Revealing Lung Affections from CTs. A Comparative Analysis of Various Deep Learning Approaches for Dealing with Volumetric Data

Radu Miron[1,2] , Cosmin Moisii[1,2] , Mihaela Elena Breaban ✉[1,2]

[1] SenticLab, Iasi, Romania
[2] Faculty of Computer Science, "Alexandru Ioan Cuza" University of Iasi, Romania
pmihaela@info.uaic.ro

Abstract. The paper presents and comparatively analyses several deep learning approaches to automatically detect tuberculosis related lesions in lung CTs, in the context of the ImageClef 2020 Tuberculosis task. Three classes of methods, different with respect to the way the volumetric data is given as input to neural network-based classifiers are discussed and evaluated. All these come with a rich experimental analysis comprising a variety of neural network architectures, various segmentation algorithms and data augmentation schemes. The reported work belongs to the SenticLab.UAIC team, which obtained the best results in the competition.

1 Introduction

Medical imaging technologies like Computer Tomography (CT) and Magnetic Resonance (MR) produce high volumes of data in the form of volumetric images. The richness of information they provide is essential to correct diagnosis but brings at the same time new challenges, both for manual/human and automatic/machine processing: these are not only about the size of the produced data but also about the complexity of the diagnosis process itself. With respect to automated diagnosis, the volumetric images, which can be seen both as matrices of pixels/voxels or series of 2D images (usually called slices), produced high effervescence in the deep learning research community, triggering a variety of new architectures and approaches.

The current paper makes use of deep learning to automatically detect tuberculosis and related affections in lung CTs, in the context of the ImageClef Tuberculosis task [1, 2]. We investigate three types of approaches, different with respect to the way the volumetric data is given as input to neural network-based classifiers. One type, popular among the participants in the previous year competition [3], is based on reducing the volumetric image to a small set of 2D projections. Obviously, this approach consistently reduces the size of the data to be processed by the classifier but inherently may lose important information. The second type exploits the whole data matrix by using 3D convolutions or by fusing the information from the slices. The third type, which was ranked as the winner of the 2020 evaluation session, consists in moving the decision layer from the whole volume of data to the slice level. All these three different approaches come with a variety of neural network architectures, various segmentation algorithms and data augmentation schemes. The work reported stays behind the SenticLab.UAIC team, obtaining the best results in the competition[3] [1].

The paper is structured as follows. Section 2 describes the challenge and the dataset. Section 3 describes the approaches developed based on reducing the volumetric image to 2D projections, starting with the previous year winning approach reported in [4], which we further enhanced to address the 2020 tasks. Section 4 presents the approaches we used to exploit the whole volumetric information. Section 5 describes the architectures used to process the information at slice level and the heuristics used to produce the diagnosis report at the CT level. Because of the large number of approaches we evaluated, some of them abandoned earlier (not submitted in the competition) due to poor results on our local validation data, we report and discuss performance results immediately after each method description. Section 6 summarises the results for the best approaches that were evaluated on the blind test set in the competition and discusses comparatively the performance of the three classes of methods. Section 7 concludes the paper.

2 ImageClef Tuberculosis: tasks, data, evaluation

The challenge in the 2020 ImageClef Tuberculosis competition is the automatic detection of tuberculosis and related lesion types in CTs. The CT report to be generated must contain 3 binary labels for each lung, indicating the presence of TB lesions in general, the presence of pleurisy and caverns in particular.

The training dataset consists of 283 CTs. All CTs present at least one lung affected, 19 have pleurisy and 126 caverns. Because we split each CT into left/right lungs, this translates to 566 inputs, 444 affected, 21 with pleurisy and 145 with caverns.

The task is therefore a multi-binary classification problem, with three target labels per lung. For each target label the AUC is computed and the ranking is done on a test set, first by computing the average AUC and then by the minimum AUC over the 3 target labels.

We split the data into train/validation in the same fashion as [4] setting apart every 4th input into the validation set and use this configuration throughout the competition.

