OptSLA: an Optimization-Based Approach for Sequential Label Aggregation

Nasim Sabetpour, Adithya Kulkarni, and **Qi Li**
Department of Computer Science, Iowa State University
{nasim, aditkulk, qli}@iastate.edu

Abstract

The need for the annotated training dataset on which data-hungry machine learning algorithms feed has increased dramatically with advanced acclaim of machine learning applications. To annotate the data, people with domain expertise are needed, but they are seldom available and expensive to hire. This has lead to the thriving of crowdsourcing platforms such as Amazon Mechanical Turk (AMT). However, the annotations provided by one worker cannot be used directly to train the model due to the lack of expertise. Existing literature in annotation aggregation focuses on binary and multi-choice problems. In contrast, little work has been done on complex tasks such as sequence labeling with imbalanced classes, a ubiquitous task in Natural Language Processing (NLP), and Bio-Informatics. We propose OPTSLA, an Optimization-based Sequential Label Aggregation method, that jointly considers the characteristics of sequential labeling tasks, workers reliabilities, and advanced deep learning techniques to conquer the challenge. We evaluate our model on crowdsourced data for named entity recognition task. Our results show that the proposed OPTSLA outperforms the state-of-the-art aggregation methods, and the results are easier to interpret.

1 Introduction

Crowdsourcing (Howe, 2008) is a popular platform to annotate massive corpora inexpensively. It has bred lots of interest in machine learning and deep learning tasks. However, when workers provide annotations, the results may be noisier comparing with labels provided by experts. Thus, it becomes essential to conduct truth inference from the noisy annotations.

One common annotation aggregation approach is Majority Voting (MV) (Lam and Suen, 1997), in which annotation with the highest number of occurrences is deemed as truth. Another naive approach is to regard an annotation as correct if a certain number of workers provide the same annotation. The concern with these methods is that they assume all workers are of the same quality, which is usually invalid in practice. In this paper, we study the annotation aggregation problem for sequential labeling tasks, a common NLP task.

Many existing crowdsourcing label aggregation methods may suffer from performance loss because they assume that data instances are independent (Zheng et al., 2017). New approaches are recently proposed to handle the particular characteristics of sequential labeling tasks, where tokens in one sentence have complex dependencies (Rodrigues et al., 2014; Simpson and Gurevych, 2019; Nguyen et al., 2017). In this line of approaches, probabilistic models are adopted to model the workers' labeling behavior and to model the dependencies between adjacent tokens. There are some drawbacks to the probabilistic models. First, they have strong statistical assumptions when modeling the sequence annotations, limiting the flexibility of the models. Second, these models need to infer complex parameters, making it hard to interpret the relations between worker's quality and token's true labels. Third, these aggregation methods can not fully unleash the power of deep learning in sequential labeling tasks.

To address these challenges, we propose an optimization framework to improve aggregation performance. Our method OPTSLA estimates workers' reliability and models the label dependencies to infer the true labels from noisy annotations. OPTSLA handles complex sequential label aggregation problem with fewer parameters comparing the state-of-the-art and produces easy-to-understand results.

We further incorporate the state-of-the-art deep

Findings of the Association for Computational Linguistics: EMNLP 2020, pages 1335–1340
November 16 - 20, 2020. ©2020 Association for Computational Linguistics

learning approach into OptSLA, where the deep learning component and the aggregation component can maturely enhance each other. To ensure high-quality training data, OptSLA chooses sentences with high confidence from the aggregation component. The deep learning model is incrementally trained with the iteratively updated aggregation results.

2 Related Works

Data aggregation and label inference tasks have received lots of attention over the past decade, and many methods are developed to handle various challenges (Li et al., 2016; Zheng et al., 2017). Earlier works such as (Dawid and Skene, 1979; Yin et al., 2008; Snow et al., 2008; Whitehill et al., 2009; Groot et al., 2011) proposed to model the worker qualities and label inference using statistical methods. Later, optimization-based methods are proposed (Zhou et al., 2012; Li et al., 2014). Intensive experiments in many applications and tasks have shown that these methods generally outperform MV, which indicates that the worker qualities estimation can play an essential role in label inference. However, in these methods, the annotation instances are assumed to be independent.

