nnU-Net for Brain Tumor Segmentation

Fabian Isensee[1], Paul F. Jäger[1], Peter M. Full[1], Philipp Vollmuth[2⋆], and Klaus H.Maier-Hein[1]

[1] Division of Medical Image Computing, German Cancer Research Center (DKFZ), Heidelberg, Germany
[2] Section for Computational Neuroimaging, Department of Neuroradiology, Heidelberg University Hospital, Heidelberg, Germany
f.isensee@dkfz-heidelberg.de

Abstract. We apply nnU-Net to the segmentation task of the BraTS 2020 challenge. The unmodified nnU-Net baseline configuration already achieves a respectable result. By incorporating BraTS-specific modifications regarding postprocessing, region-based training, a more aggressive data augmentation as well as several minor modifications to the nnU-Net pipeline we are able to improve its segmentation performance substantially. We furthermore re-implement the BraTS ranking scheme to determine which of our nnU-Net variants best fits the requirements imposed by it. Our method took the first place in the BraTS 2020 competition with Dice scores of 88.95, 85.06 and 82.03 and HD95 values of 8.498,17.337 and 17.805 for whole tumor, tumor core and enhancing tumor, respectively.

Keywords: nnU-Net · Brain Tumor Segmentation · U-Net · Medical Image Segmentation

1 Introduction

Brain tumor segmentation is considered one of the most difficult segmentation problems in the medical domain. At the same time, widespread availability of accurate tumor delineations could significantly improve the quality of care by supporting diagnosis, therapy planning and therapy response monitoring [1]. Furthermore, segmentation of tumors and associated subregions allows for identification of novel imaging biomarkers, which in turn enable more precise and reliable disease stratification [2] and treatment response prediction [3].

The Brain Tumor Segmentation Challenge (BraTS) [4,5] provides the largest fully annotated and publicly available database for model development and is the go-to competition for objective comparison of segmentation methods. The BraTS 2020 dataset [6,5,7,8] comprises 369 training and 125 validation cases. Reference annotations for the validation set are not made available to the participants. Instead, participants can use the online evaluation platform[3] to evaluate their

[⋆] né Kickingereder

[3] https://ipp.cbica.upenn.edu/

models and compare their results with other teams on the online leaderboard[4]. Besides the segmentation task, the BraTS 2020 competition features a survival prediction task as well as an uncertainty modelling task. In this work, we only participate in the segmentation task.

Recent successful entries in the BraTS challenge are exclusively based on deep neural networks, more specifically on encoder-decoder architectures with skip connections, a pattern which was first introduced by the U-Net [9]. Numerous architectural improvements upon the U-Net have been introduced, many of which are also used in the context of brain tumor segmentation, for example the addition of residual connections [10,11,12,13,14], densely connected layers [15,16,17,18,17] and attention mechanisms [19,18]. In the context of network architectures, the winning contributions of 2018 [11] and 2019 [12] should be highlighted as they extend the encoder-decoder pattern with a second decoder trained on an auxiliary task for regularization purposes. Training schemes are usually adapted to cope with the particular challenges imposed by the task of brain tumor segmentation. The stark class imbalance, for instance, necessitates appropriate loss functions for optimization: Dice loss [20,21] and focal loss [22] are popular choices [23,13,16,18,11,12,24]. Since BraTS evaluates segmentations using the partially overlapping whole tumor, tumor core and enhancing tumor regions [4], optimizing these regions instead of the three provided class labels (edema, necrosis and enhancing tumor) can be beneficial for performance [14,23,12,16]. Quantifying the uncertainty in the data has also been shown to improve results [18].

The methods presented above are highly specialized for brain tumor segmentation and their development required expertise as well as extensive experimentation. We recently proposed nnU-Net [25], a general purpose segmentation method that automatically configures segmentation pipelines for arbitrary biomedical datasets. nnU-Net set new state of the art results on the majority of the 23 datasets it was tested on, underlining the effectiveness of this approach. In the following, we will investigate nnU-Net's suitability for brain tumor segmentation. We use nnU-Net both as a baseline algorithm and as a framework for model development.

2 Method

2.1 Rankings should be used for model selection

Optimizing a model for a competition is often mistakenly treated equivalently to optimizing the model towards the segmentation metrics used in that competition. Segmentation metrics, however, only tell half the story: they describe the model on a per-image level whereas the actual ranking is based on a consolidation of these the metrics across all test cases. Ranking schemes can be differentiated in 'aggregate then rank' and 'rank then average' approaches. In the former, some aggregated metric (for example the average) is computed and then used to rank

[4] `https://www.cbica.upenn.edu/BraTS20/lboardValidation.html`

the participants. In the latter, the participants are ranked on each individual training case and then their ranks are accumulated across all cases. Different algorithm characteristics may be desired depending on the ranking scheme that is used to evaluate them. For example, in an 'aggregate then rank' scheme, median aggregation (as opposed to the mean) would be more forgiving to algorithms that produce badly predicted outliers. We refer to [26] for a comprehensive analysis on the impact of ranking on challenge results.

