UNIVERSITY OF COPENHAGEN
DEPARTMENT OF COMPUTER SCIENCE

M.Sc. Thesis

Andrei Crivoi

VisionBioGPT

Radiology report classification & generation

Advisor: Desmond Elliott

This thesis has been submitted to the M.Sc. School of The Faculty of Science,
University of Copenhagen on November 11, 2023.

Abstract

With the emergence of the Transformer model (Vaswani et al. (2017)), more and more engineers have presented different approaches in using this revolutionary architecture to solve and automate difficult medical tasks.

The task of disease classification is one such example. Doctors are usually tasked with assigning a diagnosis and execute different procedures to help them identify the issues of each patient. Within radiology, for instance, they use imaging support for disease classification, mostly represented as x-rays. This, however, is a very difficult and prone to error process that should not be taken lightly.

For this reason, we propose VisionBioGPT, a BioGPT (Luo et al. (2022)) - Vision Transformer (Dosovitskiy et al. (2021)) hybrid, built on top of the Vision Encoder Decoder system (Li et al. (2022b), Ramos et al. (2023)). We aim to evaluate the performance of BioGPT, a GPT-2 (Radford et al. (2019)) based model pre-trained from scratch on large text corpus of biomedical data from PubMed[1], when tackling radiology-specific tasks (i.e. report, x-ray disease classification and report generation).

We initially conduct some experiments to verify BioGPT's classification and generative capabilities when faced with reports from the **MIMIC-III** (Johnson et al. (2016b)) dataset and compare our results to those of Dai et al. (2022) on similar experiments using the same data, but different models. We establish that BioGPT achieves comparable performance to our point of reference and obtain new pre-trained weights that are more suitable in understanding radiology reports written in the **MIMIC** format.

We then extend our experiments to a new dataset: **MIMIC-CXR** (Johnson et al. (2019a)). We first execute a classification task and obtain promising results with our task-adaptive pre-trained weights. Then, we alter BioGPT's attention block with a cross-attention layer and use it as text-decoder, together with the Vision Transformer as image-encoder and add chest x-rays to our input sequence. This method shows considerably reduced performance compared to our baseline text-only approach.

We believe that our experiments will entice researchers in further experimenting with the BioGPT model in text-only setups for disease classification and expand the model to newer GPT architectures (Brown et al. (2020b), OpenAI (2023)).

[1] PubMed available at pubmed.ncbi.nlm.nih.gov

Acknowledgements

I would like to thank Desmond Elliott for his support, supervision and experiment-designing assistance when writing this thesis. Moreover, I would like to thank the University of Copenhagen for providing hardware and technical support for executing the experiments and my family for constant moral support during these times.

Table of Contents

Chapter 1

Introduction

Radiology is defined by the National Institute of Health as a field of medicine that employs imaging technology to diagnose and treat various diseases.[1] With increasingly powerful processing power and imagery quality, medical imaging has become the pinnacle method of diagnosis in many different areas of medicine, making radiology a very important branch to further study and improve on in order to find new ways of treating previously untreatable diseases or to streamline treatment for existing methods. Radiology is usually categorized into diagnostic (ex. computed tomography, fluoroscopy, plain x-rays, etc.) and interventional (ex. angiography, embolization to control bleeding, cancer treatments, etc.) radiology. Our research will mostly focus on the diagnostic subpart of radiology and specifically on chest x-rays.

Chest radiography is a widely used imaging technique for evaluating the human thorax and is one of the most frequently performed medical imaging examinations worldwide. It plays a crucial role in identifying both acute and chronic cardiopulmonary conditions, ensuring the correct positioning of medical devices like pacemakers, central lines, and tubes, and assisting in related medical investigations. In contrast to its importance, said procedure is difficult to execute and interpret by radiologists as it requires vast knowledge and understanding of each aspect of an x-ray and a good eye for detail. Sadly, latest trends show a decrease in the number of radiologists as a percentage of the physician workforce in the U.S. (Rosenkrantz et al. (2015)), and that radiologists tend to be concentrated in larger urban territories (Rosenkrantz et al. (2018)). These factors are amongst the biggest in impairing timely

[1] Source: official MedLinePlus website

medical imaging interpretation, which in turn compromises the care quality in large organizations. In areas with scarce resources, the situation worsens by a large margin, as radiology services are extremely hard to come by. For example, studies from 2015 show that for a population of 12 million people in Rwanda, only 11 radiologists were available (Rosman et al. (2015)), while for Liberia's population of 4 million, only 2 practicing radiologists were reported (Ali et al. (2015)). Given these statistics, finding new and improved ways of automating diagnostic radiology comes across as mandatory for the general health of the worlds population, especially in economically disadvantaged regions.

When conducting studies, radiologists are often required to write reports which serve as vital information for the patients' well-being. Such reports can contain anything from patient identifiers to details of their symptoms and methods that were used for treatment during their stay in a hospital unit. This being said, generating accurate and comprehensive radiology reports can be time-consuming, prone to human error and subjectivity and overall expensive in every sense of the word. Doctors may have different approaches in treating a patient and hospitals might have different templates for writing said reports. Our research aims to examine both new and existing methods for generating and classifying radiology reports with appropriate disease codes in order to help mitigate these issues. We initially fine-tune models to achieve ICD-9 classification and then experiment with a different labeling system.

Within the field of Natural Language Processing, the medical domain has always been a topic of interest and importance as human life and health quality can be drastically improved with the help of machine assisted diagnosis. ICD coding in Natural Language Processing has vastly been simplified as a sequence classification task and previous methods build solutions using CNNs, LSTMs and RNNs (Karimi et al. (2017), Xie and Xing (2018), Yuan et al. (2022)) and have held the state-of-the-art title for a long period of time.

In recent years, Transformer models (Vaswani et al. (2017)) have completely taken over the scene with amazing state-of-the-art results in just about every scenario they were experimented with. Models like BERT (Devlin et al. (2019)) have shown how versatile the Transformer architecture actually is and how large scale pre-training of language models can benefit their performance. Transformers are so versatile, in fact, that they can also be used as vision models and extract meaningful visual representations from images to understand context and scenery (Dosovitskiy et al. (2021)). They can also work with text-image pairs in vision and language setups (Li et al. (2022b), Ramos et al. (2023)), making them perfect for radiological diagnosis and report generation.

Past studies have shown that different BERT variants (Gu et al. (2021), Peng et al. (2019)) of the Transformer can achieve impressive results when applied to the task of ICD-coding (Ji et al. (2021a), Gao et al. (2021)), but have not been able to overthrow CNNs, LSTMs and RNNs. BERT, however, has a bidirectional nature completely scraping its ability to generate text. Generative Pre-trained Transformer (GPT, Radford et al. (2018)) is an alternative language model architecture to BERT which does exactly what BERT cannot, making it better suited for auto-regressive modeling tasks. As opposed to BERT and its variants, GPT has seen less use in biomedical research prior to the introduction of BioGPT (Luo et al. (2022)), a large-scale language model trained on biomedical text data, capable of producing coherent biomedical reports and has a vast understanding of diseases, their meanings and implications.

While a very strong tool for generating coherent biomedical text, BioGPT's limitation is given by its inability to process images. In this research project, we will examine BioGPT's performance when paired with a Vision Transformer (Dosovitskiy et al. (2021)) which was originally developed for computer vision tasks, but has demonstrated impressive capabilities in analyzing medical images and extracting significant visual features. By incorporating ViT, we aim to capture the rich visual information embedded within medical images and leverage it to generate more contextually accurate radiology reports and better ICD-9 code classification. In order to accommodate the pairing of the two models, a cross-attention layer will be added on top of the multi-head attention already existent in BioGPT.

To evaluate the effectiveness of this approach, we first compare BioGPT with similar models capable of long document classification that have already been used and proved their worth when classifying radiology reports, presented by Dai et al. (2022) in their work. At this stage, we conduct experiments using a large corpus of Intensive Care Unit (ICU) discharge summaries, each of which annotated with multiple labels using the ICD-9 system, gathered from the **MIMIC-III** dataset, evaluating the performance of BioGPT in a multi-label classification setup. We then review how the model performs when also attending to chest x-rays, alongside their corresponding radiology report and associated labels from the **MIMIC-CXR** database, again, in a multi-label classification setup.

In the following study, we will be taking a deep dive into the strengths and weaknesses of the BioGPT model and assess its generative abilities in producing accurate radiology reports by examining chest x-rays and, by doing so, we aim to answer a multitude of research questions:

1. How does BioGPT compare to its predecessors and alternative BERT-base variants and long-document classifiers when performing ICD-9 classification on radiology reports?

2. Does classification performance improve after continued pre-training the BioGPT language model on radiology reports?

3. After continued pre-training, is the new version of BioGPT able to accurately predict diseases with a new labeling system?

4. Is a BioGPT-ViT hybrid capable of learning important features in chest x-rays and produce accurate reports based on medical imagery? Do we expect better performance on disease classification when incorporating chest x-rays?

We believe that through answering these questions we will bring the following contributions to research within the fields of Natural Language Processing, Transformer-based models and radiological research:

1. We compare different methods of ICD classification with the base BioGPT and show that even at the limited 1024 sequence length it can process, it is still able to achieve comparable results to Longformer (Beltagy et al. (2020)) on **MIMIC-III** that can process up to 4096 tokens (Longformer was tested on **MIMIC-III** (Johnson et al. (2016a)) data in the work of Dai et al. (2022))

2. We continue pre-training the BioGPT model in an unsupervised manner on the **MIMIC-III** abstracts and show a similar improvement in F1-score to Longformer

3. We obtain impressive results when using the newly found weights to perform disease classification on **MIMIC-CXR** (Johnson et al. (2019b)) reports with a new label encoding, specific to this dataset and show that BioGPT is potentially a very strong choice for this task

4. We extend BioGPT's architecture with a cross-attention layer to allow it to attend to visual embeddings given by the Vision Transformer and show that it can understand connections between reports and corresponding x-rays from the **MIMIC-CXR-JPG** database while also taking into account their orientation, but classification performance plunges, indicating that BioGPT might work better as a text-only model.

In summary, this thesis presents a novel approach that combines the Vision Transformer and the BioGPT language model to automate radiology report generation and enhance ICD-9 classification accuracy. By integrating visual and textual information, our method aims to provide comprehensive and accurate reports, streamlining the radiology workflow and facilitating improved healthcare outcomes.

Chapter 2

Background & Related Work

2.0.1 Transformers

Up until 2017, recurrent neural networks (Rumelhart et al. (1986), Jordan (1986)), long short-term memory (Hochreiter and Schmidhuber (1997)) and gated recurrent neural networks (Cho et al. (2014)) have been shown to achieve state-of-the-art performance in various Natural Language Processing tasks such as language modeling, machine translation and sequence modeling. However, due to their recurrent nature, these models cannot handle parallelization when training examples, which brings severe batch size limitations at longer sequence lengths.

In their pioneering paper ”Attention Is All You Need”, Vaswani et al. (2017) introduce a revolutionizing model architecture which solely relies on an attention mechanism. This approach eliminates the necessity for recurrence, leading to significant improvements in parallelization performance. Consequently, training times are considerably reduced and models show enhanced capabilities, achieving a new state-of-the-art in translation quality after a very short training period.

The Transformer model has emerged as a highly successful architecture in various natural language processing tasks, consistently achieving state of the art performance. Figure 2.1 shows a detailed explanation of it.

