Combining and evaluating computer vision and natural language processing techniques for use in clinical contexts

Saurav Sengupta

A dissertation for the
Doctor of Philosophy degree
presented to the faculty of the
School of Data Science
University of Virginia
on 21 March 2024

Dissertation committee

Donald Brown, Advisor
Afsaneh Doryab, Chair
Laura Barnes
Sana Syed
Tom Hartvigsen

Combining and evaluating computer vision and natural language processing techniques for use in clinical contexts

Saurav Sengupta

Dissertation Defense submitted to the Faculty of the

School of Data Science, University of Virginia

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Data Science

Committee Members:

Dr. Afsaneh Doryab, Chair

Dr. Donald E. Brown, Advisor

Dr. Laura Barnes

Dr. Sana Syed

Dr. Tom Hartvigsen

March 21, 2024

Charlottesville, Virginia

Keywords: Computer Vision, Natural Language Processing, AI for Healthcare, Computational Pathology, Biomedical Imaging

ABSTRACT

In recent years, deep learning has led to the creation of accurate disease predictors using images, better object detectors for isolating cell structures within images, enabled automated tagging of clinical terms in text based data with accurate named entity recognition models and many more such advances that have the potential to enable better healthcare outcomes by assisting medical practitioners in their work. In this work, we focus on the applications of deep learning, specifically developed for computer vision and natural language processing, for Electronic Health Records (EHR) data, for structured data like diagnosis codes stored over time for patients from different health systems collected by the NIH to help pinpoint risk factors for Long COVID, as well as unstructured imaging and text data in the form of high resolution histopathology images and their paired text descriptions and develop an image captioning system that can automatically generate text descriptions from these images. We also focus on attention based interpretability methods for deep learning models and how they can be used to explain model behavior, as we believe explaining model decisions is crucial for building trust in these models.

In the first portion of this research, we describe our image captioning system developed to generate automatic text descriptions for high resolution histopathology slides built using pre-trained transformer based models to address the lack of training data. We show that by encoding the Whole Slide Image (WSI) as a sequence of image tokens using a Vision Transformer pre-trained on a wide array of histopathology data and feeding them into a Bidirectional Encoder Representations from Transformers (BERT) based language model that has been pre-trained on medical research and clinical notes, we can build a fairly performant captioning system that combines these two data modes, namely images and text. We also show the results of our investigation of cross-attention layers for explanations of model behavior.

In the second portion of this research, we describe how we used a Long Short Term Memory based neural network model, developed to analyze text sequences, to analyze temporally arranged diagnosis code data recorded up to the first COVID-19 infection to investigate risk factors for Long COVID. While modeling sequential diagnosis codes has been done before, we add to this model by re-using pre-trained embeddings to encode the diagnosis data instead of randomly initializing the embeddings. We also discuss the use of a semi-supervised learning technique called as positive unlabeled (PU) learning to improve model learning to build a better classification model. We then investigate these models using self-attention weights generated during training to answer our stated goal of building models whose behavior can be investigated.

We present in this work different methods to apply deep learning in clinical contexts. We also validate by the results presented in this work that models developed for language modeling, like LSTMs, can be successfully adapted for other sequential or time series data like temporally arranged diagnosis data. We also build an encoder-decoder model, inspired by existing image captioning models, but adapted for high resolution images that can find applications in other domains like remote sensing where there is a need to handle high resolution imaging data. We believe this work generates further proof that deep learning can be successfully used in clinical contexts, while also showing how we can investigate model behavior using attention based modeling, which is crucial for developing trust in these black box models.

Dedicated to University of Virginia.

Acknowledgments

I would like to thank my advisor Dr. Donald Brown who gave me the opportunity and the resources to tackle interesting challenges with complete freedom. He was generous with his time and his valuable feedback helped tremendously. I would also like to acknowledge the support and guidance of Dr. Sana Syed, without whom I might not be completing a PhD today.They were instrumental in guiding the direction I wanted my dissertation to go, along with providing critique and knowledge that paved the way for my research.

I would also like to thank my dissertation advisory committee members, Dr. Laura Barnes, Dr. Afsaneh Doryab and Dr. Tom Hartvigsen for their comments and feedback that helped refine my dissertation.

Without the love and support of my parents, Sumit and Maushumi Sengupta, I would not have been in this position of finishing my doctorate. Their constant moral support and encouragement helped me edge over the finish line, and I am eternally grateful to them. I would like to dedicate this work to them.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Deep Learning

Deep learning, as defined by Turing prize award winners LeCun et al. [42], refers to powerful computational models with multiple processing layers with the ability to do representation learning on the data with multiple levels of abstraction. At its core, deep learning consists of multiple non-linear layers of neurons, with each layer connected to the previous, with the ability to learn more complex representations of the data at each subsequent layer. As the authors in [42] describe, 'With multiple non-linear layers, say a depth of 5 to 20, a system can implement extremely intricate functions of its inputs that are simultaneously sensitive to minute details ..'

Deep learning has revolutionized areas like speech recognition, image recognition and segmentation with Convolutional Neural Networks (CNN) and more recently, being able to generate images from natural language input with the help of a technique called as Stable Diffusion [33, 34, 58]. This ability to handle versatile input data to build effective models has helped increase its popularity over recent years, leading to explorations of its applications into more specialized areas like medicine and finance [14, 22].

1.2 Deep learning in Healthcare

Deep learning has the capability to build effective models without robust feature engineering of the input data [11]. Early implementations of automated technologies included computer aided detection (CAD) for mammograms, which were rolled out in hospitals with the intention of helping radiologists with identifying subtle cancers that might be missed, but were unable to improve screening performance [29, 44]. The implementations mostly depended on engineered image features generated by image processing techniques reliant on specialized filters, that were then fed into a classification algorithm to determine the label [11, 70]. Convolutional neural networks (CNN) with thousands of learnable filters removed the need for generating image features using specialized filters [40]. As we train deeper networks, CNNs are able to learn more complex image features given vast amounts of training data and perform better on image classification tasks like ImageNet [23]. This performance is replicated for clinical images like chest X-rays and histopathology images [35, 65]. Therefore, there is a clear case for deep learning to be applied to varied healthcare scenarios like radiology as compared to traditional methods to drive better clinical outcomes.

1.3 Multi-modal machine learning in healthcare

Machine learning as described above is usually focused on a single mode of data at a time, be it images for disease classification or segmentation or looking at electronic health records to model disease phenotypes [39]. However, a physician usually considers multiple modes of clinical data at once to determine diagnosis and treatment. Therefore, it becomes important to build multi-modal models, that can interpret and understand data from, say for example, images as well as text to make its decisions.

1.4 Need for interpretability

The challenge with applying deep learning to healthcare is the need for the models to be interpretable, since the wrong decisions can negatively impact human life [4]. Interpretable in this case means that we should be able to query why a model made a certain decision to be completely sure of the model recommendation, be it disease classification or survival prediction among many other uses. This has the added benefit of being able to go back and apply corrections in case of mistakes. This can be especially challenging in healthcare scenarios since often model inputs are heterogeneous, like temporal medical records, clinical text or medical imaging data. This might not be a problem when outcomes are modeled using linear modeling with domain knowledge based handcrafted features, however, deep learning models with their millions of parameters can be hard to decipher. Cynthia Rudin, in her paper *'Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead'* makes the case that using models that are by design interpretable should be the way forward [59]. However, several models that are not inherently interpretable might still outperform interpretable models. And there are several post-hoc interpretation methods like saliency maps, attention layers and de-convolution that have shown promising results [60]. While the argument can be made that basing life-changing decisions like clinical diagnosis should not be made using black box models that cannot be investigated in case of mistakes, there are ways to understand decisions using the post-hoc methods mentioned earlier, while also making it clear that these models behave in an assistive capacity that can help a physician make an informed choice and streamline workloads. In conclusion, the need for interpretable methods for medical applications of deep learning help us leverage the power of modern neural networks while making them more trustworthy to use.

1.5 Contributions

Motivated by the considerations described above, this work puts forwards the following contributions:

1. In Chapter 3, we describe an extensible, transformer-based multi-modal system that incorporates imaging and text based data, by creating a captioning system that can automatically generate descriptions of a high resolution histopathology image. We have built a pipeline that incorporates existing transformer-based vision-language modeling for high resolution images, demonstrated here using histopathology whole slide images (WSI) but usable in other domains, where such high resolution imaging is prevalent, along with the added benefit of using highly parallelizable architectures that using an all transformer model provides.

2. In Chapter 4, we have evaluated the use of cross-attention for interpreting the automated report generation model described in Chapter 3 and shown that there is added benefit in understanding model decision-making using attention values. It not only enables validating model decisions, but also improves trustworthiness in the model. We are able to do this because of the nature of the architecture described in chapter 3, that is able to preserve location information while encoding a high resolution histopathology image.

3. In Chapter 5, we describe a method to improve model performance using Positive-Unlabeled (PU) learning demonstrated on a model predicting Long COVID, using a new method call RF bagging adapted from Mordelet and Vert [49]. We show that by generating a reliably negative (RN) sample, we are able to build a better the discriminative model, and therefore generate more trustworthy interpretations from the model, since the model is now more sure of its predictions. This is important

because our main goal was to investigate this model to determine risk factors for Long COVID. With the help of this model, we can now understand what in the patient's medical history would pre-dispose them towards Long COVID. This is also in keeping with our stated goal of building reliable and interpretable models.

Chapter 2

Literature Review

2.1 Transformers

With how ubiquitous large language models (LLMs) like OpenAI's ChatGPT [50] or Meta's LLaMA [72] have become in recent years, the Transformer, the underlying architecture powering them, can reliably be crowned the seminal deep learning architecture of the last decade. Proposed in 2017 by Vaswani et al. [74], transformers were an alternative to Recurrent Neural Networks (RNN) [66] for sequential modeling. RNNs suffer from the vanishing gradient problem and are non-parallelizable which increases training time [41]. Transformer blocks consisting of fast-forward neural networks and self-attention modules solved the problem of local and long term dependencies with the added benefit of being parallelizable thereby reducing training time [53]. Modifications of the underlying architecture to reduce training time further by optimizing the transformer's time complexity that scales quadratically with sequence length to linear time, have also made it more desirable and efficient to use transformers [75].

A Transformer unit as a stack of six encoders and six decoders. Each encoder or decoder unit consists of a self-attention unit and a feed-forward neural network, whose output is connected to the input of the encoder/decoder on top of it. The decoder also contains another layer that does performs attention over the encoder outputs as well, which is called as cross-attention. The presence of self-attention enables the transformer to focus on the correct word in a whole sequence of words in a source sentence, say, during translation to the target sentence in another language. These transformer layers also employ skip connections along with layer normalization, thereby propagating the input and added learned embeddings throughout the layers.

2.1.1 Multiheaded Self-Attention

In intuitive terms, attention allows a deep learning model to add context to the current input word by 'attending' or giving different weights to the words that surround that input word. This is important when the input sequence is long and therefore this context information is useful in understanding what is the intended meaning of the word at a certain point in the sequence.

In transformers, self-attention works by generating 3 vectors, query(Q), key (K) and value (V) from the input vector and then combined as follows:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V$$

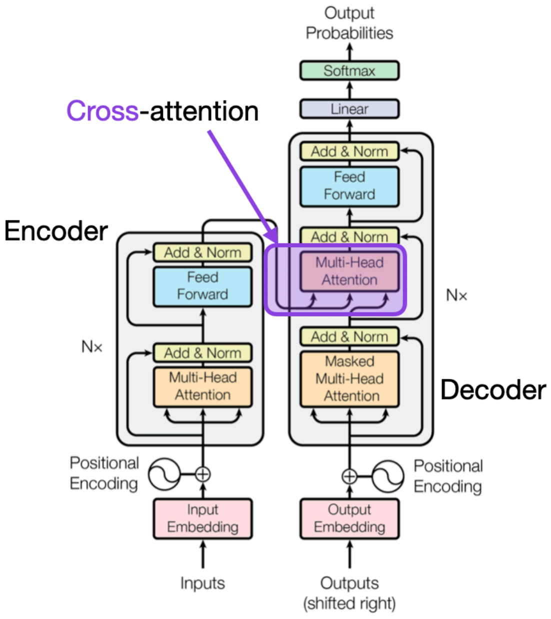
where d_k is the dimension size of the query and key vectors.

The authors of the transformer paper call it the scaled dot product attention since they use a scaling factor of $\frac{1}{\sqrt{d_k}}$ to scale it, since for higher values of d_k the dot products grow large in magnitude and therefore need to be scaled for the gradients to make sense.

Vaswani et al. [74] argue for the use of multiple h attention heads, instead of the single one described above. The intuitive explanation for this is that instead of one head generating single context weights for a input query, multiple heads generating context weights focusing on different sections of the input sequence is more helpful. Each attention head weights the surrounding context differently, thereby allowing the model to focus on different portions of the input sequence.

2.1.2 Cross Attention

Cross-attention is used to combine input sequences of different lengths, which is helpful in the context of a training the model for translating from one language to another, where the source and target sequences need not necessarily have the same length.



Source: “Attention Is All You Need” (https://arxiv.org/abs/1706.03762)

Figure 2.1: Cross-attention; source: Understanding Self-attention from scratch - Sebastian Raschka

Cross attention works by calculating the key K and value V vectors from source input sequence $S1$ and the query Q from the target output sequence $S2$. Mathematically, it is done using the following:

$$CrossAttention = softmax(\frac{(W_qS_2)(W_kS_1)^T}{\sqrt{d_k}})(W_vS_1)$$

where W_q, W_k and W_v are learned weight matrices used to calculate the queries, keys and values respectively.

Cross attention is of particular help when there is a need to translate from one embedding space to another, as is the case during machine translation or even image captioning.

2.1.3 Positional Encoding

Vaswani et al. [74] also define a position encoding that encodes the position of each word in a sentence such that it preserves the relative positioning of each word and generalizes to longer sentences without any effort. A positional encoding therefore is a d-dimensional vector that indicates the position of a word based on sinusoidal functions, with even positions represented by sine and odd positions encoded by cosine function.

2.1.4 BERT

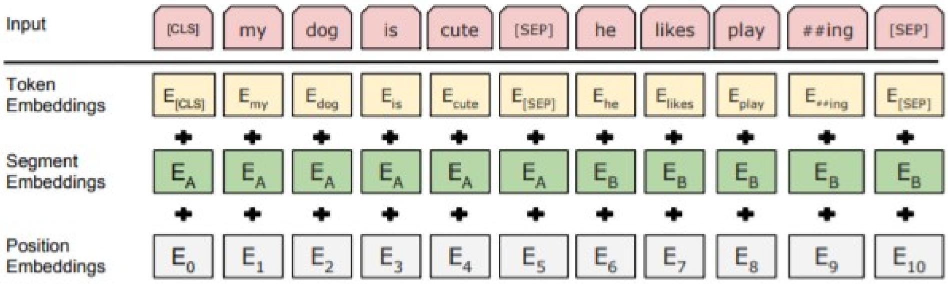


Figure 2.2: BERT input setup from Devlin et al. [25]

BERT is a transformer model pre-trained on a large corpus of unlabeled text data that is able to learn bidirectional representations for text and has been successfully used for various downstream tasks in like Question Answering (QA), Named Entity Recognition (NER) and Natural Language Inference (NLI) [24]. BioBERT and ClinicalBERT, trained on medical research articles and clinical notes data respectively, performed better on clinical downstream tasks than just BERT [3].

In the BERT paper, they achieve bidirectional pre-training of a language model by achieving two tasks, masked language modeling (MLM) and next sentence prediction. MLM means to randomly mask certain words in the input and predicting that masked word. BERT also randomly inserts another word in place of the correct word and asks the model to predict the current word. Next, sentence prediction involves giving the model two input sentences and asking it to predict if the second sentence follows the first or not.

The input for BERT is as shown in Figure 2.2. Each input consists of a single or a pair of sentences, with each word in the sentence treated as a token. At the start of each sentence, a [CLS] (classification) token is added and a [SEP] token is added between a pair of sentences. The final hidden state of the [CLS] token is used as the aggregate sequence representations for the classification task. The intuition behind the [CLS] token is since it is never masked and always in the beginning of each sentence, the 768-dimension embedding from the trained BERT model captures a weighted average of information from the input sequence when we fine-tune it in further downstream sentence classification tasks.

2.1.5 Vision Transformers

Vision transformer (ViT) was proposed by Dosovitskiy et al. [26] as an analogous architecture to the transformer, but for images. The idea here is to mimic how we tokenize sentences into words, by patching an image, such that each patch is now a visual token, which can now be fed to a transformer like architecture. The authors show that by training the ViT on a large corpus of images, they are able to beat a baseline convolution neural net (CNN) architecture. The key thing to note here is that since there is much less inductive bias when compared to CNNs and therefore need a large amount of data to beat a CNN based

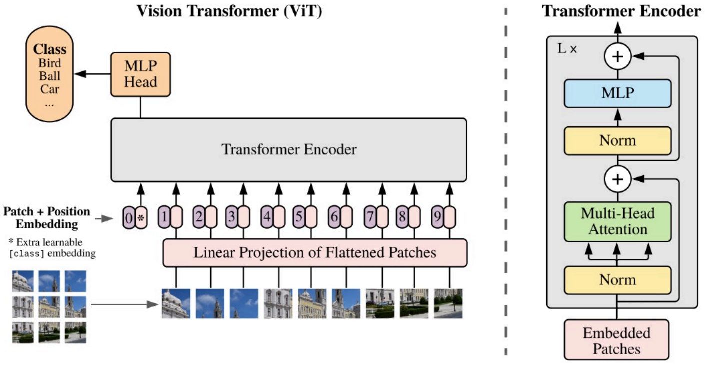


Figure 2.3: Vision transformer from Dosovitskiy et al. [26]

architectures. Nevertheless, the authors argue that since a transformer based architecture is inherently more scalable, this trumps the inductive bias advantage of CNNs by showing good results on different image classification benchmarks.