More recently, methods are developed to handle various types of correlations among annotation instances. For example, methods in (Meng et al., 2016; Yao et al., 2018; Zhi et al., 2018) are proposed to handle the spatial-temporal dependencies among instances, and methods in (Rodrigues et al., 2014; Nguyen et al., 2017; Simpson and Gurevych, 2019) are proposed to handle the sequential labeling tasks in NLP, which are more related to this paper. Rodrigues et al. (Rodrigues et al., 2014) proposed a probabilistic approach using Conditional Random Fields (CRF) to model the sequential annotations. In this model, the worker's reliability is modeled by his/her F1 score, but only one worker is assumed to be correct for any instance. Nguyen et al. relaxed the assumption and proposed a hidden Markov model (HMM) extension in (Nguyen et al., 2017). This model uses J parameters per worker to model their reliabilities, where J is the number of classes. Recently, Simpson et al. (Simpson and Gurevych, 2019) proposed a fully-Bayesian approach, where $J \times J \times J$ parameters are used to model workers' reliabilities.

The three models mentioned above are probabilistic models with significantly more parameters to tune and are harder to interpret than optimization-based methods (Zheng et al., 2017). Moreover, the existing methods do not fully unleash the power of deep learning approaches in sequential labeling tasks. In this paper, we propose an optimization-based aggregation method to address the interpretability challenge, and further include the deep learning module to boost the performance.

3 Methodology

The sequential label aggregation task aims to combine the annotations provided by different workers to infer the ground truth sequential labels. In this section, we describe our approach, an optimization-based sequential label aggregation method (OptSLA), which aggregates multiple workers' annotations with deep learning results by estimating the reliability of workers and modeling the dependencies among tokens in the sentences.

3.1 OptSLA

We first introduce the notations. Suppose m workers (indexed by j) are hired to annotate s sentences (indexed by k) with total n tokens in the corpus. Let i_k indicate the i-th token in the k-th sentence. $y^j_{i_k}$ is a one-hot vector that denotes the annotation given by the j-th worker on the i-th token in the k-th sentence. $y^*_{i_k}$ is the inferred aggregation label for the corresponding token. Each worker has a weight parameter w_j to reflect his/her annotation quality, and $\mathcal{W} = \{w_1, w_2, ..., w_m\}$ refers to the set of all worker weights. A higher weight implies that the worker is of higher reliability.

Our goal is to minimize the overall weighted loss of the inferred aggregation labels $y^*_{i_k}$ to the reliable workers' annotations $y^j_{i_k}$, deep learning predictions $\hat{y}^*_{i_k}$, and the loss of inconsistencies in sequential labels. Mathematically, we formulate the aggregation problem as an optimization problem with respect to set of worker weights $\mathcal{W}$, the weight of deep learning model w_{dl}, aggregated annotation $y^*_{i_k}$, and the deep learning parameters $\boldsymbol{\theta}$ shown in Eq (1).

$$\begin{aligned}
&\min f(\mathcal{W}, w_{dl}, \{y^*_{i_k}\}^n_{i_k=1}, \boldsymbol{\theta}) = \\
&\sum_j w_j \sum_k \xi(y^*_k) \sum_{i_k} H(y^j_{i_k}, y^*_{i_k}) \\
&+ w_{dl} \sum_k \xi(y^*_k) \sum_{i_k} H(y^*_{i_k}, \hat{y}^*_{i_k}) \\
&- \sum_j |\{y^j_{i_k}\}_{i_k}| \log(w_j) + n \log(w_{dl}) \\
&+ \sum_{i_k} (g(y^*_{i_k-1}, y^*_{i_k}) + g(y^*_{i_k}, y^*_{i_k+1})), \qquad (1)
\end{aligned}$$

where $H(\cdot,\cdot)$ is the cross entropy loss function, $\xi(y_k^*)$ is the confidence level of the k-th sentence, $|\{y_{i_k}^j\}_{i_k}|$ refers to the number of annotations provided by worker j, and $g(.,.)$ is a loss function to maintain the consistency between tokens label. More specifically, $\xi(y_k^*) = \frac{1}{l_k}\sum_{i_k} margin(y_{i_k}^*)$, where l_k is the number of tokens in sentence k and $margin(y_{i_k}^*)$ is the probability difference between the two most likely labels of $y_{i_k}^*$.

In Eq(1), $\sum_j w_j \sum_k \xi(y_k^*) \sum_{i_k} H(y_{i_k}^j, y_{i_k}^*)$ is the weighted cross-entropy loss between the inferred aggregation labels and the workers' annotations. The loss is adjusted by confidence measurement of (y_k^*). Intuitively, if a worker is more reliable (i.e., w_j is high) and the annotations are agreed with high confidence, high penalty will be received if his/her annotations are quite different from the inferred aggregation labels. In order to minimize the objective function, the inferred aggregation labels $y_{i_k}^*$ will rely more on the workers with high weights.

