Journal of Data Intelligence, Vol. 3, No. 3 (2022) 301–315


# EFFECT OF LABEL REDUNDANCY IN CROWDSOURCING FOR TRAINING MACHINE LEARNING MODELS

AYAME SHIMIZU
*Graduate School of Comprehensive Human Sciences, University of Tsukuba*
*Kasuga 1-2, Tsukuba, Ibaraki 305-8550, Japan*
*shimizu.ayame.sw@alumni.tsukuba.ac.jp*

KEI WAKABAYASHI
*Faculty of Library, Information and Media Studies, University of Tsukuba*
*Kasuga 1-2, Tsukuba, Ibaraki 305-8550, Japan*
*kwakaba@slis.tsukuba.ac.jp*

Crowdsourcing is widely utilized for collecting labeled examples to train supervised machine learning models, but the labels obtained from workers are considerably noisier than those from expert annotators. To address the noisy label issue, most researchers adopt the repeated labeling strategy, where multiple (redundant) labels are collected for each example and then aggregated. Although this improves the annotation quality, it decreases the amount of training data when the budget for crowdsourcing is limited, which is a negative factor in terms of the accuracy of the machine learning model to be trained. This paper empirically examines the extent to which repeated labeling contributes to the accuracy of machine learning models for image classification, named entity recognition and sentiment analysis under various conditions of budget and worker quality. We experimentally examined four hypotheses related to the effect of budget, worker quality, task difficulty, and redundancy on crowdsourcing. The results on image classification and named entity recognition supported all four hypotheses and suggested that repeated labeling almost always has a negative impact on machine learning when it comes to accuracy. Somewhat surprisingly, the results on sentiment analysis using pretrained models did not support the hypothesis which shows the possibility of remaining utilization of multiple-labeling.

*Keywords*: Crowdsoucing, Multiple labeling, Supervised labeling

## 1. Introduction

Supervised machine learning requires the use of annotated examples, but obtaining labels annotated by experts can be both expensive and time-consuming. Thanks to crowdsourcing platforms such as Amazon's Mechanical Turk [a], crowdsourcing has become an easy and cost-effective way to collect annotations because crowdsourced labels are cheap and fast. Labels from crowd workers often contain mistakes, but supervised models perform well enough with these non-expert labels [1], and extensive research has been conducted on how to increase the efficiency of noisy labels. The most popular approach is to collect multiple labels on each task and then aggregate them.

In supervised machine learning, ground-truth labels are used as the training data, but even widely used datasets such as CIFAR10 [2] and MNIST [3] include some uncertain labels [4, 5].

[a] https://www.mturk.com

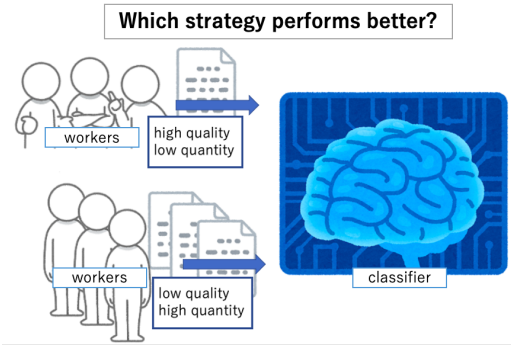


Fig. 1. The trade-off between label quality and quantity.

This means that the only difference between supervised machine learning using ground-truth labels and crowdsourced labels is the ratio of the uncertain labels.

It is also obvious that aggregating redundant labels decreases the size of the dataset. The amount of training data and the accuracy of labels both affect supervised machine learning. As shown in Fig. 1, the label quality and quantity trade-off seems to be debatable, and it's not clear if multiple labeling is always effective when designing machine learning.

In this study, we examine how label redundancy affects the supervised model. Working from the assumption that there are cases where taking a single label per task and increasing the amount of training data will result in a higher performance than redundant labeling, we came up with the following four hypotheses. We made 4 hypotheses.

1. When the budget for crowdsourcing is small, taking redundant labels decreases the performance of the classifier.

2. When the worker quality is high, taking redundant labels decreases the performance of the classifier.

3. When the task is easy for the classifier, taking redundant labels decreases the performance of the classifier.

4. The optimal amount of redundancy depends on the budget, the worker quality, and the task complexity.

The first hypothesis is that the benefit of redundancy increases as the budget becomes larger. As the amount of training data increases, the amount of improvement in the prediction accuracy of the classifier due to the increase in the amount of training data decreases. Therefore, we predict that the benefit of redundancy will increase as the budget increases.

This first hypothesis is motivated by the fact that the accuracy of machine learning models tends to saturate when the amount of training data is increased. Fig. 2 shows the experimental results of the change in prediction accuracy when the amount of training data given to the classifier was varied. The details of the data we used are provided in Table 1. In the figure, the vertical axis is the macro F1 value of the prediction results for the test data of the classifier, and the horizontal axis is the number of training data. The results for each dataset are shown.

Table 1. Datasets used in supervised learning experiment

| Name | Task | Number of labels | Pre-training |
|---|---|---|---|
| MNIST | Image classification | 10 | No |
| CIFAR10 | Image classification | 10 | No |
| CIFAR100 | Image classification | 100 | No |
| CoNLL-2003 | NER | 9 | No |
| BC5CDR | NER | 5 | No |
| IMDB | Sentiment analysis | 2 | Yes |

We found that in most cases, when the number of training data exceeded a certain level, the amount of increase in the prediction accuracy suddenly decreased.

