

BIRKBECK, UNIVERSITY OF LONDON

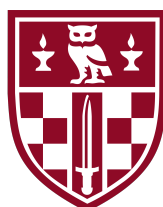
Pneumonia Detection from Chest X-Ray Images

Author:

Baran Buluttekın

Supervisor:

Dr. George Magoulas



*A project proposal submitted in fulfillment of the requirements
for the degree of MSc Data Science*

in the

Department of Computer Science

June 12, 2019

Declaration

I have read and understood the sections of plagiarism in the College Policy on assessment offences and confirm that the work is my own, with the work of others clearly acknowledged. I give my permission to submit my report to the plagiarism testing database that the College is using and test it using plagiarism detection software, search engines or meta- searching software.

The proposal may be freely copied and distributed provided the source is explicitly acknowledged.

Abstract

This Data Science project will be focusing on classification of X-ray images labelled based on patients with or without pneumonia. This proposal document will outline logistics and decisions made for the project in sequential order. First section of the proposal will give general facts about pneumonia and highlight important details about the related work carried out within the industry. More specifically related pneumonia detection research will be examined and how this research effected decisions of this project. Section 2 will give clear definition for the aim and objectives of this project. Third section of the proposal will give reasoning of the dataset of choice and explains why this specific dataset is chosen amongst others. Section four will give a quick introduction to artificial neural networks (ANN's) and some background information about the benchmark neural network architectures. Section five details each objectives and explains how they will be implemented as well as answers the question of why this specific machine learning field is chosen. Sixth section list the tools and techniques considered to be used and some brief reasoning behind them. Lastly, final section will layout time frame for the execution of the project.

Contents

Declaration	i
Abstract	ii
1 Introduction	1
1.1 Related work	1
2 Project Aim and Objectives	3
2.1 Aim	3
2.2 Objectives	3
3 Dataset	4
3.1 General guidelines while deciding on the dataset	4
3.2 OpenI database	4
3.3 ChestX-ray8	4
3.4 Cell press research	5
4 Computer Vision	6
4.1 Convolutional neural networks (CNN's)	6
4.2 Prominent computer vision architectures	7
4.3 LeNet-5	7
4.4 AlexNet	7
4.5 VGGNet	8
5 Methodology	9
5.1 Computer vision technique for this project	9
5.2 General data exploration	9
5.3 Data pre-processing and augmentation	9
5.4 Building baseline model with neural network architecture	9
5.5 Using pre-trained neural network to increase model performance	9
5.6 Visualizing neural network to ensure learning	10
5.7 Model ensembling	10
5.8 Applying different deployment options	12
6 Tools and Techniques	13
7 Project Plan	14
References	17

1 Introduction

Pneumonia is swelling (inflammation) of the tissue in one or both lungs. It is usually formed at the end of breathing tubes of the lungs and cause these tubes to inflame and fill up with fluid. In the UK, pneumonia effects around 8 in 1000 adults each year [23]. Global economic cost of pneumonia has estimated at \$17 billion annually [18]. Currently detecting pneumonia cases heavily relies on chest X-ray image examination which requires expert radiologists to diagnose. Building intelligent system to diagnose the pneumonia can help health care services to increase efficiency, reduce costs and could help increase early diagnoses in countries with inadequate access to healthcare.

1.1 Related work

There are number of research has been published about lung diseases related detection. Most preminent ones are the CheXNet [24] and ChestX-ray8 [30], both of these research carried out by training on same dataset ChestX-ray8 [30]. ChestX-ray8 comprises of approximately 100,000 frontal view chest X-ray images labelled by extracting information from the accompanied radiologists notes with using variety of different NLP (Natural language processing) techniques from the OpenI[20] database. ChestX-ray8 authored by researchers from National Institute of Health (NIH) and published at 2018. Most profound effect of this paper is the creation of the ChestX-ray8 dataset which has become one of the widely used dataset in computer vision research related to lung diseases. More detailed information about the dataset can be found in dataset section of this proposal.

CheXNet is another related article authored by researchers from Stanford University ML group. Prediction of lung diseases achieved by 121 layer convolutional neural network and designed to predict 14 pathologies in the ChestX-ray8 dataset. One of the major importance of this paper is the setting the setting benchmark for human level detection for chest X-ray images. One of the most fundamental difference of the X-ray related disease prediction is the definition of human level accuracy. Due to the nature of required expertise in X-ray images leaves general public out of the scope when it comes to human level performance of these pathologies. Anyone who have not been trained in radiology will not be able to detect any lung diseases in the Chest X-ray images. For example the image below is sample of two chest X-ray images almost indistinguishable to general audience.

