Elycheva Dray ELYCHEVA.DRAY@DAUPHINE.EU
Theo Di Piazza THEO.DIPIAZZA@GMAIL.COM

DLMI - Challenge Report - Group 28

Abstract

In the case of a cancer treatment, radiation therapy is a usefull method whose goal is to burn the tumour efficiently while leaving healthy tissue unharmed. The step of preparing this radiation therapy treatment can be difficult and must be performed with precision and rigour to obtain good results. In this work, we have implemented, trained and tested Deep Learning methods capable of predicting the dose associated with each patient from a CT scan of the possible dose mask and 10 structural masks, from the Open-KBP dataset. The trained models show strong performance and the best score for MAE Loss is 0.372 on the test set.

Keywords: Deep Learning, Medical Imaging, Radiation, Dose Prediction, Cancer Patient, U-Net, ResNet, Data augmentation

Group 28 - Submission : group28_submission.zip - CodaLab team : tdipiazz

1. Introduction

For cancer treatment, radiation therapy is a method whose aim is to burn the tumour efficiently while leaving healthy tissue unharmed. It consists in irradiating patients with Gamma rays by sending photons on the patient using a Linear Accelerator of particles (LINAC). However, designing radiation therapy treatment plans can be slow or rely too much on approximations. To address this problem, one solution may be to predict the radiation dose from 2D images of the Open-KBP challenge dataset, using a Deep Learning method.

For each patient, we have 12 2D images: 1 CT scan image (128x128), 1 image corresponding to the maximum possible irradiation (128x128) and 10 masks of organs segmentations (128x128). From these 12 images, the objective is to predict the radiation dose (128x128).

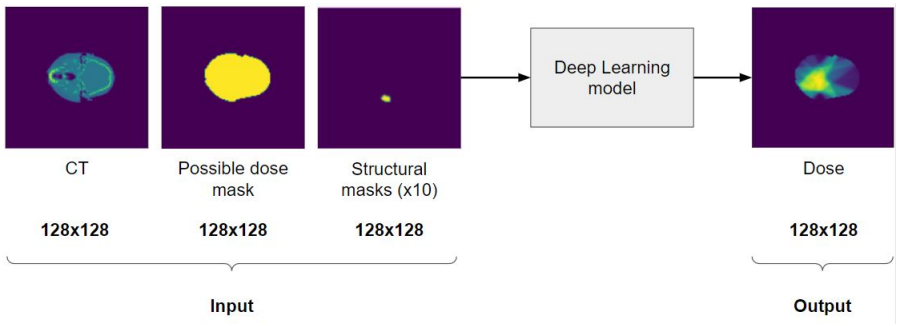

Figure 1: Scheme of the method.

Our work therefore consists in implementing, training and validating a model capable of performing this task.

2. Architecture and methodological components

Several model architectures have been tested to solve this problem. 2 models will be retained here : **U-Net** [1] and **ResNet** [2].
For these 2 models, the input is the concatenation of the **dose**, the **possible dose mask** and the 10 **structural masks**. The dimension of the input is therefore 12x128x128. The output of the model corresponds to the associated dose: the dimension of the output is therefore 1x128x128.

2.1. U-Net

U-Net is a well-known architecture in medical imaging. It consists of two parts: a succession of downsampling blocks (**encoder**) and a succession of up-sampling blocks (**decoder**). In the **encoder**, the size of the input is downsampled by 2 at each downsampling block by a layer called MaxPooling. In the **decoder** part, extracted features are upsampled progressively using a Transpose Convolution. Furthermore, in order to facilitate the reconstruction of the image in the decoder, skip-connections are present between each encoder block and a decoder block with the associated depth. Indeed, the network concatenates the higher resolution feature maps from the encoder network with the upsampled features in order to better learn representations with following convolutions.
U-Net offers the following advantages : **1.** achieves state-of-the-art performance on a wide range of medical image tasks ; **2.** allows for the use of global location and context at the same time ; **3.** skip connections help to mitigate the vanishing gradient problem and reduce overfitting ; **4.** can handle a variety of imaging modalities ; **5.** is computationally efficient.

