Stabilized Semi-Supervised Training for COVID Lesion Segmentation

Pranjal Sahu[1]
psahu@cs.stonybrook.edu
Vunnava Saikiran Kumar[2]
vunnavas.kirankr.eee15@itbhu.ac.in
Hong Qin[1]
qin@cs.stonybrook.edu

[1] Stony Brook University, New York, USA
[2] Indian Institute of Technology BHU, Varanasi, India

Abstract

We propose a novel stabilized semi-supervised training method to solve the challenging problem of covid lesion segmentation in CT scans. We first study the limitations of current models and based on our findings we introduce a lightweight SU-Net (Small U-Net) architecture. During training we feed the CT scans in sorted order of lesion occupancy and calculate a reliability score at each epoch to determine the stopping criteria. We test the proposed method on the largest publicly available COVID CT dataset called MOSMED dataset. By harnessing around 800 un-labelled COVID CT volumes comprising 25k CT slices, we improve the segmentation accuracy by around 2-4 dice percentage points depending upon the availability of labelled training data. We also compare our method with a recently published COVID lesion segmentation method called Semi-InfNet. The proposed method outperforms Semi-InfNet model and achieves state-of-the-art covid segmentation result on MOSMED dataset.

1 Introduction

Imaging technologies such as CT scans are being used as a complementary examination tool for the COVID-19 [6]. Due to volumetric information, CT scans are more reliable than the chest X-rays in the diagnosis of the disease as chest radiography is insensitive in presence of mild or early COVID-19 infection [7]. Due to the sudden surge in the number of patients, automated tools that can identify covid lesions from images are desirable which are particularly helpful in case of CT scans as it requires more reading time in comparison to chest X-Rays.

Recommendations have been made in [22] that CT is more suitable for quantifying and estimating the disease progression instead of using it as screening tool. The severity or disease progression can be estimated by calculating the lung volume occupied by covid lesion [8, 13, 20, 24]. Performing the delineation of such covid lesion from CT scans is a very tedious task and to automate this, covid lesion segmentation tools have recently been proposed [8, 9, 23, 26]. Here, primarily deep CNN based models have been proposed which typically require a large amount of labelled data to be trained. In case of covid, the labelled

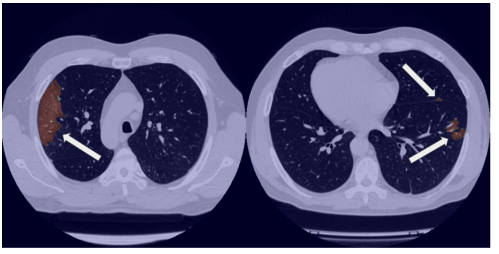

Figure 1: Figure illustrating two lung CT scans infected by COVID-19 taken from MOSMEDDATA dataset [19]. First CT slice has large covid lesion of size 139x59 px while the second CT slice has two covid lesions, one of size 6x12 px and other of size 44x41 px.

training data is still scarce and methods which are annotation efficient are preferable. Related to this efforts are being made to develop annotation-efficient deep learning models for segmentation of covid lesion from CT scans [8, 18, 23, 26]. Semi-supervised learning is a popular paradigm which tackles the issue of labelled data scarcity by utilizing the abundant un-labelled data effectively [4, 10, 16, 28].

Under the semi-supervised learning domain, two recent works have been proposed which aim to solve the covid lesion segmentation problem namely COPLE-Net [26] and Semi-InfNet [9]. In [26], authors proposed a self-ensembling based training method [25], to utilize the un-labelled data for improving the segmentation performance. They modified the ensembling scheme by reducing the contribution of student to teacher's weights when the student training loss is higher than a certain threshold. This resulted in reduction of noise in the performance of teacher making it more robust to outliers. In addition to this they proposed a novel noise-robust Dice loss function to deal with the problem of noisy annotation. Similarly, Semi-InfNet [25] model is trained following the semi-supervised approach. They perform a randomly selected iterative label propagation on un-labelled data to be used for training the segmentation model. Their model outperformed previous state-of-the-art segmentation models such as U-Net [21], Dense-UNet [15], U-Net++ [29] etc. on the COVID-19 CT segmentation dataset [1].