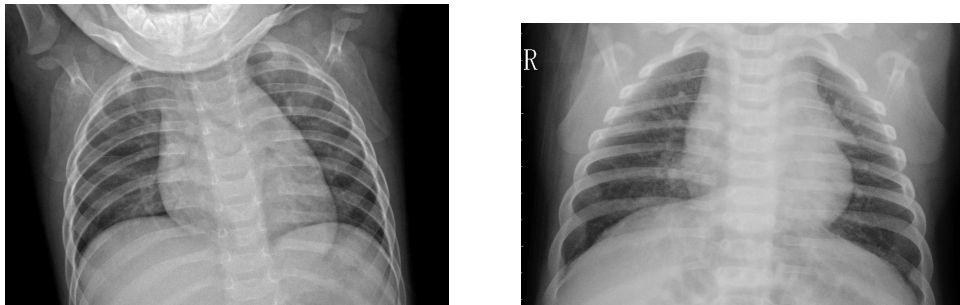


Figure 1: Two sample X-ray Chest images with and without pneumonia.

Given this challenge authors of the CheXNet conduct a test to establish benchmark for radiologists. They have collected 420 frontal chest X-rays and asked practicing radiologists in Stanford University to label them for all 14 pathologies. Radiologists selected with different range of experience, and had 4, 7, 25 and 28 years of experience. X-ray images presented to radiologists without any patient information or any symptoms experienced by patients and their diagnoses predictions measured based on underlying state of the X-ray patients. Following is the table showing the summary statistic of this test for the 4 radiologists participated to test on F1 score, which is harmonic average of precision and recall.[24]:

	F1 Score (95% CI)
Radiologists 1	0.383 (0.309, 0.453)
Radiologists 2	0.356 (0.282, 0.428)
Radiologists 2	0.365 (0.291, 0.435)
Radiologists 4	0.442 (0.390, 0.492)
Radiologists Avg	0.387 (0.330, 0.442)

Table 1: Radiologist prediction performances from CheXNet [24].

Importance of this test is that it gives us a rough estimate for human level accuracy benchmark to assess the model performance for new detection models.

2 Project Aim and Objectives

2.1 Aim

Aim of this project is to build a fully functional chest x-ray image classification pipeline that implements CI/CD principals to experimentation and deployment. Main classification algorithm I will use for this project is the neural networks, specifically convolutional neural networks. Highest possible accuracy will be aimed but due to highly iterative and time consuming nature of the neural network research, its not very likely that it will beat general benchmark set by most recent research.

2.2 Objectives

Project will be implemented with execution of fallowing objectives:

- **Carrying out general data exploration:** This part involves general check on dataset.
- **Data pre-processing and augmentation:** Preparing the data for model ready state.
- **Building baseline model with well known neural network architectures:** This step involves setting additional benchmarks with out of the box models from section 4.
- **Using pre-trained network to increase model performance:** Using pre-trained networks to help training and accuracy of the model.
- **Visualizing neural network to ensure learning quality:** For making sure model learning as intended and focusing on correct parts of the image.
- **Model ensembling:** Using ensemble method with different neural network architectures.
- **Applying different deployment options:** Implementation of different deployment options. Based on their trade offs.

3 Dataset

Choosing and processing dataset have a crucial importance on success of the any machine learning task. There are several dataset available online that relate to chest X-Ray images. Given the large number of choices for selecting the dataset there are few criteria important to check while deciding the final dataset.

3.1 General guidelines while deciding on the dataset

In this section I have highlighted my reasons for deciding on the dataset of choice for this research project. Main points for decision are:

1. **Reproducibility:** Dataset of choice must allow reader to reproduce the work in order to assess all the points discussed in the report. That would require dataset to be public.
2. **Labelling:** Dataset must contain labels of patients state. Such as being diagnosed with pneumonia or not.
3. **License:** Dataset should have a license that permits research.

I will be evaluating dataset available while considering general guidelines outlined above.

3.2 OpenI database

OpenI[20] is a database that is service of National Library of Medicine. It enables search for medical images, graphs and charts through text as well as image query. As of writing of this proposal it has over 3.7 million images, and 3,955 radiology reports. It is the main source for ChestX-ray8 dataset mentioned previously in Related Work section. Despite the fact it contains very large data for chest X-rays, this source is not suitable for this project due to the fact that images does not includes labels for the patients state (e.g., Pneumonia or normal). Chest X-ray data in this database is the image accompanied by radiologist report which is advisory document. Mainly because the lack of labels this dataset is not suitable for this project where the choice will be a supervised classification task.

3.3 ChestX-ray8

This dataset created part of the ChestX-ray8[30] paper (Also known as ChestX-ray14). Original source of this dataset is OpenI[20] medical database as mentioned in previous subsection. Authors first short-listed eight common thoracic pathologies, then related X-rays images searched from the database based on these pathology keywords. Most positive quality of this dataset amongst the other options is the sheer quantity of the data points which is by far the largest in size. This attribute especially important when it comes to certain computer vision techniques such as Neural Networks due to the fact that large datasets increases the variance and enables better generalization. Despite this positive points, quality

of the labelling of this dataset has come to questioning by radiologists[19]. Due to this considerable labelling inconsistencies this dataset have not been chosen for this project.

