Holistic Diagnosis via Multimodal Foundation Models

Enhancing Predictive Performance of Foundation Models in Healthcare Through the Integration of Multimodal Data

Master's thesis in Engineering Mathematics

Oskar Pauli

Department of Electrical Engineering

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2024
www.chalmers.se

Master's thesis 2024

Holistic Diagnosis via Multimodal Foundation Modelst

Enhancing Predictive Performance of Foundation Models in Healthcare Through the Integration of Multimodal Data

Oskar Pauli

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Electrical Engineering
Data Science & AI
Chalmers University of Technology
Gothenburg, Sweden 2024

Holistic Diagnosis via Multimodal Foundation Models
Enhancing Predictive Performance of Foundation Models in Healthcare Through the Integration of Multimodal Data
OSKAR PAULI

Supervisor: Chiara Ceccobello, AI Sweden
Supervisor: Johan Östman, AI Sweden
Examiner: Alexandre Graell i Amat, Department of Electrical Engineering

Master's Thesis 2024
Department of Electrical Engineering
Data Science & AI
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2024

Holistic Diagnosis via Multimodal Foundation Models
Enhancing Predictive Performance of Foundation Models in Healthcare Through the Integration of Multimodal Data
OSKAR PAULI
Department of Electrical Engineering
Chalmers University of Technology

Abstract

The healthcare domain has data in many different forms, or modalities. They can be in the form of x-ray images, time-series of certain events like heart rate or blood pressure, textual data from notes etc. Medical practitioners uses many different modalities every day to make informed and sound decisions. With the recent success of small and large language models, it is natural to try and incorporate them with multimodal capabilities in the healtcare domain. This thesis seeks to investigate how well small language models can perform on predictive tasks in healthcare using multimodal data. To explore this, projectors that project data from different sources to the embedding space of a language model was developed. While the results show that a multimodal language model is better than a single-sourced version, it is still being outperformed by the XGBoost model. Even though it is being outperformed, the model proposed shows promise in regards to generalizability, potentially streamlining predictive tasks in healthcare. The thesis argue that even if improvements needs to be made and the challenges it poses can be difficult to handle, further advancements can lead to facilitating medical practitioners in a very efficient way.

Keywords: ML, language models, healthcare, mimic, multi-label classification, SHAP

Acknowledgements

Firstly, I would like to thank my project partner Liv. This project is the outcome of our collective efforts. Thank you for all the energy, hard work and dedication, even moving to Gothenburg for the sake of the project.
I would also like to extend my deepest gratitude to my supervisors Dr. Chiara Ceccobello and Dr. Johan Östman. Their guidance and continuous support during the thesis, combined with their knowledge and experience proved to be invaluable for bringing this project to its conclusion.
Furthermore, I would like to extend my thanks and appreciation to AI Sweden and the people there who helped facilitating this project with a good platform and an exciting work environment.
Lastly, a deep thank you to my family and friends who continuously show their support in everything I do.

Oskar Pauli, Gothenburg, June 2024

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

BOS	Beginning-of-sentence token
EOS	End-of-sentence token
LLM	Large Language Model
SLM	Small Language Model
MLLM	Multimodal Large Language Model
vd	Visual dense features
vmd	Visual dense features aggregated over time
ts_pe	Time-series embeddings from procedure events
ts_ce	Time-series embeddings from chart events
ts_le	Time-series embeddings from lab events
n_rad	Radiology notes embedded with BiBERT
MIMIC	Medical Information Mart for Intensive Care
BERT	Bidirectional Encoder Representations from Transformers
AUC	Area under the receiver operating characteristic curve

Contents

List of Figures

List of Tables

1

Introduction

Healthcare data exists in many different forms (hereafter referred to as modalities) such as notes, x-ray images, tabular data e.g information about age or sex and time-series data e.g the mean blood pressure during the stay. Through years of training, medical practitioners become skilled in piecing together the different sources of information to make a correct diagnose, which in this project will be different lung pathogens e.g pneumothorax, 48 hour mortality prediction and length of stay prediction. However, their expertise are often limited to a specific area, making it difficult to excel in others. The aspiration to merge the knowledge from different medical fields, where one can use all modalities and information to diagnose patients, is the ultimate goal of enhancing patient care where medicine meets AI.

Recent success in using large language models (LLMs) for different multi-modal tasks[1][2] makes it a natural next step to investigate these models' capabilities as predictors using multimodal healthcare data. Given these advancements, an intriguing question arises: How good is a small language model (SLM) at learning from data that is not language-based? SLMs are trained on vast quantities of textual data, but recent studies by Moon, Madotto, Lin, *et al.* [1] and Belyaeva, Cosentino, Hormozdiari, *et al.* [2] suggest that by projecting embeddings of other modalities to the embedding space of the LLM, it can make use of the language pretraining to reason around this new, completely different, data. This thesis aims to explore the potential of SLMs in multimodal learning, aiming to see if these smaller models can handle the different data types effectively.

1.1 Goal

This study seeks to investigate how well SLMs can perform on predictive tasks in healthcare using multimodal data. The goals of the thesis are

1. Exploring different techniques to fuse modalities, such as isolated or joint fusion

2. Try to understand the reasoning of the model and look at which modality is the most important, or if it is dependent on the task.

To achieve these, one will aim to understand the predictive performance using certain fusing technique. This will include analyzing typical machine learning metrics e.g precision, recall, F_1 score and AUC.

1.2 Approach

To achieve the goal set out for this project, an SLM such as the Gemma-2B model will be adopted and equipped with multimodal functionality by using techniques developed in recent time within the field[1][2]. The language model will remain untouched while the projectors enabling multimodal functionality are fine-tuned on the MIMIC-IV-CXR-2.2[3] dataset. To evaluate the model, it will be compared to XGBoost models with respect to relevant scoring metrics such as F_1, ROC_AUC and SHAP.

1.3 MIMIC-IV-CXR-2.2 Dataset

The dataset that will be used for this project is the MIMIC-IV-CXR-v2.2[3]. This dataset was created by the Laboratory for Computational Physiology at Massachusetts Institute of Technology and is the latest version of the MIMIC (Medical Information Mart for Intensive Care) dataset family. It contains deidentified information from 383, 220 patients admitted to the intensive care unit or the emergency department at Beth Israel Deaconess Medical Center (BIDMC). The tags "CXR" and "v2.2" specifies that this dataset contains chest x-rays of patients, as well as it being version 2.2. The specific version used has one major change compared to v1 used by e.g Soenksen, Ma, Zeng, *et al.* [4], being the removal of neonates.[3]

MIMIC-IV-CXR-2.2 is divided into 5 sections: MIMIC-core, MIMIC-ED, MIMIC-hosp, MIMIC-CXR and MIMIC-ICU. Inside these sections one can find data that corresponds to four different categories, or data modalities. There is tabular data, e.g., age or sex, visual data from chest x-rays, time-series data, e.g, heart rate monitoring, and finally textual data from radiology notes.

How the dataset was created can be described as a three-step approach with the first step being data acquisition. Data of patients admitted to any of the BIDMC intensive care units or the emergency department between 2008-2019 were extracted from the respective databases. The second step is data preparation, where the data was reorganized to simplify data analysis. Importantly, no data cleaning was made to ensure that it reflected a real world dataset. Finally, the dataset was deidentified according to the HIPAA act[5]. This included replacing person identifiers, shifting dates and times of admission etc.

1.4 Related work

In 2022, a paper was released titled "Integrated multimodal artificial intelligence framework for healthcare applications"[4]. Soenksen, Ma, Zeng, *et al.* [4] proposed a framework for extracting the multimodal data in MIMIC-IV and use that for downstream ML tasks. It was called HAIM (Holistic AI in Medicine) and to prove the benefit of using multimodality in healthcare, 14,324 models over different combinations of the data modalities were trained and evaluated, ranging from logistic regression to gradient boosted trees (XGBoost). From this thorough testing, it was shown that multimodal models could consistently outperform models using the single-source equivalent approach across 12 different tasks, consisting of 48h mortality prediction, length of stay prediction and ten pathogen classification tasks. Additionally, the paper also reported the difference in importance among the modalities for the different tasks by considering the shapley score [6] of the different modalities, for example visual information being more important for pathogen prediction than for 48 hour mortality prediction.

Another study in trying to create a model that can handle multimodal input was published in 2023 [1]. The paper presented an Any-Modality Augmented Language Model (AnyMAL) that could reason around different modalities for specific tasks. For instance, they present one application where data from motion sensors along with an image and a text prompt is inputted with the objective of generating text describing the scene. The approach taken by Moon, Madotto, Lin, *et al.* [1] was to use a pre-trained LLM, in their case the LLaMA-2 (70B)[7], and feed it with the multimodal data by converting the signals by first running it through pre-trained alignment modules and then projecting these in to the embedding space of the language model. What was especially interesting was that they only trained the adapter modules that projected the modality encoders to the embedding space of the LLM. These adepters were either linear neural networks or re-samplers. AnyMAL outperformed other state of the art (SOTA) models in both image captioning and image reasoning tasks (see tab.2 and fig.3 in Moon, Madotto, Lin, *et al.* [1]). They also noted that there was not any significant gap between AnyMAL-70B and AnyMAL-13B, indicating that one might be able to go even smaller.

Even though the AnyMAL-study proved that making a model multimodal by using projectors is effective, it is difficult to directly implement the same approach for this project and expect similar results, this is due to several reasons. First, the size of the dataset is limited; AnyMAL was trained on had 200M images [1]. The MIMIC dataset has a fraction of that, 670,343 images, and the aim and setting of our project differs significantly from AnyMAL as the setting is in healthcare and the aim is more focused towards predictive ability. Hence, one needs to look more in to studies conducted in a healthcare setting, even though general concepts in Moon, Madotto, Lin, *et al.* [1] gives inspiration.

Health Large Language Model for Multimodal Understanding (HELM) [2] is a framework that uses a similar approach to AnyMAL. The study shows that one can encode different modalities of data separately and then project these to the embedding space of an LLM. In their case, they used the Flan-PaLMChilla 62B language model[8]

and projected textual, clinical and high-dimensional lung function data from the UK Biobank dataset. Despite the relatively small size of the dataset (N=13,000), the model still managed to perform well on the tasks focused on in their study, even outperforming classical ML models such as logistic regression and XGBoost in some cases[2].

The results shown by the HeLM framework is promising and will be useful for this project for two reasons. First, it demonstrates that multimodal models are beneficial in healthcare, but also that one can achieve the results using a relatively small dataset. These showings are key, as the project will be carried out in a healthcare setting with a dataset with size in the same order of magnitude.

In the midst of all the success that has come with deep learning models, a paper by Shwartz-Ziv and Armon [9] was published, showing that tree-based methods such as XGBoost can perform on par or better than the deep learning models. A tree-based model classifies items by making asking a series of questions about its features, e.g is feature $x < 0.5$? This happens at a node, and to every node are a yes and a no children that poses another question. This pattern repeats in a tree-like manner to the point where there are no children, resulting in the item being classified as the associated class of the current node[10]. The study compared the performance of XGBoost to four models, TabNet[11], Neural Oblivious Decision Ensembles (NODE)[12], DNF-Net[13] and 1D-CNN[14], for 11 different datasets. In 8 out of the 11 cases, XGBoost outperformed the other models.

Since the data from the MIMIC dataset will be treated as tabular data (in the form of feature vectors), it is unclear if a single transformer-based model is able to outperform the results seen in the HAIM-study.

1.4.1 Contribution

The purpose of this thesis is to study multimodal small language models and their capabilities in predictive diagnosis. More specifically, the thesis will present how to enable multimodal capabilities by using projectors to project different modalities and sources to the embeddings space of a language model. The significance of training the modules jointly or in isolation will also be reported. Furthermore, results will be presented and compared to previous studies, such as HAIM[4], to evaluate the multimodal approach. Finally, we will investigate the reasoning behind the model and if the impact from the different sources and modalities changes depending on the specific task.

