Geraldo A. Vera Perez
Javier Hernández Novoa
Jean Carlos Rodríguez
CIIC4019-036
Professor: Wilson Rivera
November 15, 2018

Google Inception in the Identification of Lung Cancerous Tumors

Technical approach to the problem

The problem is to use an image classification model to be able to detect tumors associated to the most common type of lung cancer: adenocarcinoma of the lung. Our approach was to retrain Google Inception's machine learning model to classify images that contain these lung adenocarcinomas. In order to retrain the aforementioned model to diagnose positive for these cancerous tumors, we used images from a collection of diagnostic CT scans from The Cancer Imaging Archive (TCIA). The aforementioned CT scans were gathered from 61 cases and show tumors ranging in size from 3 to 6mm. We used another collection from TCIA, called the Lung Image Database Consortium image collection (LDIC), to diagnose negative for cancerous tumors because it contains scans of healthy lungs and lungs with nodules gathered from 1018 cases. According to our research, most nodules are not only benign but also quite common for which reason we decided to teach our machine learning models to classify these as being negative to lung cancer. After converting all the CT scans from DICOM to JPEG format and resizing

them to 299x299 pixels, we uploaded the scans to our virtual environment and used TensorFlow for Poets 2 to create our various models.

Performance metrics and experimental settings

A chameleon cloud instance was created with CentOS 7. The instance used in this virtualization environment had the following CPU architecture:

```
Architecture:          x86_64
CPU op-mode(s):  32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):               48
On-line CPU(s) list:   0-47
Thread(s) per core:    2
Core(s) per socket:    12
Socket(s):            2
NUMA node(s):          2
Vendor ID:             GenuineIntel
CPU family:            6
Model:                63
Model name:            Intel(R) Xeon(R) CPU E5-2670 v3 @ 2.30GHz
Stepping:             2
CPU MHz:               2701.489
CPU max MHz:           3100.0000
CPU min MHz:           1200.0000
BogoMIPS:              4599.85
Virtualization:        VT-x
L1d cache:             32K
L1i cache:             32K
L2 cache:              256K
L3 cache:              30720K
NUMA node0 CPU(s):     0,2,4,6,8,10,12,14,16,18,20,22,24,26,28,30,32,34,36,38,40,42,44,46
NUMA node1 CPU(s):     1,3,5,7,9,11,13,15,17,19,21,23,25,27,29,31,33,35,37,39,41,43,45,47
Flags:                 fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush
dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx pdpe1gb rdtscp lm constant_tsc arch_perfmon pebs
bts rep_good nopl xtopology nonstop_tsc aperfmperf eagerfpu pni pclmulqdq dtes64 monitor ds_cpl
vmx smx est tm2 ssse3 sdbg fma cx16 xtpr pdcm pcid dca sse4_1 sse4_2 x2apic movbe popcnt
tsc_deadline_timer aes xsave avx f16c rdrand lahf_lm abm epb ssbd ibrs ibpb stibp tpr_shadow vnmi
flexpriority ept vpid fsgsbase tsc_adjust bmi1 avx2 smep bmi2 erms invpcid cqm xsaveopt cqm_llc
cqm_occup_llc dtherm ida arat pln pts spec_ctrl intel_stibp flush_l1d
```

Our main performance metric for the model is recall.

$$recall = \frac{true\ positives}{true\ positives\ + false\ negatives} \qquad precision = \frac{true\ positives}{true\ positives + false\ positives}$$

In the article “Beyond Accuracy: Precision and Recall”, recall is the ability of a model to find all the relevant cases within a dataset. In other words, it is the ability to identify true positives, a higher recall means less false negatives. For a lung cancer classification model this is the prioritized performance metric because a higher recall means that there is smaller possibility of diagnosing a lung CT scan as negative for cancer when in reality it should be positive, which could be dangerous given cancer’s nature.

Infrastructures and programming models used

An image classification model that uses TensorFlow to implement a deep learning model is used to diagnose whether a lung CT scan has a cancerous tumor or not. This image classification model is Google’s Inception V3 model. We took a pre-trained model trained with the ImageNet dataset, and fine tuned it with a data set of over 4000 CT scans (links to the two collections used can be found on the references section). Scans of patients with lung cancer from their diagnosis up to their operation, were labeled as positive while scans from patients with healthy lungs or lungs containing nodules were labeled as negative.

Once trained, our inception model should be able to look at a lung CT scan and determine whether it has lung cancer or not. As explained in the Inception documentation, since we are re-training a pre-trained model, only the final classification layer of the model is changed, the rest of the model is left as it was when trained with the ImageNet dataset. In the final classification layer the number of labels are changed accordingly and the weights are randomly initialized. Then it is trained with our data set.

First, a model was fine tuned with the instructions on the documentation in the Inception Github repository. With this we were able to fine tune and evaluate a model but instructions were not clear on how to classify a single image or how to test the model outside of the evaluation script. For these reasons we decided to fine tune another model by following a TensorFlow for Poets 2 tutorial and applying it to our dataset. This model was much easier to fine tune but this was thanks to the previously trained model which gave us a deeper understanding of what was going on. With this model we were able to try different hyperparameters and train various models. These various models were tested on images of lung CT scans we got from Google and Bing image searches.