2.2. ResNet

ResNet is also a widely used model in computer vision. The ResNet architecture uses residual blocks which are made up of several convolutional layers with skip connections that allow the input to be passed directly to the output of the block. Indeed, the architecture includes residual connections that skip over some layers in order to facilitate the flow of information. In fact, it architecture consistes of multiple convolutional layers, batch normalization layers, activation functions, and pooling layers. ResNet offers the following advantages : **1.** provides an innovative solution to the vanishing gradient problem with skip connections ; **2.** speeds up initial training by compressing the network into fewer layers ; **3.** has been shown to achieve state-of-the-art performance on a wide range of computer vision tasks.
Limited by computing resources, the ResNet34 architecture will be used: this corresponds to a ResNet architecture with 34 layers.

3. Model tuning and comparison

In this section, we'll explain how models were trained. First, the pre-processing and data augmentation methods will be detailed, then the architectures used will be presented.

3.1. Pre-processing

When viewing some of the images in the training and validation dataset, it can be seen that some of the images have no **possible dose mask**. In other words, all the pixels associated with this image are zero. Here is an example below:

Figure 2: Example of a sample without possible dose mask.

These samples do not provide any useful information, in the sense that there is no dose prediction possible. For this reason, it was decided to remove them from the train set and the test set. 271 (3.47%) were removed from the train set and 57 (4.5%) from the validation set.

3.2. Data Augmentation

In computer vision, data augmentation is an important part of the method. When used correctly, it allows to increase the size of the dataset by generating modified images from the training set. For some of the selected models, these different types of data augmentation will be used: flip, rotation around the y-axis, translation. When performing data augmentation, there's a probability 0.8 to have a flip, 0.4 to have a rotation and 0.8 to have a translation. Translation is made such that no information of the scan is lost. Besides, it is done such that each of the 12 slices for the same patient is transformed in the exact same way.

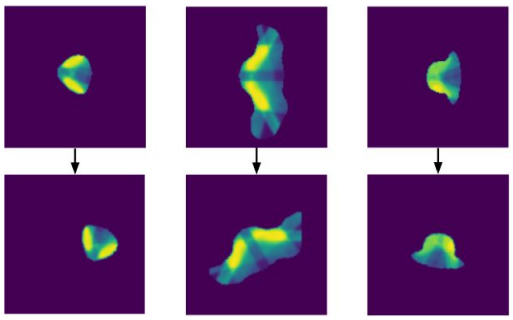

Figure 3: 3 examples of data augmentation on dose images.

3.3. Model tuning

For each model it is possible to modify the architecture and certain parameters (number of blocks for U-Net, depth for ResNet, learning rate, learning rate scheduler, etc.) which will have an impact on the quality of the model's learning.

Regarding the choice of the final architecture for U-Net, several models were tested with a different number of blocks and a different number of layers per block. There is a trade-

off between performance and complexity: the chosen model must offer solid performance, without requiring too much computing capacity. For each possible architecture, the model was trained over a dozen epochs and the one with the best results was retained.

U-Net The chosen model has 4 downsampling layers, 3 upsampling layers with 2 convolution layers per block (Appendix A).
ResNet ResNet18, ResNet34 and ResNet50 were tested. The training time with ResNet50 is significantly longer than with ResNet18 and ResNet34. In addition, the results are significantly better with ResNet34 than with ResNet18, for a correct calculation time. ResNet34 is therefore retained (Appendix B).
Parameters Different learning rates were tested: 0.01, 0.001, ..., 0.00001. The model does not seem to converge for an initial learning rate that is too low or too high. It was therefore decided to choose an initial learning rate of 1e−3 which will be reduced with a scheduler to 1e−4 after 15 epochs, 1e−5 after 20 and 1e−6 after 25 epochs. For the choice of the optimizer, Adam was selected because it gives very good results.

3.4. Validation procedure

For each model, different configurations were tried: with or without pre-processing; with a different data augmentation pourcentage (X% means that X% of the train set was used for data augmentation). The loss used is the MAE loss. The results on the validation set (average loss ± standard deviation) and the test set are shown.
In the choice of the final model, it is important to ensure that the performances on the train, validation and test sets are relatively close, in order to avoid over/under-fitting and to present a model that generalizes as well as possible.