To appreciate the problem of covid lesion segmentation, we show two different covid infected CT slices in Figure 1, having lesions of varying sizes taken from the publicly available covid CT dataset called MOSMED [19]. Segmenting such infected region is challenging due to factors such as low contrast with background, large variance in size, irregular structure [26]. To visualize the variance in the dimensions of covid lesion we plot the height vs width of all the annotated covid lesions present in the 50 CT volumes in MOSMED dataset [19] in Figure 2(a). Each point in Figure 2(a), is a lesion obtained by performing connected components on 2D CT slices. It can be observed that covid lesions come in wide range of sizes. In Figure 2(b), we plot the count of CT scans having certain number of covid lesions in it. We observe that a single CT scan can have multiple covid lesions of varying sizes. In such a challenging setting where both smaller and larger covid lesion need to be segmented accurately, we observe a certain variance in the model's performance despite being trained in a similar fashion. The variance in model's performance makes it hard to determine the stopping criteria while training, as even within two epochs a large performance drop could occur.

To solve this problem we propose a number of steps in the training method and model

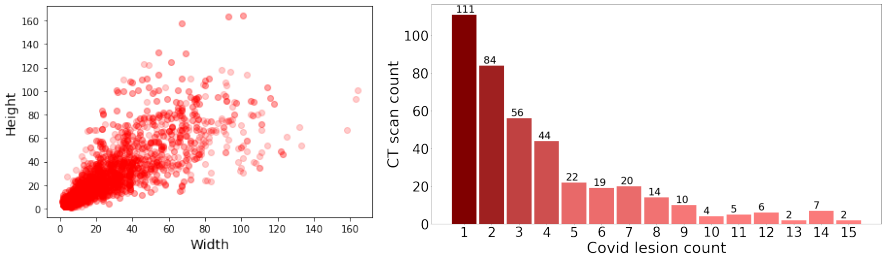

Figure 2: a) Distribution of width vs height (in pixels) of covid lesions in the MOSMED dataset [19]. (b) Count distribution of covid lesions in each CT slice in MOSMED.

architecture. The proposed steps improve the stability of the model while training and helps in selecting the right model weights resulting in higher segmentation accuracy. Through a series of experiments we first demonstrate the effect of model size on the test accuracy when the training data is scarce. Based on the findings we design a smaller U-Net model called SU-Net and add certain modifications to it in order to improve training efficiency. Later we use the abundant un-labelled data in a stabilized semi-supervised learning approach to further improve the performance. For using the un-labelled data, we perform label propagation using SU-Net model trained with labelled data and in second stage train a Semi-SU-Net model using the generated pseudo masks. The proposed model improves upon the Semi-InfNet model's performance. Our contributions in this paper are summarized here:

1. To reduce the high variance in loss observed while training the model, we introduce a sorting scheme based on lesion size to stabilize the training. A lighter version of U-Net model is proposed which we call SU-Net. The proposed model reduces the problem of over-fitting while simultaneously reduces the training time.

2. To harness the abundant available un-labelled data we propose a semi-supervised learning method which generates pseudo labels using SU-Net model. To stabilize the semi-supervised training, we propose a Reliability score to determine the best model weights from the semi-supervised training loop.

3. We compare our method against the current state-of-the-art covid lesion segmentation model called Semi-Inf Net and show superior performance on largest publicly available covid CT dataset called MOSMED.

In Section 2, we study the current segmentation CNNs and based on our findings we propose a modified U-Net architecture called SU-Net. Also, we describe the various measures taken to stabilize the training. In Section 3, we describe the datasets used for evaluation purpose and in Section 4 we compare the proposed method's performance on the MOSMED dataset and COVID-19 infection dataset.

2 Our Method

The training pipeline used in our model is shown in Figure 3. It comprises two stages, first fully-supervised training on limited labelled data and second training using un-labelled data with pseudo masks obtained from Stage 1 trained CNN. We next describe the steps taken to enhance the segmentation performance in these two steps.

