An assessment of the AI Data Readiness of the National Lung Screening Trial Data

Agnes McFarlin

03/21/2024

Abstract

The purpose of this report was to assess the AI readiness of the publicly available data from The National Lung Screening Trial (NLST) dataset. Specifically, by using the data available to train a machine learning model to identify Cancerous lung nodules without the presence of annotated slides for reference. Data was obtained, then transformed into a format easily used by a machine learning model. Training and testing were performed, and metrics were quantified. Qualitative inferences were first made about the data, then quantitative inferences were attempted.

Background:

The National Lung screening trial (NLST) was a large study involving more than 50,000 patients over several years. Patients were screened three times annually and were assigned to receive either low dose-helical CT scans or standard chest X-rays. This was done to assess whether those exposed to low dose CT scans had better outcomes than those exposed to X-rays. [1] The study indicated that patients who received low dose helical CT scans had a lower risk of dying from lung cancer than those who received X-rays only. [2]

The NLST is a large dataset of radiological images, the main data set consists of about 11 TB of data, which translates to about 26,000 different patient records. Each record is associated with 3 time points labelled T0,T1,T2. And each time point is associated with sometimes hundreds of images. But are these images useful in terms of machine learning? This paper attempted to address this question. A qualitative and quantitative analysis of the data was performed, and a machine learning model was trained on the data available.

Preface:

When finding references for what constitutes AI readiness for a data set, a paper published on the subject stated that, “Accuracy consists of well-defined labels as well as ground truth annotations for training and testing of ML models.” [3] While the NLST data had some annotation information retained in the form of a data set that gave the slide number in the collection for each patient where cancerous nodules were present(or largest). There were no medical annotations of images retained. Without having true expert labeled data to use the, the NLST data set might not be an ideal dataset for use with machine learning on its own. Which is a shame, it contains thousands of useful pieces of information and a mountain of images to work with.

I tried several sources to find annotated images, I reached out and asked a representative of the National Cancer Institute Cancer Data Access System if there were any annotated images. And they verified none were available. Also not all data was loaded to 7 bridges cancer genomics cloud repository. There was just too much. So a representative sample was uploaded instead.

Methodology

Data downloading and browsing:

The first step to data analysis of the NLST dataset was obtaining the data. And several sources were used: The first was the Imaging Data Commons, the second was the Cancer Imaging Archive and the third, the Cancer Data Access System. The data dictionaries were first obtained, and the needed patient information was gathered. And only then did actual downloading begin. Please see the pre-processing section.

The Imaging Data Commons (IDC) website itself was fairly straightforward to navigate. [4] There was no need to create a log in or request data access for the NLST data set. And images were simple to preview, with additional information being easily accessible. The images were also able to be explored as a whole and as individual slides right from the browser. For previewing images, the Imaging Data Commons was a great resource. But manually querying by more than a few case numbers was almost impossible, with no way to enter large lists of Patient IDs. Another thing the Imaging Data Commons lacked was supplementary information.

To obtain additional data (Data dictionaries and datasets) meant going to secondary websites. As the imaging data commons did not appear to host said data. The IDC instead linked to the Cancer Imaging Archive [5] for the needed information. The Cancer Imaging Archive provided links to resources such as publications and a summary of the study. The Cancer imaging archive also allowed the data to be browsed and previewed. And no log in was required for data that was already publicly available. Where it differed from the IDC was the ability to query specific patient records en-masse by using the data browser tool. Relevant patient ID numbers could be pasted into the search tool and if the data was available, patient records would be able to be downloaded via a provided manifest downloading tool.

To further complicate data collection, a third website was referenced and found, called the Cancer Data Access System. [6] Which contained some additional information when compared to the Imaging Data Commons and the Cancer Imaging Archive. For example, a nicely curated list of publications, and a list of approved NLST projects. Where the Cancer Data Access system differed the most was in stating that CT images may be requested by applying for a project [7] . While the other two sources allowed for the data to be freely downloaded.

Data downloading also varied in complexity depending upon the source. For the Imaging Data Commons, at first it seemed that the only way to download the data meant using a Google Collaboratory environment, and big-query tools as well as complex SQL syntax. But through much searching a simpler method was found. [8] Which used the idc_index package, and allowed the downloading of images directly, avoiding the use of the Collaboratory environment. The simplicity of the queries, made the process feel straightforward at first. But without the knowledge of how to perform certain types of queries, the process had to be abandoned. Even when specifying patient ID, Modality (CT), and the number of files per packet (most x-ray images for example contained just two files), too

many un-needed files were still downloaded. The un-needed files were very large composites of images, often many GB in size each. They were still CT scans, and contained the correct information but not needed at all. And as more and more files downloaded, the process slowed. After several attempts, a different method had to be utilized.