2

Theory

2.1 Extreme Gradient Boosting

Extreme gradient boosting (XGBoost) proposed by Chen and Guestrin [15] is based on a decision tree ensemble approach using gradient boosting with the aim of being higly scalable. They do this by introducing several improvements from the tree-boosting approach by

- Presenting an approximate algorithm for finding the best splitting points when the data becomes to large to fit in memory
- Introduce an approach for handling sparse data.

XGBoost is a popular model to use in machine learning [15] and performs well on a wide variety of tasks[4][9].

2.2 Employed Models for Pre-Processing

In this project, there is a need for extracting information from different modalities to leverage the clinical data effectively. This section presents how this was done efficiently using models pre-trained on images and notes from clinical data.

2.2.1 Densenet121-Res224-CheX

The DenseNet121-Res224-CheX model is an adaptation of the DenseNet architecture, specialized in analysis of chest X-ray images[16]. To process the images, the model uses convolutional layers and connects them with every other layer in a feed-forward manner. This allows for more connections between the layers, as opposed to previous instances of DenseNet that only had two connections, one from the previous layer and one for the subsequent layer [17].

This specific version of DenseNet has depth 121, making it possible to find complex relations in the images. The tag "Res224" refers to the resizing being made of all images. Pictures processed by the model will therefore first be resized to 224x224. Finally, the tag "CheX" signifies that the model has been trained on the CheXpert

dataset [18], a dataset containing 224,316 chest radiographs. The combination of the architecture and training made this model a great choice for processing x-ray images.[18][16]

2.2.2 Bio+Clinical BERT

Bio-Clinical BERT is a version of the BERT model that is specialized in processing natural language within the medical domain[19]. BERT, released in 2018 by Devlin, Chang, Lee, *et al.* [20], is based on the transformer architecture presented by Vaswani, Shazeer, Parmar, *et al.* [21] but the critical difference between BERT and other transformers at the time was that BERT used bidirectional self-attention. The model could therefore attend context from both sides of the input [20].

This version of BERT was trained by using BioBERT [22] as a base model and then train it on text from the MIMIC-III v1.4 database, which is an earlier version of the MIMIC-IV that is being used for this project. This fine-tuning allows BIO+ClinicalBERT to excel at understanding complex medical technical language. The combination of being able to extract relevant medical information together with a similar dataset as the model has been trained on, makes this model a valuable asset in the pre-processing of data [19].

2.3 Language Models

Language models are a fundamental part of todays' natural language processing (NLP) systems. In its earlier stages, these models were statistical models such as N-gram models which calculated the probability of a word appearing after a fixed number of words. Since then, these models have been superseded by RNN's and, more recently large language models [23].

A language model is based on the transformer model proposed by Vaswani, Shazeer, Parmar, *et al.* [21]. A transformer consist of an encoder and a decoder, but only the decoder architecture is used in the newer models[24][25]. The decoder consists of two sub-layers, an attention layer and a fully connected feed forward layer. The decoder is fed with an input sequence. The input sequence is generally a series of words and before being fed to the decoder it is first tokenized. This means that the sequence is being divided into word or sub-words, known as tokens. These tokens are represented as vectors in the so called embedding space. Before being processed, a positional encoding is added to the input to retain information about the word's original position in the sequence. The decoder embeds the input together with the positional encodings and are then being processed by the attention layer. The attention layer in the decoder allows the model to attend to, or focus on the most relevant parts of the input sequence. The output of the decoder is a probability distribution representing the likelihood of each token being the next in the sequence.

Gemma is a family of light-weight open language models that was released in February of 2024 [25] . Released in 2 sizes, 2B and 7B, the models are manageable for

smaller systems. At the time of its release, the Gemma models outperformed other models of similar sizes in several domains including math, reasoning and dialogue. The combination of size, time of release and performance made this model a great candidate for the project.

Phi-3 is another family of language models, developed by Microsoft[24]. The first model in the collection, Phi-3-mini-4k, was launched in April 2024, with a size of 3.8 billion parameters, which also showed promising results[24]. Specifically, the Phi-3-mini-4k has a hidden size of 3072, making it 50% larger than Gemma-2B. The similarity in size and time of release made this model a great alternative and could also serve as a comparison.

2.4 Multimodality

Multimodality in machine learning and artificial intelligence refers to that data can come from different sources and formats[26]. An example of this is a model that analyzes sounds and images to make predictions on e.g what bird species it could be.

In recent times, multimodal models has gained more traction, where studies show that models with multimodal capabilities outperforms single-source equivalents for task ranging from classification tasks in healthcare to image captioning and reasoning [1], [2], [4].

With the proven capabilities of LLM's, research has also been conducted on creating Multimodal Large Language Models (MLLM), popular examples of these being AnyMAL and GPT-4V[1][27]. A common approach for creating MLLM's is to use encoder modules that align the different modalities with the embedding space of the LLM, e.g Belyaeva, Cosentino, Hormozdiari, *et al.* [2]. Then, projectors are trained to project the encoded data in to a pre-trained LLM, that then reasons around the input. This has, as previously mentioned, shown great promise scoring wise but also that this scales well with respect to modalities[28].

2.5 Multi-task learning

In this thesis, we aim to predict several outputs, given one set of input features. A pragmatic approach towards this goal is to separate all tasks and perform single-task learning. Since every task is independent, one would therefore have to train a new model for every task. While doable, this approach does not scale well in the number of input modalities and downstream tasks. Moreover, for tasks that require intricate reasoning between different modalities, the single-task learning approach falls short. Therefore, our objective is to create a model that can learn from all available datamodalities and tasks simultaneously. This enforces joint reasoning, i.e., the model must attend to all of the input data simultaneously, potentially leading to better generalization performance [29][30]

2.5.1 Loss functions for binary classifications

This project will cover binary classifications in a multi-task manner, meaning that the model must be guided to learn all tasks at once. For this, we consider two loss functions for the multi-class case. Firstly, calculating the binary cross-entropy loss[31] for each class followed by taking the average over all classes. The other loss function is a modification of the focal loss called asymmetric loss, aimed towards multi-label classification[32][33].

Binary cross-entropy loss is a popular loss function to use when faced with a binary problem. Given the probability p corresponding to an observation of label $y = 1$, one can define the loss as

$$\mathcal{L} = -(y \log(p) + (1 - y) \log(1 - p)). \tag{2.1}$$

From (2.1), one can see that the more confident the model is in its prediction, the smaller the loss becomes. However, this loss assumes that the two classes are balanced; In our case, many of the tasks that are being investigated in this project are not balanced (see tab. 3.1), which leads to the model predicting the majority class. To mitigate this issue, a weighting factor could be added to the terms in (2.1), forcing a larger penalty when the model makes an error on the minority class. This results in the weighted binary cross-entropy loss function

$$\mathcal{L} = -(w_{\text{pos}}(y \log(p)) + w_{\text{neg}}((1 - y) \log(1 - p))) \tag{2.2}$$

where w_{pos} and w_{neg} are the weights. The weights w_{pos} and w_{neg} are calculated using the inverse class frequency method is used [34], where

$$w_{\text{pos}} = \frac{\text{total\_samples}}{2 \cdot \text{pos\_samples}} \tag{2.3}$$

$$w_{\text{neg}} = \frac{\text{total\_samples}}{2 \cdot \text{neg\_samples}}. \tag{2.4}$$

A pragmatic way of using the cross-entropy loss for multi-class classification is to average the cross-entropy loss from all tasks. This means that all tasks are equally weighted in to the final loss. However, this is not the only approach.

Another approach is to use an asymmetric loss, proposed by Ben-Baruch, Ridnik, Zamir, *et al.* [33], which is a modification of the focal loss[32]. In some cases, where the imbalance is extreme, it is not enough to weigh the classes using a constant factor as the model would still not learn the appropriate distribution. Under those circumstances, the focal loss function could be more suitable as it focuses on the samples that are hard to predict[32]. If one defines cross-entropy loss as

$$CE(p, y) = \begin{cases} -\log(p) & \textit{if} \quad y = 1, \\ -\log(1 - p) & \textit{otherwise} \end{cases} \tag{2.5}$$

and probability p_t as

$$p_t = \begin{cases} p & \textit{if} \quad y = 1, \\ 1 - p & \textit{otherwise} \end{cases} \tag{2.6}$$

one can rewrite $CE(p, y) = CE(p_t) = -\log(p_t)$. What focal-loss does is adding a modulating factor $(1 - p_t)^\gamma$ to the standard cross-entropy loss. This generates the focal-loss function

$$FL(p_t) = -(1 - p_t)^\gamma \log(p_t). \tag{2.7}$$

This modulating factor down-weights items with confident predictions to classify and focuses more on the harder examples. This can be seen by observing that a p_t close to zero would leave the loss term nearly unaffected, while a larger p_t would down-weigh the loss-term[32].

With focal loss defined, one can introduce the asymmetric loss. The authors claim that there is a trade-off with focal loss, where setting a high γ leads to down-weighing of easy negative samples, it might also lead to ignoring rare positve samples. Ben-Baruch, Ridnik, Zamir, *et al.* [33] proposes to decouple γ into γ^+ and γ^-, resulting in a reformulation of the loss as

$$\begin{cases} L_+ = (1 - p)^{\gamma+} log(p) \\ L_- = (p)^{\gamma-} log(1 - p), \end{cases} \tag{2.8}$$

called asymmetric focusing. The other mechanism they introduce is called asymmetric probability shifting

$$p_m = max(p - m, 0), \tag{2.9}$$

where m is a tuneable probability margin. This mechanism effectively discards negative samples when they are very likely to be incorrect. Adding these two mechanisms together you get the final asymmetric loss

$$\text{ASL} = \begin{cases} L_+ = (1 - p)^{\gamma+} log(p) \\ L_- = (p_m)^{\gamma-} log(1 - p). \end{cases} \tag{2.10}$$

In addition, Ben-Baruch, Ridnik, Zamir, *et al.* [33] proposes a way of dynamically adjusting the γ-parameters (See [33] Sec 2.7) to keep a good balance on which class to focus on.

2.6 Evaluation metrics and concepts

This section presents the different metrics that will be used to evaluate the models and why they are relevant in this project to quantify the performance.

2.6.1 $F1$ score

In this project, we perform classification tasks where the dataset has a strong imbalance between the positive and negative class. Having accuracy as a metric to evaluate the performance is therefore not ideal, since it can give a skewed picture of how well the model is performing. More specifically, if the ratio between the negative and positive class is 99:1, a model that classifies everything as a negative

instance will get an accuracy of 99% which could be perceived as excellent, but in reality one failed to identify a whole class.

To combat this issue, one can use other metrics such as precision and recall. In a binary task, precision can be described as how many of the retrieved positive items actually are positive. This can be written as

$$Precision = \frac{TP}{TP + FP} \tag{2.11}$$

where TP denotes the number of true positives and FP denotes the amount of false positives. Similarly, one defines recall as how many of all positive items were retrieved, written as

$$Recall = \frac{TP}{TP + FN} \tag{2.12}$$

where FN denotes the false negatives. These two metrics can paint a more accurate picture of how good the model is at finding the positive class. To assess the precision and recall in union, one can use the F_1 score.

The F_1 score is defined as the harmonic mean between precision and recall, written as

$$F_1 = \frac{2\,P\,R}{P + R}. \tag{2.13}$$

This metric is a valuable tool for analysis since it, unlike the arithmetic mean, does not treat the precision and recall equally. The harmonic mean is more sensitive to low values, which means that a good F_1 score can only be achieved if both the precision and recall is high[35]. Furthermore, if both the negative and positive class is of importance, one can then use macro-averaged F_1. This is calculated by taking the per-class F_1 scores and then arithmetically average them [36].

