COMP6258 REPRODUCIBILITY CHALLENGE: DOES LABEL SMOOTHING HELP DEEP PARTIAL LABEL LEARNING?

Yanzhe S. Zhang, William Toland, Jack Booth, Winston McCarthy
{yz25g21, welt1g21, jgb1g21, wm2g20}@soton.ac.uk

ABSTRACT

This report verifies the reproducibility of the paper "Does Label Smoothing Help Deep Partial Label Learning?". We first replicate the original experiments, then extend them by randomly selecting candidate labels instead of choosing the top-k most probable ones. Our results confirm that label smoothing improves performance in deep partial label learning, though we do not observe the same effects on pre-logits. Additionally, we find that using top-k labels may not represent the worst-case scenario for classification accuracy.

1 INTRODUCTION

In this report, we attempt to reproduce and extend the works shown in Gong et al. (2024), "Does Label Smoothing Help Deep Partial Label Learning?". We first start by analysing the paper, reviewing the experiments it provides to decide whether they are sufficient to make the claims found in the original paper. We attempt to reproduce the experiments using our own code base, analysing the results to verify the findings. Finally we extend the experiments using a different methodology to further test the claims made in the original paper.

2 PAPER ANALYSIS

The paper Gong et al. (2024) explores the concept of using Label Smoothing (LS) to improve Deep Partial Label Learning, where the machine's goal is to identify a single true label from a set of candidate labels. The paper discusses the LS as a solution for the classification performance of Deep Neural Networks (DNN), forcing the DNN to be less certain of any single predictions. The paper claims that LS encourages activation of the penultimate layer to be close to the template of the correct class, and equally distant to the templates of incorrect classes on PLL datasets, regardless of architecture.

Our reproducibility report asks the following questions:

- Can the results of the original experiments used in the paper be recreated using the datasets and methodologies originally planned?
- Do the findings still hold when selecting random labels, rather than the top-k predictions?

These questions are designed to investigate the clarity of the original methodology, determine whether we can reach the same conclusions as the paper, and test the assumptions made.

2.1 EXPERIMENTAL SETUP AND METHODOLOGY

For our experiments, we conduct all experiments using the original setup as described in the paper.

We conduct our experiments on the four datasets used in the paper:

Dataset	Model	k	Avg.#CL
Fashion-MNIST	LeNet-5	6	3, 4, 5
Kuzushiji-MNIST	LeNet-5	6	3, 4, 5
CIFAR-10	ResNet-18	6	3, 4, 5
CIFAR-100	ResNet-56	20	7, 9, 11

We used SGD with a momentum of 0.9 and 1e-3 weight decay. We used a mini-batch size of 128, learning rate of 0.01 and 200 epochs. We set the smoothing rate at 0.1, 0.3, 0.5, 0.7 and 0.9, with a weighting parameter of 0.9. ResNet models are constructed based on He et al. (2016), and since ResNet-56 is not a typical architecture specified in the paper, we decided to use [3, 4, 9, 3] blocks in each convolutional layer to make a total of 56 layers.

3 Reproduced and Further Experiments

We first focus on reproducing the experiments in the original paper. For these, using the setup, datasets and methodology previously stated, we attempt to validate: Whether LS is effective for deep PLL; and when LS benefits deep PLL.

As with the original paper, to test whether LS is effective for deep PLL, we conduct both experiments to measure the effect of label smoothing on classification performance, and also the effect of label smoothing on pre-logits. The paper does not specify the model trained on the original datasets to generate top-k partial labels, we have decided to use the same model and hyper-parameters as for the experiments.

To test the effect of LS on classification performance, we compare the test accuracies of the models above, with the listed parameters. To analyse the effect of LS on the pre-logits, we create illustrations as in the original paper to compare the pre-logits with and without LS under the noise levels stated. We also conduct experiments to determine when LS benefits PLL as in the paper, by comparing prediction accuracies at varying smoothing rates.

