Generalised Dice Overlap as a Deep Learning Loss Function for Highly Unbalanced Segmentations

Carole H. Sudre[1,2(✉)], Wenqi Li[1], Tom Vercauteren[1], Sebastien Ourselin[1,2], and M. Jorge Cardoso[1,2]

[1] Translational Imaging Group, CMIC, University College London, London NW1 2HE, UK
[2] Dementia Research Centre, UCL Institute of Neurology, London WC1N 3BG, UK
carole.sudre.12@ucl.ac.uk

Abstract. Deep-learning has proved in recent years to be a powerful tool for image analysis and is now widely used to segment both 2D and 3D medical images. Deep-learning segmentation frameworks rely not only on the choice of network architecture but also on the choice of loss function. When the segmentation process targets rare observations, a severe class imbalance is likely to occur between candidate labels, thus resulting in sub-optimal performance. In order to mitigate this issue, strategies such as the weighted cross-entropy function, the sensitivity function or the Dice loss function, have been proposed. In this work, we investigate the behavior of these loss functions and their sensitivity to learning rate tuning in the presence of different rates of label imbalance across 2D and 3D segmentation tasks. We also propose to use the class re-balancing properties of the Generalized Dice overlap, a known metric for segmentation assessment, as a robust and accurate deep-learning loss function for unbalanced tasks.

1 Introduction

A common task in the analysis of medical images is the ability to detect, segment and characterize pathological regions that represent a very small fraction of the full image. This is the case for instance with brain tumors or white matter lesions in multiple sclerosis or aging populations. Such unbalanced problems are known to cause instability in well established, generative and discriminative, segmentation frameworks. Deep learning frameworks have been successfully applied to the segmentation of 2D biological data and more recently been extended to 3D problems [10]. Recent years have seen the design of multiple strategies to deal with class imbalance (e.g. specific organ, pathology...). Among these strategies, some focus their efforts in reducing the imbalance by the selection of the training samples being analyzed at the risk of reducing the variability in training [3,5], while others have derived more appropriate and robust loss functions [1,8,9]. In this work, we investigate the training behavior of three previously published loss functions in different multi-class segmentation problems in 2D and 3D while

M.J. Cardoso et al. (Eds.): DLMIA/ML-CDS 2017, LNCS 10553, pp. 240–248, 2017.
DOI: 10.1007/978-3-319-67558-9_28

assessing their robustness to learning rate and sample rates. We also propose to use the class re-balancing properties of the Generalized Dice overlap as a novel loss function for both balanced and unbalanced data.

2 Methods

2.1 Loss Functions for Unbalanced Data

The loss functions compared in this work have been selected due to their potential to tackle class imbalance. All loss functions have been analyzed under a binary classification (foreground vs. background) formulation as it represents the simplest setup that allows for the quantification of class imbalance. Note that formulating some of these loss functions as a 1-class problem would mitigate to some extent the imbalance problem, but the results would not generalize easily to more than one class. Let R be the reference foreground segmentation (gold standard) with voxel values r_n, and P the predicted probabilistic map for the foreground label over N image elements p_n, with the background class probability being $1 - P$.

Weighted Cross-Entropy (WCE): The weighted cross-entropy has been notably used in [9]. The two-class form of WCE can be expressed as

$$\mathrm{WCE} = -\frac{1}{N}\sum_{n=1}^{N} w r_n \log(p_n) + (1 - r_n)\log(1 - p_n),$$

where w is the weight attributed to the foreground class, here defined as $w = \frac{N - \sum_n p_n}{\sum_n p_n}$. The weighted cross-entropy can be trivially extended to more than two classes.

Dice Loss (DL): The Dice score coefficient (DSC) is a measure of overlap widely used to assess segmentation performance when a gold standard or ground truth is available. Proposed in Milletari et al. [8] as a loss function, the 2-class variant of the Dice loss, denoted DL_2, can be expressed as

$$\mathrm{DL}_2 = 1 - \frac{\sum_{n=1}^{N} p_n r_n + \epsilon}{\sum_{n=1}^{N} p_n + r_n + \epsilon} - \frac{\sum_{n=1}^{N} (1 - p_n)(1 - r_n) + \epsilon}{\sum_{n=1}^{N} 2 - p_n - r_n + \epsilon}$$

The ϵ term is used here to ensure the loss function stability by avoiding the numerical issue of dividing by 0, i.e. R and P empty.

