Locating signs of Pneumonia in Chest X-rays

Josh Reid
Electrical and Computer Engineering
University of Waterloo
Waterloo, Canada
js2reid@uwaterloo.ca

Sruthy Paul
Electrical and Computer Engineering
University of Waterloo
Waterloo, Canada
s25paul@uwaterloo.ca

***Abstract*—In recent years, pneumonia detection has become a necessity as it it one of the deadliest diseases in Canada. In this work, a dataset of chest X-rays is used to attempt to replicate bounding boxes drawn by radiologists of areas that are suspected of indicating pneumonia as closely as possible. This is done by using a pre-trained deep neural network (DenseNet121) to extract important features from the chest X-ray images and then different binary classifiers are tested on these extracted feature vectors to determine their effectiveness at determining if they indicate pneumonia or not.**

***Index Terms*—biomedical imaging, DICOM, X-ray applications, Medical diagnostic imaging**

I. Introduction

In recent years, automatic pneumonia detection has become a necessity due to its entry as the eighth deadliest disease in Canada [7]. Pneumonia is characterized by an accumulation of fluid in a patients lungs, causing difficulty breathing and when severe enough can cause oxygen deprivation and death. There are various approaches used to diagnose pneumonia, starting with auditory checks by a doctor using a stethoscope and then advancing to using ultrasound scans or X-ray images of the chest to detect the accumulation of fluid in the lungs that is indicative of pneumonia. The work presented in this paper focuses on locating signs of pneumonia from chest X-rays, as these are more commonly perform due to the higher prevalence of X-ray machines globally and the higher resolution versus ultrasound images.

Doctors use X-ray images to detect pneumonia by looking for airspace opacity, lobar consolidation, or interstitial opacities with considerable overlap in the lungs. It is a challenging task to diagnose pneumonia on X-rays, because of a number of other conditions in the lungs such as fluid overload (pulmonary edema), bleeding, volume loss (atelectasis or collapse), lung cancer, or post-radiation or surgical changes. Outside of the lungs, fluid in the pleural space (pleural effusion) also appears as increased opacity on chest radiographs (CXR). A number of factors such as positioning of the patient and depth of inspiration can alter the appearance of the CXR [3]. An efficient approach can be locating the areas of pneumonia with bounding boxes from CXR and use them to automatically detect signs of pneumonia in chest X-rays.

Chest X-rays are black and white images of organs in your chest. These images can reveal the condition of the lungs i.e. help the radiologist detect cancer, heart related lung problems, changes in size and outline of heart, rib or spine fractures, etc. The prevalence and low cost of X-ray machines makes it more desirable for physicians to use for diagnosis. The main downsides with using X-rays is that X-rays are ionizing radiation, limiting the exposure that patients can handle before causing damage to other cells, and since X-rays go through all tissue this makes it difficult to isolate the lungs from the rest of the chest [2].

Pneumonia is most prevalent and dangerous in young children, people older than 65, and people with underlying medical conditions. According to the Centers for Disease Control [1], 544,000 cases of pneumonia occurred in the United States in 2015, and 51,811 of these resulted in death. Early detection of this disease allows for the correct antibiotics to be administered earlier, increasing the likelihood for survival.

The rest of the paper is organized as follows: In Section II, the background work covering existing techniques are discussed. Section III introduces the dataset used in this study. Section IV outlines the overall system design and methodology for evaluating the dataset. The results of the experiments and their analysis are discussed in Section V. Section VI concludes the paper by providing directions for future work.

II. Literature Survey

A. Pneumonia Detection Modalities

There are several works reported for pneumonia detection from ultrasound images. Automatic pneumonia detection from ultrasound images of lungs have been performed using standard neural networks [12]. Approaches for pneumonia detection from Computed Tomography (CT) images are have also been reported [13].