For the Cancer Imaging Archive downloading images consisted of either manually querying images in the data browser and downloading the NBIA data retriever which created a manifest, and then obtaining the images directly. [9] Or by using the Rest API. [10] The use of the NBIA data retriever tool and the image search function was much simpler than the REST API but, was limited by the design of the website itself. There was a text box in the website query tool where relevant patient IDs could be pasted. For very large amounts of data, this method might not have been feasible. But given how few patient records were originally needed, and due to simplicity, it was the method used. At times the manifest tool would attempt to download duplicate files despite a new cart being created and none of the previous files actually being in it. The solution to this was to reload the page and create a new manifest file. But in general, sets of several hundred patient records were pasted in and downloaded. The data used ended up being a small mixture, comprised of what was downloaded using the idc_index package and the Cancer Imaging Archive. The NBIA REST API was somewhat unclear, and therefore was not used.

There was no request for data access made with the Cancer Data Access System, so the ease of use was not assessed.

Data pre-processing:

None of the repositories contained any sort of annotated images for Lung Cancer Specifically. The Cancer Data Access System provided the following information: “CT images do not contain radiologist's findings. Findings detected in trial screening can be found in the Spiral CT Abnormalities or Spiral CT Comparison Read Abnormalities datasets. Coordinates of findings on CT screens were not collected by the NLST trial.” [7] The supportive datasets were downloaded first and used to create lists for downloading the images themselves. Due to the massive size of the NLST, simply downloading the whole data set was not reasonable or feasible given the 11 TB of files that were contained in the repositories of each location where the images were stored.

When loaded, the supportive data set, “Spiral CT Abnormalities” contained the slice numbers where the nodule or lesion was the largest as well as the study period time point. This meant querying for specific study numbers, and patient IDs by creating a list and narrowing it down. The supportive data dictionary was titled: “IDC: SCT Abnormalities: Data Dictionary”. As a note: I left the names of the data files the same in the challenge files as when I obtained them for the Cancer Imaging Archives, the data dictionaries reference the file names specifically and this might help with matching data set to dictionary, that

being said, I used simplified titles of the data sets in this paper.

There were parts to the data sets that were helpful to those with radiological training. When the “patients” data set and accompanying data dictionary “IDC: Participant: Data Dictionary” were loaded they contained information such as the actual quadrant where the cancer was located. As stated above, the coordinates of the locations of the lesions were not retained. But combined with the abnormalities data set, there could be some use there.

The data sets were loaded into Jupter/Python, the patients data set was analyzed first. The first step was choosing patients that were negative for cancer throughout the study. The study was divided into several time points T0, T1, T2. And it was decided that choosing patients that had negative screens for T0, T1 and T2 exams was the best way to minimize chances of positive scans being included in the negative sample group. The patients data set also contained columns for locations of lesions in the lungs specifically, which were to be marked with a 1 if cancer was found, 0 if it was elsewhere, and NaN (not a number) if there was nothing to report. Filtering for Cancer location values that were NA or NaN was one of the secondary checks to ensure positive and negative patients were not mixed together. No patient numbers were allowed to be in the negative group that were in the positive group. Since exact locations where nodules ended and started was not known it was the best way to prevent images with cancer from getting mixed in with those without cancer. Checks were performed through the study to ensure this was the case.

After the patients for the negative scan group were chosen, patients who would be used as the positive cancer scan group were chosen using the patients data set first. At first patients were sorted by all patients that were not negative for cancer screens at time point 0 (any variable but 1). Then further by those that had nodules, which was category 4 for scan results for time point 0. (A positive Screen) Then further by those who had nodules during each time point of T0,T1,T2. But it was found that there were no results of patients that had nodules for the third stage and had nodules for the other two stages. So it was decided to start with patients who had nodules at T0. Which originally yielded about 10,000 patients and felt very promising.

The results were then further filtered by the condition of ‘results of the screen associated with the first confirmed diagnosis’ being a positive result. As a secondary check, the staging results were confirmed to be non-empty. As a third check cancer location was confirmed to be non-empty. This left (N=838). Then the ‘abnormalitites’ data set was referenced and further filtered by patients that had nodules > 4 mm in diameter that were not calcified. Which was category 51 in the Abnormality Code number column, all other categories seemed to be non-applicable. Empty values were also filtered out of the slide number column. After the second filtering, there were about 600 patients left (N=620) with about 2000 slides worth of images.