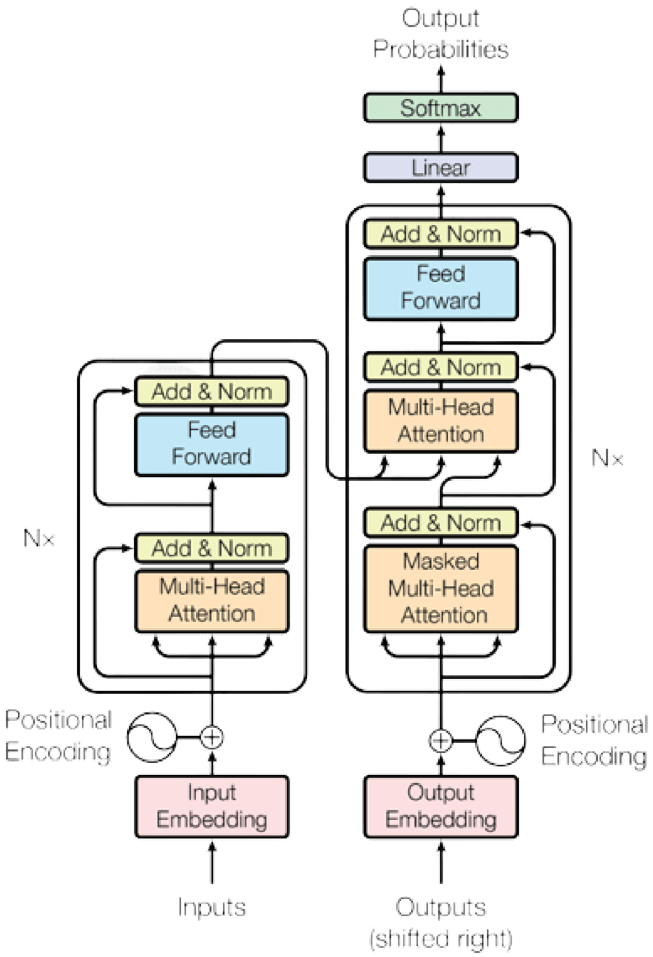

Figure 2.1: Transformer model architecture, taken from Vaswani et al. (2017)

Like most other relevant neural sequence transduction models, the Transformer is based on an encoder-decoder structure:

- **Encoder:** maps input sequence of symbol representations $(x_1, \cdots, x_n)$ to a sequence of continuous representations $\mathbf{z} = (z_1, \cdots, z_n)$
- **Decoder:** generates an output of symbols $(y_1, \cdots, y_m)$ one element at a time , based on $\mathbf{z}$ and previously generated symbols in an auto-regressive fashion (Graves (2014))

As seen in Figure 2.1, both the encoder and decoder are composed of a stack of **N** identical layers. An encoder stack has two sub-layers, both of which are followed by a layer normalization (Ba et al. (2016)) and leverage a residual connection around them (He et al. (2015)). These are a multi-head self-attention mechanism and a simple, position-wise fully connected feed-forward network. The decoder inserts a third sub-layer which performs multi-head attention over the output of the encoder stack. The self-attention mechanism in the decoder is modified to prevent positions from attending to subsequent positions.

The attention mechanism can be defined as a function that takes a query and a set of key-value pairs as inputs and produces an output, by calculating a weighted sum of the values, where the weights are determined by a compatibility function of the query and corresponding key.

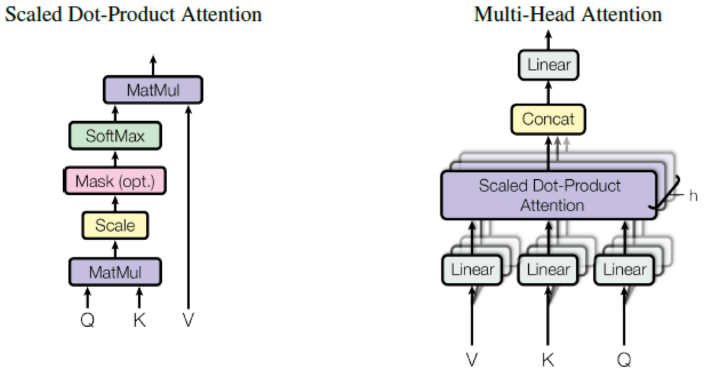

Figure 2.2: Attention mechanisms as described by Vaswani et al. (2017) | Scaled Dot-Product Attention (left), Multi-Head Attention (right)

Vaswani et al. (2017) propose the "Scaled Dot-Product Attention", an attention mechanism where the input is given by queries and keys of dimension d_q and d_k, and values of dim d_v. The query is used in a dot product operation with all keys and the result is divided by $\sqrt{d_k}$. A softmax function is applied to obtain the weights on the values.

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

In comparison to the two most frequently used attention functions, additive (Bahdanau et al. (2016)) and multiplicative, "Scaled Dot-Product Attention" is almost identical with the latter, with the main difference given by the scaling factor $\frac{1}{\sqrt{d_k}}$. Dot-product attention is usually the preferred choice for being more efficient in practice, both in time and space complexity.

The ”Multi-Head Attention” mechanism is the one primarily used in the Transformer architecture. As shown in Figure 2.2, a linear projection is applied to the query, key and value vectors h times with different linear projections to d_q, d_k and d_v dimensions, respectively. On each projection, an attention function is performed as described by the ”Scaled Dot-Product Attention”. The yielded outputs are concatenated and projected through a linear layer, resulting in the final attention output.

$$\text{MultiHead}(Q, K, V) = \text{Concat}(head_1, \cdots, head_n)W^O$$

$$\text{where } head_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

Here the projections W_i^Q, W_i^K, W_i^V, W^O are parameter matrices. $h = 8$ parallel attention layers are used in the Transformer. For each layer, $d_k = d_v = \frac{d_{model}}{h} = 64$. This reduced dimensionality leads to the total computational cost being comparable to that of single-head attention with full dimensionality.

As seen in Figure 2.1, the ”Multi-Head Attention” is used within the Transformer architecture in three different scenarios. Both the encoder and decoder posses self-attention layers. For the encoder, each position can attend to all positions in the previous layer, meaning that the queries, keys and values are all generated from the output of the previous layer. The self-attention layers in the decoder work in a similar fashion. In order to preserve the auto-regressive property, leftward information flow through the decoder is averted by masking out all values in the input of the softmax which correlate with illegal connections. The decoder also uses attention layers to attend to the outputs of the encoder. In this case, the keys and values are obtained from the encoder output, while the queries come from the the previous decoder layer.

The Transformer model represents a significant breakthrough that paves the way for advancing the development of robust language models by leveraging faster training speeds and improved computational efficiency. Since its inception, the Transformer has garnered extensive scholarly attention and has been widely explored across diverse domains, aiming to attain novel state-of-the-art performance in a multitude of tasks.

2.0.2 Pre-trained Language Models in biomedical domain

Within the deep learning field, one of the most noteworthy methods of conducting research is pre-training models on large scale unlabeled data in supervised or unsupervised manners and then fine-tuning them on various downstream tasks. Transformer-based language models are no exception, as they have shown enhanced performance when pre-trained on large scale text corpus on tasks such as masked language modeling, next sentence prediction, question answering, causal language modeling, etc.

While pre-training on large scale text corpus is highly efficient for learning contextualized representations of natural language, the resulting models develop a very broad understanding of the language itself as they work with a large vocabulary, typically constructed of the most common words used naturally. When dealing with domain-specific tasks, it can be wise to further pre-train or fully pre-train the models on in-domain text, as the vocabulary size and contextual relations between words might differ from the general day-to-day use. The works of Peng et al. (2019), Beltagy et al. (2019) and Lee et al. (2019) show how a model like BERT (Devlin et al. (2019)) can perform better on domain-specific tasks when additional pre-training is done on a large corpus of scientific publications (Beltagy et al. (2019)) or biomedical data (Peng et al. (2019), Lee et al. (2019)).

BERT, introduced by Devlin et al. (2019), is a bidirectional transformer-based contextualized language model pre-trained on large scale text corpus of English Wikipedia and BookCorpus. It provides contextualized word representations that can be used to fine-tune on downstream tasks and achieves great success on many natural language understanding tasks. BERT-based models have been extensively studied in the biomedical domain (Lee et al. (2019), Gu et al. (2021)) and have shown good language understanding and classification capabilities. However, due to the bidirectional architecture of BERT, it severely lacks in generation power. Being bidirectional, BERT can processes data both left-to-right and right-to-left and leverages this fact to jointly learn language features by observing both past and future tokens. Since with generative tasks the aim is generally to predict following tokens in a sentence based on what has already been observed, BERT is not a well suited model for such tasks as they lack ”future” context.

Generative Pre-trained Transformer (GPT) is a model architecture presented in the work of Radford et al. (2018) which is carefully designed to execute language generation tasks. It is pre-trained on large scale text corpus in a causal language modeling context (i.e. the model learns to predict the next token based solely on previously observed tokens in a sentence). As computation power got stronger, the GPT model has seen many improvements with more complex architectures like GPT-2 (Radford et al. (2019)), GPT-3 (Brown et al. (2020b)) and most recently GPT-4 (OpenAI (2023)) that benefit from larger model size (learnable parameters) and are pre-trained on text corpus at larger and larger scales. These models prove impressive performance on various language modeling tasks and classification tasks. Appropriate prompt design is although required. Well established, descriptive, complete and coherent prompts are usually the most important factor for this model to thrive and, with careful design, fine-tuning may not even be needed for certain downstream tasks.

In contrast to BERT, GPT-based models have not been as researched within the biomedical domain. As such, Luo et al. (2022) propose a BioGPT variant of the GPT-2 architecture designed for generating relevant biomedical text. Gu et al. (2021) explain in their research the importance of training only on in-domain data from scratch when training for a task related to a specific domain. BioGPT is pre-trained on 15 million titled abstracts from the **?** database. Gu et al. (2021) also show that learning in-domain vocabulary is crucial. Thus, the learned vocabulary of BioGPT corresponds to the used biomedical corpus, instead of using the full GPT-2 vocabulary. Byte pair encoding (BPE, Sennrich et al. (2016)) was used to tokenize the words into subwords and learn the vocabulary piece by piece. Figure 2.3 shows an example of how different tasks can be adapted to BioGPT using proper prompting methods.

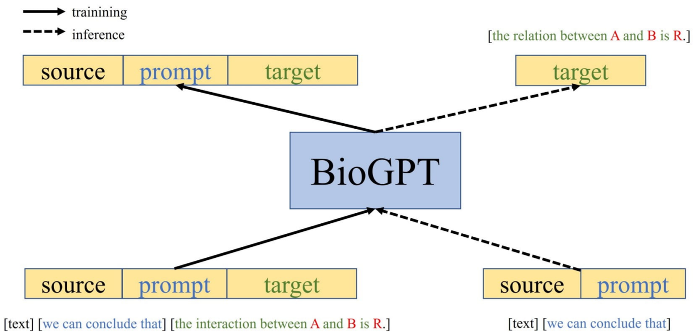
trainining
inference
[the relation between A and B is R.]
source
prompt
target
target
BioGPT
source
prompt
target
source
prompt
[text] [we can conclude that] [the interaction between A and B is R.]
[text] [we can conclude that]

Figure 2.3: BioGPT framework, as described in Luo et al. (2022)

As with the Transformer (Vaswani et al. (2017)), the core component of GPT-2 and consequently BioGPT is the multi-head attention described in Section 2.0.1. BioGPT has 24 layers with a hidden size of 1024 and 16 attention heads resulting in 347 million parameters, 8 million shy of GPT-2's 355 million. This dissimilarity comes only from the different embedding size and output projection size, caused by the inconsistency in vocabulary size. After pre-training, the model was fine-tuned on a variety of different downstream tasks such as end-to-end relation extraction, question answering, document classification and causal language modeling (text generation), all while using relevant biomedical data. It was then compared to previously established state of the art models in the biomedical domain, such as BioBERT (Lee et al. (2019)), PubMedBERT (Gu et al. (2021)), BioLinkBERT (Yasunaga et al. (2022)), BioELECTRa (Kanakarajan et al. (2021)) REBEL (Brown et al. (2020a)) on each individual task. GPT-2, as the backbone network of the model, was also reported in the comparison when relevant to show the increased capabilities of BioGPT when it comes to biomedical text generation. As expected, BioGPT exceeds its predecessors and achieves a new state-of-the-art in all executed experiments. This makes the model a good candidate for future expansion in the biomedical domain.