There are prior works reported for pediatric based pneumonia detection also. Stephanakis et al. [14] have proposed a method, especially for pneumonia detection in children, using multiresolution autoregressive filters. Spectral-based pneumonia detection from ultra sound images for children have also been reported Omar et al. [15]. Yusuf et.al. has reported cough sound analysis for pneumonia in pediatric population [16]. Pneumonia clouds detected from Otsu thresholding in chest X-rays using image processing approaches have also been reported by Abhishek et al. [18]. Insu Song [17] has reported pneumonia detection from sounds collected using low cost cell phones. The state of art performance for pneumonia detection from X-rays is reported by Pranav et.al in [19], where the algorithm, CheXNet, exceeds the average radiologist performance on the pneumonia detection task.

B. DenseNet as a Feature Extractor

Densely Connected Convolutional Networks (DenseNet), are a convolutional neural network(CNN) architecture that are able to have substantially more convolutional layers without suffering from vanishing gradient by connecting each layer to every other layer in a feed-forward fashion. Each layer of DenseNet, in which feature-maps of all preceding layers are used as inputs, into all subsequent layers [20].Since each layer has direct access to the gradients from loss function and the original input signal, it is easier to train DensetNet and information flow is better compared to other networks. As compared to traditional convolutional networks and ResNets [21], this architecture explicitly differentiates between information that is added to the network and the information that is preserved. As it is evolved from CNN, DenseNet too acts as a feature extractor and a classifier.

The availability of open-source pre-trained models [22] [23] has led to the use of off-the-shelf CNN features as complementary information channels to existing hand-crafted image features. This technique is popularly known as transfer learning. Transfer learning is defined as the ability of a system to recognize and apply knowledge and skills learned in previous tasks to novel tasks [24].

Prior studies have been successful in applying transfer learning for tasks such as object recognition [25]. Several studies have reported success in using pre-trained CNN features for medical image analysis. Kieffer et al. have evaluated the performance of pre-trained deep features and the impact of transfer learning with a small dataset of histopathology images [26]. Other studies have achieved similar results when attempting to utilize pre-trained.

III. Methods and Materials

A. Imaging data sets

For this study, an online dataset of chest X-rays labeled by doctors from the Radiological Society of North America (RSNA) and the US National Institutes of Health The Society of Thoracic Radiology with bounding boxes of areas indicative of pneumonia. This data was hosted and provided by Kaggle as part of one of their dataset competitions.

The bounding boxes are only drawn on images of patients who were found to test positive for pneumonia, patients without pneumonia had no bounding boxes. The dataset was also cleaned by RSNA, removing images that were not positioned in the frame well. were not in the coronal plane, and ones that were too over or underexposed.

The bounding boxes were provided as a CSV file which has one row for each box drawn and ID column which connects that bounding box with its corresponding chest X-ray image. Images that have multiple bounding boxes have multiple rows in this CSV file. In order to allow for these bounding boxes to be compared to the output of a deep neural network they were converted to images called masks. These masks are the same size as the provided chest X-ray image and have binary values for each pixel, 1 for that pixel indicating pneumonia and 0 if it does not, creating a black and white image. Figure 1 shows a sample of 8 images from the dataset with the mask overlaid on top of the corresponding chest X-ray image.

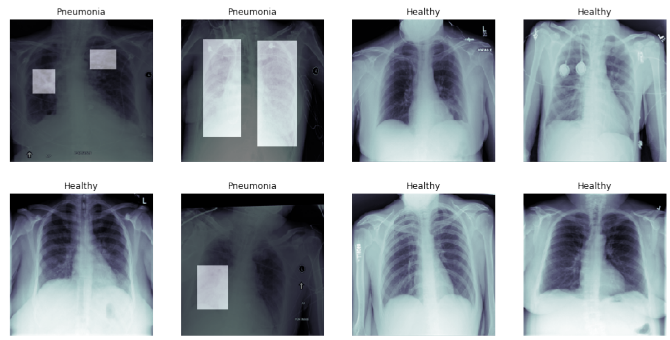

Fig. 1. A sample of the images from the dataset with the provided boxes overlaid on top in white