Both datasets were then cleaned by checking for missing values and for repeats. The groups were then compared to ensure patient IDs were unique to each one. Since there was no way to query by slice number specifically within the data access systems that I was able to perform successfully, I downloaded entire patient data sets and obtained necessary images that way. This was done using a combination of the idc_index package and the Cancer Imaging Archive manual query tool. The data had to be carefully handled, since the data from the idc_index loaded as one large mass. While data obtained through the Cancer Imaging Archive resided in sub-folders organized by patient ID. Redownloading did have to occur a few times due to issues.

But as the data was downloaded, it was parsed by removing files that were not numerous enough, or just contained X rays, the number of applicable patients for the negative group grew smaller and smaller. After parsing the negative group of ~2,000 patient records the number of unique patient IDs that were still usable was less than 200. With looming time constraints, it was decided to continue with the present sample size so model development could begin. Obtaining the data for the positive cancer group had a fairly similar outcome. Some data was obtained through the Cancer Imaging Archive and some through the idc_index. The end number of patients for the positive group was between 500 and 600. And the number of applicable scans was about 1730. An equal amount of scans from the negative patient group was selected to create a total number of roughly 3500 scans.

Once the data was downloaded, and pre-processed, all of the files were read using the pydicom dcmread package which parsed the metadata. The following variables were selected: the time point, the instance number (which was the slide number), the patient ID, and all were added to a list. Then matching the contents of the resulting list to either the negative or positive patient records lists was performed. And if there was a match, the files were moved to separate folders for further processing. This was also the second check and was performed repeatedly through the study by scanning directories and ensuring the correct files existed and matched the pre-existing patient lists. Which proved useful when the virtual environment I was using started automatically setting the current working directory to the wrong drive. The next step of pre-processing was to add labels to data, if the file contained an image that was negative, it was labelled with an X, and a positive image was labelled with a Y. All images were then shoved into one folder to be further processed.

DICOM images ended up being particularly difficult to transform into formats able to be used by machine learning models. This prompted the transformation of said images into jpeg format. Which were simpler to transform into tensors for Machine Learning. The dicom2 jpg package was used to perform the needed changes. After the data transformation, the files were stripped of any naming convention, so the images were renamed by randomly generating file names and removed from their nested file structure. And once again thrown into a large mass, which was

shuffled and randomly chosen from to create training, validation and testing sets. Which were stored in separate folders.

Machine learning pre-processing:

The lack of annotations made it difficult to work with the data. If there were clearly marked images of cancer, each image could be cut into small quadrants, say 28X28 pixels. And have a very specific quadrant clearly labelled “Cancer Here”. But without this ability, and with no radiological experience of my own, I had to do the next best thing. I Improvised. I instead proceeded to shove 512X512 grayscale images into one end of a model and hoped accurate predictions came out of the other end. The DICOM images being transformed into jpeg meant huge amounts of quality loss. But try as I might, I could not figure out how to transform DICOM into tensors. So I relented and proceeded to continue onto other parts of the task. I first transformed my images into jpeg and linked them to a Pytorch data loader. I then proceeded to Frankenstein some form of a Pytorch Convolutional Neural Network into existence. It was slow and unstable, and if I made the batch sizes too big it would crash my computer, but it was created by me, so I was proud of my abomination. I packed together convolutional layers with some padding so every piece of the input could be read. I added some LeakyRelu because it is supposed to help with instability in the model. I also added some normalization layers, and made sure to have dropout included. So some of the inputs would randomly get turned off. And then shoved all ~8000 features into a linear transformer and hoped for the best. My optimizer ended up being Adam, because I have used it before and know I can rely on it. My loss function was binary cross entropy, but not the regular kind. The kind with LogitLoss, again for the whole, “My model predicts wildly” thing. In the hopes that wrong predictions do not get over corrected and confuse matters more. The model ended up being built using python 3.11.7 , and Pytorch. A transformer flipped the images, and ensured they were all 512X512, and I also applied interpolation of 0.2, since the images were gray scale and not rgb. Then all were transformed into tensors and a custom data loader class was created to handle the pre-processing of images.

Code requirements and instruction to run script:

The Final_submission_preprocessing_and_cleanup_only (3).ipynb takes the data in the main directory that I have loaded onto 7 bridges. And adds appropriate annotations based on whether the images are in a positive or negative patient list. And then converts the images to jpeg from DICOM and moves them into train test split folders. Just wanted to provide it to show my work.

