Medical Image Captioning

Final Project discussion

Shubham Gupta 190020107
Atharva Diwan 190100028
Aanal Sonara 190070064
Omkar Ghugarkar 190020044

30 Apr 2023

Problem Statement

Motivation

Many times we have to write long tedious reports to write. This task can be automated using NLP.

Since the reports are generated by humans, we can use AI to find some smaller abnormalities (if any) that may have been missed by the reporter.

Problem Definition

Our input is an X-ray image which is resized to 224x224x3 and given to the model. Our output is a short medical report summarising the image

Related work

- **This paper uses transformer based text generation [1]**
  - Different from our earlier approach.
  - Uses two models from feature encoder: a tag classifier and then text generator
  - Computationally challenging to implement
- There has been work in multilabel classification (i.e. given an image what all medical terms/tags apply to it) [2]
- Using DL for report generation of orthopaedic trauma [3]
  - Although their claimed their unigram F1 score is very high, they have not given any results or analysis
  - Their finetuning epochs were 50 with BERT model which is very computationally heavy to be implemented

Dataset (s)

- Chest Xray reports are obtained from Indiana University Hospital
- It consisted of 14k images (including augmentation)
- Train-Test split was 80:20
- We use data augmentation on captions/reports by paraphrasing. The reports are very specific to medical terms
  - OpenAI API for text editing (Used for paraphrasing)
  - API for chat based (not used because limited to number of requests)
  - Exchange synonyms in WordNet (scrapped because gives wrong terminology)
  - Back translating (not useful because medical terminology is very specific)

Dataset (s)

True Report: Heart size and mediastinal contour are within normal limits. There is no focal airspace consolidation or suspicious pulmonary opacity. No pneumothorax or large pleural effusion. Mild degenerative change of the thoracic spine.

Augmented report 1: The heart is a normal size and shape, with no abnormalities. There is some evidence of a mild degenerative change in the thoracic spine. There is no focal airspace consolidation or suspicious pulmonary opacity. There is no pneumothorax or large pleural effusion.

Augmented report 2: Heart size, mediastinal contour, and air spaces are normal. No pneumonia, pneumothorax, or large pleural effusion. Mild degenerative change of the thoracic spine.

Workflow, Architecture, Technique

We will be comparing four different approaches which we have evaluated on the UI chest X-Ray dataset

1. ViT encoder + LSTM decoder
    a. Use ViT encoder to generate latent representations for training the lstm network by sequentially generating the captions
2. ViT encoder + pretrained GPT2 decoder fine tuned
    a. Pass the encoder output directly to GPT2 and train on given data
3. ViT encoder + pretrained GPT2 decoder fine tuned with data augmentation
    a. Pass the encoder output directly to GPT2 and train on augmented data

All of these approaches are compared in the following slides.

Workflow, Architecture, Technique

- The architecture involves an encoder and decoder model
- For fair comparison we have kept the same Vision Transformer encoder model. We have used ViT-patch16-224 model for the given computational resources. It is a pretrained model on Imagenet data

