雲南農業大學

大数据综合应用实验报告

题目：一种通过胸部 x 光图像判断病人是否存在“肺炎”的机器学习方法

小组成员：杨一帆 2019311198

余霖 2019311195

飞汧锐 2019311192

学院：大数据学院

专业：数据科学与大数据技术

目录

一种通过胸部 x 光图像判断病人是否存在“肺炎”的机器学习方法

杨一帆，余霖，飞汗锐

云南农业大学大数据学院，云南 昆明 650000

摘要

针对医学方面肺炎诊断程序有效性较低和耗时耗力的问题，并结合当前全球疫情形势严峻的情况，本文提出一种通过机器学习技术分析胸部 X 光照片初步判断病人是否存在“肺炎”的模型。考虑实际应用需求，综合网络性能和运行速度，选择卷积神经网络(CNN)作为骨干网络，并添加局部强化模块来提高检验精确度。添加池层、“扁平化”层、隐藏层等方法提高模型性能，并使用 softmax 激活函数连接重要层，强化模型的计算速度和提高计算准确率。通过一系列的准确性测试，准确度达到了 97%，检测结果实时性也较强，能够在保持实时性的同时取得更优的检验精确度，同时部署到 Web 网页，可以实时在线检测，实用价值大大提升。

关键词：X 光图像；深度学习；卷积神经网络（CNN）；肺炎检测；

引言

深度学习的使用在当前的生物医学科学界越来越流行，在最近许多类型的医疗数据的可用性激增。成为最受欢迎的前沿技术之一。肺炎给人类的健康生活带来巨大风险和挑战，特别是近年来全球新冠疫情越来越严重，当今世界有数十亿人生活在发展中国家和一些落后地区，面临医疗匮乏，环境污染严重，人口众多等等问题，世界卫生组织估计，每年有超过 400 万的人死于空气污染所导致的疾病，包括肺炎。每年有超过 1.5 亿人感染肺炎，尤其是 5 岁以下的儿童。据世界卫生组织 2022 年 6 月 23 日公布的最新数据显示，目前全球累计新冠确诊病例达 539119771 例，死亡和确诊人数还在不断增加，尤其在医疗技术不是很发达的国家，由于缺乏医疗资源和医护人员，这个问题可能会进一步恶化。这证明了这种疾病的严重性和准确检测的必要性。对于这些人群，准确而快速的诊断至关重要。它可以保证医疗机构可以快速检测，病人能及时获得治疗，政府能有效控制疫情。在当前新冠病毒肆虐全球的情况下，医疗基础设施的压力不断大，能不能快速准确的检测出肺炎已经成为一个亟待解决的问题。目前，精准医疗的提出，人工智能的发展为疾病的快速诊断奠定了基础。建立一种利用机器学习技术通过胸部 X 射线图像快速准确识别患者是否患有肺炎的工具是非常有必要的。

诊断肺炎最常用的方法是通过胸部 X 光片或胸部 X 光检查，将感染描述为肺部特定区域的不透明度增加。目前，胸部 X 光检查是诊断肺炎的最佳方法，它在临床护理和流行病学研究中发挥着至关重要的作用。然而，通过 X 光片来检测肺炎是一项具有挑战性的任务，需要依赖放射科医师的专业能力。然而，随着深度学习的兴起，卷积神经网络能够准确的应用于图像的识别与处理，为医疗诊断提供了新的技术和方法支持，为了提高诊断程序的有效性和范围，我们可以利用机器学习算法来识别胸部 X 射线图像中的异常。