Table 1: Validation and Test losses (MAE) for each method

Method	**Pre-processing**	**DataAug. %**	**Validation**	**Test**	**Time**
Baseline	-	-	-	1.499	-
U-Net	No	0	0.665 ± 0.536	0.586	1h
U-Net	Yes	0	0.674 ± 0.487	0.598	1h
U-Net	Yes	50	0.511 ± 0.348	0.445	1h30
U-Net	Yes	100	0.482 ± 0.341	0.403	2h
ResNet34	No	0	0.470 ± 0.344	0.408	1h15
ResNet34	Yes	0	0.496 ± 0.329	0.417	1h15
ResNet34	Yes	50	0.459 ± 0.323	0.382	1h50
ResNet34	Yes	100	$\mathbf{0.452} \pm 0.321$	**0.372**	2h30

We can see from **Table 1** that the models **ResNet34** present better results than **U-Net**. Moreover, data augmentation allows to significantly increase the performance of the models. Doubling the size of the dataset allows an increase of the score on the test of +0.183 for U-Net and +0.035 for ResNet34.

Implementation details All the networks were implemented using the Pytorch library and trained on one Tesla K80 GPU from Google Colab. L1-Loss optimization was done using Adam with a MultiStepLR Scheduler (lr: 0.001, steps = [15, 20, 25], $\gamma = 0.1$). Model was trained with a batch size of 32. Before pre-processing and data augmentation, there were 7800 samples from the initial train set and 1200 from the validation set. Models were trained over 30 epochs.

3.5. Qualitative results

The model selected is the one with the best score on the test set : ResNet34 with pre-processing and 100% data augmentation for a score of 0.372. Below are some results of the model predictions for different patients from the validation set. In the predictions displayed below and in the predictions made on the test set, we specify that we apply the possible mask dose to the prediction, in order to set to zero the pixels predicted by the model which cannot be taken into account in practice: this logically increases the accuracy of the model.

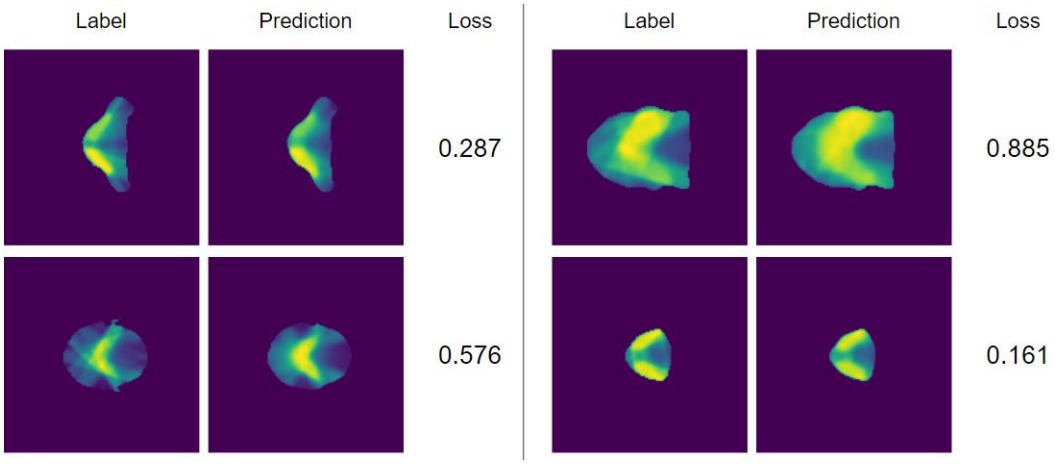

Figure 4: 4 labels and predictions from validation sample, with the best model.

It can be seen that, overall, the model does a good job of predicting the overall shape of the dose. However, it is important to note that the model seems to perform less well when there are discontinuities of intensity or inhomogeneous zones (light blue, dark blue, yellow on the images) in the dose. It tends to predict zones of homogeneous intensity.

Conclusion - Areas for improvement : This project was an opportunity to develop different Deep Learning models for dose prediction from CTs. The pre-processing and data augmentation steps were crucial to improve the performance of the implemented models and to obtain a score of 0.372 on the challenge test sample. We selected this model because it performed well on the validation sample with the lowest possible standard deviation.
In terms of model improvements, it would have been interesting to explore other approaches : pretrained models ; different architectures such as pix2pix GAN [3], Cascaded/Hierarchical U-Net [4] ; or even with implementation tips such as warmup steps.