The term $w_{dl} \sum_k \xi(y_k^*) \sum_{i_k} H(y_{i_k}^*, \hat{y}_{i_k}^*)$ is the weighted cross-entropy loss between $y_{i_k}^*$ and the predicted labels $\hat{y}_{i_k}^*$ from a trained deep learning model, where w_{dl} is the reliability of the deep learning model. In our model, the deep learning model is essentially treated as an additional worker. The training of the deep learning model is discussed in Section 3.4.

The term $\sum_j |\{y_{i_k}^j\}_{i_k}| \log(w_j) + n \log(w_{dl})$ is a constraint to ensure that the calculated weights are positive. The final term $\sum_i g(y_{i-1}^*, y_i^*, y_{i+1}^*)$ is a loss function which gives penalties if the inferred aggregation labels is not consistent with the sequential label rules. One simple example of $g(\cdot,\cdot)$ is

$$g(y_{i_k-1}^*, y_{i_k}^*) = \begin{cases} 0, & \text{if } P(y_{i_k}|y_{i_k-1}) > 0. \\ 1, & Otherwise. \end{cases} . \tag{2}$$

This function will give 0 loss if the sequence of $y_{i_k-1}^*, y_{i_k}^*$ is valid according to sequential label rules, and 1 if the sequence is invalid. Taking NER task as an example, $P(y_{i_k} = \text{'I-LOC'}|y_{i_k-1} = \text{'B-PER'}) = 0$, so $g(y_{i_k-1}^* = \text{'I-LOC'}, y_{i_k}^* = \text{'B-PER'}) = 1$. Therefore in $g(y_{i_k-1}^*, y_{i_k}^*) + g(y_{i_k}^*, y_{i_k+1}^*)$, both $y_{i_k-1}^*$ and $y_{i_k+1}^*$ are considered.

The inferred aggregation labels $y_{i_k}^*$, workers weights $\mathcal{W}$ and w_{dl}, and the deep learning model are learned simultaneously by optimizing the Eq (1). To solve the problem, we adopt the block coordinate descent method (Tseng, 2001), which will keep reducing the value of the objective function. To minimize the objective function in Eq (1), we iteratively conduct the following three steps.

3.2 Workers' Weight Update

We initialize all the workers with equal weights. To update weights in each iteration, we treat the other variables as fixed. Then

$$\mathcal{W} \leftarrow \operatorname*{argmin}_{\mathcal{W}} f(y_{i_k}^*, \mathcal{W}, \boldsymbol{\theta}). \tag{3}$$

$\mathcal{W}$ has closed form solution by taking differentiation of Eq (1) with respects to $\mathcal{W}$. The solution is shown as follows

$$w_j = \frac{|\{y_{i_k}^j\}_{i_k}|}{\sum_k \xi(y_k^*) \sum_{i_k} H(y_{i_k}^j, y_{i_k}^*)}. \tag{4}$$

w_{dl} is updated similarly.

3.3 Aggregated Annotation Update

In the second step, once the workers' weights are updated, the inferred aggregation labels $y_{i_k}^*$ are updated to minimize Eq (1) as follows.

$$\begin{aligned} \operatorname*{argmin}_{y_{i_k}^*}(&\sum_j w_j \sum_k \xi(y_k^*) \sum_{i_k} H(y_{i_k}^j, y_{i_k}^*) \\ &+ w_{dl} \sum_k \xi(y_k^*) \sum_{i_k} H(y_{i_k}^*, \hat{y}_{i_k}^*)) \\ &+ \sum_{i_k} (g(y_{i_k-1}^*, y_{i_k}^*) + g(y_{i_k}^*, y_{i_k+1}^*)). \end{aligned} \tag{5}$$

This function does not have a closed-form solution. In fact, for general label consistency loss function $g(\cdot,\cdot)$, it might be non-trivial to solve Eq (5) as variables are correlated. Therefore, we apply the gradient descent method to calculate $y_{i_k}^*$ while fixing all other variables.

