Yousef Wally

Dr. Peter Chang

CS 190

June 9th, 2022

Deep Learning for Kidney Cancer using Classification and Segmentation

INTRODUCTION

According to the national cancer institute the survival rate for cancer patients is 93% if the tumor is detected early however the survival rate declines to 70% if the cancer had spread to any near by organs or tissues. An increase in diagnostic accuracy will help detect the tumor early in patients before it spread to other parts of their body.

METHODS AND MATERIALS

The dataset came from an open source called the Kidney Tumor Segmentation Challenge (KiTS). The cohort patients selected for the KiTS either underwent partial or radical nephrectomy for suspected renal malignancy between the year 2010 and 2020. The scans were independently segmented three times and classified into three classes.

1.Kidney

2.Tumor

3.Cysts

The data was annotated using JavaScript and the code is available to anyone to view or download locally to make changes in the following link https://github.com/SenteraLLC/ULabel. The annotation type was segmentation. The annotation team was placed into 3 categories according to KiTS:

- **Experts**: Attending radiologists and urologic cancer surgeons

- **Trainees**: Medical students, undergraduates planning to study medicine, and a lowly Computer Science PhD Student. All trainees received several hours of training from the experts
- **Laypeople**: People who received no training other than a brief

There were a total of 25 members in the annotation plus the employees of iMerit that were categorized as “laypeople”

Data generators were created for both training data and validation data, those data were created for all the three models. The generator produced 321 images and 81 validation images.

For training fit, the first model was trained for 10 epochs and a validation step per 100 steps, the second model trained 20 epochs with a validation step per 100 steps and finally the third model was trained for 50 epochs and a validation step per 100 steps

NETWORK DESGIN

The overall goal of the experiment is to be able develop a model for automatic semantic segmentation and classification of renal tumors and surrounding anatomy using machine learning. There are three models,

Model 1 (Classification):

This mode uses 2D data set of input size (1, 96, 96 , 1), this model is using a classification method for binary tumor detection. This a 2D method thus it will predict tumor number on slice-by-slice basis, whereas if it was 3D it would predict the tumor number by volume basis, but that’s not the case here. This classification method uses Bottleneck method. Bottleneck operation is used to decrease the total number of filters for convolutional efficiency. Bottleneck method in this classification utilizes a 1 x 1 x 1 convolutional kernel furthermore it is defined as a full convolutional block. The model had to be converted from CNN to MLP architecture , meaning the map needs to be collapsed into a vector form . Layers.Reshape()

operation was used to convert the map to (1, 1, 1, N * N * N * C) vector. The final goal of the model was to produce a single binary classification for tumor vs no tumor per volume thus a reduction strategy was necessary to implement so the multiple predications can be collapsed into just a single global per volume prediction. The optimizer that was used for training this model was Adam with a learning rate of 2e-4.

Model 2 (Segmentation):

Model 2 uses 2D dataset of input size (1 ,96, 96, 1). This model is using a segmentation method for binary tumor detection. This a 2D method thus it will predict tumor number on slice-by-slice basis. This segmentation method using residual layer technique, this creates the highest performance using residual connection using full pre-activation. To setup full pre-activation we started by applying a naïve convolution to the first input layer which is layer zero. Furthermore, a projection lambda function had to be defined to setup third block with residual connection. The projection operation had to be strided as well so connection to layer 2 and 1 can be done to define layer 3. The final goal of the model was to produce a single binary classification for tumor vs no tumor per volume thus a reduction strategy was necessary to implement so the multiple predications can be collapsed into just a single global per volume prediction. The optimizer that was used for training this model was Adam with a learning rate of 2e-4.

Model 3 (Custom):

Model 3 uses 2D dataset of input size (1 ,96, 96, 1). This model is using a segmentation method for binary tumor detection. This a 2D method thus it will predict tumor number on slice-by-slice basis. This model was defined using a classification and implementation of Squeeze-and-Excite architecture, starting with the output of the standard convolutional block, the intermediate feature map was scaled by a constant independently across

all the channels to emphasizes the feature for the given single map. It collapsed the N x N feature maps to 1 x 1 via global pooling, excitation of two fully connected layers to model channel wise then scaling all feature maps by the learned excitation values.

