Google Presentation slides:

https://docs.google.com/presentation/d/1Yi6lkrQ4a91_uqSmim33fOknnPaibZVDXNvPbgM4Ubc/edit?usp=sharing

CNN TRANSFER LEARNING MODELS USED ON LUNG CANCER NODULES CLASSIFICATION

Thomas Butler
School of Data Science
University of Virginia
Charlottesville, USA
Tommysbutler@gmail.com
vra2cf@virginia.edu

Drew Haynes
School of Data Science
University of Virginia
Charlottesville, USA
rbc6wr@virginia.edu

Christian Schroeder
School of Data Science
University of Virginia
Charlottesville, USA
dbn5eu@virginia.edu

August 10, 2022

ABSTRACT

Computer vision is a good way to identify objects and patterns in imagery that the human eye otherwise would not notice. In recent years, the number of applications of computer vision to medical imaging has grown. One of those applications is for classifying cancerous lung nodules in CT scans. Current methods employ convolutional neural networks (CNN) to do this. Using a subset of the Lung Image Database Consortium Image Collection (LIDC-IDRI), the Lung Nodule Analysis 2016 (LUNA16) data set, we experiment with the application of CNNs with transfer learning models Xception, InceptionResNetV2, and ResNet15V2 to identify lung nodules in medical images that are likely cancerous and compare their performances. We have determined InceptionResNetV2 model achieved the best results out of the 3 transfer learning models. More work needs to be done to improve our code to a point where it can be used in a medical setting.

1 Motivation

Computer vision is the process of computers processing digital pictures into information [1]. Computer vision can be used in combination with medical imaging technology to receive medical information and make medical decisions.

The World Health Organization (WHO) states that lung cancer was the sixth leading cause of death in 2019 [2]. This makes lung cancer the deadliest cancer in the world. It is estimated that lung cancer causes 1.59 million deaths per year [3]. Since lung cancer is such a serious disease across the world, developing a model that detects lung cancer can greatly improve the quality of life for all potential lung cancer patients, decrease treatment morbidity, and improve experience of care [16].

Lung nodules are abnormal growths inside the lung, 95% of observed nodules are non-cancerous [15]. Lung nodules are examined on X-ray or computed tomography (CT) scans to determine if they are cancerous.

Deep learning networks are the models that have the most success inside of medical imaging and the best networks are convolutional neural networks (CNN) [4]. CNNs can be used on medical imaging classification, localization, detection, segmentation and registration [5]. Imaging detection is used to locate things in a image [6]. Detection can be used to find a lesion or a cancerous growth on a image [5]. Imaging classification refers to classifying items in a image. Classification is a more advanced version of detection as it detects where a object is but it also classifies exactly what that is in comparison to many different things [6]. Image localization is finding where a object is and drawing a bounding box around it [7]. Medical imaging localization example is classifying large body scan images by various body organs[5].

Image segmentation is breaking the picture down into its various parts [8]. Medical segmentation can be used to segment various parts of the brain to determine the exact boundaries of a tumor is [5]. Image registration is overlaying images taken at different times. In medicine this helps to account for breathing, movement and other dynamic forces [9].

Current methods show CNN with transfer learning using InceptionV3 having the highest accuracy at 96.6% [4]. We want to create a CNN transfer learning model trying different models and determine which is best with our data.

2 Literature Review

2.1 Current and Future Methods

Computer aided medical imaging has gone through two major methods. The first used a feature transformer called scale-invariant feature transform (SIFT) with support vector machines (SVM) to classify medical images to get a precision at 67%. SIFT was later replaced with oriented fast and rotated binary (ORB) since SIFT is a patented algorithm. CNN was the next major method to be adopted.

CNN is good for feature extraction, can avoid complex feature engineering, and is already used in medical classification [4]. The best performing model we have seen in this context is a CNN with transfer learning with the an accuracy of 96.6% . The authors of that study also compared their model's results to human experts, who saw similar sensitivity but lower specificity than the CNN. Whether a model should favor sensitivity or specificity is a question we will explore as we build out our model. The next method that holds promise is capsule neural networks. However, this method was recently invented in 2017 and requires more time and research to explore how it can best be used [4].

