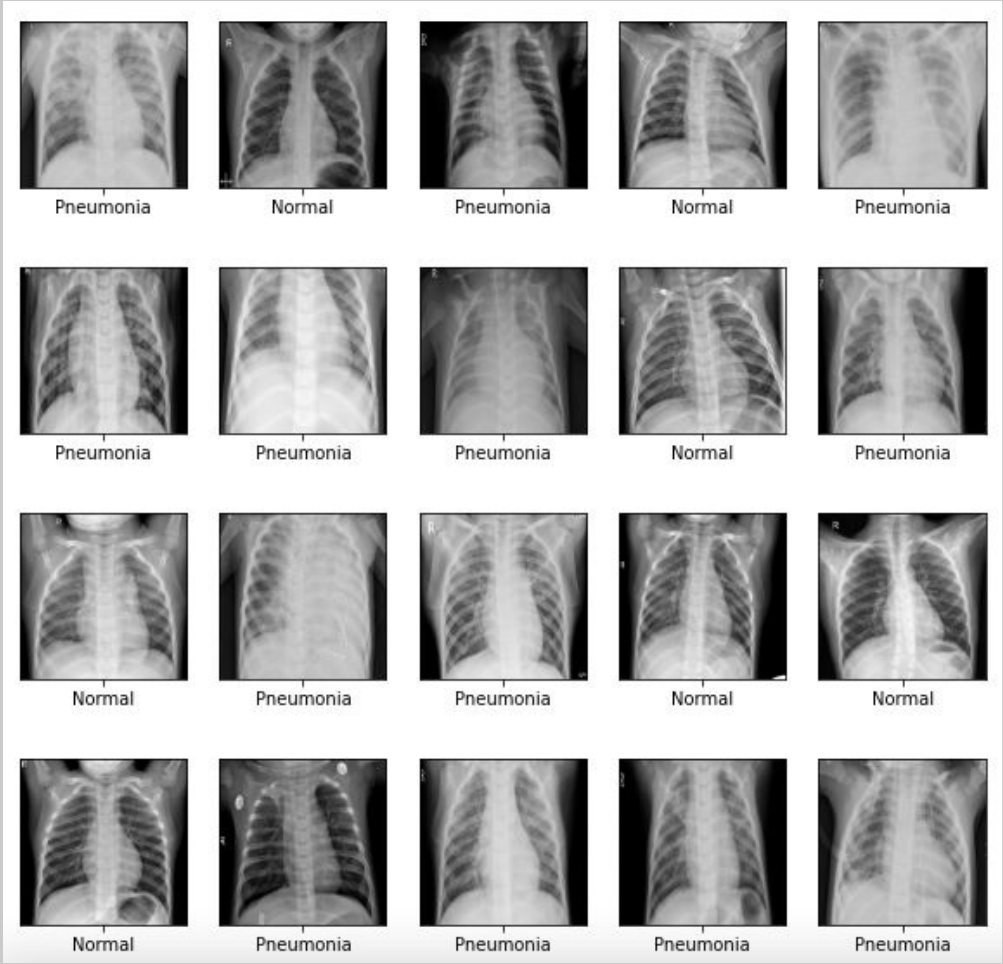
Predicting Pneumonia

Increasing the marketability of GE Healthcare's Critical Care Suite 2.01 (CCS)