2.6.2 AUC

Another metric that is popular to use when assessing models for binary classification tasks is the area under the receiver operating characteristic curve (AUC). This is a metric showing the discriminatory power of the model by computing the true positive rate (TPR) and false positive rate (FPR) for different prediction thresholds. Normally, one uses a threshold of 0.5 when deciding on what class the item belongs to, which means that a model could be very uncertain in its prediction (predicting 0.51 for the correct class) while still being correct. The AUC score rewards models with more certain predictions and penalizes uncertainty, with a score ranging from 0 to 1. A score of 0.5 indicates random guessing and a score of one is a model that distinguishes the classes perfectly.

2.6.3 Shapley score

The Shapley score was a metric proposed by Lloyd Shapley, originally designed for game theory to evaluate the contributions of different players with respect to a certain game[6]. This is applicable in machine learning, when one wants to understand

how the different features contribute to the performance of the model. Individual input features are viewed as players and the set of all chosen features as the team [37]. The framework for using Shapley-scores in machine learning for complex models and many features is called Shapley Additive exPlanations, abbreviated SHAP [38].

For this project, our focus is not mainly only on feature level explanation, but instead on explainability on a modality level. One can therefore calculate the contribution from modality m as

$$\phi_m = \sum_{A \subseteq S \setminus \{t\}} \frac{|A|! \cdot (N - |A| - 1)!}{N!} \cdot (v(A \cup \{m\}) - v(A)). \tag{2.14}$$

In (2.14), S is the set containing all modalities, N is the total number of available modalities, A is a subset not containing modality m, $v(A \cup \{m\})$ is the AUC score for the model including all modalities and $v(A)$ is the AUC score of the model using only the modalities in subset A. The formula sums over all possible subsets A, generating a weighted mean [37].

In reality, when the set of players are very large, computing the exact shapley values become computationally very expensive and one has to resort to subsampling methods[37]. However, since this project only has three "players" or modalities, one can actually compute these exactly.

3

Methodology

In this chapter, the methodology for the different steps of the project will be presented. The project can be divided into three sections, each building upon the previous with the goal of creating a multimodal model in healthcare. Relevant theory and aspects of creating such a model will be brought up to give the reader a clear understanding on the steps taken.

The first step is a pre-processing step, aimed to create a dataset from the MIMIC-IV database, where one extracts and organizes data into different modalities to facilitate an easier workflow. Since the dataset is large and complex, it is crucial that the dataset can be reused for multiple tasks to save time (Sec.3.1).

Once the dataset is created, a benchmark based on previous studies is obtained. This is done to enable a direct comparison on the same dataset between a proven model and the one that will be constructed during this project (Sec. 3.2).

The third section will revolve around the construction and training of the multimodal model that has been created. It will go into detail about key steps such as how the model processes input, overall architecture, how it was trained and also how it was evaluated (Sec. 3.3).

3.1 Pre-processing of data

As mentioned above, the first step of this project is to create a dataset to facilitate training, testing and benchmarking of models. To create this we use the MIMIC-IV-CXR-2.2 dataset, introduced in Sec.1.3 The procedure closely followed the HAIM-study [4]. The main differences between the dataset used in [4] and the dataset used in this project are: i) the removal of neonates and ii) the removal of all notes except the ones coming from radiology.

The first step in creating a version of the dataset suitable for machine learning and generative AI was to find all patients that had all modalities that was requested, in our case being textual, time series and visual data. After this step, approximately 28000 patients were left.

For the remaining patients, embeddings from their healthcare data was generated and stored in a dataframe. How they were generated will be revelaed in subsequent sections. The sources that we decided to extract data from are visual features, visual

features aggregated over time in the case where a patient had several images taken, radiology notes and time series data from procedure-, lab- and chart events.

Data which originated from images were processed using the pretrained Densenet121-res224-chex [16], specialised in extracting relevant features from chest x-rays. The output from the model is a 1024 vector representing the image. Similarly, for extracting the image vector aggregated over time, one extracted the features for every image during the patient's stay and weighted them differently based on how recent the image was taken. The later the image was taken, the more it should contribute to the embedded vector. Pseudo code for the procedure is listed in Algorithm 1.

Algorithm 1 Temporal Weighted Feature Aggregation

```
orig_imgs_weights ← times of images taken
adj_imgs_weights ← orig_imgs_weights − min(orig_imgs_weights)
imgs_weights ← adj_imgs_weights/ max(adj_imgs_weights)
                ▷ Weighted average of embedding vectors across temporal dimension
aggregated_embeddings ← mean(feature_emb, weights = img_weights)
```

The radiology notes were processed using BioBERT[19], a pre-trained language model specialized on medical text. The notes were checked if they were larger than 512 tokens and in that case needed to be split into smaller chunks to allow for BioBERT to process them. Once all natural text for a patient had been embedded, a similar temporal weighting process was conducted to end up with a 768 long vector, representing all text for a patient's stay.

Finally, the embeddings for the time series data for procedure-, lab- and chart events were generated. They have a similar structure, meaning that events for specific instruments have been tracked during a patient's stay. For each event, we extract metrics such as mean, variance, minimum value, maximum value, and 7 more values. This is done for each event and for each for the three different categories. The full list of values extracted and the different events can be seen in A.1 and A.2.

Once all embeddings are generated, it is then stored in a data frame where every row is a patient stay and a total of $3,267$ features. The features are split up into so-called sources depending on where it originated from. There were six sources, visual dense features (vd), visual dense features aggregated over time (vmd), time-series feature for chart-, procedure- and lab events (ts_ce, ts_pe, ts_le) and textual features from radiology notes (n_rad). This process has allowed for a smoother transition from examining the data to training models, the first one being the reproduction of the HAIM-study.

3.2 Reproduce HAIM-study

In order to compare our work with the original HAIM study, we selected the data types/sources to reproduce the relevant part of the HAIM study [4] and use it as a benchmark for our analysis.

Through rigorous testing, [4] found that the best perofrmance for the type of classification task they had were acheived with XGBoost and, based on their report, this is the only model that will be used in this benchmarking stage.

The pipeline for training our XGBoost model on the different tasks? is a straightforward process. The dataset is being split up into a training set and a test set with a 75/25 ratio. The training set along with its labels as well as the test data is then being fed to the XGBoost algorithm that trains a classifier by optimizing the hyperparameters through a grid search based on AUC score. When the classifier is trained, it evaluates the performance on the validation set, returning the predicted label and also the prediction probabilities which will be used to check the quality of the model. A pseudo-implementation of the pipeline can be seen in Algorithm 2. Since we do not only look for how well the model performs, but also in being able to explain what modality is most important for the different tasks, this process iterates over all different combinations of sources and modalities. Furthermore, this model is deployed over all 12 tasks, to get a wide variety of tasks with different class imbalances, sizes and other different settings. In the end, 63 different models per task was trained, resulting in a total of $63 * 12 = 756$ models.

Algorithm 2 Train and Evaluate XGBoost Classifier

Input: Dataset
Output: Predicted labels, prediction probabilities
$dataset \leftarrow$ Load dataset
$X, y \leftarrow$ Features and labels from $dataset$
$X_{train}, X_{val}, y_{train}, y_{val} \leftarrow$ Split(X, y, ratio=0.75)
$param\_grid \leftarrow$ Define({'max_depth': [3, 5, 7], 'n_estimators': [100, 200], 'learning_rate': [0.1, 0.01]})
$xgb\_model \leftarrow$ InitializeXGBClassifier()
$grid\_search \leftarrow$ GridSearchCV(xgb_model, $param\_grid$, scoring='AUC')
$best\_model \leftarrow$ grid_search.Fit(X_{train}, y_{train})
$predictions \leftarrow$ best_model.predictions
$probabilities \leftarrow$ best_model.probabilities
return $predictions, probabilities$

From all the tests, the F_1 score and AUC score are collected. The AUC score will be used for generating Shapley-values, necessary for understanding what modality impacts decisions most, and the F_1 score will be used as the main metric for assessing performance and comparing with other models.

3.3 Multimodal foundation model

The aim of this thesis is to create a model that can receive and reason around multiple modalities and then make multiple predictions based on the input. The tasks focused on in this thesis, together with the sizes of data is documented in Table 3.1. In the table, one can see that many tasks have large imbalances, which

in turn means that one has to create a model suitable for handling these scenarios. There are many possible ways to make a multimodal language model to excel in this aspect. Hence, it is cruical to design a procedure that enables us to establish which among the different model versions is the best performing one. The framework for training the model can be described in the following steps:

1. Produce an architecture for the projector modules, see Figure 3.3.
2. Train the projectors for a multi-label classification use-case, see Figure 3.1.
3. Extract scoring metrics and compare with benchmark.

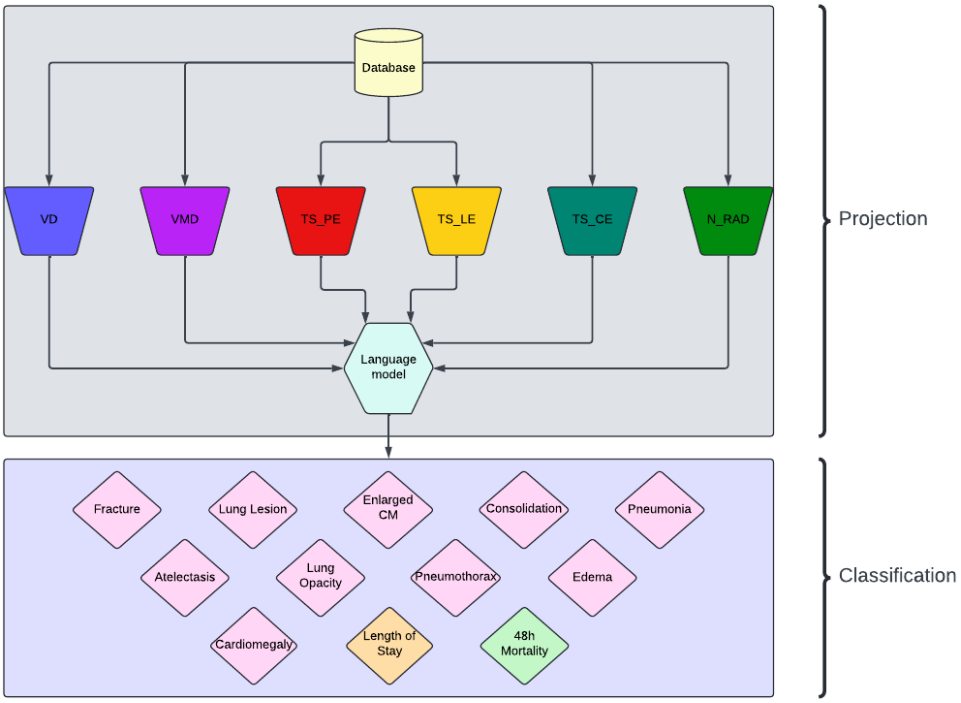


Figure 3.1: The training process for creating a multimodal model for multilabel classification. Data from the database is being split up into the original sources. They are then projected individually in to vectors that are consistent with the size of the embedding space of the language model. The projected vectors are being processed and results from the next token prediction are extracted and used for the classification tasks.

Since the variables and combinations of different architectures are endless, one has to limit what parts of the framework to tune or change. With support from the theory presented regarding multi-label classification, the two loss functions that will be tested are the average loss function using binary cross-entropy loss (Sec.2.5.1), and the asymmetric loss function that uses focal-loss (Sec.2.5.1). Furthermore, the testing will also include training the projectors jointly and in isolation. This will

result in four different scenarios. After obtaining the four different models, one will compare them with respect to the different scoring metrics presented in the theory, ultimately determining the best overall model. That model will then be further processed to allow for analysis regarding contributions of the modalities to different tasks. One will also try to train the same projector framework but with a different language model to see what impact that could have.