BraTS uses a 'rank then aggregate' approach, most likely because it is well suited to combine different types of segmentation metrics (such as HD95 and Dice). Effectively, each submission obtains six ranks per test case (one for each of the 3 evaluated regions times the 2 segmentation metrics) and the ranks are then averaged across all cases and metrics (see [5]). The final rank is normalized by the number of participating algorithms to form the ranking score, which ranges from 0 (best) to 1 (worst).

The BraTS evaluation of cases with empty reference segmentations for enhancing tumor is tailored for the ranking scheme used by the competition: if a participant predicts false positive voxels in these cases, BraTS assigns the worst possible value for both metrics (Dice 0, HD95 373.13) whereas a correctly returned empty segmentation yields perfect values (Dice 1, HD95 0). Thus, the enhancing tumor label essentially becomes binary for all participants: they either achieve the (shared) first or (shared) last rank.

As a consequence, optimizing models to maximize the mean-aggregated Dice scores and HD95 values returned by the BraTS evaluation platform may not be an ideal surrogate for optimal performance on the BraTS competition. We therefore reimplemented the ranking used by the BraTS competition and used it to rank our models against each other in order to select the best performing variant(s).

2.2 nnU-Net baseline

We base our method on nnU-Net, our recently proposed fully automatic framework for the configuration of segmentation methods. We refer to [25] for a detailed description of nnU-Net.

First, we apply nnU-Net without any modifications to provide a baseline for later modifications. The design choices made by nnU-Net are described in the following.

nnU-Net normalizes the brain voxels of each image by subtracting their mean and dividing by their standard deviation. The non-brain voxels remain at 0. The network architecture generated by nnU-Net is displayed in Figure 1. It follows a 3D U-Net[27,9] like pattern and consists of an encoder and a decoder which are interconnected by skip connections. nnU-net does not use any of the recently proposed architectural variations and only relies on plain convolutions for feature extraction. Downsampling is performed with strided convolutions and upsampling is performed with convolution transposed. Auxiliary segmentation

[5] `https://zenodo.org/record/3718904`

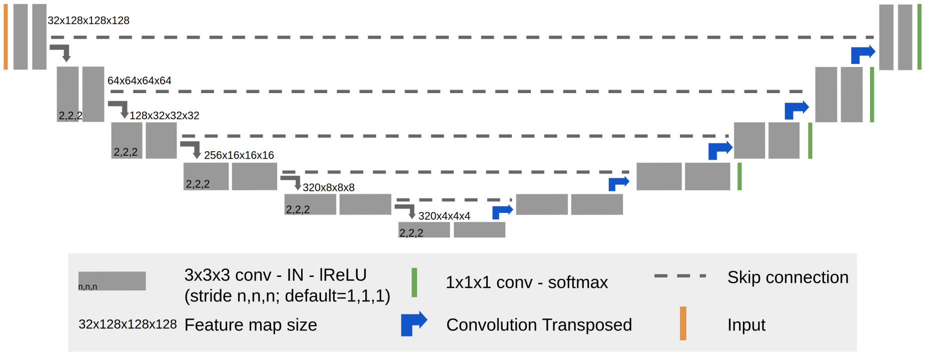


Fig. 1. Network Architecture as generated by nnU-Net. nnU-Net uses only plain U-Net-like architectures. For BraTS 2020, an input patch size of 128 × 128 × 128 was selected. Downsampling is done with strided convolutions, upsampling is implemented as convolution transposed. Feature map sizes are displayed in the encoder part of the architecture. The feature maps in the decoder mirror the encoder. Auxiliary segmentation outputs used for deep supervision branch off at all but the two lowest resolutions in the decoder.

outputs, which are used for deep supervision, branch off at all but the two lowest resolutions in the decoder. The input patch size is selected to be $128 \times 128 \times 128$ with a batch size of 2. A total of five downsampling operations are performed, resulting in a feature map size of $4 \times 4 \times 4$ in the bottleneck. Initial number of convolutional kernels is set to 32, which is doubled with each downsampling up to a maximum of 320. The number of kernels in the decoder mirrors the encoder. Leaky ReLUs [28] are used as nonlinearities. Instance normalization [29] is used for feature map normalization.

Training objective is the sum of Dice [21,20] and cross-entropy loss. The loss operates on the three class labels edema, necrosis and enhancing tumor. nnU-Net uses stochastic gradient descent with an initial learning rate of 0.01 and a Nesterov momentum of 0.99. Training runs for a total of 1000 epochs, where one epoch is defined as 250 iterations. The learning rate is decayed with a polynomial schedule as described in [30]. Training patches are cropped from randomly selected training cases. Data augmentation is applied on the fly during training (see also Supplementary Information D in [25]).

2.3 BraTS-specific optimizations

Besides its role as a high quality standardized baseline and out-of-the-box segmentation tool, we also advertise nnU-Net as a framework for method development. To underline this aspect of nnU-Net's broad functionality, we select promising BraTS-specific modifications and integrate them into nnU-Net's configuration.

Region-based training The provided labels for training are 'edema', 'non-enhancing tumor and necrosis' and 'enhancing tumor'. The evaluation of the segmentations is, however, performed on three partially overlapping *regions*: whole tumor (consisting of all 3 classes), tumor core (non-enh. & necrosis + enh. tumor) and enhancing tumor. It has been shown previously [14,23,12,16,18,11] that directly optimizing these regions instead of the individual classes can improve performance on the BraTS challenge. To this end, we replace the softmax nonlinearity in our network architecture with a sigmoid and change the optimization target to the three tumor subregions. We also replace the crossentropy loss term with a binary cross-entropy that optimizes each of the regions independently.

Postprocessing When the reference segmentation for enhancing tumor is empty, BraTS evaluation awards zero false positive predictions with a Dice score of 1 (the Dice would otherwise be undefined due to division by 0), thus placing the corresponding algorithm in the (shared) first rank for this test case, region and metric. This can be exploited with the goal of improving the average rank of the submitted method and thus its overall ranking in the challenge. By removing enhancing tumor entirely if the predicted volume is less than some threshold, one can accumulate more perfect rankings at the expense of some additional cases with a Dice score of 0 (and corresponding worst rank). Even though this strategy has the side effect of removing some true positive predictions, the net gain can out-weigh the losses. Removed enhancing tumor is replaced with necrosis to ensure that these voxels are still considered part of the tumor core. We optimize the threshold for postprocessing on our training set cross-validation twice, once via maximizing the mean Dice score and once via minimizing the ranking score in our internal BraTS-like ranking. Whenever we present postprocessed results, we select the best value even if that value was achieved by the opposing selection strategy.

2.4 Further nnU-Net Modifications

Increased batch size Over the past years, the BraTS dataset has continued to grow in size. The low batch size used by nnU-Net results in noisier gradients, which potentially reduces overfitting but also constrains how accurately the model can fit the training data. With a larger dataset, it may be beneficial to increase the batch size (bias variance trade-off). To this end, we modify nnU-Net's configuration to increase the batch size from 2 to 5 in an attempt to improve the model accuracy.

More data augmentation Data augmentation can effectively be used to artificially enlarge the training set. While nnU-Net already uses a broad range of aggressive data augmentation techniques, we make use of even more aggressive augmentations in an attempt to increase the robustness of our models. All augmentations are applied on the fly during training using the *batchgenerators* framework[6]. Relative to the nnU-Net baseline, we make the following changes:

- increase the probability of applying rotation and scaling from 0.2 to 0.3.
- increase the scale range from (0.85, 1.25) to (0.65, 1.6)
- select a scaling factor for each axis individually
- use elastic deformation with a probability of 0.3
- use additive brightness augmentation with a probability of 0.3
- increase the aggressiveness of the Gamma augmentation

Batch normalization In our participation in the M&Ms challenge [7] we noticed that more aggressive data augmentation could be used to effectively close the domain gap to other scanners, but only when used in conjunction with batch normalization (instead of instance normalization) (results available here [8], our paper is not yet available). In BraTS, Dice scores for the test cases are often lower than the reported values on the training and validation dataset, which makes us believe that there may be a domain gap between the test set and the training and validation sets. This suggests that pursuing a this strategy for BraTS as well may be beneficial.

Batch dice The standard implementation of the Dice loss computes the loss for every sample in the minibatch independently and then averages the loss over the batch (we call this the *sample Dice*). Small errors in samples with few annotated voxels can cause large gradients and dominate the parameter updates during training. If these errors are caused by imperfect predictions of the model, these large gradients are desired to push the model towards better predictions. However, if the models predictions are accurate and the reference segmentation is imperfect, these large gradients will be counterproductive during training. We therefore implement a different Dice loss computation: instead of treating the samples in a minibatch independently, we compute the dice loss over all samples in the batch (pretending they are just a single large sample, we call this the *batch Dice*). This effectively regularizes the Dice loss because samples with previously only few annotated voxels are now overshadowed by other samples in the same batch.

Abbreviations We will represent our models using the following abbreviations:

- **BL/BL***: baseline nnU-Net without modifications. * indicates a batch size of 5 instead of 2.
- **R**: Region-based training (see Section 2.3)

[6] https://github.com/MIC-DKFZ/batchgenerators

[7] https://www.ub.edu/mnms/

[8] https://www.ub.edu/mnms/results.html

- **DA/DA***: more aggressive data augmentation as described in Section 2.4. * indicates that the brightness augmentation is only applied with a probability of 0.5 for each input modality.
- **BD**: model trained with batch Dice (as opp posed to sample Dice, see Section 2.4)

We denote our models by the modifications that were applied to it, for example BL*+R+BD. Note that all models shown will have postprocessing applied to them.

3 Results

3.1 Aggregated Scores

Table 1. Training set results (n=369). All experiments were run as 5-fold cross-validation. See Section 2.4 for decoding the abbreviations

Model	Training Set (Dice, 5-fold CV)			
	Whole	Core	Enh.	Mean
BL	91.6	87.23	80.83	86.55
BL*	91.85	86.24	80.18	86.09
BL*+R	91.75	87.24	82.21	86.73
BL*+R+DA	91.87	87.97	81.37	87.07
BL*+R+DA+BN	91.57	87.59	81.29	86.82
BL*+R+DA+BD	91.76	87.67	80.94	86.79
BL*+R+DA+BN+BD	91.7	87.21	81.7	86.87
BL*+R+DA*+BN	91.6	87.51	80.94	86.68
BL*+R+DA*+BN+BD	91.47	87.13	81.33	86.64

We train each configuration as a five-fold cross-validation on the training cases (no external data is used). This not only provides us with a performance estimate on the 369 training cases but also enables us to select the threshold for postprocessing on the training set rather than the validation set. The results for the different configurations are presented in Table 1.

We use the five models obtained from the cross-validation on the training cases as an ensemble to predict the validation set. The aggregated Dice scores and HD95 values as computed by the online evaluation platform are reported in Table 2. Again we provide averages of the HD95 values and Dice scores across the three evaluated regions for clarity.

As discussed in Section 2.1, these aggregated metric should not be used to run model selection for the BraTS because they may not directly optimize the ranking that will be used to evaluate the submissions. This becomes evident when looking at the HD95 values of the enhancing tumor HD95. The HD95 averages are artificially inflated because false positive predictions in cases where

Table 2. Validation set results. Predictions were made using the 5 models from the training cross-validation as an ensemble. Metrics computed by the validation platform. See Section 2.4 for decoding the abbreviations

Model	Dice				HD95			
	Whole	Core	Enh.	Mean	Whole	Core	Enh.	Mean
BL	90.6	84.26	77.67	84.18	4.89	5.91	35.10	15.30
BL*	90.93	83.7	76.64	83.76	4.23	6.01	41.06	17.10
BL*+R	90.96	83.76	77.65	84.13	4.41	8.80	29.82	14.34
BL*+R+DA	90.9	84.61	78.67	84.73	4.70	5.62	29.50	13.28
BL*+R+DA+BN	91.24	85.04	79.32	85.2	3.97	5.17	29.25	12.80
BL*+R+DA+BD	90.97	83.91	77.48	84.12	4.11	8.60	38.06	16.93
BL*+R+DA+BN+BD	91.15	84.19	79.99	85.11	3.72	7.97	26.28	12.66
BL*+R+DA*+BN	91.18	85.71	79.85	85.58	3.73	5.64	26.41	11.93
BL*+R+DA*+BN+BD	91.19	85.24	79.45	85.29	3.79	7.77	29.23	13.60

no reference segmentation for enhancing tumor is present are hard coded to receive a HD95 value of 373.13. Given that the vast majority of HD95 for this class is much smaller (even the 90th percentile is just 11.32 for our final validation set submission), these outlier values dominate the mean aggregation and result in mostly uninformative mean HD95: Switching the metric a single case from 373.13 to 0 (due to the removal of false positive predictions) can have a huge effect on the mean. The same is true for the Dice score, but to a lesser extend: a HD95 of 373.13 is basically impossible to occur even if the predicted segmentation is of poor quality whereas a low Dice score is likely to happen.

3.2 Internal BraTS-like ranking

Table 3. Rankings of our proposed nnU-Net variants on the training and validation set. For each set, we rank twice: once based on the mean Dice and once using our reimplementation of the BraTS ranking.

