# Machine Learning for Health Care - Project 1 Report

Manuel Burger, Tobias Peter, Anej Svete *

March 2021

## 1 Introduction & Overview

This report provides a concise summary and discussion of our submission for the first project for the class *Machine Learning for Health Care*.

The objective of the project was to segment computed tomography (CT) scans containing colon cancer tissue. For this purpose, training data was provided consisting of 100 CT scans and the corresponding masks labeling the cancer tissue. Testing data was comprised of 26 CT scans. Evaluation of the submission is based on the *Intersection over Union* (*IoU* or *Jaccard Index*) metric.

## 2 Analysis

The first explorations of the dataset quickly made it clear that multiple steps of pre-processing were needed. For example, the varying depths of the scans meant that they needed to be transformed into a more unified structure. Examining the metadata of the scans showed a significant variance in voxel size, which suggested appropriate normalization. Analyzing the labels revealed a heavy imbalance between layers containing cancerous cells and those without any; cf. Table 1.

| Images ... | share |
|---|---|
| with cancerous cells | 10% |
| without cancerous cells | 90% |

Table 1: Heavy imbalance in the dataset.

## 3 Modeling & Training

### 3.1 Pre-processing

We kept a validation dataset of 20 random scans while the remaining 80 were used for training. For this, the 3D-scans were split into slices, which served as multi-channel 2D-image inputs for the model. We combined layers with two neighbors on each side resulting in 5 input channels. Due to the heavy imbalance in the dataset (cf. Table 1), we downsampled the images to obtain around 2200 images, whereof 45% contained cancerous segments. Additionally, we normalized voxel-size to its modal value to have a uniform scale across the scans.

To augment and perturb the training data, we randomly applied left-right and up-down flips as well as rotations. We also added Gaussian noise with $\sigma$ set to 0.05. After augmentation, the training set consisted of around 4400 images.

### 3.2 Model Architecture

As suggested, we used the so-called *U-Net* architecture, which is a convolutional neural network particularly developed for biomedical image segmentation similar to the task at hand.[1] The basis of our model was the Keras implementation given in [2].

Our U-Net model consisted of 4 convolutional layers on the down and upsampling path respectively and a bottleneck layer; the initial block contains 64 filters and the number of filters are doubled with each layer. To make the predictions more robust, we also used 3 output channels such that for layer $l$ of a given scan the output array $\boldsymbol{P}[l]_{i,j,k}$ was of size $512 \times 512 \times 3$. For the prediction label $p[l]_{i,j}$ of voxel $[i, j, l]$, we first averaged the corresponding channels over the neighboring layers

$$p[l]_{i,j} = \tfrac{1}{3} \left( \boldsymbol{P}[l-1]_{i,j,3} + \boldsymbol{P}[l]_{i,j,2} + \boldsymbol{P}[l+1]_{i,j,1} \right)$$

and then binarized $p$ with a threshold of 0.5.

### 3.3 Training

Training was generally performed with a batch size of 24 and used the *Adam* optimizer. To mitigate overfitting, a spatial drop-out rate of 0.2 was applied on the downsampling path of the network.

The model was then separately trained for a few epochs with IoU, Tversky loss [3], Focal loss [4] as well as Focal Tversky loss [5]. Despite the fact that IoU is used to evaluate the model's performance, the model did not train well with IoU-loss, as the loss encouraged to predict only *background* due to the small size of cancerous segments. From the other loss-functions, Focal Tversky loss showed the most promise and was picked for further training.[2] We used following variant of the

*{burgerm, topeter, asvete}@ethz.ch

[1] See [1] for details.

[2] Training is implemented in `train_keras.ipynb`.

Focal Tversky loss

$$\mathsf{FTL}(\alpha, \beta, \gamma) = \left(1 - \frac{TP + \varepsilon}{TP + \alpha \cdot FP + \beta \cdot FN + \varepsilon}\right)^{\gamma}$$

for given parameters $\alpha, \beta, \gamma$ and a small smoothing constant $\varepsilon$.[3] We emphasize hereby that $\beta$ determines the weight given to false negatives, i.e. undetected cancer segments.

Model A was then trained for 100 epochs yielding an IoU of 0.15 on the validation set.