	# samples	**neg**	**pos**		**# samples**	**neg**	**pos**
Fracture	1612	85	1527	Lung Opacity	29540	1107	28433
Lung Lesion	1611	100	1511	Pneumothorax	34171	27806	6365
Enlarged CM	6614	1831	4783	Edema	30713	11496	19217
Consolidation	9747	1701	8046	Cardiomegaly	34832	7072	27760
Pneumonia	14684	6539	8145	Length of stay	90811	82323	8488
Atelectasis	30274	808	29466	48h Mortality	90811	88581	2230
Total dataset size:		90811					

Table 3.1: The table presents the different labels and number of samples per label used in this thesis.

3.3.1 Architecture

Inspired by [1] and [2], the idea to create projector modules for the different modalities was clear. The idea is visualized in Figure 3.2. In this case, the projections necessary will be from the size of the embeddings (size varies depending on source) in the dataset, to the size of a token vector for Gemma-2B (2048). There are different approaches one could take to achieve the projection, with the main objective being to retain as much information from the original sources as possible. One mainly considered regular feed forward neural networks of varying depths and shapes. As the embeddings from Sec 3.1 are of different sizes, all below the token size of Gemma-2B, the projections must increase the dimension o the embedding vectors. This risks diluting the signal.

To combat this issue, the architecture used was comprised of two sections, visualized in Figure 3.3. The first part, called the encoder, encodes the information from the input vector in to a vector of higher dimension, specifically to the size that is required from the language model. The second step is a decoder step, where the goal is to reconstruct the input vector. For this project, both the encoder and decoder was represented as one fully connected hidden layer, connecting the input size to the expected output size. This design generates two losses, $\mathcal{L}_1$ coming from how well the language model performs in the classification tasks given the encoded vector and $\mathcal{L}_2$ which is the reconstruction error when comparing the decoded vector with the input vector. This approach is the result of extensive testing early on in the project and was the design of choice for the project.Moreover, this approach is completely agnostic to the size of input or requested output, making it easy to adapt to models and sources of varying dimensions.

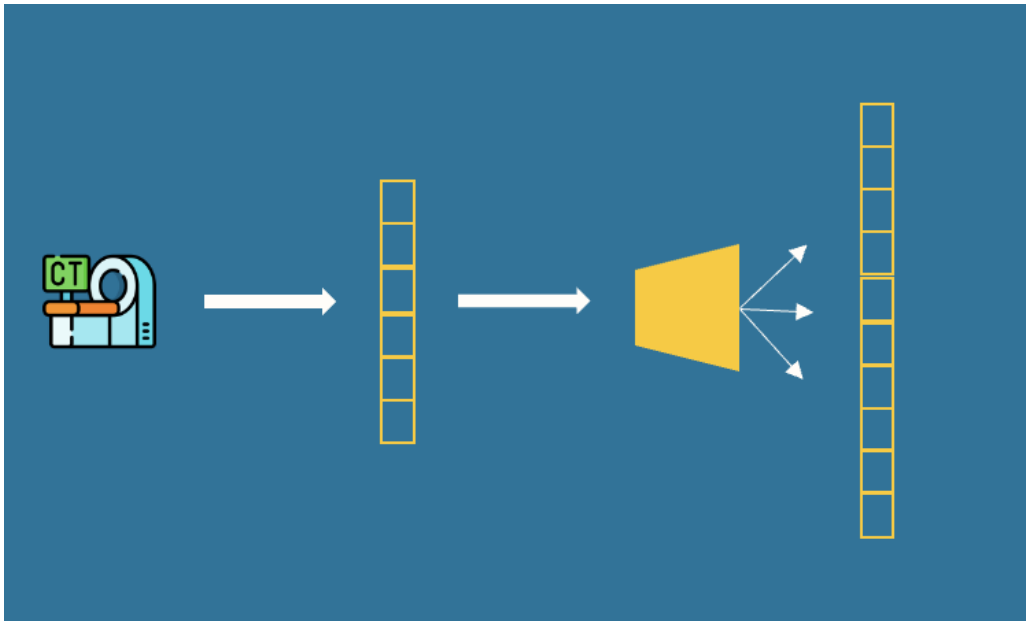


Figure 3.2: Illustration of the purpose of the projector. In this case, an x-ray image is first being converted to a feature vector as described in Sec.3.1. This vector is then projected to the embedding space of the language model.

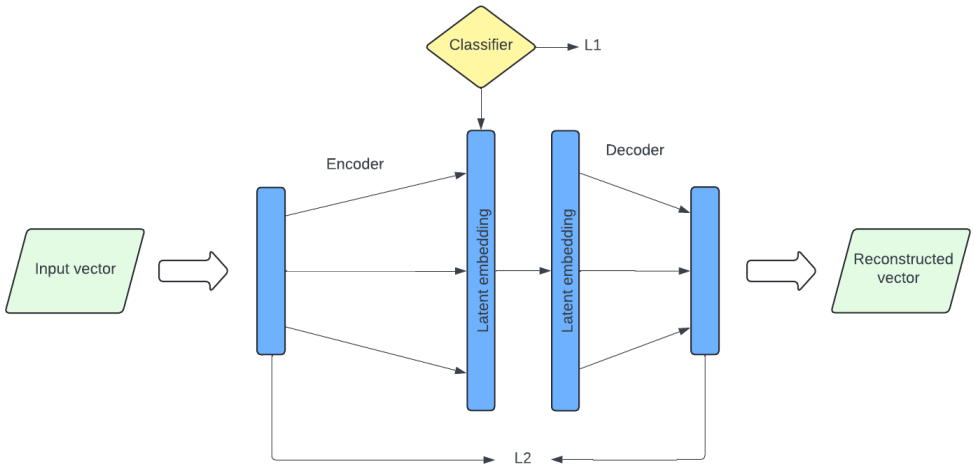


Figure 3.3: Illustration of the architecture used for the projection modules. The first part involves "encoding" information in to a higher dimensional vector, which will be used by the language model. The second part "decodes" the encoded vector back to the original size. This projector generates two ways of training and evaluating performance, as one can look at the loss generated by the classification tasks ($\mathcal{L}_1$) and also by looking at the reconstruction error ($\mathcal{L}_2$).

3.3.1.1 Setup and training

The training process can be described as follows. Utilizing the dataset, the six different sources of information mentioned in Sec. 3.1 are being extracted. These sources were then fed through their respective projector modules, generating six vectors corresponding to six tokens. These tokens are then fed to the Gemma-2B model, which performs next token prediction on the input. The logits are averaged across the predicted tokens, and the predictions for the different labels were associated with twelve different tokens were extracted. These twelve tokens were chosen arbitrarily before the training, however special tokens or frequently occurring tokens such as beginning-of-sequence (BOS) and end-of-sequence (EOS) were avoided. An illustration of the training can be seen in Figure 3.1.

Once the predictions were extracted, one could carry on the next step in classifying the binary labels. For every data point, up to twelve distinct classifications were possible. Since not every data point had the same number of labels to predict, a masking step was made to ignore the tasks which was not present for that specific patient stay. For the remaining tasks, a classification loss $\mathcal{L}_{\text{classification}}$ was calculated. $\mathcal{L}_{\text{classification}}$ was weighted together with a scaling parameter β and the reconstruction error for projector $\mathcal{L}^{(i)}_{\text{reconstruction}}$ where $i \in \{\text{vd}, \text{vmd}, \text{ts\_pe}, \text{ts\_ce}, \text{ts\_le}, \text{n\_rad}\}$, giving the final loss

$$\mathcal{L}^{(\text{i})} = \mathcal{L}_{\text{classification}} + \beta\mathcal{L}^{(i)}_{\text{reconstruction}},\ \text{i} \in \{\text{vd}, \text{vmd}, \text{ts\_pe}, \text{ts\_ce}, \text{ts\_le}, \text{n\_rad}\}. \quad (3.1)$$

The different losses was then used to update the weights of the different projectors respectively, while keeping the parameters of the language model frozen.

$\mathcal{L}_{\text{classification}}$ could be calculated either by taking the average binary cross-entropy loss from all available task, or by using asymmetric loss. Both were tested and evaluated. Furthermore, the process of extracting logits described above is applicable for the case where one wants to train the projectors jointly. For the case where the projectors are trained in isolation, only one of the projectors are used at a time, resulting in only one token being fed to the language model which makes the averaging step redundant. Both methods were tested. The following general training choices regarding hyperparameters and optimizers, one refers to Table 3.2.

Deep learning library	Pytorch
Optimizer	Adam
Batch size	32
Learning rate for projectors	5e-4
β	0.1
Epochs	100

Table 3.2: Table of hyperparameters and libraries used during the training of projectors.

In addition to training the projectors differently, we also tried Gemma-2B and Phi-3-small as the engine for reasoning and classifying. The results of training, F_1 score, AUC-score, precision and recall were saved for the different approaches, allowing for comparison between them and the benchmark.

3.3.2 Evaluation

With all training done, one wants to evaluate the models. This is done by comparing the scoring metrics extracted from the training for the different models, culminating in a decision regarding which model performed the best. Once that is decided, the best model is processed further.

The processing that the final model will undergo is to extract the contributions the modalities have on different tasks. This is done by retrieving Shapley-values.

4

Results

The experimentation and training of our multimodal language models culminate in a series of results, containing scoring metrics and plots of losses and contribution scores. In this section, one aims to get an understanding on how the different model performs, ultimately deciding on the best one. It should be noted that no tests were performed for a model with the projectors trained in isolation with averaging loss. This decision was made based on the overall poor performance shown from the model with projectors trained individually with the asymmetric loss.

4.1 Observations from loss curves

The first observation can be seen from examining Figure 4.1 and Figure 4.3, where both models have similarities regarding the attributes of the training curve. Figure 4.1 shows a noisy, although improving, training process which is consistent with Figure 4.3. What differs between them are the scale of the loss, as can be seen by looking at the y-axis of the plots.

Secondly, we compare Figure 4.2 with Figure 4.3. Once again, the plots looks similar with both models having a significantly lower training loss compared to the validation loss, which could indicate possible overfitting. What also is worth noting is the scale of the loss, with Figure 4.2 having a loss 8 times as large. Since these models uses the same loss function, this discrepancy indicates that the model in Figure 4.3 performs better. Furthermore, Figure 4.3 also shows a significant drop in training loss after epoch 70 but without any notable impact on the validation loss.

