao Zhou and Jingrui He. Crowdsourcing via tensor augmentation and completion. In *International Joint Conference on Artificial Intelligence*, pages 2435–2441, 2016.