The Quantitative_assessment.ipynb will run in the code space that is provided by seven bridges. It links to the data files located in my workspace. It is a jupyter notebook that consists of python code using whatever the latest version of

python is in seven bridges. When the notebook is opened and all cells are run. The end output is a pdf of metrics. Which will show up in the output section. It will probably not work outside of the 7 bridges environment since I had to specify the directories. All dependencies will be installed using !pip within the cells.

The ml_model.ipynb will take jpg files from the test folder, turn them into tensors and cast a prediction. And then some metric images will be stored in the output.

```
Python version 3.6, torch==2.0.0 to
rchvision==0.15.1 torchaudio==2.0.
1
```

Results and Conclusion:

This project was a great learning experience and I am glad I attempted it. I got to work in a virtual cloud network environment and see the interesting projects taking place there. I worked with complex data that was new to me. Faced issues I had never seen before and learned to overcome them. I did not have that much experience working with Pytorch and this was a great reason to do so. Overall, the model performed all right. Not sure it is going to break 70% accuracy. I was still training and modifying it as I was writing the conclusion for the rest of my paper. I was able to get my model to overfit on a small subset of training data, which led me to believe a larger scale would be applicable. And maybe with enough training my model will be able to learn the full training set. But I noticed sometimes it was overcorrecting after making a few correct predictions. Usually two very high accuracy predictions followed by a very low score. And it never flattened out in terms of converging loss between training and validation. Which was probably due to instability, the way I processed my data, or both. I performed at least 24 hours worth of epochs of training. Adjusting parameters, trying different learning rates. And in the end the loss and accuracy stabilized somewhat, but again, never did converge. While the results could be better, I understand Pytorch model architecture better now. So it was a worthy endeavor. Another issue was my reluctance to scale the images down properly. Since there were no annotations to mark specific locations on slides, I figured no cropping could occur. And due to the same factor, resizing to make images smaller during the transformation stage was also not performed, the point of interest being lost due to a shrunken image was something to be avoided. The images were all sized to be 512 X 512 if they were not, and flipped. But maybe random resizing would have produced better results. Another major issue was the lack of data in terms of sheer numbers and diversity. There were very few samples to work with for the positive group. I had no scans to add to balance the population because they did not exist. Additionally, I ended up with a smaller and less diverse negative population than I would have liked due to samples not meeting criteria or simply not existing. I was also unsure of how to properly balance the groups when I became aware that they were disproportionate. Which due to errors on my part, I realized too late. The best performance of the model was about

0.56 accuracy and F1 score of 0.63, still quite shy of the required 0.71. I would say my model failed to identify cancerous lung nodules in patients. But given the fact that I was able to get the model to overfit on some of the data meant training was possible. So either my skills in model optimization were lacking, or there needed to be more time allotted to training. But I do reverse my original assumption that the NLST is not a strong contender for the Machine Learning sphere. Given more time and a better understanding of the model architecture I am working with, I believe I would have been able to produce a much better result.

Qualitative/Quantitative Assessment

AI Data readiness involves several metrics which all interweave. Data quality measures how well a data set meets some set of metrics. In terms of completeness and accuracy the lack of fully annotated images made the NLST difficult to use for Machine learning tasks that involved identifying specific locations or regions where cancer was present. Also, when data was queried there were times when the result would come up with fewer items than what was searched for. Sometimes no results would return at all. That may have been due to errors in processing, or record keeping, but without downloading all 11TB of files, it was not possible to discern if the data was mis labelled or missing. So the completeness of the set of images in the repository was uncertain. Errors in processing not withstanding only 10% of the negative patient group that was queried actually came up with a result in the Cancer Imaging Archives. The positive patient group had more complete records, with almost all 620 records found. As a positive for accuracy, the available supplemental data sets were clearly linked to each other, and to the DICOM images using patient IDs, which were able to be matched to the DICOM images using their metadata. All of the data tables inspected and every single DICOM image queried, contained the information needed to perform analysis. Matching DICOM images to patient records was straightforward due to this. And patient IDs were unique to each person, there were no duplicates in any of the supplemental data sets used. By creating lists of cancer positive and negative patients, a few set operations were the only things needed to ensure patients from one set of samples did not end up in the other set.