One main limitation of the GPT-2 architecture, and inherently of BioGPT, is its lack of vision capabilities. Biomedical research is more and more reliant on imagery, especially within radiology where diagnosis is based on cutting edge imaging technology. To address this limitation, we employ a vision encoder-decoder setup using the Vision Transformer (Dosovitskiy et al. (2021)) as encoder and BioGPT (Luo et al. (2022)) as decoder (this can be conceptually visualized in Figure 2.1 where the left side is the encoder and right side is the decoder). To make this interaction possible, we slightly alter the implementation of BioGPT's architecture[1] (Wolf et al. (2020)) by adding a cross-attention layer to BioGPT's attention module (this can be compared to the multi-head attention layer in the decoder that receives the key, value states from the encoder in Figure 2.1, i.e. it attends to encoder outputs).

2.0.3 Vision Encoder-Decoder Setup

Li et al. (2022b) show the success of using pre-trained checkpoints for initializing image to text sequence models in this image-encoder, text-decoder

[1] original implementation from huggingface transformers code-base

setup. Their optical character recognition model (TrOCR) achieved new state-of-the-art performance on printed, handwritten and scene text recognition tasks. TrOCR leverages the transformer architecture where the purpose of the encoder is to extract meaningful features from the input image patches and obtain their representation, while the decoder generates the output sequence of words by attending to visual features from the encoder and its own previous states. Similar approaches have been made for image captioning models, where the input is formed of an image passed to a vision encoder and the output is generated by an auto-regressive decoder. Such models are pre-trained on large-scale image-text pairs and hold state-of-the-art title for most image captioning tasks. General image captioning tasks aim to find an appropriate heading to describe what the input image depicts using natural language. The works of Li et al. (2020), Li et al. (2022a), Hu et al. (2022), Wang et al. (2022) show different implementations of vision-and-language models for image captioning leveraging the encoder-decoder architecture, all with remarkable results.

Another recent significant research within image captioning and vision encoder-decoders was conducted by Ramos et al. (2023). Their work augments image captioning with prompts, aiming to help the model better understand the context and features of an input image by assisting it with a predefined prompt fed to an auto-regressive language model. Their ”SmallCap” model consists of a CLIP vision model (Radford et al. (2021)) for the encoder and GPT-2 (Radford et al. (2019)) for the decoder. CLIP is a suitable model in this scenario because, at its core, it is its own vision-and-language model, capable of working on image-text pairs. Ramos et al. (2023) use the vision-encoder of CLIP to get the patch embeddings of the input image. A cross-attention layer is added to GPT-2 such that it learns to attend to the image features. Moreover, CLIP is also used to retrieve similar captions to what the end model is expected to output, from a large collection of captions. These captions are then adapted into a prompt to push GPT-2 towards giving an accurate description of the original input image. This approach has been especially helpful in our research as it provided us with a clear explanation and implementation of the vision encoder-decoder model and instructions on how to augment the GPT architecture with a cross-attention layer (Lin et al. (2021)).

2.0.4 Vision Transformer

Since its release in 2017, the Transformer has been widely accepted as the model of choice in natural language processing tasks, thanks to its ability to train models of extraordinary size, made possible by its computational efficiency and scalability.

In computer vision, nonetheless, convolutional neural networks kept their state-of-the-art title (LeCun et al. (1989); Krizhevsky et al. (2012); He et al. (2015)). Up until 2021, many researchers have tried different approaches in using the Transformer self-attention mechanism with CNN architectures (Wang et al. (2018), Carion et al. (2020)) and even removing convolutions as a whole (Ramachandran et al. (2019), Wang et al. (2020)). While these models obtain efficient results, they can never be scaled to the level of the transformer.

In 2021, Dosovitskiy et al. (2021) introduce the Vision Transformer (ViT), a Transformer-based model built specifically for working with vision data. In order to make it scalable, the authors followed the Transformer architecture with minimal change. To achieve this, the images are split into patches, converted into linear embeddings and fed to a Transformer. Basically, the image patches are equivalent to tokens in the context of the Transformer.

Initially, Vision Transformer achieved modest performance when trained on data sets of medium size. This behaviour is to be expected as the transformer lacks some of the innate inductive capabilities present in CNNs resulting in limited generalization when trained on insufficient data. Knowing of this limitation, the model was trained on larger datasets of 14-300 million images. This proved that large scale training is more valuable than inductive bias. With this approach, the Vision Transformer comes close to and, in some cases, exceeds state-of-the-art performance on multiple image recognition benchmarks.

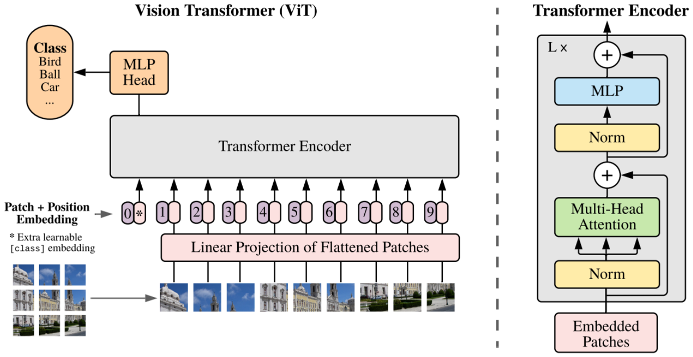

Figure 2.4: ViT model overview, taken from Dosovitskiy et al. (2021)

Figure 2.4 shows a detailed overview of how the Vision Transformer functions. Since the adopted architecture is that of the standard Transformer (Vaswani et al. (2017)), its input must be a one-dimensional collection of token embeddings. Images, on the other hand, are two-dimensional entities characterized by their dimensions of width W, height H, and the number of color channels C (typically employing RGB encoding, where $C = 3$). To tackle the dimensional disparity, the input image is split into a sequence of flattened patches of shape $(N, P^2 \times C)$, where P is both the width and height of one image patch and N is the number of resulting patches, equal to $\frac{HW}{P^2}$. The patches are then passed through a trainable linear projection to D dimensions, obtaining patch embeddings. Here, D is the constant latent vector size of the Transformer's layers.

BERT (Devlin et al. (2019)) uses a special $[CLS]$ token in front of every input example to signify the beginning of input. Likewise, a custom learnable embedding is prepended to the sequence of patch embeddings in the Vision Transformer. Each of the patch embeddings is also prepended with positional embeddings to preserve their initial ordering information. The positional embeddings are one-dimensional, as more refined embeddings with two-dimensional information did not exhibit significant performance increase. This final sequence is then fed as input to the transformer encoder.

The encoder, as described by Dosovitskiy et al. (2021), incorporates layers of multi-head self-attention (see Figure 2.2) intertwined with multi-layer perceptron (MLP) blocks. Each block is surrounded by layer normalization (Ba et al. (2016)) at the beginning and residual connections (He et al. (2015), Wang et al. (2019)) at the end.

2.0.5 Long Document Classification & ICD Coding

Dai et al. (2022) talk about the challenges of training models on long documents and specifically refer to Intensive Care Unit (ICU) discharge summaries from the **MIMIC-III** dataset (Johnson et al. (2016a)) and other long documents from various sources. The main issue when classifying long documents is that most recent language models have a cap on the number of tokens they can observe at a time. BERT (Devlin et al. (2019)), for example, is pre-trained on sequences of up to 512 tokens, while BioGPT (Luo et al. (2022)) goes up to 1024 tokens. Longformer, the long-document Transformer proposed by Beltagy et al. (2020), on the other hand, can process up to 4096 tokens. This massive increase in the number of tokens is given by a special implementation of the self-attention inside the Transformer. The vanilla Transformer's self-attention mechanism runs in $O(n^2)$ time and memory complexity for inputs of sequence length equal to n. This occurs because every token in a sequence attends to all other tokens which makes it exponentially difficult to process long documents. To address this challenge, Beltagy et al. (2020) implement a new attention function, where the attention connections between tokens are sparsified. This new implementation scales linearly with the input sequence length, allowing for longer sequences to be passed through and evaluated by the model.

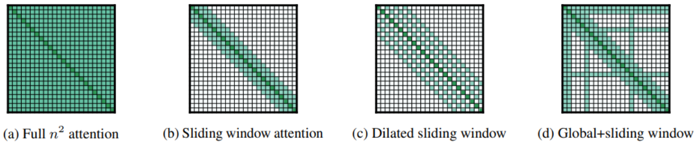
(a) Full n^2 attention (b) Sliding window attention (c) Dilated sliding window (d) Global+sliding window

Figure 2.5: Different patterns of attention in Longformer, taken from (Beltagy et al. (2020))

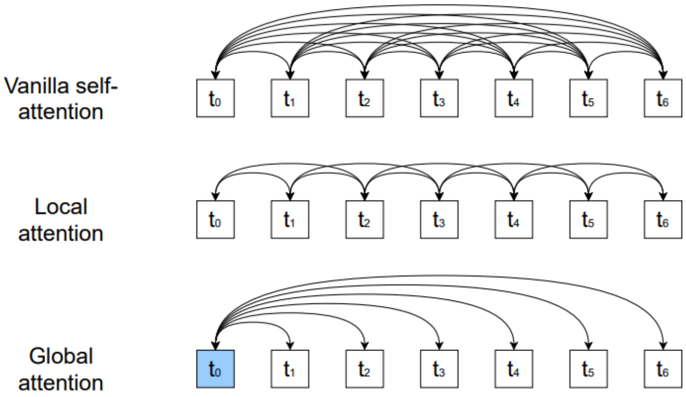

Figure 2.6: Different patterns of attention as explained by Dai et al. (2022)

Figures 2.5 and 2.6 compare Longformer's attention (2.5 (b), (c), (d)) to the classic Vaswani et al. (2017) Transformer multi-head attention (2.5 (a)). Firstly, a **sliding window** technique is used, ensuring attention to local context (2.5(b)). Each token attends to a fixed number w of tokens surrounding it, resulting in a linear time complexity of $O(n \times w)$. Secondly, attention size is extended by "**dilating**" the **sliding window** such that there are gaps of size d inside the window (2.5(c)), while maintaining the same linear complexity. Finally, the **dilated sliding window** method is combined with global attention at specific locations from the input (2.5(d)) in order to more accurately learn task-specific representations. Since the number of tokens that use global attention is predefined and relatively small, this method does not add much computation overhead and the complexity is still linear in terms of n.

The International Statistical Classification of Diseases and Related Health Problems (ICD) is a globally recognized and standardized system for classifying diseases, medical conditions, and other health-related issues. It is maintained by the World Health Organization (WHO) and provides valuable statistics on the causes and effects of human disease and death worldwide. The ICD provides a structured and consistent way to categorize diseases and health conditions, enabling healthcare professionals to better understand and monitor population health, study disease patterns, and facilitate international health information exchange. The most recent version is the ICD-11, which was released in 2018. **MIMIC-III**, however, is labeled using ICD-9 codes as it was released in 2016.

ICD classification of documents is never a lightweight task. It requires careful expertise in the field of medicine and is prone to errors as there are thousands of possible codes which makes it very easy to overlook options as humans. Research from Pestian et al. (2007), Farkas and Szarvas (2008), Johnson et al. (2016a), Koopman et al. (2015) paved the way for achieving automatic classification of documents with relevant ICD codes. Previous methods approach this task as a classic sequence classification problem and build solutions with CNNs (Karimi et al. (2017)) or LSTMs (Xie and Xing (2018)). Various BERT variants have also been researched for ICD classification. Ji et al. (2021a) and Gao et al. (2021) show that PubMedBERT (Gu et al. (2021)) and BlueBERT (originally named NCBI_BERT, Peng et al. (2019)) outperform other variants pre-trained on bio-medical literature, with the latter taking the top spot.

Dai et al. (2022) experiment using the Longformer of Beltagy et al. (2020) on labeled discharge summaries from the **MIMIC-III** dataset (Johnson et al. (2016a)) and show considerable performance increase when the model is able to process longer sequences. We will later reproduce some of their experiments and compare results in order to establish a baseline of how well the base pre-trained BioGPT model (Luo et al. (2022)) performs on a multi-label sequence classification task, where the objective is assigning possible ICD codes to each input bio-medical document. It is, again, important to note that BioGPT is limited to 1024 tokens, four times less than Longformer's 4096, so we might expect slightly worse results.