B. Software algorithm

This work will compare the performance of several different binary classification algorithms on their performance to classify the output of a pre-trained deep neural network. The source chest X-ray images are run through the pre-trained network DenseNet121 within Keras which has shown good results at classifying the ImageNet dataset [8], which means it is a good feature extractor for images. Although ImageNet does not consist of medical images, it has been found to still work well on these types of images because the features its extracting after several convolutions are simply small lines or edges which are present in all images [9]. This also had the additional benefit of reducing the time required to train the classifiers as the neural network did not also require training.

The output from DenseNet121 is a 28x28 pixel 'image' where each pixel is a 1024 length feature vector which are indicative of the features present in that area of the original image. Each of these 1024 feature vectors were given as input to each of the classifiers along with a target of 1 or 0 based on the mask in that location. The classifiers tested in this work were all from the Python library Scikit-learn and included Naive Bayes, Nearest Neighbors, Linear Support Vector Machine (SVM), Radial Basis Function (RBF) SVM, Decision Tree, Random Forest, Neural Net and AdaBoost. This is almost all of the classifiers available in Scikit-learn, with the only exclusions being Gaussian Process due to it having poor performance on data with a high number of features like this and Quadratic Discriminant Analysis due to it trying to fit a quadratic curve to the input data which also gave poor performance on this data with 1024 features.

C. Validation strategy

The accuracy of the results from the each classifier were validated using a batch of 80 images, each containing 784 vectors to be classified for a total of 12544 feature vectors.

This dataset is imbalanced towards negative cases due to only 31.6% of the provided images having pneumonia, and only 9% of the pixels in positive images indicate pneumonia on average.

The accuracy of each classifier was measured several different ways, through precision, recall and Intersection Over Union (IOU).

1) Precision: Precision is measured as the number of times the classifier correctly classified positive cases divided by the total number of cases. Although precision normally used, it is not as useful of a metric for this dataset due to it being very imbalanced. It being imbalanced against positive cases means that if the algorithm always returns negative, it will get a high precision still (¿95%).

2) Recall: Recall is measured as the number of times the classifier correctly classified positive cases divided by the total number of positive cases. This is a more useful metric for this imbalanced dataset so that classifiers that do return all negative will have a recall score of 0.

3) IOU: Accuracy for this network was determined by the percentage of overlap that the determined bounding boxes have with the actual ones drawn by radiologists. This is calculated as the Intersection over Union (IOU), which is the area of the intersections between the boxes predicted by the model plus the area of the target boxes divided by the total area of both boxes. This was calculated by determining the number of pixels that were predicted to indicate pneumonia that are also in that position in the mask and divided by the total number of pixels predicted to indicate pneumonia plus the total number of positive pixels in the mask subtract the intersection. This is the metric that is being evaluated for this competition on Kaggle, so this will be the main metric to maximize. A visualization of how to calculate this is shown in figure 2.