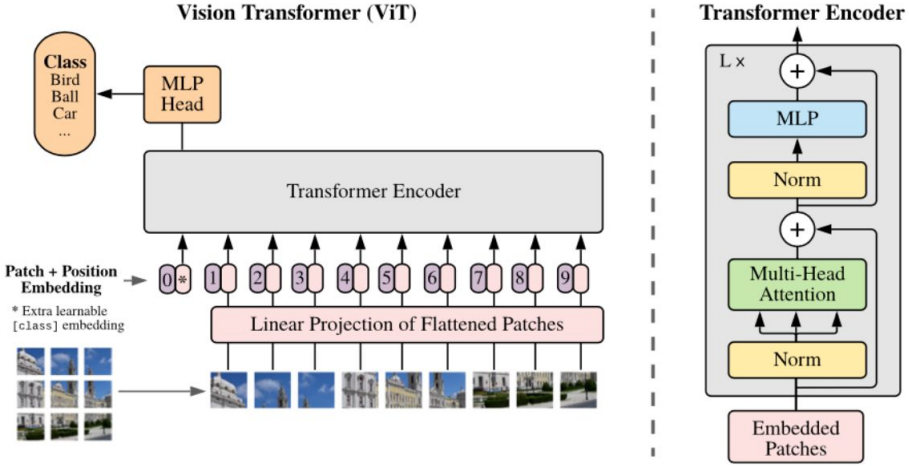


Image Source

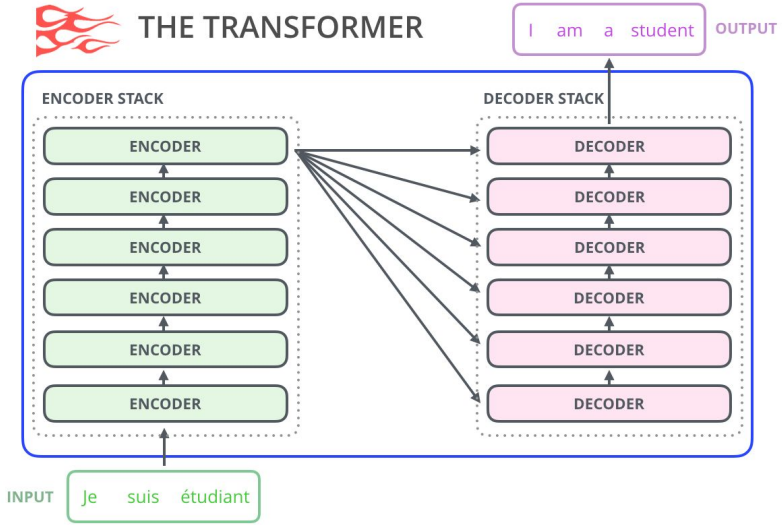


Image Source

Workflow, Architecture, Technique

- Input is an image of size 224x224x3. In case of Grayscale image, we duplicate the image 3 times for 3 channels since this is the only ViT accepts input
- The output of ViT is a 1024 dimensional feature vector of entire image. Given a batch_size, the output of ViT is [batch_size, 1024]
- This is passed as an input to GPT2 pre trained model
- Output is a tokenized sequence vector (max - 256 tokens)
- We use GPT2 pre-trained tokenizer to decode
- Loss - CrossEntropy
- Metric - Rouge Bigram

Results and Analysis

1. ViT encoder + LSTM decoder

```
....
epoch 20: train_loss = 4.567856852213541
sample_0_words = ['ENDOFSEQ']
sample_100_words = ['ENDOFSEQ']
validation_loss = 4.504607200622559
....
epoch 21: train_loss = 4.522620677947998
sample_0_words = ['ENDOFSEQ']
sample_100_words = ['ENDOFSEQ']
validation_loss = 4.5198750495910645
....
epoch 22: train_loss = 4.591316159566244
sample_0_words = ['ENDOFSEQ']
sample_100_words = ['ENDOFSEQ']
validation_loss = 4.584908485412598
....
epoch 23: train_loss = 4.583184878031413
sample_0_words = ['ENDOFSEQ']
sample_100_words = ['ENDOFSEQ']
validation_loss = 4.523614406585693
....
epoch 24: train_loss = 4.573315016428629
sample_0_words = ['ENDOFSEQ']
sample_100_words = ['ENDOFSEQ']
validation_loss = 4.533260822296143
```

Issue while Training: Due to a large maximum sequence length, most of the captions have a lot of NOTSET indices because of extra padding

```
training_description_indices[0:5] = [[4, 47, 31, 8, 40
49, 18, 26, 15, 6, 3, 44, 10, 22, 0, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2],
15, 5, 3, 11, 10, 49, 18, 26, 15, 6, 3, 44, 10, 22, 0,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
```

Results and Analysis

3. ViT encoder + pretrained GPT2 decoder fine tuned with data augmentation

Pretrained Model

Validation Loss	Rouge Precision	Rouge Recall	Rouge Fmeasure
12.154017	0.000000	0.000000	0.000000

Fine-tuning

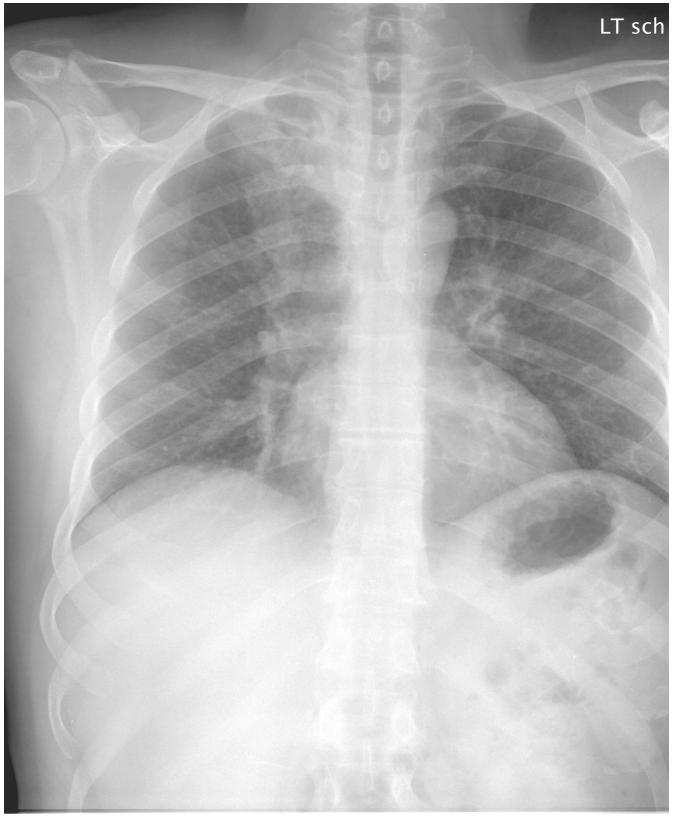
Epoch	Training Loss	Validation Loss	Rouge Precision	Rouge Recall	Rouge Fmeasure
1	0.610800	0.180321	0.201296	0.151377	0.158725
2	0.186500	0.149287	0.220982	0.165205	0.173788
3	0.139600	0.135904	0.220982	0.165205	0.173788
4	0.128300	0.127981	0.220982	0.165205	0.173788

Results and Analysis

- The initial rouge is nearly 0 because of the medical terms being used and the pre-trained models are not able to generate text accordingly
- Fine-tuning improves accuracy and also brings the medical report technically closer to actual report by using the medical terms

Demo

2. ViT encoder + pretrained GPT2 decoder fine tuned



Output of model

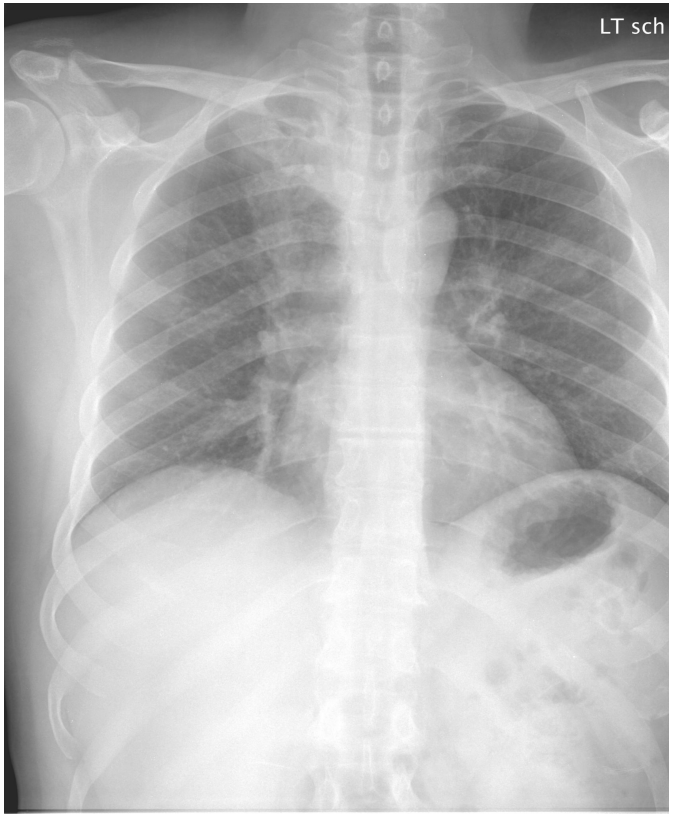
```
[31] generated_caption = tokenizer.decode(model.generate(feature_extractor(img, return_te
     print('\033[96m' + generated_caption + '\033[0m')

     <|endoftext|>No acute cardiopulmonary disease.<|endoftext|>
```

Actual report

There is XXXX increased opacity within the right upper lobe with possible mass and associated area of atelectasis or focal consolidation. The cardiac silhouette is within normal limits. XXXX opacity in the left midlung overlying the posterior left 5th rib may represent focal airspace disease. No pleural effusion or pneumothorax. No acute bone abnormality.

Demo

3. ViT encoder + pretrained GPT2 decoder fine tuned with data augmentation



Output of model

```
/usr/local/lib/python3.10/dist-packages/transformers/generation/utils.py:131
  warnings.warn(
<|endoftext|>No acute cardiopulmonary abnormality.<|endoftext|>
```

Actual report

There is XXXX increased opacity within the right upper lobe with possible mass and associated area of atelectasis or focal consolidation. The cardiac silhouette is within normal limits. XXXX opacity in the left midlung overlying the posterior left 5th rib may represent focal airspace disease. No pleural effusion or pneumothorax. No acute bone abnormality.

Demo (Appendix)

1) Data Augmentation - Untitled1.ipynb - Colaboratory (google.com)
2) Fine-tune model -
   a) Colab - https://colab.research.google.com/drive/1TNgQvLgkQkgwOw4whnOsu_bl-qqmMTil?usp=sharing
   b) Checkpoints with Augmentation- https://drive.google.com/drive/folders/10foEssYRpurMvv6gPOVaF70ecueKqRNq?usp=share_link
   c) Checkpoints without Augmentation- https://drive.google.com/drive/folders/1-LT24KqpIfi1dEgvxXnrKxXvrDhnxp_b?usp=sharing
3) ViT + RNN - https://www.kaggle.com/code/atharvadiwan/medical-image-captioning-with-pytorch-lstm