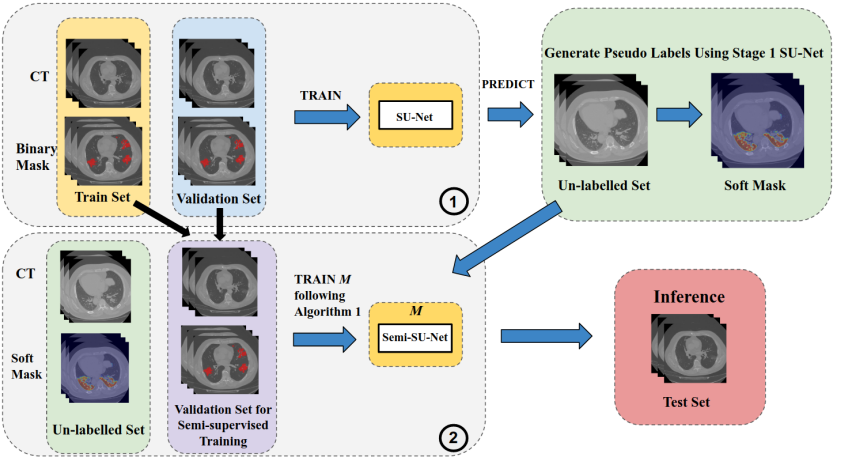

Figure 3: Semi-supervised training methodology adopted in the proposed method. A segmentation CNN is first trained on a small amount of labelled training data and validation set is used to select the best weights. The CNN is then used to generate the pseudo labels for the abundant un-labelled data. The segmentation CNN is then trained using the pseudo masks and Stage 1 train and validation set are used to monitor performance and to select best weights.

2.1 Stabilized fully-supervised training

We first study the performance of widely used segmentation models such as U-Net [21] and U-Net++ [29] on a very small amount of labelled training set (4 CT volumes comprising 57 CT scans) and record their performance (dice score) on a validation set (10 CT volumes comprising 179 CT scans) taken from MOSMED dataset in Figure 5. Both the models are trained with Binary Cross Entropy loss function, with a batch size of 2, using Adam optimizer [14] having initial learning rate of 0.0001. The last layer in both the models is a 1x1 convolution layer so that the sigmoid operation could be applied on the output logit. We observe that despite having a low initial learning rate, there is a significant fluctuation in the validation set dice score for both U-Net and U-Net++. This creates a problem in determining the correct epoch having maximum performance on un-seen data. This problem is possibly occurring because of large model capacity (parameters) in comparison to the training dataset. To avoid the over-fitting problem, we then reduce the number of filters in U-Net by 4 times across all layers and call the derived CNN as SU-Net (Small U-Net). We then observe that the fluctuations have decreased in comparison to U-Net and U-Net++, however, due to reduced capacity, the SU-Net model starts learning slowly and gives non-zero dice score on validation set after 40+ epochs, see Figure 5(a). To improve the training speed we then add a GroupNorm layer (GN) [27] after each down-sample and up-sample layer in SU-Net. The combined effect of Group Norm and lesser number of filters in SU-Net is shown in Figure 5(a).

As mentioned before, covid lesions come in a wide range of size and can occur multiple times in the same lung CT scan, see Figure 2. This wide variance in the size and count of covid lesions interferes with the learning process of CNN. To solve this problem, we simply sort the input CT slices by the sum of non-zero pixels in the ground truth mask, i.e.

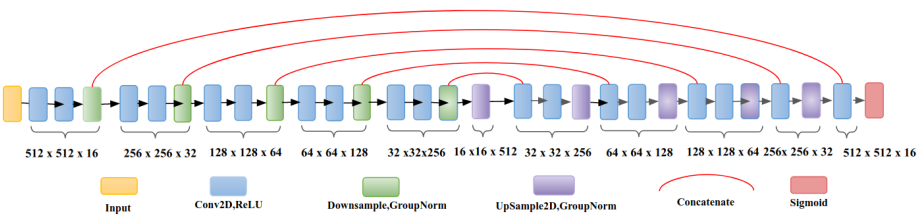

Figure 4: SU-Net architecture used in the proposed model. It uses GroupNorm after each layer to speed up training. The number of filters are 1/4th of that present in U-Net which reduces the problem of over-fitting and stabilizes the training.