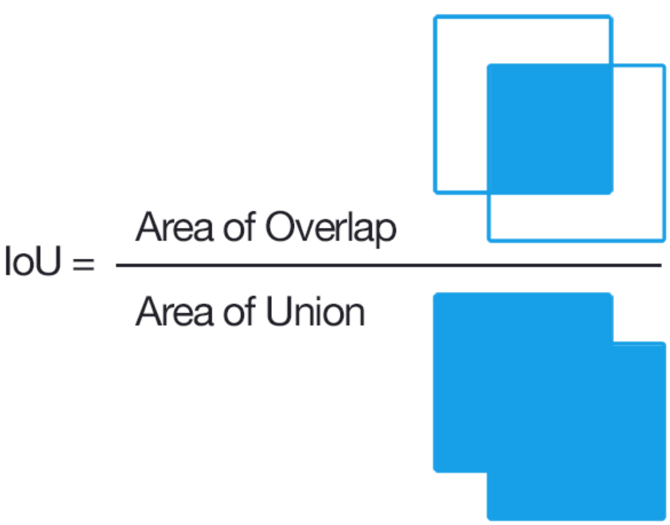

Fig. 2. A visual example of how IOU is calculated

IV. Results

A training batch of data was prepared from 80 images, each comprised of 784 feature vectors for a total of 62720 feature vectors, each with a corresponding binary label. Each of the classifiers was trained on the same training data and was initially trained with default parameters. The performance metrics defined in the Validation Strategy section of this report for each of these classifiers along with the time required to train is shown in Figure 3 and Table I

TABLE I
Classifier Performance

Classifier	Metrics (%)		
	Precision	***Recall***	***IOU***
Naive Bayes	50	89	5
Nearest Neighbors	81	22	3
Linear SVM	80	18	7
RBF SVM	78	23	17
Decision Tree	78	28	4
Random Forest	80	7	2
Neural Net	85	27	6
AdaBoost	80	9	7

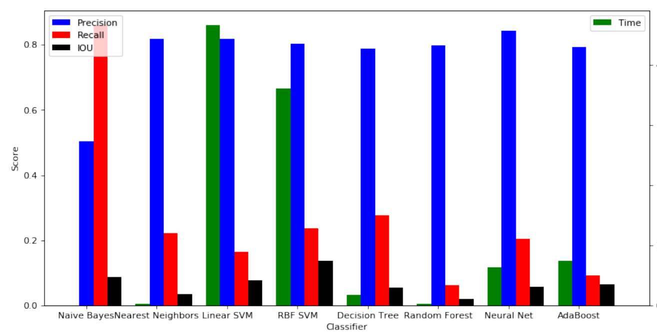

Fig. 3. Different performance metrics for each of the classifiers after training with default parameters

It was found that RBF SVM performed the best in terms of IOU, but also took the second longest time to train at just under 5 minutes. Naive Bayes also appears to perform well and took orders of magnitude less time to train at around 3 seconds, however due to it having low precision and high recall suggests that this means that it is merely classifying more of the pixels as positive at the cost of having more false positives which is not desired. Random forest performed the worse out of all of the classifiers, which due to the low recall and IOU suggests that it is due to it classifying few of the feature vectors as positive due to each of the decision trees only seeing a fraction of the training set.

A. Optimizing Hyperparameters

All of the classifiers within Scikit-learn have several different hyperparameters that can be adjusted to improve their ability to classify a given dataset. These include things like the cost of incorrectly labeled points in SVM or the number of elements to have in each of the final leafs in a decision tree. Due to the challenges of this dataset being that it has a large number of features and being imbalanced, there are a couple hyperparameters for each that can help overcome this. The hyperparameters that were selected are described in Table II.

TABLE II
CLASSIFIER HYPERPARAMETERS

Classifier	***Hyperparameter***	***Description***
Naive Bayes	var_smoothing	Portion of the largest variance of features that is added to variance
Nearest Neighbors	p	The distance calculation method used, 1 is Manhattan, 2 is Euclidean
Linear SVM	C	Cost of incorrectly labeled points
RBF SVM	C	Cost of incorrectly labeled points
Decision Tree	min_samples_leaf	The minimum number of samples required to be at a leaf node
Random Forest	min_samples_leaf	The minimum number of samples required to be at a leaf node
	n_estimators	The number of trees in the forest
Neural Net	alpha	L2 penalty, a regularization term
	hidden_layer_sizes	the number of neurons in the ith hidden layer
AdaBoost	n_estimators	The maximum number of estimators at which boosting is terminated