Sensitivity - Specificity (SS): Sensitivity and specificity are two highly regarded characteristics when assessing segmentation results. The transformation of these assessments into a loss function has been described by Brosch et al. [1] as

$$\mathrm{SS} = \lambda \frac{\sum_{n=1}^{N} (r_n - p_n)^2 r_n}{\sum_{n=1}^{N} r_n + \epsilon} + (1 - \lambda) \frac{\sum_{n=1}^{N} (r_n - p_n)^2 (1 - r_n)}{\sum_{n=1}^{N} (1 - r_n) + \epsilon}.$$

The parameter λ, that weights the balance between sensitivity and specificity, was set to 0.05 as suggested in [1]. The ϵ term is again needed to deal with cases of division by 0 when one of the sets is empty.

Generalized Dice Loss (GDL): Crum et al. [2] proposed the Generalized Dice Score (GDS) as a way of evaluating multiple class segmentation with a single score but has not yet been used in the context of discriminative model training. We propose to use the GDL as a loss function for training deep convolutional neural networks. It takes the form:

$$\mathrm{GDL} = 1 - 2\frac{\sum_{l=1}^{2} w_l \sum_n r_{ln} p_{ln}}{\sum_{l=1}^{2} w_l \sum_n r_{ln} + p_{ln}},$$

where w_l is used to provide invariance to different label set properties. In the following, we adopt the notation GDL_v when $w_l = 1/(\sum_{n=1}^{N} r_{ln})^2$. As stated in [2], when choosing the GDL_v weighting, the contribution of each label is corrected by the inverse of its volume, thus reducing the well known correlation between region size and Dice score. In terms of training with stochastic gradient descent, in the two-class problem, the gradient with respect to p_i is:

$$\frac{\partial \mathrm{GDL}}{\partial p_i} = -2\frac{(w_1^2 - w_2^2)\left[\sum_{n=1}^{N} p_n r_n - r_i \sum_{n=1}^{N}(p_n + r_n)\right] + N w_2 (w_1 + w_2)(1 - 2r_i)}{\left[(w_1 - w_2)\sum_{n=1}^{N}(p_n + r_n) + 2N w_2\right]^2}$$

Note that this gradient can be trivially extended to more than two classes.

2.2 Deep Learning Framework

To extensively investigate the loss functions in different network architectures, four previously published networks were chosen as representative networks for segmentation due to their state-of-the art performance and were reimplemented using Tensorflow.

2D Networks: Two networks designed for 2D images were used to assess the behaviour of the loss functions: UNet [9], and the TwoPathCNN [3]. The UNet architecture presents a U-shaped pattern where a step down is a series of two convolutions followed by a downsampling layer and a step up consists in a series of two convolution followed by upsampling. Connections are made between the downsample and upsample path at each scale. TwoPathCNN [3], designed for tumor segmentation, is used here in a fully convolutional 2D setup under the common assumption that a 3D segmentation problem can be approximated by a 2D network in situations where the slice thickness is large. This network involves the parallel training of two networks - a local and a global subnetwork. The former consists of two convolutional layers with kernel of size 7^2 and 5^2 with max-out regularization interleaved with max-pooling layers of size 4^2 and 2^2 respectively; while the latter network consists of a convolution layer of kernel

size 13^2 followed by a max-pooling of size 2^2. The features of the local and global networks are then concatenated before a final fully connected layer resulting in the classification of the central location of the input image.

3D Networks: The DeepMedic architecture [4] and the HighResNet network [6] were used in the 3D context. DeepMedic consists in the parallel training of one network considering the image at full resolution and another on the down-sampled version of the image. The resulting features are concatenated before the application of two fully connected layers resulting in the final segmentation. HighResNet is a compact end-to-end network mapping an image volume to a voxel-wise segmentation with a successive set of convolutional blocks and residual connections. To incorporate image features at multiple scales, the convolutional kernels are dilated with a factor of two or four. The spatial resolution of the input volume is maintained throughout the network.