[3] https://www.imageclef.org/2020/medical/tuberculosis/

3 Squeezing Volumetric Data: 2D Projections

The 3D matrix representing the volumetric image can be reduced to simpler 2D representations by traversing it in each of its three dimensions and computing statistics on numeric vectors. In the case of lungs CTs, a segmentation algorithm is firstly applied to detect the lungs and eliminate the other parts in the CT. Further, we used the method proposed in [4], where the mean, the maximum and the standard deviation is computed on each direction, generating three 2D matrices which can be interpreted as an RGB image (a single 2D image with 3 channels). All the processing steps described in [4] are kept: mask erosion, increasing the voxels intensity in the CT by 1024HU, dividing the mean values and standard deviations values (red and blue channels) by their maximum, dividing the maximum values (green channel) by 1500. At the end, for each lung we have a set of three 2D RGB images, each image corresponding to one of the three dimensions of the 3D matrix.

3.1 The Impact of Segmentation

The first step behind all our approaches is image segmentation, with the aim of identifying and isolating each lung in the volumetric image. Because the performance of further processing is greatly influenced by the quality of segmentation (especially in the case of the 2D projection approach where the projections take into account the entire volume), we tested several segmentation methods. The organizers provided for all patients two versions of automatically extracted masks of the lungs: one which relies only on anatomical assumptions [5], and one based on non-rigid registration [6]. Additionally, we used *U-net(R231)* and *U-net(LTRCLobes)* which were pre-trained for lung segmentation on large and diverse datasets [7] [4]. Our experiments show that the first technique based on anatomical assumptions behaves much like region growing not being able to catch holes or necrotic tissue in lungs, the second technique manages to capture necrotic tissue while the ones based on U-net include airpockets, tumors and effusions. The flow of the dataset creation together with some projections corresponding to several segmentation techniques can be seen in fig. 1.

Feeding a VGG neural network [8] with 2D projections obtained on the segmented volumetric image, the average AUC scores obtained on our validation set indicate the registration based method to give the best performance (**AUC=0.693**) followed at small distance by *U-net(R231)* (**AUC=0.674**) and *U-net(LTRCLobes)* (**AUC=0.668**), but consistently surpassing the anatomy-based method (**AUC=0.580**).

Consequently, all our further experiments use the segmentation provided by non-rigid registration [6].

[4] https://github.com/JoHof/lungmask

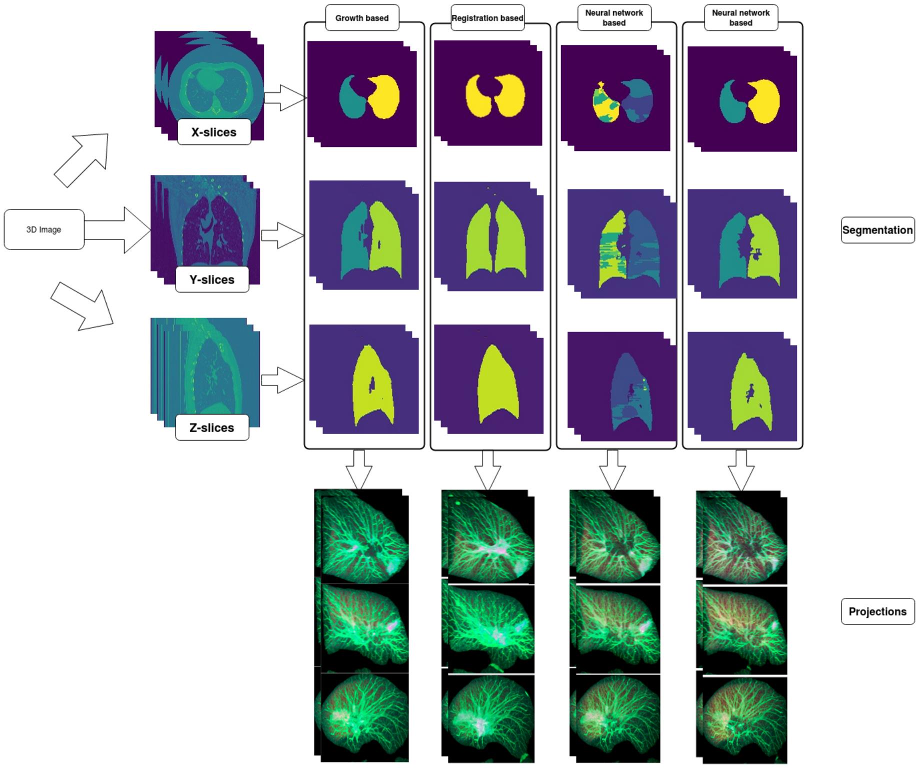

Fig. 1. Projections dataset creation flow using 4 segmentation variants (in order: growth-based, registration-based, unetLTRCLobes and unetR231). Although the 4 types of projections resulted look only slightly different, the difference in classification scores is significant.