The function GridSearchCV, which allows for parametric sweep of these parameters for any classifier, was used to test different values for each of these parameters. Due to the computation time increasing exponentially with the addition of more parameters, the number of values tests was limited to less than 12. To quantify the quality of each classifier the recall score was used as this is a metric built-in to GridSearchCV, unlike IOU. The time to train was also recorded for each classifier, and the results are shown in Table 4

1) Naive Bayes: It was found that adjusting the smoothing parameter had a negligible impact on the recall score of the classifier throughout 6 different orders of magnitude tested. There was a minor impact on the training time, however since the training time is less than 0.2s for every classifier this is possibly due to some variance in the calculation time, not due to improvements due to this parameter. The recall scores for this classifier were some of the highest, however it was seen previously that this algorithm has low precision which suggests that this is the case due to it classifying a high number of cases as positive.

2) Nearest Neighbors: It was found that changing the method of calculating distance from Euclidean to Manhattan did improve the recall score and also reduced the time required to train the classifier. This has been shown to be true for all high dimensional datasets and is due to the Euclidean distance returning similar values for all distances when the number

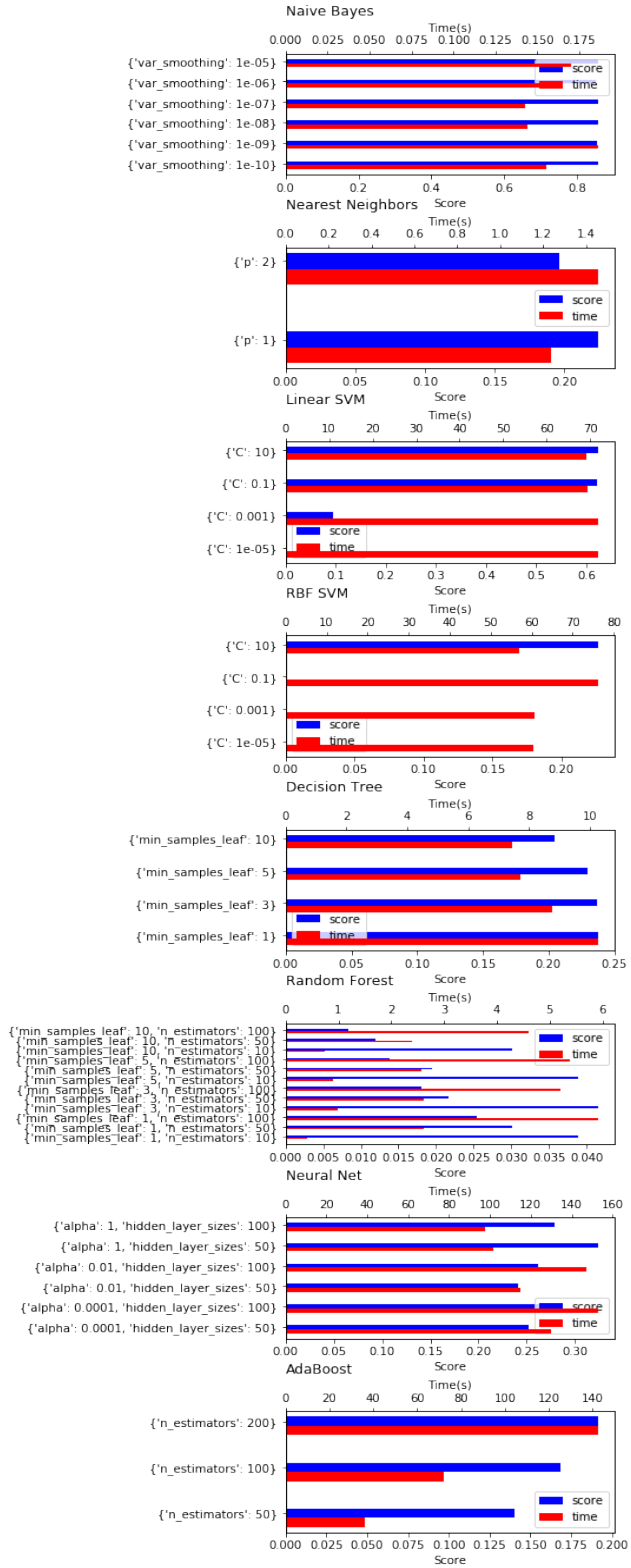

Fig. 4. Recall score and training time of different classifiers comparing the value of different hyperparameters

of dimensions is high (¿10) [10], as in this case with 1024 dimensions.