3 Experiments and Results

3.1 Experiments

The two segmentation tasks we choose to highlight the impact of the loss function target brain pathology: the first task tackles tumor segmentation, a task where tumor location is often unknown and size varies widely, and the second comprises the segmentation of age-related white matter hyperintensities, a task where the lesions can present a variety of shapes, location and size.

In order to assess each loss function training behavior, different sample and learning rates were tested for the two networks. The learning rates (LR) were chosen to be log-spaced and set to 10^{-3}, 10^{-4} and 10^{-5}. For each of the networks, three patch sizes (small: S, moderate: M, large: L), resulting in different effective field of views according to the design of the networks were used to train the models. A different batch size was used according to the patch size. Initial and effective patch sizes, batch size and resulting imbalance for each network are gathered in Table 1. In order to ensure a reasonable behavior of all loss functions, training patches were selected if they contained at least one foreground element. Larger patch sizes represent generally more unbalanced training sets. The networks were trained without training data augmentation to ensure more comparability between training behaviors. The imbalance in patches varied

Table 1. Comparison of patch sizes and sample rate for the four networks.

	UNet			TwoPathCNN			DeepMedic			HighResNet		
Batch size	5	3	1	5	3	1	5	3	1	5	3	1
Initial patch size	56	64	88	51	63	85	51	63	87	51	63	85
Effective patch size	16	24	48	19	31	53	3	15	39	15	27	49
Imbalance ratio	0.52	0.33	0.15	0.29	0.25	0.16	0.20	0.01	0.002	0.02	0.01	0.003

Table 2. Comparison of DSC over 200 last iterations in the 2D context for UNet and TwoPathCNN. Results are under the format median (interquartile range).

Patch	LR	UNet				TwoPathCNN			
		WCE	DL_2	SS	GDL_v	WCE	DL_2	SS	GDL_v
S	-5	0.71 (0.17)	0.73 (0.13)	0.37 (0.17)	0.75 (0.14)	0.56 (0.48)	0 (0)	0.53 (0.41)	0.49 (0.44)
	-4	0.77 (0.18)	0.76 (0.13)	0.74 (0.16)	0.80 (0.12)	0.80 (0.12)	0.79 (0.11)	0.81 (0.12)	0.80 (0.12)
	-3	0.70 (0.17)	0.72 (0.15)	0.39 (0.16)	0.72 (0.15)	0 (0)	0 (0)	0.77 (0.11)	0.72 (0.15)
M	-5	0.71 (0.23)	0.70 (0.22)	0.65 (0.25)	0.74 (0.19)	0 (0)	0.73 (0.18)	0.69 (0.21)	0.73 (0.19)
	-4	0.73 (0.18)	0.70 (0.22)	0.61 (0.25)	0.72 (0.19)	0.77 (0.16)	0.76 (0.17)	0.71 (0.18)	0.76 (0.17)
	-3	0.68 (0.23)	0.67 (0.21)	0.70 (0.26)	0.69 (0.22)	0 (0)	0.71 (0.22)	0.67 (0.21)	0.72 (0.19)
L	-5	0.63 (0.46)	0.62 (0.40)	0.49 (0.42)	0.56 (0.44)	0.62 (0.50)	0.50 (0.41)	0.50 (0.38)	0.56 (0.35)
	-4	0.68 (0.34)	0.64 (0.44)	0.18 (0.24)	0.66 (0.39)	0.64 (0.42)	0.59 (0.43)	0.52 (0.38)	0.64 (0.35)
	-3	0.59 (0.39)	0.57 (0.53)	0.16 (0.22)	0.59 (0.45)	0.77 (0.12)	0.77 (0.14)	0.79 (0.12)	0.79 (0.11)

greatly according to networks and contexts reaching at worst a median of 0.2% of a 3D patch.