3.4 Cell press research

Shortcomings of first two datasets compelled me to searched further for new dataset that does not have the problems I point out previously. Research from Cell Press[11], together with the data that made public, provided a solution to these problems. Dataset the team released contains 5856 hand labelled X-ray chest images from children aged between one to five years old. X-ray images collected as part of the routine clinical care at the Guangzhou Women and Children's Medical Center, Guangzhou, China. All images screened for quality control and low quality or unreadable x-rays discarded. Labels of the images also checked by two expert physician and only approved images included in the dataset. Finally this dataset released under Creative Commons license Attribution 4 (CC BY 4.0) that allow copy, distribution of the material as well as transform and building upon material for any purposes. In light of all these attributes I choose this dataset for my project.

4 Computer Vision

Importance of X-ray image analysis in pneumonia diagnoses clearly highlights that this is a computer vision problem. In essence computer vision is a scientific field aims to automate vision task usually performed by humans. Vision on earth begin approximately 543 million years ago when trilobites developed basic vision system [13]. Development of vision helped increase the specie variation and development on earth significantly in respect to reproduction, finding food and many other reasons. Largely due to this very important nature of vision, research in how vision performed in species and how to automate the vision task gained a lot of attraction. Early research in this field inspired by the biological vision system. More specifically model of mammal neural system in respect to vision was the center point. In 1959 experiment on cat visual system by Hubel and Wiesel [9] highlighted the inner functioning of vision by discovering effect of dark edges causing activation in visual cortex. They also concluded that visual cortex passing this signals from detected edges to later centers of the brain where those edges are combined to represent more complex shapes. This idea of simple to more complex neural structure inspired Japanese computer scientist Kunihiko Fukushima to propose system he called *Neocognitron* [7]. In the article he laid out simple to complex neural architecture which was very similar to todays convolutional neural network (CNN).

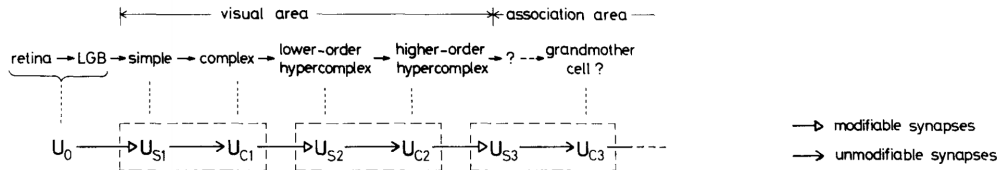


Fig. 1. Correspondence between the hierarchy model by Hubel and Wiesel, and the neural network of the neocognitron

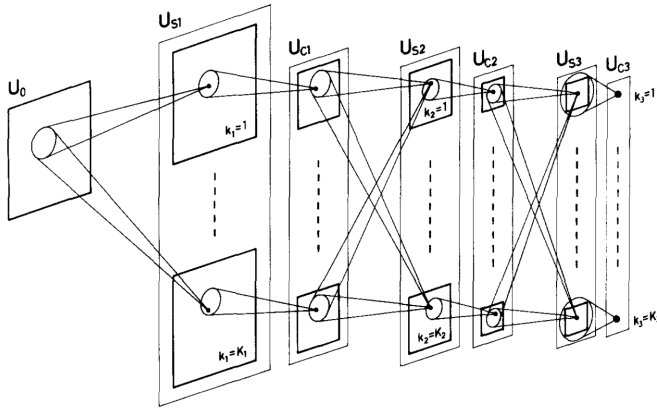


Fig. 2. Schematic diagram illustrating the interconnections between layers in the neocognitron

Figure 2: Network architecture from Neocognitron.

4.1 Convolutional neural networks (CNN's)

Although Neocognitron formed the general idea of the convolutional neural networks, it did not use the method called backpropagation that CNN's use today. Backpropagation is technique used in neural networks to propagate errors through

the layers of neural network. First CNN that have the attributes same as current CNN's was build in Bell Labs in 1989 to recognize the hand written digits of the zip codes [14]. Convolutional neural network name indicates that network uses mathematical operation called **convolution**. Explaining in a simple way using definition from the Deep Learning book [8]:

”Convolutional networks are simply neural networks that use convolution in place of general matrix multiplication in at least one of their layers.”

Convolutional neural networks are generally a good option for image or time series data problems because of the sliding window approach capturing the underlying signal.