References

[1] **U-Net: Convolutional Networks for Biomedical Image Segmentation** - O. Ronneberger, P. Fischer, T. Brox (2015)

[2] **Deep Residual Learning for Image Recognition** K. He, X. Zhang, S. Ren, J. Sun. (2015)

[3] **DoseGAN: a generative adversarial network for synthetic dose prediction using attention-gated discrimination and generation** V. Kearney, J. W. Chan, T. Wang, A. Perry, M. Descovich, O. Morin, S. S. Yom, T. D. Solberg (2020)

[4] **3D radiotherapy dose prediction on head and neck cancer patients with a hierarchically densely connected U-net deep learning architecture** D. Nguyen, X. Jia, D. Sher, M. Lin, Zohaib Iqbal, H. Liu, S. Jiang (2019)

Appendix

Appendix A : Simplified diagram of the implemented U-Net architecture.

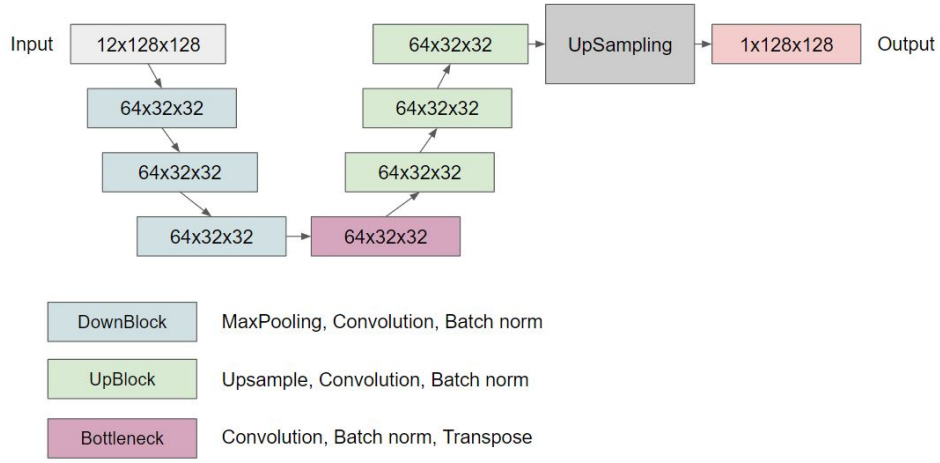

Figure 5: Simplified diagram of U-Net Architecture.

Appendix B : Simplified diagram of the implemented ResNet architecture.

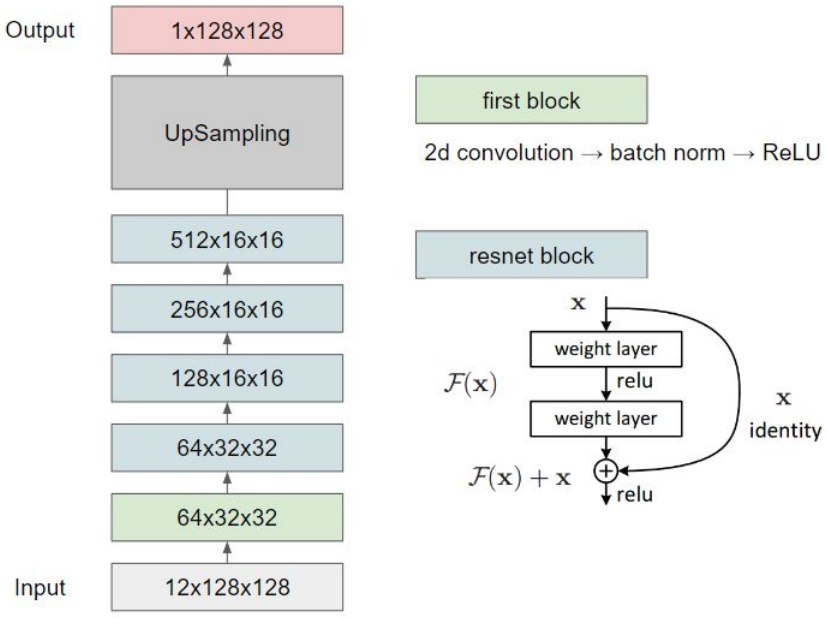

Figure 6: Simplified diagram of ResNet Architecture.