The 2D networks were applied to BRATS [7], a neuro-oncological dataset where the segmentation task was here to localize the background (healthy tissue) and the foreground (pathological tissue, here the tumor) in the image. The 3D networks were applied to an in house dataset of 524 subjects presenting age-related white matter hyperintensities. In both cases, the T1-weighted, T2-weighted and FLAIR data was intensity normalized by z-scoring the data according to the WM intensity distribution. The training was arbitrarily stopped after 1000 (resp. 3000) iterations for the 2D (resp. 3D) experiments, as it was found sufficient to allow for convergence for all metrics.

3.2 2D Results

Table 2 presents the statistics for the last 200 steps of training in term of DSC for the four loss functions at the different learning rates, and different networks while

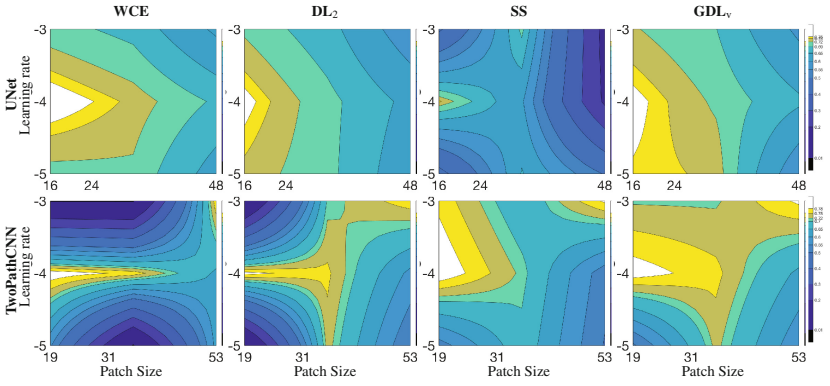

Fig. 1. Loss function behavior in terms of DSC (median over the last 200 iterations) under different conditions of effective patch size and learning rate in a 2D context. Isolines were linearly interpolated for visualization purposes.

Table 3. Comparison of DSC over 200 last iterations in the 3D context for DeepMedic and HighResNet. Results are under the format median (interquartile range).

Patch	LR	DeepMedic				HighResNet			
		WCE	DL_2	SS	GDL_v	WCE	DL_2	SS	GDL_v
S	-5	0.49 (0.17)	0.44 (0.19)	0.42 (0.14)	0.46 (0.17)	0 (0)	0 (0)	0.06 (0.15)	0.47 (0.32)
	-4	0.58 (0.20)	0.60 (0.15)	0.61 (0.22)	0.61 (0.18)	0 (0)	0.71 (0.18)	0.34 (0.20)	0.74 (0.15)
	-3	0.61 (0.12)	0.59 (0.14)	0.63 (0.15)	0.60 (0.15)	0 (0)	0 (0)	0 (0)	0 (0)
M	-5	0.05 (0.07)	0.05 (0.07)	0.05 (0.06)	0.04 (0.06)	0 (0)	0.60 (0.27)	0.15 (0.13)	0.64 (0.19)
	-4	0.09 (0.11)	0.07 (0.09)	0.08 (0.09)	0.08 (0.10)	0 (0)	0.71 (0.20)	0.20 (0.20)	0.69 (0.20)
	-3	0.45 (0.31)	0.42 (0.31)	0.17 (0.24)	0.48 (0.32)	0 (0)	0 (0)	0 (0)	0.65 (0.23)
L	-5	0.01 (0.03)	0.01 (0.03)	0.01 (0.03)	0.01 (0.03)	0 (0)	0.54 (0.27)	0.03 (0.06)	0.50 (0.32)
	-4	0.01 (0.04)	0.02 (0.04)	0.02 (0.04)	0.01 (0.04)	0 (0)	0.57 (0.32)	0.08 (0.19)	0.60 (0.30)
	-3	0.21 (0.33)	0.18 (0.30)	0.05 (0.12)	0.20 (0.33)	0 (0)	0.62 (0.18)	0.22 (0.15)	0.49 (0.34)

Fig. 1 shows the corresponding isolines in the space of learning rate and effective patch size illustrating notably the robustness of the GDL to the hyper-parameter space. The main observed difference across the different loss functions was the robustness to the learning rate, with the WCE and DL_2 being less able to cope with a fast learning rate (10^{-3}) when using TwoPathCNN while the efficiency of SS was more network dependent. An intermediate learning rate (10^{-4}) seemed to lead to the best training across all cases. Across sampling strategies, the pattern of performance was similar across loss functions, with a stronger performance when using a smaller patch but larger batch size.