2.1.6 DINO Pre-training

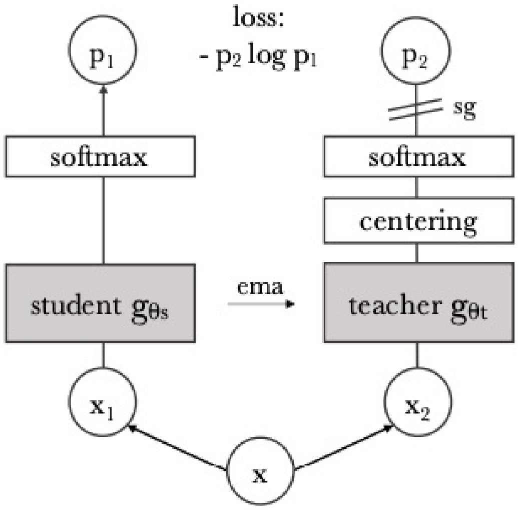


Figure 2.4: DINO illustration from Caron et al. [15]

Proposed by Caron et al. [15], self-**di**stillation with **no** labels or DINO, is a self-supervised method to train vision transformers.

It consists of a teacher and student network, each having the same architecture but different parameters. They are fed two different random transformations of the same image. The output of the teacher network is mean centered over the batch and then a softmax function is applied, while no such centering takes place on the output of the student. The K-dimensional output encodings from both networks are then compared using a cross-entropy loss that measures their similarity. The loss gradient is only back-propagated through the student network, while the parameters of the teacher network is updated by taking the exponential moving average (EMA) of the student parameters. The authors show that using this pre-training mechanism, they are able to achieve 80.1% top-1 on ImageNet in linear evaluation with ViT-Base.

2.1.7 HIPT

In recent years, Chen et al. [17] have proposed a self-supervised Vision Transformer (ViT) based image representation learning mechanism called Hierarchical Image Pyramid Transformer (HIPT). The self-supervised pre-training leverages DINO (distillation with no labels) from Caron et al. [15] at two levels, 256x256 sized patches and 4096x4096 sized patches. The authors show that this can then be leveraged for further downstream tasks like disease sub-typing and survival prediction, as the pre-trained ViT representations are now looking at the WSI at multiscale level.

2.2 Electronic Health Records (EHR)

Electronic Health Records (EHR) often contain a variety of data associated with a patient, like medication, diagnoses, blood pressure and heart rate measurements, free-text clinical notes, radiology images and much more. These heterogeneous data points are collected at different times over the duration of a patient's visit to a health system. By leveraging the temporal nature of this data as well as the information present in them, we can train models to potentially help predict clinical outcomes given the past medical history of a patient. Therefore, it is important to capture information from this wide variety of data and its temporal nature into a vector representation that can further be used for predictive modeling or patient clustering analysis.

2.2.1 Neural Networks for modeling EHR

Numerous previous studies have investigated the use of deep learning for analyzing Electronic Health Record (EHR) data. REverse Time AttentIoN model (RETAIN) for application to Electronic Health Records (EHR) data [18] used Recurrent Neural Networks (RNNs) to analyze time labeled sequence of observations like diagnosis codes where each patient is represented by a sequence of temporally arranged tuples, while also being interpretable. The model was able to assign a feature importance score to each input feature using outputs from its two constituent RNN networks for calculating attention at both visit level and variable level. One of the earliest papers used plain LSTMs on temporally arranged International Classification of Diseases-9th revision (ICD-9) diagnosis codes to predict 128 common diseases like heart failure and respiratory distress [47]. Similar studies using LSTMs for analyzing temporal diagnosis codes, vital signs and medications for future disease onset prediction have also been shown to perform well [21, 71]. These studies motivate our goal of analyzing historical diagnosis codes prior to the COVID-19 infection present in the N3C database to first predict if a patient is suffering from Long COVID, and then investigate these models for associations between a patient's diagnosis history and Long COVID.

Chapter 3

Automatic caption generation for histopathology images

3.1 Introduction

High resolution histopathology slides are a rich resource of information that current deep learning methods are able to exploit for various use cases like disease classification, cell segmentation and outcome prediction. However, as the images are very high resolution, usually in the range of 150,000x150,000px, they often require non-trivial modifications to existing state-of-the-art (SOTA) deep learning architectures to be used successfully. The most common method for handling these high resolution images is to patch the bigger image into smaller sized images that can be fed into Convolutional Neural Networks. For example, in a classification setting, this often works as a multiple instance learning problem, where each patch is given the same overall image label. A potential drawback to this is that patching can lead to removal of overall context from the whole slide image (WSI) that the model might need to learn to make the correct decision, unless handled properly.

Automatic report generation for histopathology images is an area of existing research that also suffers from the need for modifying SOTA image captioning architectures to fit researchers needs. Image captioning for histopathology helps us combine two rich sources of information, that is, high resolution WSIs and associated diagnostic reports that describe features of the image. In clinical settings, automatic report generation has been successfully used for X-ray images and claim to reduce the burden for radiologists by assisting them in describing the image [52]. Other use cases for automated image captioning in medical images can be image retrieval, as generated reports could be part of a searchable database, and encouraging standardized clinical ontologies by using words from a standard vocabulary to describe similar things. Therefore, automated image captioning for histopathology can be similarly useful for a wide variety of tasks that can assist physicians and radiologists in their tasks.

3.2 Related Work

Current research for histopathological image captioning focuses on Convolutional Neural Network (CNN) based encoder and Recurrent Neural Network (RNN) based decoder architectures [30, 73, 78]. This is inspired by Show, attend and tell paper, that in particular has the capability of using the attention mechanism to focus on certain areas of the image to generate captions [77].

Using ImageNet pre-trained CNN-based encoders to encode smaller sized patches of the high resolution WSI has been successfully used in a variety of ways to essentially reduce the size of large dimensional WSIs to smaller and computationally manageable representations [16]. In recent years, a self-supervised Vision Transformer (ViT) based image representation learning mechanism called Hierarchical Image Pyramid Transformer (HIPT) has been proposed [17]. The self-supervised pre-training leverages DINO (distillation with no labels) at two levels, 256x256 sized patches and 4096x4096 sized patches. The authors show that this can then be leveraged for further downstream tasks like disease sub-typing and survival prediction[15].

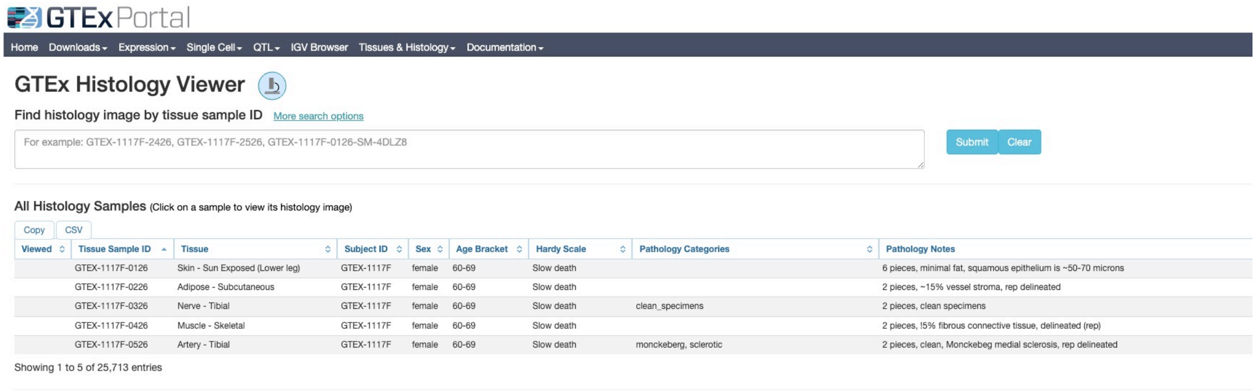
A recent work uses a two-step process in which they first encode all patches of a WSI using a triplet loss based convolutional autoencoder and use the features from the bottleneck layer to cluster the patches into $k \in [1, 2, 3..7]$ clusters [78]. In the second step they randomly sample the patches from each cluster, use a ImageNet pretrained ResNet-18 to extract N-dimensional features for the k-patches, then use attention pooling to reduce $k \times N$ dimensional feature vector to $1 \times N$ and then feed into a LSTM decoder to generate captions. More recently, Gamper and Rajpoot [30] describe the ARCH dataset which contains histopathology images extracted from textbooks and their associated descriptions, which they use for caption generation based pre-training task to generate an encoder that when used for downstream tasks like multiple instance learning shows promising results compared to other pre-trained encoders. But as noted in Tsuneki and Kanavati in [73], these images, as they are curated from textbooks and research articles, can be of mixed quality, magnifications, and resolutions that while useful, does not really solve the problem for existing high resolution WSIs being generated in hospital systems everywhere. In Tsuneki and Kanavati [73], the authors use high resolution WSIs from a Japanese hospital system and associated translated text reports, for their automated captioning system. They use EfficientNetB3 [68] and DenseNet121 [36] pretrained on ImageNet dataset and extract features from the penultimate layer for 300x300 patches extracted from the WSI. They then use global average pooling and 3x3 average pooling to reduce the feature sizes and feed them into an RNN based decoder for generating their captions.

BERT is a transformer model pre-trained on a large corpus of unlabeled text data that is able to learn bidirectional representations for text and has been successfully used for various downstream tasks in like Question Answering (QA), Named Entity Recognition (NER) and Natural Language Inference (NLI) [24]. BioBERT and ClinicalBERT, trained on medical research articles and clinical notes data respectively, performed better on clinical downstream tasks than just BERT [3].

More recently, pre-trained transformers have been successfully used for optical character recognition (OCR), that is, converting text in images to machine-readable text [45]. The authors were able to outperform SOTA approaches for OCR using pre-trained Vision Transformers and transformer-based language models. The authors utilize the image representations from the vision transformers and, along with the context generated before, use it to predict the next tokens. A '[BOS]' and '[EOS]' tokens are appended at the beginning and end of the ground truth tokens. Note that using '[BOS]' token shifts the sequence to the right by one place and is used to indicate start of generation.

Motivated by the success of pre-trained transformer models for such diverse downstream tasks, we propose a method that uses HIPT to encode WSIs that captures multi-level representations, and a BERT based decoder that is able to utilize powerful text representations to generate descriptions of the WSI.

3.3 Dataset



GTEx Histology Viewer

Find histology image by tissue sample ID More search options

For example: GTEX-1117F-2426, GTEX-1117F-2526, GTEX-1117F-0126-SM-4DLZ8

All Histology Samples (Click on a sample to view its histology image)

Viewed	Tissue Sample ID	Tissue	Subject ID	Sex	Age Bracket	Hardy Scale	Pathology Categories	Pathology Notes
	GTEX-1117F-0126	Skin - Sun Exposed (Lower leg)	GTEX-1117F	female	60-69	Slow death		6 pieces, minimal fat, squamous epithelium is ~50-70 microns
	GTEX-1117F-0226	Adipose - Subcutaneous	GTEX-1117F	female	60-69	Slow death		2 pieces, ~15% vessel stroma, rep delineated
	GTEX-1117F-0326	Nerve - Tibial	GTEX-1117F	female	60-69	Slow death	clean_specimens	2 pieces, clean specimens
	GTEX-1117F-0426	Muscle - Skeletal	GTEX-1117F	female	60-69	Slow death		2 pieces, !5% fibrous connective tissue, delineated (rep)
	GTEX-1117F-0526	Artery - Tibial	GTEX-1117F	female	60-69	Slow death	monckeberg, sclerotic	2 pieces, clean, Monckebeg medial sclerosis, rep delineated

Showing 1 to 5 of 25,713 entries

Figure 3.1: Screenshot of the GTEx portal https://www.gtexportal.org/home/histologyPage. We scraped the histology data from this website similar to Zhang et al. [78].

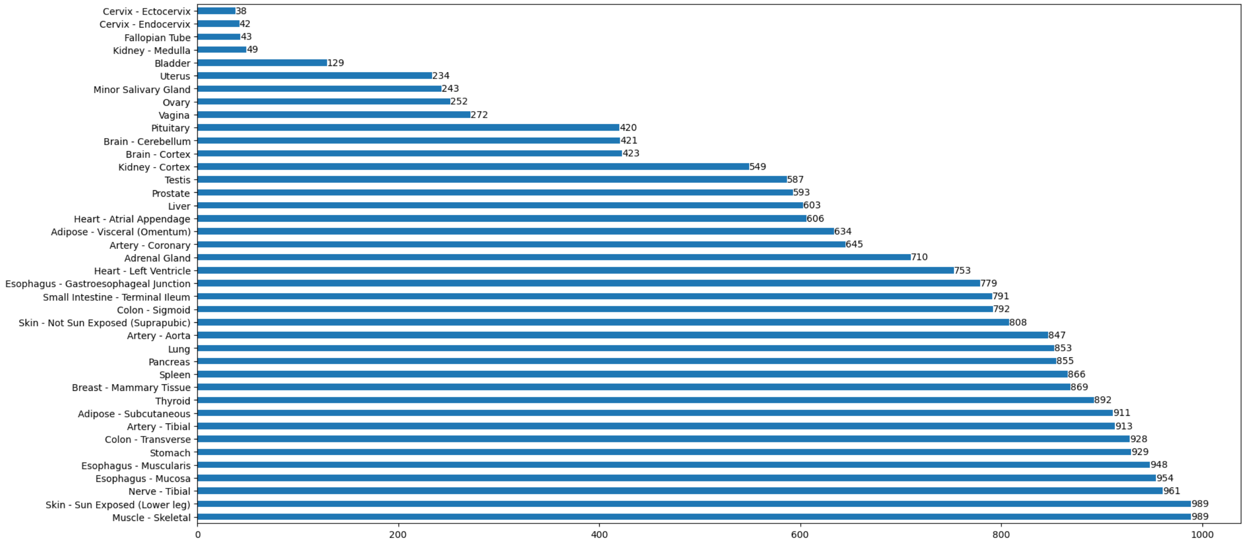


Figure 3.2: Tissue Distribution of the data from GTEx portal.

We get our imaging and associated text data from the Genotype-Tissue Expression (GTEx) portal (https://www.gtexportal.org/home/histologyPage), same as [78] (refer Figure 3.1), resulting in about 25,120 samples of WSI-text pairs. We divide our dataset into 23517 training, 603 validation and 1000 testing samples. There are 40 different tissue types, and you can find the plot showing the counts of different tissues in Figure 3.2. In Figure 3.3 you can find a word cloud describing the 'Pathology Note' section of the data. We use a stratified split such that each tissue type is represented in the training, validation and test

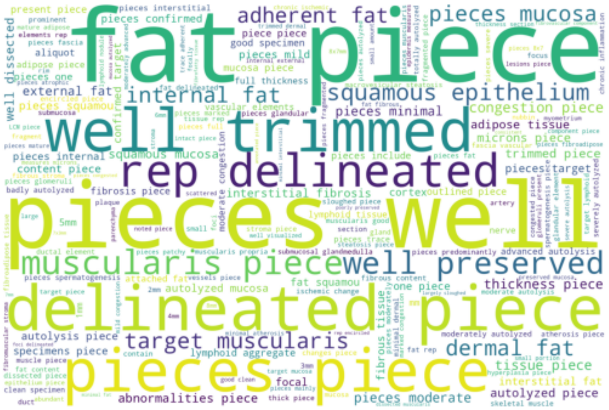


Figure 3.3: Word cloud of the pathology notes section of the data.

sets.

We do this because this is the only publicly available high resolution histology data with associated descriptions of each histology slide at the time of writing this paper that we could find. Data from Tsuneki and Kanavati [73] comes in the form of 300x300 patches[1] and the ARCH dataset by Gamper and Rajpoot[30] are of mixed quality, magnifications, and resolutions that do not meet our criteria of working with high resolution Whole Slide Images which are the most likely form of data available in health systems.

3.3.1 Caption pre-processing

Data in the GTEx portal comes in a tabular format with *tissue_type*, *sex* and *pathology_notes* in separate columns. We create the description for each tissue in the following format:

this is a {tissue_type} tissue from a {sex} patient and it has {pathology_notes}.

