Alleviating Data Imbalance Issue with Perturbed Input During Inference

Kanghao Chen[1], Yifan Mao[1], Huijuan Lu[1], Chenghua Zeng[1], Ruixuan Wang[1,2(✉)], and Wei-Shi Zheng[1,2]

[1] School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China
wangruix5@mail.sysu.edu.cn

[2] Key Laboratory of Machine Intelligence and Advanced Computing, MOE, Guangzhou, China

Abstract. Intelligent diagnosis is often biased toward common diseases due to data imbalance between common and rare diseases. Such bias may still exist even after applying re-balancing strategies during model training. To further alleviate the bias, we propose a novel method which works not in the training but in the inference phase. For any test input data, based on the difference between the temperature-tuned classifier output and a target probability distribution derived from the inverse frequency of different diseases, the input data can be slightly perturbed in a way similar to adversarial learning. The classifier prediction for the perturbed input would become less biased toward common diseases compared to that for the original one. The proposed inference-phase method can be naturally combined with any training-phase re-balancing strategies. Extensive evaluations on three different medical image classification tasks and three classifier backbones support that our method consistently improves the performance of the classifier which even has been trained by any re-balancing strategy. The performance improvement is substantial particularly on minority classes, confirming the effectiveness of the proposed method in alleviating the classifier bias toward dominant classes.

Keywords: Data imbalance · Perturbed input · Prediction bias

1 Introduction

Deep learning has been widely applied to intelligent diagnosis of various diseases from medical images [6,7,17]. The success of intelligent diagnosis often depends on large annotated data for model training. However, while it is relatively easy to collect and annotate large amount of data for commonly encountered diseases, it is very challenging (if not impossible) to collect enough data for various rare diseases. Such data imbalance across diseases in nature often causes diagnostic bias toward common diseases by the intelligent system [1,11]. To improve

K. Chen and Y. Mao—The authors contribute equally to this paper.

© Springer Nature Switzerland AG 2021
M. de Bruijne et al. (Eds.): MICCAI 2021, LNCS 12905, pp. 407–417, 2021.
https://doi.org/10.1007/978-3-030-87240-3_39

the diagnostic performance of the intelligent system especially for those rare diseases, it is crucial to investigate effective learning strategies which can help the intelligent system successfully learn the features of both common and rare diseases from the imbalanced disease dataset.

Multiple re-balancing approaches have been developed to alleviate the data imbalance issue. Among them, data re-balancing and cost-sensitive re-weighting have been well explored and commonly adopted. The basic idea of data re-balancing is to use similar amount of data for each class to train the intelligent system, either by over-sampling the limited data for the small-sample (minority) classes [3,9] or under-sampling the data for larger-sample (dominant) classes [15]. One special over-sampling strategy especially for training deep learning models is data augmentation [23] which can generate almost unlimited transformed data for minority (and dominant) classes. Different from the data re-balancing strategies on the model input side, cost-sensitive re-weighting strategies adjust the importance of loss terms in the loss function during model training, either at the class level or at the instance (individual data) level. At the class level, setting larger cost weight for minority classes has been widely adopted, where the weight is inversely proportional to the class frequency [12,25]. At the instance level, the weight for each individual training data can be adjusted based on the difficulty of being correctly classified, with well-known techniques like boosting [3] or focal loss [16]. Besides data re-balancing and cost-sensitive re-weighting strategies, another set of strategies focus on the intelligent model itself, including transfer learning and model ensembling which have become routine to improve classification performance [25,26]. However, all these strategies can just alleviate the data imbalance issue to some extent, in the sense that the well-trained model is still more or less biased toward dominant classes during inference [14,28]. Recent studies found that widely used strategies to handle data imbalance often downgrade feature representation ability in the deep learning model, while the deep learning model without adopting any re-balancing strategy has a more generalizable feature extractor [28]. With this observation, it is proposed to first learn a generalizable feature extractor regardless of data imbalance, and then the model head for classification is re-trained with certain re-balancing strategy [2,14,28].