Chapter 3

Proposed approach

3.0.1 Pre-trained models & Setup

Our end model is built on top of the vision encoder decoder setup from huggingface (Wolf et al. (2020)). As per their documentation, this generic VisionEncoderDecoder [1] class serves as a facade to initialize image-to-text-sequence models with any combination of pre-trained auto-encoding vision model and auto-regressive text model.

Similar setups were explored by Rothe et al. (2020) who have shown the effectiveness of initializing sequence-to-sequence models with pre-trained checkpoints and Li et al. (2022b) who achieved new state-of-the-art performance on various character recognition tasks as we have established in Section 2.0.3.

We use the vanilla implementation of the **VisionEncoderDecoderModel** and initialize it from pre-trained checkpoints of the encoder and decoder respectively. For the encoder we use the **Vision Transformer** (Dosovitskiy et al. (2021)) initialized from a pre-trained checkpoint released by **Google**, while the decoder is represented by an altered version of the BioGPT model (Luo et al. (2022)) to which we add a cross-attention layer to give it image-feature-attending capabilities. We give this model a very inspiring name of **BioGptWithCrossAttention**. We now examine each pre-trained checkpoint, its training and fine-tuning methods and expected use.

1. **Encoder**: google/vit-base-patch16-224 presents the Vision Transformer

[1] documentation and code available on github

model pre-trained on ImageNet-21k (a dataset of 14 million images and 21.843 classes, Ridnik et al. (2021)) at a resolution of 224x224 and fine-tuned on ImageNet 2012 (1 million images and 1000 classes, Deng et al. (2009)) at the same resolution [2]. The model receives input images as a sequence of fixed-size, linearly embedded patches of 16x16 pixels. Each image patch is accompanied by an absolute positional embedding and the sequence of patches is prepended with a custom $[CLS]$ token to represent the start of input. The model learns deep representations of the input images that can then generate features for various downstream tasks. There are many implementations of the ViT model suited for these tasks, nevertheless, we use the base model, with no modeling head attached on top of it since we want BioGPT to handle the downstream task and ViT to only use its encoding capabilities and extract relevant image features. During preprocessing, the images are rescaled to 224x224 pixels and normalized across the RGB channels. (Wu et al. (2020)). An alternative checkpoint we will also discuss is google/vit-base-patch16-384 which follows the same pre-training of the previous one, but is fine-tuned on images at a higher resolution of 384x384 allowing for more accurate representations, but larger computation overhead.

2. **Decoder**: microsoft/biogpt, originally described in the paper by Luo et al. (2022), has been pre-trained on large-scale biomedical data and fine-tuned on six different natural language processing tasks, outperforming previous similar models on most experiments. Out of the pre-trained variants of the model, we experiment with two of them, namely for causal language modeling (for auto-regressive task) and sequence classification (for ICD labeling). We will be experimenting with the base pre-trained checkpoint of 347 million parameters. An alternative would be also using microsoft/BioGPT-Large with more parameters and possibly better performance, but larger computation resource requirements.

For altering BioGPT's architecture with a cross attention layer, we follow the approach of Ramos et al. (2023) [3]. Their codebase shows examples of adding or altering cross attention layers for different models like OPT

[2] repository available on github

[3] code available on github

(Zhang et al. (2022)), GLM (Du et al. (2022)) and GPT-2 (Radford et al. (2019)) and using them as decoders in the same vision-encoder-decoder setup we have previously described in Section 2.0.3. By comparing SmallCap and related models to their original implementation in the HuggingFace (Wolf et al. (2020)) codebase, we manage to alter BioGPT in a similar fashion, by adapting its attention block to also support cross attention. In order to achieve this, we adjust the size of the projection matrices in the attention block. As described by Vaswani et al. (2017) in the multi-head attention algorithm (Figure 2.2), a linear projection is applied to the query, key and value vectors h times. The main difference is that now, the key, value projections have the same input size of $d_{k,v} = 768$ given by the initial size of the input images that we set to 384x384 (more about this will be explained in the Preprocessing section 3.0.2).

To assess the quality of the BioGPT model for ICD-9 sequence classification and radiology report generation, we follow the work and codebase of Dai et al. (2022) [4]. In this research, the authors examine the importance of models' abilities to process longer sequences of tokens and specifically evaluate their models on the **MIMIC-III** dataset, which provides hospital discharge summary reports that are annotated with ICD-9 codes, making it a valuable asset for our analysis as well. They also show that their models perform better when additional unsupervised pre-training is conducted on their task-specific dataset. We follow this pattern to obtain new weights for the BioGPT model to further increase its understanding and performance when working with ICD labels and disease classification.

The final setup of our model executes three different steps to obtain the desired output. These steps are described below and can be conceptually visualized in Figure 3.1:

1. Input images are fed through an image processor particular to ViT[5,6]. This processor handles required preprocessing steps to extract image features and transform them into patch embeddings. At this step, the processor performs resizing, resampling, rescaling and normalization to the image mean and standard deviation across the RGB channels, generating a tensor of pixel values of shape:

$$(batch\_size, num\_channels = 3, width = 224, height = 224)$$

[4] repository available on github

[5] documentation available on the huggingface website

[6] code available on github

This output tensor can be seen as the sequence of position-aware patch embeddings that serve as input for the Vision Transformer.

2. A text tokenizer specific to BioGPT[7] receives the according discharge summary or radiology report that relates to the input image and performs tokenization resulting in a tensor of input ids (text embeddings). Since BioGPT can only process 1024 tokens at a time, we truncate the input to the first 1024 tokens. While this means that part of the input is lost, it is a common approach in NLP when dealing with long text sequences. Different methods can be experimented, but may not result in a considerable performance increase.

3. The resulting pixel values and input ids are forwarded through the VisionEncoderDecoder model. Firstly, the encoder processes the pixel values and returns key-value embeddings suited for the decoder to attend to, represented by its final hidden state. Secondly, the decoder receives text embeddings alongside the encoder hidden states and attention mask. It then iterates through its layers by also retaining information of its previous outputs. Lastly, the decoder outputs some predicted disease classes or ICD-9 codes when doing sequence classification. Alternatively, we feed the pixel values as input and expect our language-decoder to generate an accurate report when performing causal language modeling.

[7] tokenizer available on github

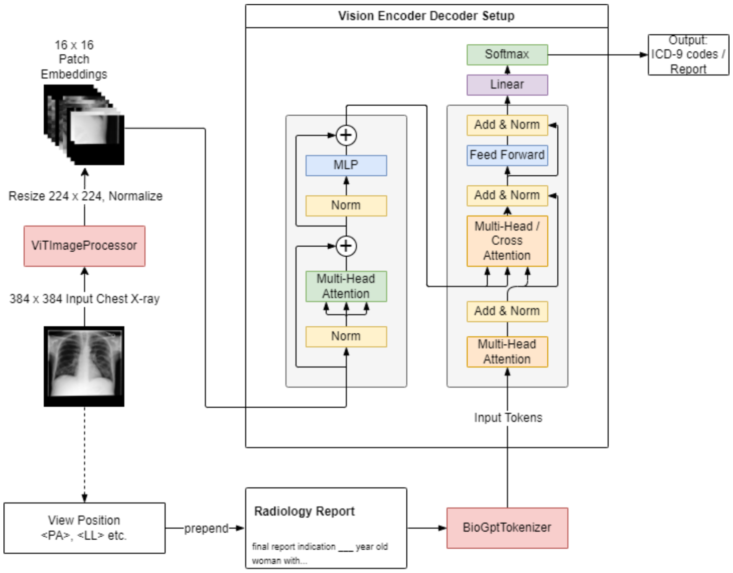

Figure 3.1: Example forward pass through our vision-biogpt hybrid (illustration inspired by Vaswani et al. (2017) and Dosovitskiy et al. (2021)). Inside the Vision Encoder Decoder block we have the ViT encoder (left) and BioGPT decoder (right)

3.0.2 Datasets & Preprocessing

The **M**edical **I**nformation **M**art for **I**ntensive **C**are (**MIMIC**) is a structured collection of deidentified electronic health records from patients from intensive care units at the Beth Israel Deaconess Medical Center in Boston, MA (Rogers et al. (2021), Johnson et al. (2016b)). Maintained by the Massachusetts Institute of Technology's Laboratory for Computation Physiology since March of 2000, the **MIMIC** database has seen four different iterations, focused on improving the quality and effectiveness of medical research using big data. All datasets are publicly available on the PhysioNet website: The Research Resource for Complex Physiologic Signals (Goldberger et al. (2000)), but require credentialed access. In order to obtain access to the **MIMIC** resources, one must create an account on the PhysioNet website, sign a data agreement for the specific datastore and potentially complete the Collaborative Institutional Training Initiative (CITI) Data or Specimens

Only Research program [8]. This is required to ensure ethical use of data and safe storage in order to protect the identity of the patients at hand, even if the data is deidentified. The main scope of MIMIC is to allow researchers to reproduce and enhance healthcare studies in manners that would otherwise be unattainable.

We conduct experiments on two datasets from the MIMIC database, namely **MIMIC-III** [9] and **MIMIC-CXR** [10].

3.0.2.1 MIMIC-III

MIMIC-III (Johnson et al. (2016b), Johnson et al. (2016a)) contains health records of over forty thousand patients who were hospitalized in intensive care units in a time span of 11 years, starting from 2001. These records are modeled as a relational database with 26 tables, which are linked by numerical identifiers that are in no way connected to the patients' identity. The dataset is extremely vast and carries a large proportion of information that is not relevant for our purpose (for example the caregiver who recorded the data or the date and time when the event was recorded). Hence, we only take into consideration three tables:

1. NOTEEVENTS: contains all deidentified clinical reports such as nursing and physician notes, ECG reports, imaging reports and discharge summaries. We are mostly interested in the discharge summaries, which we will use as textual input data in some of our experiments.

2. DIAGNOSES_ICD: holds all hospital assigned diagnoses in ICD-9 format and will be used for training our model for classification tasks.

3. PROCEDURES_ICD: records medical procedures performed on the patient and according ICD-9 classification as a result of said procedure

We download the appropriate tables in CSV format and follow a similar preprocessing approach as Dai et al. (2022)[11], which in turn is based on the work of Mullenbach et al. (2018) [12] and take the next steps towards converting the data to a more suitable format for our experiment:

[8] training available on the CITI program official website

[9] MIMIC-III available on the physionet website

[10] MIMIC-CXR available on the physionet website

[11] code available on github

[12] code available on github

1. Firstly, before merging the two ICD-labeled tables, we insert periods in the labels to better distinguish between diagnoses and procedures, since we do not want to have collisions. Generally, a period is added after the first two digits for diagnoses and after the first three digits for procedures.

2. Secondly, we concatenate the procedures and diagnoses and filter out those without a corresponding discharge summary from the NOTEEVENTS table.

3. Thirdly, we apply some preprocessing steps to the raw discharge summaries. Punctuation and numeric-only tokens are removed and quantities are normalized to 0 (example: 100 is removed, but 50mg becomes 00mg) and we convert all tokens to lowercase.

4. We then sort both the reports and ICD-labeled dataframes by their identifiers (SUBJECT_ID - the ID of any given patient and HADM_ID - the ID of any given patient's ICU stay), concatenate them in a single file and only keep the examples that were labeled with at least one of the top 50 most frequent ICD labels to make the classifier's life easier.

5. Lastly, we create "train", "test" and "development" splits from the according split files available on Mullenbach's github (Mullenbach et al. (2018)).

After all this is said and done, we end up with 8066 examples for the train set, 1573 examples for the development set and 1729 examples for the test set.

