Chest X-Ray Report Generation from Chest-X Ray Images

Stanford CS224N Custom Project

Esin Darici Haritaoglu
dhesin@stanford.edu

Aleksandr Timashov
timashov@stanford.edu

Matthew Tan
matthtan@stanford.edu

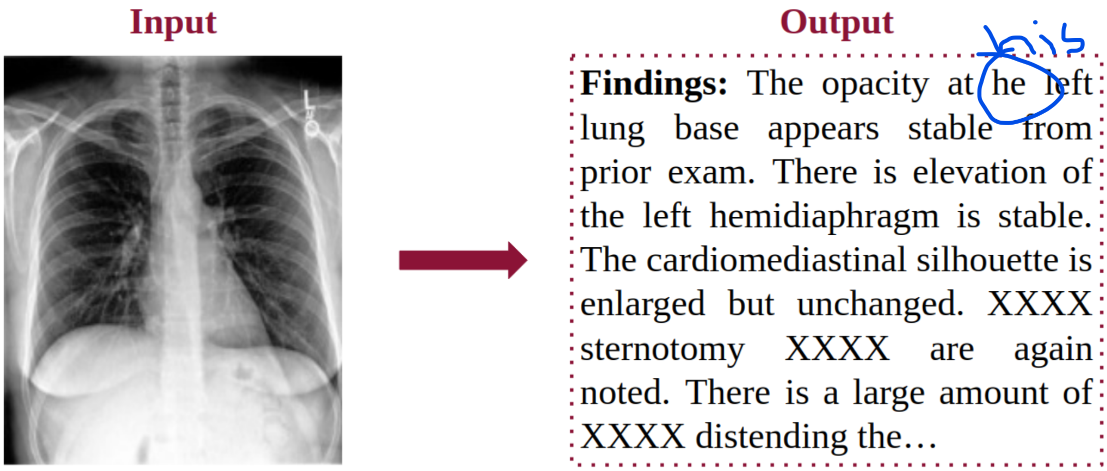
TA mentor: Kathy Yu
fyu9@stanford.edu

Abstract

The automatic generation of highly clinically accurate radiology reports could improve clinical outcomes by reducing radiologist workload, prioritizing severe cases, and augmenting existing radiograph processing pipelines.
In this project, we are going to focus on an encoder-decoder models for a generative approach, and retrieval model approach based on existing radiology reports.

1 Introduction

There has been a recent interest and developments in the automatic generation of radiology reports by analyzing chest X-ray images. Such a technology would reduce the workload of radiologist, streamline the clinical process and speed up the necessary medical intervention preventing serious illnesses or undesired outcomes. AI technologies has already shown a promising results in detection and labeling of several diseases from X-ray images such as CheXbert, CheXpert Labeler. Natural next step would be to generate radiology reports directly by analyzing these chest X-ray images.



Findings: The opacity at he left lung base appears stable from prior exam. There is elevation of the left hemidiaphragm is stable. The cardiomediastinal silhouette is enlarged but unchanged. XXXX sternotomy XXXX are again noted. There is a large amount of XXXX distending the…

Figure 1: Example of Report Generation

Although NLP technologies has also advanced dramatically and generate very convincing, realistic documents from given prompts, medical nature of the problem requires special attention while designing the solutions. First and foremost concern is the clinical efficacy of generated reports. Usual metrics used to evaluate NLP systems are not sufficient and maybe even misleading to evaluate the performance of automatic report generation system since subtle changes in the language may have a profound results in the medical outcome. because small changes in the language have other meanings & this is critical

There are two more significant differences of diagnostic image captioning compare with generic image captioning:

- Commonly, reports from diagnostic image captioning are significantly longer than results of generic image captioning.
- Generic image captioning don't have any time series information, but diagnostic reports often refers to the long history of research. So it can be used in report generation.

2 Related Work

2.1 Approaches

Many methods have been proposed to to solve the task of medical report generation. There are three the most common approaches for automatic chest x-ray report generation:

- Template-based approach, where we classify whether certain conditions are present or not, and then retrieve the corresponding template sentence for each condition. Simply
- Retrieval-based approach, where we reuse text from previous similar examples.
- Generation-based approach, where we use an image-encoder + text-decoder architecture that encodes the image and then decodes that into a sequence of words.