随着深度学习的发展，研究学者倾向采用卷积神经网络实现“肺炎”病症检测，通过训练深度学习模型（多层 CNN 模型），使 CNN 能够根据患者胸部的 X 射线图像检测患者是否患有肺炎疾病。通过 flask 构建 Web 网站系统，使其加载训练好的神经网络模型能够使其实时进行图像检测，以应用深度学习方法（CNN）进行肺炎的检测。我们使用了 ChestX-ray14 数据集训练的 5 层的深度卷积网络，该网络通过胸部 X 光片识别肺炎的准确率已经和人类放射科医生持平甚至更高。网络输入为人体正面扫描的胸片，输出患肺炎的概率。该模型层结构复杂，数据量大。CNN 通过卷积处理，优势在于共享卷积核，处理高维数据无压力；可以自动进行特征提取。卷积层可以提取特征， 卷积层中的卷积核（滤波器）真正发挥作用，通过卷积提取需要的特征。

结合 X 光诊断“肺炎”的特点和需求，本文提出一种分析胸部 X 光照片初步判断病人是否存在“肺炎”的模型。模型整体使用卷积层神经网络 CNN 为基准网络，并添加局部强化模块来提高检验精确度：在每个块添加池层简化机器计算难度，加快计算速度；在每个全连接层中，使用“密集”方法添加一个隐藏层，其中“单位”表示该层中节点/神经元的数量；使用“softmax”激活函数连接最后两个互斥的完全连接层。另外，采集多种类 X 光肺炎图像病人工标注，建立 X 光肺炎诊断专用数据集，实验结果表明，本算法对 X 光肺炎识别有良好的诊断能力， 且保持较理想的处理效率。

网络模型设计

CNN 模型网络构造和初始化模型

本文研究所用到的模型如图一所示。该模型由五个“卷积”块组成，在每个块之后，添加一个池化层（“最大池化”最大拼接用于减少输出体积的空间尺寸）。 最后一个卷积块之后的“扁平化”层准备将输入馈送到全连接层。 在每个全连接层中。dense 方法用于添加一个隐藏层，其中 units 表示该层中节点/神经元的数量。 最后一个全连接层有两个节点代表两个类——“肺炎”和“正常”，其中使用了“softmax”激活函数（因为这两种情况是互斥的）。在这种情况下，也可以使用“sigmoid”激活函数，在输出层中有一个单元，具有“二元交叉熵”损失。

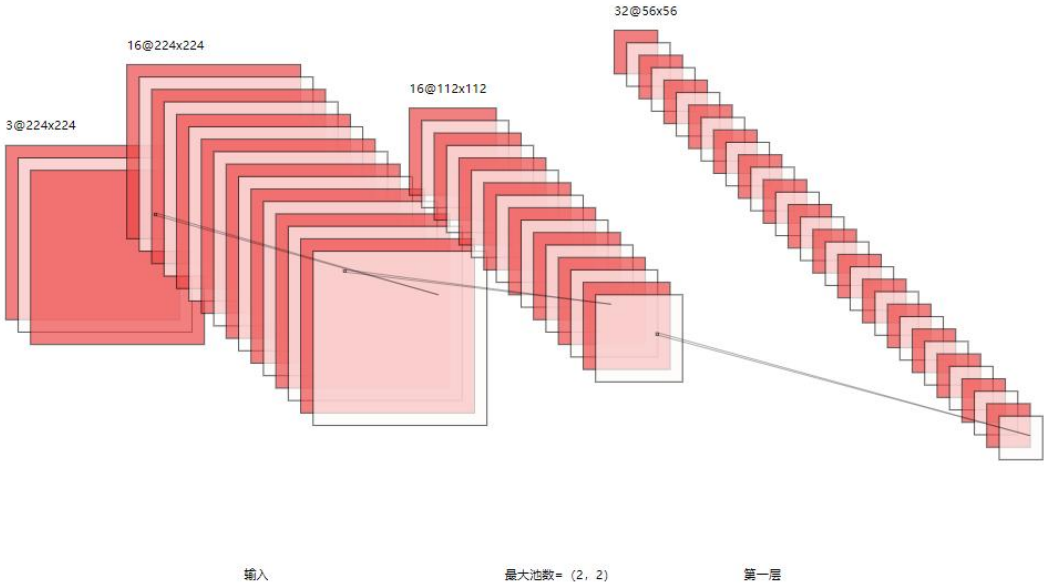

图 1 模型整体网络结构图

CNN 网络通常就是有卷积层、池化层和全连接层组成。卷积层就是用一个滤波器（filter）去过滤原来的图片，把重要的信息留下来把没用的信息去掉。比如识别猫狗，其实只有图片中间那一点点是有用的信息，边上背景都是增加计算量的无用信息。池化通常分 max pooling 和 mean pooling，它的作用就是在尽量保留数据真实度的情况下减少计算量。全连接层就是和普通 BP 神经网络类似。

