**Classification of Pneumonia-Related Diseases Using Chest X-Ray Images**

**Author:**

Ken Trinh, Trevor Dalton, Sudhir Suvva

**Abstract:**

Pneumonia, is a flu-like symptom lung infection that causes fluid and inflammation build up in the affected lung and is detectable by x-ray imaging. However, it is quite difficult to differentiate seasonal flu virus vs bacteria related pneumonia. Throughout the years, many chest x-rays were taken from patients, and efforts have been made to categorize them based on the type of disease causing agent. We believe that through image processing techniques and machine learning technology, we can better understand the differences between healthy and infected patients. We hope our results will contribute to future research into this topic.

**Project Overview**

Typically, x-ray images are taken by applying a beam of high radiation photon toward the target. This beam, after coming into contact with human tissue, projects an image on a metal firm. Since soft tissues such as skin and organs cannot absorb high energy rays, these beams pass through, imprinting only dense material like bone onto the film. In terms of infectious diseases, pneumonia in our case, physicians use x-ray imaging to look for white spots or infiltrates in the lung. The X-Rays can be taken either from anterior or PA (post anterior). The quality of x-rays are better when they are taken from PA. This white spot can be seen in the red circle region in Fig. 1 right x-ray image.

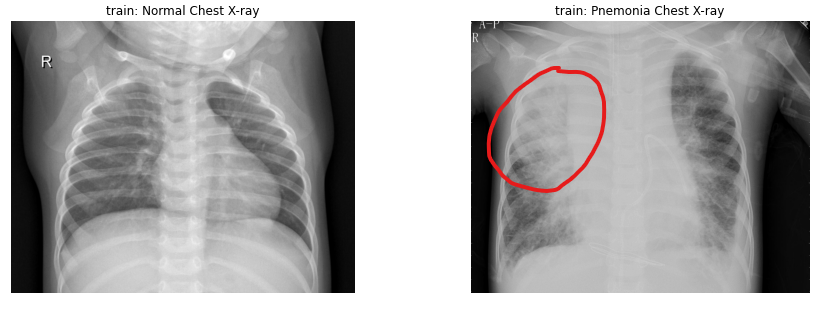
****

Fig. 1: Normal Chest X-Ray vs pneumonia Chest X-Ray with noticeable white spot/cloud

Naturally, a trained radiologist would be able to distinguish these white spots easily. However, there are some cases where the white spots are not as apparent, leading to a misdiagnosis (Fig. 2). In this case, the normal chest x-ray actually shows some resemblance of white spots while the pneumonia patient x-ray is pretty clear.

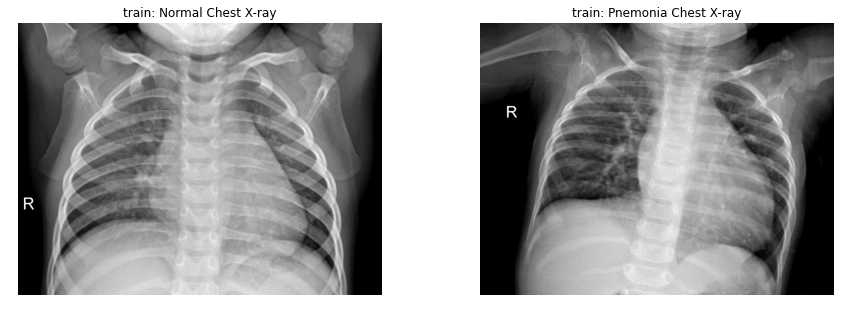


Fig. 2: Normal Chest X-Ray vs pneumonia Chest X-Ray with no noticeable white spot/cloud

For that reason, our research wishes to answer the following question: Are there any features that we can learn through a combination of image processing and machine learning to better detect pneumonia patients’ chest x-ray?

Earlier papers [[1](https://arxiv.org/abs/2004.12823), [2](https://arxiv.org/abs/2004.05405)] have found concerns with detecting COVID-19 from chest x-rays. In particular, it was found that there is a risk of models learning features that are entirely independent from the lungs that are unrelated to COVID-19. To properly address such a question, we wanted to experiment with different categorization systems to distinguish the variety of pneumonia causing agents (COVID-19, Flu Virus, or Bacteria). We believe that image processing is crucial in this project since some pneumonia patients' x-rays show very subtle differences from healthy patients. First, we propose extracting Hue Saturation Value (HSV) on each image as a way to characterize its saturation value. Second, we propose using local binary patterns to better represent similar regions of white clouds. Finally, since we noticed that some of the patients x-ray images have slight variation in origination (Figure 2) and that we will be augmenting the data to create a more representative sample size, afterwich, we will want to use SIFT features to capture these variations..

When it comes to image classification, Convolution Neural Network (CNN) is considered state of the art. In the article Convolutional Neural Networks proposed by Yamashita et al. [[3](https://link.springer.com/article/10.1007/s13244-018-0639-9)], the authors described and applied CNN on various classification tasks on radio imaging. Since this proposed work is relatively similar to Yamashita et al., we will be leveraging the CNN architecture. While CNN is state of the art, we wanted to experiment with different models, particularly Logistic Regression, Support Vector Machine, and Gradient Boosting. We decided to use these lower level Machine Learning algorithms alongside the state of the art CNN since Logistic Regression is an underlying mechanism of Neural Network. Support Vector Machine algorithm, on the other hand, tends to be more effective at higher dimensional spaces [[5](https://www.nature.com/articles/nbt1206-1565#citeas)]. Finally, we can leverage the learning from weak sets of learners from Gradient Boosting algorithm to give us insights into when and where for a feature the tree splitted [[6](https://journals.sagepub.com/doi/pdf/10.1177/15330338221087828)], which could be useful for us to better understand pneumonia x-rays.

**Data**

The data used in this project (collected by Joseph Paul Cohen of the University of Montreal) are made available on [Kaggle](https://www.kaggle.com/datasets/praveengovi/coronahack-chest-xraydataset). This dataset contains 5910 chest x-rays of normal and pneumonia affected patients: 1576 from unaffected patients; 2 from stress-smoking caused pneumonia; 1555 from viral causes; and 2781 from bacterial causes. Figure 3 outlines the sample size of this dataset:

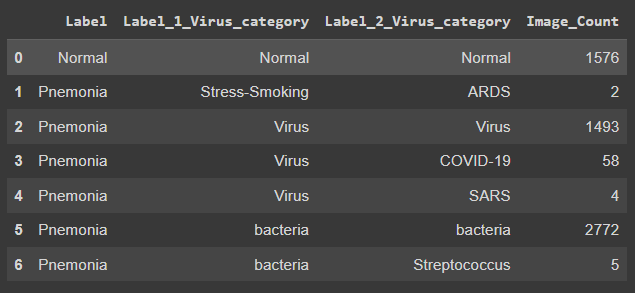


Fig. 3: Dataset Sample Size

In our research, we’re using the second label column “Label\_1\_Virus\_category” as the target label for each image. Here, it is noted that the “Stress-Smoking” category only contains two images so we will be excluding this category from our experiments. The remaining categories, “Normal”, “Virus”, and “Bacteria” represent x-ray images from people without pneumonia, pneumonia patients infected by a virus, and pneumonia patients infected by bacteria, respectively. All x-ray images came in different shapes and sizes, for that reason, we’ve decided to resize each image to 224 x 224 pixels to standardize feature extraction and model training.

For data augmentation of the images in these remaining categories, we decided to generate five randomized images using the following parameters. This is visually represented in Figure 4 to fully control for the randomized effect of patients’ orientation when taking the x-ray:

* Rotation: 0 to 20 degrees
* Shear: 0 to 0.05
* Zoom: 0 to 0.05
* Illumination: 0.75 to 1.25

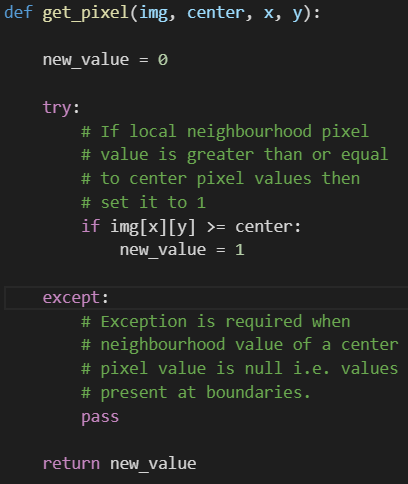


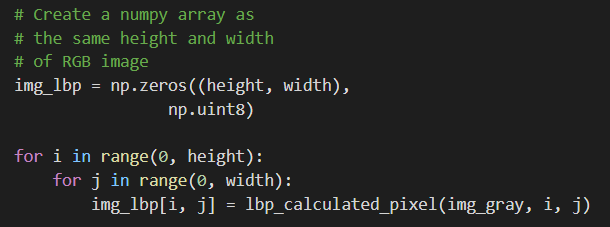
Fig. 4: Dataset Augmentation

**Feature Engineering / Vectors**:

Once augmented, each image is matched to one of three aforementioned categories: Normal, Bacteria, Virus, using the provided metadata. Subsequently, three sets of features are extracted from each image. Each individual feature is then used to train a classification model based on the three classes mentioned above.