Usually methods of medical report generation employs generation - based approach which is represented by the model with encoder - decoder architecture. [1] and [2] are the two most relevant to our work in automatic x-ray report generation. Both work uses image encoder text decoder architecture as the backbone with some differences in the pretrained image encoder and the implementation of the transformer decoder. Most notable contribution comes from the used loss functions in these works. [2] uses contrastive loss between K-means clustered documents according to their embedding space representation from ChexBert labeler. [1] uses exact entity match reward for named entities as well as factual consistency reward with NLI function.

Despite on their simplicity, retrieval - based approaches, where we reuse text from previous exams based on some similarity metrics, show competitive results and even outperform more complex generation-based approaches. It might be because medical reports are too restrictive due to limited number of potential findings and use template - based language. At the same time retrieval - based method can assist in extracting the actual knowledge presented on chest x-ray image. Hence, reports with the same findings might look similar or even the same.

2.2 Datasets

There are five well-known datasets that can be used for Chest X-Ray report generation training and evaluation: IU X-RAY, PEIR GROSS, ICLEF CAPTION, MIMIC-CXR, and CheXpert.

Dataset	Train Instances	Test and Validate Instances	Total
IU X-RAY	6,674	756	7,430
PEIR GROSS	6,698	745	7,443
ICLEF CAPTION	200,074	22,231	232,305
MIMIC-CXR	368,960	8,150	377,110
CheXpert	224,316	500	224,816

Due to the different shortcomings, small size in PEIR GROSS, and ICLEF CAPTION, extremely big size of MIMIC-CXR, and difficulty in getting access to it, we are going to focus on IU X-RAY and CheXpert datasets in our project.

3 Approach

3.1 Template-based approach

The main goal of this project is to explore NLP-related research. Template-based approach seemed the most restrictive and did not look having much space for research, so we didn't work on it. However, the multi label classification model that is used in this approach can be applied in generative-based approach to improve results. It can be done by concatenating classes probabilities results of classification model with encoder outputs of the generated-based model. The code can be found in github. [3]
To train this classification model we used EfficientNet-B5, where replaced last layer with Linear layer that output 14 classes. Initial model was pretrained on ImageNet, last layer was initialized with Xavier initialization. Since we are approaching multilabel classification, as a loss function we are using binary cross-entropy for sigmoid of each class and finally average it out.

3.2 Retrieval-based approach

Retrieval-based approaches are less restrictive than template - based but they are much more simple than generative - based ones. That is why as a baseline, we decided to use this approach as described in [4] using the *Mean@k-NN* method. This method uses the image encoding to retrieve text tags from a dataset of human generated report tags to automatically generate a report for the x-ray image.

For the retrieval method, a dense network model is used to extract embeddings from x-ray images. A tuned parameter k is used to retrieve the k most similar images by cosine similarity of image embeddings. The model returns i of the common tags between the k imaegs where i is the average number of tags among the k most similar images.

3.3 Generation-based approach

Our work also follow from the that of [2] and [1] with some differences in the image encoder, text decoder architecture as well as loss function. We use [5] UNet pretrained on brain MRI images as an image encoder. Although MRI and X-Ray images are different, we believe model pretrained on medical images are more appropriate as an image encoder compared to ImageNet pretrained models. Standard transformer encoder/decoder layers follow the pretrained UNet image encoder. Finally we use Cross Entropy Loss at the output of the text decoder.

UNet encoder takes an 3 channel input image of size (256,256) and output is a one-channel probability map of abnormality regions with the same size as the input image. Pretrained model is part of PyTorch hub and more details are available at [5]. The 256x256 UNet output is projected on to embedding space of size 760 with a sequence length of 256 and position embedding is added to the projected output. Position embedded added representation is used as an input to the 1 layer transformer encoder followed by 2 layer transformer decoder. Both encoder and decoder has 36 attention heads, Feed Forward layer of size 2048 and gelu activation. Decoder output is projected with Linear layer from embedding size to vocabulary size. During training, ground truth report is tokenized and tokenized sequence is projected into embedding space of 760 again. Cross entropy loss is calculated at the output of the decoder by shifting the ground truth report by one word.

To regulate the training and prevent overfitting, we used dropout rate of 0.3 after position embedding, within the both encoder and decoder layers and after target sequence embedding.

During inference, we started with the <start> token to indicate the start of the sentence. Then we concatenated highest probability word from the decoder output to the previous word to continue sentence generation until the <eos> token is generated. We didn't implement beam search.

As a starter code, we used parts of trainer.py and utils.py from the minGPT [6] repository. model.py which includes the model has been developed from scratch. Use of UNet is a novel approach to the best of our knowledge that has not been tried before as well as adding the labels to the image encoder representations. Our code base could be reached at [7].

CNN

As shown in Figure 2, we also experimented with concatenating labels from chest X-Ray images with the embeddings from UNet and observed modest improvement in the BLEU results. We used pretrained DenseNet from TorchXRayVision package [8] for chest X-Ray label extraction. We also

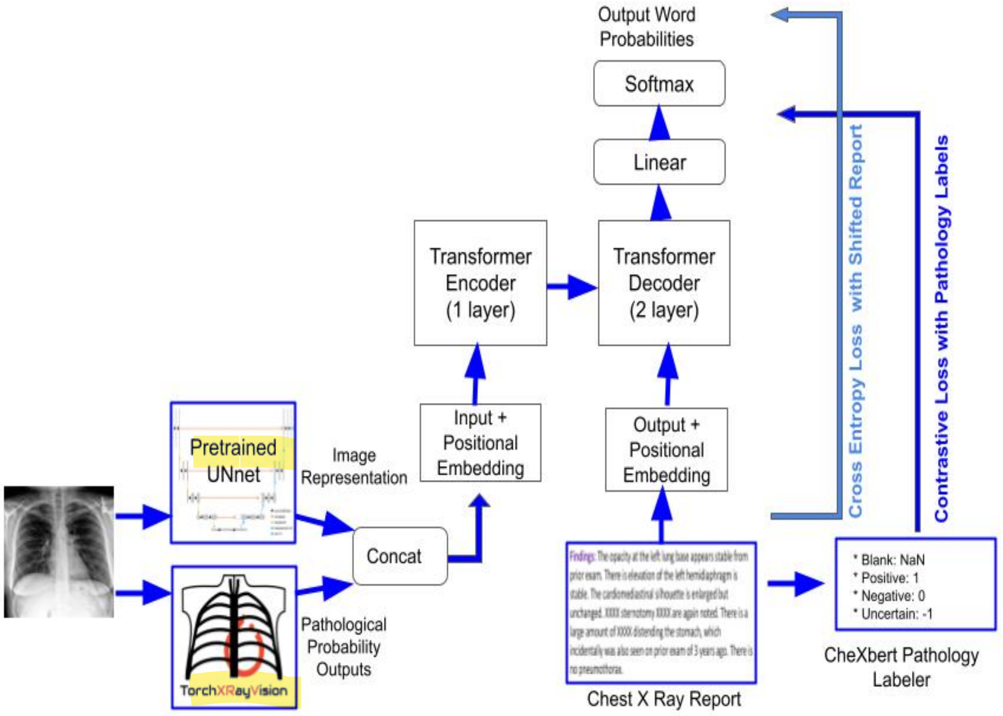

Output Word Probabilities
Softmax
Linear
Transformer Encoder (1 layer)
Transformer Decoder (2 layer)
Cross Entropy Loss with Shifted Report
Contrastive Loss with Pathology Labels
Pretrained UNnet
Image Representation
Input + Positional Embedding
Output + Positional Embedding
Concat
TorchXRayVision
Pathological Probability Outputs
Findings: The opacity at the left lung base appears stable from prior exam. There is elevation of the left hemidiaphragm is stable. The cardiomediastinal silhouette is enlarged but unchanged. XXXX sternotomy XXXX are again noted. There is a large amount of XXXX distending the stomach, which incidentally was also seen on prior exam of 3 years ago. There is no pneumothorax.
Chest X Ray Report
* Blank: NaN
* Positive: 1
* Negative: 0
* Uncertain: -1
CheXbert Pathology Labeler

Figure 2: Generation Based Image Encoder/ Text Decoder

experimented with Contrastive Loss by using the labels from CheXbert Pathology Labeler to contrast the embeddings at the transformer decoder output.