4.2 Prominent computer vision architectures

In this project I will be taking advantage of the well known network architectures for computer vision for general benchmarking purposes. These architecture generally known for their good performance in image classification competitions hence, any new design should perform better than these designs to be considered.

4.3 LeNet-5

LeNet-5 [15] is 7 layered neural network build for classifying hand written digits in checks. Numbers in checks turned into 32×32 pixel images and feed into the network for image classification task. LeNet-5 is the first successful application for combination of CNN and backpropagation. Network representation illustration and full specification of the network outlined below.

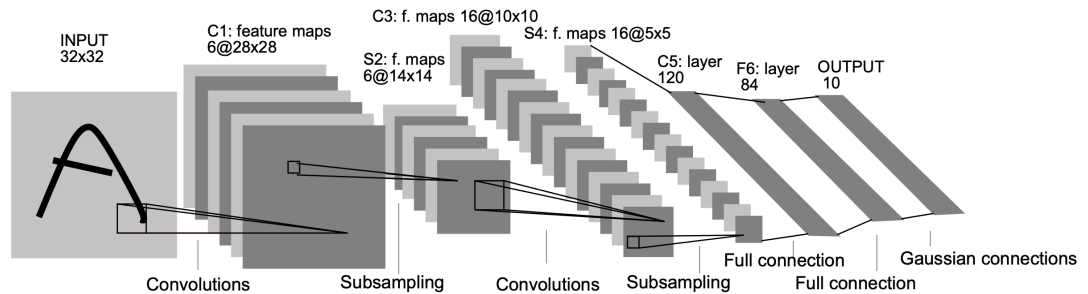


Figure 3: Network architecture of LeNet-5.

Credit: LeNet-5 [15]

4.4 AlexNet

AlexNet [12] designed and named after Alex Krizhevsky, and published with collaboration of Ilya Sutskever and Geoffrey Hinton (advisor). It was designed for ImageNet challenge [4] and utilized the idea of combining neural networks with

graphics processing units (GPU) for high performance. This combination overcome the restriction of high computational demanding nature of neural networks and proved that current computational techniques are sufficient for training neural networks in even very large dataset such as the ImageNet. Another important implication is that AlexNet brought neural networks back to spotlight for research community after long break from LeNet-5's results.

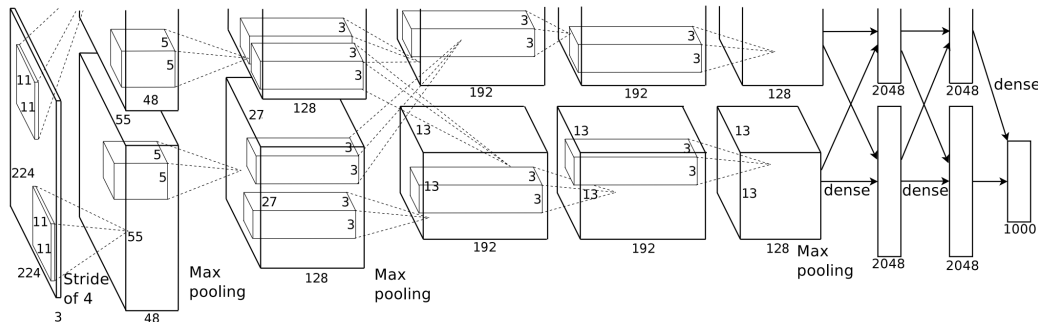


Figure 4: Network architecture of AlexNet.
Credit: AlexNet [12]

4.5 VGGNet

VGGNet [27] designed by Visual Geometry Group (VGG) from University of Oxford, this architecture also participated in ImageNet challenge [4] for the year 2014. This research investigated the depth of the convolution neural networks effect on their accuracy. Doing so article walks through their general layout for experimenting different depth and characteristic of neural networks with comparison. Trial error approach they assumed, presents useful methodology for network prototyping this project will use. After obtaining high accuracy from this new deeper neural network inspired many similar high depth neural network architectures to be build. Success of these deep neural networks also get popular which lead the new popular term *Deep Learning*.

5 Methodology

5.1 Computer vision technique for this project

In this project neural networks chosen as primary algorithm for the image classification problem. Reasons for that decision is the achievements of the neural networks withing the machine learning field, specifically within the image classification. So far neural network approach achieved near human level image classification, speech recognition, machine translation and more. Although decision to choose neural network for primary method for this project established, classification accuracy will be compered with other machine learning techniques throughout this project.

5.2 General data exploration

As a first step I will start with high level checks on data. This step involves general data exploration such as checking if there is a class imbalance in the dataset or size and quality related issues in some images.