3.2 Data Augmentation

The images go through a series of augmentations, each with a certain probability of being applied, including: horizontal and vertical flipping, small degrees of rotations, blurring, added gaussian noise, distortions, random cropping, and changing different values of hue, saturation or brightness. We used the Albumentations [5] library for most of these augmentations.

3.3 The 2D Approach with Preprocessing (*PreProcProj*)

In an effort to improve over the last year result, we used the pre-processing provided in [9], with the aim to eliminate the small vessels from the projection, thus making the affected area more obvious. We adopted all the pre-processing

[5] https://github.com/albumentations-team/albumentations

steps that the authors mention, except the regional maxima calculation. Figure 2 illustrates the difference between projections with and without further preprocessing.

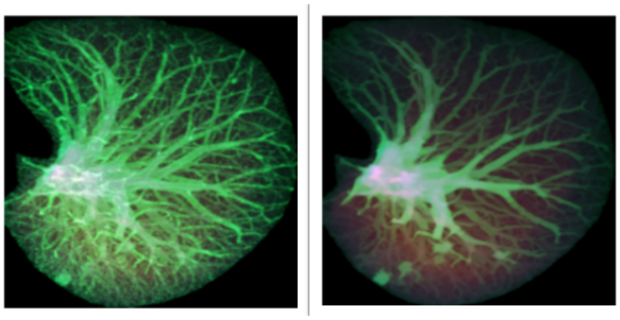

Fig. 2. Comparison between projection without(left) and with pre-processing(right)

For training we chose AlexNet[10]. The input consists of the three projections of a volume, on each axis. After extracting features from each projection with AlexNet, we concatenate all the features, feed them into a linear layer and predict probabilities for a lung to have affections, caverns, pleurisy or be healthy. With this approach we scored an AUC of 0.793 on the test set.

3.4 The 2D Approach Scoring the Best (*ResNet50Proj*)

We further tried different variants of Resnet[11] and SqueezeNet[12].

We extracted the lungs using the registration-based segmentations, computed all 3 projections, split by lung side and processed with the augmentation we listed in section 3.2. We trained the networks and aggregated the results on all 3 projections and computed the mean score to obtain the final results back at CT level. We tried different approaches in aggregating the results including training a small neural network, but found the simpler mean aggregation to give the highest score. We thus obtained our highest score in the 2D approach using a resnet-50 network [11] pretrained on Imagenet[13] with an AUC on our hold-out validation set of 0.877; however this result was not submitted. Our first submission to the competition was a resnet34 model with no augmentations which obtained on the hold-out validation set and on the test set the same AUC score of 0.825. In Figure 3 we can see the AUC progress on different models we tried.

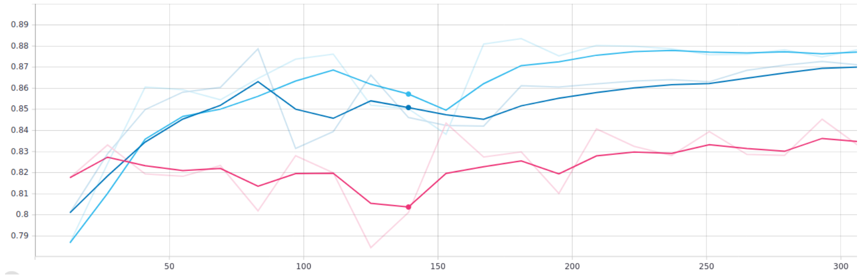

Fig. 3. The AUC score progress on the hold-out validation set for different resnet models. Pink: resnet34 with no augmentations, DarkBlue: resnet34 with augmentations, LightBlue: resnet50 with augmentations

4 Exploiting Volumetric Data as a Whole

4.1 3D Convolutions

In an attempt to make use of the whole volume at once, we used SqueezeNet in a 3D version, based on the implementation found in the repository [6]. In order to work with volumes of different sizes, we used batch size equal to 1. To handle volumes with a large number of slices, we used Apex[7] library for reducing the burden on our GPU. In a preliminary experiment we considered only the case *affected vs. not affected*. We noticed the bad results during the training: after some epochs the prediction scores stagnated, for all volumes, between 0.4 and 0.6. We concluded that 3D convolutions are not able to capture the important information on our small training set of volumetric images.