4 Experiments

4.1 Data

We used IU X-Ray dataset that includes frontal and lateral images alongside with their reports. We specifically used frontal images in our network and findings sections from the reports as a ground truth report. After filtering out lateral images, images with empty reports and duplicate reports, we got a dataset size of 1952 samples for training, and 488 samples for validation. We created a vocabulary from the findings section of the reports and had a vocabulary size of 1947 with cased words. We experimented with uncased vocabulary of size 1591 too. We expected that results would improve with smaller vocabulary size but results indicated the other way. We reason that cased vocabulary helps in decoding process by indicating end/beginning of sentences which helps better decoding. We also experimented with WordPieceTokenizer to handle variations of the same word and expected better results. The results indicated the otherwise.

4.2 Evaluation method

We used clinical efficacy metric derived from correctly extracted labels from generated documents by CheXbert labeler. Additionally, we used BLEU scores to measure the similarity of the generated text to the original one. For retrieval tag generation, clinical efficacy was determined by the percent number of images with all of the correct tags retrieved and no extraneous tags out of the total number of tested. Tag generation accuracy was determined by the percent similarity of tags between the generated and ground truth.

4.3 Experimental details

The classification model, trained on Chexpert dataset, demonstrate the following accuracy on validation part of the dataset:

Class	Accuracy	Class	Accuracy
No Finding	73.93%	Pneumonia	96.16%
Enlarged Cardiomediastinum	53.42%	Atelectasis	61.54%
Cardiomegaly	70.51%	Pneumothorax	95.23%
Lung Opacity	49.15%	Pleural Effusion	76.50%
Lung Lesion	99.15%	Pleural Other	99.57%
Edema	80.77%	Fracture	100%
Consolidation	85.90%	Support Devices	79.91%

As we can see, some classes are predicted almost perfectly, when a few others are not much better than choice. It can be improved by introducing weighted binary cross-entropy loss where larger weights are introduces to classes that harder to predict.

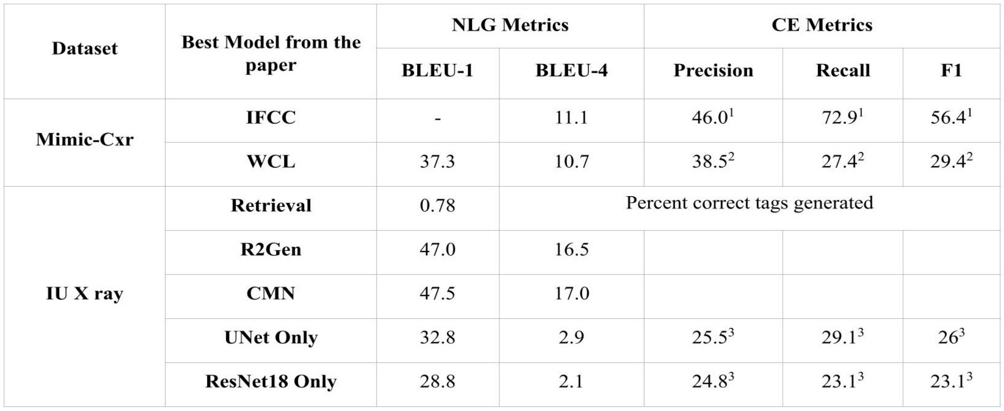
The retrieval baseline method image embedding model was trained on the IU X-Ray dataset as described. The parameter k was then trained on a 40% split of the data to maximize tag generation accuracy. The remaining 60% of the images were used to test the performance of the tag generation.