The second hypothesis is that the benefit of multiple labeling increases when the accuracy of the crowd workers is low. When the accuracy of the crowd worker is low, the increase in accuracy from taking multiple labels is larger than when the accuracy is high. Therefore, we predict that the benefit of multiple labeling will increase when the accuracy of the crowd workers is low.

The third hypothesis is that the benefit of multiple labeling is greater when the task is easy for classifier. We define the classification for a small number of labels or classification using pretrained model as easy and assume that when the task complexity is low or the model is already proficient with the task, a small amount of training data will be sufficient for training. Therefore, we predict that the benefit of multiple labeling will increase when the task is easy.

The fourth hypothesis is that the optimal redundancy varies depending on the budget, the accuracy of the crowd workers, and the task complexity. Hypotheses 1–3 are based on the idea that the benefit of redundancy will vary depending on the budget, the accuracy of the crowd workers, and the complexity of the task. Therefore, we predict that the optimal redundancy level will vary depending on each condition.

This paper explores whether repeated labeling is effective or not through experiments on image classification, named entity recognition, and sentiment analysis.[b]

## 2. Related Work

There has been extensive research on correcting noisy annotations by aggregating redundant labels. These studies can be broadly divided into two problem settings: estimating true labels from crowdsourced responses, and improving the quality of the supervised model.

The majority of methods for the first problem setting are extensions of Dawid and Skenes' model [7], also known as the EM algorithm, which models the reliability of each worker and the true label estimation. Its many extensions include works by Simpson et al. [8] and Nguyen et al. [9], who applied the EM algorithm to named entity recognition (NER) tasks. Since EM algorithm [7] was originally designed for the integration of physician diagnoses, the tasks are modeled as independent from each other. However, words in a sentence are not independent from each other, and modeling the dependency is thus required in NER. For this reason, Simpson et al. [8] extended the EM algorithm to express the dependency between the preceding and the following word. While their model was successful, the increased number of

[b] This paper is an extended version of [6] with revisions for explaining additional experimental results, and detailed descriptions for the proposed method.

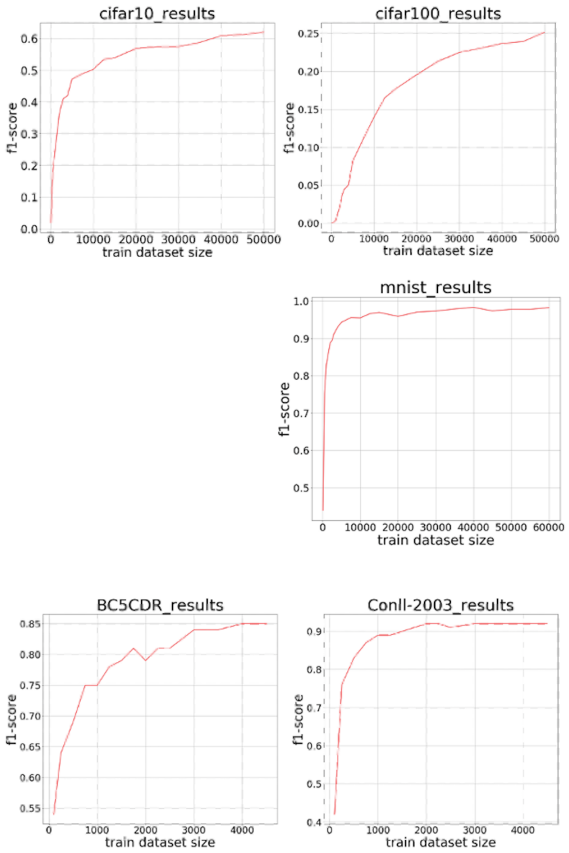


Fig. 2. Learning curves under different training models.

parameters resulted in the risk of overfitting and thus required additional labels. Nguyen et al. [9] pointed out that in crowdsourcing, most workers give only a few answers, which makes the estimation of a worker confusion matrix difficult. According to their study, models for solving this problem (such as the community-based Bayesian label aggregation model [10], which assumes that crowd workers can be grouped into a few different types and workers in each type represent similar confusion matrices) are effective in terms of the problem of only gaining a few labels from each worker, but they still require a high amount of labels since the probability distribution parameters need to be optimized. Therefore, they extended Dawid and Skenes' model for estimating worker ability to situations where the number of responses per worker is small by collapsing the confusion matrix to a confusion vector. In contrast, our work focuses on examining how redundancy affects classifier quality, so we do not look into each method. As we aim to compare the difference of the redundancy effect on image classification tasks , NER tasks, and sentiment analysis tasks, methods for specific situations are unsuitable for our experiments.