5.3 Data pre-processing and augmentation

Many images can have inherent sources of variance that likely to effect the performance of the classification tasks. Some example to these sources of variance are, angle variance, contrast variance and positional variance. Process such as histogram equalization will be apply to images to increase contrast in the images. This process will lead to information difference between bone any different tissues become more prominent. Various image processing tool available in the keras [3] and scikit-image [29] libraries for this processes. I will also make use of some classic data augmentation techniques to avoid over-fitting. One of the most common technique is to flip the images left and right also flip them 90, 180 or 270 degrees.

5.4 Building baseline model with neural network architecture

Apart from using the CheXNet [24] radiologist accuracy benchmark, architectures from section 4 will be trained to set a baseline benchmark accuracy. Because of their well established position in the machine learning field detailed specifications of these architectures are widely available. In fact keras library [3] have readily available implementation of these models.

5.5 Using pre-trained neural network to increase model performance

A common and effective technique in computer vision is to use pre-trained network to train a new network. This approach especially successful when the dataset is relatively small. Pre-trained network is usually trained in another large scale image

classification data such as ImageNet [4]. Features learned from this dataset acts as a high level model and help training of the new network. Generally this process carried out by keeping the convolutional layers (also referred as convolutional base) of the pre-trained network and connect that layers to a dense (fully connected) layer thats not been trained. Then this newly designed network will be trained on the x-ray images.

5.6 Visualizing neural network to ensure learning

It is import to visualize the neural net to identify which parts of the image it's emphasize. Wide range of different visualization approaches available for visualizing convolution networks. Two selected approach that to be applied are the fallowing:

- *Visualizing intermediate convolution network outputs.* Generally useful for understating how convolution networks transform their input.
- *Heatmaps of class activation mapping (CAM).* This visualization technique is most relevant to x-ray image classification. It emphasize part of the image identified as member to certain class which is way of localize objects in the image.

More specifically *Gradient-weighted Class Activation Mapping (Grad-CAM)* [26] method will be applied to see right part of the image is emphasized by the network and not the unrelated noise within it. Grad-CAM is derived from proposal of *Zhou et al.* [32] called *Class Activation Mapping*. This technique is achieved by taking feature map of the last convolutional layer and calculating the gradient of class with respect to feature maps of the convolutional layer. Resulting map of different strength of activations will serve as heatmap that points to pixels that most influential in the classification.

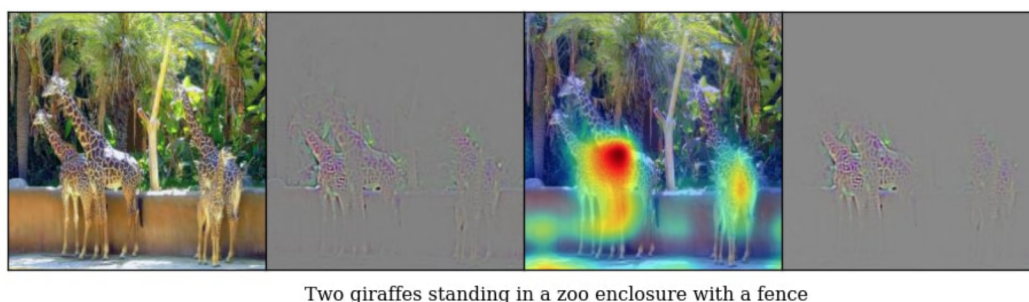


Figure 5: Grad-CAM visualization pinpoints locations in the image related to classification of the giraffes. Credit: Grad-CAM [26]

5.7 Model ensembling

Ensemble is a classic machine learning method where different models will used together in prediction to gain better result. Main assumption of the ensemble method is independent models will bring different bias to overall prediction where

the individual mistakes of the models will be mitigated by the prediction of other models. Effective application of ensemble requires different models to be trained on data and not having simply using different initialization in of the same neural network. Bagging [2] and boosting [5] are the most widely used ensemble methods.

Bagging is an acronym for *bootstrap aggregating*, it involves averaging the many predictors to achieve a more accurate prediction. In general this algorithm can be summarize as the fallowing form for the input vector x_i , number of predictors N and probability prediction of j th model $p(x_i)_j$. Then the prediction of ensemble \hat{p} can be calculated as:

$$\hat{p} = \frac{1}{N} \sum_{j=1}^N p(x_i)_j$$

In this project different neural network architectures will be trained to form an ensemble model. Method for the ensemble will be averaging and weighted average of the individual models. In case of weighted average, weights will chosen based on the individual classifiers accuracy. In other words highly accurate models will be given higher weight.

Boosting is ensemble technique of combining set of weak learners to build strong classifier. Most well known algorithm of boosting is *Adaptive boosting* (AdaBoost) [6]. AdaBoost ensembles classifiers by emphasising on the mistakes of the individual classifiers measured by the loss.