Most of our experiments included basic image encoder+transformer encoder/decoder with cross entropy loss. We experimented with ResNet18 in addition to standard UNet configuration as an image encoder. We also experimented with concatenating image encoder output with labels from CheXpert labeler. In the real world scenario, CheXpert labeler output will not be available but TorchXRayVision probability outputs will be available. We tried to integrate TorchXRayVision models for more realistic scenario but there were PyTorch version issues.

During training we used batch size of 16 and learning rate of 1e-3 with a cosine learning rate decay. We ran training until after the test loss has gone above the 5% of the minimum test loss. Test loss generally reached to its lowest around epoch 4/5 and continued to rise above 5% until epoch 8/9. Figure 4 in Appendix A shows the training and test losses over several epochs. Each epoch took approximately 31 seconds with data running parallel on 2 RTX 3090.

4.4 Results

We have summarized ours and most recent/relevant papers results in Figure 3. The Retrieval baseline model *Mean@k-NN* scored **.78% tag generation accuracy and 0% clinical efficacy.** In Poster session, our numbers were coming from weighted average results and were very high. This is due to unbalanced nature of data with very high numbers of class 2 (no mention). This time we are reporting average of macro average results for each label. Detailed results for each label can be found in Appendix A. In Appendix A, we also reported results from UNet and ResNet18 when their outputs are concatenated with CheXbpert Labeler outputs. We couldn't finalize the integration of TorchXRayVision labeler in time for more realistic inference scenario. But we believe that concatenated results from CheXbert Labeler also points to the benefit of using labels as a conditioner for transformer decoder.



Dataset	Best Model from the paper	NLG Metrics		CE Metrics		
		BLEU-1	BLEU-4	Precision	Recall	F1
Mimic-Cxr	IFCC	-	11.1	46.0[1]	72.9[1]	56.4[1]
	WCL	37.3	10.7	38.5[2]	27.4[2]	29.4[2]
IU X ray	Retrieval	0.78	Percent correct tags generated			
	R2Gen	47.0	16.5			
	CMN	47.5	17.0			
	UNet Only	32.8	2.9	25.5[3]	29.1[3]	26[3]
	ResNet18 Only	28.8	2.1	24.8[3]	23.1[3]	23.1[3]

Figure 3: Comparision of different models and ours. 1)The micro average of accuracy, precision, recall, and F1 scores are calculated over 5 observations for: atelectasis, cardiomegaly, consolidation, edema, and pleural effusion 2)It is not explicitly stated but we concluded that WCL results are macro average over all 14 observations 3)Macro average results of our models over 14 observations.