Different from the aforementioned approaches which alleviate the imbalance issue in the training phase, this paper proposes a simple yet effective approach which works not in the training phase but in the testing phase, aiming to further alleviate the model's prediction bias toward dominant classes if existing. This is achieved by slightly perturbing the test data before fed to the model based on a special single-data loss function. Different from adversarial attack methods [4,8,19,21] which try to make models make wrong predictions, the proposed approach here aims to alleviate the model bias toward dominant classes. Extensive evaluations on multiple medical image datasets and model backbones support that the proposed approach, built on models trained with various re-balancing strategies, is effective in further improving the classification performance particularly on minority classes.

2 Methodology

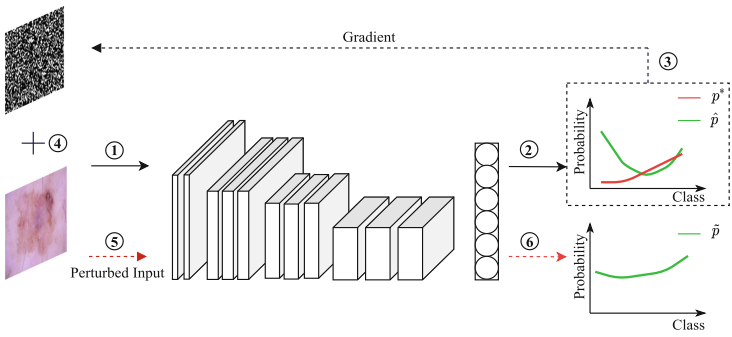

Fig. 1. Demonstrative procedure of our method. Circled number indicates the order of processing or corresponding signal flow. During inference, perturbation is computed based on gradient of a difference measure function over input pixels, and then the perturbed input is fed to the CNN model to obtain the final prediction.

The goal of this study is to improve the performance of any classifier trained on an imbalanced dataset, such that the classifier is less biased toward dominant classes in prediction. Different from most existing methods which focus on the classifier training phase, our method focuses on the testing (i.e., inference) phase. Given any classifier already well-trained on the imbalanced dataset and one test data to be classified, the intuitive idea of our method is to perturb the test data such that the classifier would be slightly inclined toward minority classes during inference. While such perturbation could downgrade the classification performance on dominant classes, it largely improves the performance on minority classes and the overall classification performance. The classification improvement on minority classes is crucial especially when missing diagnosis of rare diseases would cause serious consequence for patients.

The proposed approach is demonstrated in Fig. 1. Consider a convolutional neural network (CNN) classifier well trained based on an imbalanced dataset, where the number of training data for the c-th class is denoted by n_c, $c \in \{1, 2, \ldots, C\}$. Assume the classifier predictions over multiple test data are statistically biased toward dominant classes due to imperfect model training with the imbalanced training dataset. Then, for any test data $\mathbf{x}$ and the correspondingly original probability distribution output $\mathbf{p}$ of the classifier, the higher probability prediction in $\mathbf{p}$ would be likely biased toward the dominant classes. To alleviate such prediction bias, one naive way is to manually decrease the probability predictions by certain amount for dominant classes and increase the probability predictions by certain amount for minority classes. However, it would be very challenging and ad-hoc to manually determine the amount of prediction adjustment for each class. In this study, inspired by the strategy of

generating adversarial examples, we propose a strategy to automatically perturb the input data $\mathbf{x}$ such that the classifier output $\tilde{\mathbf{p}}$ for the perturbed input $\tilde{\mathbf{x}}$ is slightly biased toward minority classes compared to the original output $\mathbf{p}$.