1. Local Binary Pattern (LBP): LBP helps in understanding the local representation of the texture of an image. The incredible power of LBP lies in its ability to differentiate tiny differences in texture and topography. By encoding commonalities between images of various classes we allow for comparison of their features with that of another image. Through this comparison, we can determine the level of similarity between the labeled representation and an unseen image and can calculate the probability of similarities. To extract Local Binary Pattern, we utilized Patra’s implementation using openCV [[8](https://www.geeksforgeeks.org/create-local-binary-pattern-of-an-image-using-opencv-python)]. The following code outline the procedure and a representation of the output, as a 224 x 224 feature map is displayed below in Figure 5





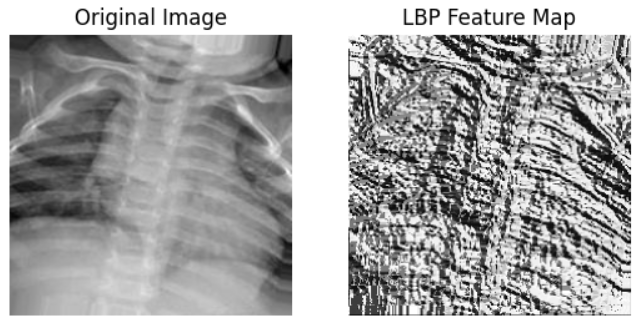


Fig. 5: Generated LBP Feature Map

1. Hue Saturation Value (HSV): HSV provides a numerical readout of the image that corresponds to the color names contained in the image. We’re looking to use this value as a representation of the degree of saturation in our x-ray images. To extract HSV, we utilized openCV pre-written API to generate a vector of feature for each image:

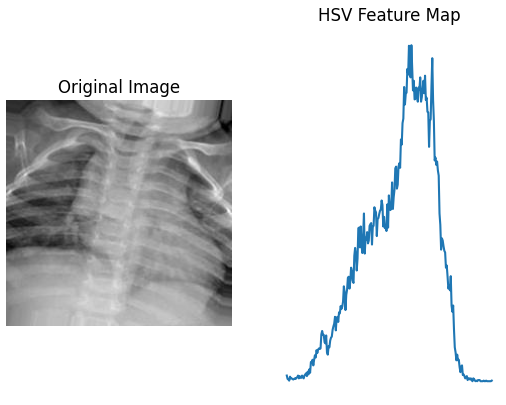
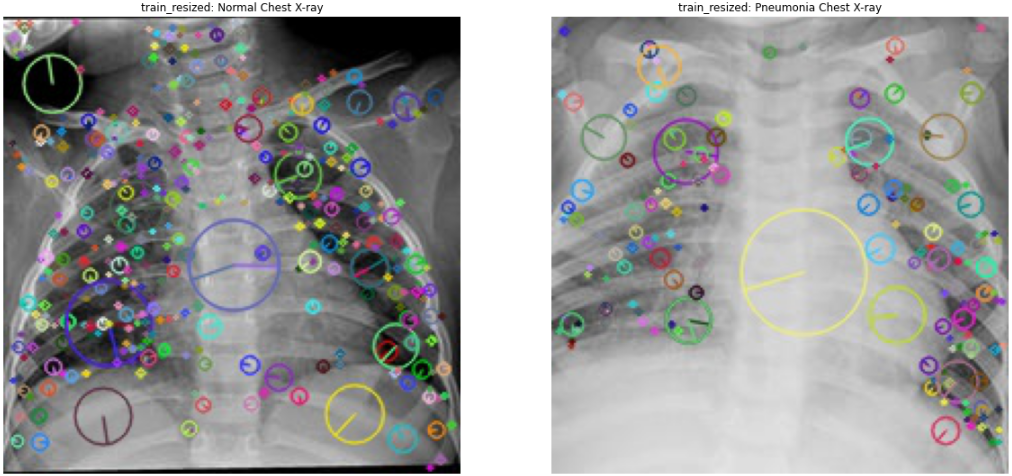


Fig. 6: Generated HSV Feature Map

1. Scale Invariant Feature Transformation (SIFT): It is used to identify distinctive features from the images and functions without concern for image scale, rotation and translation.

Here it is noted that SIFT features are often used in object detection and are generally applied on a reference object image. Since the purpose of this research is to characterize white clouds in pneumonia patients versus normal patients, we’re interested in whether or not there are substantial differences between SIFT features between the two image categories. In order to determine that answer, we set up a hypothesis test by first extracting the SIFT features of two samples, one sample with normal patients and patients with pneumonia and one sample with two different patients with pneumonia. Figure 7 showcases an example of the the SIFT features for the two samples:



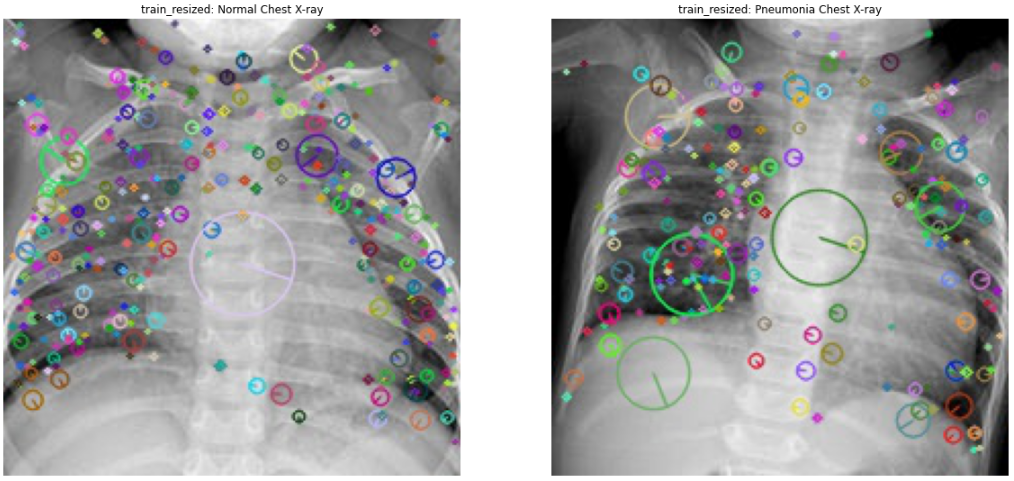


Figure 7: Normal vs patients with pneumonia sample (top) and two different patients with pneumonia sample (bottom)

For each sample, we applied a K-Nearest-Neighbor match with k =2 and used a ratio test to select the matches between two images with the shortest distances (Figure 8).

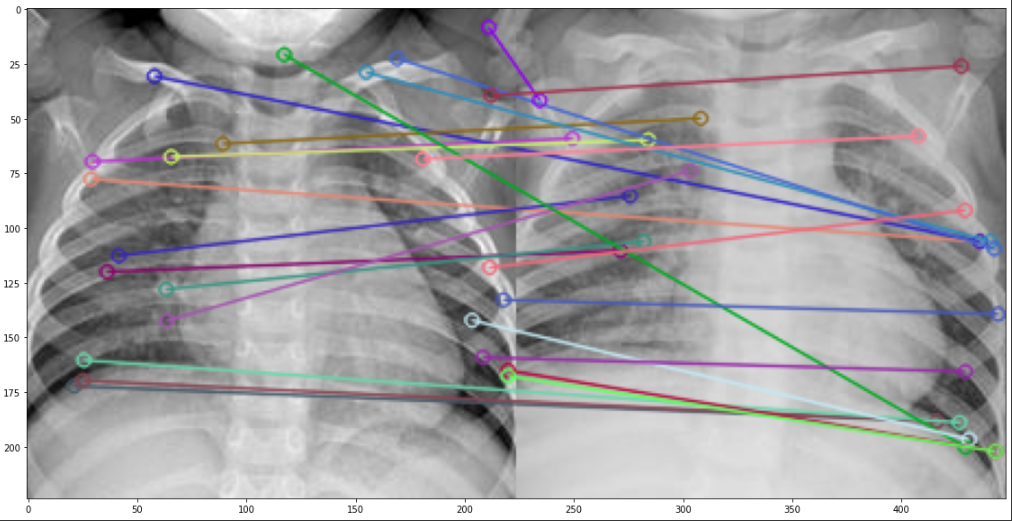


Figure 8: KNN Matches of SIFT Features

We repeat this process 1000 times for each sample to generate two vectors of selected matches. It is noted that we decided to perform the test on the natural log of each test to meet the normal distribution assumption.

From here we applied a two-samples t-test between the two samples at 95% confidence level (Figure 9). To set up the test, we form a null hypothesis that the average matches between the two samples are the same and an alternative hypothesis that the average matches between the two samples are not the same. From our test, we obtained a p-value of 2.58e-5, which is significantly less than the alpha value of 0.05 at the 95% confidence level. This result suggests that there is enough evidence to reject the null hypothesis, indicating that we could somewhat use SIFT features to distinguish pneumonia patients from normal patients.

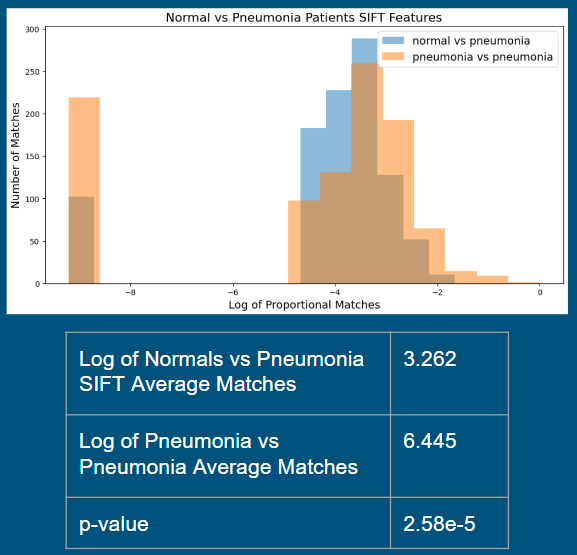


Figure 9: Two samples hypothesis test on SIFT features between normal vs patients with pneumonia sample (blue) and two different patients with pneumonia sample (orange)

**Dimensional Reduction and Analysis**

The extracted features, once flatten, are of order of 10,000+ dimensions. When that number is scaled with the 20,000 images, the training time could be impacted. To avoid large computational time, we explored two additional dimensional reduction methodology, PCA and t-SNE. For PCA, we used the explained variance plots on each of the target labels to get a better sense of the number of principal components needed for modeling. For t-SNE, we decided that it is best to control for 2 principal components to maximize the computational efficiency. Figure 10 to Figure 14 display dimensional selection of PCA and comparison between X-principal components PCA vs 2 dimensional t-SNE for each of our proposed features.