2.2 Controversy

With the application of machine learning to medical imaging becoming more prominent, more questions start to arise regarding public reception and responsibility. Patients may be reluctant to trust results from an algorithm more so than results delivered by a medical professional. This impact could potentially be mitigated by referring to the algorithm as an assistive tool to the doctor when speaking to a patient.

A bigger question is that of responsibility. It is easy to view machine learning as the obvious solution to increasing diagnostic accuracy. But, what happens when it is wrong? Who is responsible? The inability to fully understand and explain the contents of a black-box machine learning algorithm creates a lot ambiguity around that question [5]. A potential solution to artificial intelligence (AI) malpractice is the same for doctors, AI medical malpractice insurance.

Another concern is the development of error-prone behavior. Even when a model consistently outperforms humans in diagnosing patients, it is still prone to error. The belief that AI will always do better than a human could lead to blind trust in the model and overtime impair someone's ability to identify incorrect diagnoses [12].

2.3 Limitations of Prior Research

Capsule Networks (CapsNets) is a newer type of network architecture which could replace CNN. CapsNets always has equivariance, more likely to classify a crowded picture, requires less training data and is more interpretable than CNN. CapsNets are new so a state of the art use case has yet to be developed. CapsNets are also slow to train. More work should be done on CapsNets especially in the image segmentation and object detection fields of medical imaging [11].

Generative adversarial networks (GANs) are a method of unsupervised learning where two competing models, a generator and a discriminator, work simultaneously to generate images that are equally likely to be considered real or not [5]. GANs can be used to generate higher quality CT scans from magnetic resonance imaging (MRI) images avoiding the need of exposing a patient to ionizing radiation. At the very least GANs can generate higher quality images from MRI scanners which means less expensive MRI scanners can be used to generate higher quality images reducing patient costs. This has been shown for brain images this work can be done for lung cancer images too [5]. GANs can also be used to improve image quality which can be useful as a expensive form a prepossessing for a model[5].

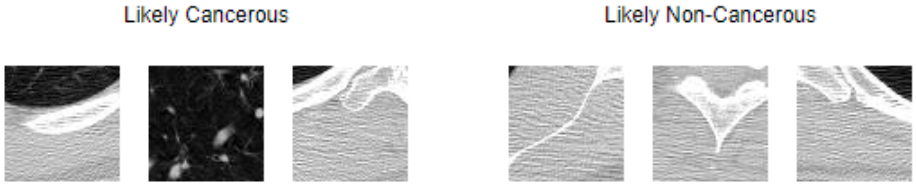
3 Data

The full data set consists of 125GB of annotated lung cancer screening scans from 1018 cases. The LIDC-IDRI url is `https://wiki.cancerimagingarchive.net/display/Public/LIDC-IDRI`. The LUNA16 data set, `https://luna16.grand-challenge.org/Data/`, excludes scans with slice thickness greater than 2.5 mm and includes 888 CT scans. For each patient, the number of lung nodules greater than and less than 3 mm in size are recorded. Nodules

are annotated through a two-phase process using four radiologists [19]. Lesions marked as non-nodules, nodules less than 3 mm, and nodules marked by less than three radiologists are considered "irrelevant findings", while a nodule is considered likely cancerous if it was marked as greater than 3 mm by three out of four radiologists. All other nodules are marked as non-cancerous [19]. Our data lacks how large each nodule is which makes it difficult to know which stage of cancer our pictures represent. We do know anything less than 3 cm in diameter is stage 1 cancer so all our nodules are at least at stage 1. Our data is largely negative lung cancer nodules, with only 0.2% of the data is positive lung cancer nodules. This unbalanced data set causes difficulties in training.