4.2 Slices fusion

In our attempt to associate the entire volumetric image to a label, we constructed a hybrid approach. We fed the volume slice by slice into a convolutional neural network, fused the resulted feature maps at channel level and continued with another small convolutional network into a prediction. The initial convolutional neural network is composed from the encoder part of a U-net[14] architecture which was pretrained on a segmentation task at the end of which we applied a squeeze connection to reduce the number of channels, fused the resulting feature maps so that the slices processed in parallel by the CNN would now be treated as channels of a single input, then applied a resnet-like small network to compile the features into a label. We no longer use the masks to extract the lungs but instead use a simple threshold based segmentation to compute the boundaries of the body and crop out the space around the it. We again split by lung side and used only horizontal flip as a preprocessing. The resulting volume is resized to

[6] https://github.com/okankop/Efficient-3DCNNs
[7] https://github.com/NVIDIA/apex

the fixed size of (128, 256, 256) The network could then be fed images in batches multiple of 128 representing the slices of a volume.

To make maximum use of the GPU memory, we used the Apex library to train using mixed precision, in a distributed manner on 2 GPUs. We could fit 2 times 128 images into the memory corresponding to 2 volumes.

The approach turned out to be cumbersome. The time to process an epoch was relatively high and the convergence of the network seemed slow. After 2 days of training we decided to stop and the network reached an AUC of around 0.6 on the hold-out validation set.

5 Sequencing Volumetric Data: a Slice by Slice Classification Approach

Having a closer look at the training set, one can observe that usually the lesions on the lungs are located only on a small number of slices from the whole volume. A natural idea is to try a 2D model that could differentiate between healthy lung slices and lung slices with lesions (caverns, pleurisy and affections) and construct the CT report based on the findings at slice level. For this purpose we need training data labeled at slice level and not CT level.

The first approach was to try to automatically detect the slices presenting lesions in the training set, using a lung nodule detector[8] constructed by the winners of a challenge in cancerous nodules detection. The results were bad, the model not being able to recognize the slices showing caverns although these correspond to big, obvious regions.

Therefore, we started to manually select from each volume of the training set the slices with lesions. We actually found that this was not as time-consuming as we initially thought, by processing only the volumes labeled with lesions, and it definitely was worth the effort, as the increase in performance shows. The caverns are usually big and obvious and the affections are either nodules or dusty lungs images (which may indicate pneumonia), with very rare cases of pneumotorax (the lung disappearing due to the outbreak of a cavern). There are many cases when these lesions appear on a very small number of slices and thus, the two approaches described in sections 3 and 4 might have not been able to reveal them.

5.1 InceptionNet

In our first tries using the slice by slice approach, we used *InceptionNet version 3*[15]. We used the annotated data in different ways. Transforming each slice of a volume into a picture, resizing each image to a size of 299 × 299, cutting the picture in half to obtain the two lungs and using vertical flip for all the pictures which are either affected, with caverns or with pleurisy, are the data pre-processing steps for our first attempt using this approach. This approach

[8] https://github.com/BCV-Uniandes/LungCancerDiagnosis-pytorch

does not use the provided segmentation masks at all. We only used 4 labels as output: *affected, caverns, ok, pleurisy.*

As input for the neural network we used several versions, having all images as 3-channel images:

a) *NaiveInception.* We added a linear layer on top of the last adaptive average pooling layer of the architecture, keeping the original InceptionNet weights freezed. With this approach we scored 0.86 AUC score on the test set, surpassing this way the best approach based on 2D projections.

b) *ThresholdInception.* The other approaches consist in using some other preprocessing steps. This time, when creating the photos from the 3D volume, we used a Window Width and Window Level equal to 1500, -500 respectively. This way we improved the results to 0.887 mean AUC and to 0.82 min AUC on the test set.

c) *TwoPicInception.* One other approach consists in mimicking the protocol a doctor has to follow in order to decide affections(including caverns) and pleurisy. In order to see the affections more clearly, a doctor uses Window Width and Window Level equal to 1500, -500 respectively, whereas for better visualisation of pleurisy, a doctor looks at pictures with Window Width and Window Level equal to 350, 50 respectively. With this thresholding, the liquid surrounding the pleura becomes more observable. Figure 4 top shows the differences between the two pictures.