Consistency between repositories was a bit of an issue, the locations of supplemental data sets (metadata) being different than the locations where the data was hosted was confusing.[5][4]. And certain sites had images that other sites did not. It may have been possible they existed but were not categorized.[4]

Another part of data readiness involved data governance. Which included data security and ensuring patient privacy as well as ensuring appropriate documentation. And anonymizing patient records was done very well, there was a paper written on it. [11] It was actually really interesting to read about such a large study being coordinated. Thousands of images were taken and anonymized and then physically

delivered to a specific location. Screening centers had 250GB external hard drives provided to them on which to store data. [11] Some of the ways they masked patient information included removing names and birthdates, as well as the time when the image was taken, gender and race were turned into numerical values. Age was not anonymized. The process of image handling also showed how seriously the study leaders and facilitators took the idea of governance and documentation. Each list of patients from a screening center went to a central location where it was matched against a core list and updated each month. A workflow was established and followed strictly by study teams. Roles were broken up and specified, having particular individuals to perform them. The information for the process was also publicly available due to the aforementioned paper. [11] Figure_of_anonymization_proccess.

Where the NLST was lacking in terms of AI readiness was diversity in the participants. While the ratio of male to female was closer to 60-40 than 50-50. The distribution of patients by race was heavily skewed in one direction. With about 90% of participants being white. This lack may cause a model to not transition well to new data. Since there were only so many records to use. When it comes to tasks such as facial recognition or photo classification for organizing images it is well understood that diversity is important. [12] So it is probably the case with medical images as well. It is also not known if a variety of patients were selected in terms of other characteristics such as height or weight, as these factors may affect bone density and mass. [13] The age range for participants was also quite narrow, with participants being between 55 and 74 years of age. [14]

Overall the NLST has a great amount of information. But the way data is accessed, and the organization of patient information would benefit most from some improvements.

From a quantitative standpoint:

For the quantitative portion of the project, data sets were assessed on a few factors. First, the breadth and depth, and the datatypes of columns. It was reasoned that if a data column contained a specific type of data for example integer or float it could stand to be a basic check for nonsense values and data type correctness. A check for duplicated columns was performed as well. (please see the Quantitative Assessment document).

Missingness was assessed by creating plots to illustrate missing values. As well as calculating null columns as a percentage. The number of null patient ID columns was calculated since this was the primary variable common to all data sets in the series.

A heatmap of nullity values illustrated the relationship between variables. In addition to this the Pearson product moment correlation coefficient was calculated and the overall percentage of highly positively correlated values was returned.

For the DICOM quantitative analysis a subset of the total samples was chosen, which consisted of the images that were going to be used for training and testing.

The slice thickness, the number of rows and columns, pixel spacing and patient gender were all collected in large lists. And compared.

The data sets analyzed included the patients dataset, the abnormalities data set, and the lung cancer data set in addition to DICOM images and the resulting jpeg images.

The patients data set was assessed first. (Section 1.4.1 in Quantitative Assessment.pdf) Which had information for 25,000 or so individuals from the study, and included demographics. The number of columns for the data set was 39, and the number of rows was 53,452. The data in each column fell into either float, integer, or object categories. There were 0 duplicate columns. There were no missing patient IDs, gender values, or age values. The actual % of null columns present was around 50%. But this was due to the fact that a significant portion of the columns were for logging cancer location data. With a 1 indicating primary tumor location in a specific area, and a 0 indicating presence but in a different location. Otherwise those specific columns were filled in with NA values. There were no duplicated patient records. The nullity matrix showed many of the variables had a strong relationship (denoted with a 1) of those most were in the cancer location category. Using the Pearson product moment correlation formula the percentage of highly positively and negatively correlated values was returned. In the patients data set about 36% of values were highly correlated.

Next to be assessed was the lung cancer data set. (Section 1.4.7 in Quantitative Assessment.pdf) Which showed information only for those patients that had a positive lung cancer diagnosis during the trial. There were 34 columns and 2150 rows. The column types consisted of float, integer and object. There were 0 duplicate columns. There were no missing patient IDs. The overall percentage of null columns was about 10%. And there were 92 entries that were a possible follow-ups. This was most likely due to repeat observations or scans. The data set showed very low nullity values. With more than 50% showing weak or no relationship. Using the Pearson product moment correlation formula the percentage of highly positively correlated values was about 5.6%.