As in adversarial learning for adversarial example generation, a specific loss function with classifier input as variables needs to be designed. Here, the temperature scaling for the softmax operation of the CNN classifier and the prior frequency distribution $\{n_c\}_{c=1}^{C}$ over classes are employed to help design the loss function and subsequently perturb the classifier input. Suppose the softmax input (i.e., logit) vector is $\mathbf{z} = [z_1, z_2, \ldots, z_C]^\intercal$ for the classifier input $\mathbf{x}$. Temperature scaling modifies the softmax function by including the temperature scaling parameter $T \in \mathbb{R}^+$, i.e.,

$$\hat{p}_c = \frac{\exp(z_c/T)}{\sum_{k=1}^{C} \exp(z_k/T)} \ , \tag{1}$$

where $\hat{p}_c$ is the c-th element of the temperature-tuned classifier output $\hat{\mathbf{p}} = [\hat{p}_1, \hat{p}_2, \ldots, \hat{p}_C]^\intercal$. By setting a large temperature value (e.g., $T = 1000$), $\hat{p}_c$'s will become approximately equivalent to each other (e.g., $\hat{p}_c \approx 1/C$), but note that each $\hat{p}_c$ is a function of the classifier input $\mathbf{x}$ no matter which value T is set. With the almost-known output $\hat{\mathbf{p}}$ thanks to a large temperature value, the difference between $\hat{\mathbf{p}}$ and any specific target vector $\mathbf{p}^*$ would always exist if $\mathbf{p}^* \neq \hat{\mathbf{p}}$. With appropriate target $\mathbf{p}^*$, the difference between $\hat{\mathbf{p}}$ and $\mathbf{p}^*$ can be used to help perturb the classifier input in analogy to the well-known adversarial learning strategy. Considering that the objective of input perturbation is to bias the classifier output slightly toward minority classes, the target vector $\mathbf{p}^* = [p_1^*, p_2^*, \ldots, p_C^*]^\intercal$ is designed as

$$p_c^* = \frac{g(n_c)}{\sum_{k=1}^{C} g(n_k)} \ , \tag{2}$$

where $g(n_c)$ is a scalar function of the inverse frequency n_c. In this study, $g(n_c) = \log(M/n_c)$, with M being a relatively larger constant such that $g(n_c)$ is non-negative for all classes (M was set to the number of training data from the largest class in experiments). The logarithmic function was adopted here such that the discrete probability distribution $\mathbf{p}^*$ is smoother across classes, which in turn would help cause smaller bias toward the minority classes. It can be seen that p_c^* is relatively larger ($p_c^* > 1/C$) for minority classes and smaller ($p_c^* < 1/C$) for dominant classes. The difference between the temperature-tuned output $\hat{p}_c$ ($\approx 1/C$) and the target output p_c^* is limited to a relatively smaller range $(-1/C, 0)$ for dominant classes and a larger range $(0, 1-1/C)$ for minority classes. Therefore, the overall difference between $\hat{\mathbf{p}}$ and $\mathbf{p}^*$ is dominated by the minority classes. This indicates that perturbing the classifier input based on the overall difference between $\hat{\mathbf{p}}$ and $\mathbf{p}^*$ would change the pre-softmax logits more largely for minority classes (i.e., larger increasing in logits) than for dominant classes (i.e., smaller decreasing in logits). As a result, slightly drawing the classifier output $\hat{\mathbf{p}}$ closer to the target $\mathbf{p}^*$ by perturbing the classifier input would bias the classifier

prediction slightly toward minority classes compared to the original prediction. With an appropriate difference measure $\ell(\hat{\mathbf{p}}, \mathbf{p}^*)$ (e.g., cross entropy) which is essentially a function of the classifier input, the perturbed classifier input $\tilde{\mathbf{x}}$ can be obtained by the signed gradient of $\ell(\hat{\mathbf{p}}, \mathbf{p}^*)$ over input $\mathbf{x}$ [8], i.e.,

$$\tilde{\mathbf{x}} = \mathbf{x} - \varepsilon \cdot \mathtt{sign}(\nabla \ell(\hat{\mathbf{p}}, \mathbf{p}^*)) \,, \tag{3}$$