Algorithm AdaBoost.M1

Input: sequence of m examples $\langle (x_1, y_1), \dots, (x_m, y_m) \rangle$ with labels $y_i \in Y = \{1, \dots, k\}$
weak learning algorithm **WeakLearn**
integer T specifying number of iterations

Initialize $D_1(i) = 1/m$ for all i .

Do for $t = 1, 2, \dots, T$

1. Call **WeakLearn**, providing it with the distribution D_t .
2. Get back a hypothesis $h_t : X \rightarrow Y$.
3. Calculate the error of h_t : $\epsilon_t = \sum_{i: h_t(x_i) \neq y_i} D_t(i)$. If $\epsilon_t > 1/2$, then set $T = t - 1$ and abort loop.
4. Set $\beta_t = \epsilon_t / (1 - \epsilon_t)$.
5. Update distribution D_t : $D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \begin{cases} \beta_t & \text{if } h_t(x_i) = y_i \\ 1 & \text{otherwise} \end{cases}$
where Z_t is a normalization constant (chosen so that D_{t+1} will be a distribution).

Output the final hypothesis: $h_{fin}(x) = \arg \max_{y \in Y} \sum_{t: h_t(x)=y} \log \frac{1}{\beta_t}$.

Figure 6: AdaBoost algorithm [6]

Using AdaBoost to build ensemble models with neural networks has been applied successfully in the past [17, 25]. Implementation of boosting algorithm similar to AdaBoost will also be implemented using weak neural network classifiers.

5.8 Applying different deployment options

This is a very broad topic with many different deployment options and design choices available. It is not possible to cover all available options within the scope of this project. Therefore selected deployment options will be implemented.

After the training of model completed project moves to process of saving the trained model in standalone file that could be used to make predictions. Most cost effective and computationally cheap option is to let prediction happen in the client side of the user. This can be achieved by using JavaScript in the web page that will be served to end user. All modern browsers support JavaScript code to be run in the browser which will enable code and model prediction to run in users device. This option is within reach of the project due to the software packages like TensorFlow.js [28] that enables trained model to be converted into JavaScript completable file that will be served in static website. Although this option cheap and easy way to provide smart solutions to everyone, it is not suitable for commercial initiatives due to fact that anyone can extract the trained model from source code of the page and use it without creators knowledge.

Most widely used option when it comes to deployment machine learning models is building RESTful API to serve incoming requests from a central server(s). This deployment will be build using open source server implemented on the cloud. Reason for choosing the RESTful approach is to building system loosely coupled from the front end design and to enable it to be implemented into any third party web application systems.

6 Tools and Techniques

For the implementation of the aims of this project python programming language is chosen as a main programming language. Reasons for this decision is two fold, first part of the reasoning is need of high level programming language. Low level programming languages such as Java or C++ are not well suited for computer vision tasks such as this project due to reason of their time consuming prototyping cycles. Second part of the decision python being de facto language of choice for majority users which enables more tools and techniques being available for application.

I will also make use of external open source machine learning packages because of the intensive computational nature of the Neural networks. Number of parameters for some of the well known neural network architectures reaches to hundreds of thousands or in some cases in millions or billions. Therefore any code that implements these architecture required to be well optimized and preferably parallelized to run in high performance hardware, such as the GPU units. Building a code base that achieves this standards require significant amount of time and resource, hence out of the scope of this project.

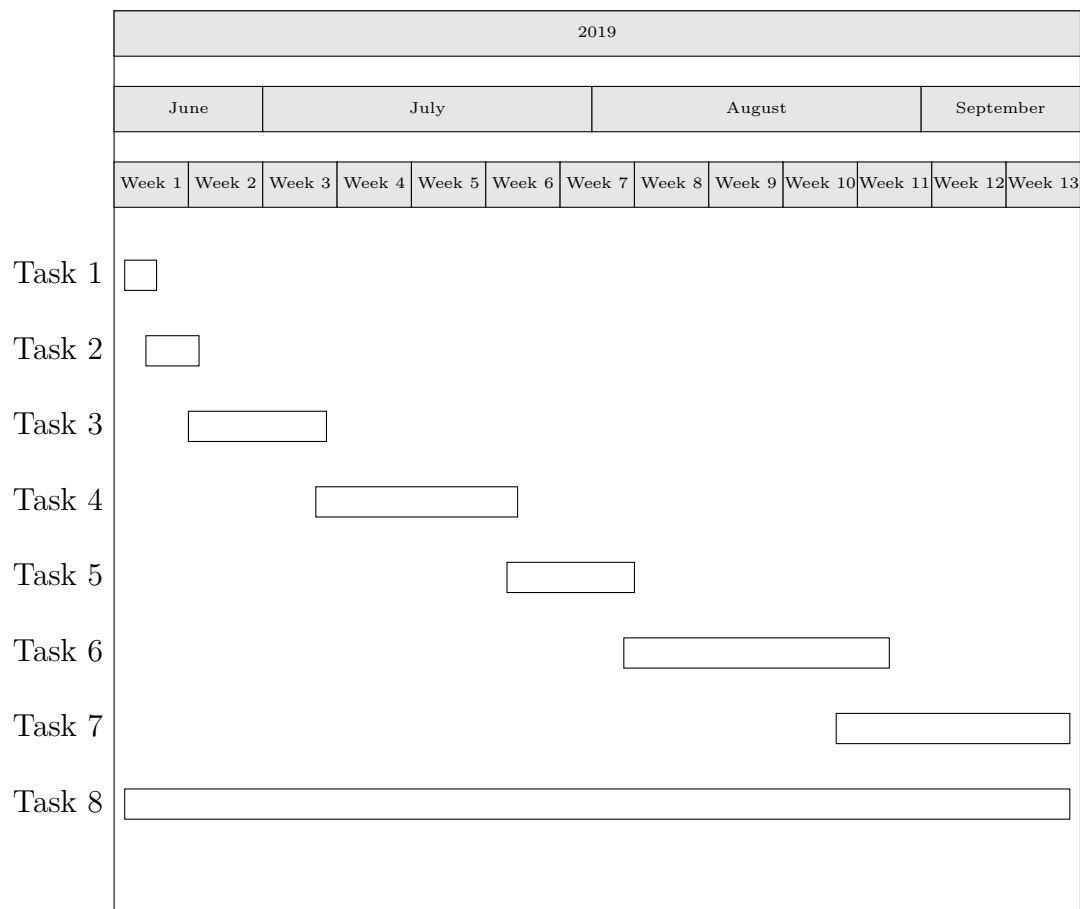
Open source projects and their intended form of usage in this project fallows:

- **Scikit-Learn [22]:** For the well versed library of machine learning algorithms and tolls such as train / test split for dataset.
- **Pandas [16]:** For general data manipulation.
- **Tensorflow [1] or PyTorch [21]:** For the implementation of the neural network.
- **Matplotlib [10] and Seaborn [31]:** Visualizing data and calculations.
- **Keras [3]:** Keras will be used to implement benchmark architectures such as LeNet-5 or AlexNet for the ease of prototyping.

7 Project Plan

Following task are based on the project objectives that explained in detail in methodology section. Project will begin on June 14th and continue until 16th of September which is deadline for report submission.

- **Task 1:** Carrying out general data exploration. (1 - 2 days)
- **Task 2:** Data pre-processing and augmentation. (2 - 4 days)
- **Task 3:** Building baseline model with well known neural network architectures. (10 - 12 days)
- **Task 4:** Using pre-trained network to increase model performance. (14 - 18 days)
- **Task 5:** Visualizing neural network to ensure learning quality. (9 - 11 days)
- **Task 6:** Model ensembling. (20 - 24 days)
- **Task 7:** Applying different deployment options. (18 - 22 days)
- **Task 8:** Project Writing (Throughout the project)



References

- [1] Martin Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: <https://www.tensorflow.org/>.
- [2] Leo Breiman. “Bagging Predictors”. In: *Mach. Learn.* 24.2 (Aug. 1996), pp. 123–140. ISSN: 0885-6125. DOI: 10.1023/A:1018054314350. URL: <http://dx.doi.org/10.1023/A:1018054314350>.
- [3] François Chollet et al. *Keras*. <https://keras.io>. 2015.
- [4] J. Deng et al. “ImageNet: A Large-Scale Hierarchical Image Database”. In: *CVPR09*. 2009.
- [5] Yoav Freund and Robert E. Schapire. *A Decision-Theoretic Generalization of on-Line Learning and an Application to Boosting*. 1996.
- [6] Yoav Freund and Robert E. Schapire. “Experiments with a New Boosting Algorithm”. In: *Proceedings of the Thirteenth International Conference on International Conference on Machine Learning*. ICML’96. Bari, Italy: Morgan Kaufmann Publishers Inc., 1996, pp. 148–156. ISBN: 1-55860-419-7. URL: <http://dl.acm.org/citation.cfm?id=3091696.3091715>.
- [7] Kunihiro Fukushima. “Neocognitron: A Self-Organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position”. In: *Biological Cybernetics* 36 (1980), pp. 193–202.
- [8] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. <http://www.deeplearningbook.org>. MIT Press, 2016.
- [9] David H. Hubel and Torsten N. Wiesel. “Receptive Fields of Single Neurons in the Cat’s Striate Cortex”. In: *Journal of Physiology* 148 (1959), pp. 574–591.
- [10] J. D. Hunter. “Matplotlib: A 2D Graphics Environment”. In: *Computing in Science Engineering* 9.3 (May 2007), pp. 90–95. ISSN: 1521-9615. DOI: 10.1109/MCSE.2007.55.
- [11] Daniel Kermany and Kang Zhang. “Labeled Optical Coherence Tomography (OCT) and Chest X-Ray Images for Classification”. 2018. DOI: <http://dx.doi.org/10.17632/rscbjbr9sj.2>.
- [12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Advances in Neural Information Processing Systems 25*. Ed. by F. Pereira et al. Curran Associates, Inc., 2012, pp. 1097–1105. URL: <http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf>.
- [13] E. N. K LARKSON. “The visual system of trilobites”. In: *Palaeontology* 22 (1979), pp. 1–22.

- [14] Y. LeCun et al. “Backpropagation Applied to Handwritten Zip Code Recognition”. In: *Neural Comput.* 1.4 (Dec. 1989), pp. 541–551. ISSN: 0899-7667. DOI: 10.1162/neco.1989.1.4.541. URL: <http://dx.doi.org/10.1162/neco.1989.1.4.541>.
- [15] Yann Lecun et al. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE*. 1998, pp. 2278–2324.
- [16] Wes McKinney. “Data Structures for Statistical Computing in Python”. In: *Proceedings of the 9th Python in Science Conference*. Ed. by Stéfan van der Walt and Jarrod Millman. 2010, pp. 51–56.
- [17] Alan Mosca and George D Magoulas. “Deep Incremental Boosting”. In: *arXiv e-prints* (Aug. 2017). arXiv: 1708.03704 [stat.ML]. URL: <https://arxiv.org/abs/1708.03704>.
- [18] Girish B. Nair and Michael S. Niederman. “Community-Acquired Pneumonia: An Unfinished Battle”. In: *Medical Clinics of North America* 95.6 (2011). Pulmonary Diseases, pp. 1143–1161. ISSN: 0025-7125. DOI: <https://doi.org/10.1016/j.mcna.2011.08.007>. URL: <http://www.sciencedirect.com/science/article/pii/S0025712511000927>.
- [19] Luke Oakden-Rayner. *Exploring the ChestXray14 Dataset Problems*. Accessed: 2019-03-25. URL: <https://lukeoakdenrayner.wordpress.com/2017/12/18/the-chestxray14-dataset-problems/>.
- [20] *Open Access Biomedical Image Search Engine*. <https://openi.nlm.nih.gov/>. Accessed: 2019-03-12.
- [21] Adam Paszke et al. “Automatic differentiation in PyTorch”. In: (2017).
- [22] F. Pedregosa et al. “Scikit-learn: Machine Learning in Python”. In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.
- [23] *Pneumonia*. URL: <https://www.nhs.uk/conditions/pneumonia>.
- [24] Pranav Rajpurkar et al. “CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning”. In: *CoRR* abs/1711.05225 (2017).
- [25] Holger Schwenk and Yoshua Bengio. “Boosting Neural Networks”. In: *Neural Comput.* 12.8 (Aug. 2000), pp. 1869–1887. ISSN: 0899-7667. DOI: 10.1162/089976600300015178. URL: <http://dx.doi.org/10.1162/089976600300015178>.
- [26] Ramprasaath R. Selvaraju et al. “Grad-CAM: Why did you say that? Visual Explanations from Deep Networks via Gradient-based Localization”. In: *CoRR* abs/1610.02391 (2016). arXiv: 1610.02391. URL: <http://arxiv.org/abs/1610.02391>.
- [27] Karen Simonyan and Andrew Zisserman. “Very Deep Convolutional Networks for Large-Scale Image Recognition”. In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. 2015. URL: <http://arxiv.org/abs/1409.1556>.

- [28] Daniel Smilkov et al. “TensorFlow.js: Machine Learning for the Web and Beyond”. In: *CoRR* abs/1901.05350 (2019). arXiv: 1901.05350. URL: <http://arxiv.org/abs/1901.05350>.
- [29] Stéfan van der Walt et al. “scikit-image: image processing in Python”. In: *PeerJ* 2 (June 2014), e453. ISSN: 2167-8359. DOI: 10.7717/peerj.453. URL: <https://doi.org/10.7717/peerj.453>.
- [30] Xiaosong Wang et al. “ChestX-ray8: Hospital-scale Chest X-ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases”. In: *CoRR* abs/1705.02315 (2017). arXiv: 1705.02315. URL: <http://arxiv.org/abs/1705.02315>.
- [31] Michael Waskom et al. *mwaskom/seaborn: v0.8.1 (September 2017)*. Sept. 2017. DOI: 10.5281/zenodo.883859. URL: <https://doi.org/10.5281/zenodo.883859>.
- [32] Bolei Zhou et al. “Learning Deep Features for Discriminative Localization”. In: *CoRR* abs/1512.04150 (2015). arXiv: 1512.04150. URL: <http://arxiv.org/abs/1512.04150>.