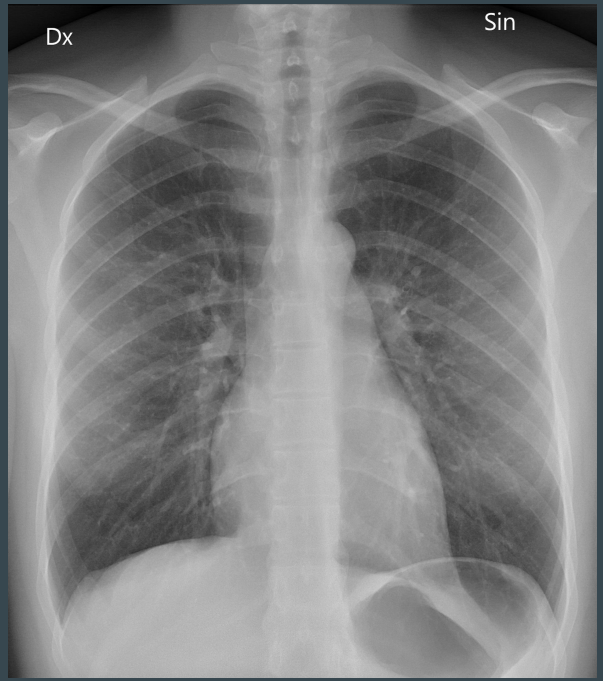


Introduction

The global X-ray detectors market is expected to reach $4.3 billion by 2029

Client Needs

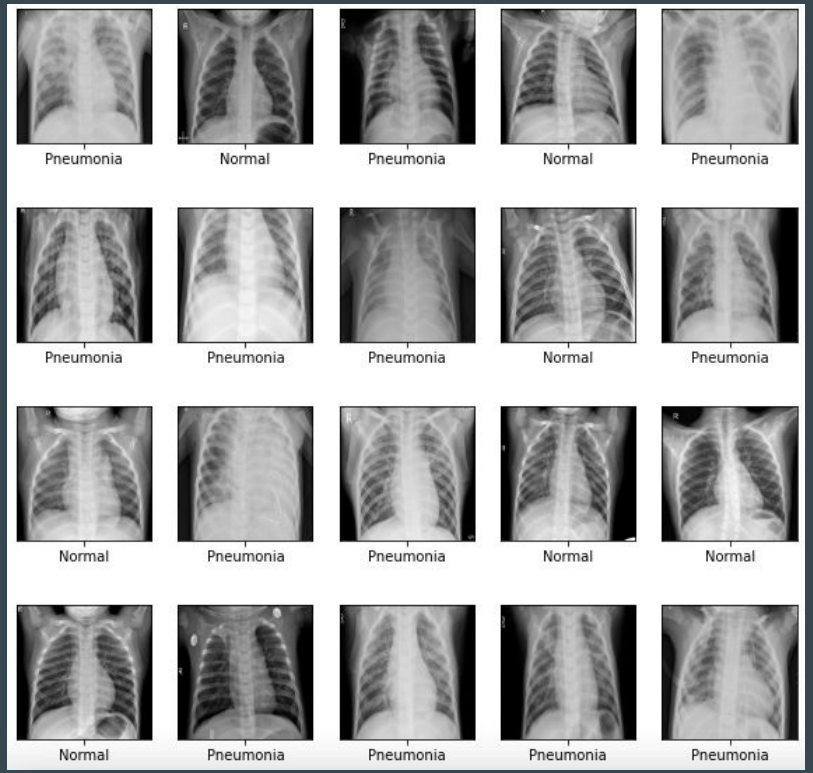
Increase prospective customer base by increasing the application potential of CCS system.



Build and test a model to detect pneumonia using chest x ray images

Image from “GE Healthcare Receives FDA Clearance of First Artificial Intelligence Algorithms Embedded On-Device to Prioritize Critical Chest X-ray Review”

Data Used



There are 5,856 radiographs

- 1,583 NORMAL
- 4,273 PNEUMONIA (both bacterial and viral)