An example caption would look like this: *this is a* ***small intestine - terminal ileum*** *tissue from a* ***male*** *patient and it has* ***6 pieces, prominent lymphoid component in 4 of 6 pieces.***

This helped achieve two objectives:

1. Increase sentence length such that this data mimics real word notes, as they are bound to be longer in length.

2. Helps check how good the captioning system is at tissue classification.

[1] https://zenodo.org/record/6021442

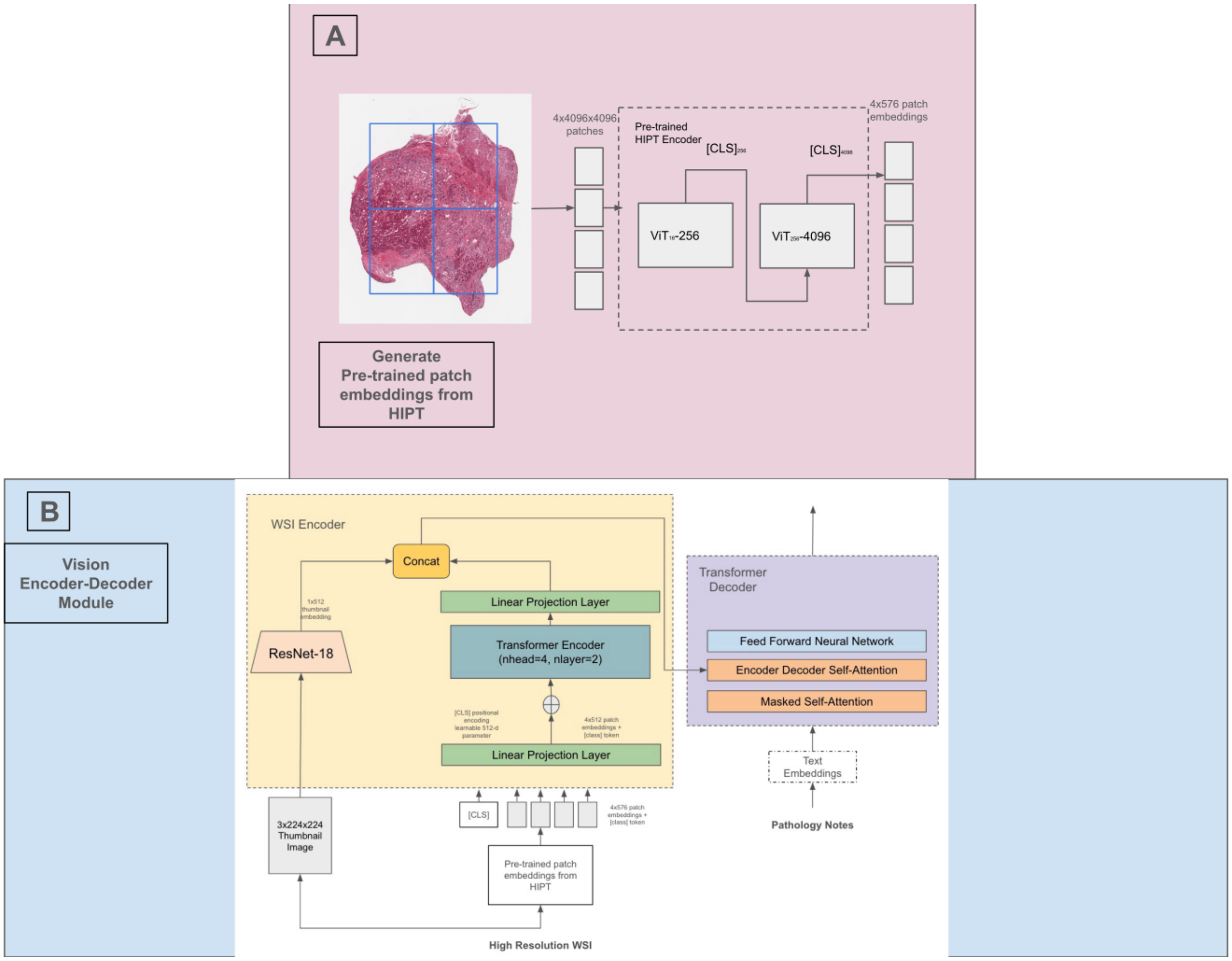

A
4x4096x4096 patches
Pre-trained HIPT Encoder
[CLS]$_{256}$
[CLS]$_{4096}$
ViT$_{16}$-256
ViT$_{256}$-4096
4x576 patch embeddings
Generate Pre-trained patch embeddings from HIPT
B
Vision Encoder-Decoder Module
WSI Encoder
Concat
1x512 thumbnail embedding
ResNet-18
Linear Projection Layer
Transformer Encoder (nhead=4, nlayer=2)
[CLS] positional encoding learnable 512-d parameter
4x512 patch embeddings + [class] token
Linear Projection Layer
3x224x224 Thumbnail Image
[CLS]
4x576 patch embeddings + [class] token
Pre-trained patch embeddings from HIPT
High Resolution WSI
Transformer Decoder
Feed Forward Neural Network
Encoder Decoder Self-Attention
Masked Self-Attention
Text Embeddings
Pathology Notes

Figure 3.4: Method overview for automatic image captioning. (A) We pre-extract embeddings from the HIPT [17]. (B) We add a learnable [CLS] token in front of all our patch tokens and feed it to the patch encoder. We separately feed the thumbnail image of the WSI to the thumbnail encoder. We concatenate these two representations such that each patch token representation has the added image context vector from the thumbnail. We feed this concatenated WSI representation to our transformer decoder for our caption generation.

3.4 Methodology

3.4.1 Creating patch embeddings

In the formulation for Vision Transformers by Dosovitskiy et al. [26], the authors break the image into 16x16 patches to generate visual tokens that act as an analogue to separating sentences into word tokens. Inspired by that, we break our high resolution WSI into 4096x4096 patches. This patch size is motivated by the patch size used by the authors of HIPT [17] for generating multiscale representations at the 16x16 and 256x256 patch level, which are generated by two different ViT architectures, denoted by ViT_{256}-16 and ViT_{4096}-256. (Notation Guide: In the notation $\mathrm{ViT}_{\mathrm{L}}$-$l$, l is the size of the $l \times l$ non overlapping tokens extracted from the $L \times L$ or $\mathbf{x}_L$ input image to each ViT subnetwork, following the notation described in [17].)

The method to tokenize each image into patches and extracting their embeddings from HIPT is done as follows:

1. Each WSI generates M patches, and for each patch we extract $256\mathrm{x}[\mathrm{CLS}]_{256} \in \mathbb{R}^{1\times384}$ from ViT_{256}-16 and one $[\mathrm{CLS}]_{4096} \in \mathbb{R}^{1\times192}$ from ViT_{4096}-256.

2. Therefore, each patch M^i now has a representation $\mathrm{P}^i_{256} \in \mathbb{R}^{256\times384}$ from the $256\mathrm{x}[\mathrm{CLS}]_{256}$ vectors and $\mathrm{P}^i_{4096} \in \mathbb{R}^{1\times192}$ from the $[\mathrm{CLS}]_{4096}$ vector.

3. We take the mean of P_{256} along dim=0 to generate $\mathrm{P}^{mean}_{256} \in \mathbb{R}^{1\times384}$ and concatenate it with P_{4096} to generate $\mathrm{P}_{patch} \in \mathbb{R}^{1\times576}$.

4. Each WSI can now be represented as a collection of M patch tokens such that WSI = $\{\mathrm{P}^1_{patch}, \mathrm{P}^2_{patch} ... \mathrm{P}^M_{patch}\}$.

We pre-extract these patch embeddings before training our vision encoder-decoder model.

3.4.2 Encoder

Our vision encoder consists of two parts, one that takes in the patch embeddings prefixed with classification token, similar to the ViT paper, and an ImageNet pre-trained ResNet-18 based thumbnail encoder, that takes in a 3x224x224 resolution thumbnail image of the WSI. The thumbnail encoder is added to provide additional visual context to the decoder to generate associated captions.

The patch encoder, consists of a linear layer, a transformer encoder and another trainable projection layer that transforms the transformer output. The intention again is to mimic the original ViT paper by Dosovitskiy et al. [26] which in turn mimics the BERT paper by Devlin et al. [25], by prepending an additional learnable classification [CLS] token, that is supposed to encode the whole image, and training a transformer encoder for image recognition. Refer to Figure 3.4 for an illustration of our vision encoder module.

We add another learnable projection layer on the transformer output and concatenate the two representations, one that encodes the WSI at the multiscale level (16x16 and 256x256) and another that encodes the WSI at a lower magnification (higher level) from the thumbnail encoder that is analogous to how a pathologist might make their diagnosis by zooming in and out of the full resolution high magnification image.

Since each WSI can have a different number of patches, we add padding to make the sizes uniform. We supply an encoder attention mask, that is 1 for all the patch and added [CLS] tokens, and 0 for the pad tokens. This is used to tell the decoder, which portions of the image to pay attention to during generation of the note.

3.4.3 Decoder

We utilize the pre-trained ‘bert-base-cased’ model from Devlin et al. [25] with 12-layers, a hidden size of 768, 12 attention heads in each layer for a total of 110 million parameters. Since we are using BERT as a decoder for caption generation, cross-attention layers (for brevity, we refer to cross-attention as *xattn* in the results section) and a final language modeling head is added to make the next token prediction.

We chose the base BERT since it has publicly available pre-trained versions like BioBERT [43] or BioClinicalBERT [3], that are pre-trained on PubMed articles and clinical notes. These pre-trained initializations are useful during fine-tuning as they incorporate clinical knowledge, which is crucial in our case since we are dealing with pathology notes.

During training, the source words are appended with a BERT specific ‘[BOS]’ token at the beginning of the sentence and an ‘[EOS]’ token at the end of the sentence to denote the end of the clinical note to generate the target sequence. The ‘[BOS]’ token shifts the target sequence to the right, and the task at training therefore becomes to predict the next token based on the previous token and the encoded image features. We are using ‘teacher-forcing’ for training, since we are supplying the ground-truth to the decoder.

3.4.4 Training

We add padding to each collection of WSI patch tokens such that the total length is 128. We use a pad size of 128 for our text tokens and use the pre-trained BERT tokenizer to tokenize the clinical notes.

For the thumbnail encoder, we add random augmentations like random resized crop and random horizontal flipping to the thumbnail image during training.

We use the CrossEntropy loss function and a learning rate of 1e-3 with the Adam optimizer. We use a batch size of 32, with mixed precision training and a learning rate scheduler that decays the learning rate by 0.8 when there is no decrease in the validation loss score for 6 consecutive epochs. We also incorporate value based gradient clipping to tackle the exploding gradient problem. If the validation loss does not improve for 12 consecutive epochs, we stop the training and save the model with the best validation loss. All our models were trained on a single NVIDIA A100 GPU, and mixed precision training helped reduce training time from 10hrs to 2hrs.

3.5 Experiments

We use natural language evaluation metrics like BLEU-4 [51], METEOR [8] and ROUGE-L[46] scores to evaluate how closely the generated captions match the actual captions. We report these metrics for the overall sentence and for just the *pathology_notes* section. We also calculate the accuracy of generated $tissue\_type_g$ to the actual $tissue\_type_a$. These metrics, we believe, allow us to holistically evaluate our model. We report our results on a held out test set of 1000 patients. Note that we forgo reporting metrics on gender classification as there is no visual way to determine gender from tissue images, and therefore it is a meaningless metric.

During inference, we choose greedy decoding (beam size=1), as our method to generate the related descriptions of the image.

In this first set of experiments, we evaluate the effect of the thumbnail encoder in our encoding process. The decoder used is BERT-base-cased with pre-trained weights from Devlin et al. [25].

In Tables 3.1 and 3.2, we report the results of the average of 3 runs on our held out test set of 1000 WSIs, with natural language metrics for both the whole generated notes in Table

3.1 and just the 'Pathology Note' section of the dataset in Table 3.2, which are denoted by the prefix N. We also report the accuracy of the tissue type detected by the model while generating the note in Table 3.1.

Table 3.1: Effect of ResNet-18 thumbnail encoder on caption generation. Note that the decoder used is BERT-base-cased for both these experiments; mean/std of 3 runs;

Model	**Tissue Acc.(%)**	**BLEU-4**	**ROUGE_L**	**METEOR**
VIT	89.867±0.723	0.565±0.006	0.737±0.003	0.703±0.002
VIT+ResNet	91.567±1.102	0.567±0.006	0.737±0.002	0.710±0.002

Table 3.2: Effect of ResNet-18 thumbnail encoder on caption generation on just the pathology notes section which aactually describes the image; mean/std of 3 runs

Model	**N BLEU-4**	**N ROUGE-L**	**N METEOR**
VIT	0.120±0.004	0.419±0.003	0.376±0.004
VIT+ResNet	0.130±0.004	0.424±0.003	0.391±0.003

We can see from results above in Tables 3.1 and 3.2 that adding the thumbnail encoder provides a sizable benefit to our caption generating model, especially when we compare performance on just the notes section, which is the most important part of the generated note. This shows that the added context of the whole thumbnail image is useful for report generation for our model.

In Figure 3.5 we show the confusion matrix for the tissue type classification of our best performing model from Table 3.1. To get the tissue type, we exploit the nature of our note construction, which has these connecting words like 'this is a' and 'from' to isolate the tissue type identified by the model during its caption generation.

We can see that the model is pretty good at recognizing where in the body tissue came from. The tissue types that it does get confused by, such as the different sections of the esophagus like 'gastroesophageal junction', 'mucosa' or 'muscularis' are relatively close in clinical nature. Of note is that there are 40 different types of tissues in our dataset and

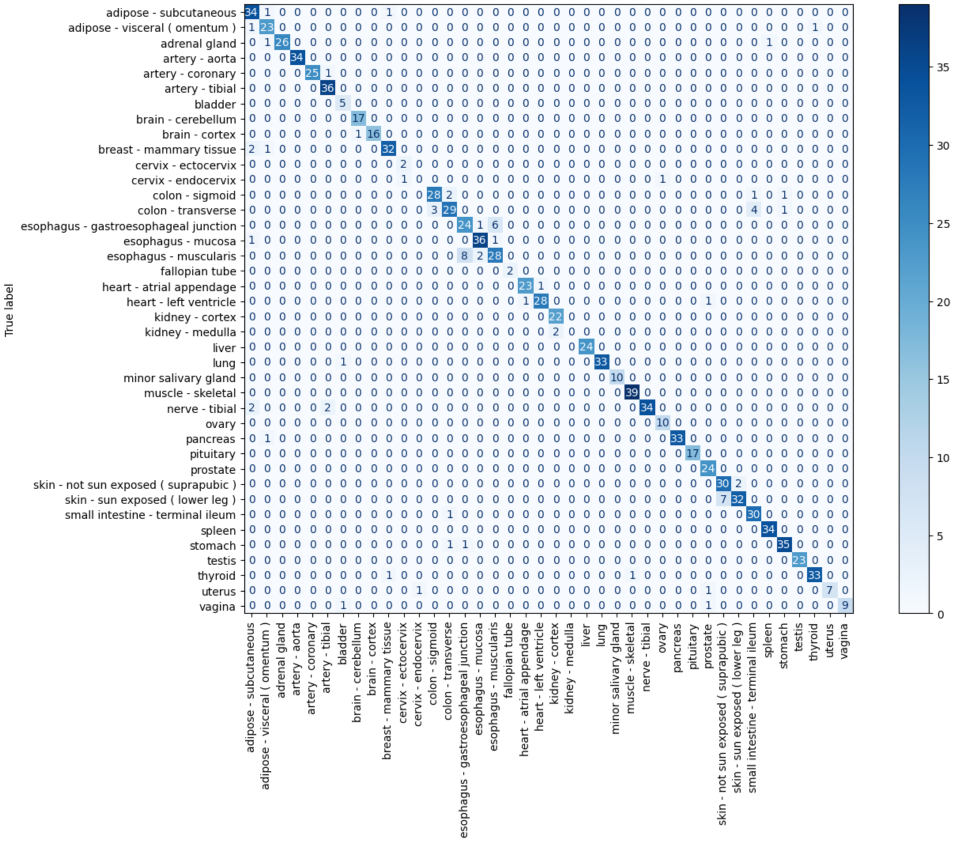


Figure 3.5: Confusion matrix for the tissue type classification performance of the best performing model from Table 3.1

based on the results in Figure 3.5, the model is pretty good at tissue type classification. This is of particular use in subtype classification problems.

3.6 Comparative Study of Histopathology Captioning Architectures

We described in Section 3.4, a method for building a captioning system using pre-trained vision encoders and decoders, that utilize the latest in the transformer based architectures currently. In this section, we compare the architecture presented here against different architectures proposed elsewhere [73] and by us [62, 63]. We hope to demonstrate the different techniques we developed for histopathology captioning and the merits and weaknesses of each.

For comparing against existing methods of histopathology image captioning presented recently, we choose the work by Tsuneki and Kanavati [73], due to its straightforward architecture and promising results. We briefly describe each architecture and then compare their performance on the actual captioning task.

3.6.1 Overview of architectures used for the comparison

CNN-RNN

Tsuneki and Kanavati [73] describe their method as follows:

"Given patches from a WSI, we extract features using a pre-trained CNN, which we then pool and reduce the dimensionality of. The caption is then generated by feeding the features into an RNN model step by step while updating its hidden state. In the case of the EfficientNetB3 model, given n patches from a given WSI, the output from the feature

extraction for a single patch was n × 10 × 10 × 1536. The global average pooling results in an output of n×1×1×1536, while the 3x3 pooling results in an output of n×3×3×1536. After the embedding dimensionality reduction, this becomes n × 1 × 1 × 256 and n × 3 × 3 × 256, respectively. These outputs are then flattened to become single dimensional vectors to be fed as input to the RNN model."

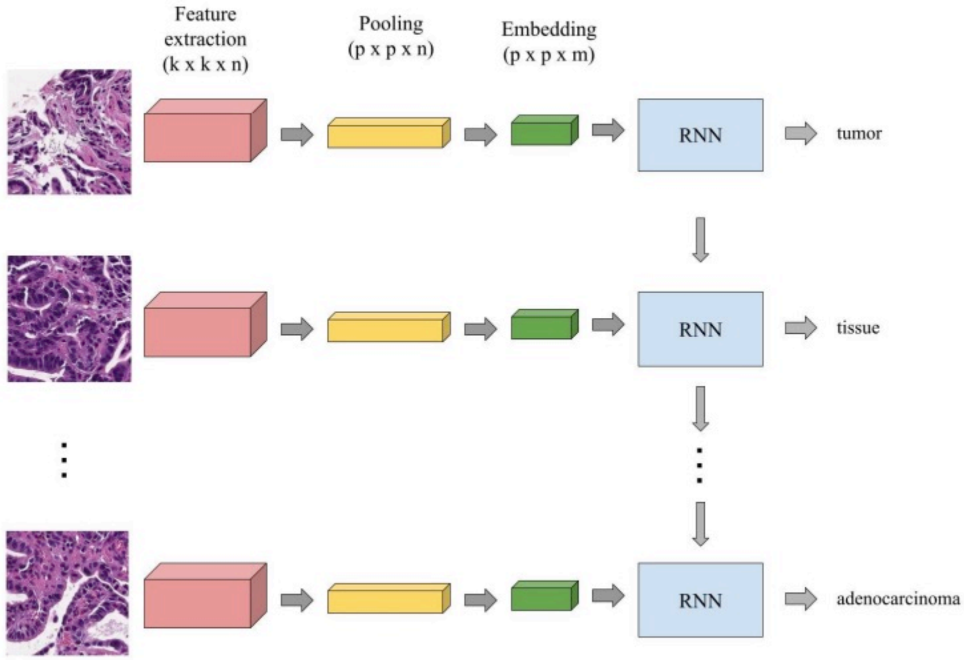


Figure 3.6: CNN-LSTM architecture from Tsuneki and Kanavati [73].