where ε is a scalar constant controlling the maximum perturbation on each data element (e.g., image pixel), $\nabla \ell(\hat{\mathbf{p}}, \mathbf{p}^*)$ is the gradient of difference measure (i.e., loss) function over the classifier input, and $\mathtt{sign}(\cdot)$ is the pixel-wise sign function. Once the perturbed input $\tilde{\mathbf{x}}$ is obtained, it can be fed to the classifier to get the final output $\tilde{\mathbf{p}}$, in which the class with the maximum output is considered as the final prediction for the original input $\mathbf{x}$.

Comparison with Relevant Studies: Our method can be considered as one type of post-hoc logit adjustment during inference [18]. In contrast to the post-hoc logit adjustment at the output side of the classifier model [18,22], our method adjusts the logit by perturbing the model input. In addition, since our method is applied during inference, it can be naturally combined with existing methods which focus on classifier training, and the combinations would often improve the classification performance compared to those original methods.

3 Experiments

3.1 Experimental Settings

The proposed method was extensively evaluated on three imbalanced medical image datasets, Skin7 [5], OCTMNIST [27], and X-ray6 (Table 1). Specially, X-ray6 contains six diseases of X-ray images (Atelectasis, Cardiomegaly, Emphysema, Hernia, Mass, Effusion), where the six classes were selected from the original 14-class dataset ChestX-ray14 [24] by removing those classes of images which may contain multiple or ambiguous diseases in single images. Although dataset scale varying a lot, all three datasets present clear data imbalance (Table 1, last column). For OCTMNIST, all the images were used for model training, and an additional set of images (250 per class) officially provided were used for testing. For Skin7 and X-ray6, a five-fold cross-validation scheme was adopted, with four folds for training and another fold for testing each time.

Table 1. Dataset statistics. Last column: imbalance ratio = image number in the largest classes divided by image number in the smallest class.

Dataset	ImageType	#Class	ImageSize	#SmallestClass	#LargestClass	Imbalance
Skin7	Dermoscopy	7	600 * 450	115	6705	58.3
OCTMNIST	OCT	4	28 * 28	7754	46026	5.9
X-ray6	X-ray	6	1024 * 1024	88	3368	38.3

Since our method was applied to a well-trained classifier model, an convolutional neural network (CNN) classifier needs to be trained in advance, either using certain re-balancing strategy or not. In experiments, three CNN backbones pre-trained on ImageNet were used, including ResNet50 [10], MobilenetV2 [20], and DenseNet169 [13]. All experiments are conducted on a single 2080Ti GPU. For model training, Skin7 images were resized to 300×300 and randomly cropped to 224×224 pixels, followed by a random horizontal flip, while X-ray6 images were resized to 224×224 pixels and OCTMNIST images were resized to 32×32 pixels followed by random horizontal flip. For testing, only similar resizing operation was performed for each test image. During model training, the stochastic gradient descent with momentum (0.9) and weight decay (0.0005) were adopted. The batch size was set 32 on Skin7 and X-ray6, and 128 on OCTMNIST. The learning rate was set 0.01 for MobilenetV2 and 0.001 for ResNet50 and DenseNet169, which was then decayed by 0.1 after every 50 epochs. Linear warm-up of learning rate was used in the first epoch. All models were trained for 200 epochs with clear convergence. During testing, unless otherwise mentioned, the difference measure $\ell(\hat{\mathbf{p}}, \mathbf{p}^*)$ was based on the cross-entropy loss. The temperature T was set 1000, and the constant ε was set 0.001 on Skin7 and OCTMNIST and 0.0001 on X-ray6, based on an extra small validation set for each dataset on the ResNet50 backbone. Because of the imbalance property in testing for Skin7 and X-ray6, the mean class recall over all classes (MCR) and the recall on the smallest class (SCR) were used for evaluation. The standard deviation of MCR and SCR over the five folds (with five-fold cross validation) were also reported when evaluated on the Skin7 and X-ray6 datasets. Note that the proposed method is only slightly slower than corresponding baseline during inference, e.g., with the average inference time 0.283 s per image by the proposed method versus 0.107 s by the corresponding baseline.