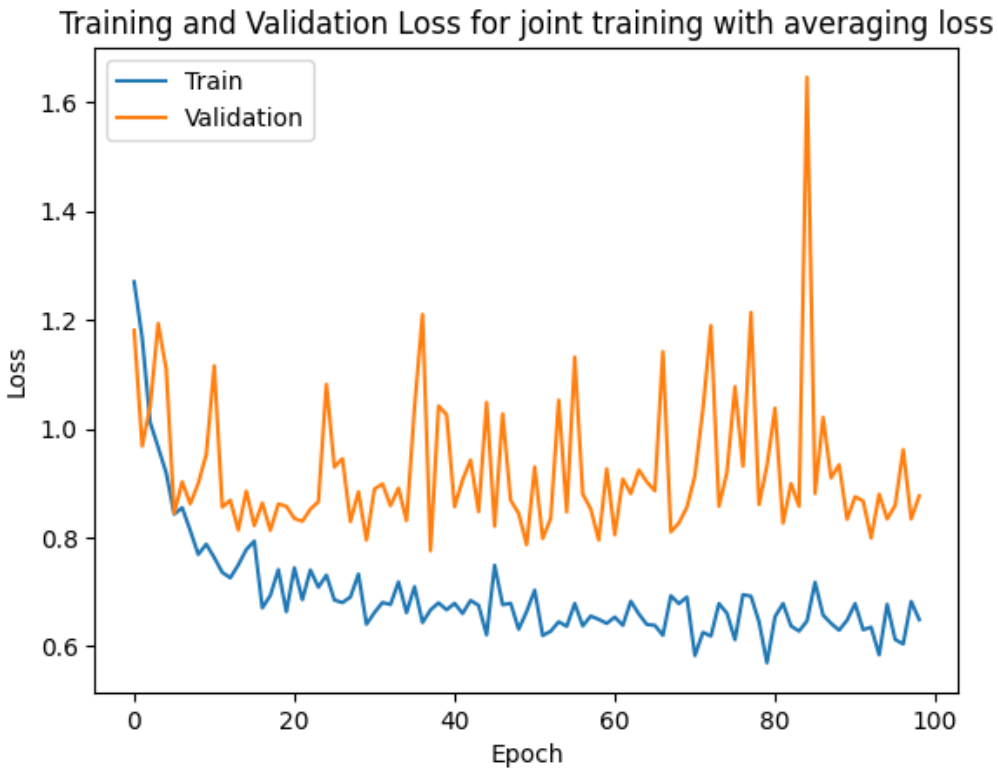


Figure 4.1: Training loss plotted against epochs for model with jointly trained projectors. The loss reported was calculated as an average over the binary cross-entropy losses for each label.

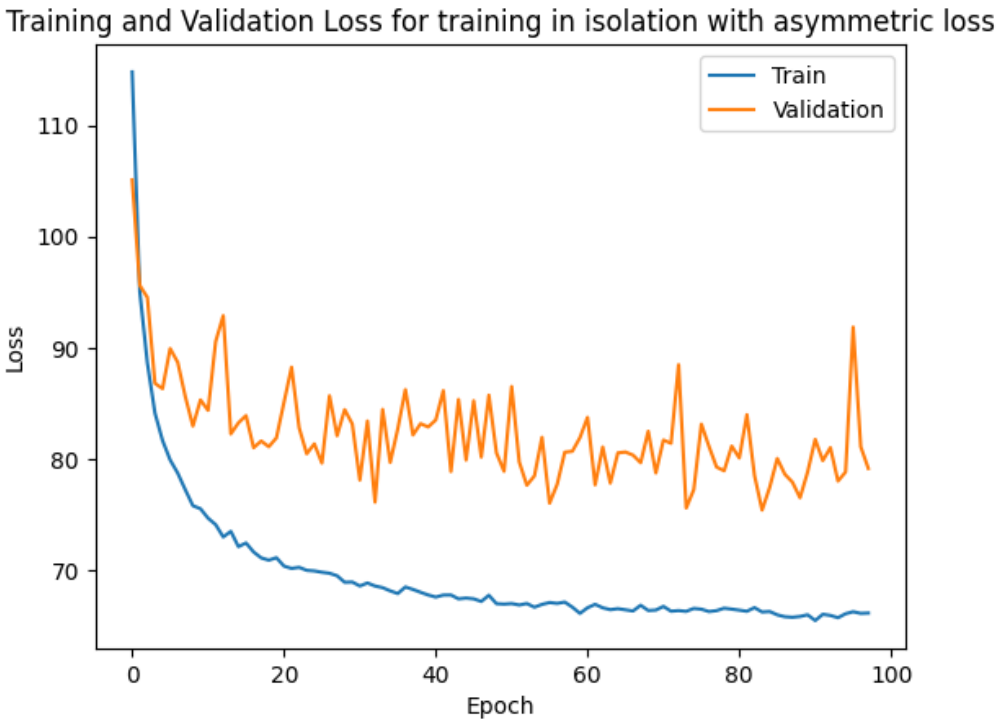


Figure 4.2: Training loss plotted against epochs for model with projectors trained in isolation. The loss reported was calculated using the asymmetric loss function.

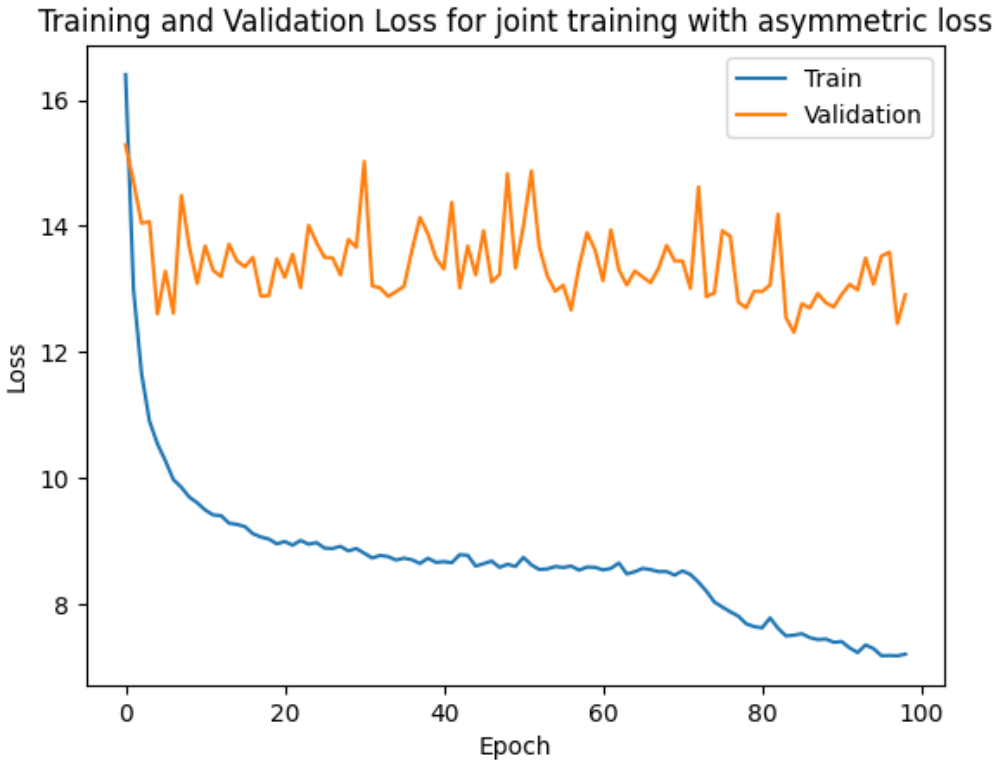


Figure 4.3: Training loss plotted against epochs for model with jointly trained projectors. The loss reported was calculated using the asymmetric loss function.

4.2 Analysis of scoring metrics

The AUC and the macro-averaged F_1 score is displayed for the different tasks and models in Table 4.1 and Table 4.2. By looking at the AUC and F_1 across all tasks, one can see that no proposed model is beating the benchmark obtained with XGBoost following the HAIM study[4]. The jointly trained projectors using asymmetric loss is getting quite close with respect to the F_1 score on the length-of-stay and 48-hour mortality tasks. In general, the largest difference in performance can be seen on the lung pathogens with smaller datasets like fracture and lung lesion, and performance increases for larger datasets like edema and cardiomegaly.

Furthermore, by observing Table 4.1 and Table 4.2 it is possible to assess differences in performance between the different models. The model with projectors trained in isolation performs significantly worse than any other model across all tasks, ruling it out as a contender for the best model. By comparing the last four models in the tables, i.e. excluding the Benchmark and Isolated-Asym-G2B, we decide to exclude the best single sourced model because it scores worse than the other models in nearly all tests. More specifically, it performs worse on 10 out of the 12 tasks in regards to the F_1 score and 11 out of 12 tasks for the AUC score, compared to Joint-Asym-G2B.

We then compared the three models with jointly trained projectors further. The Phi-3 model performs the worst on 8 out of 12 tasks with respect to the Gemma-2B models and is never the best performing one on the remaining four tasks. However, it outperforms the Gemma-2B models in the AUC-score, beating them in 8 out of 12 tasks. This indicates that even if the Phi3 model does not have the best predictive performance, it is certain in its predictions.

In addition, a comparison can be made between Joint-Avg-G2B and Joint-Asym-G2B. They appear to be quite similar in performance with the model using asymmetric loss outperforming the average loss 7 to 5 concerning the F_1 and vice versa with respect to the AUC score.

	Length of Stay		48h Mortality	
	AUC	F_1	AUC	F_1
Benchmark (HAIM)	0.904	0.684	0.843	0.649
Isolated-Asym-G2B	0.567	0.349	0.547	0.344
Joint-Avg-G2B	0.786	0.578	0.85	0.518
Joint-Asym-G2B	0.779	0.623	0.793	0.598
Joint-Asym-Phi3	0.771	0.573	0.782	0.583
Best-single-source	0.711	0.553	0.696	0.590

Table 4.1: AUC and F_1 for the LOS and 48-h mortality prediction tasks using our different experimental setups as well as the benchmark from HAIM.

	Fracture		Lung Lesion		Enlarged CM		Consolidation		Pneumonia	
	AUC	F_1	AUC	F_1	AUC	F_1	AUC	F_1	AUC	F_1
Benchmark (HAIM)	0.998	0.989	0.998	0.986	0.937	0.942	0.973	0.963	0.864	0.896
Isolated-Asym-G2B	0.519	0.5	0.618	0.472	0.61	0.512	0.649	0.566	0.639	0.488
Joint-Avg-G2B	0.685	0.567	0.52	0.697	0.75	0.629	0.847	0.692	0.797	0.724
Joint-Asym-G2B	0.576	0.516	0.645	0.502	0.71	0.596	0.83	0.717	0.764	0.624
Joint-Asym-Phi3	0.544	0.481	0.673	0.484	0.777	0.468	0.835	0.617	0.812	0.54
Best-single-source	0.618	0.534	0.572	0.523	0.573	0.546	0.766	0.607	0.695	0.511

	Atelectasis		Lung Opacity		Pneumothorax		Edema		Cardiomegaly	
	AUC	F_1	AUC	F_1	AUC	F_1	AUC	F_1	AUC	F_1
Benchmark (HAIM)	0.972	0.965	0.966	0.956	0.873	0.769	0.919	0.87	0.914	0.818
Isolated-Asym-G2B	0.613	0.502	0.605	0.507	0.606	0.421	0.622	0.52	0.625	0.507
Joint-Avg-G2B	0.687	0.494	0.694	0.494	0.727	0.596	0.84	0.743	0.815	0.618
Joint-Asym-G2B	0.728	0.53	0.707	0.532	0.748	0.644	0.837	0.722	0.832	0.694
Joint-Asym-Phi3	0.759	0.51	0.754	0.509	0.765	0.586	0.857	0.624	0.85	0.62
Best-single-source	0.661	0.504	0.605	0.516	0.74	0.602	0.749	0.608	0.696	0.59

Table 4.2: AUC and F_1 for the chest pathology diagnosis prediction tasks using our different experimental setups as well as the benchmark from HAIM. We also include scores for the best performing single modality evaluation per task (taken from the model where projectors were trained in isolation).

Since it also is of interest to see how good the remaining models are at identifying the positive class of the tasks, Table 4.3 and Table 4.4 lists the precision and recall for the different models and labels. When examining these, the models seem to be competitive with the benchmark. The differences in recall between the three models and the benchmark are very small, except for length-of-stay, 48h mortality prediction and pneumothorax. For the precision, they are quite similar once again with Joint-Asym-G2B being slightly better. However, the benchmark scores better on average with an average precision of 0.81 while Joint-Asym-G2B had an average of 0.72. This indicates that the proposed models are on par with the benchmark in regards to how well it can retrieve items from the positive class while it tend to also catch too many non-relevant items.