To prepare the data for modeling we followed the example set out by Grand Challenge Luna16's tutorial code, which demonstrated a process to convert data from .mhd to .jpg image formats[20]. One needs to join the challenge to see tutorial code. There were several steps of preparation that needed to be done to properly train on the data. We first needed to segment the full CT scans into patches of the individual candidate nodules. To do this, we loaded the images into arrays and extracted metadata regarding pixel spacing and origin, as well as loaded the list of candidate nodules and their respective coordinates within their images. Using the pixel spacing, origin, and world coordinates, we converted the coordinates to voxel coordinates and extracted a patch for each candidate nodule in the list and save them as .jpg files for training our model.

Figure 1: Example Patches



Number of Samples per class		
	Class 0 (no tumor)	Class 1 (tumor)
Original Dataset	753418	1557
Unbalanced Dataset	7570	1184
Balanced Dataset	1517	1184

There were several errors that arose during this process that we are still facing. The first is the script failing to find a specific image, the second being unable to convert a point to voxel coordinates and the third is a failure to convert a point. We believe the first issue is a specific to Google Colab, in relation to the excessive filenames of the images. We believe we can solve the second issue by padding the images with voxel widths with zeros, but have not employed that fix yet. We are not sure why there was a failure to convert a point.

Our data consists of 754,976 total candidates. We weeded out many patches that failed through the process due to errors, and were left with 571,540 total cases. Although we still have a lot of candidates, the classes are quite unbalanced, with only 1,184 considered cancerous. This can lead to issues in classification, like misleading true negative rates, where predicting negative 100% of the time would still result in very high accuracy. To try and combat this while iteratively training and tuning our models, while also mitigating excessive load times, we defined a subset of a unbalanced and balanced dataset of patches. These distributions can be found in the Number of Samples per class table. Although the distribution is not representative of our population, we believe this helped us train a model to maximize detection of true positives and avoid false negatives.

4 Method

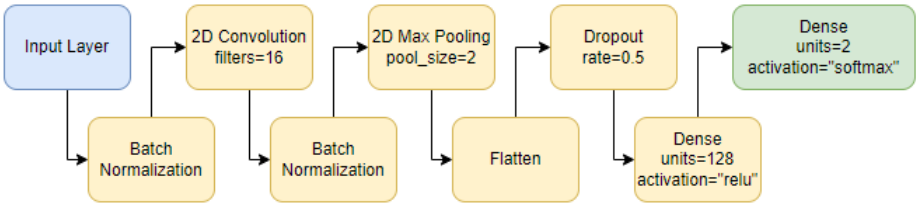
We have tried a number of different CNN models. We tried a combination of Simple CNN model and Deep CNN model and transfer learning models on Xception, InceptionResNetV2, and ResNetV2. We were trying to increase validation metrics through all of these methods.

5 Experiments

With our data being quite imbalanced between classes, we chose to evaluate model performance using F1-score, precision and recall per class, instead of accuracy.

For our base model, we turned to building our own CNN. We did some research and found the best model on Luna16 results [21] and the Fashion Mnist's CNN on "M6.3 Tutorial: Deep Computer Vision using CNNs" and tested the various ideas in those models. This process consisted of numerous iterations and lot of trial and error to fine tune the model to our liking. We experimented with batch normalization after layers, dropout between 2D convolution layers, multiple 2D convolution layers with same and different numbers of filters, differing amounts of filters or dense units in layers, different optimizers with different learning rates, different kernel sizes, and a generally deeper or shallower network.

Figure 2: Base Model Architecture



After defining our base model, a number of experiments were performed while learning the best practices to segment, pre-process, and train against our data set. Most notably, in our efforts to employ transfer learning in our process we experimented with three models; Xception, InceptionResNetV2, and ResNetV2. The level of success varied, but ultimately fell short with each method.

During our code we encountered multiple errors which we had to analyze to solve. We had issues with converting our images from .mhd to .jpg, scale of data, unbalanced data causing batches to not include both classes, metrics not running properly and shaping our data correctly for transfer models. These issues were solved with a variety of ways: searching these errors online, discussing potential solutions inside the group, and the professor guiding us to solutions.