3.0.2.2 MIMIC-CXR

MIMIC-CXR is a subset of **MIMIC-IV** [13] (Johnson et al. (2023a), Johnson et al. (2023b)) containing 227,835 imaging studies for 64,588 patients hospitalized at the Beth Israel Deaconess Medical Center Emergency Department in Boston MA between 2011 and 2016. Since each study can contain one or more image, there are a total of 377,110 x-rays available in the dataset. Each study usually contains a frontal and lateral view of the chest x-ray and is accompanied by free-text radiology reports that describe the findings of

[13] MIMIC-IV available on the physionet website

each image, written by practicing radiologists. As with **MIMIC-III** and all other MIMIC subsets, the data is completely deidentified to protect the privacy of patients (Johnson et al. (2019a), Johnson et al. (2019c)). We download the relevant reports from the **MIMIC-CXR Database** and link them with images from **MIMIC-CXR-JPG** (Johnson et al. (2019b)). We choose the latter because **MIMIC-CXR** images are stored in DICOM format, the standard within medical imagery, adding up to an extremely large 4.6 TB of data. This would be impossible to store with our current hardware, thus we opt for the JPG format, which may be less qualitative, but still stores all the relevant pixel information for our purpose. The JPG files amount to 557.6 GB of data, which is definitely much more manageable. In the end, we get 10 folders, each with 6500 sub-folders corresponding to patient identifiers. Each patient folder is then split into a different amount of subfolders representing clinical studies performed on said patient. Each study contains one to three chest x-rays and their according orientation and is assigned a radiology report that we download sepparately from the large **MIMIC-CXR** database. The rest of the data is present in compressed CSV format within the following files:

- split: contains mappings of studies to an according ”train” or ”test” split.
- chexpert & negbio: contain the diagnosis labels for each study, assigned with the chexpert and negbio labelers respectively. We will take a closer look at the format of the labels in the next paragraph.
- metadata: contains more information regarding each study, but most importantly the view orientation of each chest x-ray.
- studies: holds all radiology reports.
- records: this is the file that glues everything together and presents the link between studies, labels and metadata (all other tables can be joined together using this one).

As noted in **MIMIC-CXR-JPG**'s README, the labels are structured in a very distinct fashion. Both chexpert and negbio files contain two identifiers related to an individual patient and study and 14 more columns with values ∈(-1, 0, 1, missing). These columns are named according to the disease they classify, which can be one of: Atelectasis, Cardiomegaly, Consolidation, Edema, Enlarged Cardiomediastinum, Fracture, Lung Lesion, Lung Opacity, Pleural Effusion, Pneumonia, Pneumothorax, Pleural Other, Support Devices, No Finding. The values in each of these columns have very specific meaning:

- 1.0: Represents a positive mention of the label in the associated study and its presence in one or more of the images.
- 0.0: Shows a negative mention of the label in the associated study and that it should not be present in any of the images.
- -1.0: Shows uncertainty or ambiguity of the mentioned label. It is unclear whether or not the disease exists or not.
- Missing (empty column): No mention was made about the label at hand in the report.

Due to the ambiguity of the final two cases, we will not take into considerations these values and will only look at examples that have been labeled with at least one value of 1.0 or 0.0.

For preprocessing this new data, we take the following measures:

1. The JPG files in the dataset are of very high resolution (3000 x 2000px), which is impossible for models to process as it would result in insane amounts of computations that our current algorithms cannot handle. Thus, we first resize the images to 384 x 384 and preserve their scale in order not to alter the shapes within each chest x-ray, as that could create problems with identifying diseases. We choose this resolution as it is more suited for the Vision Transformer, and the larger one of the two pre-trained checkpoints we discussed in Section 3.0.1.
2. We read in all CSV data and join the studies with the chexpert labels (we choose chexpert over negbio arbitrarily, although 2414 studies are missing labels in the chexpert table which should incline us to also perform experiments with negbio labeling). We convert the 14 label-columns to one column made of lists of length 14 with the same values. We apply the same text preprocessing steps to the reports here as we did with the ones in **MIMIC-III**. We now have a dataframe of subject and study identifiers, text and according labels from the chexpert labeler.

3. We then convert the labels to a new, original format. We double the size of the list and consider the first half to be representative of the ”positive” encoding, while the last 14 positions represent the ”negative” encoding. To be more specific, the list of diseases is sorted and each disease code is assigned two positions in our label encoding. Once in the first half, and once after exactly 14 positions (basically, we take the sorted list of diseases and map it to a new list twice). In the first half, all diseases with a value of 1.0 (positive mention) get to keep it; all other values are normalized to 0 (”positive” encoding). In the second half, all diseases with a value of 0.0 (negative mention) are transformed into 1's and all other values are normalized to 0 (”negative” encoding). For example, a collection of labels like $[1.0, -1.0, 0.0, -, -, -, -, -, -, -, -, -, -, -]$ will be mapped to:

$$[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$$

4. We also join the records and metadata tables to obtain the view position of each chest x-ray and attach it to the labeled reports

5. Lastly, we attach split information to each of our entry and obtain two dataframes with the same columns for training and testing our models. The train set consists of 212,098 entries and the test set of 3131. We have removed all entries without at least a 1 value in its label encoding.

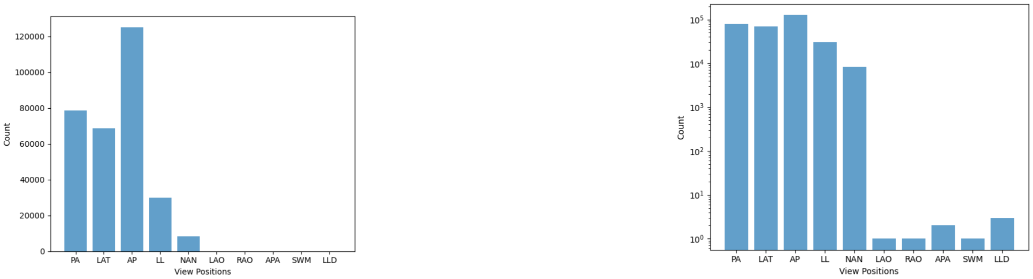

Figure 3.2: Distribution of view positions in our training data split, linear scale (left), log scale (right)

We analyze the distribution of view positions in our training data to see if we can make any assumptions about the images that do not have an orientation assigned to them (represented as NAN). We notice that even though the number is relatively small compared to the other large groups, it is still a considerable amount in the range of 10^4, while some of the other ones occupy a very small percentage. We decide to keep all images regardless of orientation and feed the model a custom learnable token alongside the radiology report to specify the view position. Arguments could be made towards filtering out the examples with an orientation of left anterior oblique (LAO), right anterior oblique (RAO), antero-posterior axial, swimmers (SWM) and postero-anterior lld (LLD) or simply assigning them one of the dominant classes that is closer to their orientation (since you can only position an image in a limited amount of ways), but we wanted to evaluate how the model can generalize for these ”outstanding” examples.

In the end, we obtain a structured dataframe containing the radiology reports, related images, their view positions and label encodings. When feeding these to a model, each report will be forwarded alongside one of the relevant images, as handling the different cases where a report has different numbers of images assigned to it would be too difficult from a technical perspective and we would have to make continuous assumptions regarding the data. This way, we ensure that the model sees each report once and assigns exactly one image to it to help it better understand the relation between them.

Chapter 4

Experimental setup

4.0.1 Experimental Setup

We run four different experiments in scope of answering the research questions 1 presented in Chapter 1.

1. We first reproduce the experiments of Dai et al. (2022) on the **MIMIC-III** (Johnson et al. (2016b)) dataset to establish a baseline of the BioGPT (Luo et al. (2022)) model for the task of ICD-9 classification and find out how well it performs compared to the Longformer (Beltagy et al. (2020)), RoBERTa (Liu et al. (2019)) and other models reported by Dai et al. (2022). We expect similar results considering BioGPT is a model built for biomedical tasks, but it has a limited capability of processing long documents, which Longformer aims to remove.

2. We then conduct unsupervised pre-training on the BioGPT language model on all of the **MIMIC-III** discharge summaries to obtain new weights that may or may not improve classification performance for ICD-9 coding. We expect a similar increase to what Dai et al. (2022) report in their task-adaptive-pre-training experiments.

3. We leverage these newly found weights to verify BioGPT's performance on **MIMIC-CXR** (Johnson et al. (2019a)), in an image-blind setup and with a different labeling system. We expect the model to perform at least as well as the previous experiment.

4. We add a cross-attention layer to BioGPT's architecture and use it as a text-decoder in a vision encoder-decoder setup, with the Vision

Transformer (Dosovitskiy et al. (2021)) as image-encoder, and assess its performance on the **MIMIC-CXR** database. We could not find reports of similar setups to compare our results with for **MIMIC-CXR**, thus we do not know what to expect in this situation and hope that our results for this experiment will inspire other researchers in their works.

All our experiments were executed on an NVIDIA®Titan RTX GPU with 24GB of video RAM from which we have assigned a set number of GB for each experiment (in truth, we should have allocated more for more efficient batching). The models are trained to minimize the binary cross-entropy loss using an AdamW optimizer (Loshchilov and Hutter (2019)) with a learning rate of $5e-5$ for classification and $2e-5$ for pre-training. Due to limited time and GPU-usage capability, we only run the experiments for 1 epoch each, apart from pre-training which we run for 6 epochs. We believe that the pre-training is very important for finding new and appropriate weights, so we allocate more time and focus to this step (also, we choose 6 epochs to mimic the process of Dai et al. (2022)). Due to limited memory allocation, we use batch sizes of 2-4 depending on the experiment and setup and use 4-16 gradient accumulation steps resulting in total batch sizes of 16 and 32 ($per\_device\_batch\_size \times n\_gpu \times gradient\_accumulation\_steps$). For most experiments we use the base variant of BioGPT with ≈347 million trainable parameters, while for the BioGPT-ViT hybrid, the end setup has ≈522 million parameters. The baseline experiment is the fastest to run and is executed in about 1 hour of GPU time (less amount of data), while all other experiments took from 20 to 38 GPU hours. Gradient clipping is applied after $nr\_gradient\_accumulation\_steps$ steps to avoid exploding gradients. The following table shows the number of examples in each split for each experiment.

Experiment	Train	Dev	Test
BioGPT Baseline	8066	1573	1729
BioGPT TAPT	47723	1631	3372
BioGPT CXR	212098	-	3131
BioGPT-ViT CXR	212098	-	3131

Table 4.1: Data splits for each experiment

All our experiments are based on the code from Dai et al. (2022)[1] and Ramos et al. (2023)[2]. The implementation is done in python, mostly using the Pytorch (Paszke et al. (2019)) library and HuggingFace API (Wolf et al. (2020)). The training and evaluation pipeline is based on Dai et al. (2022) and the cross-attention layer and vision encoder decoder model are based on Ramos et al. (2023), both of which use pre-trained models and code from the huggingface library.

4.0.1.1 Task-Adaptive Pre-Training

Dai et al. (2022) argue that continued unsupervised pre-training on data specific to a task is a promising step towards improving the performance of an already pre-trained model on said downstream task. They also show a valuable increase in performance when they perform task-adaptive pre-training on Longformer (Beltagy et al. (2020)) and RoBERTa (Liu et al. (2019)) on **MIMIC-III**. We observe a similar increase in performance for BioGPT and proceed to using the TAPT weights for further experiments.

4.0.1.2 Fine-Tuning

We fine-tune our models on multi-label sequence classification tasks, where the objective is giving an accurate diagnosis for a clinical study, be it an ICD-9 code or a custom label mapping to disease names from the **MIMIC-CXR** dataset.