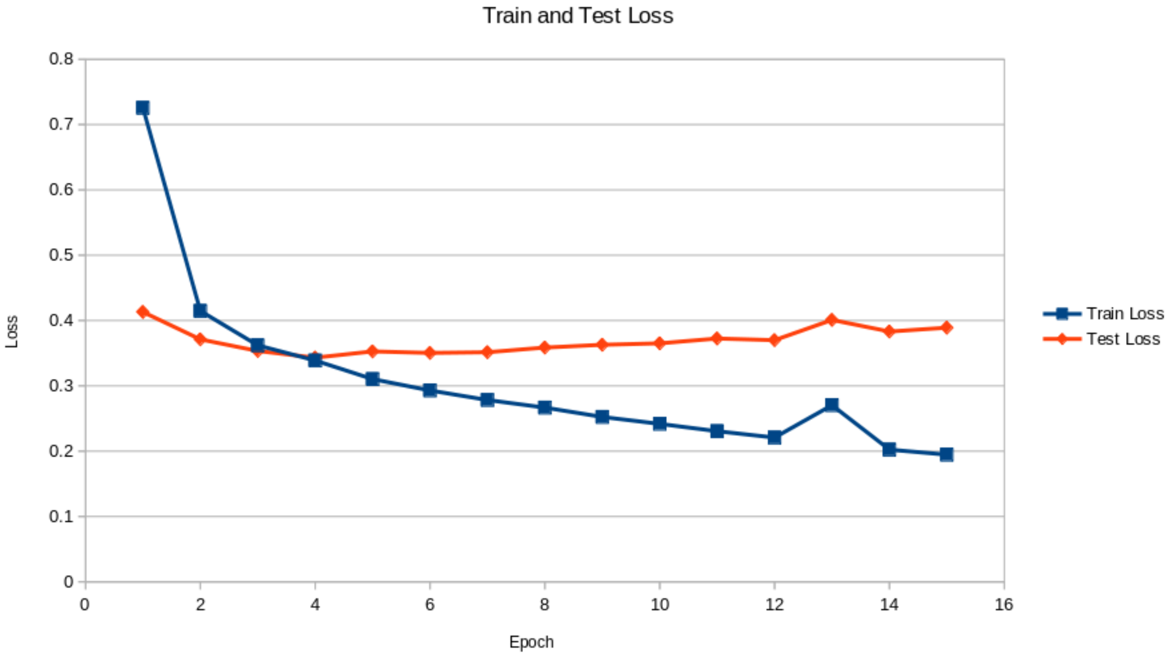


Figure 4: Training and Testing loss (averaged out over entire epoch)

5 Analysis

The retrieval baseline performed poorly in both metrics particularly in the “normal” tag generation (0 tags generated) case. Without using a specially trained tag generation model such as CheXpert labeler, the clinical accuracy based on image encoding and naive tag retrieval alone is poor. The model often attempts to retrieve incorrect tags of low probability to hit the required mean number of tags per image. This is particularly common as the k parameter was found to be 5 therefore 5 images, even those with low cosine similarity, are retrieved and skew the mean parameter of tags to retrieve. This leads to the low tag generation accuracy and an experimentally found 0% clinical efficacy.

Generation based results are very promising especially when looked into the other literature results. Our results are also comparable considering that only small set of IU X Ray frontal images and Cross Entropy Loss is used. But there are some problems stemming from the unbalanced nature of the dataset. We were only able to use IU XRay but unbalanced nature has been reported on other datasets too. This makes the results rather unbalanced too resulting in high deviation in performance for each pathology type. Therefore another line of research would be to experiment with different augmentation techniques to make the datasets more balanced. Smaller number of pathology cases could be upsampled by augmenting their x-ray images.

We explored different types of image encoders; ResNet18 and UNet. Experiment results show that image encoders specifically trained on medical images perform better. There are some chest x ray image pretrained models in the [8] package but we were unable to use those models due to PyTorch compatibility issues. UNet is the closest one to the medical images which are trained on brain MRIs. ResNet18 is trained on the ImageNet. If one can uses chest x ray pretrained image encoder, we expect much better results.

We also explored the use of auxiliary input on the transformer decoder in addition to image encoder output. Specifically we used pathology labels from CheXpert labeler. We were planning to integrate TorchXRayVision labeler but encountered problems. Including pathology labels along with image encoder outputs at the decoder side improved results in most of the pathology types. This indicates that multiple models trained on the chest xray images for different tasks may improve results by capturing more information about the chestxray images. E.g. UNet trained for image segmentation and CheXpert Labeler for pathology types.

6 Conclusion

6.1 Summary of results and significance of the work

Retrieval is useful for a baseline methodology as it is quick to train and produce grammatically and syntactically correct results, however struggles to improve in clinical accuracy without augmentation from another classification input such as CheXpert. Generation based approach offers many avenues of research, more specifically including different types of loss functions and joint optimization of them like used in [1], different pretrained image encoders, using auxiliary input like pathology labels for transformer encoder or image encoders pretrained for different tasks; e.g. segmentation, label classification and/or autoencoder.

6.2 Ideas for future work

To improve this work and get high-accuracy generated medical reports, we should focus on the following directions:

- Modify loss function for the trained multi class multi label classifier and use it as an additional input to encoder-decoder model.
- Experiment with lateral images from IU X-ray dataset.
- Use MIMIC-CXR dataset from MIT to training and validate our models.
- Experiment with different model architectures and different training techniques.