VIT-LSTM

We described in Section 3.4.1 a way to encode each WSI as a collection of M patch tokens such that WSI $= \{\mathrm{P}^1_{patch}, \mathrm{P}^2_{patch} ... \mathrm{P}^M_{patch}\}$ where each $\mathrm{P}^i_{patch} \in \mathbb{R}^{1\times576}$. The only thing different in this architecture is that we do not take the mean of $\mathrm{P}^i_{256} \in \mathbb{R}^{256\times384}$ and concatenating it with $\mathrm{P}^i_{4096} \in \mathbb{R}^{1\times192}$, to make $\mathrm{P}^i_{patch} \in \mathbb{R}^{1\times576}$, we instead broadcast $\mathrm{P}^i_{4096} \in \mathbb{R}^{1\times192}$ 256 times and concatenate it with $\mathrm{P}^i_{256} \in \mathbb{R}^{256\times384}$ to get $\mathrm{P}'^i_{patch} \in \mathbb{R}^{256\times576}$.

We also talked about using padding the tokens to make the input data uniform and an *'encoder_attention_mask'* that told the model which were the pad tokens to avoid training on them. In this architecture, instead of a mask, we use a trainable attention layer to generate a WSI representation that is a weighted representation of all the patch tokens, and then use that to feed into a LSTM based decoder as specified in the Show-Attend-Tell paper from Xu et al. [77].

$$WSI_{rep} = (\alpha \in \mathbb{R}^{1\times M}) \cdot (\mathrm{P}_{patch}^{vstack} \in \mathbb{R}^{M\times 256\times 576}) \rightarrow \mathrm{P}^{WSI} \in \mathbb{R}^{1\times 256\times 576}$$

where α are the weights generated from the attention layer.

We feed the WSI_{rep} into the attention-LSTM decoder to generate our captions. This method is illustrated in Figure 3.7

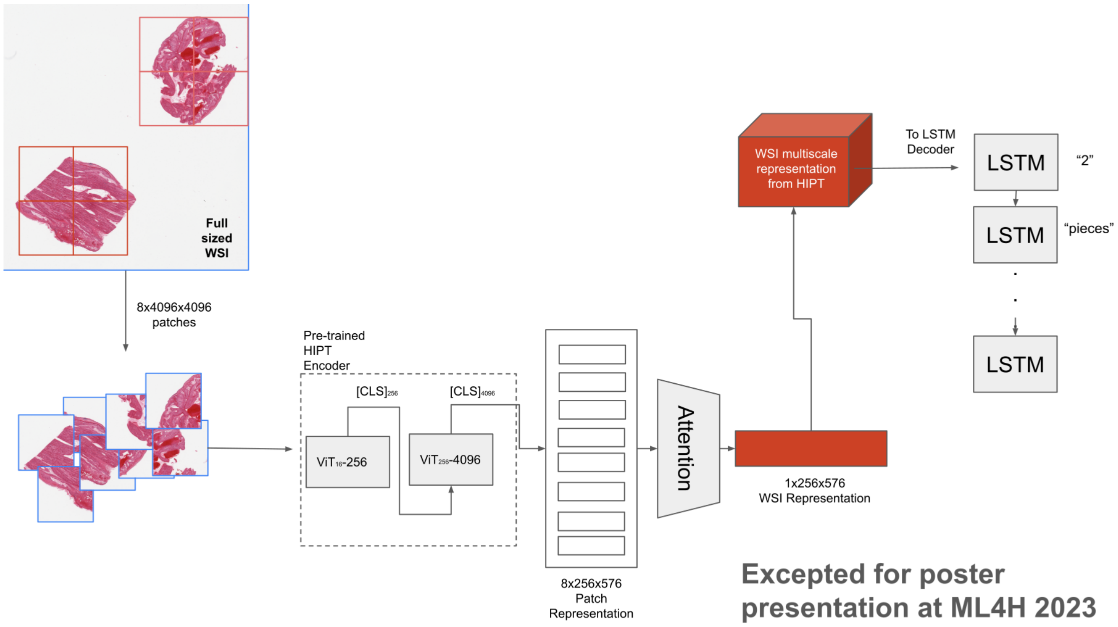
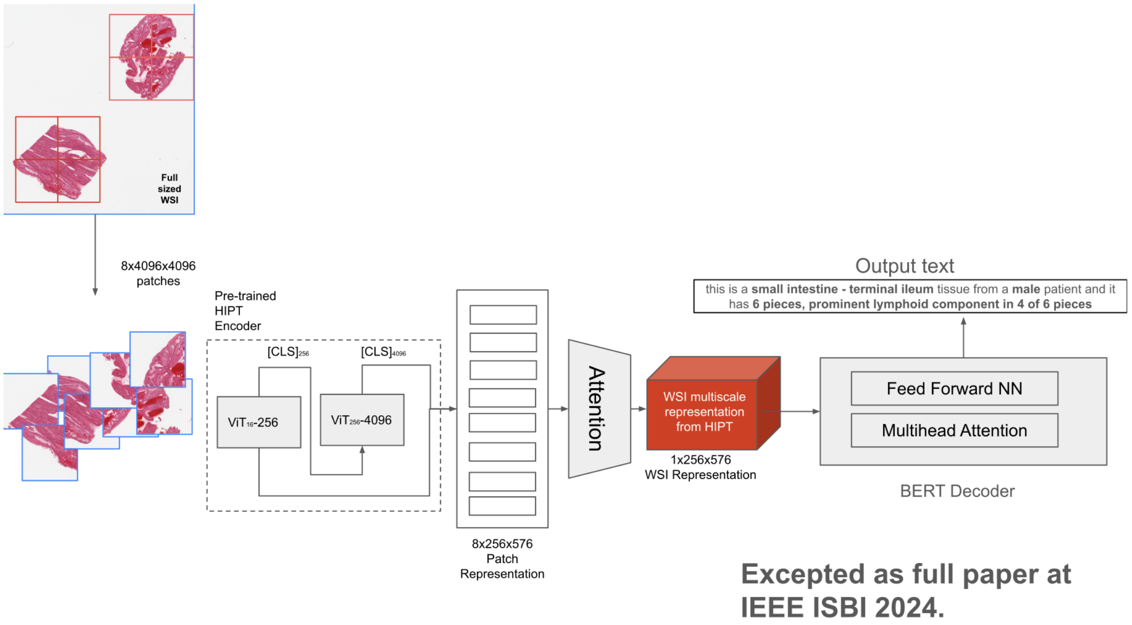

Full sized WSI
8x4096x4096 patches
Pre-trained HIPT Encoder
[CLS]256
[CLS]4096
ViT16-256
ViT256-4096
8x256x576 Patch Representation
Attention
1x256x576 WSI Representation
WSI multiscale representation from HIPT
To LSTM Decoder
LSTM
"2"
LSTM
"pieces"
LSTM
Excepted for poster presentation at ML4H 2023

Figure 3.7: VIT-LSTM architecture from Sengupta and Brown [62].



this is a **small intestine - terminal ileum** tissue from a **male** patient and it has **6 pieces, prominent lymphoid component in 4 of 6 pieces**

Excepted as full paper at IEEE ISBI 2024.

Figure 3.8: VIT-BERT architecture from Sengupta and Brown [63].

VIT-BERT

In this architecture, we simply replace the LSTM based decoder with a BERT based decoder, with added cross-attention-layers and a language modeling head [2]. This method is illustrated in Figure 3.8

VIT-BERT-ADV

This is the same architecture we described in this chapter in Section 3.4 without the thumbnail encoder to ensure that all models have a similar amount of information available for a fair comparison.

[2] https://huggingface.co/docs/transformers/model_doc/bert#transformers.BertLMHeadModel

3.6.2 Comparison with CNN-RNN

For our first study, we use a sample of 998 patients from our original cohort of 25,120 patients stratified by tissue type. We divide those into 718 training, 82 validation and 200 testing patients, again stratifying by tissue type. We reduce our training size because of the long training times required by the method described by Tsuneki and Kanavati [73] as we are using the same code as provided with the paper. Our training of one model with 718 training samples required 3hrs per epoch on an NVIDIA A100 GPU. Using all of our data would take prohibitively long. We train the CNN-RNN model for 10 epochs and report the result of the epoch with the best validation BLEU-4 score.

We show the results in Tables 3.3 and 3.4 for the full note and the pathology note section, respectively. We show the results on the smaller dataset and then the full dataset. Note that all the metrics shown are an average of 3 training runs, except for CNN-RNN [73].

Table 3.3: Comparison on the Full Note

Model	**Tissue Acc.(%)**	**BLEU-4**	**ROUGE_L**	**METEOR**
998 Patients				
CNN-RNN[73]	66.50	0.538	0.663	0.669
VIT-LSTM	**79.34**	**0.589**	**0.757**	**0.719**
VIT-BERT	72.00	0.532	0.719	0.667
VIT-BERT-ADV	71.50	0.521	0.710	0.667
All Data				
VIT-LSTM	**92.06**	**0.632**	**0.773**	**0.745**
VIT-BERT	89.53	0.578	0.742	0.703
VIT-BERT-ADV	89.86	0.565	0.737	0.703

We can see that the VIT-LSTM model outperforms other models in all metrics except for BLEU-4. This suggests that while the transformer based decoders are a more complex architecture, for our data, the LSTM based decoder is sufficient. Our hypothesis is that while LSTM based models might be sufficient for the length of sentences being considered in our data, which rarely exceed 128 tokens, real world pathology notes will contain multiple

Table 3.4: Comparison on the Pathology Note section

Model	BLEU-4	ROUGE_L	METEOR
998 Patients			
CNN-RNN[73]	0.044	0.277	0.242
VIT-LSTM	0.065	**0.416**	**0.348**
VIT-BERT	0.082	0.391	0.327
VIT-BERT-ADV	**0.088**	0.376	0.329
All Data			
VIT-LSTM	0.107	**0.448**	**0.392**
VIT-BERT	0.119	0.429	0.381
VIT-BERT-ADV	**0.120**	0.419	0.376

paragraphs. Since this would require retaining long term dependency information to generate coherent and accurate reports, we believe our transformer based decoders might outperform the LSTM based models.

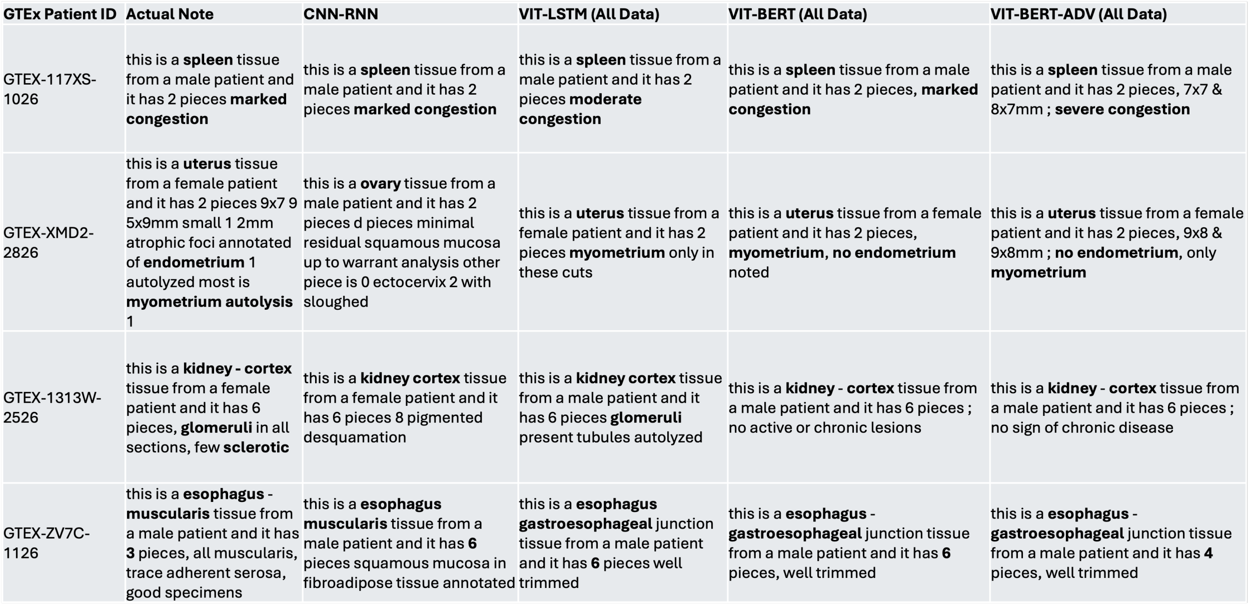
3.6.3 Qualitative caption analysis

In Figure 3.9 we show the generated captions for a selection of patients from our test set. Again the VIT-LSTM model seems to show better agreement with the actual pathology note, especially in the case of Patient ID: GTEX-1313W-2526, where the LSTM based decoder is able to detect 'glomeruli' while none of the others can.

3.7 Conclusion

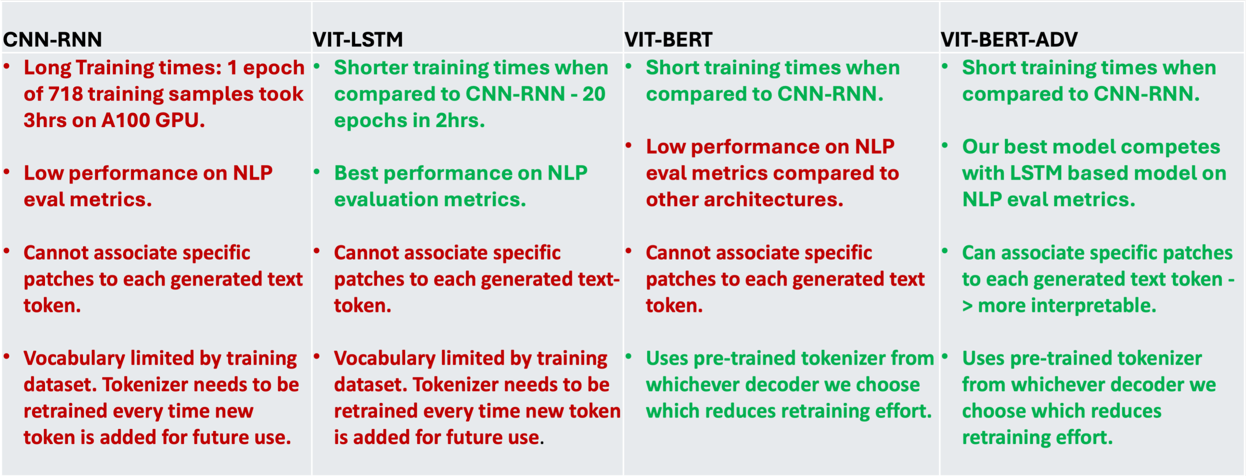
In this work, we show that powerful pre-trained ViT based representations could be used to encode a very high resolution histology image slide for another downstream task, that is, successful automatic report generation.

We also compared different architectures proposed by us and elsewhere for this task and showed that for our task, an LSTM based decoder outperformed our transformer based



GTEx Patient ID	Actual Note	CNN-RNN	VIT-LSTM (All Data)	VIT-BERT (All Data)	VIT-BERT-ADV (All Data)
GTEX-117XS-1026	this is a **spleen** tissue from a male patient and it has 2 pieces **marked congestion**	this is a **spleen** tissue from a male patient and it has 2 pieces **marked congestion**	this is a **spleen** tissue from a male patient and it has 2 pieces **moderate congestion**	this is a **spleen** tissue from a male patient and it has 2 pieces, **marked congestion**	this is a **spleen** tissue from a male patient and it has 2 pieces, 7x7 & 8x7mm ; **severe congestion**
GTEX-XMD2-2826	this is a **uterus** tissue from a female patient and it has 2 pieces 9x7 9 5x9mm small 1 2mm atrophic foci annotated of **endometrium** 1 autolyzed most is **myometrium autolysis** 1	this is a **ovary** tissue from a male patient and it has 2 pieces d pieces minimal residual squamous mucosa up to warrant analysis other piece is 0 ectocervix 2 with sloughed	this is a **uterus** tissue from a female patient and it has 2 pieces **myometrium** only in these cuts	this is a **uterus** tissue from a female patient and it has 2 pieces, **myometrium**, **no endometrium** noted	this is a **uterus** tissue from a female patient and it has 2 pieces, 9x8 & 9x8mm ; **no endometrium**, only **myometrium**
GTEX-1313W-2526	this is a **kidney - cortex** tissue from a female patient and it has 6 pieces, **glomeruli** in all sections, few **sclerotic**	this is a **kidney cortex** tissue from a female patient and it has 6 pieces 8 pigmented desquamation	this is a **kidney cortex** tissue from a male patient and it has 6 pieces **glomeruli** present tubules autolyzed	this is a **kidney** - **cortex** tissue from a male patient and it has 6 pieces ; no active or chronic lesions	this is a **kidney** - **cortex** tissue from a male patient and it has 6 pieces ; no sign of chronic disease
GTEX-ZV7C-1126	this is a **esophagus - muscularis** tissue from a male patient and it has **3** pieces, all muscularis, trace adherent serosa, good specimens	this is a **esophagus muscularis** tissue from a male patient and it has **6** pieces squamous mucosa in fibroadipose tissue annotated	this is a **esophagus gastroesophageal** junction tissue from a male patient and it has **6** pieces well trimmed	this is a **esophagus** - **gastroesophageal** junction tissue from a male patient and it has **6** pieces, well trimmed	this is a **esophagus** - **gastroesophageal** junction tissue from a male patient and it has **4** pieces, well trimmed

Figure 3.9: Qualitative analysis of captions



CNN-RNN	VIT-LSTM	VIT-BERT	VIT-BERT-ADV
• **Long Training times: 1 epoch of 718 training samples took 3hrs on A100 GPU.** • **Low performance on NLP eval metrics.** • **Cannot associate specific patches to each generated text token.** • **Vocabulary limited by training dataset. Tokenizer needs to be retrained every time new token is added for future use.**	• **Shorter training times when compared to CNN-RNN - 20 epochs in 2hrs.** • **Best performance on NLP evaluation metrics.** • **Cannot associate specific patches to each generated text-token.** • **Vocabulary limited by training dataset. Tokenizer needs to be retrained every time new token is added for future use.**	• **Short training times when compared to CNN-RNN.** • **Low performance on NLP eval metrics compared to other architectures.** • **Cannot associate specific patches to each generated text token.** • **Uses pre-trained tokenizer from whichever decoder we choose which reduces retraining effort.**	• **Short training times when compared to CNN-RNN.** • **Our best model competes with LSTM based model on NLP eval metrics.** • **Can associate specific patches to each generated text token -> more interpretable.** • **Uses pre-trained tokenizer from whichever decoder we choose which reduces retraining effort.**

Figure 3.10: Qualitative analysis of captions

decoders. However, since we only tested BERT pre-trained decoders and not more powerful transformer based models like ‘GPT2’ [50] or T5 [56], we would caution against discarding transformers completely. We present in Figure 3.10 an overview of the comparison between all methods presented in this paper.