Table 1: CUSUM values for different thresholds for the train set dice score of networks with and without sorting scheme to quantify stability while training. (lower is better)

Thresholds	0.02	0.04	0.08	0.16	0.32
CUSUM Value w/o sort	37	27	11	7	7
CUSUM Value w sort	11	4	2	0	0

CT containing smaller lesions is fed before CT containing larger covid lesions. This step improves the stability of training. The impact of this sorting scheme is shown in Figure 5(b). The outcome of these steps results in a stable training of the segmentation CNN model during fully-supervised training. One explanation for this behavior comes from the curriculum learning perspective [2] where it has been discussed how the network learning could be improved by first training it with simpler cases and later adding harder samples. In our scenario, CT scans with multiple lesions having large difference in shape and size are harder compared to the scans with only 1 or 2 lesions. Training the network with scans having fewer lesions and steadily increasing the lesion size and count makes the learning process easier for the network. Similar to the approach shown in [17] we also quantify the stability of the learning strategy. For this we calculated the CUSUM control chart values for different thresholds. The values quantify the number of epochs where the network learning was "out of control" in some sense. The values are shown in Table 1, where it can be observed that the network learning stability has improved due to the sorting scheme across all thresholds.

2.2 Stabilized Semi-supervised training

To harness the abundant un-labelled data we use the pseudo labelling approach. From each un-labelled CT volume we first select CT scans containing lung. We use the publicly available lung segmentation CNN reported in [12] for this purpose. All the CT scans in the un-labelled set are first passed through the 2D lung segmentation [12] to filter out non-lung CT scans. The filtering process results in 25000 CT scans in the un-labelled set. The CNN model trained in Stage 1 is then used to generate the soft-mask (output of sigmoid and non-thresholded) for all CT scans in un-labelled set. A second SU-Net model named M is then trained following Algorithm 1, using the un-labelled CT scans and their corresponding soft mask in Stage 2, refer Figure 3. We combine the Train set and Validation set from Stage 1 to form the Validation set for Stage 2. Here, training is only performed using un-labelled dataset. As shown in Algorithm 1, training of Stage 2 CNN is performed for T steps. In each step $e \in [0, T]$, 10 volumes are randomly selected from the 806 volumes in un-labelled

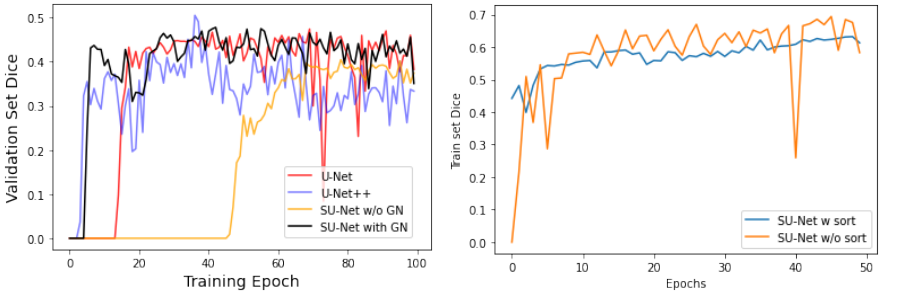

Figure 5: Comparison of baseline models with SU-Net (GN) when trained on 4 CT volumes. We observe that the oscillations are significantly reduced in comparison to U-Net and U-Net++. Feeding the scans in sorted order results in stable and higher performance.

set. CT scans present in these 10 volumes are then sorted together by the sum of pixel values in corresponding pseudo mask before training M. Even after stabilizing the training of M following steps described in 2.1, we observe a certain variance in the performance of M on validation set. This is possibly due to noisy masks generated by pseudo labelling. To counter this we introduce a metric called Reliability Score m_e for training step e, which based on M's performance on validation set in past t steps determines its robustness against noisy training step. The formulation of m_e is defined below:

$$\textit{Reliability Score, } m_e = \frac{1}{N}\sum_{c=0}^{N} w_{c,e} * d_{c,e}, \tag{1}$$

$$w_{c,e} = (1 - m_{c,e}) + \alpha * s_{c,e}, \tag{2}$$

$$m_{c,e} = \frac{1}{t}\sum_{i=e-t}^{e} d_{c,i} \quad \textit{and} \quad s_{c,e} = \sqrt{\frac{\sum (d_{c,i} - m_{c,e})^2}{t}}, \tag{3}$$

where, $d_{c,e}$ is the dice score for the c^{th} CT scan in validation set after e^{th} training step, N is the total number of CT scans in validation set, $m_{c,e}$ is the mean of the dice scores d_c for last t steps from step e, $s_{c,e}$ is the standard deviation of the dice scores d_c for last t steps from step e and α is a hyper-parameter which is fixed to value 10 to give more weight to standard deviation. The intuition behind this weighing scheme is to penalize model weights which perform poorly on CT slices with high standard deviation in dice scores. High mean and low standard deviation means that model is able to capture such lesions accurately consistently and therefore such cases are easier for the model to learn, however, lower dice and high standard deviation are the cases which are harder to be captured by the model and that is why the dice score oscillates for such cases from step to step. By giving more weights to such cases in the m_e we choose model weights which are able to capture harder cases consistently and accurately.