3.4 Incremental Deep Learning

With updated aggregation results, we update the deep learning model. To maintain a high quality model, we select sentences with high $\xi(\{y_k^*\})$ (i.e., $\xi(\{y_{i_k}^*\})$ >0.9) as training data. Since $y_{i_k}^*$ is updated iteratively, the training data change as well. However, the re-train of the deep learning model can be time_consuming. Therefore, we adopt the incremental deep learning approach (Sarwar et al., 2019) to improve algorithm efficiency.

3.5 Class Priority (ρ)

Many sequential labeling tasks have class imbalance problem. For example, in the NER task, "O" will dominate the entity annotations. To handle this problem, class priorities (ρ's) can be used to re-weight the classes. A higher ρ will increase the weight for entity labels when calculating $y^*_{i_k}$.

4 Experiments.

Datasets. We use real-world data to demonstrate the effectiveness of the proposed method OPTSLA. **NER dataset** (Sang and De Meulder, 2003) [1] consists of 5985 sentences and 47 workers are hired to identify the named entities in the sentences and annotate them as persons, locations, organizations, or miscellaneous. To make the task more challenging, we use 4515 sentences where workers had conflicting annotations, and for comparison we choose 3466 sentences to evaluate, which is the same as test set for NER dataset. [2]

To evaluate the proposed OPTSLA, we compare the span level precision, recall, and F1 score[3] of the inferred aggregation labels with three state-of-the-art baselines methods HMM-crowd (Nguyen et al., 2017), CRF-MA (Rodrigues et al., 2014), and BSC-seq result comes from (Simpson and Gurevych, 2019). For OPTSLA, Convolutional Neural Network (CNN) is employed as the deep learning component for the NER dataset. To evaluate the effect of the deep learning module, we also compare OPTSLA without the deep learning component, denoted as OPTSLA (W/O DL).

The results are shown in Table 1[4]. It is clear that the proposed OPTSLA method outperforms state-of-the-art baselines methods. The results show that the deep learning component can indeed enhance aggregation performance. $H(\cdot, \cdot)$ and $\xi(\{y^*_{i_k}\}_i)$ help in predicting worker reliability properly which in turn help in aggregation. This is because that OPTSLA only uses sentences with high $\xi(\{y^*_{i_k}\})$ for training, the deep learning model is trained properly.

Table 1: Performance Comparison

	Prec.	Rec.	F1
MV	79.9	55.3	65.4
CRF-MA	80.29	51.20	62.53
HMM-crowd	77.40	72.29	74.76
BSC-seq	**80.3**	74.8	77.4
OPTSLA (W/O DL)	76.61	74.14	75.36
OPTSLA	79.42	**77.59**	**78.49**

As the worker's reliability estimation is the key to obtain high-quality aggregation results, we further show the weights estimated for workers with respect to their actual F1 scores in Figure 1. It can be observed that there is a strong positive correlation between worker weights and their actual F1 scores. Because OPTSLA uses one parameter for each worker, the results are more straightforward to interpret and justify comparing with the baseline methods.

We observe that OPTSLA converges quickly. The algorithm stops when no more sentences can be added to the training set. Figure 2 illustrates the size of the training dataset with respect to the number of iterations.

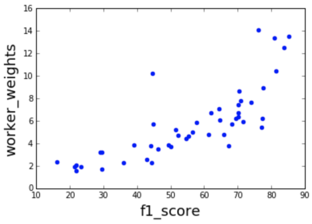

Figure 1: Worker weights w.r.t. their F1 scores

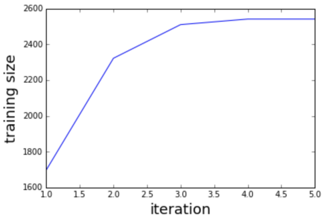

Figure 2: Training size w.r.t. iterations

5 Conclusion and Future Works

In this paper, we propose an innovative optimization-based approach OPTSLA for sequential label aggregation problem. Our model jointly considers different factors in the objective function, including the workers' annotations, workers' reliability, the deep learning model, and the characteristics of sequential labeling tasks. Our experimental results illustrate that OPTSLA outperforms the state-of-the-art sequential label aggregations methods, such as CRF-MA, HMM-Crowd, and Bayesian Sequence Combination

[1] Dataset can be found on http://amilab.dei.uc.pt/fmpr/crf-ma-datasets.tar.gz

[2] All codes, experiment scripts, datasets, and results are in a public repository https://github.com/NasimISU/OptSLA

[3] https://github.com/allenai/allennlp/tree/master/allennlp

[4] The results for CRF-MA and HMM-crowd come from (Nguyen et al., 2017), and BSC-seq results come from (Simpson and Gurevych, 2019)

(BSC) in terms of F1 score. For the future work, we will evaluate more factors such as the task assignment that may affect the aggregation performance from the deep learning model and the workers' behaviors.