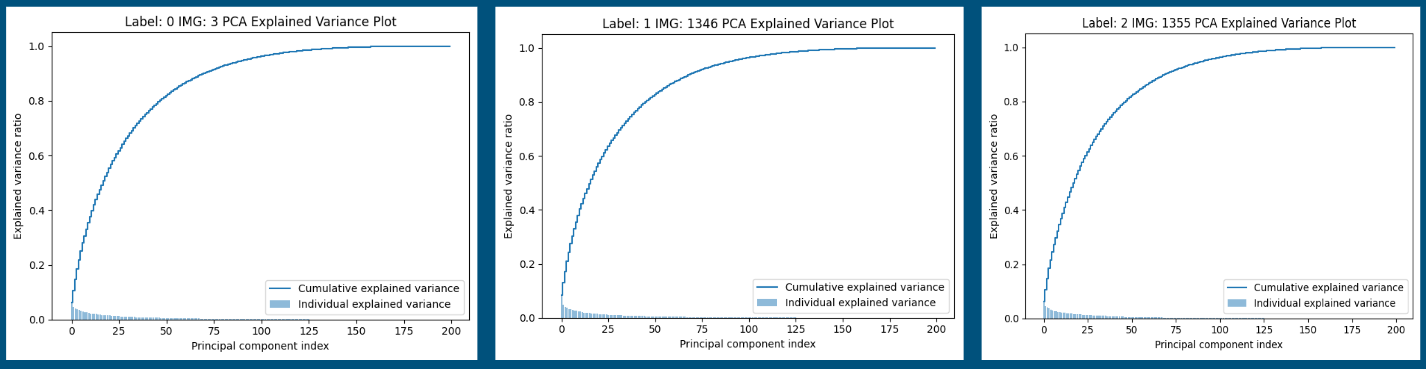


Figure 10: Local Binary Pattern PCA explained variance plot

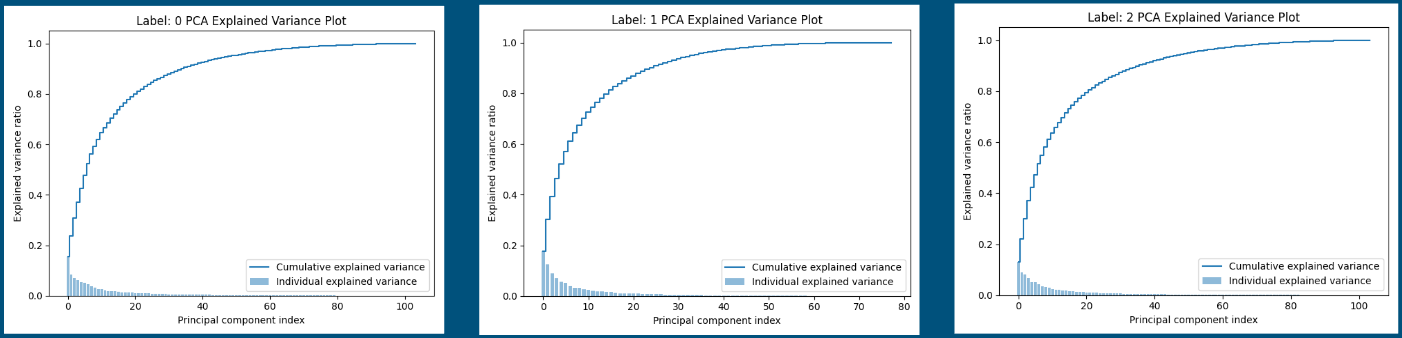


Figure 11: SIFT Feature PCA explained variance plot

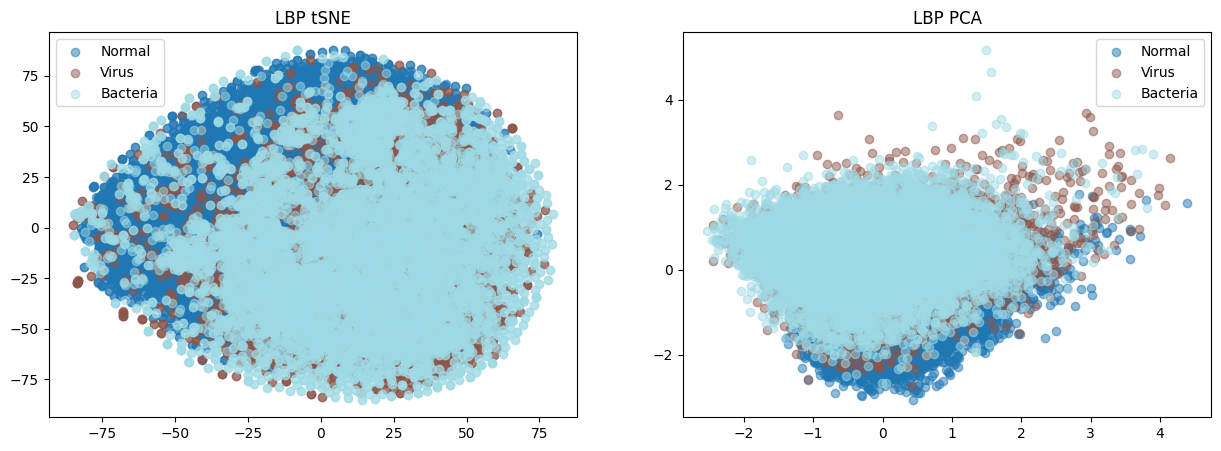


Figure 12: LBP feature t-SNE vs PCA decomposition comparison across three classes

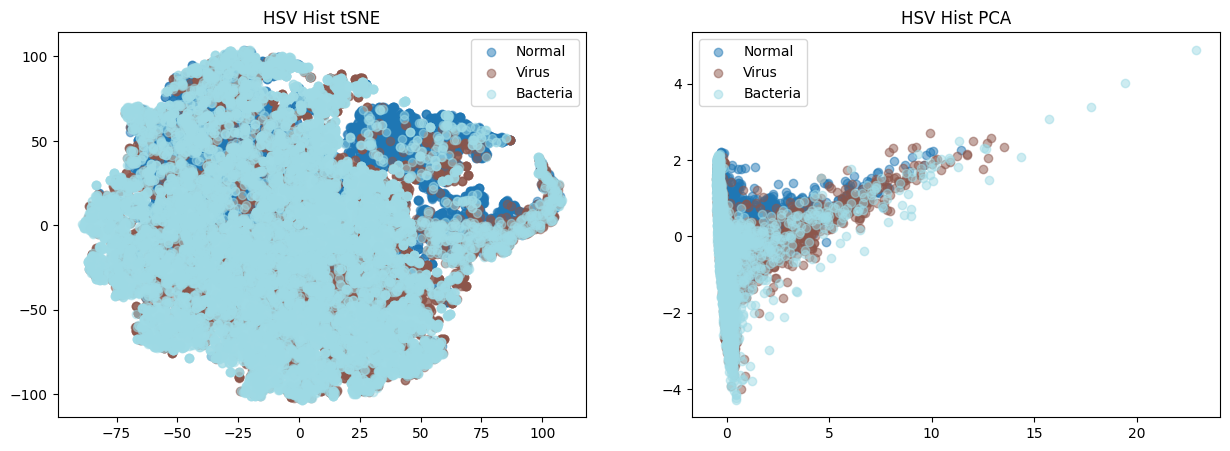


Figure 13: HSV feature t-SNE vs PCA decomposition comparison across three classes

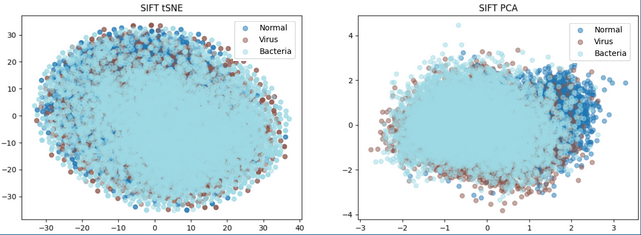


Figure 14: SIFT feature t-SNE vs PCA decomposition comparison across three classes

For our LBP feature (Figure 10), each of our image class explained variance plots suggests that approximately 50 principal components would explain ~80-90% of the total variance. For our SIFT feature (Figure 11), each of our image class explained variance plots suggests that approximately 20 principal components would explain ~80-90% of the total variance. Further study showed that t-SNE decomposition does not really distinguish between the classes when compared to PCA decomposition on all three proposed features (Figure 12 - 14)

**Model**

Since Convolutional Neural Network (CNN) isthe state of the art for solving classification problems, we’ve decided to build and train a simple CNN on both resized images and augmented images and use their test results as a starting point for benchmarking lower level models using the proposed extracted features. Due to computation constraints in this research, the CNN contains 2 sets of convolutional layers with 32 5x5 filters on the first layer and 32 2x2 layers on the second layer. The output layer is max-pooled prior to the next layer. Two additional hidden layers are used at the end with a 10% dropout rate to improve training and control for overfitting of the dataset and 15 training generations (epochs) were used. Figure 15 reported the test results from the model architecture displayed in Figure 16.

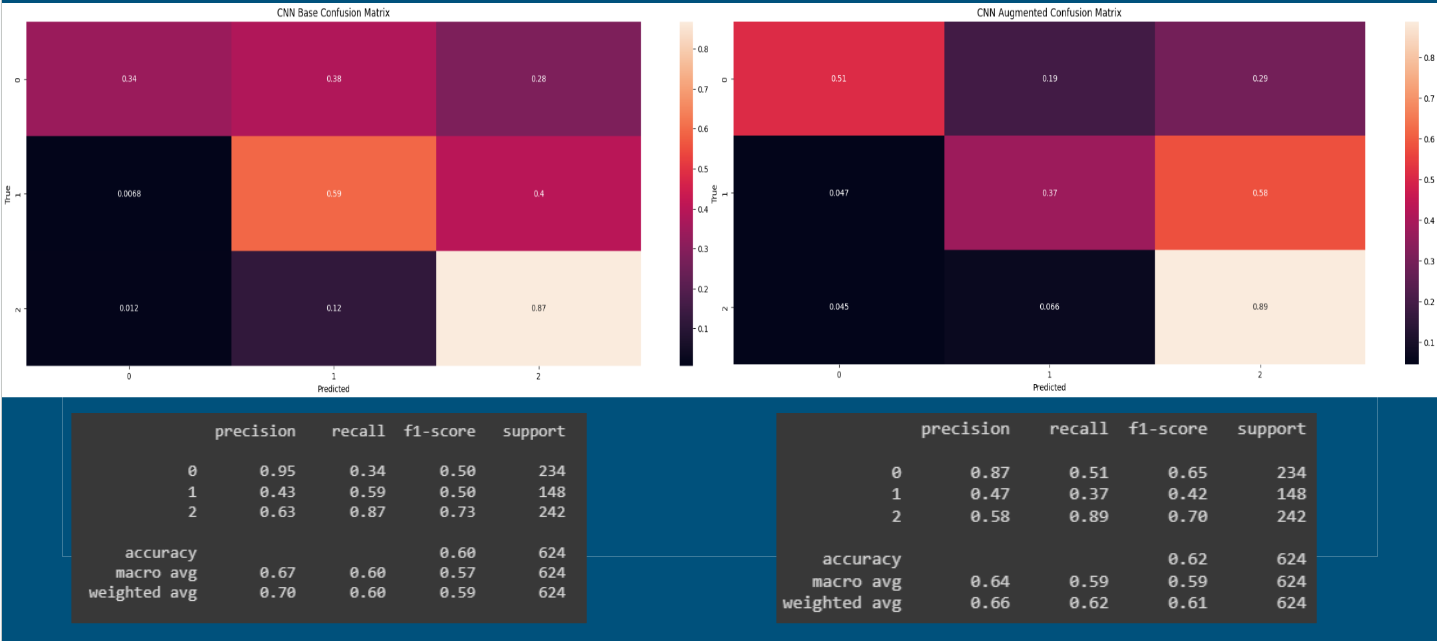


Figure 15: Simple CNN classification using only resized images (left) and using resized/augmentation images (right)

From the results, the state of the art CNN model was only able to obtain an accuracy of 60-62%. Therefore we have reason to believe that the extracted features could improve these metrics while maintaining a high precision and recall.

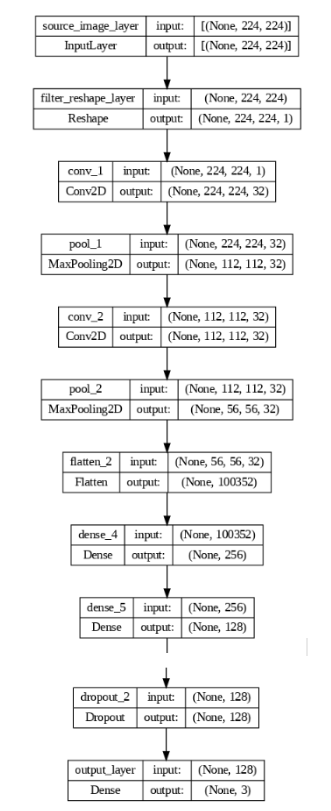


Figure 16: Simple Convolutional Neural Network Architecture

We did some further investigation on the SOTA CNN and decided to fine-tune a pre-trained ResNet50 (Figure 17). Surprisingly, compared to a simple CNN, the fine-tuned ResNet50 performance is worse. While ResNet was able to sufficiently categorize Normal vs pneumonia patients, it wasn’t able to distinguish between viral infection and bacterial infection. Mostly, pneumonia patients were all categorized as being bacterial infected. Furthermore, it is noted that in each of the categories, ResNet50 showed some residuals in other categories (ex: pneumonia in normal and vice versa). All of these phenomena could be explained by the resemblance of “white-cloud” effects in both normal vs pneumonia patients images noted earlier.

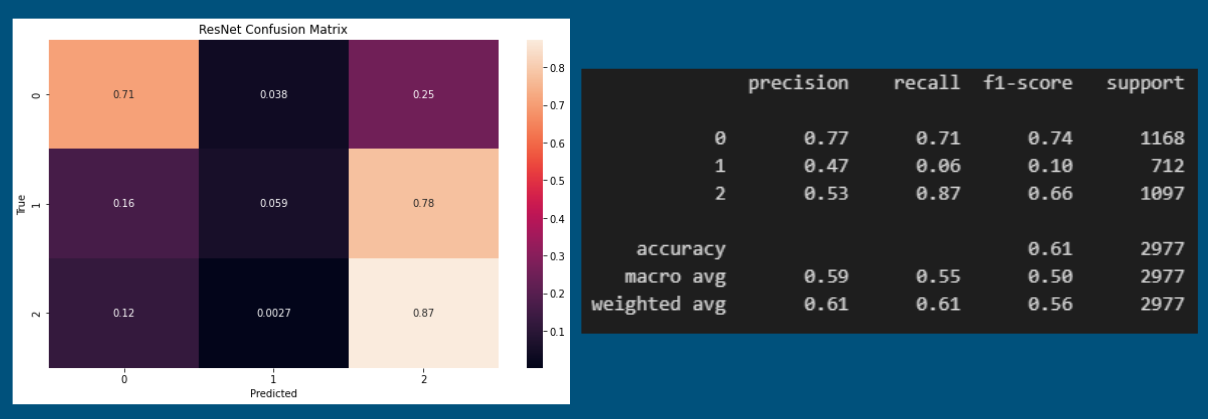


Figure 17: Fine-tuned ResNet 50 using resized/augmentation images

**Modeling:**

In all the experiments below, we utilize the dimensionality reduced features to save computation time to train three lower level models: simple logistic regression, support vector machine classifier, and a gradient boosting classifier. We also use an autoML package, published by [MLJar](https://mljar.com/), as a way to gridsearch across all features augmentation, hyperparameters, and models.

**LBP Feature Model Results**

In the LBP feature experiment, we utilized 50 principal components PCA dimensionality reduced LBP features to capture 80-95% of the total variance. After training, we use the model to predict the classes on a held-out training set. Figure 18 to X denotes the confusion matrix for each trained model.

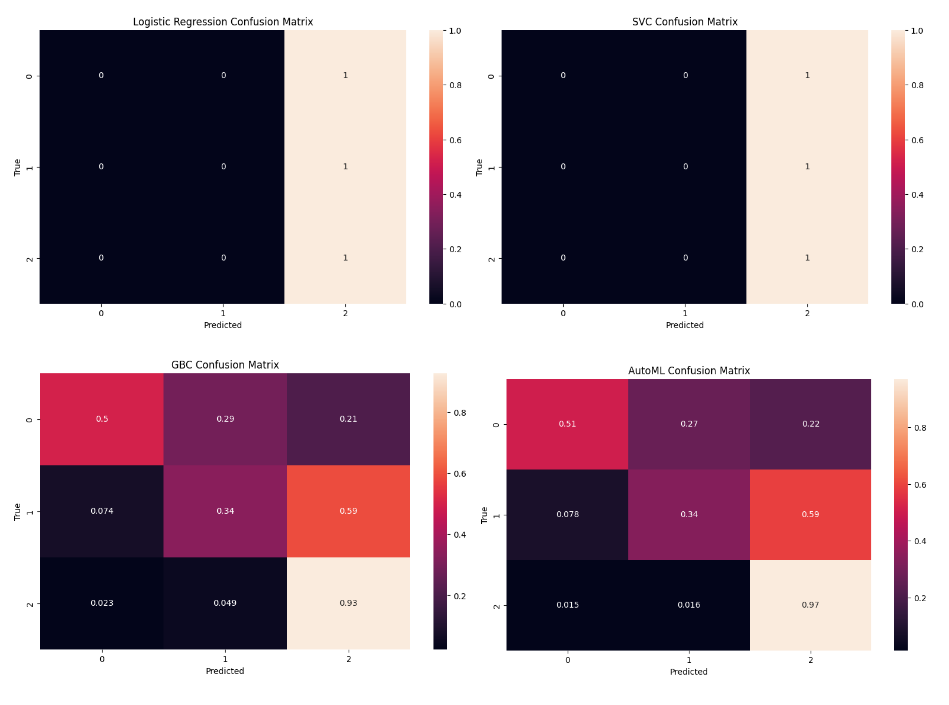


Figure 18: LBP features PCA dimensionality reduction model confusion matrix on testing data set

Note that both the logistic regression and the support vector machine models failed to predict any of the 1st and 2nd categorical variables, normal patient and pneumonia patients that were infected by virus. Furthermore, the other two models tend to favor prediction of the 2nd categorical variables, or pneumonia patients that were infected by bacteria. Interestingly, using the LBP features, the other two models declassified the 0th category, normal patients, as the other two variables approximately 50% of the time. As noted earlier, LBP was used to represent a human's anatomical structure. Thus, this phenomena could be reasoned as the LBP features weren't able to fully distinguish the differences between the white “white-cloud” of viral infection and bacterial infection as well as the “white-cloud” like features in normal patients.

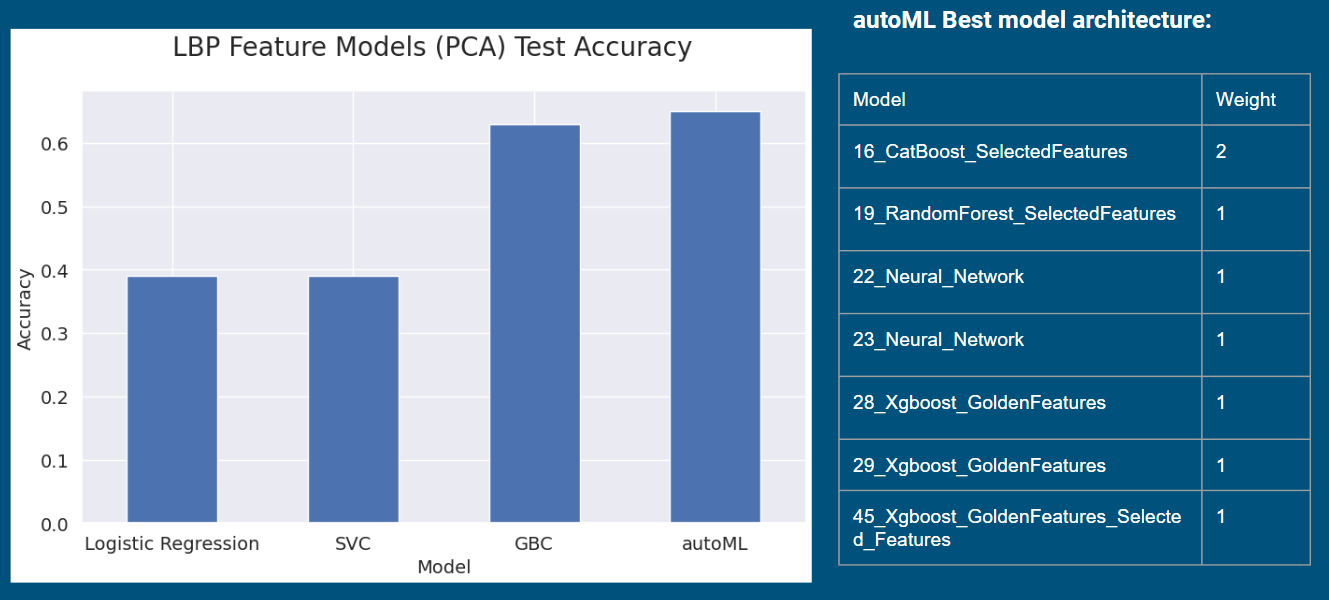
****

Figure 19: SIFT features model accuracy and autoML best architecture

In Figure 19, we extract the performance accuracy of the model on the held-out test set. Here, our results suggest that the autoML has the overall best accuracy of 65%, which managed to match the performance of the CNN by 2-3%. Further digging into the architecture of the autoML, we noted that there is a lot of usage of Neural Network and Xgboost with golden features in the supposedly “best” architecture. We reasoned that the autoML process is trying to emulate the usage of the CNN here while also deriving some additional features to capture additional information.

**HSV Feature Model Results**

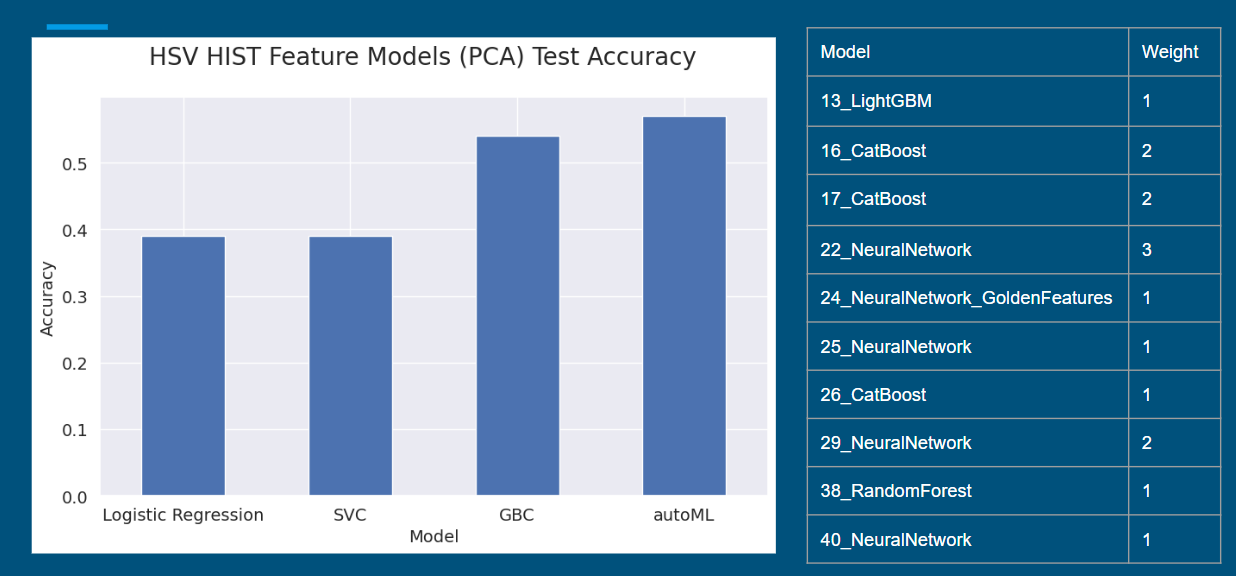
****

Figure 21: HSV features model accuracy and autoML best architecture

In Figure 21, we extract the performance accuracy of the model on the held-out test set. Here, our results suggest that the autoML has the overall best accuracy of 57%, although the gradient boosting classifier yielded very similar results at 54% accuracy. From the evidence at hand, it appears that simple classifiers such as Logistic Regression and Support Vector Machines are not sophisticated enough to detect the presence of pneumonia from the HSV feature alone.

This may be a result of the fact that HSV histograms are sensitive to data augmentation as well as any particulars to the x-ray machine itself. For example, if one machine takes brighter x-rays (or the software brightens the x-rays) then it’s likely this may have had an adverse effect on the performance of the models. The simple models are forced only to rely upon the raw density of each luminosity which could have led to them being overfit on the training data. The Gradient Boosting Classifier and the AutoML (via Neural Networks) would be more robust because they could also compare the relative densities of certain luminosities to each other.

One major flaw in the current iteration of this feature is that there are 256 parameters yet each parameter doesn’t differ much from one to the next which is likely leading to overtraining. Thus, the HSV histogram feature could likely be improved by bucketing the luminosities through pooling.

| Logistic Regression Confusion Matrix | SVC Confusion Matrix |
| --- | --- |
| GBC Confusion Matrix | AutoML Confusion Matrix |

Figure 22: HSV features PCA dimensionality reduction model confusion matrix on testing data set

**SIFT Feature Model Results**

In the SIFT feature experiment, we utilized 20 principal components PCA dimensionality reduced SIFT features to capture 80-95% of the total variance. After training, we use the model to predict the classes on a held-out training set. Figure 23 to X denotes the confusion matrix for each trained model.

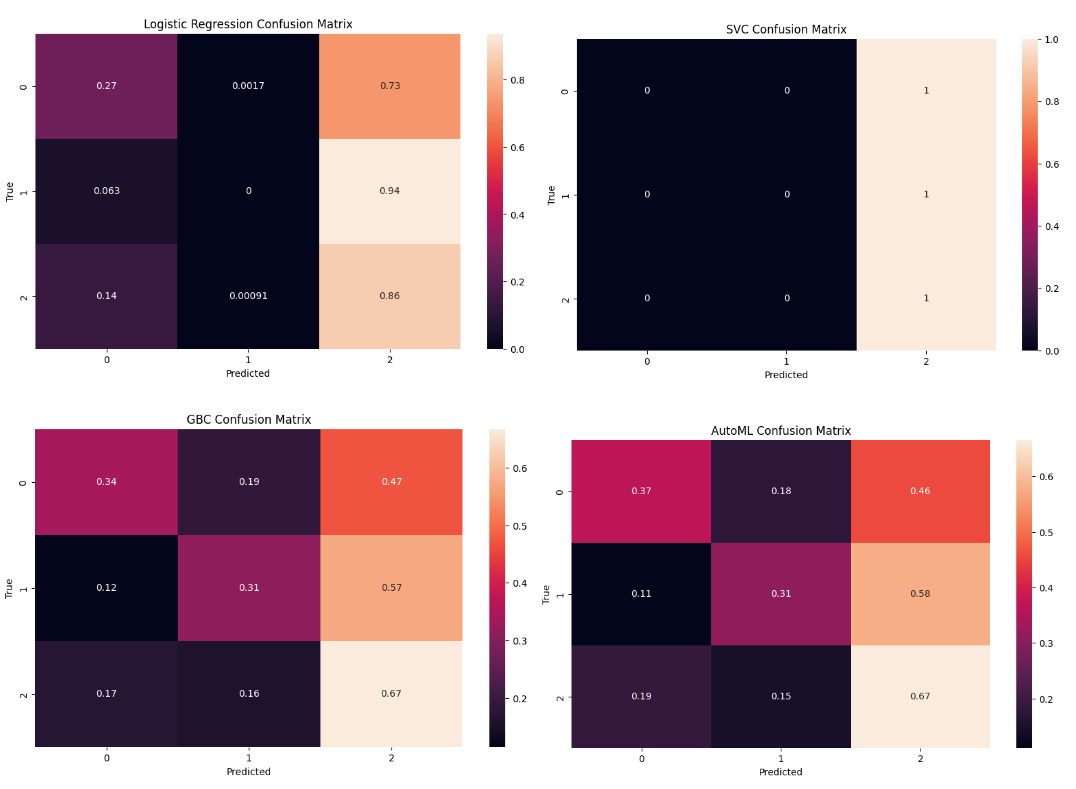


Figure 23: SIFT features PCA dimensionality reduction model confusion matrix on testing data set

Note that both the logistic regression and the support vector machine models failed to predict any of the 1st categorical variables, pneumonia patients that were infected by virus. Furthermore, the other two models tend to favor prediction of the 2nd categorical variables, or pneumonia patients that were infected by bacteria. While the results are not too far off the CNN outputs, this phenomena could be alluded to the fact that SIFT features weren't able to fully distinguish the differences between the white “white-cloud” of viral infection and bacterial infection.

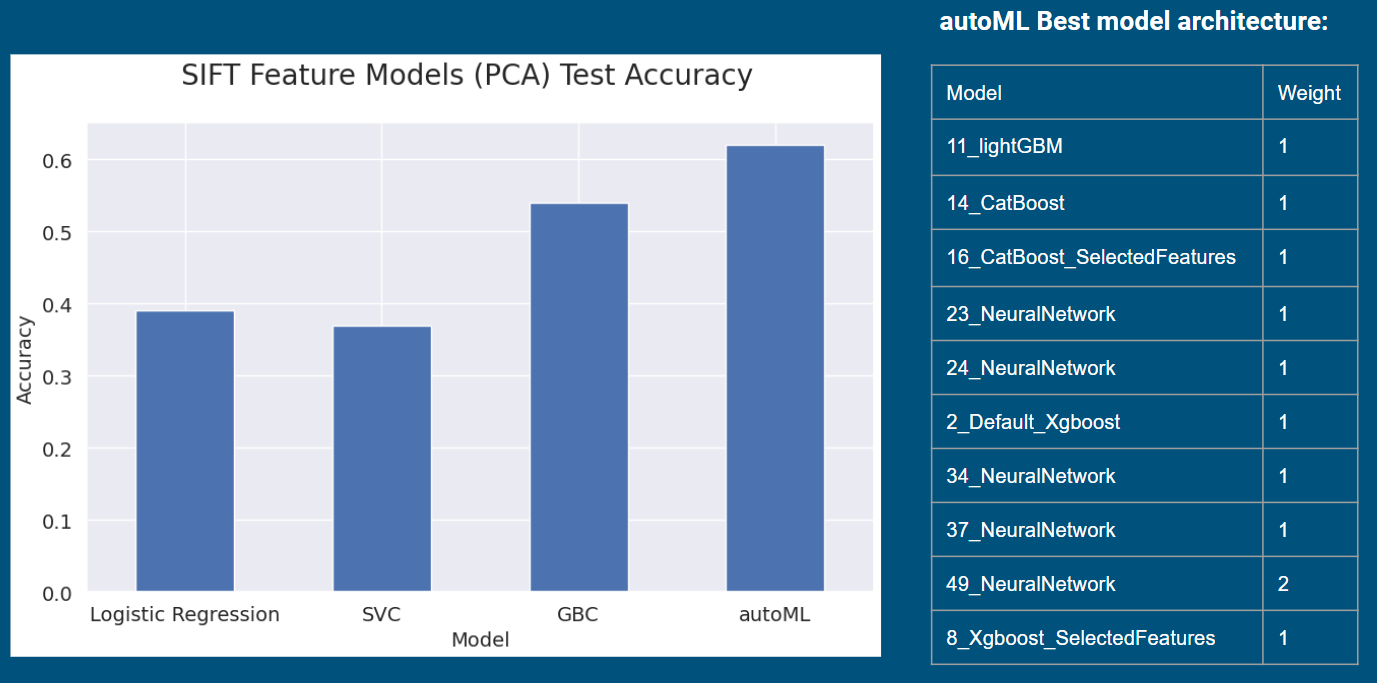


Figure 24: SIFT features model accuracy and autoML best architecture

In Figure 24, we extract the performance accuracy of the model on the held-out test set. Here, our results suggest that the autoML has the overall best accuracy of 62%, which only managed to match the performance of the CNN. Further digging into the architecture of the autoML, we noted that there is a lot of usage of Neural Network in the supposedly “best” architecture. We reasoned that the autoML process is trying to emulate the usage of the CNN here. Unfortunately, we would have to train a massive Neural Network to reach the same result as CNN if we are only using the SIFT feature.

**Combined Features Model Results**

In this final experiment, we take both the PCA dimensionality reduced and t-SNE dimensionality reduced across all features proposed in the earlier section. Figure 25 to 26 presented the confusion matrix for combined features using PCA and t-SNE dimensionality reduction across the three features, respectively

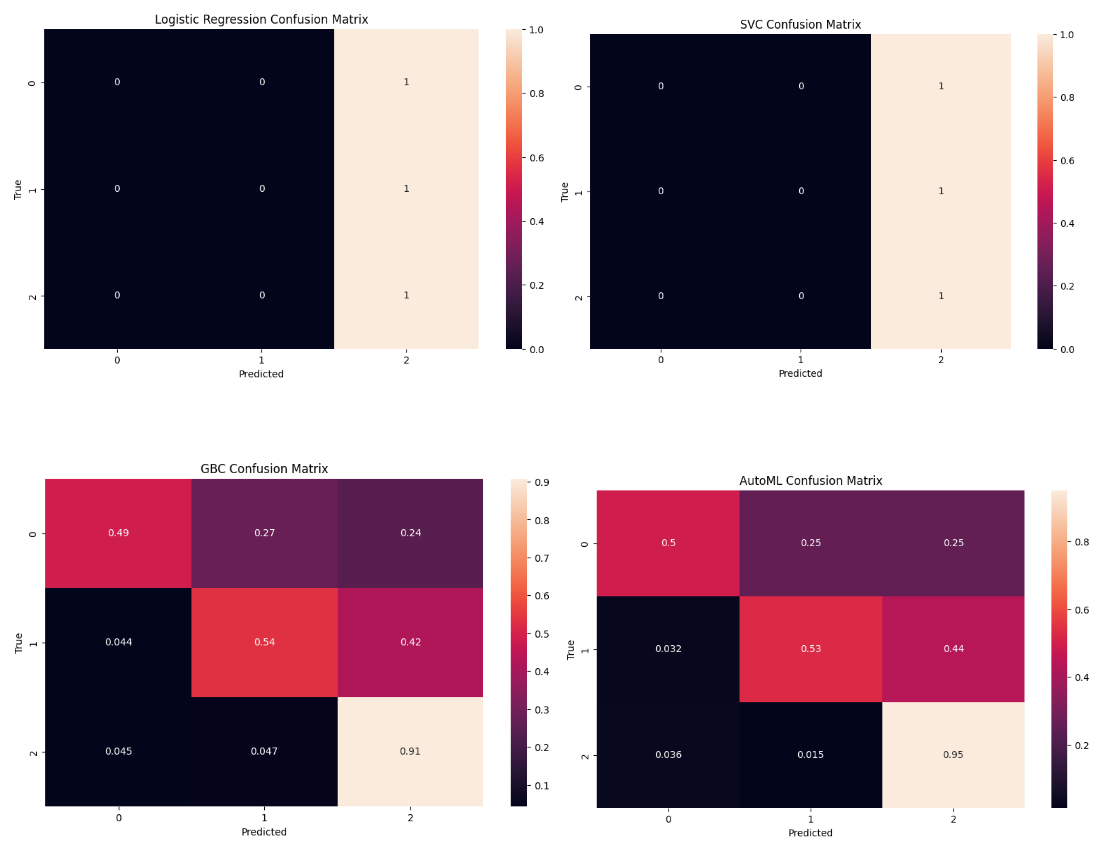
****

Figure 25: Combined features PCA dimensionality reduction model confusion

matrix on testing data set

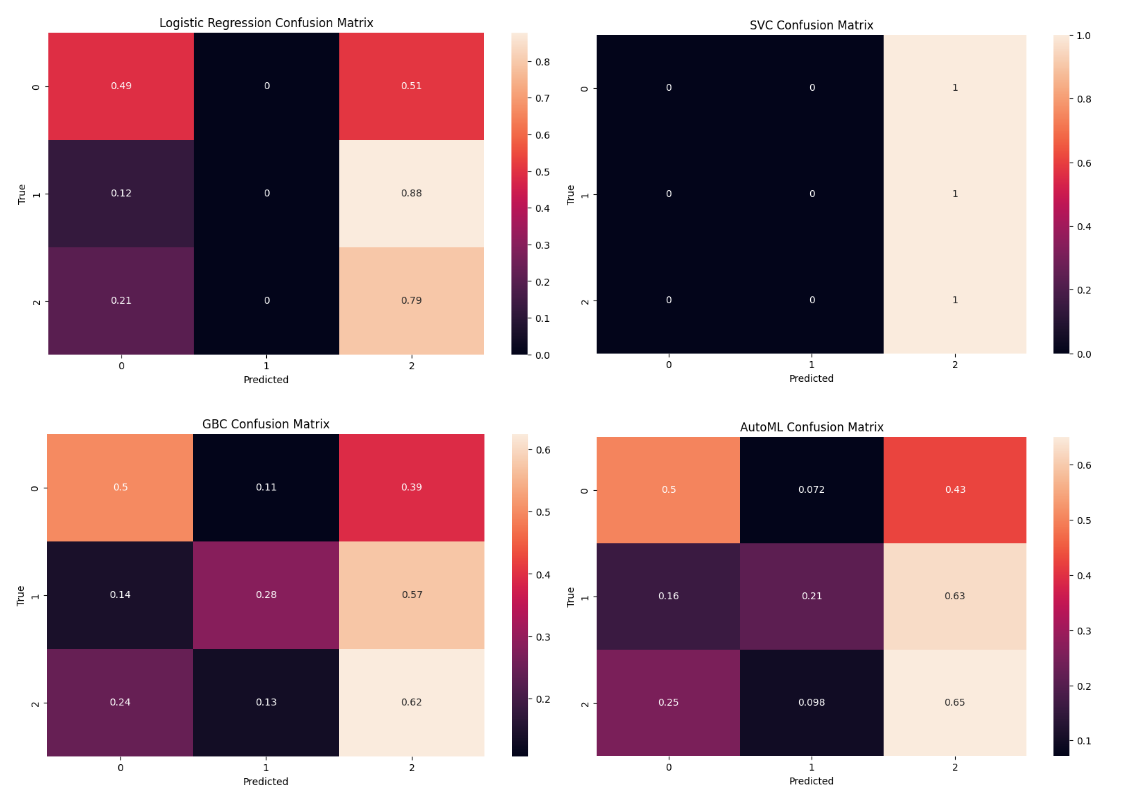
****

Figure 26: Combined features t-SNE dimensionality reduction model confusion

matrix on testing data set

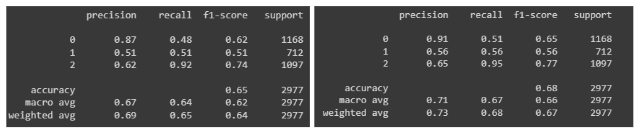


Figure 27: Gradient Boosting Classifier report (left) autoML report (right)

Interestingly, when using PCA dimensionality reduction on the three combined features, both logistic regression and the support vector machine models failed to correctly classify the features in each of the three labels. Rather, both models, by default, assign category 2 to all test images. This issue is only apparent in support vector machine classifiers when t-SNE is used. Our reasoning for this phenomenon is that there are some non-linear effects when all features are considered that these two lower level models were not able to be captured.

Similar to using just the SIFT features and the CNN models, both gradient boosting classifiers and autoML tend to favor prediction of the 2nd categorical variables, or pneumonia patients that were infected by bacteria. However, when using PCA for dimensionality reduction, the both Gradient Boosting and autoML confusion matrix shows be able to classify images across each of the three categories. Further investigation using a classification report shows that both models have high f1-score for class 0 and 2 (Figure 27). This translated to a 4-6% improvement in accuracy compared to that of the CNN outputs. While this improvement is marginal, using all three features does provide a positive impact in the classification of pneumonia patients. Finally, we’re still seeing the model favoring the classification of class 2 (bacterial infection) versus class 1 (viral infection). Once again, this phenomena could be alluded to the fact that SIFT features weren't able to fully distinguish the differences between the white “white-cloud” of viral infection and bacterial infection.

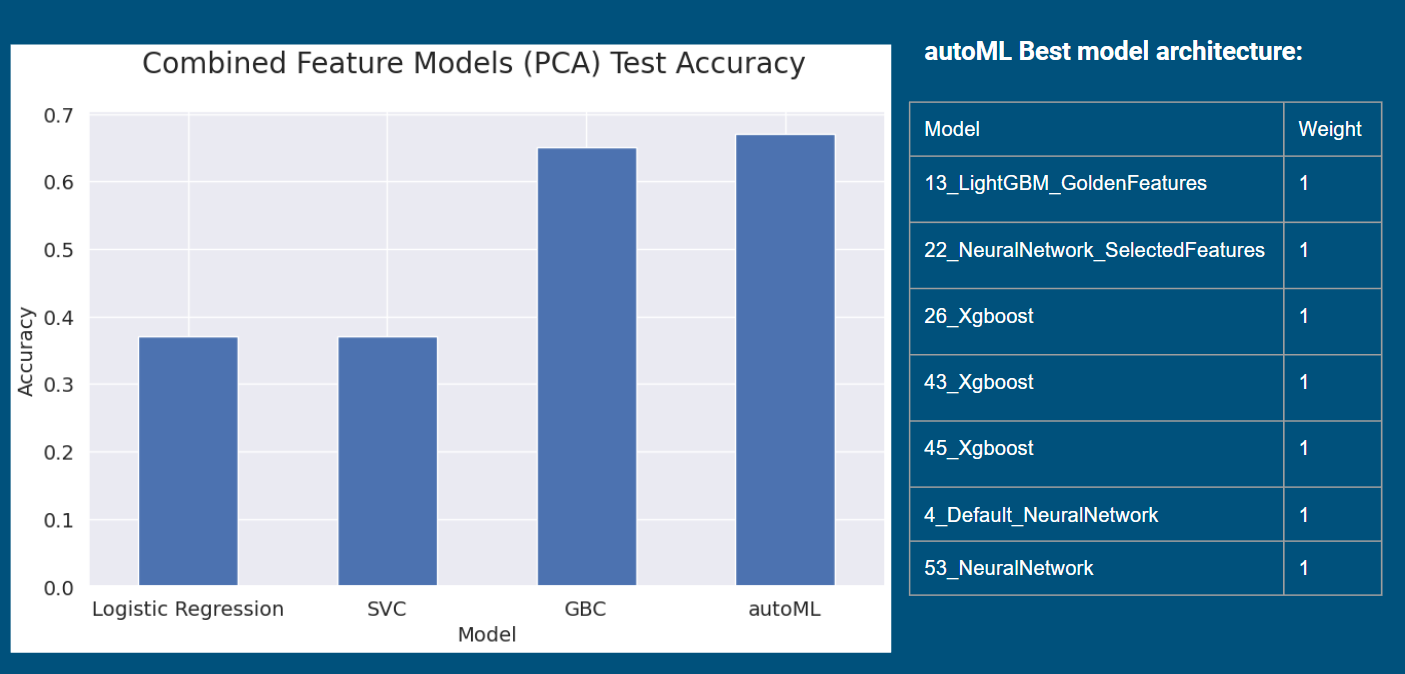
****

Figure 28: Combined features model PCA dimensionality reduction accuracy and autoML best architecture

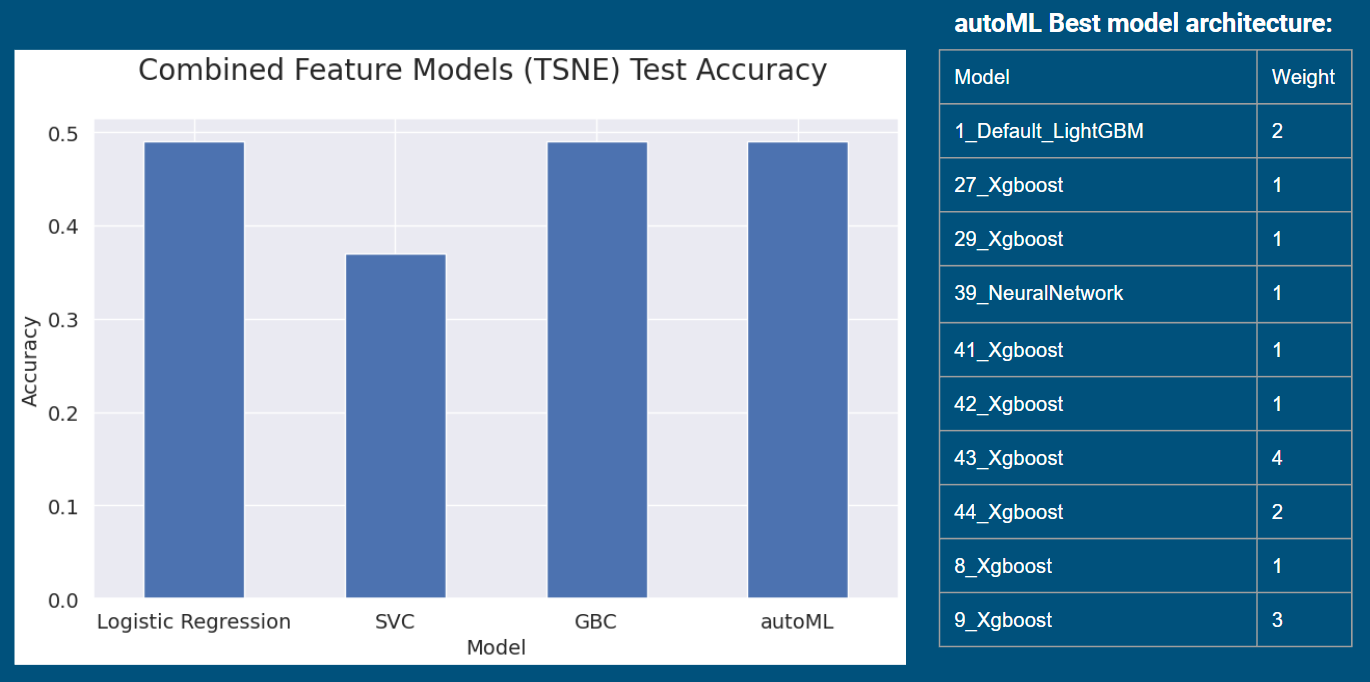
****

Figure 29: Combined features model t-SNE dimensionality reduction accuracy and autoML best architecture

In figure 28-29, we extract the performance accuracy of the model on the held-out test set. Here, our results suggest that the autoML has the overall best accuracy of 68% when using PCA decomposition across all three features. Further digging into the architecture of the autoML, we noted that there is a lot of usage of XGBoost in the supposedly “best” architecture. These results suggested that perhaps a more extreme XGBoost could better explain some additional variance in the set of features provided.

**Efficiency vs Accuracy Tradeoff**

In an effort to quantify the efficiency versus accuracy tradeoff between different combinations of features used in the model, we opted to select the best performance model across all of our test cases, the autoML process, and computed the training time as well as the accuracy performed on the held-out testing dataset. Figure 30 reports the training time and the testing accuracy.

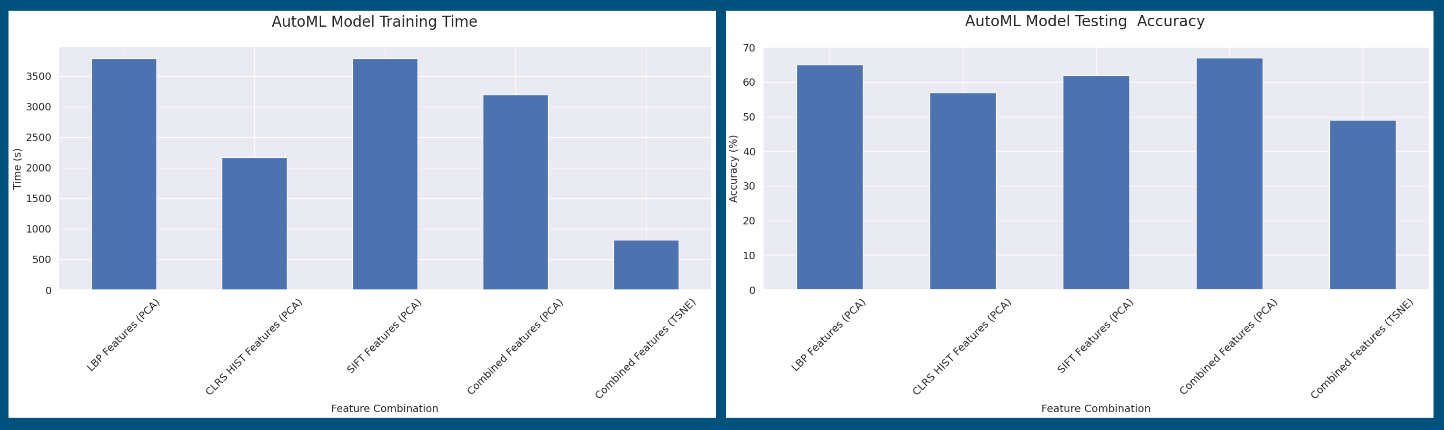


Figure 30: Efficiency vs Accuracy Tradeoff

Comparing this result to the baseline CNN, which took approximately 1000 seconds to train to reach an accuracy of 60-62%, the autoML processes are slower by a factor of 2 with the exception of using t-SNE for dimensionality reduction. This is expected because we hard fixed t-SNE to 2 principal components, which make the training dataset significantly smaller than PCA. However, when comparing test accuracy, even with combining all features, using t-SNE for dimensionality reduction resulted in a significantly lower accuracy score (< 50%) compared to the baseline CNN. This result could be attributed to the non-distinguishable classification labels noted in the earlier section.

This experiment's result pointed out that an autoML using all three features, dimensionally reduced by PCA, yields the best results in terms of accuracy ( > 65%) and has the third best training time ( ~ 3200 seconds ). For that reason, we believe that additional features or an augmented CNN architecture could help improve the training accuracy.

**CNN with LSTM Cell Architecture**

While not necessarily better, we wanted to leverage an augmented CNN architecture described by Vinyals et al. in “Show and tell: A neural image caption generator” [7]. For our approach, we decided to select only SIFT features because we thought the combined features would be more difficult to implement. To reimplement the CNN architecture proposed, we utilized the baseline CNN architecture that we put together up until the hidden layer. We then introduced an LSTM layer that is initialized by the output of the CNN and parameterized by the SIFT feature. Figure 30 outlines the architecture used in this experiment.



Figure 31: Show and Tell CNN architecture

Unfortunately, once trained, this CNN model did not provide sufficient results in terms of accuracy, precision and recall. It is noted that the model did have very high recall for the zeroth category, which suggests that the model is able to return relevant results for normal x-rays patients. However, the model did not perform well in terms of classifying viral related pneumonia, category one. These phenomena could be explained by the differences in terms of “white cloud” effect between normal and pneumonia patients, but the differences of the “white cloud” effect between viral versus bacteria infection were not fully characterized. The model results are reported in Figure 32.

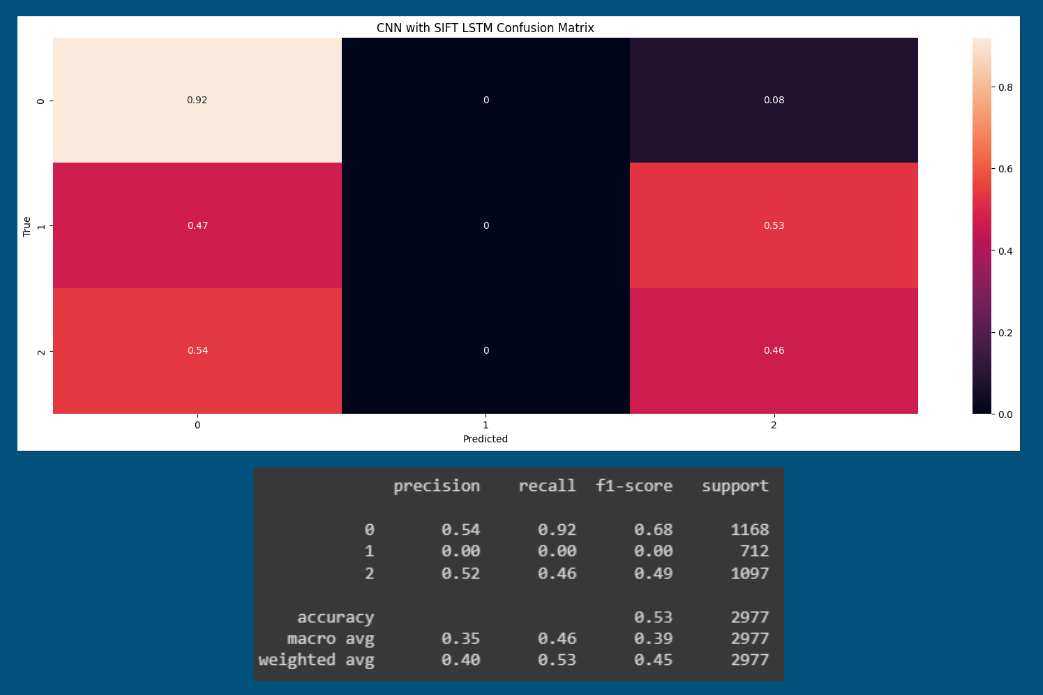


Figure 32: Augmented CNN with LSTM architecture model result

**Conclusion**

In this report, we explored the potential of several features extracted from x-ray images in classification of pneumonia patients. We started the project by looking at a simple CNN architecture trained only on the pixel images as well as fine-tuning a pre-trained ResNet50 architecture. Afterwhich, we extracted three sets of features, LBP, HSV, and SIFT, and used those to train simpler models. From our experimental results, we noticed that using individual features or combined features resulted in a better performance model. In a sense, the extracted features were able to capture a little more information compared to the modern CNN approach. When using an AutoML package, the results show marginal positive effect on some features compared to Gradient Boosting Tree. This result further suggests the need of using Grid Search/Randomized Grid Search on hyperparameters to improve model selection.

Unfortunately, the extracted features were not able to fully capture the “white-cloud” effect on pneumonia patients. While using dimensionality reduction could improve model training time, we noticed that t-SNE does not contribute to any improvement compared to PCA, which could be attributed to the white cloud effect being too difficult to distinguish.

In a separate experiment, where we decided to approach this problem using an image captioning approach. Here, we realized that the usage of bigger, more complex models is not always better. While we could explore further usage of more features as initializers for LSTM, this approach would be pointless and provide diminishing returns. For that, we reasoned that either using a fine-tuning approach on an already trained model could be more efficient. Further investigation on the attention mechanism could also improve the performance of the LSTM approach.

**Citation**

[1] Maguolo, Gianluca, and Loris Nanni. "A critic evaluation of methods for COVID-19 automatic detection from x-ray images." *Information Fusion* 76 (2021): 1-7. <https://doi.org/10.48550/arXiv.2004.12823>

[2] Tartaglione, Enzo, et al. "Unveiling covid-19 from chest x-ray with deep learning: a hurdles race with small data." *International Journal of Environmental Research and Public Health* 17.18 (2020): 6933. <https://doi.org/10.48550/arXiv.2004.05405>

[3]Yamashita, R., Nishio, M., Do, R.K.G. et al. Convolutional neural networks: an overview and application in radiology. Insights Imaging 9, 611–629 (2018). <https://doi.org/10.1007/s13244-018-0639-9>

[4] Dale A. Schumacher, II.1 - IMAGE SMOOTHING AND SHARPENING BY DISCRETE CONVOLUTION, Editor(s): JAMES ARVO, Graphics Gems II, Morgan Kaufmann, 1991, Pages 50-56, ISBN 9780080507545, <https://doi.org/10.1016/B978-0-08-050754-5.50021-9>.

[5] Noble, W. What is a support vector machine?. Nat Biotechnol 24, 1565–1567 (2006). https://doi.org/10.1038/nbt1206-1565

[6] Vamvakas, A., Tsivaka, D., Logothetis, A., Vassiou, K., & Tsougos, I. (2022). Breast cancer classification on multiparametric MRI – increased performance of boosting ensemble methods. *Technology in Cancer Research & Treatment*, *21*, 153303382210878. https://doi.org/10.1177/15330338221087828

[7] Vinyals, O., Toshev, A., Bengio, S. & Erhan, D. (2015). Show and tell: A neural image caption generator. CVPR (p./pp. 3156-3164), : IEEE Computer Society. ISBN: 978-1-4673-6964-0

[8] Patra, Pankaj. “Create Local Binary Pattern of an Image Using Opencv-Python.” GeeksforGeeks, GeeksforGeeks, 3 Jan. 2023, https://www.geeksforgeeks.org/create-local-binary-pattern-of-an-image-using-opencv-python