Model	Training set				Validation set			
	rank based on mean Dice		BraTS ranking		rank based on mean Dice		BraTS ranking	
	value	rank	value	rank	value	rank	value	rank
BL	86.55	8	0.3763	8	84.18	6	0.4079	8
BL*	86.09	9	0.3767	9	83.76	9	0.4236	9
BL*+R	86.73	5	0.3393	5	84.13	7	0.4005	7
BL*+R+DA	87.07	**1**	0.3243	3	84.73	5	0.3647	5
BL*+R+DA+BN	86.82	3	0.3377	4	85.20	3	0.3577	4
BL*+R+DA+BD	86.79	4	0.3231	2	84.12	8	0.3726	6
BL*+R+DA+BN+BD	86.87	2	0.3226	**1**	85.11	4	0.3487	3*
BL*+R+DA*+BN	86.68	6	0.3521	6	85.58	**1**	0.3125	**1***
BL*+R+DA*+BN+BD	86.64	7	0.3595	7	85.29	2	0.3437	2*

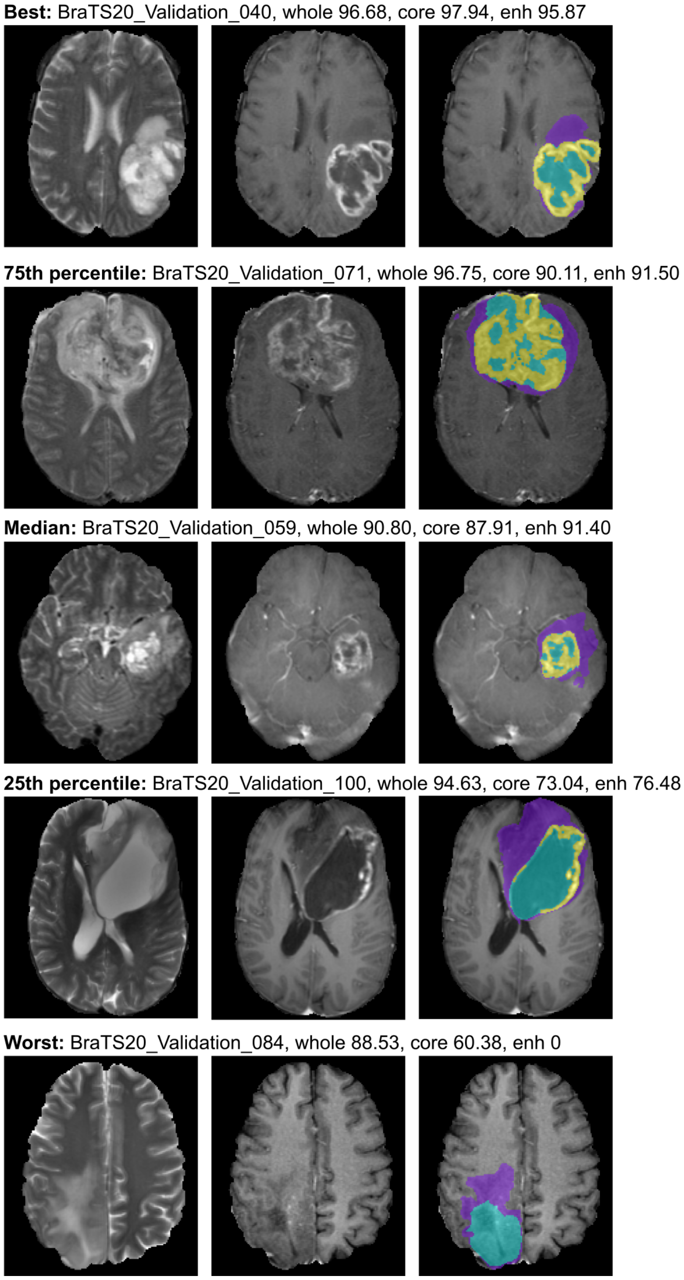
To overcome the shortcomings of metric aggregation-based model selection we create an internal leaderboard where we rank our nnU-Net variants presented above (as well as others that we omitted for brevity) against each other using a reimplementation of the BraTS ranking. Results of both ranking schemes are summarized in Table 3. While the ranks between the mean Dice and the BraTS ranking scheme are different, both still seem to follow similar trends. On the training set, for example, the BL and BL* models constitute the worst contenders while BL*+R+DA+BN+BD performs quite well. Within the validation set the correlation between the ranking schemes es even more pronounced. When comparing the model performance between the training and validation sets (any of the ranking schemes) the rankings differ drastically: the DA* models perform exceptionally well on the validation set (rank 1 and 2) whereas they underperform on the training set (rank 6 and 7). This creates a difficult situation. How do we choose the models used for the test set submission? Do we trust more in the 125 validation cases because we do not have access to the reference labels and thus can not overfit as easily or do we favor the training set because is has almost 3 times as many cases and should therefore provide more reliable performance estimates?