Acknowledgements

The work was supported in part by the National Science Foundation under Grant NSF IIS-2007941. Any opinions, findings, and conclusions or recommendations expressed in this document are those of the author(s) and should not be interpreted as the views of any U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

References

A. Philip Dawid and Allan Skene. 1979. Maximum likelihood estimation of observer error-rates using the em algorithm.

Perry Groot, Adriana Birlutiu, and Tom Heskes. 2011. Learning from multiple annotators with gaussian processes. In *International Conference on Artificial Neural Networks*, pages 159–164. Springer.

Jeff Howe. 2008. *Crowdsourcing: How the power of the crowd is driving the future of business*. Random House.

Louisa Lam and SY Suen. 1997. Application of majority voting to pattern recognition: an analysis of its behavior and performance. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 27(5):553–568.

Qi Li, Yaliang Li, Jing Gao, Bo Zhao, Wei Fan, and Jiawei Han. 2014. Resolving conflicts in heterogeneous data by truth discovery and source reliability estimation. In *Proc. of the ACM SIGMOD International Conference on Management of Data (SIGMOD'14)*, pages 1187–1198.

Yaliang Li, Jing Gao, Chuishi Meng, Qi Li, Lu Su, Bo Zhao, Wei Fan, and Jiawei Han. 2016. A survey on truth discovery. *ACM Sigkdd Explorations Newsletter*, 17(2):1–16.

Chuishi Meng, Houping Xiao, Lu Su, and Yun Cheng. 2016. Tackling the redundancy and sparsity in crowd sensing applications. In *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM*, pages 150–163.

An T Nguyen, Byron C Wallace, Junyi Jessy Li, Ani Nenkova, and Matthew Lease. 2017. Aggregating and predicting sequence labels from crowd annotations. In *Proceedings of the conference. Association for Computational Linguistics. Meeting*, volume 2017, page 299. NIH Public Access.

Filipe Rodrigues, Francisco Pereira, and Bernardete Ribeiro. 2014. Sequence labeling with multiple annotators. *Machine learning*, 95(2):165–181.

Erik F Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. *arXiv preprint cs/0306050*.

Syed Shakib Sarwar, Aayush Ankit, and Kaushik Roy. 2019. Incremental learning in deep convolutional neural networks using partial network sharing. *IEEE Access*.

Edwin D. Simpson and Iryna Gurevych. 2019. A Bayesian approach for sequence tagging with crowds. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1093–1104, Hong Kong, China. Association for Computational Linguistics.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y Ng. 2008. Cheap and fast — but is it good? Evaluating non-expert annotations for natural language tasks. In *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP'08)*, pages 254–263.

Paul Tseng. 2001. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of optimization theory and applications*, 109(3):475–494.

Jacob Whitehill, Ting-fan Wu, Jacob Bergsma, Javier R Movellan, and Paul L Ruvolo. 2009. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *Advances in neural information processing systems*, pages 2035–2043.

Liuyi Yao, Lu Su, Qi Li, Yaliang Li, Fenglong Ma, Jing Gao, and Aidong Zhang. 2018. Online truth discovery on time series data. In *Proceedings of the 2018 SIAM International Conference on Data Mining*, pages 162–170. SIAM.

Xiaoxin Yin, Jiawei Han, and Philip S. Yu. 2008. Truth discovery with multiple conflicting information providers on the web. *IEEE Transactions on Knowledge and Data Engineering*, 20(6):796–808.

Yudian Zheng, Guoliang Li, Yuanbing Li, Caihua Shan, and Reynold Cheng. 2017. Truth inference in crowdsourcing: Is the problem solved? *Proceedings of the VLDB Endowment*, 10(5):541–552.

Shi Zhi, Fan Yang, Zheyi Zhu, Qi Li, Zhaoran Wang, and Jiawei Han. 2018. Dynamic truth discovery on numerical data. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 817–826. IEEE.

Dengyong Zhou, Sumit Basu, Yi Mao, and John C Platt. 2012. Learning from the wisdom of crowds by minimax entropy. In *Advances in Neural Information Processing Systems (NIPS'12)*, pages 2195–2203.