Experimental results

	MODEL 1	MODEL 2	MODEL 3	MODEL 4
TRUE POSITIVES	8	7	7	5
FALSE NEGATIVES	2	3	3	5
TRUE NEGATIVES	2	3	3	3
FALSE POSITIVES	8	7	7	7

Table 1

Model #	Recall	Precision
Model 1	0.80	0.50
Model 2	0.70	0.50
Model 3	0.70	0.50
Model 4	0.50	0.42

Table 2

Table 1 gives a type of confusion matrix for the four models trained. A threshold of 50% was used. Table 2 shows the recall and precision calculated from the values in table 1. 80 tests (classifications) were performed in total and the average time it took to classify each image was 0.1473375 seconds. The raw experimental results can be found in the repository as a pdf file named: “Experimental Results”.

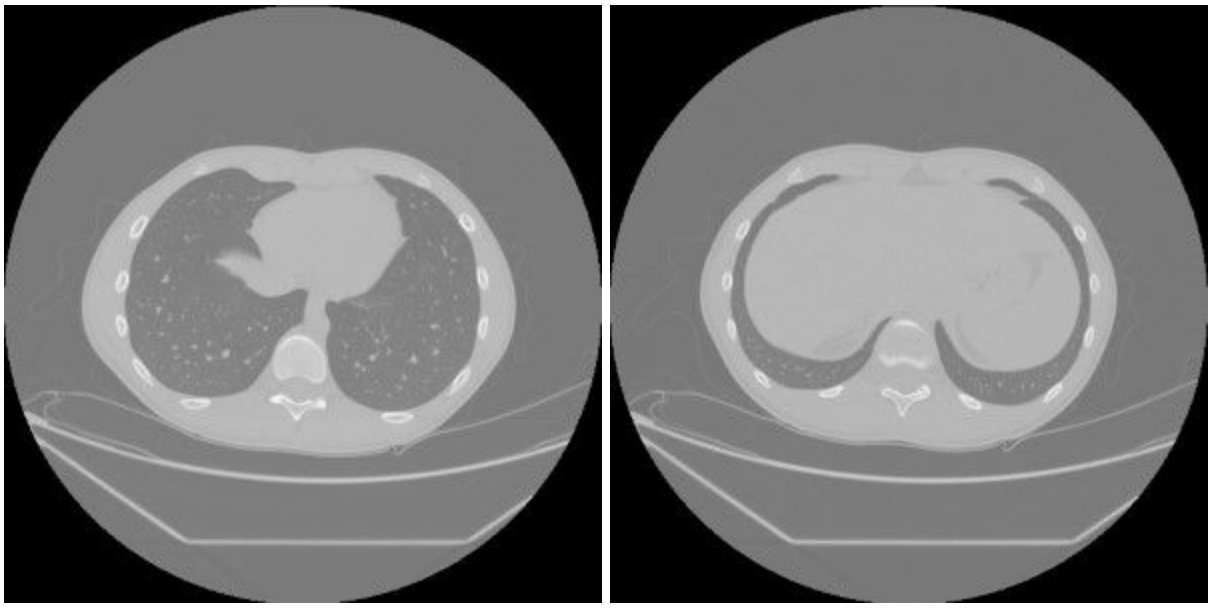
Conclusions and Future Work

We tested our fine-tuned models on ten images of healthy lungs and ten images of lungs with cancerous tumors, all of which were found in online image searches. Our best model was able to detect lungs with cancer quite inaccurately with a precision of 50% since it gave too many false positives when classifying healthy lungs. Our best model's recall was of 80% meaning that there is a 20% chance of diagnosing a lung as being healthy when in reality it isn't. This behavior is not bad but should be much better for this lung cancer detection image classification model to be considered a reliable model.

Both our recall and precision should be higher for this, different hyperparameters were used to improve our model but to no success. We believe this might be due to our healthy lung dataset because it contains CT scans of lungs taken at different phases of breathing meaning that some are during the person's inhalation, others are during the person's exhalation, and others are before or after said phases. Since these two phases look differently in the CT scans (as seen in the images below) it most likely affected the model's learning process. Since most of the tumor-containing scans seem to have been taken only during one of these phases (in which the tumors are clearly visible in white over the black space), we conclude that this led to the model to be much better at positive-classifying than negative-classifying. This conclusion is visible when one observes Table 1 in the previous section and takes into consideration that all images chosen for testing are in the respiratory phase which is abundant in the scans which

contain cancerous tumors. Because of all of this, it can be deduced that if a consistent healthy lung dataset is acquired, classification should drastically improve and this would lead to a model that provides much more accurate diagnoses.

Example of the lung scans being shown at different respiration phases within the healthy lung dataset:



References:

- Dataset used to teach the model what should be considered as Positive for Lung Cancer - “LungCT-Diagnosis” : https://wiki.cancerimagingarchive.net/display/Public/LungCT-Diagnosis

- Dataset used to teach the model what should be considered as Negative for Lung Cancer - “LIDC-IDRI” : https://wiki.cancerimagingarchive.net/display/Public/LIDC-IDRI

- The Cancer Image Archive: http://www.cancerimagingarchive.net/

- TensorFlow for Poets 2 GitHub: https://github.com/googlecodelabs/tensorflow-for-poets-2

- TensorFlow for Poets 2 Tutorial: https://codelabs.developers.google.com/codelabs/tensorflow-for-poets/?utm_campaign=chrome_series_machinelearning_063016&utm_source=gdev&utm_medium=yt-desc#3

- Google Inception GitHub https://github.com/tensorflow/models/tree/master/research/inception

- Google AI blog post about Google Inception https://ai.googleblog.com/2016/03/train-your-own-image-classifier-with.html

- Beyond Accuracy: Precision and Recall https://towardsdatascience.com/beyond-accuracy-precision-and-recall-3da06bea9f6c