We opted for trusting the validation set over the training set and thus selected the three top performing models to build our final ensemble: $\mathrm{BL}^* + \mathrm{R} + \mathrm{DA}^* + \mathrm{BN} + \mathrm{BD}$, BL*+R+DA*+BN and BL*+R+DA+BN+BD. Note that additionally to the 5 models from the cross-validation we had 10 more models of the BL*+R+DA*+BN configuration (each trained on a random 80:20 split of the training cases). Thus, our final ensemble consisted of 5 + 5 + 15 = 25 models. Note that ensembling was implemented by first predicting the test cases individually with each configuration, followed by averaging the sigmoid outputs to obtain the final prediction. Therefore, the 15 models of the BL*+R+DA*+BN configuration had the same influence on the final prediction as the other configurations.

Our final ensemble achieved mean Dice scores of 91.24, 85.06 and 79.89 and HD95 of 3.69, 7.82 and 23.50 for whole tumor, tumor core and enhancing tumor, respectively (on the validation set). With a mean Dice score of 85.4 the ensemble appears to perform worse in terms of the 'aggregate then rank' approach. However, since it sits comfortably on the first place of our internal 'rank then aggregate' ranking with a score of 0.2963 we are confident in its segmentation performance.

3.3 Qualitative results

Figure 2 provides a qualitative overview of the segmentation performance. It shows results generated by our ensemble on the validation set. To rule out cherry picking we standardized the selection of presented validation cases: They were selected as best, worst, median and 75th and 25th percentiles based on their Dice scores (averaged over the three validation regions). As can be seen in the figure, the segmentation quality is high overall. The low tumor core score for the *worst* example hints at one of the potential issues with the definition of this class in



Best: BraTS20_Validation_040, whole 96.68, core 97.94, enh 95.87

75th percentile: BraTS20_Validation_071, whole 96.75, core 90.11, enh 91.50

Median: BraTS20_Validation_059, whole 90.80, core 87.91, enh 91.40

25th percentile: BraTS20_Validation_100, whole 94.63, core 73.04, enh 76.48

Worst: BraTS20_Validation_084, whole 88.53, core 60.38, enh 0

Fig. 2. Qualitative validation set results. Cases were selected as best, worst, median and 75th and 25th percentile. Within each row, the raw T2 image is shown to the left, the T1c image in the middle and on overlay with the generated segmentation on the T1c image is shown on the right. Edema is shown in violet, enhancing tumor in yellow and necrosis/non-enhancing tumor on turquoise.

the reference segmentations (see Discussion). The enhancing tumor score of 0 in the absence of predicted enhancing tumor voxels indicates either that our model missed a small enhancing tumor lesion or that it was removed as a result of our postprocessing. An inspection of the non-postprocessed segmentation mask reveals that the enhancing tumor lesion was indeed segmented by the model and must have been removed during postprocessing.

3.4 Test set results

Table 4. Quantitative test set results. Values were provided by the challenge organizers.

	Dice			HD95		
	Enh	Whole	Core	Enh	Whole	Core
Mean	82.03	88.95	85.06	17.805	8.498	17.337
StdDev	19.71	13.23	24.02	74.886	40.650	69.513
Median	86.27	92.73	92.98	1.414	2.639	2.000
25th percentile	79.30	87.76	88.33	1.000	1.414	1.104
75th percentile	92.25	95.17	96.19	2.236	4.663	3.606

Table 4 provides quantitative test set results. Our submission took the first place in the BraTS 2020 competition (see `https://www.med.upenn.edu/cbica/brats2020/rankings.html`).

4 Discussion

This manuscript describes our participation in the segmentation task of the BraTS 2020 challenge. We use nnU-Net [25] not only as our baseline algorithm but also as a framework for method development. nnU-Net again proved its generalizability by providing high segmentation accuracy in an out-of-the-box fashion. On the training set cross-validation, the performance of the nnU-Net baseline is very close to the configurations that were specifically modified for the BraTS challenge. On the validation set, however, the proposed BraTS-specific modifications provide substantially higher segmentation performance. Our final ensemble was selected based on the three best models on the validation set (as determined with our reimplementation of the BraTS ranking). It obtained validation Dice scores of 91.24, 85.06 and 79.89 as well as HD95 of of 3.69, 7.82 and 23.5 for whole tumor, tumor core and enhancing tumor, respectively.

Our approach won the BraTS 2020 competition. On the test set, our model obtained Dice scores of 88.95, 85.06 and 82.03 and HD95 values of 8.498, 17.337 and 17.805 for whole tumor, tumor core and enhancing tumor, respectively.

We should note that this manuscript only spans a small number of modifications and lacks sufficiently extensive experimental validation thereof. While most of our nnU-Net configurations achieve good performance, our results do not allow us to accurately pinpoint which aspects contributed the most. Some design choices may even have reduced our overall performance. Increasing the batch size from 2 to 5 gave worse results according to both evaluated ranking schemes and on both the training and validation set (see Table 3). This is a clear signal that the batch size should have been reverted to 2 for the remaining experiments. We also believe that a more thorough optimization of hyperparameters regarding both the training scheme as well as the data augmentation could result in further performance gains.