Our paper focuses on the second problem setting, which involves only the quality of the model and not the labels. The challenge of excluding noisy labels by predicting worker quality and aggregating redundant labels also appears in the second problem setting, but researchers

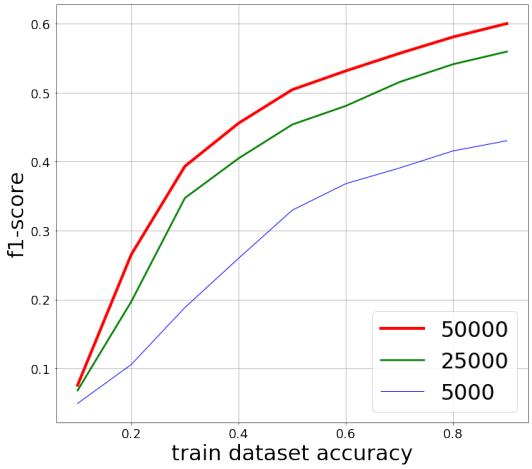


Fig. 3. Macro F-1 score of test data under different accuracy levels of training data.

[4, 5] have pointed out that there is also a chance of ground-truth data containing imperfect samples, meaning that the only difference between ground-truth labels and crowdsourced labels is the ratio of wrong labels.

Khetan et al. [11] extended Dawid and Skenes' EM algorithm [7] by using the prediction results of the classifier to estimate the worker's ability, thus making it possible to weigh the worker's answer without taking multiple labels. They examined the change in the performance of the classifier when 1) the classifier was given a small amount of training data with high accuracy by taking multiple labels and 2) the classifier was given a large amount of training data with low accuracy. Their results showed that the performance of the classifier without multiple labeling was higher than that with high redundancy. Although their work demonstrated that there are at least some cases where multiple labeling is not the best way to construct a training dataset, their experiments did not explore the effect of budget nor pretrained models, and were conducted only on image classification. In contrast, we set budget as an experimental variable and examined the effects on image classification tasks, NER tasks, and sentiment analysis tasks.

As for multiple-labeling and sentiment analysis, Sinha et als [12] work extended Dawid and Skenes' model [7] to allow faster convergence. As our work focuses on the effect of the pretrained model, we do not examine the speed.

## 3. Experiment setting

As stated in section 1, we came up with four hypotheses and examine them through experiments using synthetic data. In the experiments, we created synthetic workers and used them to generate crowdsourced labels. The synthetic crowdsourced data is fed to the classifier as training data. The redundancy that maximizes the performance evaluated by the macro F1 value of the test data is determined as the best redundancy for each experimental condition. The following three variables are defined in the experiments.

- Budget size

- Worker accuracy
- Task difficulty

The size of the training data is set as the budget divided by the redundancy, i.e., when the redundancy is 1, the size of the budget is the size of the training dataset. We set three values for the budget: the maximum value of the number of training data, $\frac{1}{2}$ of the maximum value, and $\frac{1}{10}$ of the maximum value. The accuracy of a worker's response is determined by a confusion matrix representing the worker's ability. The diagonal component of the confusion matrix represents the probability of the worker answering the correct label, and we can manipulate the accuracy of the synthetically sourced dataset by setting the value of the confusion matrix according to each experimental condition. By changing the amount of the diagonal component of the confusion matrix, we investigate the worker accuracy's effect on the optimal redundancy level.

We also conduct experiments with different tasks and different datasets to investigate the effect of task difficulty. Six datasets are used: three for the image classification task, two for the NER task, and one for sentiment analysis task.

The redundancy is a value that indicates the number of workers allocated to the same task. The budget divided by the redundancy is the size of the training dataset. In each experimental condition, we vary the redundancy level to 1, 5, 20, and 50 for the image classification and NER tasks and to 1, 5, 10, and 20 for the sentiment analysis tasks, and investigate the effect on the macro F1 value of the test dataset.

The size of the training data instances indicates the number of training data fed to the classifier. In the image classification task, a set of an image and a label is considered one data instance, and in the NER and sentiment analysis tasks, a sentence is considered one dataset.

The following two methods are used in aggregating worker's answers for image classification and NER.

- Simple majority voting
- Weighted majority voting

In sentiment analysis, we only used simple majority voting since the experiment was focused on the effect of redundancy when using pretrained models and we did not want to make the experiment too complex. In simple majority voting, each worker's vote is treated equally, and the worker's answers are aggregated by majority voting. In weighted majority voting, the worker's answers are weighted according to the worker's accuracy level estimated by Dawid and Skenes' model [7], and then the majority vote is taken to aggregate the worker's answers. Each worker's accuracy level is provided by the confusion matrix.

The experiments are carried out in the following steps.

1. The number of training data is determined by the dataset, budget size, and redundancy.
2. Create the synthetic data for each worker. Details are described in this section.
3. Aggregate the synthetic data of workers by simple majority vote or weighted majority vote to create a training dataset.

4. Feed the training dataset to the classifier and train it.

5. Run the classifier trained in step 4 on the test dataset and determine the macro F1 score.

Our experiments deal with an image classification task, NER task, and sentiment analysis task. The datasets we use are shown in Table 1. In the image classification task, we used the MNIST dataset [3], which is a dataset of handwritten images and labels of numbers 0–9, the CIFAR10 dataset [2], an image dataset for object recognition classified into ten classes, and the CIFAR100 dataset [2], an image dataset for object recognition classified into 100 classes. In the NER task, we use the Conll-2003 dataset [13], which defines the names of persons, places, organizations, etc. extracted from news articles as entities, and the BC5CDR dataset [14], which defines pharmacological terms as entities. In the sentiment analysis task, we use the IMDB dataset [15], which is a movie review dataset used for binary sentiment classification.