Given the results discussed above, we weighted the different metrics against each other and concluded that using Gemma-2B as the language model with jointly trained projectors and an asymmetric loss resulted in the best model. A more detailed analysis to why this was chosen will be presented in the discussion. What then is left is to present the contribution scores for this model.

	Length of stay		48h Mortality	
	Prec	Rec	Prec	Rec
Benchmark (Haim)	0.314	0.742	0.248	0.481
Joint-Avg-G2B	0.23	0.686	0.077	0.719
Joint-Asym-G2B	0.265	0.497	0.174	0.298
Joint-Asym-Phi3	0.203	0.638	0.13	0.418

Table 4.3: Precision and recall for the LOS and 48-h mortality prediction tasks using a subset of our different experimental setups as well as the benchmark from HAIM.

	Fracture		Lung lesion		Enlarged CM		Consolidation		Pneumonia	
	Prec	Rec	Prec	Rec	Prec	Rec	Prec	Rec	Prec	Rec
Benchmark (Haim)	0.958	0,999	0.949	0.998	0.866	0.94	0.915	0.976	0.831	0.852
Joint-Avg-G2B	0.946	0.969	0.94	0.995	0.789	0.94	0.88	0.982	0.739	0.784
Joint-Asym-G2B	0.94	0.938	0.939	0.965	0.776	0.959	0.896	0.945	0.637	0.907
Joint-Asym-Phi3	0.938	0.988	0.939	1.0	0.745	0.996	0.862	0.995	0.602	0.984
	Atelectasis		Lung opacity		Pneumothorax		Edema		Cardiomegaly	
	Prec	Rec	Prec	Rec	Prec	Rec	Prec	Rec	Prec	Rec
Benchmark (HAIM)	0.982	0.999	0.973	0.999	0.879	0.461	0.876	0.905	0.906	0.96
Joint-Avg-G2B	0.976	0.999	0.964	0.999	0.307	0.654	0.777	0.904	0.842	0.976
Joint-Asym-G2B	0.977	0.997	0.966	0.992	0.366	0.612	0.759	0.917	0.868	0.953
Joint-Asym-Phi3	0.976	0.999	0.965	0.999	0.3	0.746	0.699	0.975	0.842	0.986

Table 4.4: Precision and recall of the chest pathology diagnosis prediction tasks using a subset of our experimental setups as well as the benchmark from HAIM.

4.2.1 Contribution scores

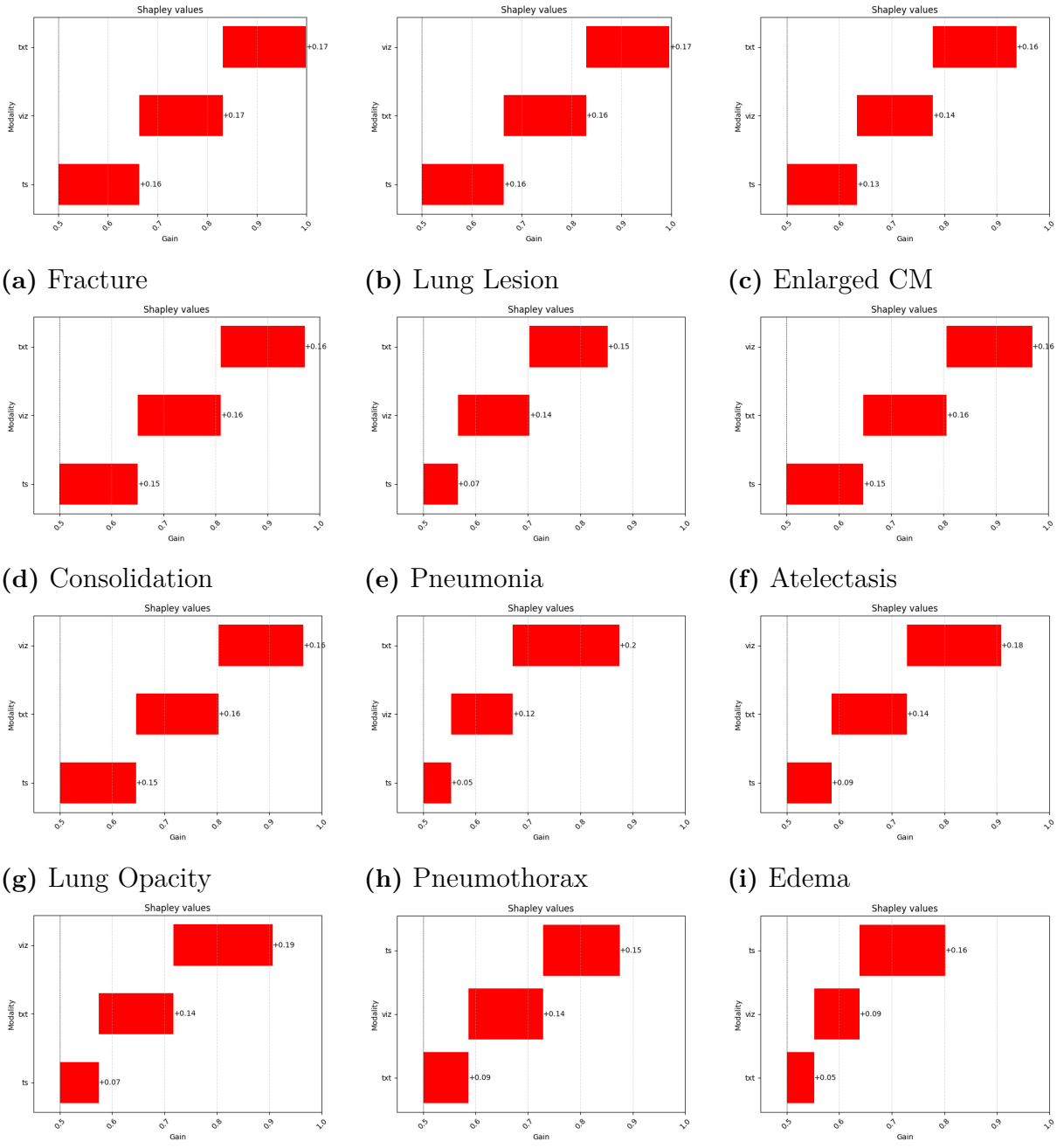
With the Gemma-2B language model combined with jointly trained projectors and an asymmetric loss crowned as the best performing model, Shapley-values were extracted for the different modalities and tasks and presented in Figure 4.5. The Shapley-values for the XGBoost benchmark are also visualized in Figure 4.4.

By observing the plots, one can see that the visual and textual modalities are the main contributors for performance in pathogen classifications for both the benchmark and the multimodal language model developed. What is interesting is that in Figure 4.5a, 4.5b, 4.5c and 4.5g the time-series modality seems to negatively impact the models performance. This behaviour cannot be seen in Figure 4.4, where all modalities are contributing in positively.

Furthermore, in Figure 4.5k and 4.5l one observes that the main contributor is the time-series. The textual and visual data is still contributing but does not carry the same importance as for the pathogen classification. This behaviour is also consistent with the benchmark in Figure 4.4k and 4.4l.

To summarize, the multimodal language model has similar behaviour as the benchmark, with two main differences. The first one being that in certain scenarios, the

time-series modality can have a negative impact on the AUC-score, and the second one being that the overall total scores are better for the XGBoost model.



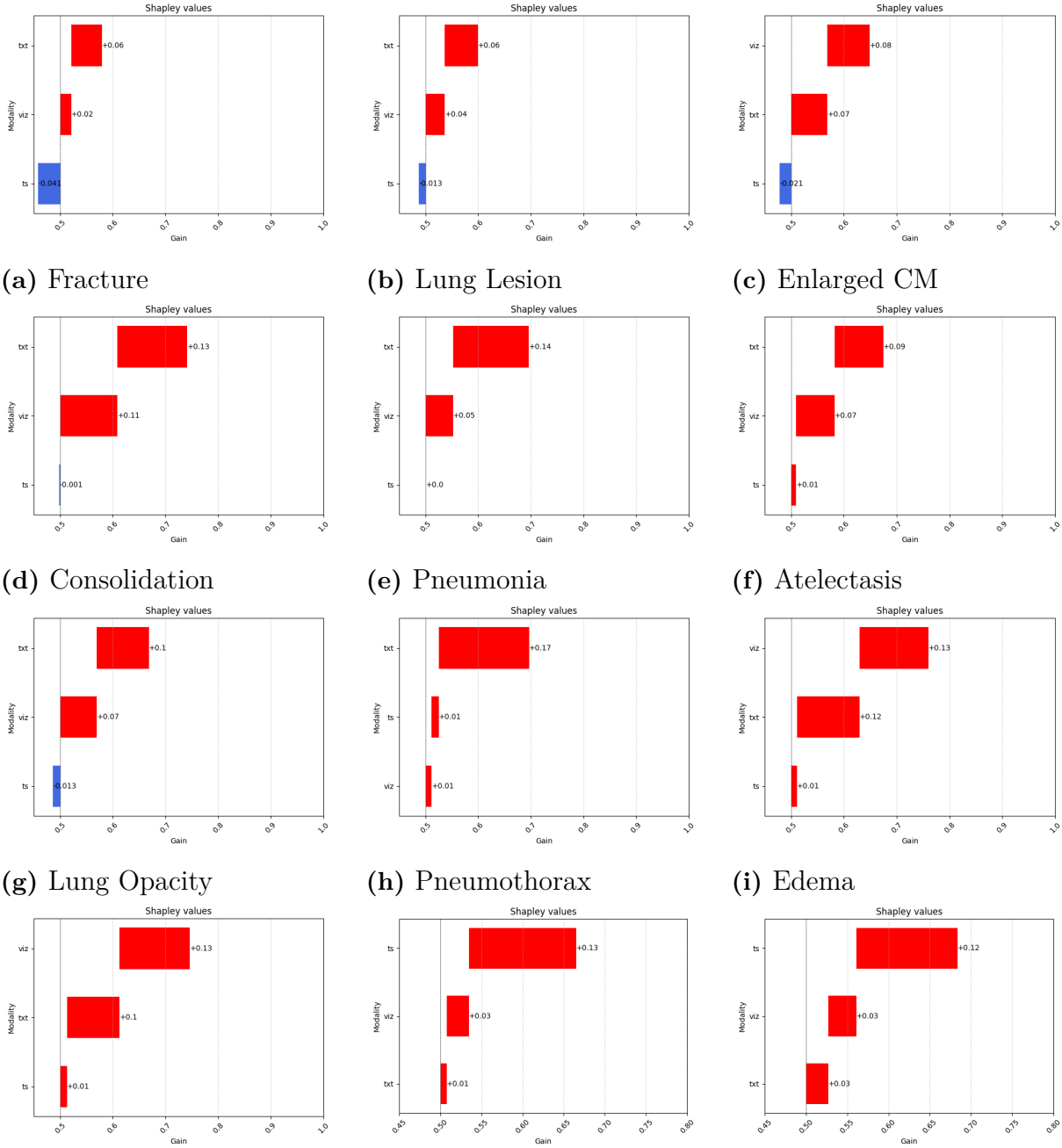
(a) Fracture **(b)** Lung Lesion **(c)** Enlarged CM

(d) Consolidation **(e)** Pneumonia **(f)** Atelectasis

(g) Lung Opacity **(h)** Pneumothorax **(i)** Edema

(j) Cardiomegaly **(k)** Length-of-Stay **(l)** 48h Mortality

Figure 4.4: Shapley-score generated for the different tasks and modalities. Scores are generated using the ROC_AUC-score from the XGBoost models used as benchmarks.



(a) Fracture **(b)** Lung Lesion **(c)** Enlarged CM

(d) Consolidation **(e)** Pneumonia **(f)** Atelectasis

(g) Lung Opacity **(h)** Pneumothorax **(i)** Edema

(j) Cardiomegaly **(k)** Length-of-Stay **(l)** 48h Mortality

Figure 4.5: Shapley-score generated for the different tasks and modalities. Scores are generated using the ROC_AUC-score from the the multimodal model with projectors trained jointly with asymmetric loss.