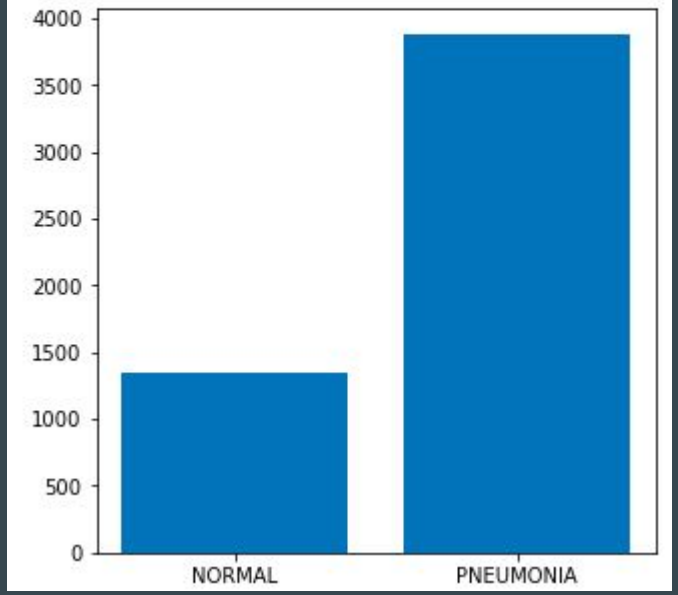
Data available on **Mendeley Data**

Testing data distribution:

- 234 NORMAL
- 390 PNEUMONIA

Distribution of all the training data

	label	count	percentage
0	NORMAL	1349	25.78%
1	PNEUMONIA	3883	74.22%



Preliminary models downsampled to:

- Training set
  - 250 NORMAL
  - 250 PNEUMONIA
- Validation set
  - 25 NORMAL
  - 25 PNEUMONIA

Data Used

in FINAL MODEL

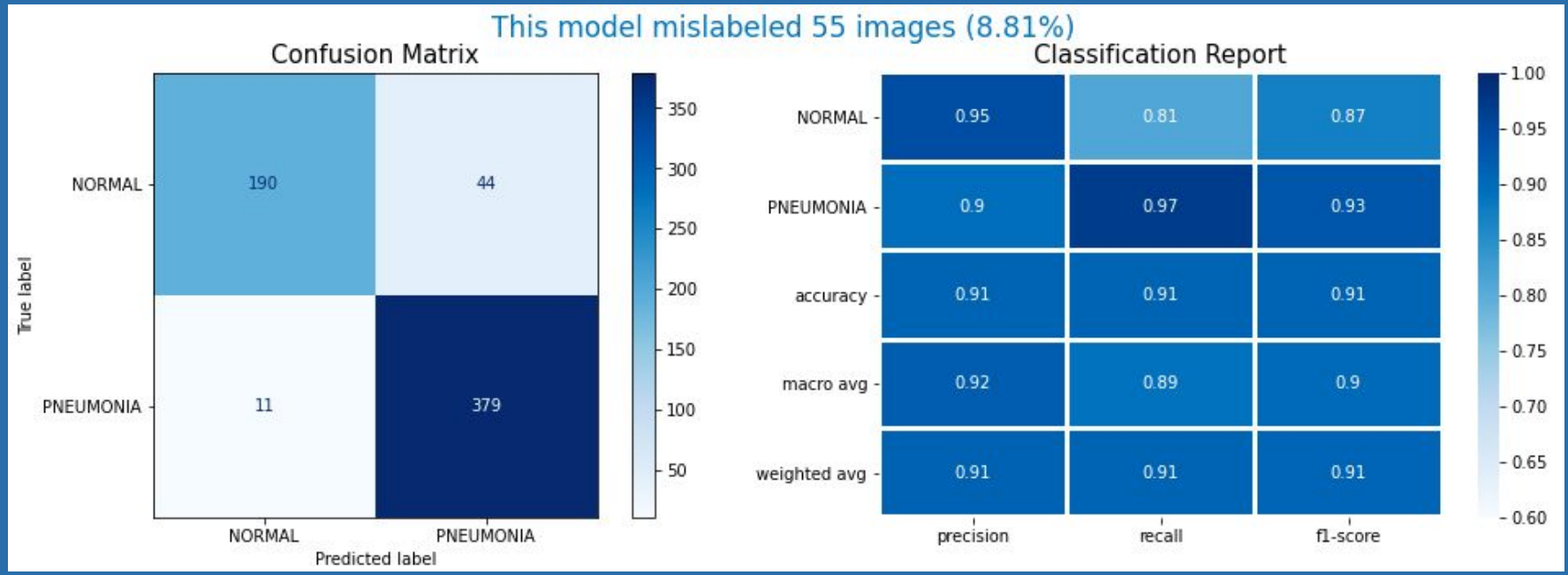
Data used to train Final Model:

- Training set
  - 1324 NORMAL
  - 1324 PNEUMONIA
- Validation set
  - 25 NORMAL
  - 25 PNEUMONIA

Final Model

performance

Final Model performance on unseen test data:

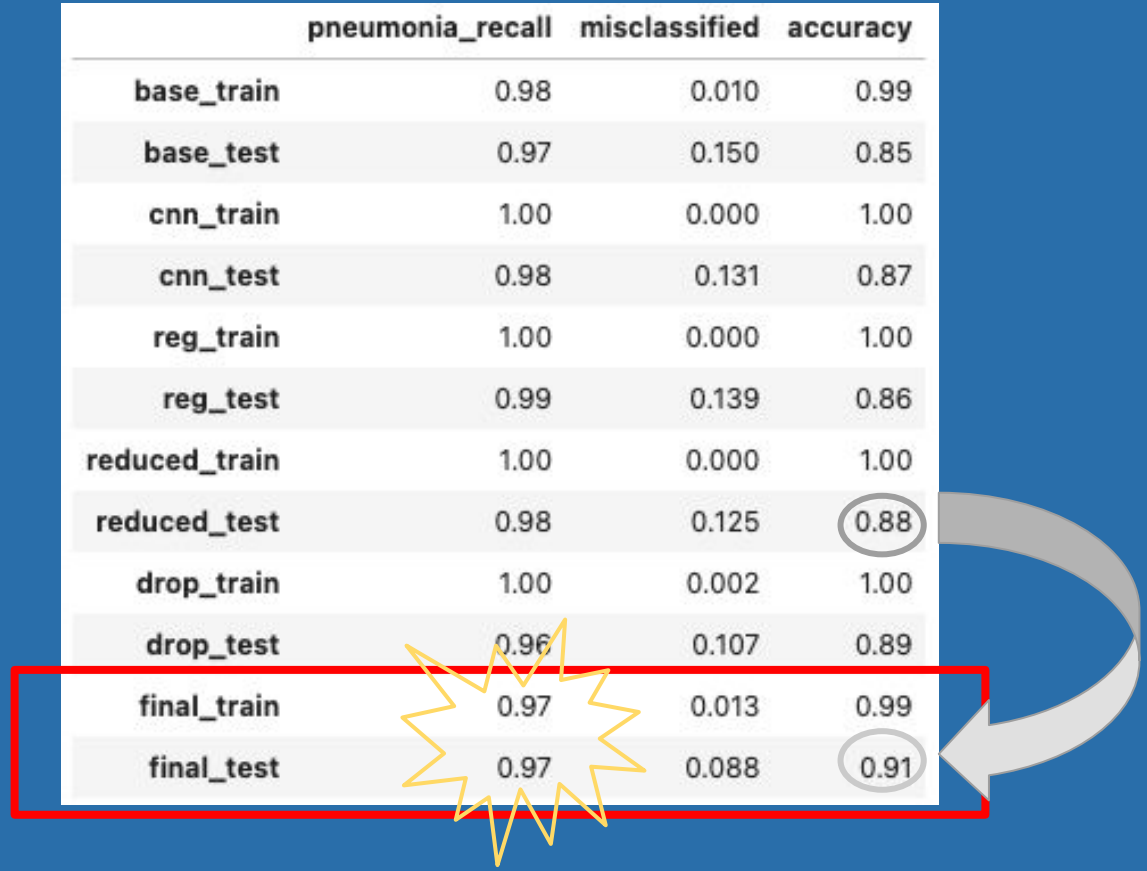


Wrap Up

- Improve clinical outcomes
- Elevate patient experiences
- Increase the marketability of the CCS system.