In section 5.1, the paper claims choosing from the top-k labels in the label space in descending order of probability makes the label space more competitive. We aim to test this claim by comparing the results found in the previous experiments to tests selecting random labels, rather than the top-k. If the claim made by the paper is true, then selecting random labels rather than from the top k would retain or increase performance from the previous set of tests. If the tests perform worse, then the results found in the previous tests may only hold under certain circumstances which could indicate selecting the top-k results provides additional information or is a specific case in which the method performs better.

4 Results

Though the entire experiment proved to be reproducible, the results varied. We chose to recreate the smoothed data set used in the original paper. For the partial label experiment, the models were trained on a standard computer with a powerful GPU. This took 35 hours to train and test. The random partial label experiment was reproduced on iridis 5. This took 2.5 days to train and test. The models in both experiments used about 3GB of memory.

4.0.1 Classification performance

Tables 1 and 2 shows the classification performance for each dataset with and without LS for each noise level. The smoothing rate with the highest accuracy is chosen for each dataset and noise level.

Dataset	Architecture	Acc (Avg.#CL=3)	Acc (Avg.#CL=4)	Acc (Avg.#CL=5)
Fashion-MNIST	W/O LS	91.12	91.11	90.25
	W/ LS	**93.02**	**93.63**	**94.92**
Kuzushiji-MNIST	W/O LS	89.23	71.50	79.54
	W/ LS	**93.93**	**94.21**	**87.5**
CIFAR-10	W/O LS	63.20	58.65	64.96
	W/ LS	**77.3**	**78.99**	**82.21**

Table 1: Accuracy results with and without LS

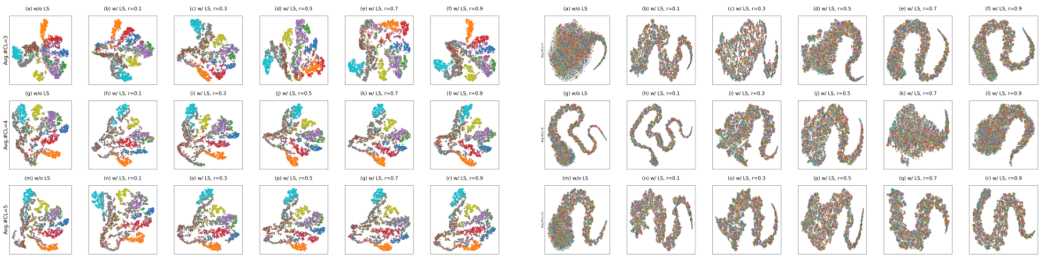
Dataset	Architecture	Acc (Avg.#CL=7)	Acc (Avg.#CL=9)	Acc (Avg.#CL=11)
CIFAR-100	W/O LS	16.69	18.99	19.86
	W/ LS	**24.22**	**28.76**	**32.99**

Table 2: Accuracy results with and without LS for CIFAR-100

These results differ from those in Gong et al. (2024) in various ways. Firstly, our results show that the average accuracy for most datasets without LS tends to be higher than in the original experiment, with the accuracy for all datasets without LS being higher except for CIFAR-100. This is despite using an identical experimental setup to that claimed in the original paper. The accuracy of CIFAR-100 was significantly lower both with and without LS.

Despite this, the accuracy with LS still shows a significant improvement over the accuracy without LS. Despite the differences in the results to that of the paper, we believe it is still possible with these results to claim LS improves performance for deep PLL.

4.0.2 Effect on pre-logits



(a) Visualisation of pre-logits for Fashion-MNIST

(b) Visualisation of pre-logits for CIFAR-100

Figure 1: Pre-logit visualisations

Figure 1 shows the results of visualising the pre-logits of the datasets for varying noise levels and smoothing rates. Figure 1a shows the pre-logits for Fashion-MNIST. Comparing this to the original paper results, we can see the results are similar, however the classes show more variation than the original results. However, for the other results the pre-logits look much more different. Figure 1b shows the pre logit visualisations for the CIFAR-100 dataset. These are vastly different from those shown in the paper, with the classes failing to group into clusters. We do not believe from these figures that we can make the claim that LS encourages activations in the penultimate layer to be close to the template of the correct class.