Based on our observations in Figure 2, one major source of error appears to be the tumor core prediction. In particular the *worst* presented example has a rather low Dice score for this region, even though the result seems visually plausible. We believe that this failure mode is not necessarily an issue inherent to our model but potentially originates from an inconsistency in the definition of the *non-enhancing tumor and necrosis* label, particularly in LGG cases. While the necrosis part of this label is easy to recognize, the non-enhancing tumor region often has little evidence in the image and the associated annotations may be subjective.

The enhancing tumor class is arguably the most difficult to segment in this dataset. What makes this class particularly challenging is the way its evaluation is handled when the reference segmentation of an image does not contain this class. The BraTS evaluation scheme favors the removal of small enhancing tumor lesions and thus encourages such postprocessing. In a clinical scenario where the accurate detection of small enhancing tumors could be critical, this property is not necessarily desired and we recommend to omit the postprocessing presented in this manuscript.

Just like many other BraTS participants, we have used the aggregated Dice scores and HD95 values for model selection in previous BraTS challenges [13,23]. As discussed in this manuscript, this strategy may not be ideal because the ranking that is used to determine the winning contributions works on a different principle. Indeed, when comparing the rankings of our models relative to each other on the training and validation set, we observed disparities between 'aggregate then rank' and 'rank then aggregate'. However, the overall trends were similar between the ranking methods indicating that a model selection based on metric aggregation is still going into the right direction.

While experimenting with the 'rank then aggregate' ranking on our model configurations we noted some instability in the ranks obtained by out models: when models are added or removed from the pool of methods, the ranking of other methods relative to each other could change. This seems to be an inherent property of this ranking [26]. Even though this might appear alarming at first glance, we attribute most of the instability to the similarity of the methods within the pool which has certainly overemphasized this potential issue. We expect the ranking to be more stable the more diverse the evaluated methods are (especially on the test set where all methods are compared against each other).

Given that aggregated metric are at least in part disconnected from the actual evaluation of the challenge, it is frustrating that the online leaderboards for training and validation sets only display mean values for the competing methods and therefore do not allow for a perfectly accurate comparison with other participants. Especially in cases where the enhancing tumor class is absent from the reference segmentation, the dichotomy of the returned metric values (0 or 1 for the Dice score and 0 or 373.13 for the HD95) can overshadow the more minute (but still very meaningful) differences between the teams on the validation cases that do actually contain this label. We would love to see the 'rank then aggregate' ranking scheme implemented in the online leaderboards as well so that they become more informative about a models performance on the BraTS challenge.

The source code for our submission is publicly available as part of the nnU-Net repository (`https://github.com/MIC-DKFZ/nnUNet`). A docker image for reproducing our test set predictions is available on Docker Hub[9].

References

1. P. Kickingereder, F. Isensee, I. Tursunova, J. Petersen, U. Neuberger, D. Bonekamp, G. Brugnara, M. Schell, T. Kessler, M. Foltyn *et al.*, "Automated quantitative tumour response assessment of mri in neuro-oncology with artificial neural networks: a multicentre, retrospective study," *The Lancet Oncology*, vol. 20, no. 5, pp. 728–740, 2019.
2. P. Kickingereder, U. Neuberger, D. Bonekamp, P. L. Piechotta, M. Götz, A. Wick, M. Sill, A. Kratz, R. T. Shinohara, D. T. Jones *et al.*, "Radiomic subtyping improves disease stratification beyond key molecular, clinical, and standard imaging characteristics in patients with glioblastoma," *Neuro-oncology*, vol. 20, no. 6, pp. 848–857, 2018.
3. P. Kickingereder, M. Götz, J. Muschelli, A. Wick, U. Neuberger, R. T. Shinohara, M. Sill, M. Nowosielski, H.-P. Schlemmer, A. Radbruch *et al.*, "Large-scale radiomic profiling of recurrent glioblastoma identifies an imaging predictor for stratifying anti-angiogenic treatment response," *Clinical Cancer Research*, vol. 22, no. 23, pp. 5765–5771, 2016.
4. B. H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, Y. Burren, N. Porz, J. Slotboom, R. Wiest *et al.*, "The multimodal brain tumor image segmentation benchmark (brats)," *IEEE transactions on medical imaging*, vol. 34, no. 10, pp. 1993–2024, 2014.
5. S. Bakas, M. Reyes, A. Jakab, S. Bauer, M. Rempfler, A. Crimi, R. T. Shinohara, C. Berger, S. M. Ha, M. Rozycki *et al.*, "Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the brats challenge," *arXiv preprint arXiv:1811.02629*, 2018.
6. S. Bakas, H. Akbari, A. Sotiras, M. Bilello, M. Rozycki, J. S. Kirby, J. B. Freymann, K. Farahani, and C. Davatzikos, "Advancing the cancer genome atlas glioma mri collections with expert segmentation labels and radiomic features," *Scientific data*, vol. 4, p. 170117, 2017.
7. S. Bakas, H. Akbari, A. Sotiras, M. Bilello, M. Rozycki, J. Kirby, J. Freymann, K. Farahani, and C. Davatzikos, "Segmentation labels and radiomic features for

[9] `https://hub.docker.com/repository/docker/fabianisensee/isen2020`

the pre-operative scans of the tcga-lgg collection," *The Cancer Imaging Archive*, 2017.

8. ——, "Segmentation labels and radiomic features for the pre-operative scans of the tcga-gbm collection. the cancer imaging archive," 2017.
9. O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *MICCAI*. Springer, 2015, pp. 234–241.
10. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
11. A. Myronenko, "3d mri brain tumor segmentation using autoencoder regularization," in *International MICCAI Brainlesion Workshop*. Springer, 2018, pp. 311–320.
12. Z. Jiang, C. Ding, M. Liu, and D. Tao, "Two-stage cascaded u-net: 1st place solution to brats challenge 2019 segmentation task," in *International MICCAI Brainlesion Workshop*. Springer, 2019, pp. 231–241.
13. F. Isensee, P. Kickingereder, W. Wick, M. Bendszus, and K. H. Maier-Hein, "Brain tumor segmentation and radiomics survival prediction: Contribution to the brats 2017 challenge," in *International MICCAI Brainlesion Workshop*. Springer, 2017, pp. 287–297.
14. G. Wang, W. Li, S. Ourselin, and T. Vercauteren, "Automatic brain tumor segmentation using cascaded anisotropic convolutional neural networks," in *International MICCAI brainlesion workshop*. Springer, 2017, pp. 178–190.
15. G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.
16. Y.-X. Zhao, Y.-M. Zhang, and C.-L. Liu, "Bag of tricks for 3d mri brain tumor segmentation," in *International MICCAI Brainlesion Workshop*. Springer, 2019, pp. 210–220.
17. R. McKinley, R. Meier, and R. Wiest, "Ensembles of densely-connected cnns with label-uncertainty for brain tumor segmentation," in *International MICCAI Brainlesion Workshop*. Springer, 2018, pp. 456–465.
18. R. McKinley, M. Rebsamen, R. Meier, and R. Wiest, "Triplanar ensemble of 3d-to-2d cnns with label-uncertainty for brain tumor segmentation," in *International MICCAI Brainlesion Workshop*. Springer, 2019, pp. 379–387.
19. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.
20. F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *International Conference on 3D Vision (3DV)*. IEEE, 2016, pp. 565–571.
21. M. Drozdzal, E. Vorontsov, G. Chartrand, S. Kadoury, and C. Pal, "The importance of skip connections in biomedical image segmentation," in *Deep Learning and Data Labeling for Medical Applications*. Springer, 2016, pp. 179–187.
22. T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.
23. F. Isensee, P. Kickingereder, W. Wick, M. Bendszus, and K. H. Maier-Hein, "No new-net," in *International MICCAI Brainlesion Workshop*. Springer, 2018, pp. 234–244.

24. K. Kamnitsas, W. Bai, E. Ferrante, S. McDonagh, M. Sinclair, N. Pawlowski, M. Rajchl, M. Lee, B. Kainz, D. Rueckert *et al.*, "Ensembles of multiple models and architectures for robust brain tumour segmentation," in *International MICCAI Brainlesion Workshop.* Springer, 2017, pp. 450–462.
25. F. Isensee, P. F. Jäger, S. A. Kohl, J. Petersen, and K. H. Maier-Hein, "Automated design of deep learning methods for biomedical image segmentation," *arXiv preprint arXiv:1904.08128*, 2019.
26. L. Maier-Hein, M. Eisenmann, A. Reinke, S. Onogur, M. Stankovic, P. Scholz, T. Arbel, H. Bogunovic, A. P. Bradley, A. Carass *et al.*, "Why rankings of biomedical image analysis competitions should be interpreted with care," *Nature communications*, vol. 9, no. 1, p. 5217, 2018.
27. Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3d u-net: learning dense volumetric segmentation from sparse annotation," in *International conference on medical image computing and computer-assisted intervention.* Springer, 2016, pp. 424–432.
28. A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. icml*, vol. 30, no. 1, 2013, p. 3.
29. D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *arXiv preprint arXiv:1607.08022*, 2016.
30. L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2017.