Thank You

	pneumonia_recall	misclassified	accuracy
base_train	0.98	0.010	0.99
base_test	0.97	0.150	0.85
cnn_train	1.00	0.000	1.00
cnn_test	0.98	0.131	0.87
reg_train	1.00	0.000	1.00
reg_test	0.99	0.139	0.86
reduced_train	1.00	0.000	1.00
reduced_test	0.98	0.125	0.88
drop_train	1.00	0.002	1.00
drop_test	0.96	0.107	0.89
final_train	0.97	0.013	0.99
final_test	0.97	0.088	0.91



More data used to train the model

```
final_model = models.Sequential()
final_model.add(layers.Conv2D(50, (3, 3), padding='same', activation='relu', input_shape=(256, 256, 1)))
final_model.add(layers.MaxPooling2D((2, 2)))

final_model.add(layers.Conv2D(40, (3, 3), padding='same', activation='relu'))
final_model.add(layers.MaxPooling2D((2, 2)))

final_model.add(layers.Conv2D(35, (3, 3), padding='same', activation='relu'))
final_model.add(layers.MaxPooling2D((2, 2)))

final_model.add(layers.Conv2D(32, (3, 3), padding='same', activation='relu'))
final_model.add(layers.MaxPooling2D((2, 2)))

final_model.add(layers.Flatten())

final_model.add(layers.Dense(32, kernel_regularizer=regularizers.l2(.005), activation='relu'))
final_model.add(layers.Dense(16, kernel_regularizer=regularizers.l2(.005),activation='relu'))
final_model.add(layers.Dense(5, kernel_regularizer=regularizers.l2(.005),activation='relu'))
final_model.add(layers.Dense(1, activation='sigmoid'))

final_model.compile(optimizer='SGD',
                    loss='binary_crossentropy',
                    metrics=['accuracy'])

final_model.summary()
```

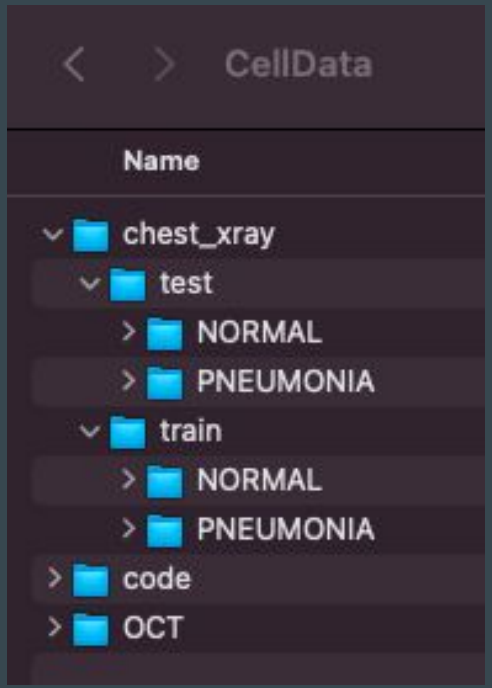
What more data can do for a model

	train_loss	train_acc	test_loss	test_acc	loss_diff	acc_diff
base_model	0.040549	0.988000	0.598990	0.849359	0.558442	0.138641
base_cnn	0.001954	1.000000	0.899908	0.868590	0.897954	0.131410
reg_cnn	1.012247	1.000000	1.839380	0.862179	0.827133	0.137821
reduced_nodes	0.321644	1.000000	1.059986	0.875000	0.738342	0.125000
dropout	0.348353	0.998000	0.967050	0.892628	0.618697	0.105372
final_model	0.113301	0.986782	0.449658	0.911859	0.336356	0.074924

Train

Test

Differences between train and test


CellData
Name
chest_xray
test
NORMAL
PNEUMONIA
train
NORMAL
PNEUMONIA
code
OCT

Recommendations

- Include a pneumonia recognition model on CCS systems
- Radiologist confirm or reject predictions
- Push software updates

Next Steps

- Increase lambda coefficient in l2 regularization
- Reduce nodes further
- Remove a layer
- Decrease dropout p value
- Instead of downsampling Pneumonia, upsample Normal