Another downside of the methods we compare against in our comparative analysis in Section 3.6 is the lack of interpretability. We cannot associate the generated word tokens with areas of the image, thus reducing trust. We built the model described in this chapter specifically to be interpretable, which we detail in Chapter 4. Most previous methods for caption generation in histopathology required having a method to encode these high resolution images into manageable representations that could then be used for model learning, which can now successfully be replaced by powerful self-supervised pre-trained encoders like HIPT.

We also see that of the 40 different tissue types available in our test data, our model was able to correctly classify over 90% of them, which suggests that we can successfully use the captioning model for multi-class classification, which is a valuable objective. Even under-represented tissue types like ‘cervix’ and ‘fallopian tube’ were for the most part accurately classified. This points to the robust classification ability of language models grounded by visual features using the vision encoder and cross-attention.

We present, in this work, a method to utilize powerful pre-trained transformer models for automatic report generation specifically for histopathology with an end-to-end training mechanism that to the best of our knowledge had not been proposed before. We believe this work broadens the scope of research in histopathology by introducing transformers in place of traditional CNN-RNN based encoder-decoder models for histopathological caption generation.

Chapter 4

Evaluating cross attention for report generation

4.1 Introduction

In Chapter 3 of this dissertation, we described a method to build a caption generation pipeline that first converts high resolution WSIs into patches, exploits a powerful self-supervised model, HIPT [17] to generate patch embeddings, and then feeds it through a custom encoder-decoder model that generates descriptions of the whole slide image. One of the motivations of this dissertation was not only to combine multi-modal datasets like images and text for the histopathology, but also to investigate methods that help us investigate model decisions. While deep learning models are powerful, they are often black boxes which can hinder their trustworthiness. In clinical settings, in the event of an error, it is highly useful to have a mechanism to investigate model decisions.

Towards that goal, due to the nature of the caption generation architecture presented in Chapter 3, we are able to get cross-attention values for each word generated by the model. This provides us with a unique opportunity to glance at what part of the image the model was looking at for generating a word, thereby providing context in case of failure modes.

The main motivation of investigating cross-attention for model interpretation comes from Tang et al. [69], where the authors investigate Stable Diffusion models [57] by aggregating cross-attention maps in the denoising module of the Stable Diffusion model. Using that, they are able to generate text-image attribution maps, that help highlight sections of the image associated with words in the text prompt. Similar qualitative analysis of attention maps have been done by the authors of METransformer [76], which is a model to generate captions for X-ray images.

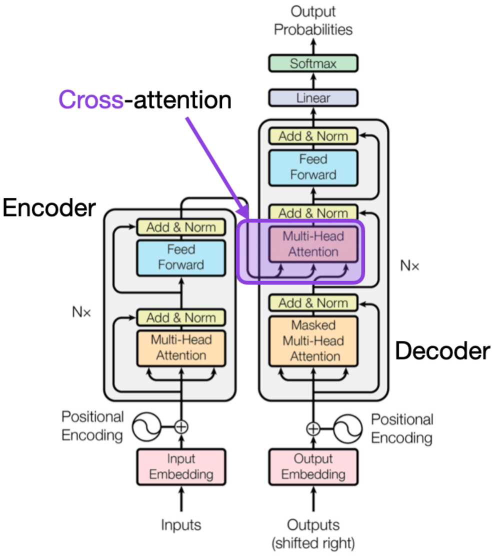
For histopathology caption generation, Zhang et al. [78] utilize attention maps generated from the attention LSTM decoder to highlight 256x256 patch clusters that were given importance for generating a particular word. While there have been other caption generation methods for histopathology, as described in Tsuneki and Kanavati [73] and Gamper and Rajpoot [30], the authors do not investigate attention attribution maps. This is mainly due to the high resolutions associated with WSIs which usually lead to compressing the WSI into a K-dimensional vector, mostly through aggregating patch representations, as described in the papers by Zhang et al. [78] and Tsuneki and Kanavati [73]. This compression often leads to loss of location information since once aggregated over patches, it is hard to assign importance to an area of the original high resolution image. Indeed, our previous work on histopathology image captioning also suffers from this loss of localization due to compression [62, 63].

Part of our motivation in building the architecture in Chapter 3 the way we did was to retain the ability to perform such analyses, since they can be potentially useful clinically. Our architecture retains the patch location information, since each patch is represented by a multi-scale embedding vector extracted from the HIPT which had been pre-trained in a self-supervised manner over other histopathology images. We do no aggregation over the patches to generate a WSI representation, as each WSI is now a sequential collection of visual tokens, with each token representing a 4096x4096 area of the image. While the patch size is relatively big, leading to some loss in fidelity of the generated attention maps, it is nevertheless useful for model validation.

In this section of the dissertation, we first describe the mechanism to extract cross-attention values for each generated word, then we show examples of our generated attention maps. We got the 10 of the attention maps validated by a board certified pathologist, who confirmed the existence of clinical features corresponding to the word being generated.

4.2 Cross Attention extraction

Our BERT-base transformer decoder consists of 12 transformer layers, where each transformer layer consists of a multi-head self-attention layer and a feed-forward neural network. The number of attention heads in each layer is 12. Due to it being used as a decoder, we also have a cross-attention layer between the self-attention and feed-forward layers. This is illustrated in Figure 4.1.



Source: "Attention Is All You Need" (https://arxiv.org/abs/1706.03762)

Figure 4.1: Cross-attention; source: Understanding Self-attention from scratch - Sebastian Raschka

Cross attention works by calculating the key K and value V vectors from the source input sequence $S1$, in this case the vision encoder outputs and the query Q from the target output sequence $S2$, which in this case is the output of the self-attention head in the decoder. Mathematically, it is done using the following:

$$CrossAttention = softmax(\frac{(W_q S_2)(W_k S_1)^T}{\sqrt{d_k}})(W_v S_1)$$

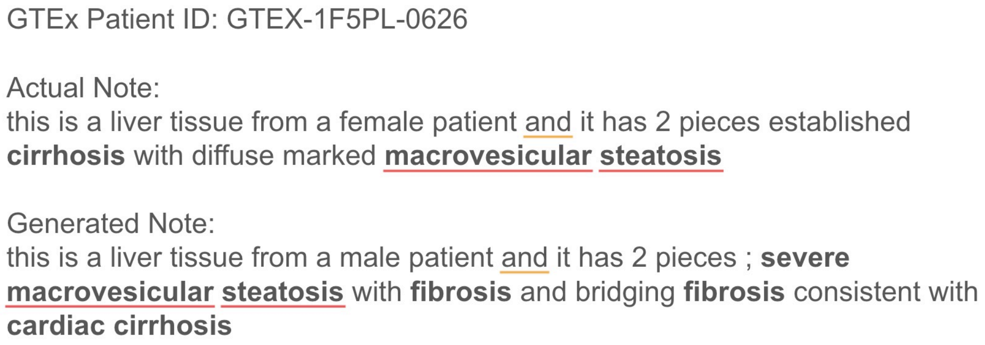
where W_q, W_k and W_v are learned weight matrices used to calculate the queries, keys and values respectively. Since there can be $n_{attention\_heads}$ attention heads and n_{layer}-layers, for each token generated we get $n_{layer} \times n_{attention\_heads} \times$ (number of visual tokens). For the case of BERT-base and its derivatives, n_{layer} and $n_{attention\_heads}$ are equal to 12.

For the sake of simplicity, we use greedy decoding during inference, which results in only the single most probable token generated based on previous tokens and encoded image. We do not use the beam search mechanism for token generation, which can often lead to k candidate tokens at every decode step, where k is the size of beam search.

For further simplification, we average over the number of heads per layer. Therefore,

$$cross\_attention(token, layer) = \frac{\sum_{i=0}^{n_{attention\_heads}} cross\_attention(token, i, layer)}{n_{attention\_heads}}$$

In the next section, we show outputs of visual attention maps associated with some chosen words in the generated caption for the cross-attention values from the final layer of the decoder.



GTEx Patient ID: GTEX-1F5PL-0626

Actual Note:
this is a liver tissue from a female patient and it has 2 pieces established **cirrhosis** with diffuse marked **macrovesicular steatosis**

Generated Note:
this is a liver tissue from a male patient and it has 2 pieces ; **severe macrovesicular steatosis** with **fibrosis** and bridging **fibrosis** consistent with **cardiac cirrhosis**

Figure 4.2: Comparison between the actual notes vs. the generated note. The patient is from the held out test set of 1000 patients.

4.3 Cross Attention Maps

In Figure 4.2, we show the caption generated for a patient from the held out set of 1000 patients described in Chapter 3. We can see that the generated caption closely follows the actual description from a pathologist. For the same patient, in Figure 4.3, we show the top 5 ranked patches by attention value, highlighted in blue shades, based on the cross attention values extracted from the last layer of the BERT decoder. The rest of the patches not in the top 5 are artificially set to an attention value of zero for simplicity and represented by highlighting them in red. You can find the code to generate these visualizations at https://github.com/ssen7/histo_cap_transformers_v2.

We can see that 'macrovesicular steatosis' generates similar attention maps, while the rest produce different attention maps. The word 'fibrosis' appears twice and generates different attention maps with some overlap each time. While the words 'cardiac' and 'cirrhosis' also generate differing attention maps.

When reviewed by a board certified pathologist, we were able to confirm the existence of image features present in the patches that were consistent with the associated token being generated.

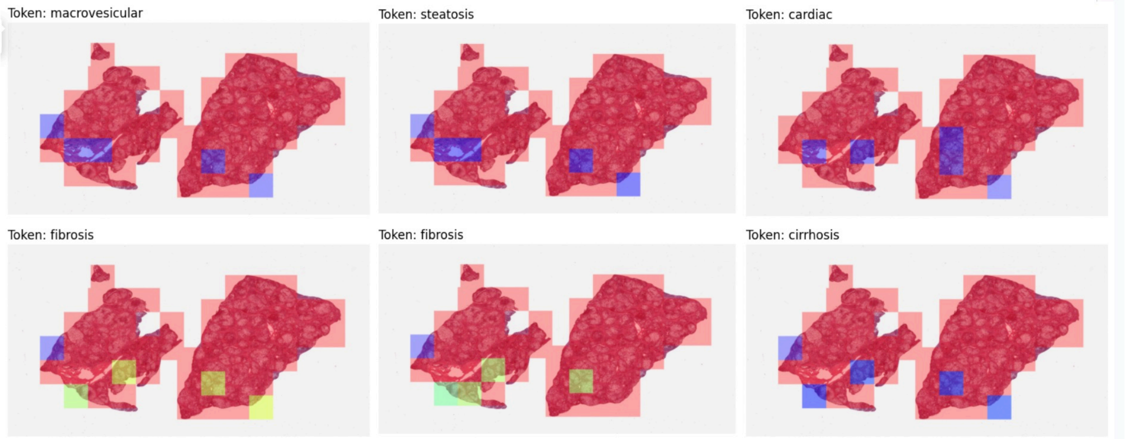


Figure 4.3: Each image shows the top 5 ranked patches by attention value, highlighted in blue shades, based on the cross attention values extracted from the last layer of the BERT decoder. Darker blue indicates higher attention value. We can see that 'macrovesicular steatosis' generates similar attention maps, while the rest produce different attention maps.

Next, in Figure 4.4 we show the top 5 ranked patches by attention value, highlighted in blue shades, based on the cross attention values extracted from the last 3 layers of the BERT decoder. The token of interest is 'squamous'. We can see from all the layers, the patches are ranked fairly similarly.

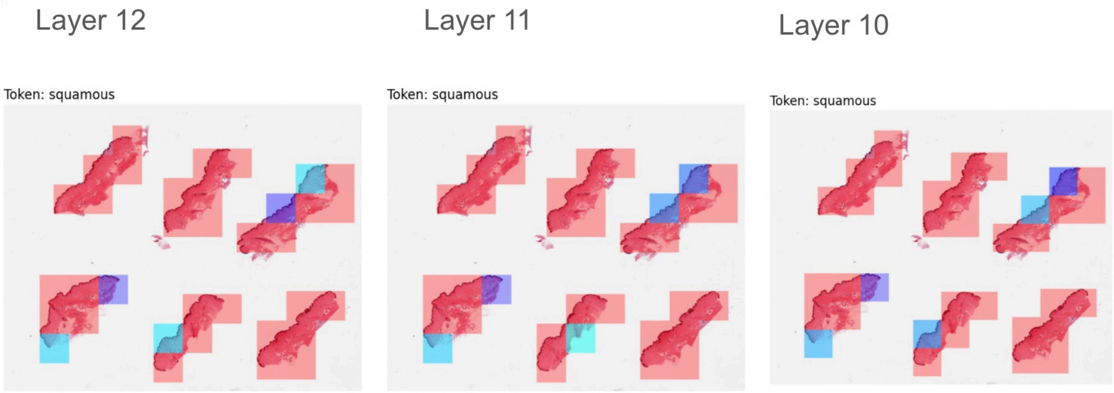


Figure 4.4: Each image shows the top 5 ranked patches by attention value, highlighted in blue shades, based on the cross attention values extracted from the last 3 layers of the BERT decoder. Darker blue indicates higher attention value. The token of interest is 'squamous'.

Next, in Figure 4.5 we show the top 5 ranked patches by cross-attention value from the final layer of the decoder. When reviewed by a pathologist, it showed evidence of 'squamous' epithelium, which are usually on the edge of tissues.

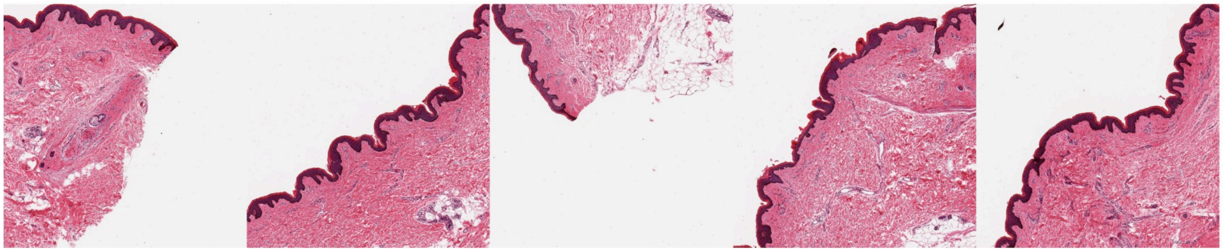


Figure 4.5: Each patch from highest ranked patches based on the cross-attention scores from the final layer of the decoder. Highest rank on the left.

4.4 Conclusion

This shows evidence that the model is giving importance to the correct things while generating associated tokens. We can also see that different layers of the decoder are giving attention to similar areas of the image, but in different magnitudes. This is promising since this validates the efficacy of the model architecture while also giving us a way to investigate the model in case of failure modes. To the best of our knowledge, this is the first work looking at cross-attention values for histopathology data and would set a starting point for more work in increasing the fidelity of these attention maps, so that we can one day associate smaller, more targeted areas of the image with each predicted token.

Chapter 5

Positive Unlabeled learning for detecting Long COVID

5.1 Introduction

Post-acute sequelae of SARS-CoV-2 infection (PASC), also known as Long COVID, is an emerging medical condition in the aftermath of the COVID-19 pandemic. Research on this disease is limited by its newness and the lack of reliable controls, which can hinder model development. The National COVID Cohort Collaborative (N3C)[1] contains Electronic Health Record (EHR) data for 7 million COVID positive patients from 76 sites across the United States, of which there are fifty thousand Long COVID patients. For this study, we model our risk factor analysis as Positive Unlabeled (PU) problem, where we treat Long COVID patients as the positive sample and rest of the COVID positive patients as unlabeled data. We first curate reliable controls using a PU modeling technique called bagging. We then use this cohort of positive and the curated negative samples to model risk factors for Long COVID. We utilize an attention-based deep learning approach using Long Short Term Memory (LSTM) networks on historical diagnosis data prior to COVID-19 infection, to first predict for Long COVID and then extract the model attention values to score input diagnoses for each patient.

[1] https://ncats.nih.gov/n3c

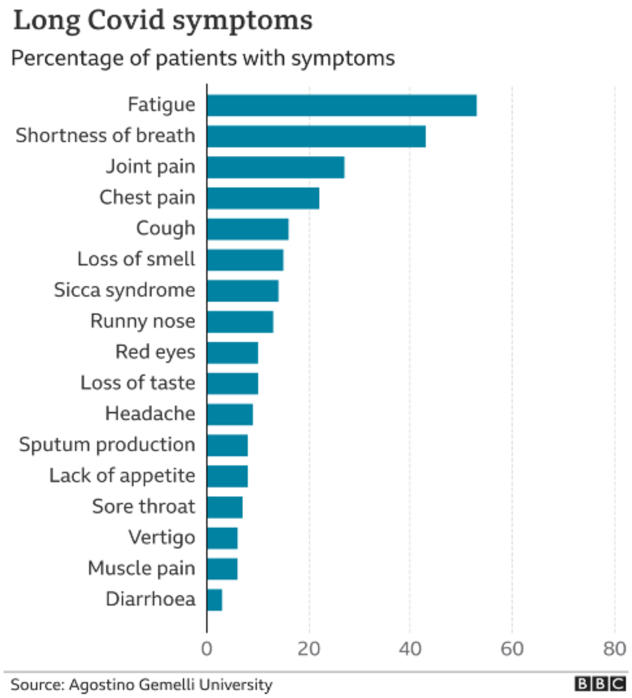


Figure 5.1: Long COVID common symptoms.