3.2 Effectiveness and Generalizability Evaluation

The effectiveness of our method was extensively evaluated by comparing with the widely used strategies to handle data imbalance, including the data re-sampling (RS) for class-balanced mini-batch, the class-level re-weighting (RW), the instance-level re-weighting with focal loss (FL) [16], and the recently proposed state-of-the-art methods, including the two-stage deferred re-sampling (DRS) [28] and the margin-based method LDAM with deferred re-weighting [2]. The model trained with conventional cross-entropy loss (CE) (i.e., without using any re-balancing strategy) was also used for comparison.

From Table 2, it can be observed that, although the performance varies across baselines on each of the three datasets, our method (built on the model trained by the baseline method) always performs better than the corresponding baseline when measured by MCR for all classes. Importantly, the performance boosting on the smallest class is much more significant than for all classes, as seen in the SCR columns. Detail performance on each Skin7 class from Fig. 2 also shows that our method obtains substantial improvement on small classes, although with certain

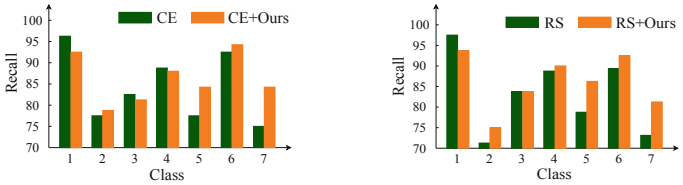

Fig. 2. Performance comparison with corresponding baseline (CE and RS) on each Skin7 class with ResNet50 backbone. X-axis: 1 for the largest class and decreasingly 7 for the smallest class.

decreased performance on dominant classes as often observed from state-of-the-art methods (e.g., LDAM [2]). These results clearly support that our method is effective in alleviating the model's bias toward dominant classes when the model was trained on imbalanced dataset with or without any re-balancing training strategy. Note that the larger variance of SCR than that of MCR on the X-ray6 and Skin7 datasets is probably due to the relatively smaller testing images on the smallest classes (only 22 and 23 images in each fold for the smallest class). Interestingly, some re-balancing baselines (e.g., RS, FL) performed worse than the plane cross-entropy baseline (CE). This may be due to the heavy imbalance in the datasets which cannot be well addressed with those re-balancing strategies. However, the inclusion of our method during inference consistently improves the performance of all the models trained with different strategies.

Table 2 also suggests that our method has desired generalizability. Our method consistently improves the performance particularly on minority classes for multiple classification tasks (Skin7, OCTMNIST, X-ray6), in combination with various re-balancing strategies (RW, DRS, LDAM, etc.), and with different classifier backbones (ResNet50, MobileNetV2, and DenseNet169). Because of its simple and independent operation on the inference phase, our method is expected to work well for more types of tasks and on various model architectures.

3.3 Robustness to Hyper-parameters

Our method is robust to the choice of perturbation magnitude ε. As shown in Fig. 3 (left and middle), when ε is smaller enough (e.g., in the range $(0, 0.001]$), our method performs consistently better than the corresponding baseline (with $\varepsilon = 0$ on the curve), no matter which baseline and CNN backbone is used. From this figure, we can also see that the best choice of ε varies when our method is combined with different baselines. This also indicates that the previously reported performance (Table 2) of our method on the Skin7 dataset is indeed conservative, where $\varepsilon = 0.001$ (not the best choice in most cases) was adopted in all comparisons. Actually, from Fig. 3 (middle), it can be expected that consistently better performance than reported in Table 2 would be obtained if setting ε smaller (e.g., 0.0005) when combining our method with most baselines on the MobileNetV2 backbone.