3) SVM: For both linear and RBF SVM classifiers it was found that increasing the cost parameter for incorrect answers greatly improved the performance. This is due to the vectors returned from DenseNet121 having a wide variety of values which requires close fitting of the support vectors to each training datapoint. Due to there being many negative cases, at low cost the SVM will classify almost all cases as negative and just suffer the cost of the incorrectly labelled positive cases. This results in the recall being 0 due to there being almost no positive cases. Increasing the cost causes the SVM to become sensitive enough to detect the positive cases, and recall improves. The linear SVM also performed significantly better than the RBF which is due to similar reasons as nearest neighbours where calculating the distance linearly performs better with higher dimensional data than to a higher dimension kernel.

4) Decision tree, Random Forest and Adaboost: Three different varieties of decision trees were tested on this dataset, basic decision tree, random forest and adaptive boosting (AdaBoost). The basic decision tree performed the best on this dataset, which shows that the optimizations performed by random forest and AdaBoost do not work well when there is a large number of dimensions. For random forest, this is likely due to there being multiple decision trees that each see a small portion of the dataset, suggesting that decision tree classifiers perform better when trained on a larger dataset. This means that the random forest classifier has high variance, resulting in fewer positive classifications due to low confidence.

On the other hand, AdaBoost still sees the entire dataset for each learner, it instead starts with a weak learner initially and 'boosts' the difficult to classify results from this learner. This means that Adaboost is trying to fit all outliers, and with this dataset there are too many outliers causing the final classifier to have too high of a bias towards these outliers. This occurs because the loss function within AdaBoost is exponential, which will place a much higher weight on outliers that are far from the predicted result, compared to a polynomial loss function like mean square error.

For the basic decision tree classifier the only parameter that was adjusted was the minimum samples required in a leaf node, which determines how many dimensions to use to split the samples. By having a smaller number in each leaf, more splits must be performed which makes the decision tree have higher variance but lower bias. Due to the high dimensionality of this data and its high variance, a lower value was found to produce better results but also increased the training time.

5) Neural Net: The final classifier tested is the multilayer perceptron classifier, which is a neural network that has a final layer consisting of the same number of neurons as the number of classes, in this case 2. As a result there are many potential hyperparameters for this classifier, but the two that were picked are the L2 regularization penalty and the number of layers [11]. Overall, having a higher L2 term improved the performance of this classifier, with a value of 1 having the best performance. Also, having a lower number of layers performed better and trained faster. This is due to the reduced number of neurons to train allowing for faster training, which in turn allowed for lower variance, higher confidence results. This allowed for this classifier to correctly classify positive cases in this unbalanced dataset.

B. Generalizing Dataset

Keras provides a tool called ImageDataGenerator which allows for the generation of more data from a given dataset with slight modifications such as rotations, translation and colour shifting the image. This allows for the creation of new images from the current dataset which are slightly different to help improve the ability for the classifiers to generalization the data they can handle. This allows for the creation of more positive results to help reduce the imbalance in the dataset while keeping the data generalized. This was used to double the size of the positive samples in the dataset, helping to make the dataset closer to balanced. However, since each image is mostly not indicative of pneumonia (only 9% of each positive image is indicative of pneumonia on average), the dataset is still imbalanced against positive cases. The parameters that were adjust were flipping along the X-axis, shifting vertically up to 5%, shifting horizontally by 2%, rotating the image up to 3 degrees and zooming in on the image up to 5%. The change in the performance of the different classifiers is shown in Table III and Figure 5

TABLE III
CLASSIFIER PERFORMANCE IMPROVEMENT WITH EXPANDED DATASET

Classifier	Metrics (%)		
	Precision	***Recall***	***IOU***
Naive Bayes	0	0	0
Nearest Neighbors	0	1	0
Linear SVM	0	0	0
RBF SVM	0	13	10
Decision Tree	0	-1	0
Random Forest	0	0	0
Neural Net	0	-8	-4
AdaBoost	0	5	4

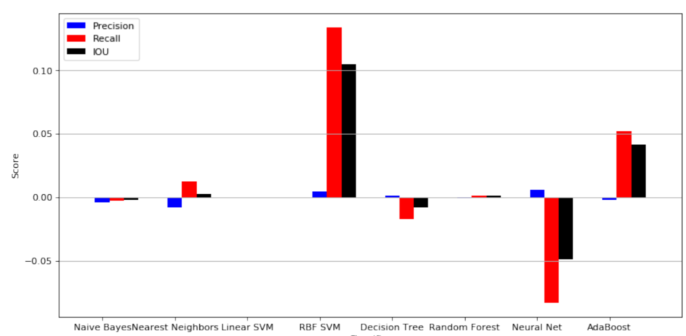

Fig. 5. The change in the performance of each of the classifier after expanding the dataset with additional data from ImageDataGenerator

It can be seen that the performance of Naive Bayes, RBF SVM and AdaBoost were all improved by the expanding the dataset. This suggests that they are underfitting and would benefit further from additional data. Reducing the imbalance helps remove the bias towards the negative case in the dataset which also helped AdaBoost's performance since it suffers from high bias.

Nearest neighbors, decision tree and random forest were not significantly affected by the expanded dataset due to their final result only using a fraction of the training data points, so few of the new data points were included in the final results.

It has also been found that the linear SVM and neural net classifiers performed significantly worse with the larger dataset. The neural net appears to have been impacted significantly by the addition of more outliers, which increased variance and reduced its ability to classify a feature vector as positive. As for the linear SVM, this saw improved IOU at the cost of a reduction in recall which means that the fewer feature vectors that were being classified as positive were less likely to be incorrect. This suggests that it now has lower bias but higher variance as a result of the expanded dataset making the SVM more generalized.

V. Conclusion

In conclusion, it has been found that using a classifier on the feature vectors output from a deep neural network (DenseNet121) pre-trained on the ImageNet dataset are effective at finding areas in chest X-ray images that indicate pneumonia. Due to this dataset being imbalanced towards negative cases and the feature vectors returned from DenseNet121 having 1024 dimensions this was a challenging dataset to classify for many conventional classifiers. Overall, it was found that the best classifier in terms of maximizing IOU was RBF SVM at 27% due to its ability to handle high dimensionality well and with a high enough C parameter was able to also handle the imbalanced nature of this dataset as well. The benefit of this method is that the training is done relatively fast and takes advantage of two established techniques together to produce effective results on this dataset. Due to densely connected networks being a well studied neural network architecture [4] the logic behind the generation of feature vectors can be better understood. Together with SVMs whos decision making can also be easily explained due to its support vectors that it picks the technique presented in this paper can be more easily understood why it picks the areas that it does versus a custom neural network. This is important in the medical field as the FDA needs to understand how any medical dvice works before approving it, they cannot be black boxes.

Some future work that would benefit this project would be testing the combination of DenseNet121 and SVM classification on different imaging datasets aside from CXR images. These include other medical or general imaging datasets that are interested in finding areas of interest in images. Also testing of different pre-trained deep neural networks would be beneficial to determine if there are other architectures that are better at being classified than those output by DenseNet121.

References

[1] Centers for Disease Control and Prevention. https://www.cdc.gov/nchs/fastats/pneumonia.htm

[2] https://www.mayoclinic.org/diseases-conditions/pneumonia/symptoms-causes/syc-20354204

[3] Kelly B. The Chest Radiograph. Ulster Med J 2012;81(3):143-148.

[4] M. D. Zeiler, R. Fergus, "Visualizing and Understanding Convolutional Networks" https://arxiv.org/pdf/1311.2901v3.pdf

[5] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[6] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[7] Statistics Canada. Table 13-10-0394-01 Leading causes of death, total population, by age group.

[8] Gao Huang and Zhuang Liu and Kilian Q. Weinberger, "Densely Connected Convolutional Networks"

[9] Brady Kieffer, Morteza Babaie, Shivam Kalra, and H.R.Tizhoosh "Convolutional Neural Networks for Histopathology Image Classification: Training vs. Using Pre-Trained Network" https://arxiv.org/pdf/1710.05726.pdf

[10] Charu C. Aggarwal, Alexander Hinneburg, and Daniel A. Keim, "On the Surprising Behavior of Distance Metrics in High Dimensional Space" https://bib.dbvis.de/uploadedFiles/155.pdf

[11] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors" https://arxiv.org/pdf/1207.0580.pdf

[12] Barrientos, Ronald, et al. "Automatic detection of pneumonia analyzing ultrasound digital images." Central American and Panama Convention (CONCAPAN XXXVI), 2016 IEEE 36th. IEEE, 2016.

[13] Arabi, Punal M., et al. "Identification of ground glass structure in lung CT images by computer aided diagnosis technique." Intelligent Systems and Control (ISCO), 2016 10th International Conference on. IEEE, 2016.

[14] Stephanakis, Ioannis M., George K. Anastassopoulos, and Aggelos D. Tsalkidis. "Multiresolution autoregressive filtering for pneumonia detection in medical images." Digital Signal Processing, 2002. DSP 2002. 2002 14th International Conference on. Vol. 2. IEEE, 2002.

[15] Zenteno, Omar, Benjamn Castaeda, and R. Lavarello. "Spectral-based pneumonia detection tool using ultrasound data from pediatric populations." Engineering in Medicine and Biology Society (EMBC), 2016 IEEE 38th Annual International Conference of the. IEEE, 2016.

[16] Amrulloh, Yusuf, et al. "Cough sound analysis for pneumonia and asthma classification in pediatric population." Intelligent Systems, Modelling and Simulation (ISMS), 2015 6th International Conference on. IEEE, 2015.

[17] Song, Insu. "Diagnosis of pneumonia from sounds collected using low cost cell phones." Neural Networks (IJCNN), 2015 International Joint Conference on. IEEE, 2015.

[18] Sharma, Abhishek, Daniel Raju, and Sutapa Ranjan. "Detection of pneumonia clouds in chest X-ray using image processing approach." Engineering (NUiCONE), 2017 Nirma University International Conference on. IEEE, 2017.

[19] Rajpurkar, Pranav, et al. "Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning." arXiv preprint arXiv:1711.05225 (2017).

[20] Huang, Gao, et al. "Densely connected convolutional networks." CVPR. Vol. 1. No. 2. 2017.

[21] Szegedy, Christian, et al. "Inception-v4, inception-resnet and the impact of residual connections on learning." AAAI. Vol. 4. 2017.

[22] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 2818-2826)

[23] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014)

[24] SPan, Sinno Jialin, and Qiang Yang. "A survey on transfer learning." IEEE Transactions on knowledge and data engineering 22.10 (2010): 1345-1359.

[25] Razavian, Ali Sharif, et al. "CNN features off-the-shelf: an astounding baseline for recognition." Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on. IEEE, 2014.

[26] Kieffer, Brady, et al. ”Convolutional Neural Networks for Histopathology Image Classification: Training vs. Using Pre-Trained Networks.” arXiv preprint arXiv:1710.05726 (2017).