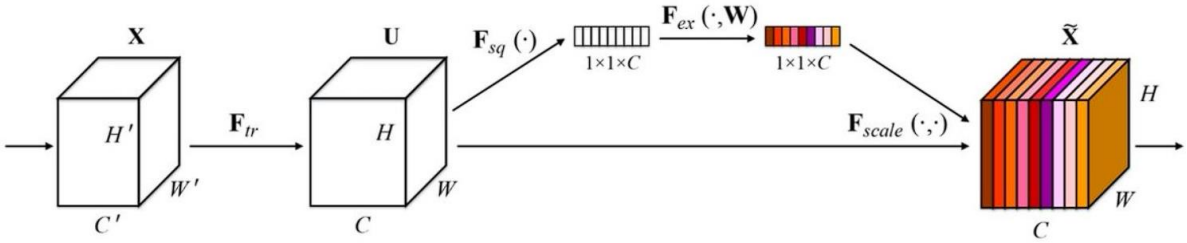

Figure 1 Squeeze-and-Excite architecture

The final goal of the model was to produce a single binary classification for tumor vs no tumor per volume thus a reduction strategy was necessary to implement so the multiple predications can be collapsed into just a single global per volume prediction. The optimizer that was used for training this model was Adam with a learning rate of 2e-4.

IMPLEMENTATION

Python version 3, the NumPy, Pandas, and TensorFlow libraries which constructed most of the models and layers jarvis-md library was also utilize,. was used to run these experiments on AMD 6900XT GPU with 16 GB memory. The first model was trained for 10 epochs with 100 per epoch the second model trained 20 epochs with a with 100 step per epoch and finally the third model was trained for 50 epochs and a with a 100 step per epoch. As model one and three worked on classification, it used an accuracy metric. Model 2 used Dice score metrices. All three used a softmax cross-entropy loss metric

RESULTS

	Training					Validation				
	accuracy	sens	spec	PPV	NPV	accuracy	sens	spec	PPV	NPV
Model 1	0.8536	0.8679	0.8395	0.8415	0.8662	0.6173	0.6171	0.6176	0.6905	0.5385
Model 2	0.9626	0.9686	0.9568	0.9565	0.9688	0.5802	0.5745	0.5882	0.6585	0.5001
Model 3	0.9938	0.9999	0.9877	0.9876	0.9999	0.6543	0.6383	0.6765	0.7317	0.575

Figure 2 results of training and validation

Model 1 had a validation accuracy value of 0.6173. Model 2 had a validation accuracy value of 0.5802. Model 3 had a validation accuracy value of 0.6543. From the results it appears model 3 that used Squeeze-and-Excite architecture performed better than Residual layer and Bottleneck architecture. It is notable that model 2 is ranked for accuracy value. Model 3 also had the best training accuracy value than the other models sitting at a value accuracy of 0.9938.

Discussion

All models produced similar results that is not far from each other, however Model 3 that used Squeeze-and-Excite architecture performed the best. It can be noticed from the results table that the training accuracy value for model 1 is the lowest, that could be related that the model went through a degree of overfitting. Because validation shows the decent performance of the models, it can be concluded that poor training could be a result of bad training of data. Because model 3 trained more epochs than the other two models it can also be concluded that the higher training value performance is related to training more epochs.

Works Cited

"The 2021 Kidney Tumor Segmentation Challenge," *KiTS21*, Mar-2021. [Online]. Available: https://kits21.kits-challenge.org/. [Accessed: 7-June-2022].

"Cancer of the kidney and renal pelvis - cancer stat facts," *SEER*, 2021. [Online]. Available: https://seer.cancer.gov/statfacts/html/kidrp.html. [Accessed: 7-June-2022].