Table 2. Comparison with various baselines on multiple datasets with different CNN backbones. Standard deviation of MCR & SCR are in brackets for Skin7 and X-ray6.

Model	Method	Skin7		OCTMNIST		X-ray6	
		MCR	SCR	MCR	SCR	MCR	SCR
ResNet50	CE	84.54(0.86)	75.66(6.63)	75.60	26.40	58.77(2.63)	40.00(15.56)
	CE+ours	**86.42**(1.29)	**84.36**(5.00)	**77.20**	**43.60**	**59.81**(2.39)	**43.64**(14.59)
	RS	83.23(1.36)	73.04(9.95)	78.70	37.60	57.90(2.06)	34.56(13.11)
	RS+ours	**86.34**(1.11)	**81.76**(7.14)	**78.90**	**52.40**	**58.82**(1.98)	**37.28**(14.18)
	RW	85.03(0.97)	74.78(8.43)	76.00	28.40	62.17(0.76)	53.64 6.68
	RW+ours	**87.82**(1.10)	**87.83**(5.07)	**78.10**	**43.20**	**62.89**(1.43)	**56.36**(7.39)
	FL	83.10(0.94)	73.92(6.88)	74.80	23.20	57.69(2.42)	33.64(9.43)
	FL+ours	**85.90**(1.50)	**86.12**(4.75)	**77.60**	**38.80**	**58.23**(2.38)	**34.56**(9.43)
	DRS	84.12(1.46)	75.66(9.00)	79.20	39.20	54.85(2.20)	30.00(9.43)
	DRS+ours	**86.37**(0.81)	**84.36**(5.00)	**80.60**	**61.20**	**56.42**(2.54)	**32.72**(11.31)
	LDAM	83.48(1.47)	73.94(9.73)	79.60	40.00	59.44(2.69)	41.82(13.03)
	LDAM+ours	**84.81**(0.92)	**79.12**(9.92)	**81.60**	**54.40**	**60.04**(2.69)	**51.82**(10.44)
MobileNetV2	CE	84.46(2.05)	77.40(8.37)	76.40	25.20	56.44(1.20)	26.36(8.77)
	CE+ours	**85.56**(1.54)	**88.70**(3.87)	**77.60**	**45.20**	**57.22**(1.28)	**29.10**(11.43)
	RS	82.42(1.42)	66.98(11.36)	76.20	27.60	56.58(1.64)	30.92(5.92)
	RS+ours	**86.07**(0.78)	**84.36**(7.87)	**78.40**	**45.60**	**59.30**(2.20)	**41.82**(8.72)
	RW	85.75(0.91)	80.00(4.43)	77.70	32.40	60.70(1.82)	50.00(11.85)
	RW+ours	**85.96**(1.33)	**90.43**(3.25)	**78.20**	**52.40**	**61.74**(1.76)	**53.64**(10.52)
	FL	84.23(1.40)	76.54(10.05)	77.10	35.20	57.24(1.96)	35.46(9.90)
	FL+ours	**84.81**(0.99)	**89.58**(5.00)	**79.20**	**51.60**	**58.10**(2.32)	**38.18**(10.95)
	DRS	84.80(1.52)	78.26(8.72)	77.80	32.80	54.41(1.44)	25.46(10.49)
	DRS+ours	**85.73**(1.57)	**88.70**(3.87)	**78.40**	**56.00**	**56.07**(1.00)	**29.10**(8.25)
	LDAM	83.63(1.05)	76.54(10.01)	80.90	48.00	60.25(3.65)	17.26(16.84)
	LDAM+ours	**84.42**(0.61)	**82.64**(7.52)	**82.70**	**64.00**	**60.70**(3.59)	**36.36**(23.40)
DenseNet169	CE	84.92(1.10)	76.56(9.54)	73.90	21.60	60.74(2.18)	40.92(11.58)
	CE+ours	**86.20**(1.72)	**86.98**(6.86)	**76.10**	**36.00**	**61.44**(2.43)	**41.84**(11.76)
	RS	82.43(1.48)	69.56(9.75)	75.20	23.60	58.86(2.27)	36.38(14.35)
	RS+ours	**85.33**(1.67)	**79.14**(7.14)	**78.00**	**38.40**	**60.17**(1.86)	**40.90**(11.57)
	RW	84.50(0.68)	76.52(7.58)	75.50	20.40	63.68(1.88)	53.64(11.28)
	RW+ours	**86.48**(0.85)	**84.35**(8.95)	**77.40**	**38.00**	**64.44**(2.04)	**56.36**(10.98)
	FL	84.00(1.73)	80.00(3.87)	75.10	20.00	59.46(2.75)	40.00(12.21)
	FL+ours	**85.64**(1.23)	**87.84**(3.61)	**78.80**	**34.80**	**60.53**(2.14)	**43.64**(8.86)
	DRS	83.53(1.82)	71.32(10.00)	73.70	23.20	57.95(2.39)	35.46(12.61)
	DRS+ours	**85.81**(1.22)	**82.64**(7.55)	**75.00**	**40.40**	**59.26**(2.29)	**39.10**(13.08)
	LDAM	83.71(1.46)	72.18(12.54)	81.60	46.40	62.14(4.76)	44.54(17.83)
	LDAM+ours	**85.77**(1.00)	**82.62**(8.15)	**83.40**	**63.20**	**62.61**(4.54)	**46.36**(17.58)