Long COVID is a newly developing condition that has affected people infected with the COVID-19 virus even after recovering from initial symptoms. It has been characterized by fatigue, dyspnea [20], loss of taste or smell, impaired concentration and memory problems [13], brain fog [5] and other symptoms that continue for weeks or months after the initial infection. Numerous studies have investigated risk factors associated with these Long COVID symptoms using retrospective analysis of survey responses [1] or follow up [28]. These risk factors include female sex [9], older age, hospitalization [28, 37] and vaccination status[6]. All of this motivates a deeper look into patient diagnosis data prior to and up to the COVID-19 pandemic that would make patients susceptible to suffer from Long COVID.

The N3C database contains diagnosis, medications, lab measurements data and more for 7 million COVID positive patients. While other studies have focused on data collected from questionnaires asking patients if they were suffering from Long COVID [6] which imparts a degree of reliability to the assigned labels, this process can be time-consuming and expensive if we want to use information from all patients we have in the database. This entails leveraging data from some patients labeled with a Long COVID diagnosis as well as selecting controls from a large set of COVID positive patients whom may not have been asked or had a Long COVID condition documented.

In recent years, self-supervised learning, which uses unlabeled data to inform downstream supervised learning, has shown good results in computer vision and natural language processing tasks [55]. Positive Unlabeled (PU) learning is a subset of self-supervised learning, that is, semi-supervised learning where the learner only has access to the positive samples and the rest of the data is unlabeled, which is similar to our case, where we can reliably identify patients suffering from Long COVID ($\mathcal{P}$) and the rest we cannot reliably label ($\mathcal{U}$) [10]. In this paper, we build on prior work and use attention based Long Short Term Memory (LSTM) models in the absence of labeled control data using a positive unlabeled learning technique called bagging to first identify a set of reliably negative control patients and then show that this performs better than a random selection of control patients. We then use attention scores generated for each correct prediction to rank the input diagnosis codes to help understand associations between them and Long COVID. We are also able to capture and plot the time distribution of the most common diagnosis codes.

5.2 Related Work

5.2.1 Deep Learning for Electronic Health Records

Numerous previous studies have investigated the use of deep learning for analyzing Electronic Health Record (EHR) data. REverse Time AttentIoN model (RETAIN) for application to Electronic Health Records (EHR) data [18] used Recurrent Neural Networks (RNNs) to analyze time labeled sequence of observations like diagnosis codes where each patient is represented by a sequence of temporally arranged tuples, while also being interpretable. The model was able to assign a feature importance score to each input feature using outputs from its two constituent RNN networks for calculating attention at both visit level and variable level. One of the earliest papers used plain LSTMs on temporally arranged International Classification of Diseases-9th revision (ICD-9) diagnosis codes to predict 128 common diseases like heart failure and respiratory distress [47]. Similar studies using LSTMs for analyzing temporal diagnosis codes, vital signs and medications for future disease onset prediction have also been shown to perform well [21, 71]. These studies motivate our goal of analyzing historical diagnosis codes prior to the COVID-19 infection present in the N3C database to first predict if a patient is suffering from Long COVID, and then investigate these models for associations between a patient's diagnosis history and Long COVID.

5.2.2 Positive Unlabeled Learning

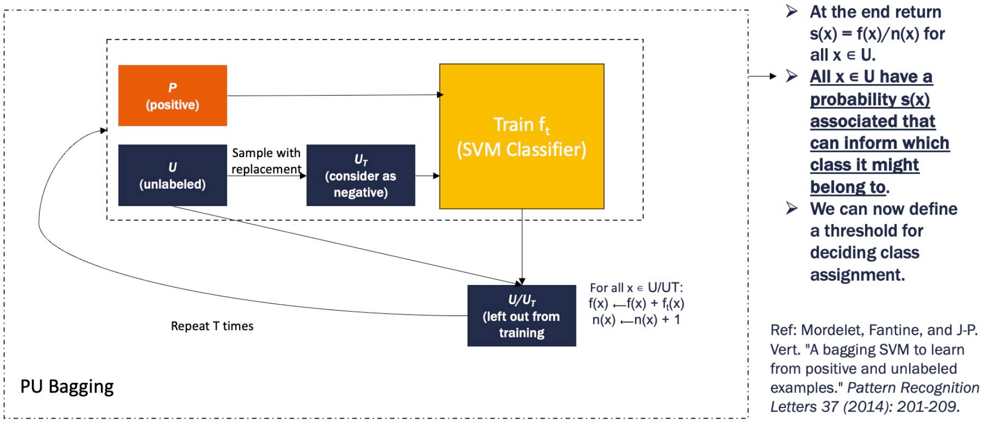
There are several strategies used for PU learning. In [48], a two-step technique is used where the first step is generating reliable negative samples from the unlabeled dataset, and using a supervised learning method using the positives and the reliable negative samples. Various techniques are used for generating reliable negatives, like Spy, where some labeled samples are turned in spies by adding them to the unlabeled samples and the reliable negative samples are assigned to all samples whose posterior probability is lower than the posterior probability of the spies. The second step utilizes supervised classification algorithms like Support Vector Machine (SVM) to build robust classifiers. The other class of positive unlabeled methods broadly treat an unlabeled dataset as noisy negative samples, that is, contaminated with positive samples which can make learning harder [61]. Bagging SVMs train multiple SVM models by taking the positives and treating a random sample of the unlabeled as negatives, and then averaging their predictions [49]. Robust ensemble SVMs follow the same idea, but randomly sample the positives and use the bootstrap approach [19]. Other techniques involved defining loss functions that minimize the non-negative risk [38].

5.2.3 PU Learning Assumptions–Selected Completely At Random

There are a number of assumptions that enable us to learn from positive unlabeled datasets. The most common among them is Selected Completely At Random (SCAR) where we assume that all labeled samples are selected from a distribution of positive samples [27]. Under this assumption, the probability for a sample to be labeled is directly proportional to the probability of it being positive. Therefore, it reduces the PU learning problem into a binary classification model where if we train an estimator to classify between positive and unlabeled samples, the estimator in theory should also be able to discriminate between positive and negative samples [49]. This enables us to train classifiers like biased-SVM where high penalties are given for false negatives and low penalties for false positives or a weighted logistic regression approach, since we know that the unlabeled samples are contaminated with positive samples.

5.2.4 Bagging for PU Learning

Bootstrap aggregating or Bagging creates a set of classifiers trained on a perturbed sample of the training data, and aggregating their predictions has also been successfully used for PU learning [49]. In the inductive setting, where the goal is to make predictions on new or unseen data, this method which is also called SVM bagging, takes T-samples of size K from the unlabeled dataset $\mathcal{U}$, trains T-classifiers by treating the sample from the unlabeled dataset as negatives. For previously unseen data, we can simply average the predictions from all T models. In the transductive setting, where the goal is to identify positive data from



- At the end return s(x) = f(x)/n(x) for all x ∈ U.
- **All x ∈ U have a probability s(x) associated that can inform which class it might belong to.**
- We can now define a threshold for deciding class assignment.

Ref: Mordelet, Fantine, and J-P. Vert. "A bagging SVM to learn from positive and unlabeled examples." *Pattern Recognition Letters 37 (2014): 201-209.*

Figure 5.2: Illustration of the SVM Bagging technique as described by Mordelet and Vert [49]

the unlabeled dataset, models are trained using subsample $\mathcal{U}_T$ from the unlabeled sample $\mathcal{U}$, is used to assign prediction scores on all samples $\mathcal{U} \backslash \mathcal{U}_T$ held out from the training. For all data points in the unlabeled sample, the prediction scores are aggregated, and a threshold is chosen, above which each data point is assigned a positive label. The authors also found that having the $K = size(\mathcal{P})$ was a safe choice for bagging predictors.

5.2.5 Attention based neural networks

Attention in neural networks has been previously used in Neural Machine Translation [7] and Image captioning [77] where it has been successfully used to "attend" to relevant context in the input sentence and image respectively. It is essentially used to "look" at the most useful feature of a given input to generate a response. In machine translation it is used for exploiting relationships between sequences of words in the input sentence, while in image captioning it is used to focus on the area of image most relevant for generating the next word in the caption that best describes the image. There are many flavors of attention, like deterministic "soft" or stochastic "hard" attention. Here, we focus on the simpler and differentiable "soft" attention mechanism that we use in our model.

In practice, soft attention is essentially a trainable linear layer that calculates the weights on the annotation h_i generated by the previous LSTM layers. From [7]: Each annotation h_i contains information about the whole input sequence, with a strong focus on the parts surrounding the i-th word of the input sequence. The context vector c_i is, then, computed as a weighted sum of these annotations h_i as $c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j$. The weight α_{ij} of each annotation h_j is computed by $\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{T_x} \exp(e_{ik})}$ where $e_{ij} = a(s_{i-1}, h_j)$ is an alignment model, which scores how well the inputs around position j and the output at position i match.

5.2.6 Attention for interpretation

Since attention provides a weighted average on representations of the input features, it has often been used to assign relative importance of these inputs for the prediction, and has found particular use in medical settings where model reliability and interpretability is of the utmost importance [18]. There has been much debate around whether attention values should be used for explanations [12]. In [31], the authors find that there can be meaningful interpretations on the relative importance of input features in classification settings and there is a higher chance of the interpretations being meaningful if the model itself is robust. The authors recommend using ensembling methods to improve those interpretations in case of failure modes where the model itself is good, but the attention values do not hold significant information.

5.3 Dataset

Please refer to Table 5.1 for the inclusion criteria for patient selection. These criteria were designed to focus the analysis on adult patients who are at sites with at least some patients having coded labels of Long COVID in our dataset (visit at a Long COVID specialty clinic or

Long COVID diagnosis U09.9). We also only want to include patients whose COVID index occurred early enough to allow for the development of Long COVID. We define the COVID-19 index event as the first time the patient tests positive for COVID-19 or a diagnosis of COVID-19 was recorded in the EHR.

Table 5.1: Dataset Inclusion Criteria

Criteria	**N**
Total N3C patients (June 14th 2023)	19,376,170
Total COVID positive Patients (June 14th 2023)	7,578,139
Age at COVID>=18	5,627,459
Patients from data partners sending Long COVID clinic visit flags and/or U09.9 codes	4,711,807
Remove patients who died within 45 days of COVID diagnosis	4,659,781
Keep patients with COVID index date between >=2020-03-01 and <=2022-12-01	4,464,134

5.4 Methodology

5.4.1 Site Based Filtering

For each Long COVID patient:

1. find controls that are from the same site.
2. look for controls that have an index date within ± 45 days or patient's COVID index date.
3. take a sample of up to 10 random controls.

To create a list of potential controls, for each Long COVID patient, we randomly sample 5 of these random controls to create 1:5 case vs control patients. We define a Long COVID patient as anyone who has the U09.9 ICD-10 code in their EHR or has a Long COVID clinic visit. At the end of this process we have 50,240 Long COVID patients and 251,200 patients who are potential controls, and we treat as our unlabeled dataset.

Table 5.2 contains the demographic breakdown of our study cohort. We see that our Long COVID cohort has a higher median age than those chosen by our site filtering method for controls. We also see that female patients are a higher percent of our Long COVID patients, which matches what other studies have found regarding Long COVID.

Table 5.2: Breakdown of Demographics After Site Based Filtering

	All patients 301,440	**Long COVID** 50,240	**Unlabeled** 251,200
Age			
Median (Q1,Q3)	48 (33,62)	54 (42,65)	46 (32,61))
Race/ethnicity			
White	189,758 (63.0%)	34,005 (67.7%)	155,753 (62.0%)
Hispanic	37,927 (12.6%)	5,634 (11.2%)	32,293 (12.9%)
Black/African American	42,252 (14.0%)	6,732 (13.4%)	35,520 (14.1%)
Asian	6,288 (2.1%)	870 (1.7%)	5,418 (2.2%)
Other	5,204 (1.70%)	536 (2.1%)	4,668 (1.9%)
Unknown	20,011 (6.6%)	2,463 (4.9%)	17,548 (7.0%)
Sex			
Male	124,174 (41.3%)	17,371 (34.6%)	106,803 (42.5%)
Female	177,076 (58.7%)	32,854 (65.4%)	144,222 (57.5%)

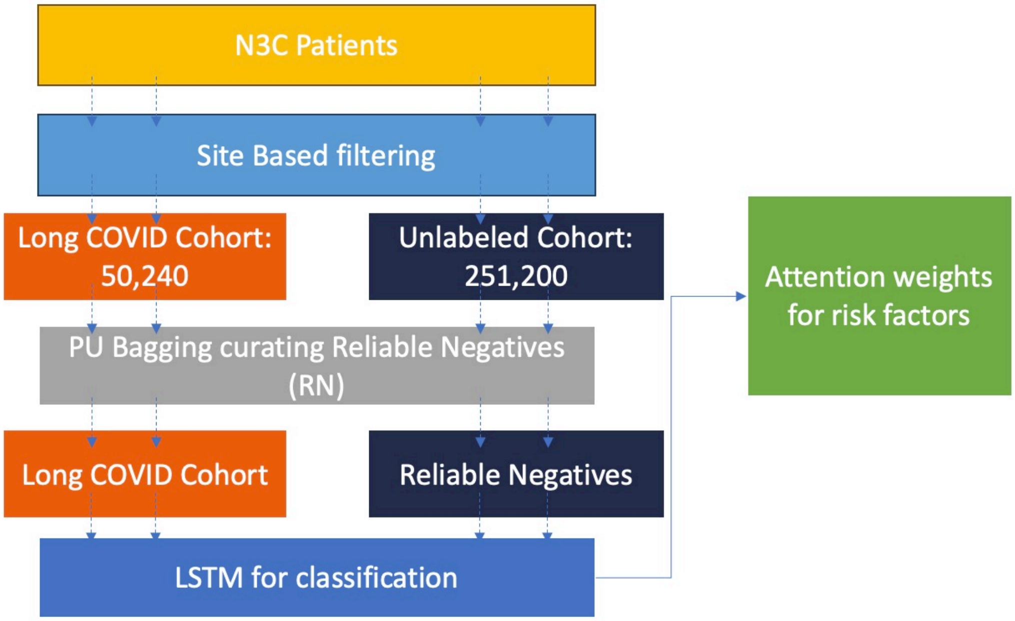

N3C Patients
Site Based filtering
Long COVID Cohort: 50,240
Unlabeled Cohort: 251,200
PU Bagging curating Reliable Negatives (RN)
Long COVID Cohort
Reliable Negatives
LSTM for classification
Attention weights for risk factors

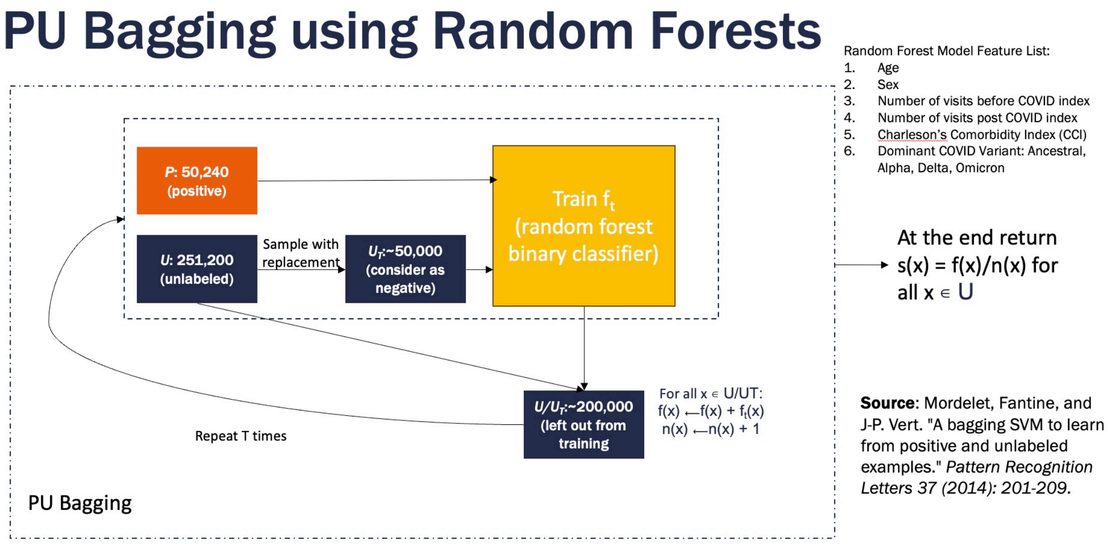
Figure 5.3: Positive Unlabeled learning for Long COVID detection, a method overview

5.4.2 RF Bagging method

To curate the probable negatives from the unlabeled sample, $\mathcal{U}$ we use the following process:

1. Sampling with replacement from the unlabeled dataset $\mathcal{U}$ to create a set of “negatives” $\mathcal{U}_T$ and combine them with all our positive samples $\mathcal{P}$ to create multiple training datasets $\mathcal{D}_T = \mathcal{P} \cup \mathcal{U}_T$. Note that these “negatives” $\mathcal{U}_T$ can contain positive samples. Also, each sample size K roughly matches the size of positive samples as found by [49]. For each such “training” dataset created, the rest of the unlabeled samples $\mathcal{U} \backslash \mathcal{U}_T$ are used for prediction.