In order to use information from 2 pictures during training, we used two InceptionNet modules, with trainable parameters and concatenated the two outputs of the adaptive average pooling layer. The decision was made based on the output of the last linear layer applied on the concatenation discussed above. With this approach we scored 0.89 mean AUC on the test set.

d) *AttentionInception.*We wanted to gain insight into how accurate the methods can be. We tried to check the predictions produced by the methods proposed and discovered that the models found in a big proportion correct slices of the volumes which contained certain affections. In order to work on the explainability of our model, we modified the structure of ThresholdInception, using the idea from [16], introducing an attention mechanism. Instead of feeding the output of the last pooling layer into the linear layer, we used dot product attention [17]. We computed similarity scores between the output of the pooling layer and three different convolutional layers in the architecture. After using the compatibility scores as weights for the features extracted by the three layers, we concatenated the new features. Using a linear layer on top, we predicted scores for the 4 categories. After plotting the attention we noticed that the attention on the first layer selected highlighted the whole lung area - supporting the idea that we don't need the segmentation masks, whereas the second layer of attention highlighted affections on the lungs. With this approach we scored 0.85 mean AUC. We believe this lower performance is due to the fact that the attention on the last layer was not good. Checking the visualisation for that layer we noticed useless areas highlighted (Fig. 4, bottom). Because of the limit imposed on the number of submissions, we stopped investigating this direction.

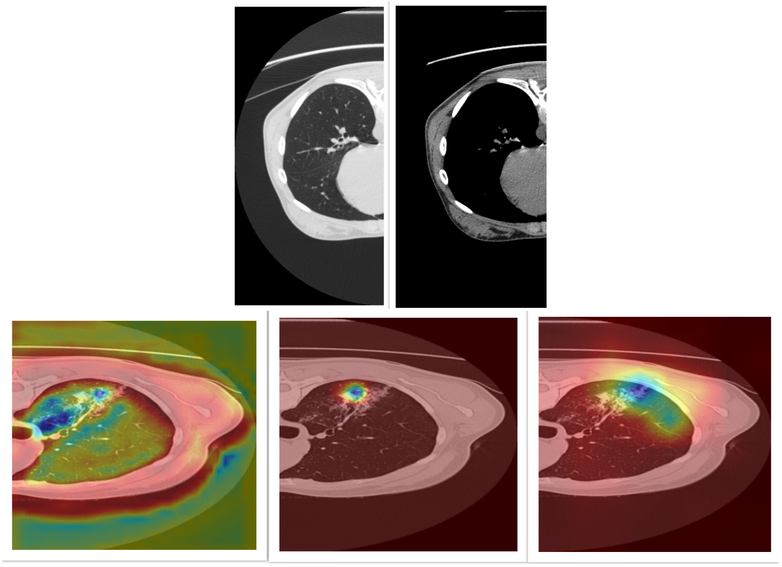

Fig. 4. Top: Comparison between different threshold values for the HU units Bottom: Attention visualization for the three layers from InceptionNet

For all the methods above based on InceptionNet, training was performed for 30 epochs on one GPU Nvidia RTX 2070 with 8GB of memory, using Stochastic gradient descent optimizer. We divided the learning rate at each 10 epochs by 10 and used binary cross-entropy as loss function.

In order to establish the diagnosis for a volume we applied the following heuristic: we applied the inference step on all the pictures/slices from the volume and for each of the possible classes we took the maximum score encountered; if only one slice was found with an affection score higher than 0.8, then we divided the score of affected by 2.

5.2 EfficientNet

In an effort to use a powerful, yet small footprint network, in our last approaches we used efficientnet[18], specifically the b4 variant which has only 19M parameters but reaches top 1 accuracy of 82,6% on Imagenet. We used a Pytorch implementation pre-trained on Imagenet[13].

The preprocessing we used here is similar to the ones we used before and took place at run-time on load. We applied the registration-based mask per slice based on a threshold to crop the body and remove much of the surrounding space, split the lungs into left/right (just by using splitting the image in half) and applied the same series of augmentations as in the previous approaches. The split and cropped image has dimension 256 × 256 and after randomly cropping it reduces to 224 × 224. For ease of working we also kept the size of the volumetric image depth to a fixed 128 slices per volume.

If otherwise specified for this approaches as for the others we used a window level of -500 and range of 1500 corresponding to the most common values used in areas of acute differing attenuation values (example: lungs) where air and vessels will sit side by side.