设计好的滤波器（filters）和池化方法（pooling）组合可以最有效的滤出图片中的有用信息，同时减少计算量。

部分关键代码如下:

网络构造,初始化模型

cnn_model = Sequential ()

添加一个池化层

cnn_model.add (Conv2D(16, (3, 3), padding='same',input_shape=(224,224,3), activation='relu'))

添加一个最大池化层

cnn_model.add (MaxPooling2D((2, 2)))

添加 BN 层

cnn_model.add (BatchNormalization())

Dropout 正则化处理

cnn_model.add (Dropout(rate=0.2))

扁平化处理，转换为 12544 维

cnn_model.add (Flatten ())

采用 BatchNormalization（简称 BN）算法：

训练深度网络的时候经常发生训练困难的问题，因为，每一次参数迭代更新后，上一层网络的输出数据经过这一层网络计算后，数据的分布会发生变化，为下一层网络的学习带来困难（神经网络本来就是要学习数据的分布，要是分布一直在变，学习就很难了），此现象称之为 Internal Covariate Shift.

采用 BN 算法可以加快训练速率，即可以增大学习率，加快模型的收敛速度,不过分依赖网络初始值,一定程度上抑制了过拟合情况，降低了 Dropout 的必要性

BN 层指的是对输入神经网络的一批次 feature map 数据的每一个通道进行归一化操作，使得输入每一维度满足均值为 0 方差为 1 的数据分布。这里需要注意的是，如果每一层都满足均值为 0 方差为 1 的标准正态分布，那么网络很难学习到新的信息，所以在最后需要添加缩放以及平移信息，以达到每一次数据经过归一化后还保留的有学习来的特征。

所以在每一卷积块中添加 BN 算法可以大大加快模型训练速度。

采用 dropout 正则化方法，增强模型泛化能力

dropout 是一种针对神经网络模型的正则化方法。是在训练过程中，随机的忽略部分神经元。它强迫一个神经元单元和随机挑选出来的其他神经单元共同工作，达到较好的效果，减弱了神经节点间的联合适应性，增强了泛化能力。经过验证，隐含节点 dropout 率等于 0.5 的时候效果最好。此时 dropout 随机生成的网络结构最多。本模型在第四个卷积块中采用该方法，在 keras 的每个权重更新周期中，按照给定概率（如 20%），随机选择要丢弃的节点，以实现 dropout。用于加快模型训练，提升泛化能力。

dropout 也可以用在输入层，作为一种添加噪音的方法。dropout 只能在模型的训练过程中使用，在评估模型时不能使用。

实验结果及分析

数据准备与预处理

数据准备:

ChestX-ray14 包含 112000 张图像，其中包括 30000 名患者。有些病人需要多次扫描，这将被考虑在内。所有图像最初都是 1024*1024 像素。由于数据来源和损坏问题，图像数据集包含原始 112000 张图像中的 5000 多张。所有数据用于结构化模型。此外，还为每个图像提供结构化数据。该数据集包括年龄、随访次数、AP 与 PA 扫描以及患者性别等特征。进一步检查数据，标签是分层的。例如，一些标签只是“肺气肿”，而其他标签是“肺气肿、心脏问题”。 图像数据总体分为 Normal 和 Pneumonia 两类，对应正常和患病，数据图例如下所示:

有上图可观察到: 正常胸部 X 光片肺部轮廓清晰，而患肺炎的则模糊，医学上就以此来区分是否患病。

图 2 Pneumonia 胸部 X 光图

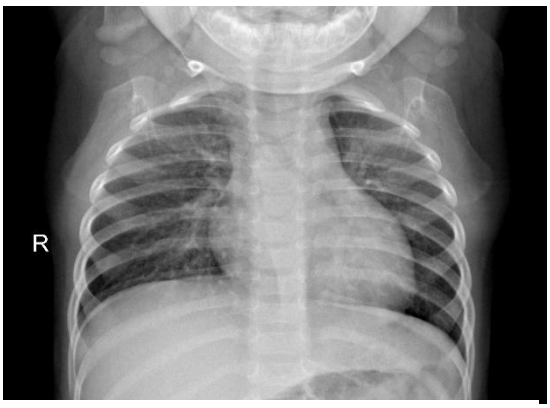

图 3 Normal 胸部 X 光图

将 5347 图像数据集按比例分为训练集、测试集和验证集。数据分类结构如下图所示:

数据预处理

采用 ImageDataGenerator 方法图片增强方法，对 5347 张图片数据进行 0.3 比例

Dataset Type	Normal	Pneumonia
Training	1341	4006
Test	366	390
Validation	47	48

图 4 数据分类结构图

的缩放，对图片进行 10 度范围的图片旋转以增强模型的泛化能力。

采用 flow_from_directory 方法将图片大小转换为 224*224。通过对图像数据进行一系列处理，减小了模型训练、数据处理时对 CPU 的压力，可以加快神经网络学习速度，提升了模型的数据处理性能。

模型训练

防止模型过拟合

采用 EarlyStopping 该函数的目的是防止模型出现过拟合，因为在我们训练模型的过程中，很有可能出现过拟合的情况，对于模型训练学习过程中会出现识别率过高的情况。

这个时候训练集表现很好，但是验证集表现就会下降。这时候我们需要提前结束训练，得到“最佳”（只能判断是在全局范围内最佳）的结果。

模型训练

采用 cnn_model.fit 方法训练 CNN 模型，将训练集，测试集，验证集输入到模型中。设置迭代次数为 100，当模型训练过程中可以看到随着一个 epoch 的进行，准确率不断升高，损失函数一直减少，说明模型训练效果越来越好。

最终模型训练到了第 26 轮停止，说明此时模型已经达到了很好的分类效果，可以对模型进行下一步的验证测试了。

模型评估及实验结果分析

```
Output exceeds the size limit. Open the full output data in a text editor
Epoch 1/100
168/168 [==============================] - 524s 3s/step - loss: 0.3701 -
Epoch 2/100
168/168 [==============================] - 480s 3s/step - loss: 0.2863 -
Epoch 3/100
168/168 [==============================] - 547s 3s/step - loss: 0.2571 -
Epoch 4/100
168/168 [==============================] - 587s 3s/step - loss: 0.2508 -
Epoch 5/100
168/168 [==============================] - 505s 3s/step - loss: 0.2460 -
Epoch 6/100
168/168 [==============================] - 1093s 7s/step - loss: 0.2177 -
Epoch 7/100
168/168 [==============================] - 505s 3s/step - loss: 0.2063 -
Epoch 8/100
168/168 [==============================] - 415s 2s/step - loss: 0.1996 -
Epoch 9/100
```

图 5 模型训练图

对训练模型进行一系列的效果评估:

模型验证集准确度如下:　　模型损失量如下:

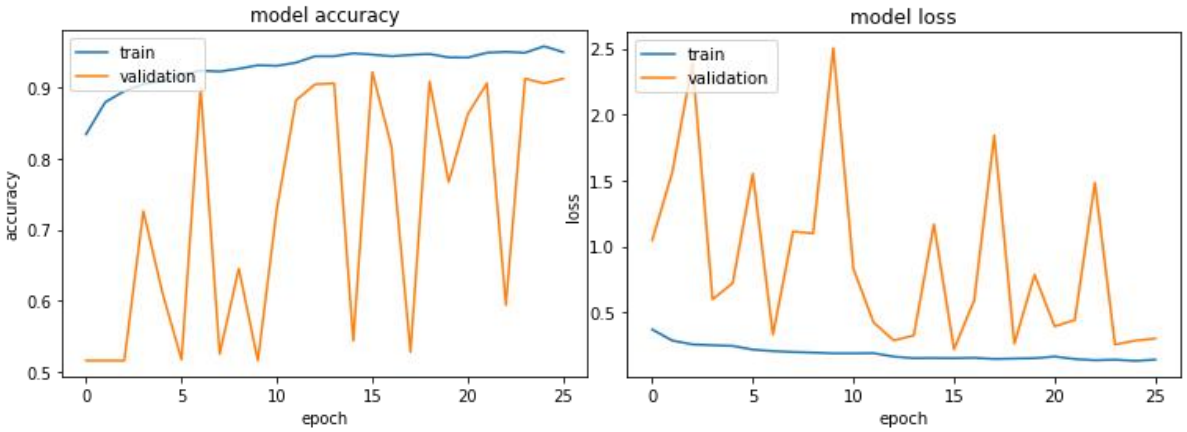

图 7 模型验证集准确度图　　**图 6 模型损失量图**

由上图可知：随着训练轮数增加，训练集准确率始终保持在 80%以上，模型损失率也保持在较低水平，并且在第 24 轮左右准确率达到最高 97%，验证集准确度不断发生变化，说明模型训练过程中采用 dropout 正则化方法，通过随机关闭一些神经网络节点来加快模型训练，提升泛化能力。

测试集混淆矩阵:

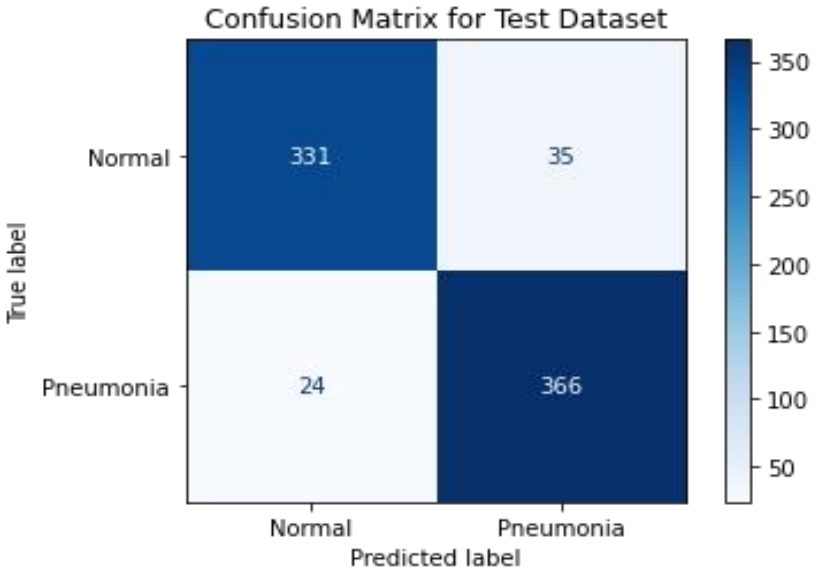

图 8 测试集混淆矩阵图

由图可以看出模型对测试集数据的分类的正确率较高

ROC AUC 曲线:

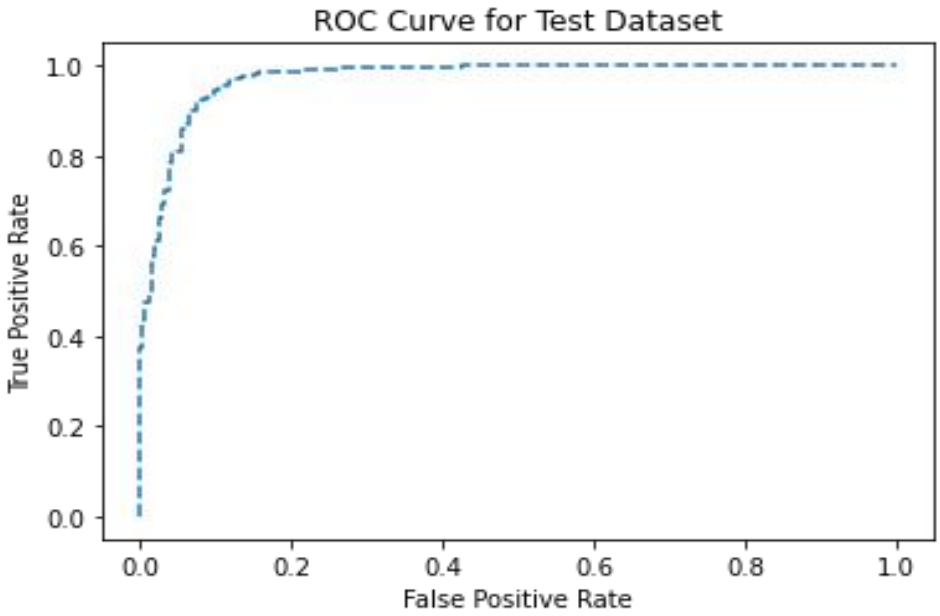

图 9 ROC AUC 曲线图

ROC 曲线的纵坐标为真阳率 true positive rate (TPR) (也就是 recall)，横坐标为假阳率 false positive rate (FPR)，这两个指标的分母都是相对真实 label 说的，TPR 即真实正例中对的比例，FPR 即真实负例中的错的比例。计算 ROC 曲线下方的面积大小结果为：ROC AUC (Test Dataset) 0.97。AUC 值在 [0.85, 0.95]之间表明模型分类效果很好，大于 0.95 说明模型的分类效果非常好。

应用

为了采用机器学习对 X 光片判断是否患肺炎这一技术能有更大范围的应用，我们将创建一个在线 Web 检测网站，一方面，我们将为病患用户提供自主查询服务，用户只需将自己的 X 光片上传，就能在几秒钟内得到检测结果，能给用户提供一定的医学参考，让用户更了解自己的病情。另一方面我们将对医疗机构提供快速检测服务，特别是在当前新冠病毒肆虐时期，医院对新冠肺炎病例的检测需求大大增加，每天都有数以万计的 X 光片需要医生进行判断，如果采用机器学习分类技术进行批量大规模处理，将大大节省人力物力，更好的控制疫情，保护人民的生命安全。

Web 检测网站页面如下图:

只需点击一张需要分类检测的图片，然后提交，便会有相应的分类结果和概率。

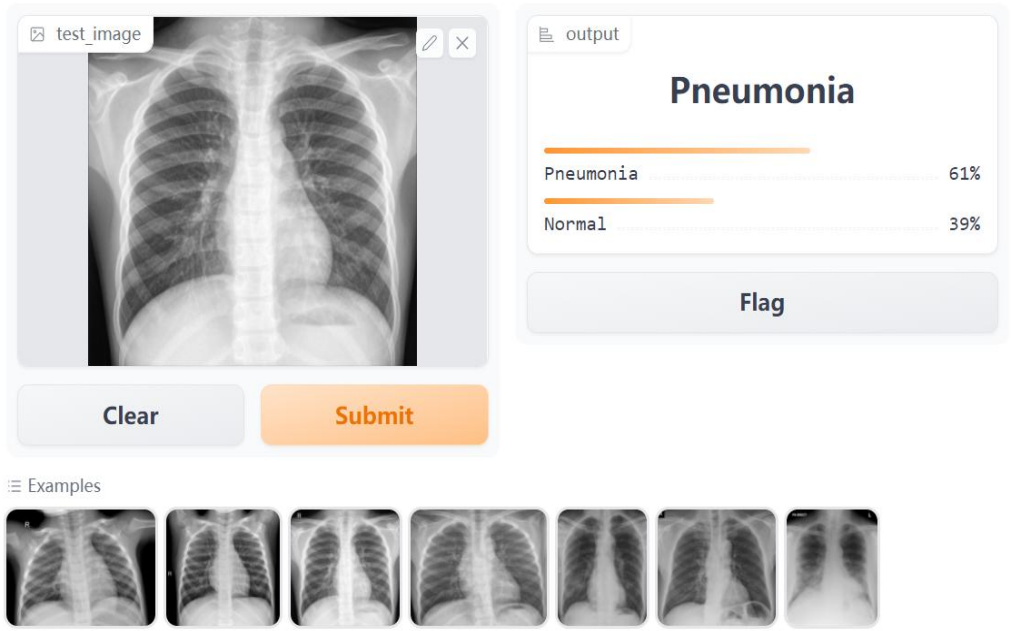

图 10 Web 检测网站页面图

结论

针对全球新冠疫情严重性居高不下的问题，通过这个项目，我们构建了一个强大，准确，快速的神经网络模型，能以 97%的准确率来识别肺炎病毒疾病病例，通过对 X 光片的机器识别，通过图片中的一些特征就能判断患者是否患病，从而给医生的下一步诊断提供参考，将该模型部署在 Web 网站中，能够大规模普及该技术，特别是一些医疗资源紧张、疫情严重的地区，能够大规模、快速的筛查新冠患者，能够缓解当下紧张的疫情局势，节省大量人力物力，保护和挽救更多人的生命，具有巨大的应用价值

贡献

Baishali Dutta 负责这里展示的整个数据和代码。 可以找到他的代码在 https://github.com/baishalidutta/Pneumonia-Detection 。 这个项目是共享的，所有项目报告都可以在那里找到。

参考笔记

[1]Yao, Li and Poblenz, Eric and Dagunts, Dmitry and Covington, Ben and Bernard, Devon and Lyman, Kevin. (2017). Learning to diagnose from scratch by exploiting dependencies among labels.

[2]Wang, Xiaosong and Peng, Yifan and Lu, Le and Lu, Zhiyong and Bagheri, Mohammadhadi and Summers, Ronald. (2017). ChestX-ray8: Hospital-scale Chest X-ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases. arXiv:1705.02315.

[3]Rajpurkar, Pranav and Irvin, Jeremy and Zhu, Kaylie and Yang, Brandon and Mehta, Hershel and Duan, Tony and Ding, Daisy and Bagul, Aarti and Langlotz, Curtis and Shpanskaya, Katie and Lungren, Matthew and Ng, Andrew. (2017). CheX-Net: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning.

[4]Oakden-Rayner, Luke. “Exploring the ChestXray14 Dataset: Problems.” Luke Oakden-Rayner, 18 Dec. 2017, lukeoakdenrayner.wordpress.com/2017/12/18/the-chestxray14-dataset-problems/

[5]Oakden-Rayner, Luke. “CheXNet: an in-Depth Review.” Luke Oakden-Rayner, 24 Jan. 2018, lukeoakdenrayner.wordpress.com/2018/01/24/chexnet-an-in-depth-review/.

[6]Chattopadhyay, Aditya and Sarkar, Anirban and Howlader, Prantik and Balasubramanian, Vineeth. (2017). Grad-CAM++: Generalized Gradient-based Visual Explanations for Deep Convolutional Networks.

[7]Tang, T.H. “Weakly Supervised Learning for Findings Detection in Medical Images.” GitHub, 7 Aug. 2019, github.com/thtang/CheXNet-with-localization.
[8]Lee, WonKwang. A Simple Pytorch implementation of Grad-CAM, and Grad-CAM++.

GitHub, 3 Aug. 2018, https://github.com/1Konny/gradcamplusplus - pytorch