6 Results

We conducted these experiments in a Google Colab notebook, which can be found here, Unbalanced Dataset code Balanced Dataset code. The code was also submitted to UVA Collab submission.

6.1 Balance Dataset Model

When comparing the performances of the transfer learning models against our CNN base model, we found the base model to be the most performant on our test data set. Our transfer learning models performed similarly or slightly worse than the base model. The Balanced Dataset Models table shows all performance metrics.

Balanced Dataset Models				
Models	F1_score	Precision	Recall	auroc
Base	0.6264	0.4618	1.0000	0.9015
Test	0.6055	0.4375	1.0000	0.9942
Xception	0.5997	0.4306	1.0000	1.0000
InceptionResNetV2	0.6131	0.4481	1.0000	0.9973
ResNet152V2	0.5997	0.4306	1.0000	0.5171

6.2 Unbalance Dataset Model

When comparing the performances of the transfer learning models against our CNN base model, we found the InceptionResNetV2 to be the most performant on our test data set. Our transfer learning models performed slightly better than the base model. The Unbalance Dataset Models table shows all performance metrics.

Unbalanced Dataset Models			
Models	F1_score	Precision	Recall
Base	0.2215	0.1250	1.0000
Test	0.2282	0.1295	1.0000
Xception	0.2439	0.1414	1.0000
InceptionResNetV2	0.2560	0.1548	0.9524
ResNet152V2	0.2352	0.1349	1.0000

7 Conclusion

Our models need more work before they can be used to help diagnose cancer. If our models are improved enough to beat a average radiologist's ability to interpret cancerous tumors from CT scans we can help diagnose cancer on more patients leaving time for radiologist to focus more on other aspects of their job overall increasing the detection rate of cancer and general quality of care of cancer patients.

During our project we have seen InceptionResNetV2 outperform the other two transfer learning methods. The other models should be explored but this one should be the focus as it seems most promising.

7.1 Future Work

Our results leave a lot of room for improvement and multiple areas to explore.

7.1.1 Threshold to maximize metrics

Creating a function that can use Bayesian optimization or some other optimization to determine which threshold maximizes F1 Score is required for this project. Currently all our values use a threshold of 0.5 but our accuracy and auroc metrics are high for both the balance and unbalanced datasets. This leads me to believe the current problem with poor metrics is not our models but a sub optimal threshold. We are hoping this solves our poor metrics problem.

7.1.2 Large Scale Modeling

Another step will be applying our model to the entire data set. Currently Google colab can't handle more than 1% of our 0 class data. Rivanna had some difficulties storing our full data and because of time constraints that couldn't be solved. Also we are interested in seeing how bringing in the much larger LIDC-IDRI data will affect the performance of our model.

7.1.3 Early Stopping

Implement Early Stopping into our code to prevent overfitting of models. We need to determine the best metric to measure for this.

8 Contribution

Thomas has been the majority contributor to the project, as he has played the role of the leader over the past several weeks. Thomas has taken the initiative in defining the goals of the project, initiating steps, writing the paper, and writing the code.

Christian has helped write a large portion of this paper as well as trouble shoot code with Thomas.

Drew wrote the framework for converting our .mhd files into segments of .jpg. He also helped write the paper and contributed a few references to the paper.

Thomas Butler 33%

Christian Schroeder 33%

Drew Haynes 33%

References

[1] What is Computer Vision? | IBM. (n.d.). Retrieved June 27, 2022, from https://www.ibm.com/topics/computer-vision

[2] The top 10 causes of death. (n.d.). Retrieved June 27, 2022, from https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death

[3] Serj, M. F., Lavi, B., Hoff, G., & Valls, D. P. (2018). A Deep Convolutional Neural Network for Lung Cancer Diagnostic (arXiv:1804.08170). arXiv. http://arxiv.org/abs/1804.08170

[4] Yadav, S. S., & Jadhav, S. M. (2019). Deep convolutional neural network based Emedical image classification for disease diagnosis. Journal of Big Data, 6(1), 113. https://doi.org/10.1186/s40537-019-0276-2