2. Training a Random Forest model $\mathcal{M}_T$ on each $\mathcal{D}_T$ using patient features we know from previous studies to be correlated with a Long COVID diagnosis, namely age, sex, the total number of visit days the patient was recorded having prior and after to their



PU Bagging using Random Forests

Random Forest Model Feature List:

1. Age
2. Sex
3. Number of visits before COVID index
4. Number of visits post COVID index
5. Charleson's Comorbidity Index (CCI)
6. Dominant COVID Variant: Ancestral, Alpha, Delta, Omicron

P: 50,240 (positive)

U: 251,200 (unlabeled)

Sample with replacement

U_T:~50,000 (consider as negative)

Train f_t (random forest binary classifier)

U/U_T:~200,000 (left out from training

For all x ∈ U/UT:
$f(x) \longleftarrow f(x) + f_t(x)$
$n(x) \longleftarrow n(x) + 1$

Repeat T times

PU Bagging

At the end return s(x) = f(x)/n(x) for all x ∈ U

Source: Mordelet, Fantine, and J-P. Vert. "A bagging SVM to learn from positive and unlabeled examples." *Pattern Recognition Letters 37 (2014): 201-209.*

Figure 5.4: RF Bagging technique overview.

COVID index event visit (with hospitalizations collapsed to a single visit), and the patients overall comorbidity profile as represented by their Charleson's Comorbidity Index score at the time of COVID index event. We chose a Random Forest for this task as it outperformed other classification models like Logistic Regression using Area Under the ROC Curve metric and has a lower chance of overfitting on the data.

3. For each trained model $\mathcal{M}_T$, we assign probabilities $Pr_{S_i}(y = 1)$ to the unlabeled samples, $\mathcal{U} \backslash \mathcal{U}_T$ where S represents the number of times a sample was held out from training.

4. Aggregate the predictions probabilities $\frac{\sum_{i=1}^{S} Pr_i(y=1)}{S}$ for all the unlabeled samples. Note that this is similar to bootstrap aggregating or bagging technique defined used in [49].

5.4.3 Choosing the threshold

Since we now have aggregated probabilities for all the unlabeled data we have, we choose an appropriate threshold below which we can reasonably assume that the samples are reliably negative, thereby reducing the noise in our unlabeled dataset. The following criteria is applied for choosing the threshold:

1. Use all patients whose average prediction <0.50 to create our reliably negative population RN.

2. If $size(RN) > size(\mathcal{P})$, keep on reducing the threshold until $size(RN) \approxeq size(\mathcal{P})$.

Here, we prioritize creating a balanced sample of our reliably negative population.

5.4.4 Prediction Task

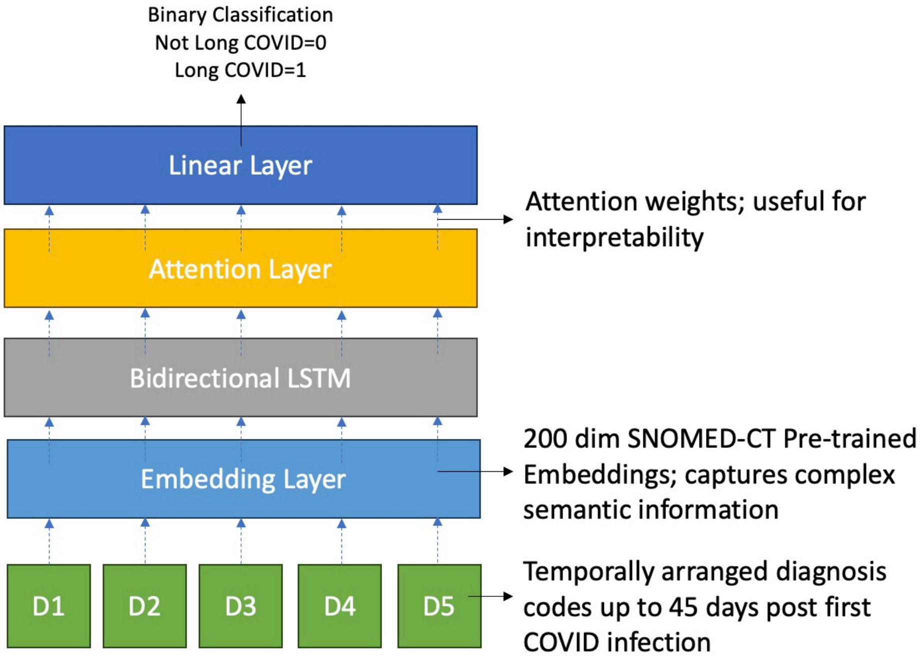

Binary Classification
Not Long COVID=0
Long COVID=1
Linear Layer
Attention weights; useful for interpretability
Attention Layer
Bidirectional LSTM
200 dim SNOMED-CT Pre-trained Embeddings; captures complex semantic information
Embedding Layer
D1
D2
D3
D4
D5
Temporally arranged diagnosis codes up to 45 days post first COVID infection

Figure 5.5: Bidirectional LSTM model with an attention layer used for classification.

Our second step involves using our positives and the reliable negatives identified in step 1 to train our Bidirectional LSTM with Attention model for binary classification of Long COVID, with all Long COVID patients assigned a label of 1 and the reliably negative patients assigned a label 0. Each patient P_i is represented as a temporally ordered list of diagnosis codes, $[C_0, C_1, ...C_{T_x}]$ where T_x represents the time step at which the diagnosis occurred. We limit the max length of input features to 1000 tokens as it covers 99 percent of the Long COVID patients in our study cohort. We collect all diagnoses up to 45 days after the COVID-19 index event. We have access to a Limited Dataset (LDS) as defined here https://ncats.nih.gov/n3c/about/data-overview, and therefore have access to actual dates on which each diagnosis occurred.

The embedding layer is initialized using 200-dimensional pre-trained SNOMED-CT embeddings trained using Snomed2Vec [2] and frozen during training. We identified 16,396 unique diagnosis codes, of which we found pre-trained embeddings for 15,563 codes and treat the rest as unknown tokens assigned a special [‘UNK’] token. Our model consists of 2 bidirectional LSTM layers followed by an attention layer and a final classification linear layer. We use the Cross Entropy loss and Adam optimizer for training, with a learning rate of 0.001. We can then extract a list of α_{ij} scores for each input diagnosis code C_i. We show through experiments that this trained classifier outperforms the same model trained on a randomly chosen subsample of unlabeled data as negatives.

5.5 Experiments

5.5.1 Baseline Comparison

For baseline models, we use Naïve Bayes and Support Vector Machines (SVM) in a Bag of Words (BoW) setting, where each patient feature set is defined by the count of times a particular diagnosis occurs in their patient record. We chose the BoW setting because it satisfies our stated goal of analyzing all historical diagnosis codes for analysis. We compare the performance of these models against the neural network models. Our experiments show that while neural networks outperform these baseline methods, model performance is still not adequate. Note that these are trained on all data with 1:5 train to unlabeled proportion, making it an imbalanced dataset. All performance measures are on the same held out test set. See Table 5.3.

Table 5.3: Baseline Comparison

Model	AUC	Accuracy	F1 Score
Naïve Bayes BoW Model	0.50	0.22	0.29
SVM BoW Model	0.55	0.83	0.18
BiLSTM CNN Model [64]	0.75	0.66	0.44
BiLSTM Attention Model	0.80	0.74	0.49

We see that the BiLSTM-Attention Model performs better than the others, and we use this model for our experiments from here on out.

5.5.2 Balanced Random Control Sampling

Here we sought to balance the input dataset such that we have 1:1 ratio of positive and unlabeled data. We randomly sampled 3 times from our unlabeled data, $\mathcal{U}$ and assigned them a label of 0 and used the BiLSTM Attention model for classification. See Table 5.4 for results.

Table 5.4: 3 Controls randomly drawn from unlabeled samples $\mathcal{U}$

Model	AUC	Accuracy	F1 Score
BiLSTM Attention Random Balanced v1	0.77	0.71	0.73
BiLSTM Attention Random Balanced v2	0.79	0.71	0.72
BiLSTM Attention Random Balanced v3	0.84	0.76	0.77

We can see that it is random luck of the draw to build a performant model and all performance metrics are highly dependent on the unlabeled sample we chose as negatives.

5.5.3 Balanced Pruned Controls with Different Bagging predictors

Here we show the results of our two-step technique, where we first use T predictors to find our reliable control samples using the method described above and then run our best performing neural network model on the dataset created using all our positive samples and reliable controls found in the first step.

Applying our method for pruning controls results in consistently improved model performance. Table 5.5 shows the effect of the number of times T we train a model for bagging predictions of our reliably negative controls. As expected, as we increase the number of bagging predictors, we get better at finding reliably negative patients and are able to build better models.

Table 5.5: Controls Pruned with Different Bagging predictors

Number of predictors	**AUC**	**Accuracy**	**F1 Score**
T=3	0.87	0.80	0.81
T=4	0.92	0.84	0.86
T=5	0.93	0.86	0.88

5.6 Analysis

In [31], the authors propose that using an ensemble of models improves robustness of attention based interpretations. They propose two flavors of this, where they train multiple models with the same hyperparameters with different seeds or take an ensemble of the top performing models. In our case, we train the same model with different seeds, produce attention scores for the input features for every correctly classified Long COVID patient from each model and average the scores.

We first report the performance metrics for models trained with different seeds in Table 5.6. We see that our chosen model is stable even when we have different initialization weights due to a different choice of the random seed.

Table 5.6: Performance of models used for attention scoring

Model	**AUC**	**Accuracy**	**F1 Score**
random seed v1	0.93	0.86	0.88
random seed v2	0.94	0.87	0.88
random seed v3	0.93	0.87	0.89

We are then able to extract the attention scores for all correctly classified Long COVID patients from all 3 models, normalize them, and then average them. We then extract the top 10 features by average attention score for these patients. Since we know when these diagnoses were recorded, we can capture the distance in days from the first COVID index event.

Figures 5.6 and 5.7 contains the top 10 features identified by the average normalized attention scores for two correctly classified Long COVID patients, while Figure 5.8 contains the top 10 features summarized over all patients correctly classified by the model as having Long COVID.

We see in Figure 5.8a that "Mixed Hyperlipidemia" and "Essential Hypertension" are the top 2 conditions that occur in the 10 most highly scored diagnoses for correctly classified Long COVID patients [32, 67]. We also see unknown token (<unk >) show up in the top 10 conditions and when we look at the actual condition recorded in the EHR, we see that most of those unknowns are "Platelet count-finding", which comes before "Platelet count-abnormal" and ""Platelet count-normal" in the SNOMED-CT hierarchy. This points to a

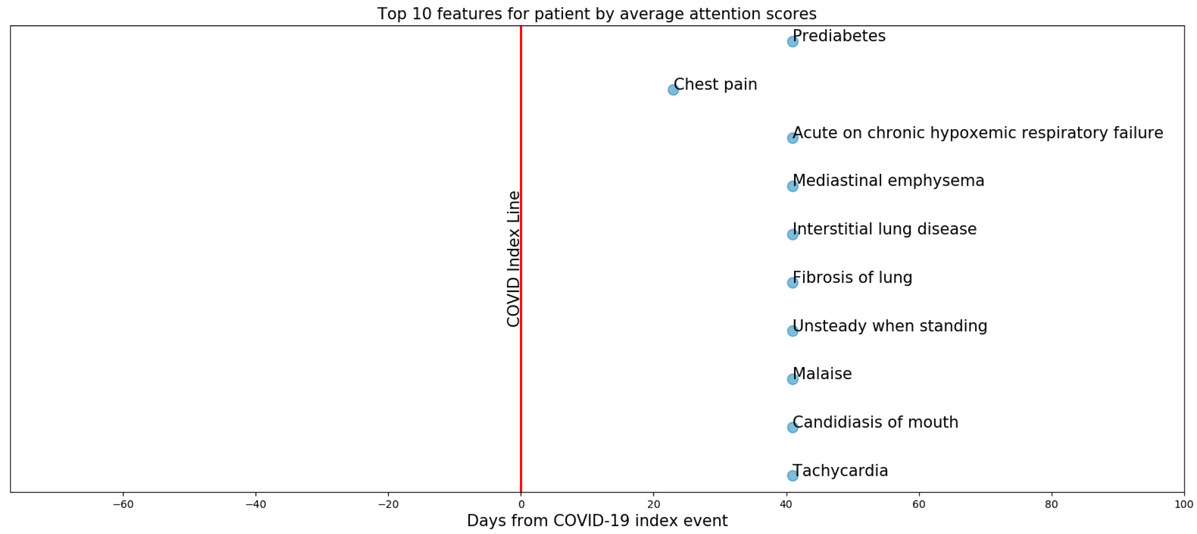


Figure 5.6: Top 10 Features for a correctly classified Long COVID patient identified using average attention scores.

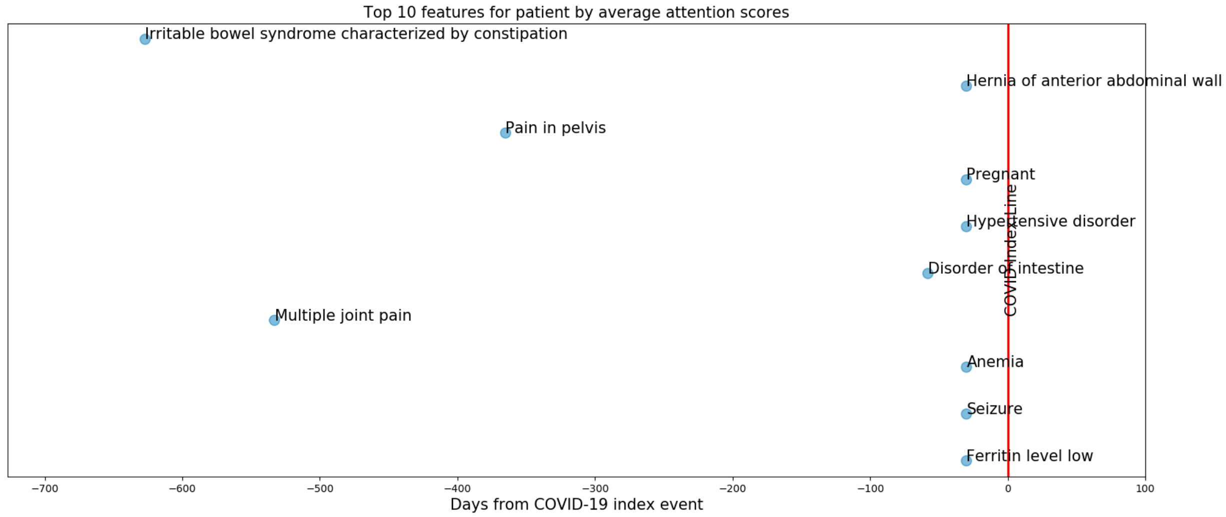
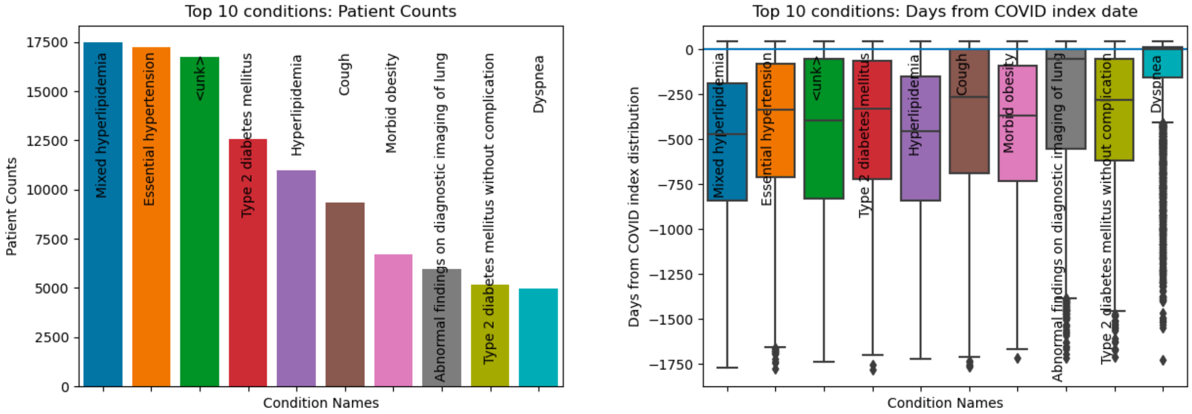


Figure 5.7: Top 10 Features for a correctly classified Long COVID patient identified using average attention scores.



(a) Top 10 conditions:patient counts

(b) Day distribution from COVID index

Figure 5.8: Top 10 attention score distribution

limitation of analyzing EHR data, where recording practices in hospital systems are often oriented towards billing. This finding will also guide our future work, where instead of replacing tokens with unavailable pre-trained embeddings with a special token, we will work towards identifying the next closest embedding for the token in the SNOMED-CT hierarchy.

In Figure 5.8b, we can see the distribution in number of days before or after the COVID index event the diagnosis occurred. We see that "Dyspnea" usually is recorded closest to and around the COVID index event, which suggests that having difficulty breathing around the COVID-19 infection can be a significant risk factor for Long COVID.

In Figures 5.6 and 5.7 we see the top 10 features by average attention scores for 2 different patients as identified by the models, plotted over how far in days from the COVID index event these were recorded in the EHR. For Patient 1 in 5.6 all the diagnoses occur within 45 days of the COVID index event while the opposite is true for Patient 2 in 5.7, although most of the highly scored features are observed nearer to the COVID-19 index event. These visualizations can be useful for physician oversight into the trustworthiness of the model's predictions and offer insight into individual patient characteristics.

5.7 Conclusion

In this chapter, we described utilizing a Positive Unlabeled learning based bagging technique to curate reliable negative samples when finding such samples from large, unlabeled data can be expensive and time-consuming to do. We then show that we are able to build effective models when we consider these negative samples for our modeling. We propose that this technique will potentially be useful in further analyses which utilize data from multiple institutions, such as in the N3C, where labeled data can be hard to come by. We also show the applicability of using attention based neural network methods and the interpretability they offer when applied to EHR data collected from multiple hospital sites from across the United States, and hope this establishes a foundation for further research using state of the art neural network models on the N3C platform.

Chapter 6

Conclusion and future work

We had a number of goals that we hoped to achieve with the works described in this dissertation. One of the foremost among them was to build a multi-modal system for healthcare, especially in the histopathology domain. As described before, histopathology datasets are challenging to deal with because they require workarounds for computationally handling their large image sizes. Therefore, building multi-modal models, require a number of considerations to handle this issue of size.

