ImageCLEF 2018 Tuberculosis Task: Ensemble of 3D CNNs with Multiple Inputs for Tuberculosis Type Classification

Adam Ishay[1] and Oge Marques[2]

Department of Computer and Electrical Engineering and Computer Science, Florida
Atlantic University, 33431 Boca Raton FL
{aishay,omarques}@fau.edu

Abstract. Convolutional neural networks have achieved state-of-the-art results in general image classification tasks and have shown success in several applications within the medical imaging domain. In this paper, we apply a 3D convolutional neural network (CNN) to a dataset of tuberculosis-positive computed tomography (CT) scans to solve the task of automatically categorizing each tuberculosis (TB) case into one of five possible TB types in the context of the ImageCLEFtuberculosis 2018 challenge. The size of the volumetric scans poses unique constraints on the network and the training process. The CT volumes are segmented with the provided masks, which are further pre-processed prior to training our model. Our best run ranked 2^{nd} with an unweighted Cohen's Kappa of 0.1736 and an accuracy of 35.33%.

Keywords: 3D-CNN · Deep Learning · Tuberculosis · Image Classification · Medical Imaging

1 Introduction

For the second year, ImageCLEF [5] has proposed the ImageCLEFtuberculosis 2018 task [3], in efforts to reduce the time required and cost of medical image analysis. This year there are three subtasks: multi-drug resistance (MDR) detection, tuberculosis type classification, and severity scoring. The goal of the MDR task is to predict probabilities of a patient having a drug-resistant form of tuberculosis. The third task, severity scoring, aims at predicting a severity score from 1 (very bad) to 5 (very good). Finally, the task that this paper addresses is the tuberculosis type classification. We are tasked with classifying the type of tuberculosis, given a positive image. These types are: (1) Infiltrative, (2) Focal, (3) Tuberculoma, (4) Miliary, and (5) Fibro-cavernous.

Deep learning approaches have been shown to be successful on a large variety of computer vision and image analysis tasks [7]. Deep learning and CNNs in particular have now broadly been applied to medical imaging [8]. We apply a deep 3D CNN to the medical image dataset for classification.

2 Data Pre-processing

The training set provided by the ImageCLEF organizers consisted of patient chest CT scans of five different types of TB along with their labels. Often patients had multiple scans and all scans of the same patient were of the same type. There were 228, 210, 100, 79, and 60 patients belonging to Infiltrative, Focal, Tuberculoma, Miliary, and Fibro-cavernous types, respectively. The dataset totaled 677 patients with 1008 scans (Table 1). Each scan consists of approximately 100 512×512 slices. The depth of each scan varies and was changed to a constant number of slices.

Table 1. Train and test distribution of the dataset.

Class	Train Patients (Scans)	Test Patients (Scans)
Infiltrative (1)	228 (376)	89 (176)
Focal (2)	210 (273)	80 (115)
Tuberculoma (3)	100(154)	60 (86)
Miliary (4)	79(106)	50 (71)
Fibro-cavernous (5)	60 (99)	38 (57)
Total	677 (1008)	317 (505)

The pre-processing stage consisted of 7 steps (Figure 1). The supplied masks [4] were applied to the original scans to segment the lungs. The distance between slices along with the resolution of each slice varied among scans. For easier training, the images were resampled to an isotropic resolution of 1×1×1 mm. After this step, the scans were roughly 300×300×300 in size. All scans were then cut to remove the excess zeros in the background from the mask. The voxel values were clipped between -1000 and 400, and normalized between 0 and 1. The values outside of this range are not useful. Then, the largest lungs were used to find the new width and height to which all images would be padded, with the common background voxel value in the scans. The scans were also padded in the depth dimension to the depth of the largest scan. Next, the mean pixel was calculated and subtracted from all scans to zero-center the data for better training. Finally, the resulting scans were resized to reduce the data to a more reasonable size for the network. Using this process, two datasets of different sized images were created (see Figure 2). The purpose of this was to combine two different networks to predict the label. The batch sizes used are a function of the size of the input and the architecture of the network. In the networks used, most of the memory consumption was due to the first few layers of the network, since in these layers the images were still large.

The two datasets each had a respective train/validation split of 80/20. Initially this split was done by scan and validation accuracy was relatively high. However, when submitting results on the test set the accuracy was much lower and close to random. This was thought to be at least partially due to the method for splitting. Because some patients had multiple scans, there were scans in the

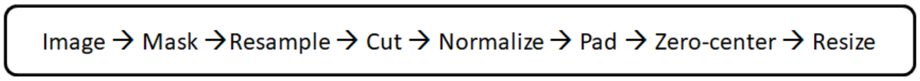
Image → Mask → Resample → Cut → Normalize → Pad → Zero-center → Resize

Fig. 1. The 7 steps that made up the pipeline for processing the scans.

train and validation set from the same patient. Upon visual inspection, scans from the same patient were indeed similar (see Figure 3).