As input we tried several options, all of them maintaining 3 channels per image:

a) Micro-volumes (*MicroVolSlice*): The importance of volumetric data is evident. This seemed especially apparent when we try to manually identify caverns which can present as rounded or irregularly shaped black centers surrounded by a white contoure. The caverns can range in size from small with a thin contoure line to large with thick and diffuse borders. The small caverns we found especially hard to identify as it can be confused with a section of a larger blood vessel. As untrained individuals, to eliminate the confusion we traced the potential cavern a few slices up or down to verify if it continues into a vessel or forms a pathology. To try and mitigate this type of confusion in a model we composed the 3 channels of the image from 3 consecutive (or equidistant) slices. In case the slice is at the beginning or end of the sequence we simply duplicated it to fill the channels.

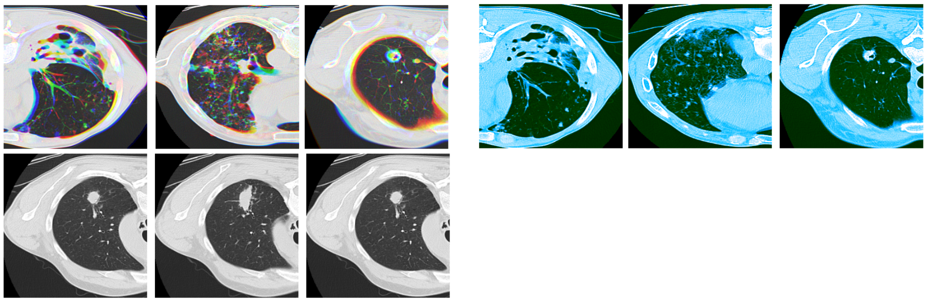

Fig. 5. Top left: sample of micro-volume images. Top right: Sample of false-color images. Bottom: Sample of "naive" images.

b) False-color (*FalseColorlSlice*): To make use of the entire range of values of an MRI image we established 3 intervals in the Hounsfield units range to correspond to the 3 channels of an image. The window size and level used [9] are (1500, -500) corresponding to the usual values used for lung imaging, (350, 40) called narrow window (used when examining areas of similar attenuation, for example, soft tissue) and (500, -600) a narrower window of the usual values for lung imaging in an attempt to retain more information around the values corresponding to blood vessels and soft tissues. The result with this method however was not submitted to the site as the result on the hold-out set was poorer than the others.

[9] https://radiopaedia.org/articles/windowing-ct

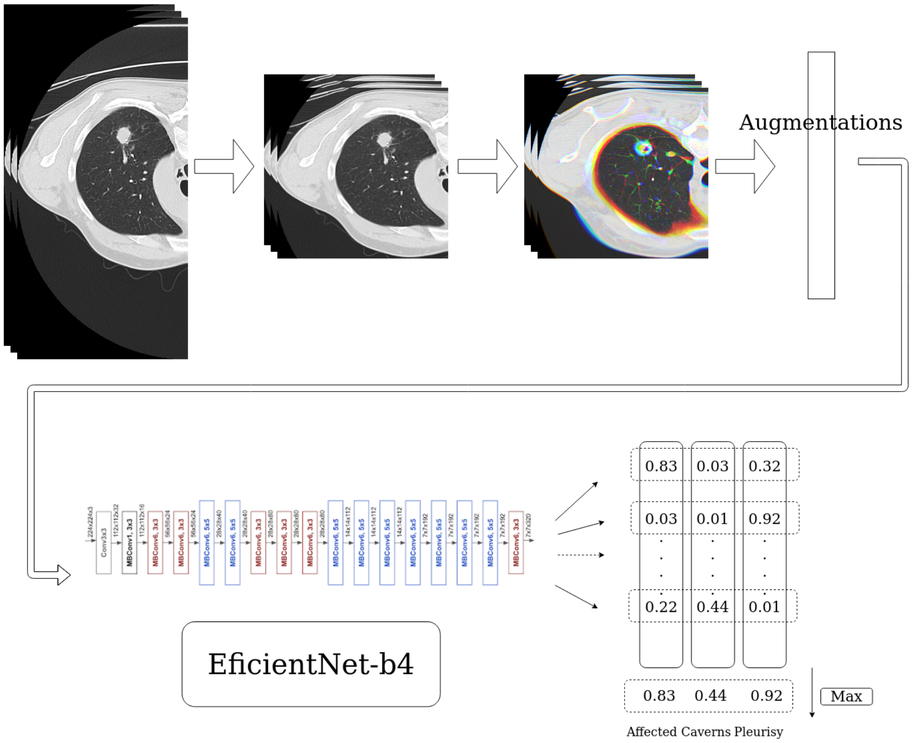

Fig. 6. The schematic of the slice approach (microvolumes). It follows a simple flow. The volumetric image, split by left/right side is cropped using a simple threshold based segmentation, then composed (in this case) to microvolumes, augmented and passed through the model. The output from all the slices of a side of a volume is then aggregated and the max per label selected to compose the final result for a side.