The robustness of our method to the temperature scaling T is demonstrated in Fig. 3 (right). It shows that our method would work stably better than corresponding baselines when T is larger than 100. This also confirms the effectiveness of the temperature scaling at a larger value. In addition, besides the cross-entropy

loss for the difference measure $\ell(\hat{\mathbf{p}}, \mathbf{p}^*)$, some other choices including the mean squared error and focal loss were also tried for $\ell(\hat{\mathbf{p}}, \mathbf{p}^*)$, resulting in equivalent performance compared to that from the cross-entropy loss.

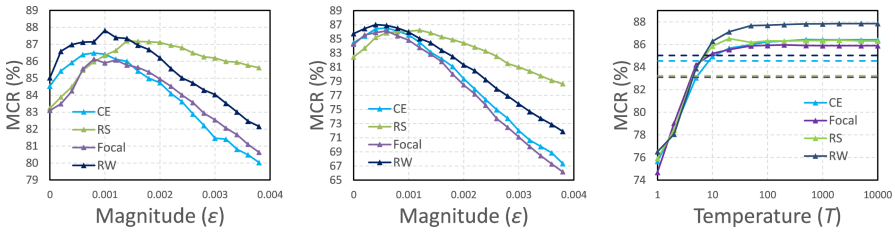

Fig. 3. Robustness to hyper-parameters. Left & middle: performance of our method with varying perturbation ε respectively on ResNet50 & MobileNetV2. Right: with varying temperature T on ResNet50 (dashed curves for corresponding baselines). Different curves for combinations of ours with different baselines. Skin7 was used here.

4 Conclusion

Here we propose a simple yet effective method to alleviate the data imbalance issue not during model training but during inference. The natural combination of our method with existing methods further alleviates the classifier's bias toward dominant classes, as supported by extensive evaluations on three medical datasets with different data-imbalance methods and model backbones. The applications of our method to more tasks like lesion detection will be explored.

Acknowledgments. This work is supported by the National Natural Science Foundation of China (No. 62071502, U1811461), the Guangdong Key Research and Development Program (No. 2020B1111190001, 2019B020228001), and the Meizhou Science and Technology Program (No. 2019A0102005).