Algorithm 1: Stabilized Semi-supervised training

Input : Fully-supervised trained SU-Net model S, trained using labelled data D_L. CT volumes v_i^{UL} where $i \in [0,P]$ from un-labelled data D_{UL}, CT scans c_i^{VAL} from validation set D_{VAL}, T total number of training steps

Output: Trained model M for inference

Obtain output for each CT slice c_i^{UL} in D_{UL} using S

for $e \leftarrow 0$ **to** T **do**

  Obtain 10 random CT volumes v^{UL} from D_{UL}.

  Sort all the CT slices c_i^v together, taken from the 10 CT volumes v^{UL} by sum of pixel values in corresponding soft pseudo mask p_i^v in preparation to train model M.

  Train model M using c_i^v and p_i^v.

  Obtain the result using M on the CT slices c_i^{VAL} in validation set of Stage 2.

  Obtain the reliability score m_e for the step e using the Equation 2 if $e > 100$.

  Store the current weights of model M.

end

return M with weights having maximum reliability score m_e

Table 2: Count of CT scans (and volumes) used in different splits while training in different experiments.

Labelled Data %	Train Set	Validation Stage (1)	Un-labelled Set	Validation Stage (2)	Test Set
12.5%	57 (4)	179 (10)	25000 (806)	236 (14)	126 (10)
25%	110 (8)	179 (10)	25000 (806)	289 (18)	126 (10)
50%	217 (15)	179 (10)	25000 (806)	396 (25)	126 (10)
100%	480 (30)	179 (10)	25000 (806)	659 (40)	126 (10)

3 Dataset and Training

3.1 MOSMED Dataset

We use the publicly available MOSMED [19] lung CT dataset with covid infection for this study. It contains anonymised human lung CT scans with COVID-19 annotation. The CT scans are collected between 1st of March, 2020 and 25th of April, 2020 and are provided by municipal hospitals in Moscow, Russia. The dataset comprises a total 1100 studies (CT volumes) out of which a small set of 50 volumes are annotated by the experts. To reduce the size of dataset, MOSMED gives slices in a gap of 10 along the z direction. This dataset in our knowledge is by far the largest publicly available dataset of covid infected CT scans. The dataset is divided into 5 categories based on severity of the lung tissue abnormalities.

The category CT-0 has no covid infection, and the volume of covid infected region increases from category CT-1 to CT-4. We don't use CT scans from CT-0 category as it has no covid infection. For the experiments we randomly split the annotated 50 volumes into 3 sets namely Train (30 volumes), Validation (10 volumes) and Test (10 volumes). We conduct experiments to study how amount of labelled data impacts the segmentation performance. For this we use varying amount of labelled data (12.5%, 25%, 50%, 100%) in Stage 1. The validation and test set remains the same across all these 4 experiments. All the CT scans are

Figure 6: Qualitative Comparison.

added with an offset of 1024 value and then divided by 1024 in pre-processing step.

3.2 Training Details

We train all the SU-Net models using the Binary Cross Entropy (BCE) loss as defined below:

$$BCE = -\frac{1}{S}\sum_{i=0}^{S} y_i \cdot \log(\hat{y}_i) + (1-y_i) \cdot \log(1-\hat{y}_i), \tag{4}$$

For Stage 1 training, $y_i \in \{0,1\}$ is the ground truth binary mask value, while for Stage 2 training y_i is a real value $\in [0,1]$, $\hat{y}_i$ is the output of the sigmoid layer of 2D CNN at pixel i. Adam optimizer is used to optimize the CNN's weights and a learning rate of 0.0005 is used in Stage 2 training while for Stage 1 learning rate of 0.0001 is used. For Stage 1, training is performed for a maximum of 100 epochs and the model weights with the highest validation set dice score is taken for performing inference. Stage 2, training is performed for a maximum of $T = 300$ steps. A warm-up period of 100 steps is taken before calculating the Reliability Score m_e. After the training is done for T steps weights with highest reliability score m_e is chosen and returned.

4 Results and Comparison

We compare our model against baseline segmentation models such as U-Net++ [29], U-Net [21] as well as one recently published state-of-the-art covid lesion segmentation model called Semi-InfNet [9]. Both U-Net and U-Net++ are trained using the Binary Cross Entropy loss, with Adam Optimizer and an initial learning rate of 0.0001. For both U-Net and U-Net++ the last layer is sigmoid and training is done for a maximum of 100 epochs. Model weights with highest validation set dice score is selected to perform inference on the Test split. For Semi-InfNet, we use the publicly available, author provided code to train the model. Experiments are done by varying the amount of labelled training data . We evaluate the model's performance by using standard segmentation metrics such as Dice, Sensitivity and Specificity. All the segmentation masks are thresholded at 0.5 value while calculating these metrics. Quantitative results are shown in Table 3 where we compare the performance of models trained using only labelled data i.e. fully supervised and unlabelled data i.e. semi-supervised training. We compare the performance of our proposed Semi-SU-Net segmentation model with the current state-of-the-art model called Semi-InfNet. For the Semi-InfNet and InfNet models we use the ResNet50 [11] as backbone. InfNet model is trained with the default hyper-parameters mentioned in [9] for 100 epochs. For Semi-InfNet model, the Inf-Net model is used to generate pseudo labels for un-labelled dataset. Training of Semi-Inf-Net

Table 3: Performance comparison of covid lesion segmentation methods. (Mean ± std). D = Dice, Se= Sensitivity and Sp = Specificity.

		Labelled Data Percentage			
	Model	12.5%	25%	50%	100%
		Supervised Methods Comparison			
D	U-Net [21]	0.55±0.331	0.553±0.33	0.528±0.339	0.551±0.338
	U-Net++ [29]	0.541±0.308	0.432±0.292	0.523±0.307	0.613±0.318
	Inf-Net [9]	0.431±0.274	0.483±0.257	0.571±0.257	0.6±0.264
	SU-Net (Ours)	**0.627**±0.292	**0.596**±0.307	**0.62**±0.269	**0.651**±0.269
Se	U-Net [21]	0.99±0.032	0.99±0.32	0.99±0.033	0.99±0.032
	U-Net++ [29]	0.993±0.023	0.993±0.025	0.992±0.028	0.993±0.023
	Inf-Net [9]	**0.997**±0.006	**0.999**±0.003	**0.997**±0.009	0.997±0.009
	SU-Net (Ours)	0.993±0.016	0.994±0.02	0.996±0.008	**0.997**±0.004
Sp	U-Net [21]	0.602±0.325	0.608±0.304	0.602±0.335	0.634±0.315
	U-Net++ [29]	0.563±0.294	0.43±0.306	0.558±0.314	0.708±0.288
	Inf-Net [9]	0.435±0.297	0.463±0.284	0.601±0.27	0.666±0.259
	SU-Net (Ours)	**0.769**±0.25	**0.667**±0.3	**0.678**±0.25	**0.849**±0.184
		Semi-Supervised Methods Comparison			
D	Semi-Inf-Net	0.558±0.265	0.548±0.268	0.582±0.245	0.609±0.264
	Semi-SU-Net (M)	**0.644**±0.282	0.617±0.302	0.635±0.266	0.663±0.266
	Semi-SU-Net (S)	0.64±0.281	**0.643**±0.285	**0.641**±0.276	**0.664**±0.259
Se	Semi-Inf-Net	**0.995**±0.013	**0.997**±0.009	**0.997**±0.01	0.996±0.011
	Semi-SU-Net (M)	0.992±0.021	0.993±0.021	0.996±0.01	**0.998**±0.003
	Semi-SU-Net (S)	0.992±0.021	0.995±0.014	0.996±0.008	**0.998**±0.003
Sp	Semi-Inf-Net	0.648±0.28	0.563±0.274	0.599±0.254	0.675±0.244
	Semi-SU-Net (M)	0.776±0.215	0.705±0.287	0.667±0.237	**0.79**±0.229
	Semi-SU-Net (S)	**0.792**±0.206	**0.706**±0.264	**0.717**±0.242	0.771±0.227