[5] Ker, J., Wang, L., Rao, J., & Lim, T. (2018). Deep learning applications in medical image analysis. IEEE Access, 6, 9375–9389. https://doi.org/10.1109/ACCESS.2017.2788044

[6] Renukasoni. (2019, July 31). Image detection, recognition and image classification with machine learning. AITS Journal. https://medium.com/ai-techsystems/image-detection-recognition-and-image-classification-with-machine-learning-92226ea5f595

[7] Karagiannakos, S. (2019, May 3). Localization and object detection with deep learning. Medium. https://towardsdatascience.com/localization-and-object-detection-with-deep-learning-67b5aca67f22

[8] Tyagi, M. (2021, July 19). Image segmentation: Part 1. Medium. https://towardsdatascience.com/image-segmentation-part-1-9f3db1ac1c50

[9] Image registration. (2022). In Wikipedia. https://en.wikipedia.org/w/index.php?title=Image_registration&oldid=1080045907

[10] Jakimovski, G., & Davcev, D. (2019). Using double convolution neural network for lung cancer stage detection. Applied Sciences, 9(3), 427. https://doi.org/10.3390/app9030427

[11] Hinton, G., Sabour, S., & Frosst, N. (2018). Matrix capsules with EM routing. https://openreview.net/pdf?id=HJWLfGWRb

[12] Colak, E., Moreland, R. & Ghassemi, M. (2021) Five principles for the intelligent use of AI in medical imaging. Intensive Care Med 47, 154–156 (2021). https://doi.org/10.1007/s00134-020-06316-8

[13] Jakimovski, G., & Davcev, D. (2018). Lung cancer medical image recognition using Deep Neural Networks. 2018 Thirteenth International Conference on Digital Information Management (ICDIM), 1–5. https://doi.org/10.1109/ICDIM.2018.8847136

[14] G, A. I., Samuel, McLennan, G., Bidaut, L., McNitt-Gray, M. F., Meyer, C. R., Reeves, A. P., Zhao, B., Aberle, D. R., Henschke, C. I., Hoffman, E. A., Kazerooni, E. A., MacMahon, H., Van Beek, E. J. R., Yankelevitz, D., Biancardi, A. M., Bland, P. H., Brown, M. S., Engelmann, R. M., Laderach, G. E., ... Clarke, L. P. (2015). Data from lidc-idri. The Cancer Imaging Archive. https://doi.org/10.7937/K9/TCIA.2015.LO9QL9SX

[15] Lung nodules (Pulmonary nodules): Diagnosis, causes & treatment. (n.d.). Cleveland Clinic. Retrieved July 3, 2022, from https://my.clevelandclinic.org/health/diseases/14799-pulmonary-nodules

[16] Whitaker, K. (2020). Earlier diagnosis: The importance of cancer symptoms. The Lancet Oncology, 21(1), 6–8. https://doi.org/10.1016/S1470-2045(19)30658-8

[17] Team, K. (n.d.). Keras documentation: Keras applications. Retrieved July 3, 2022, from https://keras.io/api/applications/

[18] Masood, A., Yang, P., Sheng, B., Li, H., Li, P., Qin, J., Lanfranchi, V., Kim, J., & Feng, D. D. (2020). Cloud-based automated clinical decision support system for detection and diagnosis of lung cancer in chest ct. IEEE Journal of Translational Engineering in Health and Medicine, 8, 1–13. https://doi.org/10.1109/JTEHM.2019.2955458

[19] Luna16—Grand challenge. (n.d.). Grand-Challenge.Org. Retrieved July 3, 2022, from https://luna16.grand-challenge.org/Data/

[20] Luna16—Grand challenge. (n.d.). Grand-Challenge.Org. Retrieved July 18, 2022, from https://luna16.grand-challenge.org/Tutorial/

[21] Luna16—Grand challenge. (n.d.). Grand-Challenge.Org. Retrieved July 18, 2022, from https://luna16.grand-challenge.org/Results/