References

1. Buda, M., Maki, A., Mazurowski, M.: A systematic study of the class imbalance problem in convolutional neural networks. Neural Netw. **106**, 249–259 (2018)
2. Cao, K., Wei, C., Gaidon, A., Arechiga, N., Ma, T.: Learning imbalanced datasets with label-distribution-aware margin loss. In: Advances in Neural Information Processing Systems, vol. 32 (2019)
3. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: synthetic minority over-sampling technique. J. Artif. Intell. Res. **16**, 321–357 (2002)
4. Chen, P., Sharma, Y., Zhang, H., Yi, J., Hsieh, C.J.: EAD: elastic-net attacks to deep neural networks via adversarial examples. In: AAAI (2018)

5. Codella, N.C.F., et al.: Skin lesion analysis toward melanoma detection: a challenge at the 2017 international symposium on biomedical imaging (ISBI), hosted by the international skin imaging collaboration (ISIC). In: IEEE International Symposium on Biomedical Imaging, pp. 168–172 (2018)
6. Esteva, A., et al.: Dermatologist-level classification of skin cancer with deep neural networks. Nature **542**, 115–118 (2017)
7. Esteva, A., et al.: A guide to deep learning in healthcare. Nat. Med. **25**, 24–29 (2019)
8. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572 (2015)
9. Han, H., Wang, W.Y., Mao, B.H.: Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning. In: International Conference on Intelligent Computing, pp. 878–887 (2005)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
11. Horn, G.V., Perona, P.: The devil is in the tails: fine-grained classification in the wild. arXiv preprint arXiv:1709.01450 (2017)
12. Huang, C., Li, Y., Loy, C.C., Tang, X.: Learning deep representation for imbalanced classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5375–5384 (2016)
13. Huang, G., Liu, Z., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2261–2269 (2017)
14. Kang, B., et al: Decoupling representation and classifier for long-tailed recognition. In: Proceedings of the International Conference on Learning Representations (2020)
15. Kubat, M., Matwin, S.: Addressing the curse of imbalanced training sets: one-sided selection. In: International Conference on Machine Learning, vol. 97, pp. 179–186 (1997)
16. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2980–2988 (2017)
17. Litjens, G., et al.: A survey on deep learning in medical image analysis. Med. Image Anal. **42**, 60–88 (2017)
18. Menon, A., Jayasumana, S., Rawat, A., Jain, H., Veit, A., Kumar, S.: Long-tail learning via logit adjustment. arXiv preprint arXiv:2007.07314 (2020)
19. Moosavi-Dezfooli, S.M., Fawzi, A., Frossard, P.: DeepFool: a simple and accurate method to fool deep neural networks. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2574–2582 (2016)
20. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: MobilenetV2: inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4510–4520 (2018)
21. Su, J., Vargas, D.V., Sakurai, K.: One pixel attack for fooling deep neural networks. IEEE Trans. Evol. Comput. **23**, 828–841 (2019)
22. Tang, K., Huang, J., Zhang, H.: Long-tailed classification by keeping the good and removing the bad momentum causal effect. Adv. Neural. Inf. Process. Syst. **33**, 1513–1524 (2020)
23. Wang, J., Perez, L.: The effectiveness of data augmentation in image classification using deep learning. Stanford University Research Report (2017)

24. Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., Summers, R.M.: ChestX-ray8: hospital-scale chest X-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3462–3471 (2017)
25. Wang, Y.X., Ramanan, D., Hebert, M.: Learning to model the tail. Adv. Neural. Inf. Process. Syst. **30**, 7032–7042 (2017)
26. Xiang, L., Ding, G., Han, J.: Learning from multiple experts: self-paced knowledge distillation for long-tailed classification. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12350, pp. 247–263. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58558-7_15
27. Yang, J., Shi, R., Ni, B.: MedMNIST classification decathlon: a lightweight AutoML benchmark for medical image analysis. arXiv preprint arXiv:2010.14925 (2020)
28. Zhou, B., Cui, Q., Wei, X.S., Chen, Z.: BBN: bilateral-branch network with cumulative learning for long-tailed visual recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 9716–9725 (2020)