model takes few hyper-parameters such as number of groups to divide the un-labelled data and the epochs to train the model for each group. We use 25 groups and the 25000 unlabelled CT scans are divided into groups of 1000 each. The Semi-InfNet model is trained iterativaly for 1 epoch for each group following the methodology proposed in [9]. U-Net, U-Net++ and SU-Net are trained on Titan Xp GPU while the Semi-InfNet model is trained using Tesla T4 NVIDIA GPU 16GB memory on ubuntu 16.04 LTS vitual machine. Training of Semi-InfNet takes around 14 hours in total while training of Semi-SU-Net completes within 2 hours.

In Table 3, we observe that the proposed model SU-Net consistently out-performs the models in comparison across all variations of labelled data in terms of Dice metric. However, we also notice that Inf-Net model results in higher Sensitivity at 12.5%, 25% and 50% of labelled data experiments. Due to higher specificity, the resultant Dice score is higher for SU-Net. We also observe that the Stabilized Semi-SU-Net model again out-performs the state-of-the-art Semi-Inf-Net model across all variations of labelled data in terms of Dice metric. Due to the semi-supervised training, the Dice score improves from 0.627 to 0.64, 0.596 to 0.643, 0.62 to 0.641 and from 0.651 to 0.664 for 12.5%, 25%, 50% and 100% of labelled data respectively in case of stabilized Semi-SU-Net. We also notice that by choosing the weights based only on the maximum validation dice score results in drastic drop in dice score (0.617) for the 25% labelled data experiment. This is because of the noisy

Table 4: Quantitative Results of Infection regions on Fully Labelled COVID-SemiSeg Dataset. D = Dice, Se= Sensitivity and Sp = Specificity.

Model	D	Se	Sp
Inf-Net [9]	0.682	0.692	**0.943**
SU-Net with sort (Ours)	**0.726**	0.949	0.765
SU-Net without sort (Ours)	0.712	**0.950**	0.730

training in semi-supervised setting. The comparison of Semi-SU-Net (Stabilized, S) and Semi-SU-Net (Max Val dice, M) demonstrates the benefits of the Reliability score metric as it leads to more stable and consistent performance across all levels of labelled dataset. A qualitative comparison of the output from Semi-Inf-Net and Semi-SU-Net is shown in Figure 6. Additional results are present in the supplementary document.

We also performed experiments on the COVID-Semiseg [9] which is built of 100 labeled CT slices from the COVID-19 CT Segmentation dataset [1] and 1600 unlabeled images from the COVID-19 CT Collection dataset [5]. The results shown in Table 4 depict that SU-Net with sort performs better than the state-of-the-art Inf-Net model when only using the labelled data for training.

5 Conclusion

For training deep learning models, data annotation has always been an issue. This problem is particularly severe in current situation where a novel disease like COVID-19 has disrupted lives of common people including clinicians. To obtain the best performance from limited number of pixel annotated ground truth mask we propose a semi-supervised training approach. We study the problems in the training of current CNN architectures for the task of covid lesion segmentation and propose a novel model called SU-Net derived from U-Net. Using a novel training methodology, we are able to stabilize the training of model which results in state-of-the-art performance on the MOSMED dataset. Our results also demonstrate that superior segmentation performance could be obtained even with very less amount of labelled data. Being lightweight (1.08 M parameters), training of the proposed model could be done on a single GPU which can be beneficial in situations with constrained hardware resources.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. IIS-1715985 and IIS-1812606.

References

[1] Covid-19 ct segmentation dataset. (accessed July 24, 2020). https://medicalsegmentation.com/covid19/.

[2] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48, 2009.

[3] Yukun Cao et al. Longitudinal assessment of covid-19 using a deep learning–based quantitative ct pipeline: Illustration of two cases. *Radiology: Cardiothoracic Imaging*, 2(2):e200082, 2020.

[4] Krishna Chaitanya, Neerav Karani, Christian F. Baumgartner, Anton Becker, Olivio Donati, and Ender Konukoglu. Semi-supervised task-driven data augmentation for medical image segmentation, 2020.

[5] Joseph Paul Cohen, Paul Morrison, Lan Dao, Karsten Roth, Tim Q Duong, and Marzyeh Ghassemi. Covid-19 image data collection: Prospective predictions are the future. *arXiv preprint arXiv:2006.11988*, 2020.