c) Naive (*NaivelSlice*): The image is simply duplicated into the 3 channels.

To compile the final results for each volume we aggregate the individual results per slice and choose the max of each each label across all the slides, which are then used to compute the AUC.

Loss: Cross entropy vs Binary cross entropy : A strange case comes from the fact that using the CrossEntropyLoss (on a multi-label classification) without softmax before the loss (thus assigning a predominant label for each slice) we obtained higher results than using BinaryCrossEntropyLoss. The nature of the results is also very different, the first giving results on the extremes while the latter hovering around 0.5, but both giving decent results around 90% AUC.

Micro-volumes and just repeating the image gave similar results on the test set (92.2% and 92.4% respectively), however the training on the "naive" case was done on 130 epochs on 3 GPUs with batch 56 × 3 whereas the "micro-volumes" case was done on 60 epochs on 2 GPUs with batch 56 × 2. Therefore, the approach with the highest score on our hold out set was c) the simple one which also represented the highest of the submitted scores. Our second highest submitted model is represented by the approach in a) micro-volumes.

For training we used Nvidia RTX 2070 with 8GB of memory. We again, used the Apex library from NVidia to train using mixed precision and DistributedDataParallel with one process per GPU.

6 Comparative results

6.1 Results on the competition test set

Table 1 summarizes the results obtained in the competition on the test set. Submissions were made only for the methods based on 2D projections and the ones based on predictions at slice level; as shown on the hold-out validation data, the attempts to use the whole volume using 3D convolutions or fusing the information at slice level did not obtain good results and therefore were not used in the competition test phase.

Table 1. Results reported on the test set, in the order of submission. The first two entries use 2d projections, while all the others make predictions at slice levels. (CE suffix represents models with CrossEntropyLoss and BCE models with BinaryCrossEntropyLoss

Method	mean AUC	min AUC
ResNet50Proj	0.825	0.766
PreProcProj	0.793	0.703
NaiveInception	0.860	0.772
ThresholdInception	0.887	0.821
AttentionInception	0.853	0.788
TwoPicInception	0.892	0.830
NaiveSliceCE	**0.924**	**0.885**
MicroVolSliceCE	0.922	0.860
NaiveSliceBCE	0.899	0.862

6.2 Discussion

By comparing the results both on our hold-out validation set and on the test set, the following conclusions can be drawn.

- As indicated by the low training accuracy, the approaches using the entire volumetric data as a whole corresponding to the segmented lungs (described in section 4), involving 3D convolutions or slice fusion, seem to be overwhelmed by the amount of parameters to fit and are not able to identify the lesions in cases where these are small or present only on a few slices of the CT, or either converge slowly.

- The 2D approaches based on projections computed over the segmented volume (described in section 3) give (unreasonable) good results, which indicates that simple (normalized) statistics like mean, maximum and standard deviation, when used together, are able to catch important information about the presence of lesions in lung CTs. The quality of segmentation of the lungs is of critical importance in this case, as a bad segmentation may introduce noise into the projections. After obtaining the set of 2D projections, data augmentation increased the generalization capability of the classifier.
- The best approach, surpassing significantly the ones based on 2D projections, exploits all the information present in the segmented volumetric lungs in a slice-wise manner. Instead of predicting the presence of the affection per CT, we predict it for each slice. To obtain the report back at lung level the probabilities over slices are aggregated by extracting the maximum. An important pre-processing step consisted in fixing the window and range levels to specific values used by radiologists when inspecting lung CTs.