5

Discussion

This chapter will discuss the results generated. It will cover how we established which model performed the best, what choices were made during training and how that impacted the performance. Furthermore, a discussion around the comparison between the best model trained and the XGBoost model used as benchmark will be presented, discussing differences in performance and importance of different modalities. Moreover, a portion of this chapter will be dedicated to discussing how this project has an effect from an ethical and social standpoint. Finally, further improvements and future works rounds of the discussion.

5.1 Determining the best model

As stated in the results, one determined that the Gemma-2B language model equipped with projectors trained jointly with an asymmetric loss was the best performing model. However, since the scores were quite similar, one had to determine what metrics are of most importance.

When assessing the models, one could take several approaches. A possibility could be to look purely at the averages for all metrics and then make a decision solely based on these. Even if it is sensible, this would imply that that all tasks are equally important but one also needs to take account the frequency of occurence and sizes of the different labels. Furthermore, looking at the average means that a model can have a lot of variety in performance, as a poor performance on one label can be weighed up with an excellent performance on another. In this thesis, we valued a more evenly performing model.

Out of precision, recall, macro-averaged F_1 and AUC, what should impact the most? Considering precision and recall, even though they are telling us information about how good a model is on finding the positive predictions, it does not tell us the predictive performance on the negative labels. If we were to exclusively cares about the positive class, then this is would be a reasonable choice. However, in the healthcare domain and with this given set of tasks, it is of interest to know how good a model is at detecting, for example, if a patient would not die in 48 hours. To do so, we need to take the negative class into account as well. This leads us to the macro-averaged F_1 score, which considers this. As an example, we look at the precision, recall and F_1 score for Joint-Avg-G2B for predicting atelectasis in Table 4.2 and 4.4. In this case, the precision and recall values are excellent, while the macro-averaged F_1 score

is average, indicating that it is performing worse on the negative class. Therefore, F_1 is considered more important than precision and recall. The AUC score is also worth some attention, as it indicates the discriminatory power of the model. A higher AUC score would indicate that a model is more confident in its predictions but the balance between precision and recall for both classes is more informative for these cases with imbalanced datasets.

From the reasoning above, the decision for best model was influenced by evenness in performance regarding the F_1 score followed by the AUC score, which led to Gemma-2B with jointly trained projectors with asymmetric loss crowned the best.

5.2 Training procedure

During the training of the models, several experiments were conducted using different approaches. This section intends to discuss how the different choices affected the performance.

5.2.1 Train projectors jointly or in isolation

The first aspect to bring up is the differences in training the projectors jointly or in isolation. It was clear from the results that a joint training procedure is to prefer to the isolated one. This could have several explanations, starting with the overall structure of the model. Since the input is being split up into different sources, where each source has its own projector, makes it so that training the models in isolation forces the projectors to predict classes of labels where that specific input might not be relevant. This is supported by looking at Figure 4.5, where one can see that e.g time-series data is not particularly relevant but is in the isolated training case forced to be that. Having the projectors trained in isolation means that the language model is only being fed one token of information, which means that this token has to carry enough information to be able to make 12 accurate predictions. Even though some labels might be closely related in the real world (ten of them being lung pathogens), that does not necessarily have to apply to the embedding space of a language model. The projector then has to mediate between all labels, even the ones that might not be of relevance to the input, when projecting the input. Since all projectors share these problems, the final model will consists of projectors that try to contribute equally to every label, even if the source is not relevant, resulting in poor performance.

On the other hand, in the case where the projectors are trained jointly, the problems presented in the previous paragraph are not as prevalent. Training them jointly means that different projectors can get more specialized in conveying information for specific tasks. In addition, the language model also has more tokens to reason around in training which also could be a factor to the performance.

5.2.2 Average or asymmetric loss function

The difference in performance between a model using average loss or asymmetric loss were not large in an absolute sense, but it had different nuances and characteristics.

Starting with the more pragmatic approach, averaging the cross-entropy loss over all labels, means that all labels will be weighed equally to the final loss. This would mean that the model could become very good at some tasks and worse on others. As described previously, this is what the asymmetric loss tries to mitigate. This is supported by the F_1 score in Table 4.1 and 4.2, where the model trained with average loss for some labels has good performance (e.g Lung Lesion) mixed with tasks with poor performance (e.g Lung Opacity). One then has to decide on what is most important - a model trained to be good on average or a model trained to never perform poorly.

5.2.3 Usage of SLM for classification

In this project, the main focus was to enable an SLM with multimodal capabilities for predictive tasks. From this, a big decision had to be made in how one would utilize such a model for performing the relevant objective. The approach taken, with associating each label with a specific token is believed to be sensible, though it could still pose some challenges. Special tokens were ignored, but Gemma-2B and Phi-3 has a vocabulary size of $256,000$ and $32,064$ respectively, and in either case there are many tokens to choose from. In this project, the tokens were chosen arbitrarily, but one should test if the choice of tokens has any effect.

Changing the SLM from Gemma-2B to Phi-3-mini-4k resulted in a worse performance with respect to the F_1-score across all tasks but an increase in performance in regards to AUC in 9 out of 12 cases (see Table 4.1 and 4.2. Reformulated, Phi-3 is worse at balancing precision and recall for the different classes and labels, but it became better at discriminating them. This suggests that the threshold set for predictions, in this case 0.5, might need an adjustment to see the maximal performance from this model in this setting.

If adjusting the threshold is not making an improvement, another reason for the drop in performance could be explained by the embedding spaces of the different SLMs. The embedding space of Gemma-2B and Phi-3-mini-4k is 2048 and 3072 respectively. This means that the projector modules needs to project whatever input it is responsible for to a vector that is 50% larger for Phi-3-mini-4k compared to Gemma-2B. Having a larger input vector means that you can feed more information, which in principle could be favorable, but the sizes of vectors that are being fed to the projectors are quite small. The largest one is of size 1024, meaning that one would effectively try to represent a vector from one space to another that is 3 times the size. If Phi-3-mini-4k performs worse due to its larger embedding space, it indicates that the projector module might face some challenges if one wanted to adopt language models with even larger embedding spaces.

5.2.4 Design of projectors

The other core part of this thesis was to create the projectors. This was not trivial as the structure of the data as well as the size and what the different features represented varied greatly between the modalities and sources. Early testing was done with feed forward neural networks with varying depths, activation functions and normalization layers. Despite the efforts, we realized that it was hard to obtain results better than random guessing, which suggested that the information was not transmitted correctly to the language model. Therefore, the reconstruction part of the projector was added. We then could add a reconstruction error to the loss function. This addition is believed to have helped the projector transfer information more accurately, as the evaluation metrics improved. Even if this was an improvement, judging by the loss plots presented in the results, the training loss is not improved smoothly, indicating that improvements can still be made.

In this project, we used the same architecture for all modalities and sources, but since they vary greatly, further research should be conducted on making them more specific. An approach could to reconsider which modalities should be projected and which ones that could be fed through a text-prompt. For instance, many of the time-series features, see Appendix A.1 and A.2, might serve a better purpose in text form instead of having to try and expand the vectors describing them. Moreover, since the expansion of vectors might be an issue (Sec.5.2.3) one could try the approach used by Moon, Madotto, Lin, *et al.* [1] and directly map images to the embedding space without the intermediate step of converting it using a DenseNet for example. This would mean that the input are e.g $224 * 224 = 50,176$ pixels for a 224 by 224 image, eliminating the potential issue of vector expansion.

5.3 Performance

Once the best model using the method proposed in this thesis was selected, we compared that with the XGBoost model. As stated in the results, our model did not manage to reach the AUC and macro-averaged F_1 score that was set by XGBoost, although competitive on some tasks. These results were consistent with Shwartz-Ziv and Armon [9], and somewhat expected from the introduction of the paper. Partly because of the raw performance of XGBoost as the scores of many tasks are excellent, but also because of how the data was structured. Even though one originally had different modalities, they all were pre-processed into feature vectors. These could be viewed as just tabular data, which is a data type that XGBoost excels at[9]. Furthermore, a comparison between the best single-sourced model and the multimodal model suggests that the multimodal model is more capable, strengthening the base of this project and is consistent with e.g Moon, Madotto, Lin, *et al.* [1].

What is positive about the comparison is that the model's precision and recall for the lung pathogens are very competitive with XGBoost. The main difference is that when the XGBoost model is performing worse, it tends to have worse recall while

retaining a high level of precision (see pneumothorax in Table 4.4). In those cases, the language model instead increases its recall to the loss of precision. Which of these scenarios is the preferable one would largely depend on the domain. Since this is the healthcare domain, one would suggest that it is more important to find as many instances as possible with the risk of having some false positives, with the motivation that medical doctors could analyze the data itself and confirm or deny with their medical expertise. If one were to chose the other alternative, one could be more certain of the verdict of the patients that were flagged for the pathogen but resulting in a larger set of positive patients not getting spotted.

Another aspect to emphasize is that the model developed makes all of these predictions at the same time, while one needs to fit one XGBoost model to every task. This is manageable for smaller sets of tasks, but keeping track of potentially hundreds of models could be cumbersome. Our approach streamlines this into one model, which also makes it more effective if one wants to add more modalities or tasks. It is not as apparent if one adds a new task, since our model has to be re-trained, and a new one has to be created for with XGBoost. However, for the case where a new source or modality is being added, the model proposed in this thesis still only has to be re-trained once while, in the case of using XGBoost, one has to create new models for every task. From an efficiency and scalability standpoint, using the multimodal approach with a language model is to prefer.

5.3.1 Shapley values

When analyzing the shapley values for the different tasks between XGBoost and our model, one found both similarities and contrasts. They share what modalities should dominate certain tasks, with textual and visual data being the most important for lung pathogens. This is believable from a human perspective as an x-ray image should carry more relevant information regarding a potential lung disease than e.g the mean heart-rate during the stay, but needs further research to be verified. What also is interesting is that the time-series modality has very little impact on the lung pathogens, and even negative impact on some. This could further strengthen the proposal of changing how that data is being used or projected.

Furthermore, it should be noted that the shapley values are calculated using the AUC as contribution function. For future works, it could be of interest to instead calculate the contributions based on the macro-averaged F_1 score and see if the behaviour changes.

5.4 Ethical aspects

For this model to work in a real-life situation the model has to be improved, but even if one could show very good results, there are many things to take into consideration.

Firstly, this model should operate in the healthcare domain which means that it has to be trained on healthcare data. During this project, one had the possibility of

using MIMIC-IV but in reality, hospitals will be limited by patient data laws when trying to train it. One option is to take a model that has been trained on a larger dataset such as MIMIC-IV and try to fine-tune it using their local data, but then one has to be able to ensure that the local data is enough to make it specialized for their use-cases without it being to general. Another option would be to train it from scratch. It is not likely that a single hospital has enough data to train a model to the standard they want, which in turn means that hospitals need to work together and share data between them. Then patient data sharing laws need to be redefined for this to work and while it is possible, it will not be a quick and easy change. An alternative to data sharing is to take a federated learning approach [39], which would enable hospitals to collaboratively train a model without sharing sensitive data.

Secondly, one needs to consider the impact the model could have on medical decisions. Having an overconfident model can lead to a lack of due diligence as more and more trust goes into the model, possibly resulting improper treatments. On the other hand, having a model that is very uncertain serves no purpose and will be a waste of energy and resources. Therefore it is of importance to have a clear understanding of what the model is capable of and what one should look out for. A good example of this is what previously was discussed in relation to the performance of the model, that the model had a tendency to flag more people than necessary. Having this prior knowledge can make the use of the model more effective.