The Spiral CT scan abnormalities data set was assessed after that. (Section 1.4.13 in Quantitative Assessment.pdf) Which contained information about each abnormality found during the study in the CT scan arm. The data consisted of 12 columns and 177,487 rows. The data types consisted of float, integer, and object. There were 0 duplicate columns. There were 0 null patient IDs and there were 152,970 entries that were either follow-ups or additional information. About 40% of columns contained null values. And most columns showed nullity close to 1. If cancer was found, more than likely, the slice number of the relevant scan and other information had to be recorded. Leading to a nullity value closer to 1. The percentage of highly correlated values was about 28.5%.

For the images:

The images for negative and positive patients were grouped and analyzed. (Section 1.4.19 in Quantitative Assessment.pdf) The shape of each data set was 10 columns and 1735 rows. And both had data types of integer, float and object. There were 0 duplicate columns. There were 0 Null patient IDs and the two columns that were null should have been. Study time and Patient's Sex were masked during the study. Otherwise none of the columns contain null or missing values.

Pixel values in the range of 0-255 were obtained for the converted jpg images in the positive and negative cancer groups. This was done out of curiosity. There were a few locations where the two groups differed. But the significance of this was not yet known.

Recommendations:

For the Cancer imaging Data Commons: Better organization of data, ensure metadata is in the same locations where patient records are. Provide more information on how to use the REST API for those less familiar. For example the Idc_Index would have been amazing if I had been able to specify I did not want composite images. Or images larger than a certain size. But I had no idea how to ask for that. I also had to really search to find that the idc_index existed to use, increasing visibility would be helpful.

Maybe a realignment of the data between the different storages. Purge non-existent patient records (if that is allowed).

BIBLIOGRAPHY

NLST Landing page - the Cancer Data Access System. (n.d.). https://cdas.cancer.gov/nlst/

https://cdas.cancer.gov/nlst/ [1]

National Lung Screening Trial (NLST). (2014, September 8). National Cancer Institute.

https://www.cancer.gov/types/lung/research/nlst

https://www.cancer.gov/types/lung/research/nlst [2]

Ng, Madelena Y et al. "Perceptions of Data Set Experts on Important Characteristics of Health Data Sets Ready for Machine Learning: A Qualitative Study." *JAMA network open vol. 6,12 e2345892. 1 Dec. 2023, doi:10.1001/jamanetworkopen.2023.45892*

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10692863/ [3]

The Cancer Imaging Data Commons Data Portal

https://portal.imaging.datacommons.cancer.gov/explore/ [4]

The Cancer Imaging Archive Collections. (2024, January 10). NLST - *The Cancer Imaging Archive (TCIA). The Cancer Imaging Archive (TCIA).*

https://www.cancerimagingarchive.net/collection/nlst/

https://www.cancerimagingarchive.net/collection/nlst/ [5]

NLST – A summary. (n.d.-b). https://cdas.cancer.gov/nlst/

https://cdas.cancer.gov/nlst/ [6]

NLST_IMAGES - the Cancer Data Access System. (n.d.-b). https://cdas.cancer.gov/nlst/

https://cdas.cancer.gov/learn/nlst/images/ [7]

Learn to Download Data- IDC USER GUIDE - Cancer IDC

https://learn.canceridc.dev/data/downloading-data

https://learn.canceridc.dev/data/downloading-data [8]

Access Data- Cancerimagingarchive.net

https://www.cancerimagingarchive.net/access-data/ [9]

Downloading TCIA IMAGES – Cancer Imaging Archive help center

https://wiki.cancerimagingarchive.net/display/NBIA/Downloading+TCIA+Images [10]

Clark, Kenneth W et al. "Collecting 48,000 CT exams for the lung screening study of the National Lung Screening Trial."

Journal of digital imaging vol. 22,6 (2009): 667-80. doi:10.1007/s10278-008-9145-9

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3043737/ [11]

BBC News. (2015, July 1). "Google apologises for Photos app's racist blunder."

BBC News. https://www.bbc.com/news/technology-33347866

https://www.bbc.com/news/technology-33347866 [12]

Kim, Sang Jun et al. "Relationship between Weight, Body Mass Index and Bone Mineral Density of Lumbar Spine in Women."

Journal of bone metabolism vol. 19,2 (2012): 95-102. doi:10.11005/jbm.2012.19.2.95

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3780918/ [13]

"Reduced Lung-Cancer Mortality with Low-Dose Computed Tomographic Screening." (2011). *The New England Journal of Medicine*, *365*(5), 395–409.

https://www.nejm.org/doi/10.1056/NEJMoa1102873?url_ver=Z39.88-2003&rfr_id=ori:rid:crossref.org&rfr_dat=cr_pub%20%200www.ncbi.nlm.nih.gov [14]