6.3 Code

- `https://github.com/dhesin/xray-report-gen` (Main generation-based approach)
- `https://github.com/atimashov/cxr-report-generation` (classification for findings and retrieval-based approach)

References

[1] Emily Bao Tsai Curtis P. Langlotz Dan Jurafsk Yasuhide Miura, Yuhao Zhang. Improving factual completeness and consistency of image-to-text radiology report generation. In *https://arxiv.org/pdf/2010.10042.pdf*, 2021.

[2] Xing Lu Jiang Du Eric Chang Amilcare Gentili Julian McAuley Chun-Nan Hsu An Yan, Zexue He. Weakly supervised contrastive learning for chest x-ray report generation. In *https://arxiv.org/pdf/2109.12242.pdf*, 2021.

[3] cxr-report-generation. In *https://github.com/atimashov/cxr-report-generation*.

[4] Vasiliki Kougia, John Pavlopoulos, and Ion Androutsopoulos. Medical image tagging by deep learning and retrieval. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 154–166. Springer, 2020.

[5] mateuszbuda. U-net for brain mri. In *https://pytorch.org/hub/mateuszbuda*$_{b}rain - segmentation - pytorch_{u}net/$.

[6] Karphaty. mingptl. In *https://github.com/karpathy/minGPT*.

[7] xray-report-gen. In *https://github.com/dhesin/xray-report-gen*.

[8] Torchxrayvision. In *https://github.com/mlmed/torchxrayvision*.

A Appendix

	UNet Only	UNet+CheXpertLabels	ResNet-18 Only	ResNet18+CheXpertLabels
BLEU-1	0.328	0.315	0.288	0.317
BLEU-2	0.113	0.133	0.087	0.127
BLEU-3	0.053	0.075	0.036	0.066
BLEU-4	0.029	0.047	0.021	0.037
BLEU-5	0.014	0.032	0.013	0.022

Class Labels: -1: Uncertain, 0: Negative, 1: Positive, 2:No Mention

	UNet Only				UNet+CheXpertLabels
No Finding	precision	recall	f1-score	support	f1-score
1	0.22	0.29	0.25	133	0.49
2	0.70	0.63	0.66	354	0.69
accuracy			0.53	487	0.61
macro avg	0.46	0.46	0.46	487	0.59
weighted avg	0.57	0.53	0.55	487	0.63

	UNet Only				UNet+CheXpertLabels
Enlarged Cardiomediastinum	precision	recall	f1-score	support	f1-score
-1	0.00	0.00	0.00	41	0.45
0	0.53	0.99	0.69	260	0.75
1	0.00	0.00	0.00	24	0.06
2	0.00	0.00	0.00	162	0.65
accuracy			0.53	487	0.66
macro avg	0.13	0.25	0.17	487	0.48
weighted avg	0.28	0.53	0.37	487	0.66

	UNet Only				UNet+CheXpertLabels
Cardiomegaly	precision	recall	f1-score	support	f1-score
-1	0.00	0.00	0.00	6	0.00
0	0.00	0.00	0.00	270	0.88
1	0.00	0.00	0.00	63	0.72
2	0.30	1.00	0.47	148	0.87
accuracy			0.30	487	0.85
macro avg	0.08	0.25	0.12	487	0.62
weighted avg	0.09	0.30	0.14	487	0.84

	UNet Only				UNet+CheXpertLabels
Lung Lesion	precision	recall	f1-score	support	f1-score
-1	0.00	0.00	0.00	2	0.00
0	0.00	0.00	0.00	20	0.00
1	0.00	0.00	0.00	24	0.20
2	0.91	1.00	0.95	441	0.94
accuracy			0.91	487	0.89
macro avg	0.23	0.25	0.24	487	0.29
weighted avg	0.82	0.91	0.86	487	0.86

	UNet Only				UNet+CheXpertLabels
Lung Opacity	precision	recall	f1-score	support	f1-score
-1	0.00	0.00	0.00	12	0.00
0	0.14	0.01	0.01	130	0.31
1	0.18	0.35	0.24	93	0.38
2	0.51	0.59	0.54	252	0.75
accuracy			0.38	487	0.60
macro avg	0.21	0.24	0.20	487	0.36
weighted avg	0.33	0.38	0.33	487	0.55