[1] code available on github

[2] code available on github

As opposed to usual single-label classification, multi-label classification is the task of assigning zero or more possible classes to an input item. This is generally done by minimizing a Binary Cross Entropy Loss function. Assuming y to be a target label and $\hat{y}$ the predicted probability of positively assigning y, binary cross entropy is calculated using the following formula:

$$BCE(y, \hat{y}) = -(y \log \hat{y} + (1 - y) \log(1 - \hat{y}))$$

and averaged for all n classes in the dataset:

$$BCE = \frac{1}{n} \sum_{i=1}^{n} BCE(y_i, \hat{y_i})$$

4.0.1.3 Evaluation & Metrics

Evaluation is done by computing various metrics on the trained models to assess their performance on downstream tasks. The most common metric used for multi-label classification is the F1-score and we chose Micro F1 as reference point for our models. We also compute and report a handful of other metrics[3]. Let TP_i, FP_i, FN_i, TN_i represent true positives, false positives, false negatives and true negatives for a class i, respectively and n is the number of classes. We compute metrics using the formulas listed below:

1. Micro and Macro precision:

$$micro\_p = \frac{\sum_i TP_i}{\sum_i (TP_i + FP_i)}$$

$$macro\_p = \frac{1}{n} \sum_i \frac{TP_i}{TP_i + FP_i}$$

2. Micro and Macro recall

$$micro\_r = \frac{\sum_i TP_i}{\sum_i (TP_i + FN_i)}$$

$$macro\_r = \frac{1}{n} \sum_i \frac{TP_i}{TP_i + FN_i}$$

[3] metrics computed using the code from the trldc github repository of Dai et al. (2022)

3. Micro-F1, Macro-F1

$$micro\_f1 = 2 \cdot \frac{micro\_p \cdot micro\_r}{micro\_p + micro\_r}$$

$$macro\_f1 = 2 \cdot \frac{macro\_p \cdot macro\_r}{macro\_p + macro\_r}$$

4. Micro-AUC, Macro-AUC
AUC score is computed by drawing a receiver operating characteristic (ROC) curve, which plots the true positive rate (recall) against the false positive rate ($\frac{FP}{FP+TN}$). The AUC score is calculated as the area under this curve. Macro AUC score is computed as the average of AUC scores for each class independently.

5. Precision@k for k ∈[5, 8, 15]

$$p@k = \frac{\text{\#TP in top k predictions}}{k}$$

6. Perplexity score
We only compute perplexity when performing continued unsupervised pre-training on BioGPT for text generation. Perplexity is a metric commonly used to evaluate a language model's ability to predict a future tokens. In our case, we measure how well BioGPT predicts future tokens for generating elongated radiology reports. The mathematical formula for perplexity score of a tokenized sequence $X = (x_0, x_1, \cdots, x_n)$ is (taken from huggingface documentation):

$$PPL(x) = exp(-\frac{1}{n}\sum_{i}^{t} \log p_\theta(x_i|x_{<i}))$$

where $\log p_\theta(x_i|x_{<i})$ is the log-likelihood of the ith token conditioned on the preceding tokens. Naturally, this can be considered an estimation of the model's capability to predict uniformly across the vocabulary of tokens.

7. BlEU score[4]
The BLEU (BiLingual Evaluation Understudy) score measures the sim-

[4] Mathematical formulas and explanation based on Understanding the BLEU Score

ilarity of the machine-generated text (candidate) to a series of qualitative references.

$$\text{BLEU} = \min(1, exp(1 - \frac{reference\_length}{output\_length}))(\prod_{i=1}^{4} precision_i)^{\frac{1}{4}}$$

where

$$precision_i = \frac{\sum_{snt \in Cand-Corpus} \sum_{i \in snt} min(m^i_{cand}, m^i_{ref})}{w^i_t = \sum_{snt' \in Cand-Corpus} \sum_{i' \in snt'} m^i_{cand}}$$

Here, m^i_{cand} and m^i_{ref} refer to the count of i-grams in the candidate that match the reference and the count of i-grams in the reference respectively, while w^i_t represents the total number of i-grams in the candidate. We compute BLEU scores to assess the correctness and coherence of report generation when faced with an image-only task.

Chapter 5

Results

In this chapter we report and examine the results of each of our experiments. For those that are relevant, we also report comparative metrics from similar experiments with different models (mainly from Dai et al. (2022)).

5.0.1 BioGPT on MIMIC-III ICD-9 classification baseline metrics

Our first experiment aims to establish a baseline of BioGPT's potential to accurately classify clinical reports using the ICD-9 coding system. We use the same data splits from **MIMIC-III** provided by Dai et al. (2022) that we have shown in Table 4.1, perform ICD-9 classification and compute various metrics described in Section 4.0.1.3. We compare our results to those of Dai et al. (2022) on their base classification experiment on the **MIMIC-III** development set using RoBERTa (Liu et al. (2019)) and Longformer (Beltagy et al. (2020)). We observe comparative performance, although somewhat worse than the point of reference. Table 5.1 shows that BioGPT in its base form is ever-so-slightly inferior to Longformer, even when the latter is also limited to 1024 tokens. Interestingly, BioGPT performs better in terms of AUC score, but worse in terms of F1 score.

	AUC		F_1		
	Macro	Micro	Macro	Micro	P@5
BioGPT (1024)	**84.0**	**87.9**	**37.5**	**52.0**	**54.9**
RoBERTa (512)	81.6	85.0	43.2	53.9	54.0
Longformer (512)	81.4	85.1	39.2	52.2	53.3
Longformer (1024)	**83.6**	**87.3**	**43.2**	**56.3**	**56.5**
Longformer (2048)	86.5	89.8	48.2	60.5	59.4
Longformer (4096)	88.4	91.5	53.1	64.0	62.0
Baselines processing up to 512 tokens					
First	83.0	86.0	47.0	56.1	55.4
Random	82.5	85.4	42.7	51.1	52.3
Informative	82.7	85.8	46.4	55.2	54.8

Table 5.1: Results on MIMIC-III ICD-9 classification for vanilla models & baselines

Dai et al. (2022) also report metrics for three baseline models with a maximum sequence length of 512, that use different strategies for reading text:

1. First: process first 512 tokens
2. Random: process 512 tokens at random
3. Informative: identify informative tokens using TF-IDF encodings

None of these three methods stand out, but processing the first tokens seems to be a good approach, making this the reason why we have also chosen to process the first tokens in each document, more so than any other strategy.

5.0.2 BioGPT Task-Adaptive-Pre-Training on MIMIC-III & ICD-9 classification

Pursuing the path of Dai et al. (2022), we continue pre-training BioGPT on all discharge summaries available in the **MIMIC-III** dataset. We conduct this experiment in an unsupervised fashion and allow the model to go through a large amount of reports catered for our specific task. We run this process over 6 epochs with an effective batch size of 32 (batch size of 4 × 8 gradient accumulation steps). This results in the 47,723 examples being processed in 8946 steps. We compute perplexity on the development split around every 300 steps, or after the model has processed ≈10 million tokens ($steps \times batch\_size \times sequence\_length = 300 \cdot 32 \cdot 1024 = 9,830,400$ tokens) and follow its evolution. Table 5.2 shows an evaluation of the perplexity metric before and after conducting pre-training:

	Before TAPT	After TAPT
Perplexity	144.6246	6.6399

Table 5.2: Perplexity on **MIMIC-III** test set before and after Task-Adaptive-Pre-Training

We observe a decline in perplexity after (and during) Task-Adaptive-Pre-Training, indicating that the process has been successful and we should expect better performance with the newly found weights. Table 5.3 shows a comparison between the base models and their Task-Adaptive-Pre-Trained versions on the same task of ICD-9 labeling on **MIMIC-III** as the previous experiment.

		AUC		F_1		
		Macro	Micro	Macro	Micro	P@5
BioGPT (1024)	Vanilla	84.0	87.9	37.5	52.0	54.9
	TAPT	85.8	89.4	40.7	55.8	58.3
RoBERTa (512)	Vanilla	81.6	85.0	43.2	53.9	54.0
	TAPT	82.3	85.5	48.8	56.7	55.3
Longformer (4096)	Vanilla	88.4	91.5	53.1	64.0	62.0
	TAPT	90.3	92.7	60.8	68.5	64.8

Table 5.3: The effects of Task-Adaptive-Pre-Training on **MIMIC-III**

Whilst still lower than Longformer and RoBERTa, BioGPT's performance increases at a similar rate. We observe a 3.2 increase in Macro F1 and 3.8 in Micro F1. Part of the reason that Longformer and RoBERTa see slightly larger performance increase could be caused by the fact that they have not initially been pre-trained on biomedical data. BioGPT is already used to medical context and language and, while further pre-training is very helpful for our downstream task, its effects are not as prominent as with those which previously did not have any knowledge of the biomedical domain and are now shown many examples of it.

5.0.3 Comparison to state-of-the-art on MIMIC-III

	AUC		F1		
Experiment	Macro	Micro	Macro	Micro	P@5
BioGPT-Base dev	84.0	87.9	37.5	52.0	54.9
BioGPT-Base test	83.6	87.4	37.1	51.1	55.1
BioGPT-TAPT dev	85.8	89.4	40.7	55.8	58.3
BioGPT-TAPT test	85.3	88.9	40.3	55.3	58.1
State-of-the-art					
CAML (Mullenbach et al. (2018))	57.6	63.3	88.4	91.6	61.8
PubMedBERT (Gu et al. (2021))	63.3	68.1	88.6	90.8	64.4
GatedCNN-NCI (Ji et al. (2021b))	62.9	68.6	91.5	93.8	65.3
LAAT (Vu et al. (2020))	66.6	71.5	92.5	94.6	67.5
MSMN (Yuan et al. (2022))	**68.3**	**72.5**	**92.8**	**94.7**	**68.0**

Table 5.4: Comparison of our experiments on MIMIC-III against state-of-the-art models

We report a general pattern of high AUC score and low F1 for our BioGPT variants. This entails that the model predicts a high amount of true positives but fails to classify other examples correctly and generates a high number of false negatives as well. Sadly, our methods do not come any close to previous state-of-the-art models. Yuan et al. (2022) hold this title with their RNN-based Multiple Synonyms Matching Network. Dai et al. (2022) setups show comparative performance to the state-of-the-art, but also do not surpass it. Thus, further research needs to be conducted to find a Transformer-based setup that can achieve even greater performance.

5.0.4 BioGPT image-blind approach on MIMIC-CXR

With the newly established pre-trained weights, we can now fine-tune our version of BioGPT to a disease classification task with a new labeling system present in **MIMIC-CXR**. Before setting up the ViT-BioGPT hybrid, we run an image-blind experiment using just the radiology reports from **MIMIC-CXR** to examine how well BioGPT performs with this new classification setup and establish a baseline of comparison for our future image-aware experiment.

	AUC		F_1		
	Macro	Micro	Macro	Micro	P@5
BioGPT-CXR	99.5	99.8	88.3	97.3	56.1

Table 5.5: BioGPT TAPT disease classification on **MIMIC-CXR** radiology reports (image-blind approach)

Table 5.5 depicts our best results so far, with our pre-trained BioGPT achieving near-perfect AUC and Micro F1 scores. This shows that the model works specifically well with the format of the studies from **MIMIC-CXR** and our customized labeling format. Reports from **MIMIC-CXR** are considerably smaller than the ones from **MIMIC-III**, allowing our sequence-length-limited-model to lose less (if at all) context and get more information of the text at hand. The possible number of target labels is also reduced and, given the nature of **MIMIC-CXR** data, the labels themselves are more closely related to what the report states (remember, **MIMIC-CXR**'s label system shows the presence, absence or uncertainty of the appropriate label being mentioned in the report).

All these factors contribute to making BioGPT better suited for this task and dataset, resulting in this amazing performance which we hope can inspire other researchers to pursue the usage of BioGPT in such setups.

In order to assess the quality of these results, we might run this experiment in two different setups, taking the positive and negative encodings sepparately and jointly learn from them. Since our model learns from both encodings at the same time, it may be easier for it to understand the relation between each label and its mentioning in the text.

5.0.5 BioGPT-ViT Vision Encoder Decoder on MIMIC-CXR

Our final experiment studies BioGPT's classification capabilities when also being prompted with chest x-rays, alongside the usual radiology reports. We implement a new attention block for BioGPT that supports cross-attention, following examples from Ramos et al. (2023) and use the Vision Transformer (Dosovitskiy et al. (2021)) as encoder for the images.