4.1 When LS benefits deep PLL

Dataset	Smooth Rate	Acc (Avg.#CL=3)	Acc (Avg.#CL=4)	Acc (Avg.#CL=5)
Fashion-MNIST	0.1	92	87.77	63.92
	0.3	85.09	90.60	88.54
	0.5	85.56	91.07	88.60
	0.7	92.46	87.51	89.30
	0.9	**93.02**	**93.63**	**94.92**
Kuzushiji-MNIST	0.1	65.90	56.53	53.62
	0.3	88.10	78.87	59.71
	0.5	89.66	66.62	80.26
	0.7	88.86	87.84	81.06
	0.9	**93.93**	**94.21**	**87.50**
CIFAR-10	0.1	73.49	74.08	74.91
	0.3	75.13	75.61	79.90
	0.5	76.01	77.09	80.20
	0.7	76.64	77.32	81.66
	0.9	**77.30**	**78.99**	**82.21**

Table 3: Accuracy results by smoothing rate and noise level

Tables 3 and 4 show the accuracy by each smoothing rate and noise level. These results show that all datasets tested prefer large smoothing rates, as expected from the results of the original paper.

Dataset	Smooth Rate	Acc (Avg.#CL=7)	Acc (Avg.#CL=9)	Acc (Avg.#CL=11)
CIFAR-100	0.1	19.93	24.83	30.35
	0.3	20.42	28.29	29.17
	0.5	19.23	22.47	28.99
	0.7	19.62	23.56	30.03
	0.9	**24.22**	**28.76**	**32.99**

Table 4: Accuracy results by smoothing rate and noise level for CIFAR-100

Unlike the original paper, we found all datasets and noise levels had best performance at the largest smoothing rate. These results do not entirely fit the claims of the original paper, instead showing all datasets prefer more smoothing.

4.2 WITH RANDOM LABELS

Dataset	Architecture	Acc (Avg.#CL=3)	Acc (Avg.#CL=4)	Acc (Avg.#CL=5)
Fashion-MNIST	W/O LS	87.72	85.88	50.03
	W/ LS	**89.02**	**87.18**	**50.20**
Kuzushiji-MNIST	W/O LS	88.45	84.93	79.13
	W/ LS	**93.84**	**93.15**	**89.47**
CIFAR-10	W/O LS	58.03	53.04	55.54
	W/ LS	**75.38**	**74.69**	**74.25**

Table 5: Accuracy results with and without LS with random labels

Dataset	Architecture	Acc (Avg.#CL=7)	Acc (Avg.#CL=9)	Acc (Avg.#CL=11)
CIFAR-100	W/O LS	10.21	12.08	11.93
	W/ LS	**13.21**	**14.06**	**13.21**

Table 6: Accuracy results with and without LS for CIFAR-100 with random labels

Tables 5 and 6 show the testing accuracy with and without LS with random candidate labels. These results differ from those of Tables 1 and 2, showing more variation in accuracy depending on the noise level in some data sets such as Fashion-MNIST and a lower average classification accuracy overall, both with and without LS. This shows the claim that the top-k results would always produce the worst case, and therefore the worst results, does not hold. Despite this, the accuracy is still higher with LS in all cases, and as such the claim that LS improves classification accuracy for deep PLL still holds.

5 CONCLUSION

Our results show the paper by Gong et al. (2024) has some valid claims, with our findings agreeing that LS improves classification accuracy for deep PLL. However, we also find many inconsistencies between our results and the original, including the failure for the pre-logits to show tight clusters, meaning lack of evidence for LSS to encourage activations in the penultimate layer to be close to the template of the correct class. Our findings also show LS increases accuracy with higher smoothing rates consistently, rather than the papers fining of each dataset and model having a different optimal smoothing rate. Finally, we show the assumption that selecting from the top-k most probable classes would always be the worst case is false, and cannot be safely assumed.

REFERENCES

Xiuwen Gong, Nitin Bisht, and Guandong Xu. Does label smoothing help deep partial label learning? In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. doi: 10.1109/CVPR.2016.90.