[6] Tom MH de Jaegere et al. Radiological society of north america chest ct classification system for reporting covid-19 pneumonia: Interobserver variability and correlation with rt-pcr. *Radiology: Cardiothoracic Imaging*, 2(3):e200213, 2020.

[7] D. Dong et al. The role of imaging in the detection and management of covid-19: a review. *IEEE Reviews in Biomedical Engineering*, pages 1–1, 2020.

[8] Omar Elharrouss, Nandhini Subramanian, and Somaya Al-Maadeed. An encoder-decoder-based method for covid-19 lung infection segmentation. *arXiv preprint arXiv:2007.00861*, 2020.

[9] Deng Ping Fan et al. Inf-net: Automatic covid-19 lung infection segmentation from ct images. *IEEE Transactions on Medical Imaging*, 2020.

[10] Abdur R Feyjie, Reza Azad, Marco Pedersoli, Claude Kauffman, Ismail Ben Ayed, and Jose Dolz. Semi-supervised few-shot learning for medical image segmentation. *arXiv preprint arXiv:2003.08462*, 2020.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, et al. Deep residual learning. *Image Recognition*, 2015.

[12] Johannes Hofmanninger, Forian Prayer, Jeanny Pan, Sebastian Röhrich, Helmut Prosch, and Georg Langs. Automatic lung segmentation in routine imaging is primarily a data diversity problem, not a methodology problem. *European Radiology Experimental*, 4(1):1–13, 2020.

[13] Lu Huang et al. Serial quantitative chest ct assessment of covid-19: Deep-learning approach. *Radiology: Cardiothoracic Imaging*, 2(2):e200075, 2020.

[14] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980.

[15] Xiaomeng Li, Hao Chen, Xiaojuan Qi, Qi Dou, Chi-Wing Fu, and Pheng-Ann Heng. H-denseunet: hybrid densely connected unet for liver and tumor segmentation from ct volumes. *IEEE transactions on medical imaging*, 37(12):2663–2674, 2018.

[16] Yuexiang Li, Jiawei Chen, Xinpeng Xie, Kai Ma, and Yefeng Zheng. Self-loop uncertainty: A novel pseudo-label for semi-supervised medical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 614–623. Springer, 2020.

[17] Gongbo Liang, Sajjad Fouladvand, Jie Zhang, Michael A Brooks, Nathan Jacobs, and Jin Chen. Ganai: Standardizing ct images using generative adversarial network with alternative improvement. In *2019 IEEE International Conference on Healthcare Informatics (ICHI)*, pages 1–11. IEEE, 2019.

[18] Jun Ma et al. Towards efficient covid-19 ct annotation: A benchmark for lung and infection segmentation. *arXiv preprint arXiv:2004.12537*, 2020.

[19] SP Morozov et al. Mosmeddata: Chest ct scans with covid-19 related findings dataset. *arXiv preprint arXiv:2005.06465*, 2020.

[20] Xiaolong Qi et al. Machine learning-based ct radiomics model for predicting hospital stay in patients with pneumonia associated with sars-cov-2 infection: A multicenter study. *medRxiv*, 2020.

[21] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[22] Geoffrey D Rubin et al. The role of chest imaging in patient management during the covid-19 pandemic: a multinational consensus statement from the fleischner society. *Chest*, 2020.

[23] Fei Shan et al. Lung infection quantification of covid-19 in ct images with deep learning, 2020.

[24] Cong Shen et al. Quantitative computed tomography analysis for stratifying the severity of coronavirus disease 2019. *Journal of Pharmaceutical Analysis*, 2020.

[25] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in neural information processing systems*, pages 1195–1204, 2017.

[26] Guotai Wang et al. A noise-robust framework for automatic segmentation of covid-19 pneumonia lesions from ct images. *IEEE Transactions on Medical Imaging*, 2020.

[27] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

[28] Yi Zhou, Xiaodong He, Lei Huang, Li Liu, Fan Zhu, Shanshan Cui, and Ling Shao. Collaborative learning of semi-supervised segmentation and classification for medical images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2079–2088, 2019.

[29] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: A nested u-net architecture for medical image segmentation. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pages 3–11. Springer, 2018.