	AUC		F_1		
	Macro	Micro	Macro	Micro	P@5
BioGPT-CXR	99.5	99.8	88.3	97.3	56.1
BioGPT-ViT-CXR	99.3	99.6	44.17	54.4	56.0

Table 5.6: BioGPT TAPT disease classification on **MIMIC-CXR** radiology reports

Table 5.6 shows a much lower performance when pairing radiology reports with their corresponding chest x-rays, comparable to the metrics we have seen in our **MIMIC-III** experiments. Seeing these results, we can assume that BioGPT is not well suited for processing image data. It was not designed with this scope and its pre-training data has little to no reference to medical imagery. In order to address this, there are two possible approaches:

1. Pre-train this BioGPT-ViT setup from scratch using a large corpus of text-image pairs in biomedical domain, with a focus on x-rays.

2. Freeze the Vision Transformer Encoder and perform Task-Adaptive-Pre-Training on a freshly initialized BioGPT language model to again discover new weights with a better understanding of visual context. It

is, although, highly unlikely that this method would yield performances similar to what we have obtained in the previous experiment.

Chapter 6

Discussion & Analysis

Looking back at our experiments, we aggregate all our metric results in the following tables (ranked by micro-F1):

	Precision		Recall		F1		AUC	
Experiment	Macro	Micro	Macro	Micro	Macro	Micro	Macro	Micro
BioGPT-TAPT-CXR	88.7	97.2	88.5	97.4	88.3	97.3	99.5	99.8
BioGPT-TAPT dev	58.2	74.4	35.2	44.7	40.7	55.8	85.8	89.4
BioGPT-TAPT test	62.0	73.9	34.9	44.2	40.3	55.3	85.3	88.9
BioGPT-ViT-CXR	30.7	37.4	95.5	99.8	44.1	54.4	99.3	99.6
BioGPT-Base dev	58.6	72.8	31.9	40.4	37.5	52.0	84.0	87.9
BioGPT-Base test	57.3	71.8	31.7	39.7	37.1	51.1	83.6	87.4

Table 6.1: Evaluation Metrics for all experiments

Experiment	P@5	P@8	P@15
BioGPT-TAPT-CXR	56.1	36.5	19.5
BioGPT-TAPT dev	58.3	46.7	31.4
BioGPT-TAPT test	58.1	46.8	31.7
BioGPT-ViT-CXR	56.0	36.5	19.5
BioGPT-Base dev	54.9	44.4	30.5
BioGPT-Base test	55.1	44.5	31.0

Table 6.2: P@k for all experiments

Initially, we run ICD-9 classification on the vanilla (base) pre-trained BioGPT (Luo et al. (2022)) checkpoint and establish a baseline performance for our future experiments. The results are comparative to those of Dai et al. (2022) on the same data splits and task. This shows that our pipeline is working correctly and that our code is bug-free. Thus, we are confident to move on with our research.

We continue pre-training BioGPT on all discharge summaries of **MIMIC-III** (Johnson et al. (2016b)). We observe an increase in performance of $\approx 0.3 - 0.4$, which is also comparable to what Dai et al. (2022) report. Albeit a tad bit smaller progress, this was to be expected as BioGPT is already pre-trained on in-domain data, making this step of task adaptive pre-training less impactful. With Longformer, the performance increased at a much higher rate for **MIMIC-III** than it did on the other datasets that they have experimented on, which are not domain-specific. This was to be expected, as Longformer was not trained for biomedical focus and it should benefit much more when given the opportunity to learn new data, as opposed to BioGPT, who has already learnt contextual meanings and representations of biomedical text. Again, the results of this experiment show that our setup is working correctly and we can confidently run experiments on **MIMIC-CXR**, even if we have no point of comparison.

Our third installment gives us the best results yet, with a whopping precision and micro-F1 of $\approx$97%. This is most surprising, as we use a custom labeling system described in Section 3.0.2.2 and were unsure how well the model will be able to predict with this setup. We believe there are multiple factors that make these results possible:

1. Reports in **MIMIC-CXR** are much shorter than discharge summaries of **MIMIC-III**, allowing for our model to process more context. Dai et al. (2022) have shown that being able to process more tokens leads to increased performance. Conversely, if we are not able to process more than 1024 tokens, we can expect better performance when our input data is shorter than or equal to 1024 tokens. Since we can only read 1024 tokens, being able to read a higher percentage (if not all) of the text can greatly help our models.

2. Labels in **MIMIC-CXR** are much more closely related to their corresponding reports. Each label that we consider shows the presence or absence of mentioning the disease in the report. This can mean that BioGPT learns to predict a certain class if its name verbosely appears in the text, which is much easier to understand rather than having to assume a disease code based on symptoms or any other type of ambiguous description. This can also be proven by the high recall (true positive rate); our model predicts a high number of true positives, meaning it is highly capable of understanding that a label should be categorized as positive if its name is present in the report.

When adding image support to the reports from **MIMIC-CXR**, we observe a high loss in performance and return to numbers that more closely resemble the baseline performance on **MIMIC-III**. The model maintains a high recall and AUC score, but scores much lower in precision and F1. This shows that our model might have become biased towards true positives and could in some way be overfitting. The model is returning a high number of true positives, but with a low accuracy, indicating an even higher number of false negatives. We assume that this issue stems from the fact that BioGPT is a language model built specifically for text-generation purposes and has zero knowledge of what an image is. Another reason might be the low amount of training data (only 212,098 examples). While it may have seen descriptive text of medical imagery during pre-training, not learning how to work with pixel values can be detrimental. We doubt the issue is related to the Vision Transformer, as it has shown amazing capabilities in understanding medical imagery in many other scenarios (Henry et al. (2022)).

In the end, we examine the text generation capabilities of the Vision Encoder Decoder setup using both the task-adaptive-pre-trained weights and standard BioGPT pre-trained weights. Instead of feeding the language model text and have it extend the input with future tokens, we feed the encoder a chest x-ray from **MIMIC-CXR** and expect the decoder to generate a radiology report based on it. As we have seen the massive decrease in perplexity during task-adaptive-pre-training, we expect the generated report to be similar to and follow the format of **MIMIC-III** reports when using the corresponding weights. We examine this process for a few randomly selected examples from **MIMIC-CXR** dataset and compare the generated output to the original report. The vanilla BioGPT is proven to be of no use in this setup. Without textual context, the model becomes greatly confused and outputs some gibberish biomedical terms and random characters with no real meaning or coherence.

On the other hand, with the task-adaptive-pre-trained weights the model generates much more cohesive and coherent text that looks to be structured in the same format as **MIMIC-III** reports (for example, it mentions the admission date, discharge date, sex and service offered to the patient, with all numbers normalized to 0). However, even though the generated report is nicely organized, its informational content is completely lackluster. We do not observe any relevant mentions of diseases as expected in **MIMIC-CXR** reports, but rather a miss-classification and confusion with other diseases. We observe an average BLEU score of 0.25 from the examples we examined, confirming the terribly poor performance that we have noticed with the naked eye.

Had we conducted task-adaptive-pre-training on **MIMIC-CXR** reports instead of **MIMIC-III**, we should have gotten better performance in terms of BLEU score at this stage as the language model would be more closely fine-tuned to the textual format of **MIMIC-CXR**.

Chapter 7

Conclusions & Future Work

Transformer (Vaswani et al. (2017)) models have revolutionized the field of Natural Language Processing and have proven to be extremely strong in various applications. Large Language Models (LLMs) and Vision Models are amongst the top benefactors of this architecture, achieving state-of-the-art results across the board for many downstream tasks. These models are trained on large collections of data (be it text, images or a combination of both) and then fine tuned for any specific task, method which has provided successful research in many different areas and domains.

In our research, we experiment using a combination of two Transformer-based models in a novel setup where we combine radiology reports with corresponding chest x-rays with the aim of evaluating BioGPT (Luo et al. (2022)), a pre-trained language model specific to the biomedical domain, in its capability of generating accurate clinical reports and classify them with a disease code.

We use data from two sources comprised of deidentified hospital records (radiology reports, discharge summaries, chest x-rays) collected over the past two decades from Beth Israel Deaconess Medical Center, Boston MA, in **MIMIC-III** (Johnson et al. (2016b)) and **MIMIC-CXR** (Johnson et al. (2019a)).

We conduct a series of experiments, initially following the work of Dai et al. (2022) to establish a baseline of BioGPT and then innovate by using this language model as text-decoder alongside a Vision Transformer (Dosovitskiy et al. (2021)) image-encoder. By doing so, we provide the following answers to the research questions 1 that we defined in Chapter 1.

1. BioGPT in its base pre-trained form achieves comparable performance to other Transformer-based models that are designed to perform long document classification and have been studied on **MIMIC-III**. We observe slightly lower performance than what Dai et al. (2022) report and definitely lower than the state-of-the-art (Yuan et al. (2022)) on this task. This was to be expected, as **MIMIC-III** discharge summaries usually contain a large number of tokens (over 2000), while our language model is only capable of processing 1024 at a time. By choosing to truncate the text after the first 1024 tokens, the model loses a high percentage of contextual information, resulting in poorer performance. These results show that BioGPT can work decently well when faced with ICD-9 classification.

2. By initializing BioGPT from its pre-trained checkpoint and conducting continued pre-training on the **MIMIC-III** corpus of discharge summaries in an unsupervised manner, we aim to get new weights for the model that are more understanding of the **MIMIC** format of reports. We observe that classification performance increases after this process by a similar ratio to what Dai et al. (2022) report. While still not as good as the other models we use for comparison, this new pre-trained checkpoint for BioGPT will serve as initializer for the rest of our experiments.

3. We define a new labeling system for the 14 diseases from **MIMIC-CXR** and test BioGPT's classification performance on this new set of radiology reports. We observe impressive results with very high precision, recall, AUC and F1 scores and can confidently say that shorter reports benefit our model greatly. We are pleasantly surprised by how well the model performs on our label encoding (explained in Section 3.0.2.2) and encourage further research in this direction.

4. We incorporate chest x-rays alongside the usual reports during classification training in a vision-encoder-decoder setup. We obtain worse results than our previous outstanding experiment and conclude that BioGPT is better at text classification and generation and only gets confused when it attends to image features as generated by the Vision Transformer.

All our experiments are targeted towards the performance of BioGPT and ViT. Many other researchers have implemented vision-encoder-decoder architectures (Li et al. (2022b), Ramos et al. (2023)) with high success, but, as of writing this thesis, this combination of models was not priorly studied. We hope our work will contribute to future researchers who aim to study new biomedical models and approaches for classification and generation of radiology reports.

We aim to continue this research ourselves by conducting further experiments:

1. Reproduce all our past experiments using the Large variant of BioGPT [1] with a larger number of trainable parameters and, theoretically more powerful than the Base variant that we have been working with.

2. Reproduce all our past experiments using a larger resolution for the images (i.e. a different ViT checkpoint [2]). We believe that in combination with $BioGPT_{Large}$ this new setup could achieve greater results when trained on text-image pairs.

3. Conduct additional unsupervised pre-training on the whole BioGPT-ViT setup and establish radiology report generation capabilities in a text-blind approach.

4. Separate our image-blind experiment into two tasks of classifying using only the positive and negative encodings respectively. This should be done to assess the quality of our surprising results and test whether the model would be capable of achieving performance close to this without having both encodings in context.

5. Examine different methods for our setup to jointly learn the two tasks of classification and generation when using image-text pairs.

All this said and done, we believe BioGPT is a strong model, capable of producing accurate radiology reports and classify them using disease codes and encourage its development and evolution. With the latest rise of technology and artificial intelligence, we believe that new versions of BioGPT, based on newer GPT-architectures (OpenAI (2023)) or other cutting-edge large language models (Touvron et al. (2023)) could provide great value in the world of radiology and could be the next large medical discovery.

[1] $BioGPT_{Large}$ available on the huggingface website

[2] $ViT_{384patch16}$ available on the huggingface website

Bibliography

Farah Ali, Samantha Harrington, Stephen Kennedy, and Sarwat Hussain. Diagnostic radiology in liberia: A country report. *Journal of Global Radiology*, 1, 11 2015. doi: 10.7191/jgr.2015.1020.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey Hinton. Layer normalization, 2016.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate, 2016.

