A Simple Solution for MICCAI FLARE21 Challenges

Ti Bai

UT Southwestern Medical Center

Ti.Bai@UTSouthwestern.edu

Abstract.

Abdominal organ segmentation plays an important role in radiation therapy. Despite promising accuracy has been achieved with the aid of data driven deep learning techniques, dramatic performance degradation is observed when there exists distribution drift between the training and testing environments. To investigate some potential solutions and thereby alleviate this challenge, we implemented and compared the performances with different loss types: only Dice loss (DLoss), Dice + Cross Entropy losses (DCELoss), and Dice + Focal losses (DFLoss). Pseudo validation results (split from the original training dataset) shows that the DFLoss can lead to the best performance.

Keywords: Segmentation, Deep learning, Loss

1. Introduction

Accurate organ segmentation is critical in radiation therapy since it will affects the downstream dose prediction and calculation procedures. The inference speed is another important factor in clinical practice, for example, online adaptive radiotherapy (online ART).

Benefits from the success of the modern data driven deep learning (DL) techniques, a rich amount of work has been conducted to improve the segmentation accuracy. Most of them focus on the performance improvement given a pre- and well-defined training and testing datasets. However, in practice, the trained DL models may face poor generalizability issue.

In this work, we investigated the effects of different loss designs on the model generalizability. Specifically, we chose three different loss design: the popular Dice loss, the combination of Dice and cross entropy losses, and the combination of Dice and focal losses. We found that the combining the Dice and focal losses can lead to the best performance.

2. Method

We used three different loss designs to investigate their impacts on the model performance.

The loss designs include:

- Dice loss
- Dice + cross entropy losses
- Dice + focal losses

2.1. Preprocessing

- Enlarging: Given any loaded sample, we first generate a volumetric image with a size of 576x576x160 by either using cropping or padding strategies if necessary.
- Cropping strategy: we used the RandSpatialCrop operator implemented in MONAI to crop out a 512x512x128 sub-volume image for final model training.
- Resampling method for anisotropic data: None.
- Intensity normalization method:
  First, the dataset is firstly added by 1000 and then divided by 2000 for normalization purpose.

2.2. Proposed Method

- We used the popular 3D U-Net as the default network architecture. The input and output channel number are one and five, respectively. The feature numbers per each layer in the encoder part are 16, 32, 64, 128, 256. The strides are set to 2 for all the down sampling layers. We used two residual units when construct the convolutional layers. The readers are referred to the MONAI (https://docs.monai.io/en/latest/_modules/monai/networks/nets/unet.html#UNet) implementation for more specific details.
- Loss function: we use three different losses designs, see the Method section for details. All the hyper-parameters involved in the losses were default as in MONAI. (https://docs.monai.io/en/latest/losses.html)
- Number of model parameters: 4809646 (can be computed via such as torchsummary[1] library for Pytorch)
- Number of flops: 2105540608 (can be computed via such as fvcore[2] library for Pytorch)

2.3. Post-processing

None.

3. Dataset and Evaluation Metrics

3.1. Dataset

- A short description of the dataset used:
  The dataset used of FLARE2021 is adapted from MSD [4] (Liver [5], Spleen, Pancreas), NIH Pancreas [6, 7, 8], KiTS [9, 10], and Nanjing University under the license permission. For more detail information of the dataset, please refer to the challenge website and [11].
- Details of training / validation / testing splits:
  The total number of cases is 511. An approximate 70%/10%/20% train/validation/testing split is employed resulting in 361 training cases, 50 validation cases, and 100 testing cases. The detail information is presented in Table 1.

Table 1. Data splits of FLARE2021.

Data Split	Center	Phase	#Num.
Training (361 cases)	The National Institutes of Health Clinical Center	portal venous phase	80
	Memorial Sloan Kettering Cancer Center	portal venous phase	281
Validation (50 cases)	Memorial Sloan Kettering Cancer Center	portal venous phase	5
	University of Minnesota	late arterial phase	25
	7 Medical Centers	various phases	20
Testing (100 cases)	Memorial Sloan Kettering Cancer Center	portal venous phase	5
	University of Minnesota	late arterial phase	25
	7 Medical Centers	various phases	20
	Nanjing University	various phases	50

3.2. Evaluation Metrics

- Dice Similarity Coefficient (DSC)
- Normalized Surface Distance (NSD)
- Running time
- Maximum used GPU memory (when the inference is stable)

4. Implementation Details

[1] https://github.com/sksq96/pytorch-summary
[2] https://github.com/facebookresearch/fvcore

4.1. Environments and requirements

The environments and requirements of the baseline method is shown in Table 2.

Table 2. Environments and requirements.