	UNet Only				UNet+CheXpertLabels
Edema	precision	recall	f1-score	support	f1-score
-1	0.00	0.00	0.00	4	0.00
0	0.00	0.00	0.00	24	0.40
1	0.00	0.00	0.00	3	0.00
2	0.94	1.00	0.97	456	0.97
accuracy			0.94	487	0.95
macro avg	0.23	0.25	0.24	487	0.34
weighted avg	0.88	0.94	0.91	487	0.93

	UNet Only				UNet+CheXpertLabels
Consolidation	precision	recall	f1-score	support	f1-score
0	0.33	0.01	0.01	174	0.56
1	0.00	0.00	0.00	4	0.00
2	0.63	0.91	0.75	309	0.87
accuracy			0.58	487	0.77
macro avg	0.32	0.31	0.25	487	0.48
weighted avg	0.52	0.58	0.48	487	0.75

	UNet Only				UNet+CheXpertLabels
Pneumonia	precision	recall	f1-score	support	f1-score
0	0.00	0.00	0.00	12	0.59
1	0.00	0.00	0.00	2	0.00
2	0.97	1.00	0.99	473	0.99
accuracy			0.97	487	0.98
macro avg	0.32	0.33	0.33	487	0.53
weighted avg	0.94	0.97	0.96	487	0.98

	UNet Only				UNet+CheXpertLabels
Atelectasis	precision	recall	f1-score	support	f1-score
-1	0.00	0.00	0.00	15	0.00
1	0.00	0.00	0.00	18	0.00
2	0.93	1.00	0.96	454	0.96
accuracy			0.93	487	0.93
macro avg	0.31	0.33	0.32	487	0.32
weighted avg	0.87	0.93	0.90	487	0.90

	UNet Only				UNet+CheXpertLabels
Pneumothorax	precision	recall	f1-score	support	f1-score
-1	0.00	0.00	0.00	1	0.00
0	0.81	0.99	0.89	394	0.86
1	0.00	0.00	0.00	6	0.08
2	0.00	0.00	0.00	86	0.63
accuracy			0.80	487	0.79
macro avg	0.20	0.25	0.22	487	0.39
weighted avg	0.65	0.80	0.72	487	0.81

	UNet Only				UNet+CheXpertLabels
Pleural Effusion	precision	recall	f1-score	support	f1-score
-1	0.00	0.00	0.00	8	0.00
0	0.79	0.99	0.88	387	0.87
1	0.00	0.00	0.00	9	0.13
2	0.00	0.00	0.00	83	0.55
accuracy			0.79	487	0.78
macro avg	0.20	0.25	0.22	487	0.39
weighted avg	0.63	0.79	0.70	487	0.79

	UNet Only				UNet+CheXpertLabels
Pleural Other	precision	recall	f1-score	support	f1-score
-1	0.00	0.00	0.00	2	0.00
1	0.00	0.00	0.00	6	0.00
2	0.98	1.00	0.99	479	0.99
accuracy			0.98	487	0.98
macro avg	0.33	0.33	0.33	487	0.33
weighted avg	0.97	0.98	0.98	487	0.98

	UNet Only				UNet+CheXpertLabels
Fracture	precision	recall	f1-score	support	f1-score
-1	0.00	0.00	0.00	2	0.00
0	0.00	0.00	0.00	6	0.00
1	0.00	0.00	0.00	22	0.00
2	0.94	1.00	0.97	457	0.97
accuracy			0.94	487	0.94
macro avg	0.23	0.25	0.24	487	0.24
weighted avg	0.88	0.94	0.91	487	0.91

	UNet Only				UNet+CheXpertLabels
Support Devices	precision	recall	f1-score	support	f1-score
0	0.00	0.00	0.00	5	0.00
1	0.00	0.00	0.00	24	0.00
2	0.94	1.00	0.97	458	0.97
accuracy			0.94	487	0.94
macro avg	0.31	0.33	0.32	487	0.32
weighted avg	0.88	0.94	0.91	487	0.91