We described, in Chapter 3, an image captioning system that can automatically generate reports given a histopathology slide image. We were able to incorporate existing state of the art in both vision and language modeling, by using vision transformers and pre-trained transformer based language models like BERT. Our model is flexible in the sense that we can swap out different and more heavy-duty large language model decoders like T-5 [56] or GPT2-Large [54] that has 770 Million parameters that can be trained for better performance. We were also able to preserve patch location information in our image encoding pipeline. The primary objective for this report generation system was to assist clinicians and pathologist in their work of understanding the disease from images. The report generated from this system can give a clinician a starting point in their investigation of a patient's symptoms. We also showed a comparative analysis of different architectures in Chapter 3 and showed that an LSTM based decoder marginally beat out our transformer based decoders.

The other goal of this dissertation was to necessarily build models that were interpretable in some way, so that if/when these models are to be deployed in a real world clinical setting, there are ways to investigate said models. In Chapter 4 we showed how we can use cross-attention values from the decoder to generate visual attention maps associated with the generated token. We were able to qualitatively validate the efficacy of the attention maps from a board certified pathologist. These interpretation mechanisms are crucial in sensitive settings like healthcare where if the model makes the wrong decision it can lead to potentially fatal consequences.

In Chapter 5, in aid to our stated goal of building trustworthy models, we showed a Positive-Unlabeled (PU) learning method that showed improved performance for detecting Long COVID in the data from the N3C. The goal of improving the performance of our attention based LSTM model, was to make the model more confident in its predictions, which we showed by the improved F1 scores. The primary objective of the research was to determine risk factors for Long COVID, which we did by investigating the attention scores assigned to the input sequence of temporally arranged diagnosis code. To better trust these attention scores, building a more robust model, that is able to better predict Long COVID was crucial.

Some of the publications building towards this work are listed below:

- Sengupta, Saurav, and Donald E. Brown. "Automatic Report Generation for Histopathology images using pre-trained Vision Transformers and BERT." *Accepted to IEEE International Symposium of Biomedical Imaging (ISBI) 2024, Athens, Greece.* https://arxiv.org/abs/2312.01435

- Sengupta, Saurav, and Donald E. Brown. "Automatic Report Generation for Histopathology images using pre-trained Vision Transformers." *Machine Learning for Health (ML4H) 2023 Findings Track.* https://arxiv.org/abs/2311.06176

- Sengupta, Saurav, Johanna Loomba, Suchetha Sharma, Scott A. Chapman Donald E. Brown. “Determining risk factors for Long COVID using Positive Unlabeled learning on Electronic Health Records data from NIH N3C” *accepted at IEEE International Conference of Machine Learning Application (ICMLA) 2023, Jacksonville, Florida*

- Sengupta, Saurav, et al.“Analyzing historical diagnosis code data from NIH N3C and RECOVER Programs using deep learning to determine risk factors for Long Covid.” *2022 IEEE International Conference on Bioinformatics and Biomedicine (BIBM). IEEE, 2022.*

One of the obvious limitations of the attention maps described in Chapter 4 is that the fidelity of the maps are constrained by the large patch size of 4096x4096. While useful, these sometimes cover too large an area of the WSI, to be actually helpful for generating new clinical insights. The most we can do is validate the token generated has some clinical association with the highest ranked patches in terms of attention scores. Future work in this area could focus on improving the fidelity of attention maps.

Our image captioning system was trained on the only available large scale histology image-text paired data that we know of in the GTEx Tissue portal. However, real world clinical notes are more descriptive and comprehensive. The advantage of our architecture is that since it uses existing pre-trained language models as the decoder, once more descriptive clinical notes are available, our model can be fine-tuned on the new data, relatively easily.

For the method described in Chapter 5, we were constrained by the what pre-trained embeddings for the SNOMED-CT coded diagnosis codes that were available, thereby leading to the ‘<unk>’ showing up in Figure 5.8. Future work would focus on using more comprehensive embedding space that covers the breadth of diagnosis codes such that we are able to capture all possible conditions that might lead to pre-disposition to Long COVID.

Bibliography

[1] Mona Mohammed Abdelrahman, Noha Mohammed Abd-Elrahman, and Tasneem Mohammed Bakheet. Persistence of symptoms after improvement of acute covid19 infection, a longitudinal study. *Journal of medical virology*, 93(10):5942–5946, 2021.

[2] Khushbu Agarwal, Tome Eftimov, Raghavendra Addanki, Sutanay Choudhury, Suzanne Tamang, and Robert Rallo. Snomed2vec: Random walk and poincar\'e embeddings of a clinical knowledge base for healthcare analytics. *arXiv preprint arXiv:1907.08650*, 2019.

[3] Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. Publicly available clinical bert embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, 2019.

[4] Julia Amann, Alessandro Blasimme, Effy Vayena, Dietmar Frey, and Vince I Madai. Explainability for artificial intelligence in healthcare: a multidisciplinary perspective. *BMC Medical Informatics and Decision Making*, 20(1):1–9, 2020.

[5] AA Asadi-Pooya, A Akbari, A Emami, M Lotfi, M Rostamihosseinkhani, H Nemati, et al. Long covid-19 syndrome-associated brain fog [published online ahead of print, 2021 oct 21]. *J Med Virol*, 10, 2021.

[6] Daniel Ayoubkhani, Matthew L Bosworth, Sasha King, Koen B Pouwels, Myer Glickman, Vahé Nafilyan, Francesco Zaccardi, Kamlesh Khunti, Nisreen A Alwan, and A Sarah Walker. Risk of long covid in people infected with severe acute respiratory syndrome coronavirus 2 after 2 doses of a coronavirus disease 2019 vaccine: community-

based, matched cohort study. In *Open Forum Infectious Diseases*, volume 9, page ofac464. Oxford University Press US, 2022.

[7] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[8] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005.

[9] J Baruch, C Zahra, T Cardona, and T Melillo. National long covid impact and risk factors. *Public Health*, 213:177–180, 2022.

[10] Jessa Bekker and Jesse Davis. Learning from positive and unlabeled data: A survey. *Machine Learning*, 109:719–760, 2020.

[11] Kaustav Bera, Kurt A Schalper, David L Rimm, Vamsidhar Velcheti, and Anant Madabhushi. Artificial intelligence in digital pathology—new tools for diagnosis and precision oncology. *Nature reviews Clinical oncology*, 16(11):703–715, 2019.

[12] Adrien Bibal, Rémi Cardon, David Alfter, Rodrigo Wilkens, Xiaoou Wang, Thomas François, and Patrick Watrin. Is attention explanation? an introduction to the debate. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3889–3900, 2022.

[13] Bjørn Blomberg, Kristin Greve-Isdahl Mohn, Karl Albert Brokstad, Fan Zhou, Dagrun Waag Linchausen, Bent-Are Hansen, Sarah Lartey, Therese Bredholt Onyango, Kanika Kuwelker, Marianne Sævik, et al. Long covid in a prospective cohort of home-isolated patients. *Nature medicine*, 27(9):1607–1613, 2021.

[14] Longbing Cao. Ai in finance: Challenges, techniques, and opportunities. *ACM Computing Surveys (CSUR)*, 55(3):1–38, 2022.

[15] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[16] Chi-Long Chen, Chi-Chung Chen, Wei-Hsiang Yu, Szu-Hua Chen, Yu-Chan Chang, Tai-I Hsu, Michael Hsiao, Chao-Yuan Yeh, and Cheng-Yu Chen. An annotation-free whole-slide training approach to pathological classification of lung cancer types using deep learning. *Nature communications*, 12(1):1193, 2021.

[17] Richard J Chen, Chengkuan Chen, Yicong Li, Tiffany Y Chen, Andrew D Trister, Rahul G Krishnan, and Faisal Mahmood. Scaling vision transformers to gigapixel images via hierarchical self-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16144–16155, 2022.

[18] Edward Choi, Mohammad Taha Bahadori, Jimeng Sun, Joshua Kulas, Andy Schuetz, and Walter Stewart. Retain: An interpretable predictive model for healthcare using reverse time attention mechanism. *Advances in neural information processing systems*, 29, 2016.

[19] Marc Claesen, Frank De Smet, Johan AK Suykens, and Bart De Moor. A robust ensemble approach to learn from positive and unlabeled data using svm base models. *Neurocomputing*, 160:73–84, 2015.

[20] Vered Daitch, Dana Yelin, Muhammad Awwad, Giovanni Guaraldi, Jovana Milić, Cristina Mussini, Marco Falcone, Giusy Tiseo, Laura Carrozzi, Francesco Pistelli, et al.

Characteristics of long-covid among older adults: A cross-sectional study. *International Journal of Infectious Diseases*, 125:287–293, 2022.

[21] Suparno Datta, Ariane Morassi Sasso, Nina Kiwit, Subhronil Bose, Girish Nadkarni, Riccardo Miotto, and Erwin P Böttinger. Predicting hypertension onset from longitudinal electronic health records with deep learning. *JAMIA open*, 5(4):ooac097, 2022.

[22] Thomas Davenport and Ravi Kalakota. The potential for artificial intelligence in healthcare. *Future healthcare journal*, 6(2):94, 2019.

[23] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[24] Jacob Devlin, Ming-Wei Chang, and Kenton Lee. Google, kt, language, ai: Bert: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186, 2019.

[25] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2019.

[26] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[27] Charles Elkan and Keith Noto. Learning classifiers from only positive and unlabeled

data. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 213–220, 2008.

[28] Jose Estrada-Codecido, Adrienne K Chan, Nisha Andany, Philip W Lam, Melody Nguyen, Ruxandra Pinto, Andrew Simor, and Nick Daneman. Prevalence and predictors of persistent post-covid-19 symptoms. *Official Journal of the Association of Medical Microbiology and Infectious Disease Canada*, 7(3):208–219, 2022.

[29] Joshua J Fenton, Stephen H Taplin, Patricia A Carney, Linn Abraham, Edward A Sickles, Carl D'Orsi, Eric A Berns, Gary Cutter, R Edward Hendrick, William E Barlow, et al. Influence of computer-aided detection on performance of screening mammography. *New England Journal of Medicine*, 356(14):1399–1409, 2007.

[30] Jevgenij Gamper and Nasir Rajpoot. Multiple instance captioning: Learning representations from histopathology textbooks and articles. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16549–16559, 2021.

[31] Jonathan Haab, Nicolas Deutschmann, and María Rodríguez Martínez. Is attention interpretation? a quantitative assessment on sets. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 303–321. Springer, 2022.

[32] Timotius Ivan Hariyanto and Andree Kurniawan. Dyslipidemia is associated with severe coronavirus disease 2019 (covid-19) infection. *Diabetes & Metabolic Syndrome: Clinical Research & Reviews*, 14(5):1463–1465, 2020.

[33] Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal processing magazine*, 29(6):82–97, 2012.

[34] Geoffrey E Hinton, Alex Krizhevsky, and Ilya Sutskever. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25(1106-1114):1, 2012.

[35] Jie Hou and Terry Gao. Explainable dcnn based chest x-ray image analysis and classification for covid-19 pneumonia detection. *Scientific Reports*, 11(1):1–15, 2021.

[36] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[37] George N Ioannou, Aaron Baraff, Alexandra Fox, Troy Shahoumian, Alex Hickok, Ann M O'Hare, Amy SB Bohnert, Edward J Boyko, Matthew L Maciejewski, C Barrett Bowling, et al. Rates and factors associated with documentation of diagnostic codes for long covid in the national veterans affairs health care system. *JAMA network open*, 5 (7):e2224359–e2224359, 2022.

[38] Ryuichi Kiryo, Gang Niu, Marthinus C Du Plessis, and Masashi Sugiyama. Positive-unlabeled learning with non-negative risk estimator. *Advances in neural information processing systems*, 30, 2017.

[39] Adrienne Kline, Hanyin Wang, Yikuan Li, Saya Dennis, Meghan Hutch, Zhenxing Xu, Fei Wang, Feixiong Cheng, and Yuan Luo. Multimodal machine learning in precision health: A scoping review. *npj Digital Medicine*, 5(1):171, 2022.

[40] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[41] Phong Le and Willem Zuidema. Quantifying the vanishing gradient and long distance

dependency problem in recursive neural networks and recursive lstms. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 87–93, 2016.

[42] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553): 436–444, 2015.

[43] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, 2020.

[44] Constance D Lehman, Robert D Wellman, Diana SM Buist, Karla Kerlikowske, Anna NA Tosteson, Diana L Miglioretti, Breast Cancer Surveillance Consortium, et al. Diagnostic accuracy of digital screening mammography with and without computer-aided detection. *JAMA internal medicine*, 175(11):1828–1837, 2015.

[45] Minghao Li, Tengchao Lv, Jingye Chen, Lei Cui, Yijuan Lu, Dinei Florencio, Cha Zhang, Zhoujun Li, and Furu Wei. Trocr: Transformer-based optical character recognition with pre-trained models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 13094–13102, 2023.

[46] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.

[47] Zachary C Lipton, David C Kale, Charles Elkan, and Randall Wetzel. Learning to diagnose with lstm recurrent neural networks. *arXiv preprint arXiv:1511.03677*, 2015.

[48] Bing Liu, Yang Dai, Xiaoli Li, Wee Sun Lee, and Philip S Yu. Building text classifiers using positive and unlabeled examples. In *Third IEEE international conference on data mining*, pages 179–186. IEEE, 2003.

[49] Fantine Mordelet and J-P Vert. A bagging svm to learn from positive and unlabeled examples. *Pattern Recognition Letters*, 37:201–209, 2014.

[50] OpenAI. Introducing chatgpt. https://openai.com/blog/chatgpt, 2022. Accessed: March 1, 2024.

[51] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

[52] Hyeryun Park, Kyungmo Kim, Jooyoung Yoon, Seongkeun Park, and Jinwook Choi. Feature difference makes sense: a medical image captioning model exploiting feature difference and tag information. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 95–102, 2020.

[53] Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Huanqi Cao, Xin Cheng, Michael Chung, Matteo Grella, Kranthi Kiran GV, et al. Rwkv: Reinventing rnns for the transformer era. *arXiv preprint arXiv:2305.13048*, 2023.

[54] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[55] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021.

[56] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning

with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21 (1):5485–5551, 2020.

[57] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[58] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and BjÃ¶rn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. URL https://github.com/CompVis/latent-diffusionhttps://arxiv.org/abs/2112.10752.

[59] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, 2019.

[60] Zohaib Salahuddin, Henry C Woodruff, Avishek Chatterjee, and Philippe Lambin. Transparency of deep neural networks for medical image analysis: A review of interpretability methods. *Computers in biology and medicine*, 140:105111, 2022.

[61] Clayton Scott and Gilles Blanchard. Novelty detection: Unlabeled data definitely help. In *Artificial intelligence and statistics*, pages 464–471. PMLR, 2009.

[62] Saurav Sengupta and Donald E Brown. Automatic report generation for histopathology images using pre-trained vision transformers. *arXiv preprint arXiv:2311.06176*, 2023.

[63] Saurav Sengupta and Donald E Brown. Automatic report generation for histopathology

images using pre-trained vision transformers and bert. *arXiv preprint arXiv:2312.01435*, 2023.

[64] Saurav Sengupta, Johanna Loomba, Suchetha Sharma, Donald E Brown, Lorna Thorpe, Melissa A Haendel, Christopher G Chute, and Stephanie Hong. Analyzing historical diagnosis code data from nih n3c and recover programs using deep learning to determine risk factors for long covid. *arXiv preprint arXiv:2210.02490*, 2022.

[65] Aman Shrivastava, Karan Kant, Saurav Sengupta, Sung-Jun Kang, Marium Khan, S Asad Ali, Sean R Moore, Beatrice C Amadi, Paul Kelly, Donald E Brown, et al. Deep learning for visual recognition of environmental enteropathy and celiac disease. In *2019 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*, pages 1–4. IEEE, 2019.

[66] Richard Socher, Christopher D Manning, and Andrew Y Ng. Learning continuous phrase representations and syntactic parsing with recursive neural networks. In *Proceedings of the NIPS-2010 deep learning and unsupervised feature learning workshop*, volume 2010, pages 1–9. Vancouver, 2010.

[67] Zirui Song and Mia Giuriato. Demographic and clinical factors associated with long covid: Study examines demographic and clinical factors associated with long covid among people who suffer symptoms long after they were first diagnosed with covid-19 (long haulers). *Health Affairs*, 42(3):433–442, 2023.

[68] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[69] Raphael Tang, Linqing Liu, Akshat Pandey, Zhiying Jiang, Gefei Yang, Karun Kumar,

Pontus Stenetorp, Jimmy Lin, and Ferhan Ture. What the daam: Interpreting stable diffusion using cross attention. *arXiv preprint arXiv:2210.04885*, 2022.

[70] Matthew Toews, Christian Wachinger, Raul San Jose Estepar, and William M Wells. A feature-based approach to big data analysis of medical images. In *International Conference on Information Processing in Medical Imaging*, pages 339–350. Springer, 2015.

[71] Nenad Tomašev, Xavier Glorot, Jack W Rae, Michal Zielinski, Harry Askham, Andre Saraiva, Anne Mottram, Clemens Meyer, Suman Ravuri, Ivan Protsyuk, et al. A clinically applicable approach to continuous prediction of future acute kidney injury. *Nature*, 572(7767):116–119, 2019.

[72] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[73] Masayuki Tsuneki and Fahdi Kanavati. Inference of captions from histopathological patches. In *International Conference on Medical Imaging with Deep Learning*, pages 1235–1250. PMLR, 2022.

[74] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[75] Sinong Wang, Belinda Z Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*, 2020.

[76] Zhanyu Wang, Lingqiao Liu, Lei Wang, and Luping Zhou. Metransformer: Radiology

report generation by transformer with multiple learnable expert tokens. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11558–11567, 2023.

[77] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057. PMLR, 2015.

[78] Renyu Zhang, Christopher Weber, Robert Grossman, and Aly A Khan. Evaluating and interpreting caption prediction for histopathology images. In *Machine Learning for Healthcare Conference*, pages 418–435. PMLR, 2020.