3.3 3D Results

Similarly to the previous section, Table 3 presents the statistics across loss functions, sample size and learning rates for the last 200 iterations in the 3D experiment, while Fig. 2 plots the representation of robustness of loss function to the parameter space using isolines. Its strong dependence on the hyperparameters

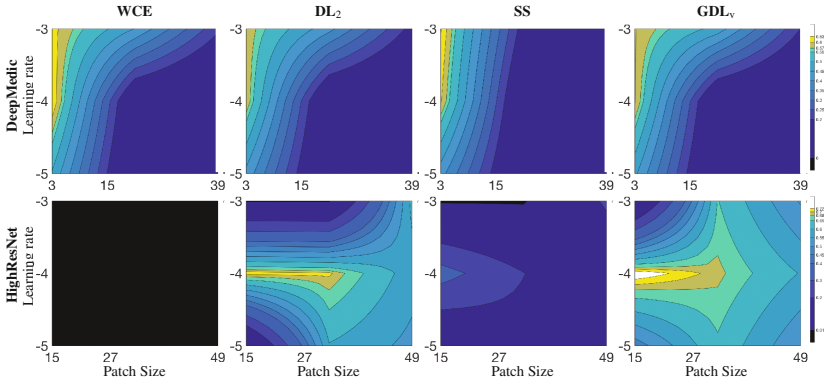

Fig. 2. Loss function behavior in terms of DSC (median over the last 200 iterations) under different conditions of effective patch size and learning rate in a 3D context. Isolines were linearly interpolated for visualization purposes.

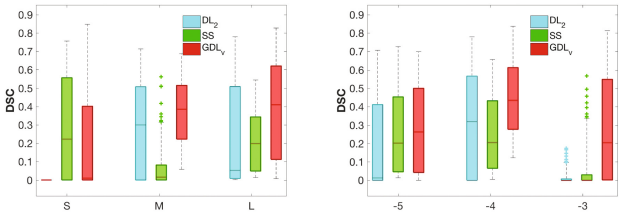

Fig. 3. Test set DSC for all loss functions across patch sizes (left) and across learning rates (right). WCE was omitted as it was unable to cope with the imbalance.

made DeepMedic agnostic to the choice of loss function. In the 3D context with higher data imbalance, WCE was unable to train and SS dropped significantly in performance when compared to GDL_v. DL_2 performed similarly to GLD_v for low learning rates, but failed to train for higher training rates. Similar patterns were observed across learning rates as for the 2D case, with the learning rate of 10^{-5} failing to provide a plateau in the loss function after 3000 iterations. We also observed that learning rates impacted network performance more for smaller patch sizes, but in adequate conditions (LR = 10^{-4}), smaller patches (and larger batch size) resulted in higher overall performance.

3D test set. For the 3D experiment, 10% of the available data was held out for testing purposes. The final HighResNet model was used to infer the test data segmentation. Figure 3 shows the comparison in DSC across loss functions for the different sampling strategies (right) and across learning rates (left). Overall, GDL_v was found to be more robust than the other loss functions across experiments, with small variations in relative performance for less unbalanced samples. Figure 4 presents an example of the segmentation obtained in the 3D experiment with HighResNet when using the largest patch size at a learning rate of 10^{-4}.

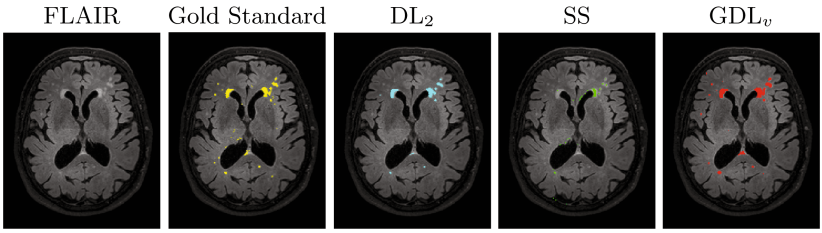

Fig. 4. The segmentation of a randomly selected 3D test set using different loss functions. Note the increased ability to capture punctuate lesions when using GDL_v. Loss functions were trained using a single patch of size 85^3 per step at learning rate 10^{-4}.