Iz Beltagy, Kyle Lo, and Arman Cohan. Scibert: A pretrained language model for scientific text, 2019.

Iz Beltagy, Matthew Peters, and Arman Cohan. Longformer: The long-document transformer, 2020.

Noam Brown, Anton Bakhtin, Adam Lerer, and Qucheng Gong. Combining deep reinforcement learning and search for imperfect-information games, 2020a.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020b.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers, 2020.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation, 2014.

Xiang Dai, Ilias Chalkidis, Sune Darkner, and Desmond Elliott. Revisiting transformer-based models for long document classification, 2022.

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.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. Glm: General language model pretraining with autoregressive blank infilling, 2022.

Richárd Farkas and György Szarvas. Automatic construction of rule-based icd-9-cm coding systems. *BMC bioinformatics*, 9 Suppl 3:S10, 02 2008. doi: 10.1186/1471-2105-9-S3-S10.

Shang Gao, Mohammed Alawad, Michael Young, John Gounley, Noah Schaefferkoetter, Hong-Jun Yoon, Xiao-Cheng Wu, Eric Durbin, Jennifer Doherty, Antoinette Stroup, Linda Coyle, and Georgia Tourassi. Limitations of transformers on clinical text classification. *IEEE Journal of Biomedical and Health Informatics*, PP:1–1, 02 2021. doi: 10.1109/JBHI.2021.3062322.

Ary Goldberger, Luís Amaral, L. Glass, Shlomo Havlin, J. Hausdorg, Plamen Ivanov, R. Mark, J. Mietus, G. Moody, Chung-Kang Peng, H. Stanley, and Physiotoolkit Physiobank. Components of a new research resource for complex physiologic signals. *PhysioNet*, 101, 01 2000.

Alex Graves. Generating sequences with recurrent neural networks, 2014.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare*, 3(1):1–23, oct 2021. doi: 10.1145/3458754. URL https://doi.org/10.1145%2F3458754.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

Emerald U. Henry, Onyeka Emebob, and Conrad Asotie Omonhinmin. Vision transformers in medical imaging: A review, 2022.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9:1735–80, 12 1997. doi: 10.1162/neco.1997.9.8.1735.

Xiaowei Hu, Zhe Gan, Jianfeng Wang, Zhengyuan Yang, Zicheng Liu, Yumao Lu, and Lijuan Wang. Scaling up vision-language pre-training for image captioning, 2022.

Shaoxiong Ji, Matti Hölttä, and Pekka Marttinen. Does the magic of BERT apply to medical code assignment? a quantitative study. *Computers in Biology and Medicine*, 139:104998, dec 2021a. doi: 10.1016/j.compbiomed.2021.104998. URL https://doi.org/10.1016%2Fj.compbiomed.2021.104998.

Shaoxiong Ji, Shirui Pan, and Pekka Marttinen. Medical code assignment with gated convolution and note-code interaction. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. Association for Computational Linguistics, 2021b. doi: 10.18653/v1/2021.findings-acl.89. URL https://doi.org/10.18653%2Fv1%2F2021.findings-acl.89.

Alistair Johnson, Tom Pollard, and Roger Mark. Mimic-iii clinical database (version 1.4). https://doi.org/10.13026/C2XW26, 2016a. PhysioNet.

Alistair Johnson, Tom Pollard, Lu Shen, Li-wei Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Celi, and Roger Mark. Mimic-iii, a freely accessible critical care database. *Scientific Data*, 3:160035, 05 2016b. doi: 10.1038/sdata.2016.35.

Alistair Johnson, Tom Pollard, Seth Berkowitz, Nathaniel Greenbaum, Matthew Lungren, Chih-ying Deng, Roger Mark, and Steven Horng. Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific Data*, 6:317, 12 2019a. doi: 10.1038/s41597-019-0322-0.

Alistair Johnson, Tom Pollard, Nathaniel Greenbaum, Matthew Lungren, Chih ying Deng, Yifan Peng, Zhiyong Lu, Roger Mark, Seth Berkowitz, and Steven Horng. Mimic-cxr-jpg, a large publicly available database of labeled chest radiographs, 2019b.

Alistair Johnson, Tom Pollard, Roger Mark, Seth Berkowitz, and Steven Horng. Mimic-cxr database (version 2.0.0). https://doi.org/10.13026/C2JT1Q, 2019c. PhysioNet.

Alistair Johnson, Lucas Bulgarelli, Tom Pollard, Steven Horng, Leo Celi, and Roger Mark. Mimic-iv (version 2.2). https://doi.org/10.13026/6mm1-ek67, 2023a.

Alistair Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, Li-wei Lehman, Leo Celi, and Roger Mark. Mimic-iv, a freely accessible electronic health record dataset. *Scientific Data*, 10:1, 01 2023b. doi: 10.1038/s41597-022-01899-x.

Michael I Jordan. Serial order: a parallel distributed processing approach. technical report, june 1985-march 1986. 5 1986. URL https://www.osti.gov/biblio/6910294.

Kamal Raj Kanakarajan, Bhuvana Kundumani, and Malaikannan Sankarasubbu. Bioelectra:pretrained biomedical text encoder using discriminators. pages 143–154, 01 2021. doi: 10.18653/v1/2021.bionlp-1.16.

Sarvnaz Karimi, Xiang Dai, Hamed Hassanzadeh, and Anthony Nguyen. Automatic diagnosis coding of radiology reports: A comparison of deep learning and conventional classification methods. 08 2017. doi: 10.18653/v1/W17-2342.

Bevan Koopman, Sarvnaz Karimi, Anthony Nguyen, Rhydwyn Mcguire, David Muscatello, Madonna Kemp, Donna Truran, Ming Zhang, and Sarah Thackway. Automatic classification of diseases from free-text death certificates for real-time surveillance. *BMC medical informatics and decision making*, 15:53, 07 2015. doi: 10.1186/s12911-015-0174-2.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation*, 1(4):541–551, 12 1989. ISSN 0899-7667. doi: 10.1162/neco.1989.1.4.541. URL https://doi.org/10.1162/neco.1989.1.4.541.

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, sep 2019. doi: 10.1093/bioinformatics/btz682. URL https://doi.org/10.1093%2Fbioinformatics%2Fbtz682.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation, 2022a.

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, 2022b.

Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. Oscar: Object-semantics aligned pre-training for vision-language tasks, 2020.

Hezheng Lin, Xing Cheng, Xiangyu Wu, Fan Yang, Dong Shen, Zhongyuan Wang, Qing Song, and Wei Yuan. Cat: Cross attention in vision transformer, 2021.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019.

Renqian Luo, Liai Sun, Yingce Xia, Tao Qin, Sheng Zhang, Hoifung Poon, and Tie-Yan Liu. BioGPT: generative pre-trained transformer for biomedical text generation and mining. *Briefings in Bioinformatics*, 23(6), 09 2022. ISSN 1477-4054. doi: 10.1093/bib/bbac409. URL https://doi.org/10.1093/bib/bbac409. bbac409.

James Mullenbach, Sarah Wiegreffe, Jon Duke, Jimeng Sun, and Jacob Eisenstein. Explainable prediction of medical codes from clinical text, 2018.

OpenAI. Gpt-4 technical report, 2023.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library, 2019.

Yifan Peng, Shankai Yan, and Zhiyong Lu. Transfer learning in biomedical natural language processing: An evaluation of bert and elmo on ten benchmarking datasets, 2019.

John Pestian, Chris Brew, Pawel Matykiewicz, DJ Hovermale, Neil Johnson, Bretonnel Cohen, and Wlodzislaw Duch. A shared task involving multi-label classification of clinical free text. In *Biological, translational, and clinical language processing*, pages 97–104, Prague, Czech Republic, June 2007. Association for Computational Linguistics. URL https://aclanthology.org/W07-1013.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. *OpenAI Technical Report*, 2018. URL https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Technical Report*, 2019. URL https://d4mucfpksywv.cloudfront.net/better-language-models/language_models_are_unsupervised_multitask_learners.pdf.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

Prajit Ramachandran, Niki Parmar, Ashish Vaswani, Irwan Bello, Anselm Levskaya, and Jonathon Shlens. Stand-alone self-attention in vision models, 2019.

Rita Ramos, Bruno Martins, Desmond Elliott, and Yova Kementchedjhieva. Smallcap: Lightweight image captioning prompted with retrieval augmentation, 2023.

Tal Ridnik, Emanuel Ben-Baruch, Asaf Noy, and Lihi Zelnik-Manor. Imagenet-21k pretraining for the masses, 2021.

Paul Rogers, Dong Wang, and Zhiyuan Lu. Medical information mart for intensive care: Foundation for the fusion of artificial intelligence and real-world data. *Frontiers in Artificial Intelligence*, 4, 2021. ISSN 2624-8212. doi: 10.3389/frai.2021.691626. URL https://www.frontiersin.org/articles/10.3389/frai.2021.691626.

Andrew Rosenkrantz, Danny Hughes, and Richard Duszak. The u.s. radiologist workforce: An analysis of temporal and geographic variation by using large national datasets. *Radiology*, 279:150921, 10 2015. doi: 10.1148/radiol.2015150921.

Andrew Rosenkrantz, Wenyi Wang, Danny Hughes, and Richard Duszak. A county-level analysis of the us radiologist workforce: Physician supply and subspecialty characteristics. *Journal of the American College of Radiology*, 15(4):601–606, 2018. ISSN 1546-1440. doi: https://doi.org/10.1016/j.jacr.2017.11.007. URL https://www.sciencedirect.com/science/article/pii/S1546144017314333.

David Rosman, Jean Nshizirungu, Emmanuel Rudakemwa, Crispin Moshi, Jean Tuyisenge, Etienne Uwimana, and Louise Kalisa. Imaging in the land of 1000 hills: Rwanda radiology country report. *Journal of Global Radiology*, 1, 03 2015. doi: 10.7191/jgr.2015.1004.

Sascha Rothe, Shashi Narayan, and Aliaksei Severyn. Leveraging Pre-trained Checkpoints for Sequence Generation Tasks. *Transactions of the Association for Computational Linguistics*, 8:264–280, 06 2020. ISSN 2307-387X. doi: 10.1162/tacl_a_00313. URL https://doi.org/10.1162/tacl_a_00313.

David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning internal representations by error propagation. In David E. Rumelhart and James L. Mcclelland, editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1: Foundations*, pages 318–362. MIT Press, Cambridge, MA, 1986.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units, 2016.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.

Thanh Vu, Dat Quoc Nguyen, and Anthony Nguyen. A label attention model for ICD coding from clinical text. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, jul

2020. doi: 10.24963/ijcai.2020/461. URL https://doi.org/10.24963%2Fijcai.2020%2F461.

Huiyu Wang, Yukun Zhu, Bradley Green, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Axial-deeplab: Stand-alone axial-attention for panoptic segmentation, 2020.

Qiang Wang, Bei Li, Tong Xiao, Jingbo Zhu, Changliang Li, Derek F. Wong, and Lidia S. Chao. Learning deep transformer models for machine translation, 2019.

Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks, 2018.

Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. Simvlm: Simple visual language model pretraining with weak supervision, 2022.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Huggingface's transformers: State-of-the-art natural language processing, 2020.

Bichen Wu, Chenfeng Xu, Xiaoliang Dai, Alvin Wan, Peizhao Zhang, Zhicheng Yan, Masayoshi Tomizuka, Joseph Gonzalez, Kurt Keutzer, and Peter Vajda. Visual transformers: Token-based image representation and processing for computer vision, 2020.

Pengtao Xie and Eric Xing. A neural architecture for automated ICD coding. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1066–1076, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1098. URL https://aclanthology.org/P18-1098.

Michihiro Yasunaga, Jure Leskovec, and Percy Liang. Linkbert: Pretraining language models with document links, 2022.

Zheng Yuan, Chuanqi Tan, and Songfang Huang. Code synonyms do matter: Multiple synonyms matching network for automatic ICD coding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 808–814, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-short.91. URL https://aclanthology.org/2022.acl-short.91.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. Opt: Open pre-trained transformer language models, 2022.