

ImageCLEF2020: Laterality-Reduction Three-Dimensional CBAM-Resnet with Balanced Sampler for Multi-Binary Classification of Tuberculosis and CT Auto Reports

Xing Lu²[0000-0001-6517-7497], Eric Y Chang^{1,2}[0000-0003-3633-5630], Zhaohui Liu³[0000-0002-0413-6023], Chun-nan Hsu²[0000-0002-5240-4707], Jiang Du²[0000-0002-9203-2450], Amilcare Gentili^{1,2}[0000-0002-5623-7512]

¹ San Diego VA Health Care System, San Diego, CA, USA

² University of California, San Diego, CA, USA

³ RIMAG Medical Imaging Corporation, Beijing, China
agentili@ucsd.edu

Abstract. Detection and characterization of tuberculosis and the evaluation of lesion characteristics are challenging. In an effort to provide a solution for a classification task of tuberculosis findings, we proposed a laterality-reduction 3D CBAM Resnet with balanced-sampler strategy. With proper usage of both provided masks, each side of the lung was cropped, masked, and rearranged so that laterality could be neglected, and dataset size doubled. Balanced sampler in each batch sampler was also used in this study to address the data imbalance problem. CBAM was used to add an attention mechanism in each block of the Resnet to further improve the performance of the CNN.

Keywords: Tuberculosis, Convolutional Neural Network, Laterality-Reduction, Dataset Imbalance, Attention Mechanism

1 Introduction

Tuberculosis (TB) is a bacterial infection caused by the germ *Mycobacterium tuberculosis* and is a leading cause of death from infectious disease worldwide. An epidemic in many developing regions, such as Africa and Southeast Asia, it was responsible for 1.6 million deaths in 2017 alone. There are different manifestations of TB which require different treatments, making the detection and characterization of TB disease and the evaluation of lesion characteristics critically important tasks in the monitoring, control, and treatment of this disease. An accurate and automated method for the classification of TB from CT images may be especially useful in regions of the world with few radiologists.

Copyright © 2020 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0). CLEF 2020, 22-25 September 2020, Thessaloniki, Greece

The ImageCLEF 2020 Tuberculosis – CT report challenge [1,2] was concentrated on the automated CT report generation task. This year, three labels for each side of the lungs were provided, namely labeling for the presence of TB lesions, pleurisy, and caverns. In addition, a dataset containing chest CT scans of 403 TB patients (283 for training and 120 for testing) was provided.

2 Methods

2.1 Data

The dataset provided for the CT report task training contained a total of 283 patients, with labeling provided for six categories. As seen in Figure 1(a), the training dataset distribution of pathology was quite unbalanced, with “lung affected” at both sides being the most commonly seen label, caverns being seen less, and pleurisy being the most rarely observed condition. 17 sub-category combinations of the six categories are shown in Figure 1(b), with “lung affected for both sides” (represented by $[1,1,0,0,0,0]$) as the sub-category with the most dataset counts of 73.

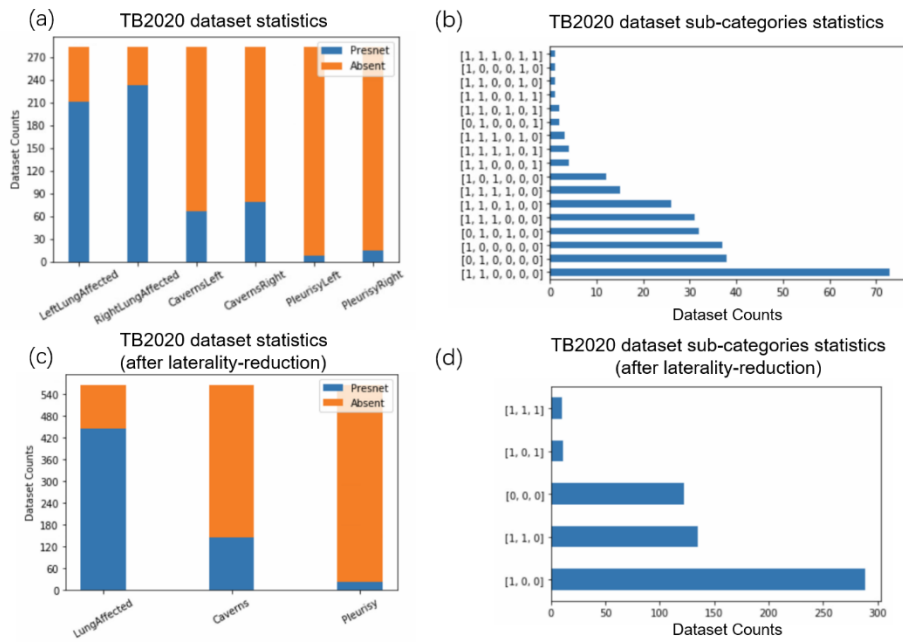


Fig. 1. TB2020 dataset statistics. (a) Original dataset with six categories, (b) 17 sub-category combinations with “lung affected with both sides” showing the most counts, (c) dataset after laterality-reduction with laterality neglected in three categories, and (d) dataset for all sub-category combinations after laterality-reduction.

By neglecting the laterality of the lungs and re-arranging the dataset, we found that the dataset counts doubled to 576, but the categories for classification were sharply reduced from six to three, as shown in Figure 1(c). When combining these resulting three categories, there were only five sub-categories, as shown in Figure 1(d). The “lung affected by lesion” category (represented by [1,0,0]) had the most counts of 288, while lung with all three pathologies present had the fewest counts of 10.

2.2 Pre-processing

To perform laterality-reduction, it was necessary to properly obtain images from both sides of the lungs from the original dataset. Laterality-reduction images were obtained according to the algorithm pipeline as shown in Figure 2. The images for the ImageCLEF tuberculosis task were provided as NIFTI 3D datasets. Two versions of lung segmentation masks were also provided [3,4]. The first version of segmentation (denoted as Mask 1) provided more accurate masks, containing masks for left and right laterality individually (values equal 1 for left and 2 for right), but in the most severe TB cases, there was a tendency to miss large abnormal regions of lungs. On the other hand, the second segmentation (denoted as Mask 2) provided less precise bounds, but was more stable in terms of including lesion areas, though it contained the entire lung area (including both left and right sides of the lungs).

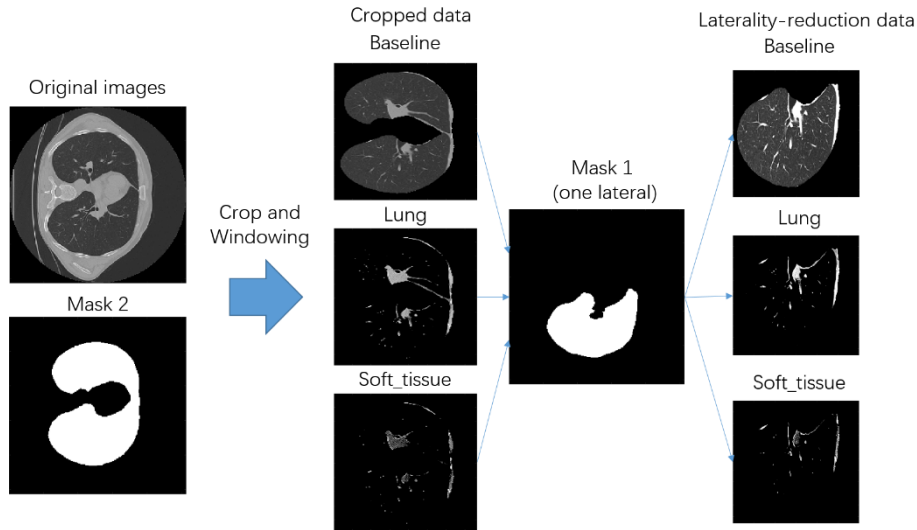


Fig. 2. Pipeline for the proposed 2-step laterality-reduction segmentation of the lungs.

In order to take advantages of both masks, a two-step mask-cropping algorithm was proposed in this study. As shown in Figure 2, both segmentation versions were used to generate a laterality-reduction lung segmentation. First, the original NIFTI-formatted dataset was transformed into image data using the NiBabel package [5]. Then, the reformatted images were adjusted to three different window levels, namely baseline,

and soft tissue, and then normalized. For baseline window level, the foreground was obtained via the `otsu_thresholding` algorithm provided in openCV package [6]; for lung and soft tissue, the image levels were set as [-600,1500] and [50,350], respectively. Then, images were normalized to [0,1] with their mean and std value. Afterward, each laterality of the images was cropped according to Mask 1. For the left laterality of the lungs, the right boundary of the lungs was found and used to crop the left side from the images at stage 1. Similarly, the right laterality used the left boundary to obtain the right side from the images at stage 1. Finally, all three windows and levels of laterality-reduction data were saved, and annotation file was rearranged for use in further training.

2.3 Network Design and Training Strategy

As shown in Figure 3, a 3D convolutional block attention module (CBAM)-Resnet was designed to train the model for 3-class binary classification based on the PyTorch framework. A standard 3D-resnet34 [7] was used as the convolutional neural network backbone, with three fc layers to be the classifier. CBAM [8] was used to implement channel and spatial attention mechanisms for each block of the Resnet. Sigmoid was used as the activation function for binary classification.

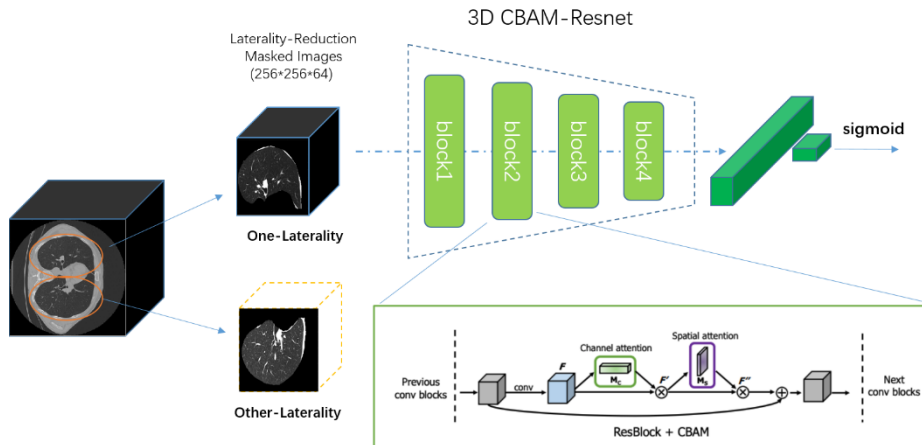


Fig. 3. Proposed laterality-reduction 3D CBAM-Resnet architecture. Each laterality of original data is cropped and masked, then fed into a 3D Resnet for training and inference. Each block of the Resnet is modified with convolutional block attention module (CBAM).

To train the neural network, we used a workstation with 4 Nvidia GTX 1080 Ti video cards, 128 GB RAM, and a 1 TB solid state drive. The training dataset was randomly split to form a validation cohort comprised of 20% of the original dataset. During the training process, to avoid over-fitting, image augmentation and balanced sampler were implemented in each batch. For each batch, 12 datasets that were fed into the network were dynamically generated from saved metadata with different window levels as a single channel and were interpolated into a $3*64*256*256$ size torch tensor. For the image augmentation, traditional data augmentation methods, including brightness,

shear, scale, and flip were applied. The balanced sampler strategy was adopted during the training process, which equalized the data sampled from all three classes for each batch [9].

Binary CrossEntropy (BCE) was used as the baseline for the multi-binary classification loss. Then, to improve the performance of the network, weighted BCE loss was applied to let the network focus more on the “lung affected” and “caverns” categories. Weighted focal loss was also applied in order to let the network focus further on more difficult examples [10]. All losses were realized on the PyTorch platform according to equations (1) and (2):

$$\text{loss}_{focal}(l_{bce}, \alpha, \gamma) = \alpha(1 - l_{bce})^\gamma l_{bce} \quad (1)$$

$$l_{bce}(o, t, w) = -1/n \sum_i w[i] * (t[i] * \log(o[i]) + (1 - t[i]) \log(1 - o[i])). \quad (2)$$

Here, o means calculated output, t means target, and w means weights of classes. When $\alpha = 1$ and $\gamma = 0$, loss_{focal} is the same with BCE loss. In this study, w equals [4, 2, 1] $\gamma = 2$ for focal loss.

3 Results

3.1 Experimental Results

In order to find the best combination of techniques for submission, we tested various combinations using half of the dataset with 20 epochs of training. For each epoch, half of the dataset was randomly selected for training. The experiments were conducted with and without balanced sampling, with and without CBAM, and with various losses (i.e., BCE, wBCE, wFocal). During the training, epochs with the best mean AUC value were saved. Then, models of different experiments were evaluated using the same validation dataset, with the results shown in Figure 4.

Fig. 4. (a) and (b), which show techniques without and with balanced sampler (bsmp), demonstrate that mean AUC is significantly improved from 0.678 to 0.838. With CBAM in (c), the mean AUC is slightly improved to 0.844. With wBCE and wFocal as the loss instead of BCE in (d) and (e), the mean AUC is improved to 0.885 and 0.892. Then, with the full dataset used as the training dataset, the mean AUC combined with wFocal achieved the highest mean AUC score of 0.916.

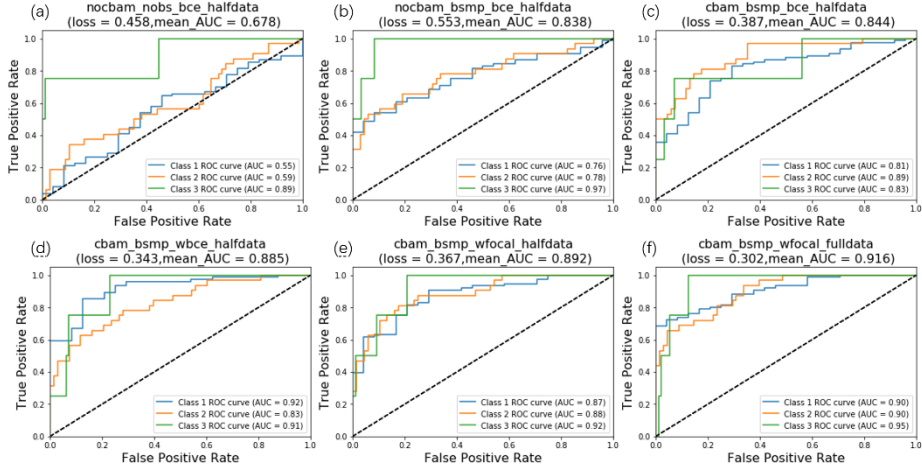


Fig. 4. Model performance comparison of different combinations of the techniques, including without cbam, without balanced sampler (bsmp), and with different losses (i.e., binary cross entropy (bce), focal losses, weighted bce, and weighted focal)

A comparison of experiment results is also summarized in Table 1.

Table 1. Experimental Results Summary

Experiments				Metrics		
CBAM	Balanced Sampler	Loss	Dataset Scale	Loss	Min AUC	Mean AUC
		BCE	Half	0.458	0.55	0.678
	√	BCE	Half	0.553	0.76	0.838
√	√	BCE	Half	0.387	0.81	0.844
√	√	wBCE	Half	0.343	0.83	0.885
√	√	wFocal	Half	0.367	0.87	0.892
√	√	wFocal	Full	0.302	0.90	0.916

3.2 Inference and Submission

The provided TST dataset included 120 image files for testing. With our pre-processing pipeline, the TST data were cropped according to the provided Mask 1 and Mask 2 to generate 240 laterality-reduction image files. After prediction by the trained model, the results were rearranged so that both lateralities of one patient were combined again according to the requirement and saved as the .txt file to be submitted.

As different techniques were applied, thus generating different results, some results were ensembled in order to generate better results. Ensembling results of weighted binary cross-entropy loss and weighted focal loss gave the best mean AUC. Test time augmentation was also attempted, and although it produced the best minimum AUC, it

did not have the best mean AUC. A detailed description of our submissions is as follows:

For submission ID 67838, the technique used was cbam + balanced sampler + wBCE, number of epochs was 60, and the best model with validation mean AUC of 0.916 was saved and used. The mean AUC obtained on the TST dataset was 0.872, with min AUC of 0.810.

For submission ID 67839, the technique used was cbam + balanced sampler + wFocal, number of epochs was 60, and the best model with validation mean AUC of 0.918 was saved and used. The mean AUC obtained on the TST dataset was 0.874, with min AUC of 0.809.

For submission ID 67920, the technique used was cbam + balanced sampler + Focal, number of epochs was 48, and the best model with validation mean AUC of 0.907 was saved and submitted. The mean AUC obtained on the TST dataset was 0.832, with min AUC of 0.779.

For submission ID 67921, the technique used was cbam + balanced sampler + BCE, number of epochs was 48, and the best model with validation mean AUC of 0.86 was saved and submitted. The mean AUC obtained on the TST dataset was 0.737, with min AUC of 0.708.

For submission ID 67950, the submitted results were a combination of submission IDs 67838 and 67839. A mean AUC of 0.875 was achieved.

4 Discussion and Conclusion

In an effort to provide a CNN solution for a multi-binary classification task of tuberculosis findings, we proposed a laterality-reduction 3D CBAM Resnet. As severe class imbalance exists in the dataset provided, we tried several techniques to improve the model performance. First, with proper usage of both provided masks, each side of the lungs was cropped, masked, and rearranged so that laterality could be neglected. By cropping each side of the lungs, task number was reduced from six binary classifications to three, but the size of datasets doubled. Balanced sampler in each batch sampler was also used in this study to address the data imbalance problem. CBAM was used to add an attention mechanism in each block of the Resnet to further improve the performance of the CNN. Modified binary focal loss was also realized in the PyTorch framework to allow the network to focus on more difficult examples. Using all the aforementioned techniques, we achieved a mean AUC of 0.875 in the evaluation of the test dataset, and placed second in this competition.

5 Perspectives for Future Work

In this study, we only tested Resnet-based CNN architecture, as there was a limited timeframe and as the 3D dataset-based CNN was slow to train. In the future, more CNN architectures should be tested, such as 3D Resnet 50, 3D Resnet 101, 3D Densenet [11], 3D Efficientnet [12], etc. Besides, even with our best performing model, the overfit still

existed during training. While this was mostly due to the limited training dataset, additional image augmentation techniques, such as non-linear transformation, contrast random adjusting, channel shuffling, etc. could be tested in the future to obtain even better results. Additionally, because we did not perform the k-fold cross-validation, the training and validation dataset used in this study contained some bias in the distribution of the category. In the future, at least a 5-fold cross-validation will be performed, and the results will be ensembled to form the final model.

Reference

1. Kozlovski, S., Liauchuk, V., Dicente Cid, Y., Tarasau, A., Kovalev, V., Müller, H.: Overview of ImageCLEF tuberculosis 2020 - automatic CT-based report generation. In: CLEF2020 Working Notes. CEUR Workshop Proceedings, Thessaloniki, Greece, CEUR-WS.org <http://ceur-ws.org> (September 22-25, 2020).
2. Ionescu, B., Müller, H., Peteri, R., Abacha, A.B., Datla, V., Hasan, S.A., Demner-Fushman, D., Kozlovski, S., Liauchuk, V., Cid, Y.D., Kovalev, V., Pelka, O., Friedrich, C.M., de Herrera, A.G.S., Ninh, V.T., Le, T.K., Zhou, L., Piras, L., Riegler, M., Halvorsen, P., Tran, M.T., Lux, M., Gurrin, C., Dang-Nguyen, D.T., Chamberlain, J., Clark, A., Campello, A., Fichou, D., Berari, R., Brie, P., Dogariu, M., Stefan, L.D., Constantin, M.G.: Overview of the ImageCLEF 2020: Multimedia retrieval in medical, lifelogging, nature, and internet applications. In: Experimental IR Meets Multilinguality, Multimodality, and Interaction. Volume 12260 of Proceedings of the 11th International Conference of the CLEF Association (CLEF 2020), Thessaloniki, Greece, LNCS Lecture Notes in Computer Science, Springer (September 22-25 2020)
3. Dicente Cid, Y., Jiménez del Toro, O.A., Depeursinge, A., Müller, H.: Efficient and fully automatic segmentation of the lungs in CT volumes. In Goksel, O., Jimenez del Toro, O.A., Foncubierta-Rodriguez, A., Müller, H., eds.: Proceedings of the VISCERAL Anatomy Grand Challenge at the 2015 IEEE ISBI. CEUR Workshop Proceedings, CEUR-WS.org <<http://ceur-ws.org>> (May 2015) 31–35
4. Liauchuk, V., Kovalev, V.: Imageclef 2017: Supervoxels and co-occurrence for tuberculosis CT image classification. In: CLEF2017 Working Notes. CEUR Workshop Proceedings, Dublin, Ireland, CEUR-WS.org <<http://ceur-ws.org>> (September 11-14, 2017)
5. Brett, M., Hanke M., Markiewicz C., Côté M.-A., McCarthy P., and Cheng C., [nipy/nibabel: 2.3.3 Zenodo.](https://doi.org/10.21203/rs.3.rs-1000000), (2019)
6. OpenCV: Image Thresholding. https://docs.opencv.org/master/d7/d4d/tutorial_py_thresholding.html
7. He K., Zhang X., Ren S., Sun J.: Deep Residual Learning for Image Recognition. In CVPR. (2016)
8. Woo S., Park J., Lee J.Y., Kweon I.S.: CBAM: Convolutional Block Attention Module. ECCV. (2018)
9. Imbalanced Dataset Sampler. <https://github.com/ufoym/imbalanced-dataset-sampler>
10. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: ICCV. (2017)
11. Huang G., Liu Z., van der Maaten L., Weinberger K.Q.: Densely Connected Convolutional Networks. CVPR. (2017)
12. Tan M., Le Q.V.: EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. ICML. (2019)