A machine learning algorithm that classifies the stages of cancer based on medical images taken by radiologists will have two main types of data input. Medical images taken by radiologists follow mainly the Digital Imaging and Communications in Medicine (DICOM) standard depicts a protocol through which one can transmit non-proprietary medical image data and related information (Bidgood et al., 1997). Although there are DICOM files, the DICOM standard does not pertain to the resulting file, but rather the procedure required to transfer data from one computer to another securely (Bidgood et al., 1997). By using this standard, one can use seamlessly analyze data, supplement it with further information from predictions, and transfer it to either a complimentary app or any application of the radiologist’s preference.

Although the DICOM standard functions to transmit the data, the data it transmits is simply imaging and textual in nature. The DICOM format specified in part ten of the DICOM standard is composed of a header and image data set packaged within a single file (Varma, 2012). The first data point within the DICOM image file is composed of the header, which stores basic information related to the patient as well as certain information pertaining to the minimum requirements for a computer to appropriately display the image (Varma, 2012). This data will be used as categorical fields within the machine learning algorithm to further support the prediction together with the image data. This also provides security to the algorithm as separating the header from the image file will not allow the computer with the capability to read and display the image in the end at the cost of creating a large file. This is the reason as to why tools such as the Imaging Data Commons have appeared. This tool exists to empower data scientists and provide them with the ability to access data for research (*Cancer Research Data Commons | CRDC*, n.d.). In the end, the data processed from the DICOM files represents the patient and the affected body part of the patient.

To process the data and understand the state of the patient, one will require the python programming language in conjunction with the Pydicom and TensorFlow libraries. Although TensorFlow is not the only machine learning library, it contains straightforward instructions and is faster to design than its alternative (PyTorch) and in this case is the optimal choice. Pydicom is a python library whose main purpose is to allow for control of the DICOM standard digital workflow so that one may build tools using the said workflow (*About Pydicom |*, n.d.). TensorFlow is a machine learning platform used by many to develop machine learning algorithms with the flexibility of training the model through either the GPU, CPU, or even a TPU, it can also preprocess data before training and assist in the creation of data pipelines, optimize model training and more through its multitude of tools (*What Is TensorFlow?*, n.d.). With Pydicom and TensorFlow, one can create a machine learning algorithm that classifies tumors based on DICOM files.

With Pydicom, one can load the data for processing and reading without having to change its DICOM format nor separating the header data found within the DICOM file. The Pydicom library will be used to remove any unnecessary fields, edit the header data to fit a standard (i.e., make all letters lowercase), and create a copy of the DICOM file containing the predictions of the machine learning algorithm. Pydicom will potentially be used for decompressing compressed pixel data from images as well, and TensorFlow will be used to load the resulting NumPy array and finally process the image data by dividing each pixel value by the max pixel value, then the data will be augmented by resizing the image and rescaling randomly. These preprocessing steps will be included as part of the model for the sake of normalizing all input data before the model can classify the sample. To provide an appropriate comparison, separate models will be created to compare metrics, one model without preprocessing layer, one with singular pair of convolutional layer and pooling layer and more. The model will be also composed of three layers of convolutions paired together with averaging pooling layers to extract majority of the features related to the image data. Once the key features from the image data have been extracted, the textual data will be encoded by a simple categorical encoder to be passed through a dense (or linear) layer that further reduces the number of features in two stages. Finally, the output will then be used to determine whether the tumor is malignant or not (or what stage the tumor is on).

The aforementioned model will be a template with certain variations (i.e. more convolutional layers vs the three, or average pooling layer vs max pooling layer) which will undergo training and validation through TensorFlow’s training and validation algorithms. Accuracy, AUC/ROC, and the confusion matrix will be used in conjunction to measure the performance of the machine learning algorithms. The goal of the algorithm is to have performance like doctors at diagnosing cancer. According to a study published in 2017, approximately 20% of patients with serious conditions have been misdiagnosed (Bernstein, 2017). The end goal of the model created is to have an accuracy of at least 80% to consider the model a success, while the true goal for the model to be considered a complete product would be 95%. Although the model may be more accurate at diagnosing cancer, it may be more difficult to accept the results from a machine rather than a human being as it is expected that a machine should perform better than a human being when possible. Once the model is finished making the predictions, it can now move on to visualizing the results.

The model will not only classify the image, but also highlight the general area where the tumor may be. This model will work in the background of an app that will allow the user to load either a singular file or a multitude of file through an interface where one is able to load any number of DICOM files through the press of the button. A preview depicting the names of the DICOM files, together with a predict button at the bottom right will be displayed. Once the predict button is pressed, a progress bar will be used to show the progress of the model at making predictions upon the loaded files. After the predictions are finished, the images will be displayed with the possible tumor annotated within the said images. A note at the bottom right each image will display the possible stage of the tumor or cancer and a separate field on the header data will also contain the tumor stage. Multiple images will be cyclable, and a save button will also be available to save on the work provided. This visualization will be done to allow radiologists with the ability to observe for themselves whether the classification matches what they are seeing, in the case where the prediction is inaccurate a report button will be put in place that will send the finalized DICOM file, anonymized using the Pydicom data for further analysis and retraining of the model. The main point of the data model will be focused upon the model, which will be proprietary and inaccessible to the user. The user will be able to use the model offline, without the capability of creating its own model.

Source(s):

*About Pydicom |*. (n.d.). Pydicom.github.io. Retrieved September 14, 2022, from https://pydicom.github.io/about

Bernstein, L. (2017, April 3). 20 percent of patients with serious conditions are first misdiagnosed, study says. *The Washington Post*. https://www.washingtonpost.com/national/health-science/20-percent-of-patients-with-serious-conditions-are-first-misdiagnosed-study-says/2017/04/03/e386982a-189f-11e7-9887-1a5314b56a08\_story.html

Bidgood, W. D., Horii, S. C., Prior, F. W., & Van Syckle, D. E. (1997). Understanding and Using DICOM, the Data Interchange Standard for Biomedical Imaging. *Journal of the American Medical Informatics Association*, *4*(3), 199–212. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC61235/

*Cancer Research Data Commons | CRDC*. (n.d.). Datacommons.cancer.gov. Retrieved September 14, 2022, from https://datacommons.cancer.gov/cancer-research-data-commons

*Dataset basics: read, access, modify, write — pydicom 2.3.0 documentation*. (n.d.). Pydicom.github.io. https://pydicom.github.io/pydicom/stable/tutorials/dataset\_basics.html

*DICOM Library - Anonymize, Share, View DICOM files ONLINE*. (n.d.). Www.dicomlibrary.com. Retrieved September 14, 2022, from https://www.dicomlibrary.com/dicom/dicom-tags/

Varma, D. (2012). Managing DICOM images: Tips and tricks for the radiologist. *Indian Journal of Radiology and Imaging*, *22*(1), 4. https://doi.org/10.4103/0971-3026.95396

*What is TensorFlow?* (n.d.). NVIDIA Data Science Glossary. Retrieved September 14, 2022, from https://www.nvidia.com/en-us/glossary/data-science/tensorflow/#:~:text=Heavily%20used%20by%20data%20scientists