Windows/Ubuntu version	CentOS 7
CPU	Intel(R) Xeon(R) CPU E5-2680 v3 @ 2.50GHz
RAM	64GB
GPU	Nvidia V100
CUDA version	10.2
Programming language	Python3.7
Deep learning framework	Pytorch (Torch 1.7.0, torchvision 0.8.0)
Specification of dependencies	SimpleITK

4.2. Training protocols

The training protocols of the baseline method is shown in Table 3.

Table 3. Training protocols.

Data augmentation methods	Random cropping
Initialization of the network	“he” normal initialization
Patch sampling strategy	Random cropping
Batch size	1
Patch size	512×512×128
Total epochs	555
Optimizer	Adam (beta1=0.9, beta2=0.999)
Initial learning rate	0.001
Learning rate decay schedule	Reduce by 10 times at iteration 100,000 and 150,000 iterations
Stopping criteria, and optimal model selection criteria	Stopping criterion is reaching the maximum number of iterations (200,000).
Training time	55.5 hours
CO$_2$eq[1]	

4.3. Testing protocols

- Pre-processing steps of the network inputs: The same strategy is applied as training steps.
- Post-processing steps of the network outputs: No post-processing step is used.
- Patch aggregation method: we used the sliding window method to do patch aggregation with equal weights to all the predictions.

5. Results

6. Discussion and Conclusion

Acknowledgment

The authors of this paper declare that the segmentation method they implemented for participation in the FLARE

challenge has not used any pre-trained models nor additional datasets other than those provided by the organizers.

References

[1] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation, in *International Conference on Medical image computing and computer-assisted intervention*, 2015, pp. 234–241. 1

[2] J. Ma, J. Chen, M. Ng, R. Huang, Y. Li, C. Li, X. Yang, and A. L. Martel, "Loss odyssey in medical image segmentation," *Medical Image Analysis*, vol. 71, p. 102035, 2021. 1

[3] A. L. Simpson, M. Antonelli, S. Bakas, M. Bilello, K. Farahani, B. Van Ginneken, A. Kopp-Schneider, B. A. Landman,G. Litjens, B. Menze *et al.*, "A large annotated medical image dataset for the development and evaluation of segmentation algorithms," *arXiv preprint arXiv:1902.09063*, 2019. 2

[4] P. Bilic, P. F. Christ, E. Vorontsov, G. Chlebus, H. Chen,Q. Dou, C.-W. Fu, X. Han, P.-A. Heng, J. Hesser *et al.*, "The liver tumor segmentation benchmark (lits)," *arXiv preprint arXiv:1901.04056*, 2019. 2

[5] H. Roth, A. Farag, E. Turkbey, L. Lu, J. Liu, and R. Summers, "Data from pancreas-ct. the cancer imaging archive (2016)." 2

[6] H. R. Roth, L. Lu, A. Farag, H.-C. Shin, J. Liu, E. B. Turkbey, and R. M. Summers, "Deeporgan: Multi-level deep convolutional networks for automated pancreas segmentation," in *International conference on medical image computing and computer-assisted intervention*. Springer, 2015, pp. 556–564. 2

[7] K. Clark, B. Vendt, K. Smith, J. Freymann, J. Kirby, P. Koppel, S. Moore, S. Phillips, D. Maffitt, M. Pringle *et al.*, "The cancer imaging archive (tcia): maintaining and operating a public information repository," *Journal of digital imaging*, vol. 26, no. 6, pp. 1045–1057, 2013. 2

[8] N. Heller, F. Isensee, K. H. Maier-Hein, X. Hou, C. Xie,F. Li, Y. Nan, G. Mu, Z. Lin, M. Han *et al.*, "The state of the art in kidney and kidney tumor segmentation in contrastenhanced ct imaging: Results of the kits19 challenge," *Medical Image Analysis*, vol. 67, p. 101821, 2021. 2

[9] N. Heller, S. McSweeney, M. T. Peterson, S. Peterson,J. Rickman, B. Stai, R. Tejpaul, M. Oestreich, P. Blake,J. Rosenberg et al., "An international challenge to use artificial intelligence to define the state-of-the-art in kidney and kidney tumor segmentation in ct imaging." American Society of Clinical Oncology, vol. 38, no. 6, pp. 626–626, 2020. 2

[10] J. Ma, Y. Zhang, S. Gu, C. Zhu, C. Ge, Y. Zhang, X. An,C. Wang, Q. Wang, X. Liu, S. Cao, Q. Zhang, S. Liu,Y. Wang, Y. Li, J. He, and X. Yang, "Abdomenct-1k: Is abdominal organ segmentation a solved problem?" IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021. 2, 3, 4