7 Conclusions

Volumetric images like CTs and MRIs provide rich information about the internal body structure, necessary in the diagnosis of many affections. With the advancements of neural networks, automatic diagnosis in volumetric images became possible at high precision, useful for prioritizing patients and assisting doctors in final decisions. After a thorough experimental analysis of various architectures, the current paper devised an approach able to produce highly accurate CT reports about the presence of tuberculosis related affections. The method, based on computing predictions at slice level, has, beside high accuracy in predicting lesion type, the advantage of offering more information in terms of localization of the lesions. With a current mean AUC score of 0.924 on test data, its performance can be increased if more data, capturing various cases, is provided in the training phase.

8 Acknowledgements

Our strong belief is that data science without domain knowledge can never reach its full potential. We would like to thank Mirela Iordache, an outstanding radiologist at the Regional Institute of Oncology in Iasi, for sharing her knowledge and valuable insights on the medical niche approached in the competition.

References

1. Serge Kozlovski, Vitali Liauchuk, Yashin Dicente Cid, Aleh Tarasau, Vassili Kovalev, and Henning Müller. Overview of ImageCLEFtuberculosis 2020 - automatic CT-based report generation. In *CLEF2020 Working Notes*, CEUR Workshop Proceedings, Thessaloniki, Greece, September 22-25 2020. CEUR-WS.org <http://ceur-ws.org>.

2. Bogdan Ionescu, Henning Müller, Renaud Péteri, Asma Ben Abacha, Vivek Datla, Sadid A. Hasan, Dina Demner-Fushman, Serge Kozlovski, Vitali Liauchuk, Yashin Dicente Cid, Vassili Kovalev, Obioma Pelka, Christoph M. Friedrich, Alba García Seco de Herrera, Van-Tu Ninh, Tu-Khiem Le, Liting Zhou, Luca Piras, Michael Riegler, Pål Halvorsen, Minh-Triet Tran, Mathias Lux, Cathal Gurrin, Duc-Tien Dang-Nguyen, Jon Chamberlain, Adrian Clark, Antonio Campello, Dimitri Fichou, Raul Berari, Paul Brie, Mihai Dogariu, Liviu Daniel Ştefan, and Mihai Gabriel Constantin. 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, September 22-25 2020. LNCS Lecture Notes in Computer Science, Springer.
3. Yashin Dicente Cid, Vitali Liauchuk, Dzmitri Klimuk, Aleh Tarasau, Vassili Kovalev, and Henning Müller. Overview of imagecleftuberculosis 2019-automatic ct-based report generation and tuberculosis severity assessment. In *CLEF (Working Notes)*, 2019.
4. Vitali Liauchuk. Imageclef 2019: Projection-based ct image analysis for tb severity scoring and ct report generation. In *CLEF (Working Notes)*, 2019.
5. Yashin Dicente Cid, Oscar Alfonso Jiménez del Toro, Adrien Depeursinge, and Henning Müller. Efficient and fully automatic segmentation of the lungs in ct volumes. In Orcun Goksel, Oscar Alfonso Jiménez del Toro, Antonio Foncubierta-Rodríguez, and Henning Müller, editors, *Proceedings of the VISCERAL Anatomy Grand Challenge at the 2015 IEEE ISBI*, CEUR Workshop Proceedings, pages 31–35. CEUR-WS, May 2015.
6. Vitali Liauchuk and Vassili Kovalev. Imageclef 2017: Supervoxels and co-occurrence for tuberculosis ct image classification. In *CLEF2017 Working Notes*, CEUR Workshop Proceedings, Dublin, Ireland, September 11-14 2017. CEUR-WS.
7. Johannes Hofmanninger, Florian Prayer, Jeanny Pan, Sebastian Rohrich, Helmut Prosch, and Georg Langs. Automatic lung segmentation in routine imaging is a data diversity problem, not a methodology problem. *arXiv preprint arXiv:2001.11767*, 2020.
8. Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
9. Gustavo Pérez and Pablo Arbeláez. Automated detection of lung nodules with three-dimensional convolutional neural networks. In *13th international conference on medical information processing and analysis*, volume 10572, page 1057218. International Society for Optics and Photonics, 2017.
10. Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.
11. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
12. Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and¡ 0.5 mb model size. *arXiv preprint arXiv:1602.07360*, 2016.
13. Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

14. 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.
15. Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.
16. Saumya Jetley, Nicholas A Lord, Namhoon Lee, and Philip HS Torr. Learn to pay attention. *arXiv preprint arXiv:1804.02391*, 2018.
17. Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.
18. Mingxing Tan and Quoc V Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *arXiv preprint arXiv:1905.11946*, 2019.