Introduction

- This presentation demonstrates using GradCAM for
    - captions
    - regression tasks

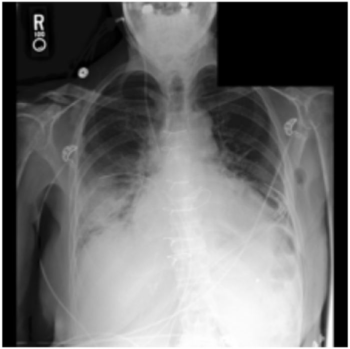
MedCLIP Model Initialization

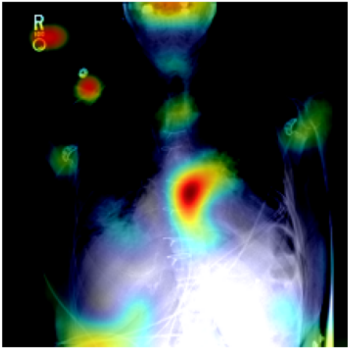
```
model = MedCLIPModel(vision_cls=MedCLIPVisionModelViT)
vision_encoder = model.vision_model
```

HeatMapModel Definition

```
class HeatMapModel(nn.Module):
    def __init__(self, vision_encoder, regression_model):
        super().__init__()
        self.vision_encoder = vision_encoder
        self.regression_model = regression_model

    def forward(self, pixel_values):

        embeddings = self.vision_encoder(pixel_values)

        regression = self.regression_model(embeddings)

        return regression
```

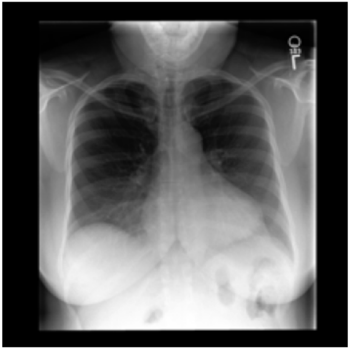
Caption Results: Heart Location



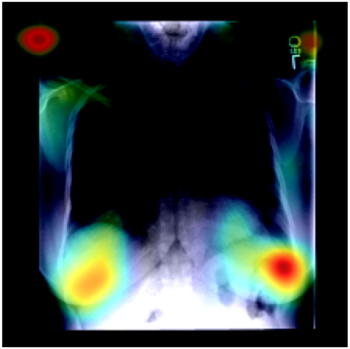


there is some pneumonia near the heart

Caption Results: Lung Location



there is pleural effusion on the lower lung



Phenotyping

Given the image embedding for the same patient **i**, we can find the top *k* phenotypes by computing the inner product (a matrix multiplication).

$$\begin{bmatrix} — & w_1 & — \\ & \vdots & \\ — & w_k & — \end{bmatrix} \begin{bmatrix} | \\ i \\ | \end{bmatrix} = \begin{bmatrix} s_1 \\ \vdots \\ s_k \end{bmatrix}$$

- We find the top *k* similarities and extract those.
- In our example "There is Pneumonia on the left" if we return for k $=$ 2, (2, 5) → (Pneumonia, left) would be our phenotypes

Matching Embedding Shapes

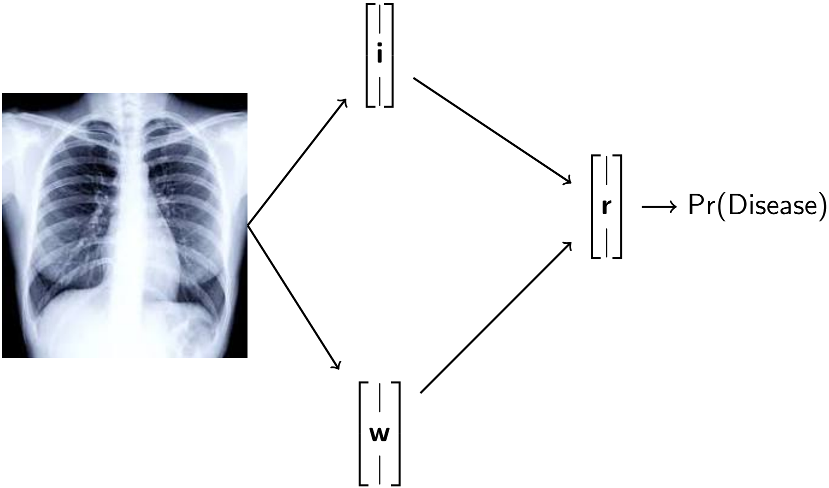
- There is one image with one image embedding $512 vector$
- If we select $k = 5$ phenotypes, then the concatenated text embedding will be of size $512 * 5 = 2560$
- Thus for the regression models to be of the same architecture, I create a single Embedding by using the form: “The embeddings for the x-ray are: {list_of_embeddings}”

Residual Regression 1

We have image embedding $\mathbf{i}$ and text embedding $\mathbf{w}$ where $\mathbf{w}$ is the embedding: "The phenotype are: {List of phenotype}"
Residual embedding $\mathbf{r} = \mathbf{i} - \mathbf{w}$
Train a regression $f : \mathbb{R}^{512} \to [0,1]$

$$f(\mathbf{r}) = \Pr(\text{diease is present})$$

Residual Regression 1: Full Model



$\mathbf{i}$

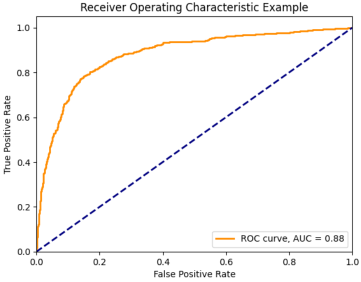
$\mathbf{w}$

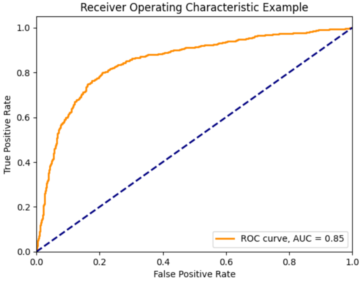
$\mathbf{r} \longrightarrow \Pr(\text{Disease})$

NB $\mathbf{r} = \mathbf{i} - \mathbf{w}$

Residual Regression 1: ROC Curve

The Residual regression model outperforms the image embedding only model.





Residual Regression 2

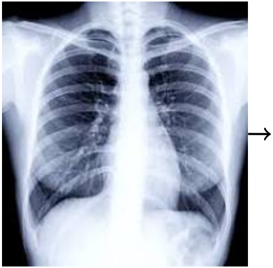
We have image embedding $\mathbf{i}$ and text embedding $\mathbf{w}$ where $\mathbf{w}$ is the embedding: “The phenotype are: {List of phenotype}”

Train a model $f : \mathbb{R}^{512} \to \mathbb{R}^{512}$ to predict $\mathbf{w}$ from $\mathbf{i}$

Then, define $\mathbf{r} = \hat{f}(\mathbf{i}) - \mathbf{w}$

Then, train a model $g : \mathbb{R}^{512} \to [0, 1]$ to predict disease or no disease.

Residual Regression 2: Full Model



$$\begin{tikzcd} {} \arrow[r] & \begin{bmatrix} | \\ \mathbf{i} \\ | \end{bmatrix} \arrow[r, "\hat{f}"] & \begin{bmatrix} | \\ \mathbf{w} \\ | \end{bmatrix} \arrow[r] & \begin{bmatrix} | \\ \mathbf{r} \\ | \end{bmatrix} \arrow[r, "\hat{g}"] & \Pr(\text{Disease}) \end{tikzcd}$$

NB $\mathbf{r} = \mathbf{i} - \mathbf{w}$

Residual Regression 2: Results

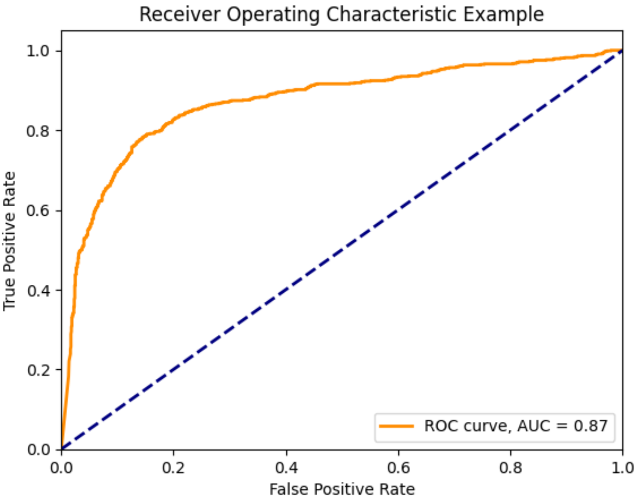


Figure: Accuracy for predicting disease from residuals **r**

Heatmap Interpretation