4 Discussion

From the observation of the training behavior of four loss functions across learning rates and sampling strategies in two different tasks/networks, it appears that a mild imbalance is well handled by most of the loss strategies designed for unbalanced datasets. However, when the level of imbalance increases, loss functions based on overlap measures appeared more robust. The strongest reliability across setups was observed when using GDL_v. Overall this work demonstrates how crucial the choice of loss function can be in a deep learning framework, especially when dealing with highly unbalanced problems. The foreground-background ratio in the most unbalanced case in this study was of 0.02% for the 3D experiment (white matter lesions). Future work will focus on more extreme imbalance situations, such as those observed in the case of the detection of lacunes and perivascular spaces (1/100000), where deep learning frameworks must find a balance between learning the intrinsic anatomical variability of all the classes and the tolerable level of class imbalance. The studied loss functions are implemented as part of the open source NiftyNet package (http://www.niftynet.io).

Acknowledgments. This work made use of Emerald, a GPU accelerated HPC, made available by the Science & Engineering South Consortium operated in partnership with the STFC Rutherford-Appleton Laboratory. This work was funded by the EPSRC (EP/H046410/1, EP/J020990/1, EP/K005278, EP/H046410/1), the MRC (MR/J01107X/1), the EU-FP7 project VPH-DARE@ IT (FP7-ICT-2011-9-601055), the Wellcome Trust (WT101957), the NIHR Biomedical Research Unit (Dementia) at UCL and the NIHR University College London Hospitals BRC (NIHR BRC UCLH/UCL High Impact Initiative- BW.mn.BRC10269).

References

1. Brosch, T., Yoo, Y., Tang, L.Y.W., Li, D.K.B., Traboulsee, A., Tam, R.: Deep convolutional encoder networks for multiple sclerosis lesion segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 3–11. Springer, Cham (2015). doi:10.1007/978-3-319-24574-4_1
2. Crum, W., Camara, O., Hill, D.: Generalized overlap measures for evaluation and validation in medical image analysis. IEEE TMI **25**(11), 1451–1461 (2006)
3. Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A., Bengio, Y., Pal, C., Jodoin, P.M., Larochelle, H.: Brain tumor segmentation with deep neural networks. MIA **35**, 18–31 (2017)
4. Kamnitsas, K., Ledig, C., Newcombe, V.F., Simpson, J.P., Kane, A.D., Menon, D.K., Rueckert, D., Glocker, B.: Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. MIA **36**, 61–78 (2017)
5. Lai, M.: Deep learning for medical image segmentation arXiv:1505.02000 (2015)
6. Li, W., Wang, G., Fidon, L., Ourselin, S., Cardoso, M.J., Vercauteren, T.: On the compactness, efficiency, and representation of 3D convolutional networks: brain parcellation as a pretext task. In: Niethammer, M., Styner, M., Aylward, S., Zhu, H., Oguz, I., Yap, P.-T., Shen, D. (eds.) IPMI 2017. LNCS, vol. 10265, pp. 348–360. Springer, Cham (2017). doi:10.1007/978-3-319-59050-9_28

7. Menze, B.H., et al.: The multimodal brain tumor image segmentation benchmark (BRATS). IEEE TMI **34**(10), 1993–2024 (2015)
8. Milletari, F., Navab, N., Ahmadi, S.A.: V-Net: fully convolutional neural networks for volumetric medical image segmentation. In: 2016 Fourth International Conference on 3D Vision (3DV), pp. 565–571. IEEE, October 2016
9. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). doi:10.1007/978-3-319-24574-4_28
10. Zheng, Y., Liu, D., Georgescu, B., Nguyen, H., Comaniciu, D.: 3D deep learning for efficient and robust landmark detection in volumetric data. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9349, pp. 565–572. Springer, Cham (2015). doi:10.1007/978-3-319-24553-9_69