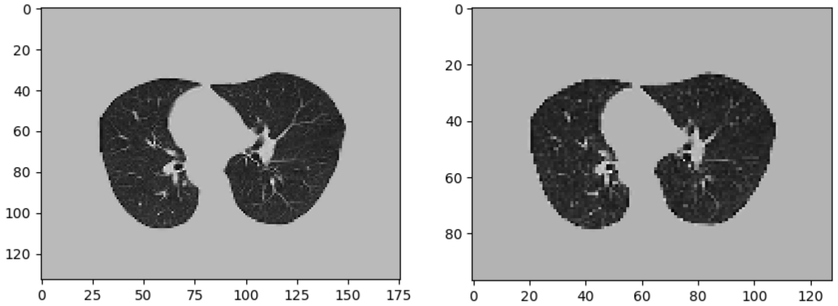

Fig. 2. The same scan processed into two different sizes which were used in the ensemble. The bigger scan is size (176×133×195) and the smaller one is (128×97×142).

3 Methodology

The trained models were 3D convolutional neural networks using the software library Keras [2] with Tensorflow [1] backend. We opted for 3D convolutions because they naturally capture the 3D nature of the scans. We trained two networks, one for each dataset created in the pre-processing stage. The combination of the two networks achieved better results than either of them alone. To alleviate the class imbalance problem shown in Table 1, oversampling was used during the training phase. Classes three, four, and five were oversampled to approximately match the test distribution. This meant that a full epoch of training was reached when roughly 900 patients (677 + ∼200) were processed by the network.

Each network (Figure 4) had five convolution layers with rectified linear unit (ReLU) activations, each with a following batch normalization and max pooling with dropout layer. These led to two fully connected layers, each with batch normalization and dropout. Finally, these activations went through a softmax layer, which output a tensor of size five, for each category. Categorical cross-entropy was used as the loss function, and Adam [6] was used for optimization.

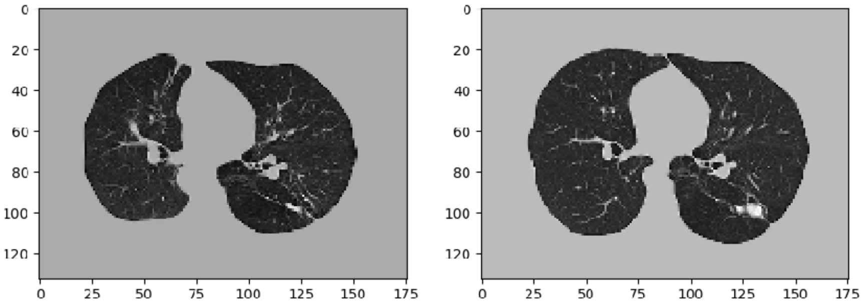

Fig. 3. Slices of scans of the same patient taken 7 years apart. It is easy to see the similarities visually, and this was reason to split the dataset by patient instead of scan to avoid potential overfitting.

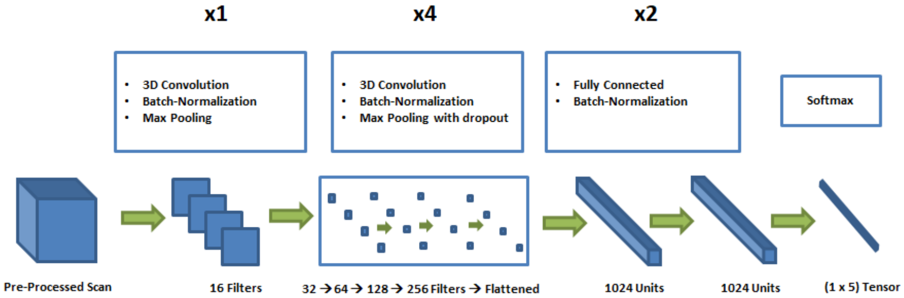

Fig. 4. Diagram of the 3D CNNs used for training.

One of the restrictive parts of training this model was the batch size. Such small batches make it harder to converge.

Our most successful model was the combination of the two best models, which had inputs of different size image volumes. The outcome of this ensemble were 5 probabilities, one for each class. The probabilities were summed across the two models, and then this vector was iteratively scaled by a weight vector which was calculated from the class distribution. This resulted in output labels which more closely matched the data distribution. This combination of networks shown in Figure 4 was used for predicting the test labels.