We build an independent machine learning model for each task. For the image recognition, we use the convolutional network model from PyTorch Tutorials [c]. For NER, we use a model using Bi-LTSM CRF [16] and predict the labels for each word. The prediction results are checked for each entity, and the correct answer is obtained when the entire entity is extracted. In sentiment analysis, we use a model pretrained by BERT [17]. BERT consists of pretrained deep bidirectional representations from unlabeled text and can be fine-tuned for many natural language tasks by training an additional output layer. In the image classification task and sentiment analysis task, a confusion matrix is created from a probability distribution. The simulation workers are created as follows. First, we create a confusion matrix for the number of simulated workers that represents the response capability of the workers. The answers of the simulated workers are generated from the confusion matrix and the true answers of the task. We take samples from the beta distribution and set $W$ to be the list of simulation workers to ensure that the average accuracy of the workers' responses matches the determined experimental condition $\alpha$. We set $C^w$ to be the response accuracy of worker $w \in W$. Then, we sample the number of labels from the beta distribution for each worker so that the sampled value is the average of the diagonal components of the confusion matrix for each worker. For the beta distribution $Sx^{a-1}(1-x)^{b-1}$ (where S is a normalization constant and $a$, $b$ are parameters), the value of parameter b is fixed at $b = 10$, and parameter $a$ is set so that the expected value of the distribution $\alpha$ becomes the average of the response accuracy of the workers determined by the experimental conditions from Eq. (1).

$$a = \frac{\alpha b}{1-\alpha} \tag{1}$$

$K$ is the number of labels in a task. For each worker $w \in W$ and each label $1 \leq i \leq K$, we generate a sample $c_i^w$ from the beta distribution described above, and let each be the value of each diagonal component of the simulation worker's confusion matrix. The off-diagonal values of the confusion matrix are sampled from the uniform distribution on $[0, 1]$ and normalized so that the sum becomes $1 - c_i^w$.

[c] https://pytorch.org/tutorials/beginner/blitz/neural_networks_tutorial.html

In the NER task, a confusion matrix is created based on the actual responses of the workers. The frequency of occurrence differs greatly depending on the label. In particular, O-tags, which indicate that the label is not an entity, account for more than half of the correct answers in most cases. Therefore, a decrease in the accuracy of an O-tag answer has an extremely large impact on the overall answer, and should not be treated equally with the values of other labels in the confusion matrix. In NER, entities are labeled by two types of tags: B-tags, which represent the starting position of the named entity, and I-tags, which follow B-tags. It is natural to assume that the answer-ability of B-tags and I-tags in each named entity of a worker has dependence on each other. For these reasons, we believe that in named entity extraction, it is inappropriate to create a confusion matrix of simulation workers by obtaining samples from a specific distribution, as is done in image classification. Therefore, we create a confusion matrix representing the worker's answering ability from the actual worker's answers.

## 4. Results

Figs. 4, 5, and 6 show the experimental results. Fig. 4 shows the results of the simple majority voting experiment, and Fig. 5 shows the results of the weighted majority voting experiment. For Figs. 4 and 5, the rows indicate the dataset, and the columns indicate the accuracy of the workers' responses.

Fig. 6 shows results of experiment using pretrained models. In Fig. 6, the columns indicate the accuracy of the workers' responses. The horizontal axis of each graph shows the degree of redundancy and the vertical axis shows the macro F1 value in the test dataset. For each experimental condition, ten experiments were conducted independently. The solid line indicates the mean value of the results, and the colored area indicates the maximum and minimum values of the experimental data.

Comparing Figs. 4 and 5, we can see that the overall results of each dataset were quite similar. Under experiments not using pretrained models, the slope of the graph increased as the budget decreased. This shows that when the budget is small, the benefit of redundancy is small, and when the budget is large, the benefit of redundancy is relatively large. This supports Hypothesis 1, which states that the benefits of redundancy increase as the budget increases.

Also, under conditions not using pretrained models, the slope of the graph increased as the accuracy of the worker's response increased. This indicates that the benefits of redundancy are greater when the response accuracy of the worker is low and relatively smaller when the response accuracy of the worker is high. This supports Hypothesis 2, which states that the benefit of redundancy increases when the response accuracy of the worker is low.

In the experiment using CIFAR100, regardless of the size of the budget or the accuracy of the worker's response, results in redundancy level 1, which means not taking multiple labels had the highest scores, and the slope of the graph became smaller as the redundancy level increased under all conditions. The higher the accuracy of the worker's response, the larger the slope of the graph, and the smaller the benefit of reducing the redundancy level.

In the experiment using CIFAR10, regardless of the size of the budget or the accuracy of the worker's response, results in redundancy level 1, meaning not taking multiple labels had the highest scores. In results with the worker response of accuracy 0.3 and the budget size of

50,000 or 25,000, the descent of the macro F1 value was nearly constant when the redundancy level increased. In the results with the worker response accuracy of 0.9 and the budget size of 50000 or 25000, the slope was steep when the redundancy changed from 1 to 5 but became gentle as the redundancy became bigger. When the redundancy level increased, the macro F1 value decreased. This indicates that the benefit of reducing the redundancy level decreases as the accuracy of the worker's response increases.

In experiments using MNIST, regardless of the size of the budget or the accuracy of the worker's response, results in redundancy level 1, meaning not taking multiple labels had the highest scores. In the results with budget sizes of 60,000 or 30,000, the descent of the macro F1 value was nearly constant when the redundancy level increased. In the results with the budget size of 6000, the slope was steep when the redundancy changed from 1 to 5, but became gentle as the redundancy became bigger. The slope of the graph was extremely small when the accuracy of the worker's response was 0.3 and the size of the budget was 60000, which means that the benefit of redundancy in improving the accuracy of the training dataset and the disadvantage of reducing the data amount is close to being balanced.

In the experiment using the Conll-2003 dataset, the results in redundancy level 1 had the highest scores, except for when the worker response accuracy was 0.3 and the budget size was 4500. In the results with the worker response accuracy of 0.9 or 0.6, the drop in the macro F1 value when the redundancy level increased was largest when the redundancy level was changed from 1 to 5, and as the redundancy level increased, the amount of decrease of the macro F1 value became smaller. In the results with the worker response accuracy of 0.3 and the budget size 4500, the macro F1 value decreased when the redundancy level was changed from 1 to 5, and then slightly increased as the redundancy level increased. This result is discussed in more detail in section 5.

In the experiment using the BC5CDR dataset, regardless of the size of the budget or the accuracy of the worker's response, results in redundancy level 1 had the highest scores. The drop in the macro F1 value when the redundancy level increased was largest when the redundancy level changed from 1 to 5, and the slope was steepest among all experimental conditions. As in the other experimental results, the slope became gentle as the redundancy level increased.

For the NER task, there were no significant differences in the Conll-2003 and BC5CDR datasets. For the image classification tasks, the slope of the graph was highest for the CIFAR100 dataset and lowest for MNIST. CIFAR100 classifies 100 colored images and MNIST classifies ten black and white images, meaning CIFAR100 is the most difficult task for the classifier and MNIST the least difficult. This indicates that the higher the difficulty of the task for the classifier, the larger the slope of the graph becomes. These results show that Hypothesis 3, which states that the benefit of redundancy increases with decreasing task difficulty for the classifier, is supported.

In most of the experimental conditions in Figs. 4 and 5, the graphs showed a downward trend, indicating that results in redundancy level 1 had the highest scores. This supports Hypothesis 4, which states that the optimal redundancy level varies depending on the budget, the accuracy of the crowd worker's answers, and the difficulty of the task.

In the experiment using a pretrained model, redundancy level 1 didnt always have the highest scores. The results did not show a downward trend like the image classification and

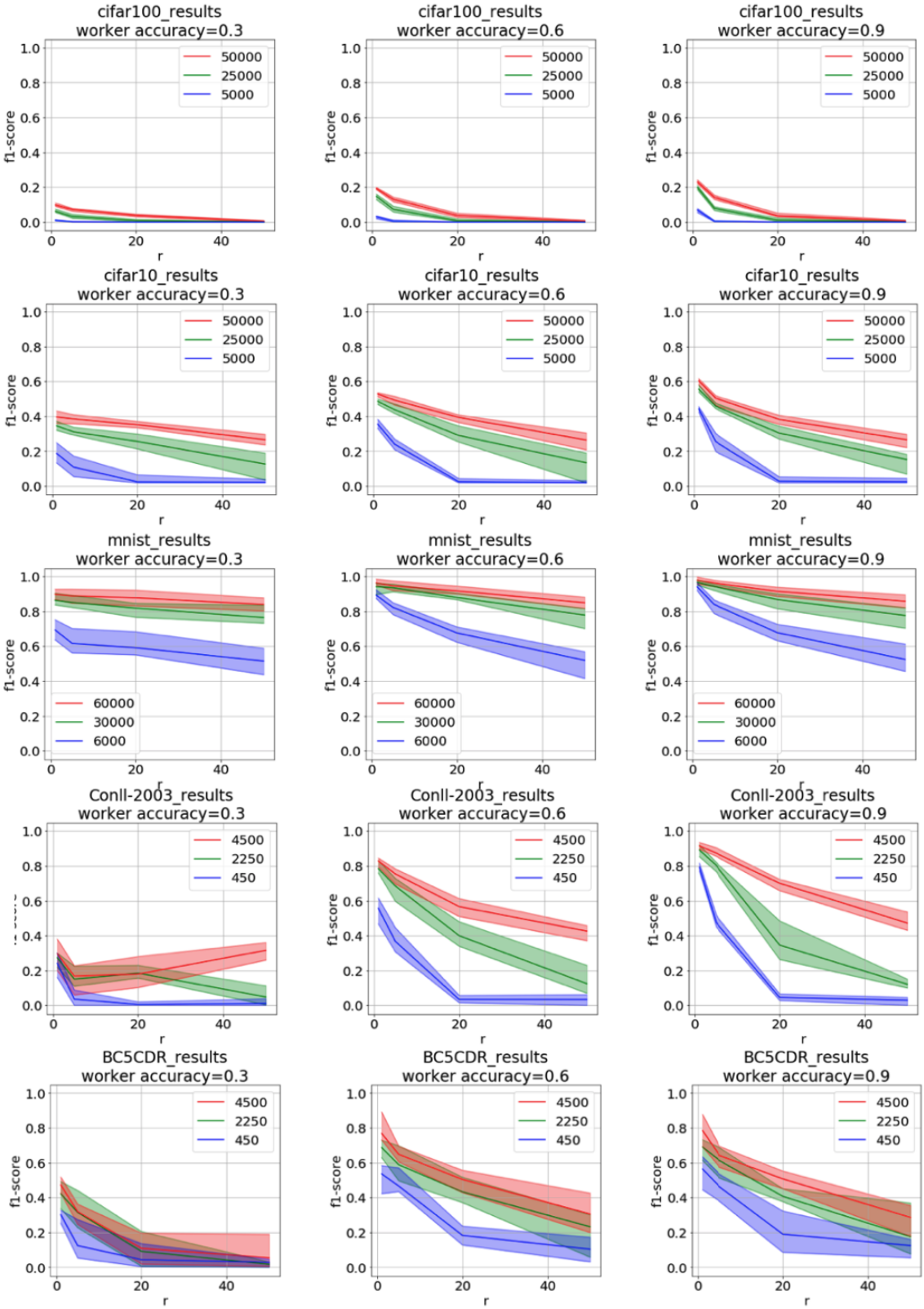


Fig. 4. Training from simple aggregation.

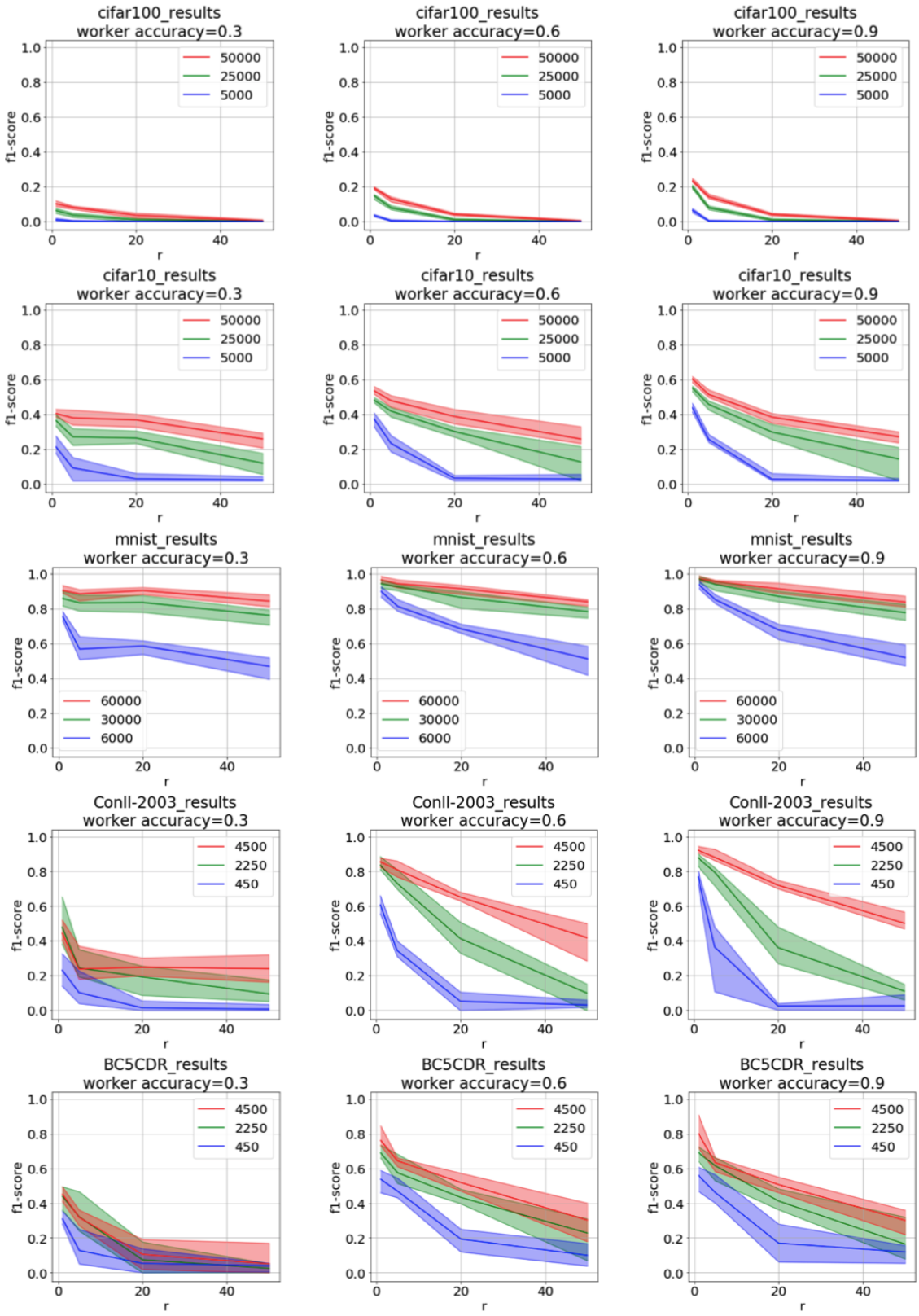


Fig. 5. Training from weighted aggregation.

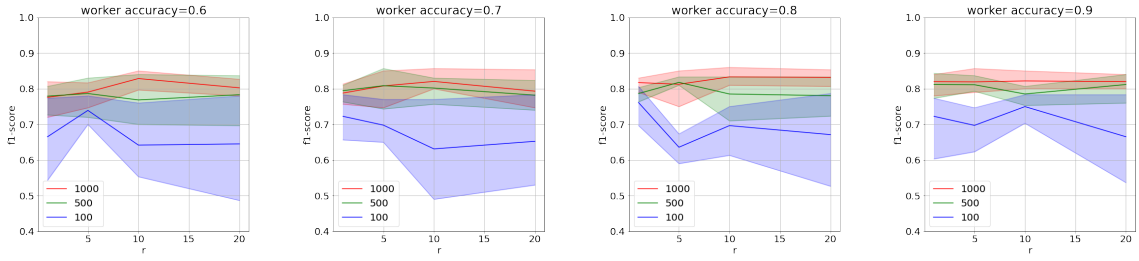


Fig. 6. Using pretrained model.

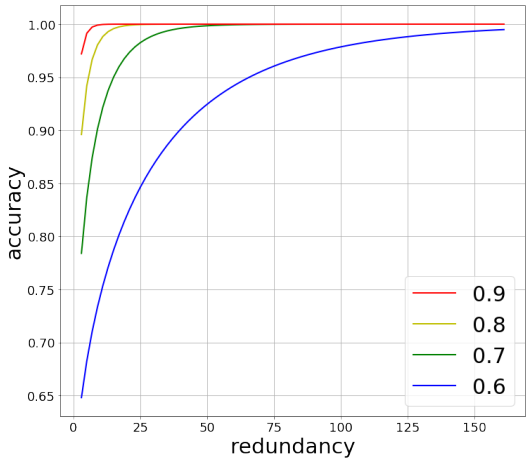


Fig. 7. Efficiency of redundancy.

NER results. In most of the experimental conditions in Fig. 6, the graph was flat, showing that the advantage and disadvantage of increasing the redundancy was balanced.

This indicates that Hypothesis 3, which states that when the task is easy for the classifier, taking redundant labels decreases the performance of the classifier, is rejected when the argument is about pretrained models. Hypothesis 3 seems to be convincing when the task complexity is the indicator of task difficulty, but not when considering the prior knowledge of the model.

## 5. Discussion

Figs. 7 and 8 are reference data to discuss the results of the experiment. Fig. 7 shows that the probability of obtaining the correct answer to a binary classification when a majority vote is taken by the workers, assuming that the accuracy of the workers is constant. The vertical axis shows the probability of obtaining the correct answer, the horizontal axis shows the degree of redundancy, and lines show the results for each worker accuracy. Fig. 8 shows the results of an experiment to investigate the change in classification accuracy when the accuracy of the training data given to the classifier is varied. The horizontal axis shows the macro F1 value

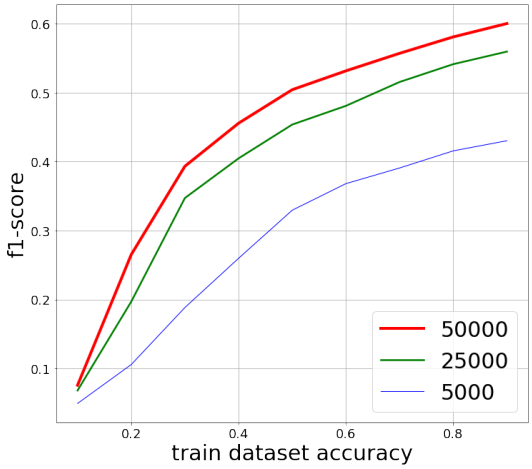


Fig. 8. Effect of training dataset accuracy.

of the prediction for the test dataset of the classifier, the horizontal axis shows the accuracy of the training dataset, and the lines show the results for each training data.

Looking back to Figs. 4 and 5, they show that the benefit of increasing the redundancy level decreases as the size of the budget decreases. This is because when the amount of training data gets smaller, the benefit of increasing the amount of training data becomes larger, and as the budget gets smaller, the benefit of increasing the redundancy becomes relatively smaller. We can also see that the benefit of increasing the redundancy level decreases as the worker accuracy increases.

In situations where the worker accuracy is high, the answer rate converges with a small number of workers. Fig. 7 shows that after convergence, there is only a little benefit from the increase in redundancy. Furthermore, Fig. 8 shows that the increase of macro F1 value by the increase of the training dataset accuracy becomes gentle as the training dataset accuracy gets higher. This means that the higher the worker accuracy, the smaller the benefit of increasing the redundancy.

From the results of the experiment using the Conll-2003 dataset, we found that when the size of the budget was 4500 and the accuracy of the worker's response was 0.3, the macro F1 value dropped rapidly when the redundancy changed from 1 to 5, and then increased as the redundancy level increased.

Fig. 2 shows that in the Conll-2003 dataset, the prediction accuracy increased rapidly until the number of training data reached a certain value, and then the growth of the prediction accuracy became very small. Therefore, in the experiments using the Conll-2003 dataset with a large budget, the benefit of increasing the redundancy level was significant.

Furthermore, when the accuracy of the worker response was low, the benefit of increased redundancy was not significant at low redundancy levels. This is why, in the experiment using the Conll-2003 dataset with a budget size of 4500 and a worker response accuracy of 0.3, changing the redundancy from 1 to 5 resulted in a sharp decrease in the macro F1 value.

In experiments using pretrained models, gaining redundancy level did not give much impact on the classifier. The sentiment analysis model seemed to learn well even when the

training data amount was relatively small, presumably because the pretrained model is quite powerful.

Machine learning acquires a stable predictive performance by training based on the trends of the entire training data. Therefore, the training of the classifier itself can be interpreted as a majority vote on a set of similar data instances. This implies that for tasks that can be solved by statistical approaches, there is no need for majority voting on repeated human labels in advance. However, when the model has a powerful prior knowledge, low accurate labels seem to mess up the results, and label quality has higher priority.

## 6. Conclusion

In this paper, we demonstrated that the prediction accuracy of a classifier is at its highest when the redundancy is 1, regardless of the worker's accuracy or the difficulty of the task for the classifier. In particular, when the size of the budget is small, the accuracy of the worker's answer is high, and when the difficulty of the task for the classifier is high, the benefit of reducing the redundancy level is large. On the other hand, when the model acquires prior knowledge, taking multiple labels may be beneficial since label quantity is relatively less important.

We focused on image classification, NER, and sentiment analysis tasks in this work, but crowdsourcing training data are also utilized in more complex tasks such as translation and summarization. In future work, we aim to examine the significance of redundancy in these tasks.

## Acknowledgements

This work was supported by JSPS KAKENHI Grant Number 19K20333, 21H03552 and JST CREST #JPMJCR16E3 AIP Challenge.

## References

1. Rion Snow, Brendan OConnor, Daniel Jurafsky, and Andrew Ng. 2008. Cheap and Fast But is it Good? Evaluating Non-Expert Annotations for Natural Language Tasks. In Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 254263.
2. Alex Krizhevsky. 2009. Learning multiple layers of features from tiny images. Technical Report. 2009.
3. Yann LeCun and Corinna Cortes. 2010. MNIST handwritten digit database. (2010). http://yann.lecun.com/exdb/mnist/
4. Joshua C. Peterson, Ruairidh M. Battleday, Thomas L. Griffiths, and Olga Rus- sakovsky. 2019. Human uncertainty makes classification more robust. CoRR abs/1908.07086. arXiv:1908.07086 http://arxiv.org/abs/1908.07086
5. Xinbin Zhang. 2017. A Proof of Bad Handwriting in MNIST Training Dataset Making CNN to Predict Good Handwriting Wrong V2. SSRN (2017). https: //ssrn.com/abstract=3063083
6. Ayame Shimizu and Kei Wakabayashi. Examining Effect of Label Redun- dancy for Machine Learning Using Crowdsourcing, page 8794. Association for Computing Machinery, New York, NY, USA, 2021.
7. Dawid Philip and Skene Allan. 1979. Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm. Applied Statistics 28, 1 (1979), 2028.
8. Edwin Simpson and Iryna Gurevych. 2018. Bayesian Ensembles of Crowds and

Deep Learners for Sequence Tagging. CoRR abs/1811.00780 (2018). arXiv:1811.00780 http://arxiv.org/abs/1811.00780

9. An Thanh Nguyen, Byron Wallace, Junyi Jessy Li, Ani Nenkova, and Matthew Lease. 2017. Aggregating and Predicting Sequence Labels from Crowd Annota- tions. In Proceedings of the 55th Annual Meeting of the Association for Computa- tional Linguistics (Volume 1: Long Papers). 299309.
10. Matteo Venanzi, John Guiver, Gabriella Kazai, Pushmeet Kohli, and Milad Shok- ouhi. 2014. Community-Based Bayesian Aggregation Models for Crowdsourcing. In WWW 14 Proceedings of the 23rd international conference on World wide web. ACM, 155164.
11. Ashish Khetan, Zachary C. Lipton, and Anima Anandkumar. 2018. Learning From Noisy Singly-labeled Data. In International Conference on Learning Repre- sentations.
12. Vaibhav B Sinha, Sukrut Rao, and Vineeth N Balasubramanian. Fast dawid-skene: A fast vote aggregation scheme for sentiment classification, 2018.
13. Erik Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003. 142147.
14. Shang Jingbo, Liu Liyuan, Ren Xiang, Gu Xiaotao, Ren Teng, and Han Jiawei. 2018. Learning Named Entity Tagger using Domain-Specific Dictionary. CoRR abs/1809.03599 (2018). arXiv:1809.03599 http://arxiv.org/abs/1809.03599
15. Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. (2011). Learning Word Vectors for Sentiment Analysis. The 49th Annual Meeting of the Association for Computational Linguistics (ACL 2011).
16. Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural Architectures for Named Entity Recognition. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 260270.
17. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language un- derstanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 41714186, Minneapolis, Minnesota, June 2019. Association for Computational Lin- guistics.