#### 3.3.1 Pretraining

Due to the limited amount of training data and since proper initialization of a neural network can significantly improve convergence, we decided to first train the model on a different dataset.[4] We used the publicly available Kaggle dataset [6], which is based on the dataset [7]. The dataset consists of lung, heart, and trachea segmentations. Due to its comparable size to the cancer segments, we used only the trachea segmentations. Data augmentation and perturbation were applied similarly as for the actual training set (see Section 3.1).

The network was then trained with the same parametrization as Model A for 40 epochs on 100 CT scans corresponding to 6000 images. The final weights were then used to initialize the model for the actual training with further 100 epochs. The resulting Model B achieved a significantly higher IoU of 0.179.

#### 3.3.2 Improving cancer detection

As we observed that significant portions of cancerous segments were still undetected by Model B, we decided to downsample the data even further such that 65% of the images contained cancer. Additionally, we increased $\beta$ in the Focal Tversky loss to 2.0 punishing false negatives more. After 80 epochs of training with the same initialization as Model B, we obtained Model C with a validation IoU of 0.168.

#### 3.3.3 Ensemble & Final Submission

We then formed an ensemble out of Model B and Model C by averaging the outputs of the models before thresholding. The resulting Model D achieved the highest validation IoU with 0.197 (see Table 2), which was therefore selected for our submission.

| Model | Description | IoU |
|---|---|---|
| A | without pretraining | 0.150 |
| B | with pretraining | 0.179 |
| C | B + higher focus on cancer | 0.168 |
| D | ensemble (B + C) | 0.197 |

Table 2: Validation IoU for the various models.

[3] We used $\alpha = 0.5$, $\beta = 1.0$, $\gamma = 4.0$ as default.
[4] For details see `pretrain_keras.ipynb`.

## 4 Conclusion & Further Work

Despite an apparently competitive IoU on the validation set of 0.197, we observed large differences in prediction quality between individual patients (between 0.00 and 0.69), which we find concerning. Consequently, we doubt that the model could be beneficial to a medical professional in detecting cancerous tissue quicker in this form and we see the following opportunities for improvement.

First, as pretraining improved the model performance significantly, this could be further leveraged by using additional datasets forcing the model to extract better features before training on the target dataset.

Secondly, except for averaging layers, we did not use any post-processing for our submission. Experiments with multiple filters (e.g. Gaussian, median) as well as morphological transformations did not yield any substantial improvements. In the case of filters, we suspect that the averaging of the various channels already provided the same benefit. Yet, further post-processing methods should be investigated, as they might be able to remove small incorrect cancer labels without affecting the rest significantly.

Furthermore, there are images with significant cancer segments where the model predicts no cancer at all and vice versa. It could be beneficial to train a second model - either directly integrated into the U-Net network or used afterward for post-processing - to just detect the presence of cancer without trying to localize any segments.

Lastly, we suspect that a more tailored loss-function, e.g. a combination of Focal Tversky loss and IoU, could significantly improve training and ultimately the performance of the model.

### Main Contributions

| | |
|---|---|
| M. Burger | Implementation of the model and its pipeline, training. |
| T. Peter | Data analysis, post-processing, and writing the report. |
| A. Svete | Pre-processing, training, and code review. |

# References


[1] Olaf Ronneberger, Philipp Fischer, Thomas Brox. *U-Net: Convolutional Networks for Biomedical Image Segmentation* (2015). http://arxiv.org/abs/1505.04597.

[2] Karol Zak. *U-Net Implementations in Keras*. https://github.com/karolzak/keras-unet.

[3] Seyed Sadegh Mohseni Salehi, Deniz Erdogmus, Ali Gholipour *Tversky Loss Function for Image Segmentation Using 3D Fully Convolutional Deep Networks* (2017). https://arxiv.org/abs/1706.05721

[4] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, Piotr Dollár. *Focal Loss for Dense Object Detection* (2018). https://arxiv.org/abs/1708.02002.

[5] Nabila Abraham and Naimul Mefraz Khan. *A Novel Focal Tversky Loss Function with Improved Attention U-Net for Lesion Segmentation* (2018). https://arxiv.org/abs/1810.07842

[6] Kaggle dataset. *Chest CT Segmentation*. https://www.kaggle.com/polomarco/chest-ct-segmentation.

[7] Kónya et al. *Lung Segmentation Dataset* (2000). https://www.kaggle.com/sandorkonya/ct-lung-heart-trachea-segmentation.