4 Results

Only runs for the tuberculosis type subtask 2 were submitted. Our initial submissions accuracies were barely better than random chance (≈28%). After combining models and weighting probabilities, the accuracy and kappa score did im-

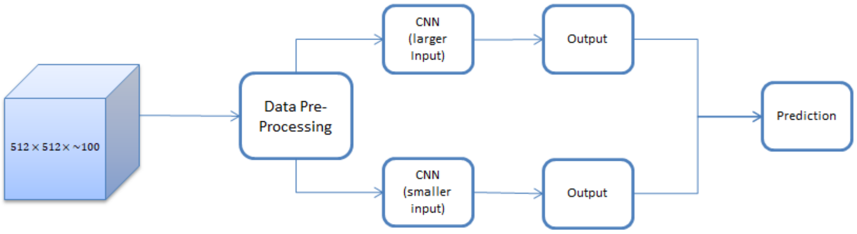

Fig. 5. Diagram of the pipeline for predicting the labels of the test scans.

Table 2. Top six rankings from the ImageCLEFtuberculosis [3] 2018 tuberculosis type task.

Model (Group Name)	Kappa Score	Accuracy	ranking
run_TBdescs2_zparts3_thrprob50_rf150 (UIIP_BioMed)	0.2312	0.4227	1
m4_weighted (fau_ml4cv)	**0.1736**	**0.3533**	**2**
AllFeats_std_euclidean_TST (MedGIFT)	0.1706	0.3849	3
Riesz_AllCols_euclidean_TST (MedGIFT)	0.1674	0.3849	4
Run-02-Mohan-RF-F20I1500S20-317 (VISTA@UEvora)	0.1664	0.3785	5
m3_weighted (fau_ml4cv)	0.1655	0.3438	6

prove. Our best run (indicated in **bold** in Table 2) ranked second in unweighted kappa coefficient, but tenth in accuracy.

5 Conclusion

This paper applies a 3D CNN to pre-processed CT scans of the lungs. The question of whether a CNN can extract the information necessary for labeling types of TB remains open. Making predictions on image data alone has proved a challenging problem. The large size of the images and small size and class imbalance of the datasets are characteristic of medical imaging tasks. In this analysis, batch sizes were restricted to sizes of five and fourteen samples for the two networks used. A feasible way of effectively training with a much larger batch size is to accumulate the gradients of each batch and only update the weights of the network after storing a sufficient number of batches gradients. The average of the gradients for each batch can be used to update the weights. This allows for effectively training on larger batch sizes, circumventing memory problems.

References

1. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J.,

Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X.: TensorFlow: Large-scale machine learning on heterogeneous systems (2015), https://www.tensorflow.org/, software available from tensorflow.org

2. Chollet, F., et al.: Keras. https://keras.io (2015)
3. Dicente Cid, Y., Liauchuk, V., Kovalev, V., , Müller, H.: Overview of ImageCLEF-tuberculosis 2018 - detecting multi-drug resistance, classifying tuberculosis type, and assessing severity score. In: CLEF2018 Working Notes. CEUR Workshop Proceedings, CEUR-WS.org <http://ceur-ws.org>, Avignon, France (September 10-14 2018)
4. 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., Jiménez del Toro, O.A., Foncubierta-Rodríguez, A., Müller, H. (eds.) Proceedings of the VISCERAL Anatomy Grand Challenge at the 2015 IEEE ISBI. pp. 31–35. CEUR Workshop Proceedings, CEUR-WS (May 2015)
5. Ionescu, B., Müller, H., Villegas, M., de Herrera, A.G.S., Eickhoff, C., Andrearczyk, V., Cid, Y.D., Liauchuk, V., Kovalev, V., Hasan, S.A., Ling, Y., Farri, O., Liu, J., Lungren, M., Dang-Nguyen, D.T., Piras, L., Riegler, M., Zhou, L., Lux, M., Gurrin, C.: Overview of ImageCLEF 2018: Challenges, datasets and evaluation. In: Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Ninth International Conference of the CLEF Association (CLEF 2018), LNCS Lecture Notes in Computer Science, Springer, Avignon, France (September 10-14 2018)
6. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
7. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436 (2015)
8. Litjens, G., Kooi, T., Bejnordi, B.E., Setio, A.A.A., Ciompi, F., Ghafoorian, M., van der Laak, J.A., van Ginneken, B., Sánchez, C.I.: A survey on deep learning in medical image analysis. Medical image analysis **42**, 60–88 (2017)