Finally, it should be stressed that a model of this kind is in no way a substitute to actual medical practitioners. The main goal is to develop a supporting device that facilitates medical decisions, not making them.

5.5 Future works

Excluding the suggestions of how to improve the model already made in the discussion, there still is room for improvement. What is yet to be mentioned is fine-tuning on the language model itself. Currently, the language model has remained unchanged, but parameter efficient fine-tuning techniques such as LoRA has shown to improve performance as well[40].

6

Conclusion

This thesis delved into the world of multimodal models in the healthcare domain. We posed the question - can one make a language model reason among different modalities and is it performing better than using only a single source? This is a hot topic, especially in healthcare where data can take many different forms and not all is fit to be described textually.

To answer the questions posed, one had to come up with a plan of equipping a language model with multimodal capabilities. With inspiration drawn from previous work, the projector module was developed. By keeping the weights of the language model frozen, the projectors were trained in a fashion where the goal was to transmit as much information as possible to the language model. The language model could then make multiple predictions of a fixed set of lung pathogens and other relevant tasks, from the same input.

The results showed that a multimodal model outperforms a single-sourced model in the vast majority of tasks presented. Comparing the multimodal model to an established model like XGBoost showed that although XGBoost is better, in some scenarios the proposed model could be of more use. Furthermore, contribution scores for the different modalities were presented and their importance was discussed.

Moreover, the thesis discusses the streamlined approach that this project takes where one only has to keep track of one model instead of all XGBoost models that has to be fitted for each task. The multimodal approach presented generalizes well for more modalities, sources and tasks.

While the model has it advantages, improvements are presented regarding what could be done differently and possible reconsiderations. This includes refining the projector module, fine-tuning the language model itself and re-evaluate which sources should belong to what modality are all discussed. Since the project is based in the healthcare domain, ethical considerations were also taken into account.

In conclusion, while the multimodal approach chosen for this project is an improvement over the single-source case, one should strive to improve on this. This thesis hopes to spark further improvements and advancements in this field. Even if having capable multimodal models will pose challenges regarding patient data safety, predictive certainty among other, it will be of great importance in the future and worth the investment.

Bibliography

[1] S. Moon, A. Madotto, Z. Lin, *et al.*, "Anymal: An efficient and scalable any-modality augmented language model," *ArXiv*, vol. abs/2309.16058, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:263137930.

[2] A. Belyaeva, J. Cosentino, F. Hormozdiari, *et al.*, *Multimodal llms for health grounded in individual-specific data*, 2023. arXiv: 2307.09018 [q-bio.QM].

[3] A. E. W. Johnson, L. Bulgarelli, L. Shen, *et al.*, "MIMIC-IV, a freely accessible electronic health record dataset," *Scientific Data*, vol. 10, no. 1, p. 1, Jan. 2023. DOI: 10.1038/s41597-022-01899-x. [Online]. Available: https://doi.org/10.1038/s41597-022-01899-x.

[4] L. R. Soenksen, Y. Ma, C. Zeng, *et al.*, "Integrated multimodal artificial intelligence framework for healthcare applications," *npj Digital Medicine*, vol. 5, no. 1, p. 149, Sep. 2022, ISSN: 2398-6352. DOI: 10.1038/s41746-022-00689-4. [Online]. Available: https://doi.org/10.1038/s41746-022-00689-4.

[5] *Health Insurance Portability and Accountability Act of 1996 — aspe.hhs.gov*, https://aspe.hhs.gov/reports/health-insurance-portability-accountability-act-1996, [Accessed 24-05-2024].

[6] L. Shapley, "A value for n-person games," in *Contributions to the Theory of Games (AM-28), Volume II*, H. Kuhn and A. Tucker, Eds., Princeton University Press, 1953, pp. 307–318. DOI: 10.1515/9781400881970-018.

[7] H. Touvron, L. Martin, K. Stone, *et al.*, *Llama 2: Open foundation and fine-tuned chat models*, 2023. arXiv: 2307.09288 [cs.CL].

[8] H. W. Chung, L. Hou, S. Longpre, *et al.*, *Scaling instruction-finetuned language models*, 2022. arXiv: 2210.11416 [cs.LG].

[9] R. Shwartz-Ziv and A. Armon, *Tabular data: Deep learning is not all you need*, 2021. arXiv: 2106.03253 [cs.LG].

[10] C. Kingsford and S. L. Salzberg, "What are decision trees?" *Nature Biotechnology*, vol. 26, no. 9, pp. 1011–1013, Sep. 2008, ISSN: 1546-1696. DOI: 10.1038/nbt0908-1011. [Online]. Available: https://doi.org/10.1038/nbt0908-1011.

[11] S. O. Arik and T. Pfister, *Tabnet: Attentive interpretable tabular learning*, 2020. arXiv: 1908.07442 [cs.LG].

[12] S. Popov, S. Morozov, and A. Babenko, *Neural oblivious decision ensembles for deep learning on tabular data*, 2019. arXiv: 1909.06312 [cs.LG].

[13] A. Abutbul, G. Elidan, L. Katzir, and R. El-Yaniv, *Dnf-net: A neural architecture for tabular data*, 2020. arXiv: 2006.06465 [cs.LG].

[14] *GitHub - baosenguo/Kaggle-MoA-2nd-Place-Solution — github.com*, https://github.com/baosenguo/Kaggle-MoA-2nd-Place-Solution, [Accessed 24-05-2024].

[15] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16, ACM, Aug. 2016. DOI: 10.1145/2939672.2939785. [Online]. Available: http://dx.doi.org/10.1145/2939672.2939785.

[16] J. P. Cohen, J. D. Viviano, P. Bertin, *et al.*, *Torchxrayvision: A library of chest x-ray datasets and models*, 2021. arXiv: 2111.00595 [eess.IV].

[17] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, *Densely connected convolutional networks*, 2018. arXiv: 1608.06993 [cs.CV].

[18] *CheXpert: Chest X-rays — aimi.stanford.edu*, https://aimi.stanford.edu/chexpert-chest-x-rays, [Accessed 14-05-2024].

[19] E. Alsentzer, J. R. Murphy, W. Boag, *et al.*, *Publicly available clinical bert embeddings*, 2019. arXiv: 1904.03323 [cs.CL].

[20] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, *Bert: Pre-training of deep bidirectional transformers for language understanding*, 2019. arXiv: 1810.04805 [cs.CL].

[21] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, *Attention is all you need*, 2023. arXiv: 1706.03762 [cs.CL].

[22] J. Lee, W. Yoon, S. Kim, *et al.*, "Biobert: A pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, vol. 36, no. 4, J. Wren, Ed., pp. 1234–1240, Sep. 2019, ISSN: 1367-4811. DOI: 10.1093/bioinformatics/btz682. [Online]. Available: http://dx.doi.org/10.1093/bioinformatics/btz682.

[23] A. Cabello, *The Evolution of Language Models: A Journey Through Time — adria.cabello*, https://medium.com/@adria.cabello/the-evolution-of-language-models-a-journey-through-time-3179f72ae7eb, [Accessed 16-05-2024].

[24] M. Abdin, S. A. Jacobs, A. A. Awan, *et al.*, *Phi-3 technical report: A highly capable language model locally on your phone*, 2024. arXiv: 2404.14219 [cs.CL].

[25] G. Team, T. Mesnard, C. Hardin, *et al.*, *Gemma: Open models based on gemini research and technology*, 2024. arXiv: 2403.08295 [cs.CL].

[26] L. Parcalabescu, N. Trost, and A. Frank, *What is multimodality?* 2021. arXiv: 2103.06304 [cs.AI].

[27] OpenAI, J. Achiam, S. Adler, *et al.*, *Gpt-4 technical report*, 2024. arXiv: 2303.08774 [cs.CL].

[28] S. Yin, C. Fu, S. Zhao, *et al.*, *A survey on multimodal large language models*, 2024. arXiv: 2306.13549 [cs.CV].

[29] R. Caruana, "Multitask learning," in *Learning to Learn*, S. Thrun and L. Pratt, Eds. Boston, MA: Springer US, 1998, pp. 95–133, ISBN: 978-1-4615-5529-2. DOI: 10.1007/978-1-4615-5529-2_5. [Online]. Available: https://doi.org/10.1007/978-1-4615-5529-2_5.

[30] C. Ngufor, S. Upadhyaya, D. Murphree, D. Kor, and J. Pathak, "Multi-task learning with selective cross-task transfer for predicting bleeding and other important patient outcomes," in *2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, 2015, pp. 1–8. DOI: 10.1109/DSAA.2015.7344836.

[31] T. A. E. Team, *How did Binary Cross-Entropy Loss Come into Existence? — pub.towardsai.net*, https://pub.towardsai.net/how-did-binary-cross-entropy-loss-come-into-existence-68e38509d2b, [Accessed 24-05-2024].

[32] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, *Focal loss for dense object detection*, 2018. arXiv: 1708.02002 [cs.CV].

[33] E. Ben-Baruch, T. Ridnik, N. Zamir, *et al.*, *Asymmetric loss for multi-label classification*, 2021. arXiv: 2009.14119 [cs.CV].

[34] R. Abhinav, *Improving Class Imbalance with Class Weights in Machine Learning — ravi.abhinav4*, https://medium.com/@ravi.abhinav4/improving-class-imbalance-with-class-weights-in-machine-learning-af072fdd4aa4, [Accessed 22-05-2024].

[35] Y. Sasaki, "The truth of the f-measure," *Teach Tutor Mater*, Jan. 2007.

[36] *Understanding Micro, Macro, and Weighted Averages for Scikit-Learn metrics in multi-class classification with example — iamirmasoud.com*, https://iamirmasoud.com/2022/06/19/understanding-micro-macro-and-weighted-averages-for-scikit-learn-metrics-in-multi-class-classification-with-example/, [Accessed 22-05-2024].

[37] K. Aas, M. Jullum, and A. Løland, *Explaining individual predictions when features are dependent: More accurate approximations to shapley values*, 2020. arXiv: 1903.10464 [stat.ML].

[38] S. Lundberg and S.-I. Lee, *A unified approach to interpreting model predictions*, 2017. arXiv: 1705.07874 [cs.AI].

[39] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, *Communication-efficient learning of deep networks from decentralized data*, 2023. arXiv: 1602.05629 [cs.LG].

[40] E. J. Hu, Y. Shen, P. Wallis, *et al.*, *Lora: Low-rank adaptation of large language models*, 2021. arXiv: 2106.09685 [cs.CL].

A

Appendix

A.1 Events recorded of patients

- Chart events - Heart Rate, Non Invasive Blood Pressure systolic, Non Invasive Blood Pressure diastolic, Non Invasive Blood Pressure mean, Respiratory Rate, O2 saturation pulseoxymetry, GCS - Verbal Response, GCS - Eye Opening, GCS - Motor Response
- Lab events - Glucose, Potassium, Sodium, Chloride, Creatinine, Urea Nitrogen, Bicarbonate, Anion Gap, Hemoglobin, Hematocrit, Magnesium, Platelet Count, Phosphate, White Blood Cells, Calcium, Total, MCH, Red Blood Cells, MCHC, MCV, RDW, Platelet Count, Neutrophils, Vancomycin
- Procedure events - Foley Catheter, PICC Line, Intubation, Peritoneal Dialysis, Bronchoscopy, EEG, Dialysis - CRRT, Dialysis Catheter, Chest Tube Removed, Hemodialysis

A.2 Features extracted from events

- Maximum value
- Minimum value
- Mean
- Variance
- Average difference
- Average absolute difference
- Maximal difference
- Sum of absolute difference
- Difference from first recorded value to last
- Number of peaks, threshold being the median
- Trend of the event

DEPARTMENT OF SOME SUBJECT OR TECHNOLOGY
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY