Automating Radiology Report Generation with CDGPT3.5: A Deep Learning Approach for Enhancing Medical Image Interpretation

by

Khaled Mahmud
20101007

A thesis submitted to the School of Data and Sciences
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

School of Data and Sciences
Brac University
January 2025

Declaration

It is hereby declared that

1. The thesis submitted is my own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. I have acknowledged all main sources of help.

Student's Full Name & Signature:

Khaled Mahmud
20101007

Approval

The thesis titled "Automating Radiology Report Generation with CDGPT3.5: A Deep Learning Approach for Enhancing Medical Image Interpretation" submitted by

1. Khaled Mahmud (20101007)

Of Fall, 2024 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on February 03, 2025.

Examining Committee:

Supervisor:
(Member)

Dr. Md Ashraful Alam
Associate Professor
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Md. Golam Rabiul Alam, PhD
Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Dr. Sadia Hamid Kazi
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

The process of generating radiology reports at healthcare facilities is time-consuming and needs vast experience from the Radiologists. Here I am introducing a deep learning approach for the automatic generation of radiological reports using chest X-ray and mammogram images as input. This technique consists of three major stages: (1) tuning a pre-trained CheXNet model to label the relevant tags from images during training time, with each tag representing visual signal(s); (2) extracting weighted high-level semantic features out of these embeddings; and then set up the retrieved visual/semantic signals as conditioning signals towards a GPT-2 medical report generator. It was trained using the publicly available IU-XRay dataset (with reports for chest X-rays) and a new Categorized Digital Database for Low energy and Subtracted Contrast Enhanced Spectral Mammography images (CDD-CESM) dataset that included mammograms with full-text reports. There is a new hierarchy mammogram dataset because of the few public and open-source resources. I used this word-overlap metric to assess the reports generated as well as some new semantic similarity measures. The researchers showed that the proposed model, (CDGPT3.5) achieved very competitive quantitative results compared to several non-hierarchical recurrent and transformer-based models while being trained much faster. Significantly, it can be used with any programming language and easily modified to support different datasets without changing the architecture. This contribution is one of the prior attempts at to publicly available dataset for contrast-enhanced mammogram samples along with both semantic and visual report modalities, utilizing pre-trained transformer models to generate reports that encourage further research in multimodal medical imaging.

Keywords: ChexNet, Transformer, NLP, Report generation.

Acknowledgement

I begin by offering our heartfelt gratitude to the Almighty for guiding me through this thesis journey without significant disruptions.
My sincere appreciation goes to my esteemed supervisor, Dr. Md. Ashraful Alam sir for his unwavering support and invaluable guidance that have been pivotal in shaping my research.
I would also like to extend my thanks to my parents, whose boundless encouragement and prayers have been the driving force behind my academic accomplishments. This thesis would not have reached its completion without the collective support and belief of these individuals, for which I am deeply thankful.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Imaging technologies for medicine are essential across the world, furnishing important insights for both the diagnosis of diseases and the supervision of patients' health progress. However, because of the need for thorough expertise, radiologists must meticulously analyze each image and compile extensive reports. The manual process is time-consuming as well as requires a major degree of skill and experience for accuracy assurance. Even though radiologists aim to deliver precise and unambiguous conclusions, a lot of reports conclude with findings that require additional diagnostic testing through biopsies or advanced imaging techniques. Due to the complexity of the diseases possible to diagnose via imaging, these follow-up procedures are needed more often than not.
In addition, the duration necessary to develop these reports is a vital challenge. On average, it takes more than 10 minutes for a radiologist to complete a report, which greatly limits their ability to assess cases each day, notably in crowded or limited resource locations. The escalating demand for medical imaging alongside the deficit of trained radiologists is imposing serious stress on healthcare systems around the world, greatly affecting high-volume locations such as metropolitan hospitals. The focus of this thesis is on presenting a machine-learning solution for the automatic generation of radiological reports from medical images, with the dual purpose of increasing efficiency and accuracy while dealing with the issue of insufficient annotated datasets for mammography. In conjunction with the National Cancer Institute of Egypt, a new dataset is being created to help enhance radiological reporting efficiency via artificial intelligence (AI).

1.1 Motivation

The rationale for this study is a result of the important need for reducing radiologist workloads and enhancing the timeliness and precision of medical diagnoses. Both respiratory disorders and breast cancer are major globally recognized health threats, and early diagnosis via medical imaging can markedly improve patient results. For instance, pneumonia was accountable for 2.56 million deaths in 2017, indicating how important it is for chest disease diagnosis to be early and accurate. Just in 2020, 2.3 million women received a diagnosis of breast cancer, leading to 685,000 deaths worldwide [35]. Over the past five years, breast cancer, the most common cancer worldwide, has resulted in 7.8 million women receiving a diagnosis [49].
In spite of the essential part radiological reports serve in the diagnosis of these condi-

tions, the procedure is mainly manual and linked to the expertise of the radiologist. Radiologists dedicate about 35% of their clinical time to diagnostic reporting, and chest X-rays are a major part of that [34], [36], [46]. This is especially challenging in high-demand scenarios, where report generation delays may endanger patient care. These difficulties demonstrate the requirement for automatic systems that output medical image reports with minimal human oversight. Recent breakthroughs in deep learning, mostly in natural language processing (NLP) and image captioning, present hopeful routes for addressing this issue [39]. The development of an automated report generation system from chest X-rays and mammograms is intended to cut down the time and effort needed for diagnostic reporting, so radiologists can concentrate on more demanding tasks.
This research also seeks to enhance efficiency and address the critical issue of unavailable comprehensive datasets for mammography, key for training deep learning models. A deficiency of annotated medical images, particularly mammograms, has stopped the growth of AI-driven solutions in this field. Inspired by this deficiency, I linked up with the National Cancer Institute to develop a novel dataset that features mammograms paired with complete radiology reports, which allows for the training and assessment of AI models in the field of automatic report generation.

1.2 Research Problem

The main research concern addressed by this thesis is the automation of detailed radiology reports from chest X-rays and mammograms, framed as an image captioning activity [31]. Image captioning consists of generating textual descriptions for images, which, in the healthcare field, means the production of reports that include, for instance, "Impression," "Findings," and "Tags"). At present, the method used for radiological reporting is manual and depends largely on the radiologist's skill in correctly interpreting medical images. Due to the active clinical environments, this procedure can be inconsistent, last a while, and be prone to human mistakes. In addition, a deep knowledge of normal and pathological elements is essential for medical image interpretation, because the differences can be slight and hard to identify.
The challenges faced during the automation of this process are diverse, including the difficulty in preparing a model to recognize complex medical conditions from image data, crafting humanized language descriptions, and verifying that those reports are not just correct but also clinically useful. In addition, the lack of substantive, annotated datasets, especially in relation to mammography, complicates the creation of effective AI models in this field. In order to rise to these challenges, the study examines incorporating transformer-based models, which have shown better performance in NLP activities, to produce radiological reports from both chest X-rays and mammograms [11]. The intention is to set up a system that both renders accurate and cohesive reports and helps alleviate radiologists' workloads, so they can give more attention to vital aspects of patient care.

1.3 Research Objective

The purpose of this thesis is to produce a deep learning framework aimed at automating the generation of medical reports derived from chest X-ray and mammo-

gram images. The specific objectives of this research are outlined as follows:

- Dataset Development: The priority is to generate a national dataset that comprises mammography images and all text reports. Looking at the National Cancer Institute, this dataset will provide important support for training and evaluating deep learning models.
- Model Development: The principal goal of this study is the creation of a transformer-based model able to produce extensive medical reports. This framework will extract visual and semantic characteristics from medical images to generate reports that are both precise and detailed.
- Evaluation and Analysis: We will perform both quantitative and qualitative evaluations in order to evaluate the performance of the proposed model. A quantitative assessment will involve measurements such as BLEU and ROUGE scores, designed to gauge how alike the generated reports are to the ground truth. Qualitative evaluation will depend on input from radiologists to make certain that the produced reports hold clinical relevance.
- Comparison with State-of-the-Art Models: Yet another goal is to evaluate the proposed model when compared to established techniques for generating automatic reports. This will consist of evaluations against models created using datasets accessible to the public such as the Indiana University Chest X-Ray dataset (IU-XRay).

Chapter 2

Background

In this chapter, major themes important to the study are presented. Foundational concepts in deep learning, including transformers and transfer learning, are the subject of Section 2.1, while Section 2.4 reviews different image classification models and Section 2.5 examines the application of deep learning within the medical domain.

2.1 Deep Learning

Since the early 2000s, neural networks have been important aspects of machine learning, yet recent improvements in GPU and CPU computing power have made deep neural networks both functional and highly effective. These networks have accomplished leading results in numerous machine learning tasks, not demanding extensive feature engineering. Deep learning is a term that denotes deep artificial neural networks inspired by the layout of the human brain [36]. By scaling effectively with vast datasets, they become applicable to real-world challenges such as image classification or language translation.

2.2 Transformers

Before the advent of transformers, Recurrent Neural Networks (RNNs) were the foundational models in natural language processing (NLP) alongside machine translation. Still, the sequential characteristics of RNNs impeded their potential for complete parallel processing with GPUs. In [32], the proposal of the transformer model included attention mechanisms to effectively quicken the training. The transformer architecture has encoders and decoders that are like those found in RNNs. The encoder consists of six layers, each containing two key components: an auto-attention mechanism along with a feed-forward neural network. The input sequence undergoes conversion into embeddings at the bottom encoder layer, where each word's embedding is processed in isolation, permitting parallelization and quicker training. The self-attention mechanism relies on three trainable weight matrices: Query, Key, and Value. The Query matrix reflects the axes for which attention must be determined, the Key matrix contains the axes that need attention, and the Value matrix generates the resultant axes. Moreover, positional encoding is implemented to account for the arrangement of words in the input sequence by embedding distance vectors that are learnable. The decoder includes six layers that mainly correspond

to the encoder's design. This methodology is different, primarily because it places the top encoder's output into its attention mechanism, which emphasizes important parts of the input sequence. In addition, the decoder's self-attention attends to past positions while obscuring future ones. The linear layer and a softmax function process the output to predict the next word in the sequence, where the model is trained with cross-entropy loss.
Although originally designed for machine translation, the transformer model was later adapted for broader language modeling tasks with some modifications. The GPT-2 model [42] stripped away its encoder layers, which left only the decoder layers; as a result, it became fit for autoregressive tasks like text generation. Masked self-attention in GPT-2 shelters the model from future words during the training phase. A collection of variants of GPT-2 was produced with various layer configurations, and in [43] a distilled version emerged using the teacher-student approach, which permits a compact model to mirror a larger one. As an example, the performance of DistilGPT2 is approximately the same as GPT-2, but it is faster by 37% and has fewer parameters. An additional important model, BERT [39], substitutes the transformer's encoder block for the decoder, permitting it to catch bidirectional context by focusing on both prior and subsequent tokens, thereby advancing word embeddings for tasks like sentiment analysis. While training, BERT masks a number of words randomly and the model learns to predict what those masked tokens represent.

2.3 Transfer Learning

Transfer learning describes the method of exploiting a model created from a single dataset for use in a different challenge or area. Transfer learning stands out as a valuable option when the amount of data is vast for deep learning models, particularly in situations where only a little dataset exists. In addition, it has the ability to notably lower training time and boost model generalization. By analyzing the data, participants in the ILSVRC competition found that transfer learning has proven particularly fruitful in image classification, producing models such as VGG16, DenseNet, and ResNet that resulted from the broad ImageNet dataset. As a result of their efficiency in learning fundamental image features, these models are able to fit new datasets. Depending on the task at hand, different strategies can be employed when using pre-trained models:

- Freeze all layers: Using the model without modifications, known as fine-tuning, is the strategy proposed; the weights remain the same, making it attractive for situations where there is significant likeness between the new dataset and the initial data, such as recognizing people in pictures, where 'person' is a class in ImageNet.

- Freeze early layers: Towards the model input, layers extract low-level features, while layers deeper in adopt a learning approach to pecuniary patterns. In situations where the datasets show some variations, it can help to retraining exclusively the later layers.

- Train all layers: If there is a notable gap between the new dataset and the original, it might become necessary to change the whole model.

Transfer learning has changed NLP considerably as a result of advances in transformer-based models. Models that are broad, including BERT [39] and GPT-2 [42], trained with large datasets, produce embeddings of better quality fit for multiple NLP tasks. These pre-trained models can be adapted for specific tasks such as:

- Domain-Specific Language Modeling: Fitting a language model that has been pre-trained in general language to text from a specific domain can greatly raise performance on tasks linked to that field.
- Text Classification: The placement of the classifier at the top of a pre-trained transformer permits the model to perform spam detection with few extra training needs.
- Task-Specific Fine-Tuning: We can suitably adjust pre-trained models for tasks such as text summarization or question answering.

Finally, the fast development of deep learning, particularly fuelled by transformers and transfer learning, has spurred changes in several domains, from classifying images to interpreting language naturally, providing flexible tools for modern machine learning tasks.

2.4 Image Classification

The purpose of image classification is to retrieve important data from a photo and to automatically categorize it into a known segment. Convolutional neural networks (CNNs), part of the 1998 research by LeCun et al. [2], are vital to a variety of image classification tasks. Rather than traditional artificial neural networks (ANNs), which have a full connection between every pixel and each node in the succeeding layer, CNNs completely connect just the last layer, which permits them to train considerably faster. Also, ANNs reject spatial information and alterations in images because they flatten pixels into connections to nodes within the next layers. The convolutional layer in CNNs resolves this by performing a dot product between a kernel—a learnable parameter set—and the relevant part of the image's receptive field.

CNN models proved themselves capable of performing tasks in image classification with the arrival of ImageNet [8]. By implementing convolutional layers widely, AlexNet [9] cut the error rate for the ImageNet dataset from 25% down to 15%. The total layers in AlexNet are 8, consisting of 5 convolutional layers and 3 fully connected layers, sharing 60 million parameters altogether. The use of 16 convolutional layers with smaller kernel sizes reduced the error rate to 7.3% with the VGG16 model [17] following AlexNet. The model was able to capture fine details owing to the smaller kernels and to distinguish extensive spatial characteristics in images, thanks to the additional depth.

The Inception V1 model from 2015 [18] improved the situation by lowering the error rate to 6.6%. It reached this goal by using varied kernel sizes within a layer and by adding 1x1 convolutions to improve performance. The technological architecture at its inception attempted multiple kernel sizes in one layer and merged the resulting features. That same year brought us ResNet [24], which, in turn, lowered the error rate to 3.57%. Skip connections in ResNet helped soften the vanishing and exploding

gradient problems in deep networks. The ability of these skip connections to leapfrog layers posing obstacles to training permitted the utilization of deeper models while preserving performance.
DenseNets [27], following ResNets, optimized the connectivity structure in order to maximize gradient flow by connecting each layer to the ones that come after. The connectivity design used by DenseNets resulted in using less parameter count than that of ResNet and Inception networks, mostly via the reusage of feature maps and adding only a few more. With the increase in model size, there was an exponential growth in parameters, making a need for better efficiency in architectures more apparent.
To solve this issue, EfficientNets [45] provides a design that is more compact alongside greater power. Before EfficientNets emerged, the normal approach for scaling ConvNets involved boosting depth (the number of layers), width (the number of channels), or the resolution of the images. EfficientNets has released compound scaling, which helps to sustain network efficiency by balancing depth, width, and resolution. This method performed admirably with architectures that are current, including MobileNet [26] and ResNet [22], by leveraging compound scaling. By scaling the EfficientNet-B0 model, a family of EfficientNet versions emerged, including EfficientNet B1 through B7; the model development used the neural architecture search technique MNasNet [44]. EfficientNet-B7 has established a new performance standard by delivering 84.4% top-1 accuracy, which is 8.4 times more compact and 6.1 times quicker than predecessors.

2.5 The application of Deep Learning in medical areas

Over the last few years, deep learning has completely reshaped the medical field, giving healthcare professionals instruments that are modernizing their practices. Deep learning along with AI technologies has sped up and improved the accuracy of processing medical information. The improvements have caused applications such as medical imaging analysis, chatbots for interpreting patterns of patient symptoms, and systems that diagnose certain types of cancer. Medical imaging, particularly, has experienced important advancements due to deep learning techniques, which facilitate both early disease detection and tailored care for patients.

2.5.1 Medical Imaging Techniques

Medical imaging is the term for various approaches to visualizing different body parts. The most used imaging approach is X-rays thanks to their quick turnaround and affordable price. They are repeatedly used to evaluate bone fractures and to diagnose instances of pneumonia. Due to being inherently two-dimensional, X-ray images generally experience standard preprocessing approaches like normalization and standardization, but they also face noise from radiation. The PURE-LET method models noise as a Poisson distribution [6], reduces noise in X-ray images by minimizing the mean squared error between the denoised and the noisy images. Autoencoders are among the deep learning approaches that have been applied to denoise X-rays by learning relations between messy and tidy images [21].

Faithful to its nature, a CT scan utilizes multiple two-dimensional images to form a 3D representation of the internal organs and tissues. Like X-rays, CT scans depend on radiation and are likely to produce noise artifacts. Approaches created for removing noise from X-rays are frequently also suitable for CT scans. Due to the image generation of CT scans in high-resolution formats like 12-bit or 16-bit DICOM, it's common during training to normalize pixel intensities to match the standard image formats expected by pretrained models.
Detailed cross-sectional images of body parts supplied by magnetic resonance imaging (MRI) are of superior quality to those from CT scans. Nevertheless, the difficulty in preprocessing MRI images originates from the inconsistent intensity scale. The technique [30] employed by Nyul is commonly used to normalize MRI intensities through the learning of a transformation function from a large data set. Using high-frequency acoustic waves instead of radiation, ultrasound imaging generates images of lower quality but provides a valuable tool for determining tumors in regions where X-rays are ineffective. The use of deep learning strategies, such as generative adversarial networks (GANs), has led to the generation of synthetic ultrasound images and the augmentation of datasets [40].

2.5.2 Machine Learning in Radiology

Radiology has seen great benefits from using machine learning, and in particular, CNNs, which provide support for disease classification and the generation of radiology reports. A major obstacle in the field of medical imaging is the absence of ample annotated datasets since labeling medical images can be laborious. This problem is generally managed by using transfer learning. As an example, a pre-trained inception model tuned with millions of images produced a match for the performance of certified dermatologists on a dataset consisting of 130,000 skin cancer images [38]. DenseNet121 is the architecture that The ChexNet [31] used to categorize thoracic diseases derived from chest X-rays. Fine-tuned on the ChestX-ray14 dataset and based on pretraining on ImageNet, ChexNet demonstrated better performance than a team of Stanford University radiologists in recognizing conditions including pneumonia.
Researchers have turned to 3D convolutions, particularly in the context of medical imaging, specifically when analyzing MRI and CT images, as they can articulate more detailed spatial context. As an illustration, a 3D CNN trained to differentiate Alzheimer's disease from MRI images [41] performed better than classic 2D convolutions. Also, lesion classification is a more difficult task than general exam classification, requiring multi-stream CNNs to include 3D spatial information [28].
Organ localization is an important responsibility in medical image analysis, especially during segmentation tasks, which serve for surgical planning and diagnosis. U-Net [16], one of the major models for medical image segmentation, employs an encoder-decoder structure to obtain detailed localization. Variants like 3D U-Net [20] have made their way into applications for 3D medical imaging tasks like the identification of vascular borders. In addition, medical report generation is a developing sector where visual models extract image features, and language models produce matching reports.

Chapter 3

Methodology

This proposed model consists of two key components, as depicted in Figure 3.1: the report generation pathway and the automatic segmentation pathway are two strategies. The report generation pathway incorporates two primary models: The two ideas put forth are a visual model and a transformer model.
Based on this image, the encoder part of this visual model converts chest X-rays into vectors, using a fine-tuned ChexNet model [31] initially trained for angiogram classification of 14 diseases on chest X-rays. I further fine-tune ChexNet to predict the most commonly occurring labels in this dataset. The output of this visual model is an array of visual features and the confidence of the predicted label set. For implementing medical semantic mappings, the confidence score of each label is then summed up with its weighted medical score obtained from the word2vec pre-existing word-embodied dictionary [11], which was constructed through a vast medical data set. This process produces weighted semantic features itself, as well as the use of visual features for conditioning the decoder.
In this architecture, the decoder is another transformer model called CDGPT3.5, an abbreviated form for conditioned distil generative pre-trained transformer 3.5. This model is a version of the DistilGPT3.5 model [43], with a changed attention mechanism where I added more variance components. These help the model to capture both, visual and semantic representations of the image, the latter of which assists the model in producing voluminous reports on the patient's condition.
Besides, the report generation approach, the second contribution of this paper is the proposal of an automatic method that does not need particular annotations of the images in order to segment images according to the described classes. This process entails reusing the same visual model to predict the medical conditions within the images and then applying Grad-CAM [48] to generate attention maps of the images. These attention maps are then thresholded to get the segmentation regions. Further, the performers refine the outcomes of the segmentation; they remove the black pixels because the tissues' information is not contained therein.

3.1 Feature Engineering

As a result of the dependence of feature extraction on the decoder that generates the reports, I initially established a visual model. In this model, the visual features are obtained from the image and the tags are predicted from the image to get matching semantic embeddings from a medical embedding dictionary trained beforehand.

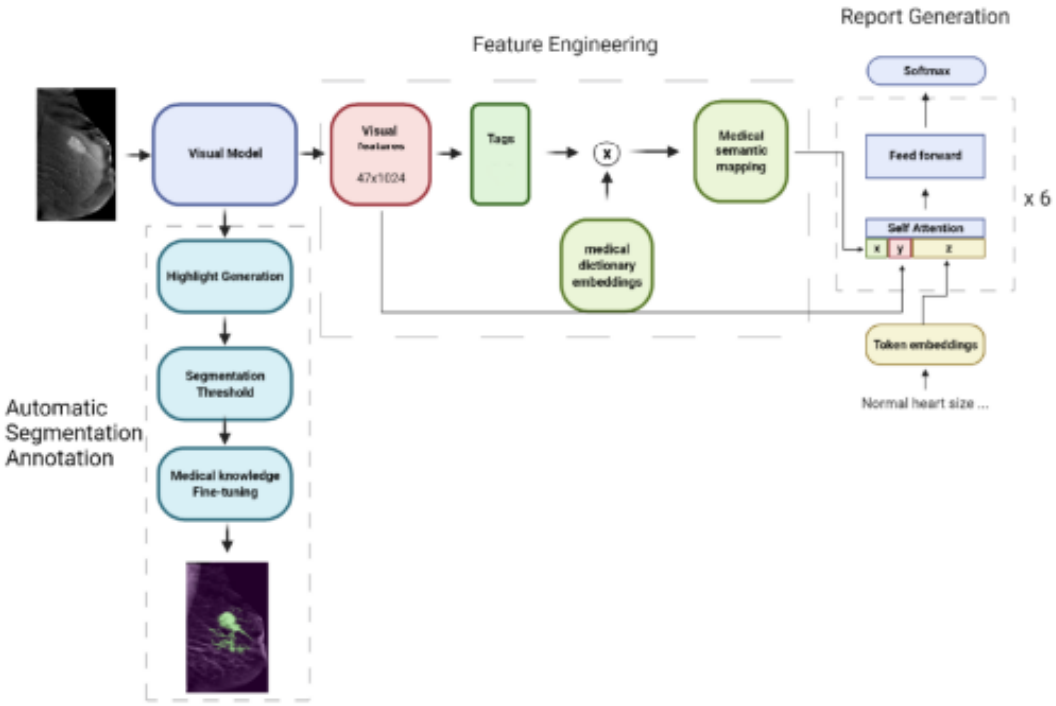

Figure 3.1: The comprehensive design of the CDGPT3.5 model along with the automated segmentation pipeline.

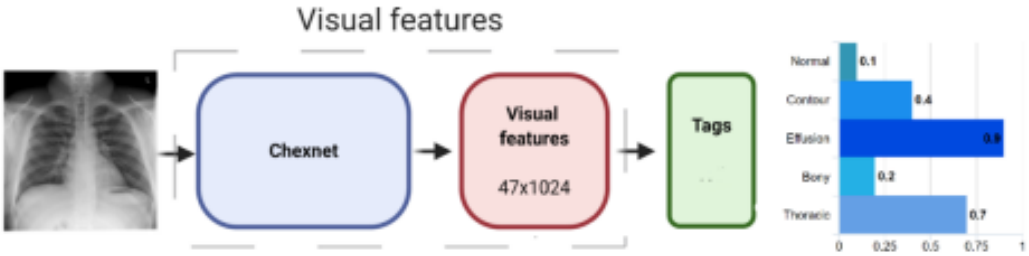

Figure 3.2: Suggested visual framework for extracting visual features and predicting tags.

3.1.1 Visual Model Development

The visual model is used to process the image to get visual features and tag predictions. The implementation I base this model on is ChexNet [31] running on the DenseNet201 [27] architecture pre-trained on the ChestX-ray14 dataset [33] for detecting and identifying 14 distinct illnesses or abnormalities. This base model delivered good visual features, but turned out I didn't have enough semantic information, having only 14 tags. Therefore, by first replacing the final layer with a new layer that classifies the most common tags as found in this dataset, and then fine-tuning that model to this dataset (Figure 3.2), I achieved this goal.
Table 3.1 shows the model architecture that consists of successive dense blocks and transition layers. Convolution and pooling additions begin this, followed by dense blocks and transition layers. Each of the dense blocks has two convolution operations with 1x1 and 3x3 kernel sizes repeated in equal amounts depending on the block

Table 3.1: Modified architecture of the Chexnet model.

Layers	**Size**	**Operates**
Input image	226 × 226	-
Pooling	58 × 58	3 × 3 max pool, stride 2
Convolution	114 × 114	7 × 7 conv, stride 2
Dense Block 1	58 × 58	1 × 1 conv 3 × 3 conv × 5
Transition Layer 1	29 × 29	1 × 1 conv2 × 2 avg pool, stride 2
Dense Block 2	29 × 29	1 × 1 conv 3 × 3 conv × 10
Transition Layer 2	15 × 15	1 × 1 conv2 × 2 avg pool, stride 2
Dense Block 3	15 × 15	1 × 1 conv 3 × 3 conv × 22
Transition Layer 3	8 × 8	1 × 1 conv2 × 2 avg pool, stride 2
Dense Block 4	8 × 8	1 × 1 conv 3 × 3 conv × 14
Pooling	1 × 1	8 × 8 global avg pool
Classification Layer	-	Fully connected + sigmoid

configuration. It consists of a 1×1 convolution, followed by a 7x7 average pooling layer with a stride of 2, half reducing the channels. A global average pooling layer flatters the features out to a 1x1 dimension, and then the final fully connected layer with a sigmoid activation layer to predict the tag probabilities.

3.1.2 Preprocessing

Since the ChexNet model was trained on images having the size of 224×224, I sized these input images to the same dimensions. Interpolation and anti-aliasing techniques were used to resize, and normalization in which the mean was subtracted from each pixel and divided with the standard deviation was performed. It makes pixel data uniform so that convergence of the model training is fast.

3.1.3 Image Augmentation

I applied random batch augmentations followed by feeding the images into the model to introduce the variability and prevent them from overlearning the data itself. They allow for increased diversity at the expense of extra processing time, and hence these augmentations do not interfere with the automated action planning in Poncho's perception. The following augmentations were applied randomly:

- Horizontal Flip: There is a 50% chance that each image is flipped horizontally.
- Cropping: 10–15% random cropping on each side.
- Perspective Transform: Random four-point perspective transformations.
- Rotation: Up to 90 degrees of the image was rotated in either direction.
- Translation: The amount of shift, along the X or Y axis, was up to 20%.
- Shear: The transformations were applied shears.
- Scaling: Between 0.8 and 1.5, vertical or horizontal scaling.

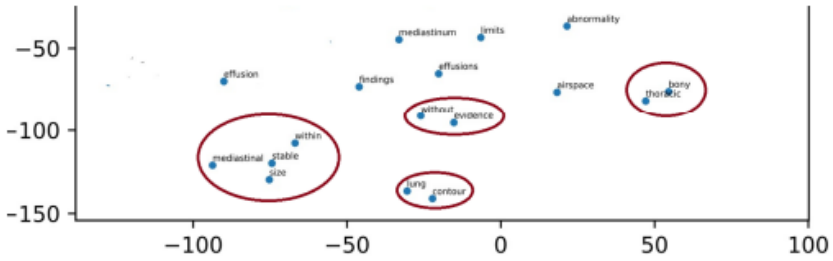

Figure 3.3: An example of the words in the reports and the distances between their embeddings after being reduced to two dimensions using t-SNE.

- Contrast Adjustment: The delta variation was adjusted with some sigmoid or gamma contrast.

3.1.4 Model Training

Binary cross entropy loss was used with the training of the visual model to perform binary-label classification. The loss for a batch is computed in equation 3.1, where $L(b)$ is the loss for batch b, T is the number of tags, and N is the batch size:

$$L(b) = -\sum_{t=1}^{T}\sum_{i=1}^{N} y_{it} \cdot \log \hat{y}_{it} + (1 - y_{it}) \cdot \log(1 - \hat{y}_{it}) \qquad \text{(Eq 3.1)}$$

I used the Adam optimizer [14] and mini-batch gradient descent based on batches of 32 images. I initialized the learning rate as $1e^{-3}$ and decayed it to $1e^{-7}$ to 0.1 if the validation loss plateaus for four epochs. Dropout with a drop probability of 0.4 was used to combat overfitting before predicting the tag. That prevents neurons from synchronizing to optimize their weights and thereby increases generalization. To address the class imbalance, class weights were assigned according to the following equation, where CW represents the class weight, c_i the class, P_{c_i} the positive instances, and N the total samples:

$$CW(c_i) = \begin{cases} \frac{P_{c_i}}{N}, & \text{if } c_i = 0 \\ \frac{N - P_{c_i}}{N}, & \text{if } c_i = 1 \end{cases} \qquad \text{(Eq 3.2)}$$

The output of the model includes predicted tags and a feature matrix of size 53×1024, which is generated by the final layer before global pooling.

3.1.5 Medical Embedding Dictionary

The confidence scores for each tag are combined by the visual model with a value between 0 and 1. Pre-trained word2vec embeddings from the MEDLINE/PubMed corpus [37] are used for these tags. Using a Skip-Gram model and a window size of 5, with 28 M articles trained to embeddings, I get a vocabulary of 2.6 M words.

3.1.6 Medical Semantic Mapping Process

The weighted semantic embeddings are produced by multiplying each tag's output from the visual model by its corresponding pre-trained word2vec embedding. The

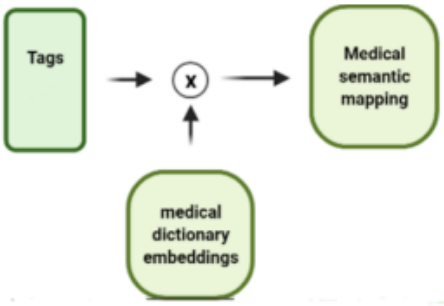

Figure 3.4: The process of generating the medical semantic mapping.

underlying embeddings emphasize which are most important in the image for the decoder, setting tags with low confidence scores aside. The embeddings for multi-word tags are averaged. Figure 3.4 depicts this process that guarantees the decoder gets meaningful semantic input, based on visual and textual input.

3.2 Automated Report Generation Process

In this section, I describe the models used for the automatic generation of medical reports, namely CDGPT3.5 and VSGRU; I show how I implemented these transformers and recurrent decoders to compare the performance of the two types.

3.2.1 CDGPT3.5 Model

At this phase, I utilize a pre-trained transformer-based model, depending on visual and semantic information, which generates medical reports. As illustrated in Figure 3.5, the architecture comprises three core components: the transformer-based decoder, the aggregation of semantic features, and the visual model. The image itself has both its visual features and predicted tags encoded by the visual model as an encoder. Tags are multiplied with their embedding slots computed using pre-trained embeddings and the confidence scores to get semantic features. A pre-trained distilGPT3.5 model [19] with a compact architecture of six layers 768 hidden units, 12 attention heads, and 85M parameters is used to decode the process. By being pre-trained on the OpenWebTextCorpus, a reconstruction of the original WebText dataset used to train GPT3.5 [42], this model runs twice as fast as the standard GPT3.5 model. However, the model still has the output tokens (50868) as GPT3.5 to handle medical contents well.

Structure

Table 3.2: Model architecture table.

Layer	Dimensions	Operation
Input Embeddings	50868 x 750	Embedding matrix
Position Encoding	1000 x 750	Positional embedding
Decoder Layer	750 units	Self-Attention and Feed-Forward, repeated 7x
Dense layer	50868	Fully connected layer for word prediction

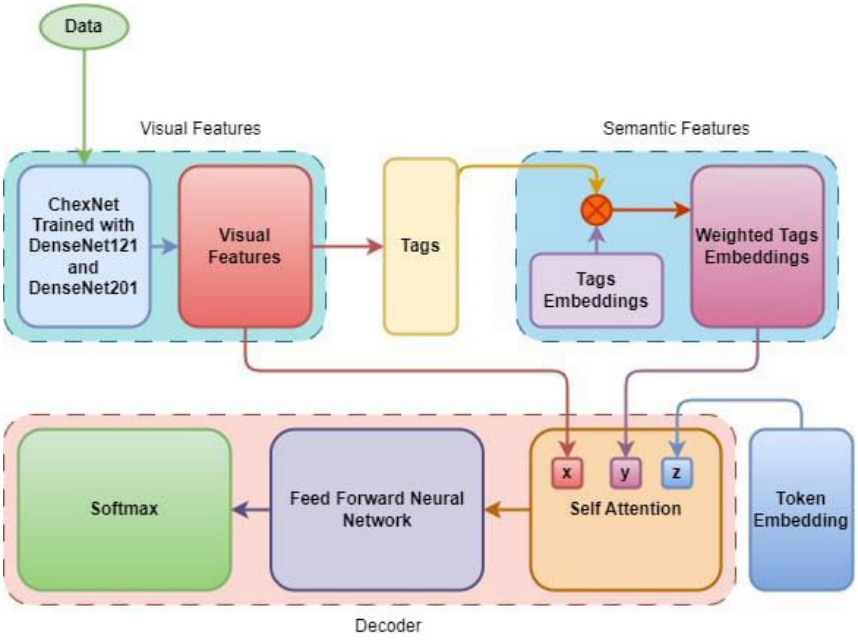

Figure 3.5: Overview of the CDGPT3.5 architecture workflow.

Table 3.2 shows that the CDGPT3.5 model has four major components. The pre-trained embeddings for all piecewise tokens from the GPT3.5 training process are stored in token embeddings. The 50257 x 768 embedding matrix for this case reflects the (50257 tokens, 768 dimensional) embeddings using (from 0 to 1) this embedding matrix. As transformer models process tokens independently, I add a positional encoding layer to the model to indicate the possible location of each token. What that means for the positional encoding matrix is that they go from 0 to 1024, or 1024 x 768. The decoder block is composed of self attention mechanism and a path to a fully connected layer, in which the attention mechanism assigns weights to tokens in a sequence and identifies the most relevant ones. This normally involves three weights matrices each combining with other matrixes to make a 768x768x3 matrix. But in CDGPT3.5 I drive separate keys and values for visual and semantic features, of 1024x768 and 402x768 dimensions respectively so the model can combine token embeddings with visual or semantic features. Six times, a decoder block is run in parallel and the final linear layer with softmax activation predicts the next token of the sequence.

Preprocessing

The image is preprocessed (Section 3.1.2) as the same for the visual model before training CDGPT3.5. However, the image augmentation was only pruning, translation, and contrast adjustment. For the text in the reports, I preprocessed all characters as lowercase case and removed punctuation and special characters. The tokenizer used was the default tokenizer for GPT3.5, given that its default tokenizer relies on byte pair encoding (BPE) [23]. First, BPE processes two-byte characters as tokens, and then gradually concatenates these tokens together dividing words into smaller units based on frequency patterns. Consider that if the letters "e" and "r" are sometimes seen together, the token "er" is included in the vocabulary. It goes on until the vocab size you want, which is 500387 for GPT3.5.

Conditioning Mechanism

As shown in Eq 3.4, self-attention is computed in the default transformer architecture with query keys and values. The Eq 3.3 defines the softmax function used to normalize attention scores. These equations are in which input token embeddings are represented as Z and the weight matrices W_q, W_k, W_v, represent queries and keys, and values, respectively.

$$\text{softmax}(x_i) = \frac{\exp(x_i)}{\sum_j \exp(x_j)} \tag{Eq 3.3}$$

$$S_A(Z) = \text{softmax}\left(\left(ZW_q\right)\left(ZW_k\right)^\top\right)\left(ZW_v\right) \tag{Eq 3.4}$$

The conditioning method in [47] introduced new parameters for keys and values are introduced initialized random solution for the mappings from encoder outputs to the attention space of the decoder. I expanded this approach to merge semantic and visual features into the model's attention mechanism λ, written as in Eq 5.5. This mechanism is realized via CSA (Conditioned Self Attention) here where Z are input token embeddings, X are semantic features, and Y are visual features. The newer semantic and visual information in the added keys (U_k, H_k) and values (U_v, H_v) are considered.

$$\text{CSA}(X, Y, Z) = \text{softmax}\left(ZW_q\right) XU_k Y H_k Z W_k + T + XU_v Y H_v Z W_v \tag{Eq 3.5}$$

The model can hold the entire knowledge learned during pre-training with this conditioning method. The original pre-trained parameters are unchanged since it only adds new weights to focus on how visual and semantic features. This method decreases the interference with the pre-trained weights of the model, which then enables fast learning and quickens the training process as a whole.

3.2.2 VSGRU

In order to compare the training efficiency and performance of the transformer model, I created a recurrent model with visual and semantic attention mechanisms, VSGRU (Visual-Semantic Gated Recurrent Units). Figure 3.6 shows the architecture. The visual feature of the visual model and the predicted tags to be predicted are initialized at first. They go on to extract visual features after the global average pooling layer, which reduces dimensionality from 47 $\times$ 1024 to 1024 dimensional vector. I experimentally determine a rate at which the tag predictions are labeled with 0s and 1s, giving a binary vector reflecting the existence or not of a tag. These binary tags are multiplied with the pre-trained tag embeddings to generate binary tag embeddings of size tags $\times$ 400, which only contain the embeddings of the tags that meet the 0.1 threshold. This 400-dimensional vector represents the semantics of the image, with binary tag embeddings averaged to represent the image's semantics. Since Bahdanau's attention mechanism is computationally expensive, the number of concatenated parameters to the visual features can be reduced through the averaging of both the visual and semantic features. They learn these visual and semantic features and combine them into a 1424-dimensional vector, which they pass through a GRU cell with Bahdanau's attention mechanism [13] to start generating their medical report.

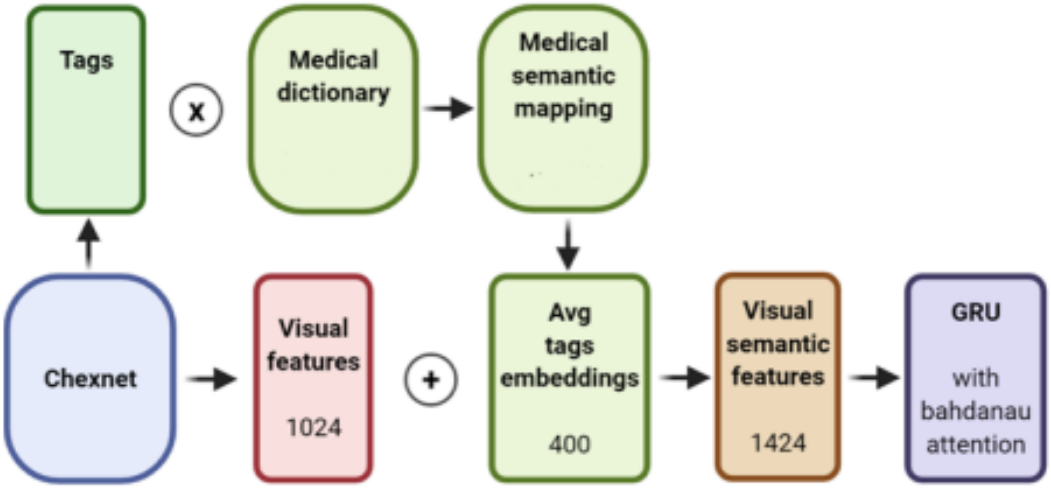

Figure 3.6: An illustration depicting the VSGRU model.

Architecture

Table 3.3: Parameters of the VSGRU Model

Layer	Dimensions	Description
Embedding	1600 × 420	Matrix of embeddings
Attention	1100	Three fully connected layers of size 1100
Gated Recurrent Unit (GRU)	1100	GRU cell with three gates, each of size 1100
Feedforward	1100	Fully connected layer of size 1100
Predictor	1600	Fully connected layer size 1600 for word

The VSGRU model comprises four main components, as summarized in Table 3.3: This includes the embeddings, the attention layer, the GRU cell, and the classifier. The pre-trained embeddings are provided by the embeddings layer as input tokens are converted to their corresponding embeddings from the embeddings layer as they are present holder of the pre-trained embeddings of the 1500 most frequent words in the dataset vocabulary. The attention mechanism is comprised of three fully connected layers, two of which are the alignment layer and the attention weights, and one is the context vector. Eq 3.5 describes the attention mechanism, and S is an alignment score, F is the visual-semantic feature vector, H is a hidden layer, Aw is the attention weights, C_v stands for the context vector, and W_1, W_2, V represents the weight matrices along the way of attention mechanism.

$$S = \tanh(W_1 \cdot F + W_2 \cdot H)$$

$$A_w = \text{softmax}(V \cdot S)$$

$$C_v = A_w \cdot F \qquad \text{(Eq 3.5)}$$

An alignment score is calculated to measure the closeness of the current word to the visual semantic features as well as other neighboring tokens, and a softmax

function is used to give the attention weights. The sequence weight for each token is how much that token is important in the sequence. Then I concatenate the input word embeddings with the context vector as the input to the GRU cell. The GRU (Gated Recurrent Unit) cell utilizes two gates: It is either the update gate or the reset gate. The update gate decides which information to keep and which to lose; the reset gate decides whether previous information should be forgotten. The GRU cell operations on the right-hand side of Eq 3.6 can be summed up by z, r, $\hat{h}$ = intermediate memory, and h = final output.

$$z = \sigma(W_z \cdot X_t + U_z \cdot h_{t-1} + b_z)$$

$$r = \sigma(W_r \cdot X_t + U_r \cdot h_{t-1} + b_r)$$

$$\hat{h} = \tanh(W_h \cdot X_t + r \cdot U_h \cdot h_{t-1} + b_h)$$

$$h = z \cdot h_{t-1} + (1 - z) \cdot \hat{h} \qquad \text{(Eq 3.6)}$$

The reset and update gates help GRU cells forget irrelevant past information while being efficient in capturing long-term dependencies.

Preprocessing

When training the VSGRU, the preprocessed and augmented images are applied in the same way as the preprocessed and augmented images described for the visual model in Sections 3.1.2 and 3.1.3. Same as it, are also preprocessed the corresponding medical reports by removing punctuation, and special characters and converting the text into lowercase. Additionally, for extremely rare words, they were labeled as an 'unknown' token; and a vocabulary of the 1500 top most frequent words was chosen. It truncated the reports to a maximum length of 170 words. Tokenization means each word is assigned with a unique identifier. I also included special tokens to mark the beginning of the sequence, end of the sequence, unknown words, and padding for reports of less than 170 words.

Conditioning Mechanism

The weights of the visual model were fixed for training the VSGRU model, with only the decoder and attention layers updated. We then combined the visual and semantic features using a simple fusion layer, and dropped out this combined feature using a dropout layer with a drop probability of 0.35 to introduce variability and reduce overfitting. A GRU cell with 1024 units was used in the decoder and Bahdanau's attention was used over both visual and semantic features.
For the embedding layer of the GRU cell, we used pre trained semantic embeddings with size 1400 $\times$ 400, and kept the weights trainable during training. An output layer was additionally used with an experimental chosen dropout probability of 0.35 to prevent overfitting. In order to keep the signal values balanced through the network, the GRU cell's weights were initialized with Xavier initialization [100]. In the GRU cell, we had an activation of ReLU function and always start the input

sequence with a start token and set the hidden state to zeros at the start of each batch.
The loss function for training was categorical cross-entropy, as shown in Eq. 5.8, where W represents the word count in sentence i and N is the batch size. The loss calculation excludes padding tokens.

$$L(b) = -\sum_{i=1}^{N}\sum_{w=1}^{W} y_{iw} \cdot \log \hat{y}_{iw} \tag{3.1}$$

For optimization, the Adam optimizer [49] was used with a fixed learning rate of 1×10^{-3}. The batch size was set to 14 to balance computational efficiency with memory usage.

3.3 Automated Segmentation and Annotation Method

In this section, I describe how automated segmentation works, the combination of visual model, and specific training parameters, as well as how domain knowledge in medicine was integrated into the segmentation process.

3.3.1 Model Structure

An EfficientNetB0 originally trained on ImageNet serves as the foundational model for segmentation annotation. The underlying architecture consists of 6 inverted residual blocks with different kernel sizes and different numbers of channels. We utilize the swish activation function [101] and these blocks are made up of squeeze-and-excitation modules. Specifically, we define swish in Eq. 3.2 which is an improvement to a ReLU, allowing it to retain some negative values rather than completely discarding them.The output of squeeze-and-excitation modules has size 1×1 channels, in which each channel is assigned adaptive weights by the modules.rameters, and how domain knowledge in medicine was incorporated into the segmentation process.

$$\text{Swish}(x) = x \cdot \text{Sigmoid}(x) \tag{3.2}$$

Table 3.4: EfficientNetB0 Architecture Parameters

Layers	**Output Size**	**Channels**
Input image	220 × 220	3
Convolution 3×3	110 × 110	30
MBConv1, k3×3	110 × 110	15
MBConv6, k3×3	55 × 55	22
MBConv6, k5×5	27 × 27	42
MBConv6, k3×3	13 × 13	75
MBConv6, k5×5	6 × 6	115
MBConv6, k5×5	6 × 6	185
MBConv6, k3×3	1 × 1	310
Fully Connected	-	1260

3.3.2 Training Details

This model aims to generate visual highlights within images and was trained using the entire dataset. The loss function, as presented in Eq. 3.3, was categorical cross-entropy, where N denotes the batch size. We trained the model using mini-batch gradient descent with a batch size of 30, utilizing the Adam optimizer [49]. During training, all model parameters were optimized. The initial learning rate was set to 1×10^{-3}, decaying to 1×10^{-7}. The learning rate was reduced by a factor of 0.1 and capped at 1×10^{-7} if the validation loss did not improve over the next four consecutive epochs. Additionally, a dropout layer with a high drop probability of 0.8 was added before the fully connected layer to mitigate overfitting, as all images were used for training.

$$L(b) = -\sum_{i=1}^{N} y_i \cdot \log \hat{y}_i \tag{3.3}$$

3.3.3 Generation Technique of Highlight

Grad-CAM [93] was used to identify the regions most relevant to each of the predicted tags. By design, fully connected layers discard spatial information, but convolutional layers keep spatial content in their final layers, which is often a better trade off than having both high level semantic content and spatial accuracy in fully connected layers alone. And in these layers: neurons will detect information that is unique to each class. By assigning priority scores to the neurons based on the gradient information approaching the last convolutional layer, this model is able to trace back from output neurons and discover key neurons. In particular, backpropagation from output nodes to previous layers is performed, recording the nodes with the greatest weights, which are later tied to input image and show which image regions are influential for each class. We then predict tags for each image, and retrace from the output node of the correct label back to the image and create highlights for those portions. Also, corners of the image were removed, as these are often the points of image highlights which are more commonly found in the normal pectoral muscle regions instead of clinical important regions.

Chapter 4

Dataset

Digital Mammography (DM) is the primary imaging technique for the early detection of breast cancer. However, its sensitivity is reduced in patients with dense breast tissue, albeit not to the extent that impairs the overall functionality of the technique [19]. The approach is contrast-enhanced spectral mammography (CESM) which accepted FDA approval in 2011 as a supplementary tool to DM and ultrasound for the identification and analysis of hidden or ambiguous lesions [7]. The method involves the acquisition of dual-energy images such as low-energy and high-energy images. Low energy images have been shown to closely resemble standard DM images and to be similarly diagnostic accurate [15]. Yet after image acquisition, high-energy images are not interpretable, and consequently, image processing techniques are used to recombine and subtract the images, thus increasing the background breast tissue visibility. Figure 4.1 shows the subtracted images used for diagnostic assessment, which have areas of contrast enhancement in a background of muted breast tissue [24]. Then lesion characteristics can be assessed in terms of density, morphology, and enhancement features. Nevertheless, lesion appearance remains highly variable such that benign and malignant lesions cannot be distinguished without a radiologist's review [4].

In the early 2000s, computer-aided detection (CAD) systems were developed by computer scientists to help radiologists detect abnormalities or diseases in mammographic images. However, their use has become complicated by an increase in the rate of false positives, which can dilute the contribution of radiologists to the reading [10]. Artificial intelligence (AI) integration in radiology is just beginning. However, algorithms that deal with pixel data can detect patterns in images that even experienced radiologists couldn't have seen [25]. It has so much promise for so many applications that some of these systems can be used to help radiologists in the automatic detection of lesions or can be as accurate as possible to enhance the diagnostic accuracy of radiologists. However, in order to train new deep-learning models or to fine-tune pre-existing ones there is a critical need of large, and well-annotated datasets.

Recently a few public mammography datasets have been made available, for example, the Digital Database for Screening Mammography (DDSM) [3], Image Retrieval in Medical Applications (IRMA) project [5], MIAS database [1] and Curated Breast Imaging Subset of DDSM (CBIS-DDSM) [29]. The images in these datasets only consist of DM images without CESM image representation. To fill this gap, I created a CESM-specific dataset, which was developed in partnership with Egypt's

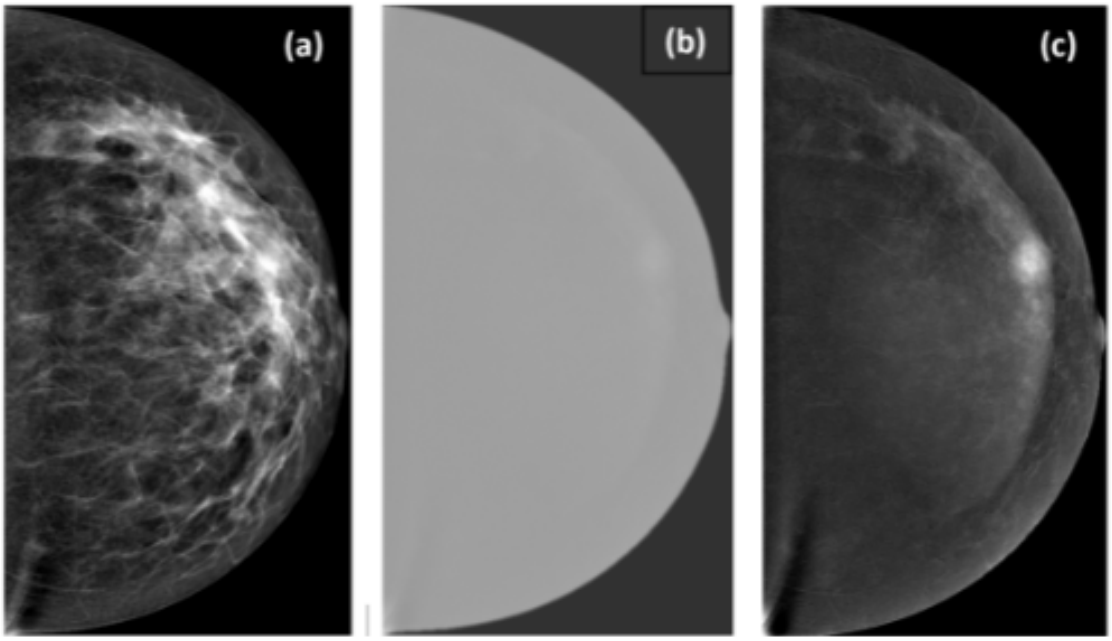

Figure 4.1: (a) Low-energy, (b) High-enrgy, and (c) Subtracted image.

National Institute of Cancer, and which contains low energy images, subtracted CESM images, annotations of abnormality segmentation, proven medical reports, and pathologic diagnoses for all cases. The purpose of this new dataset is to facilitate ongoing progress with deep learning applications for mammography.

4.1 Data Aquisition

The images keep a high resolution, being about 2300 × 1300 pixels. Roughly 326 women between 18 and 90 years of age provided approval from the institutional review board and given consent by the patients. Two distinct imaging systems were utilized for the image acquisition: It is with these GE Healthcare Senographe DS and Hologic Selenia Dimensions Mammography Systems. The dataset includes 1003 subtracted CESM images and 1003 low-energy images as well as CC and MLO views. Figure 4.2 shows samples of the low-energy images together with the corresponding CESM images. Standard DM equipment is supplemented with software that is able to perform dual energy image acquisition. Two minutes after intravenous injection of a non-ionic low osmolar iodinated contrast agent (dose 1.5 mL/kg), CC and MLO views are obtained. Each view consists of two exposures: That is one at low energy (peak kilo-voltage values from 26 to 31 kVp) and the other at high energy (45 to 49 kVp). This takes between 5 and 6 minutes. It consists of 757, 587, and 662 normal, benign, and malignant cases respectively. Furthermore, the data include 445 cases of invasive ductal carcinoma, 42 cases of invasive lobular carcinoma, 28 cases of mixed invasive ductal carcinoma, 17 cases of pure ductal Carcinoma in situ, and 40 cases of inflammatory breast cancer. The distribution of cases spans various age groups: Patients under 40 (58), 40 to 49 (100), 50 to 59 (95), 60 to 69 (59), and over 70 (14).

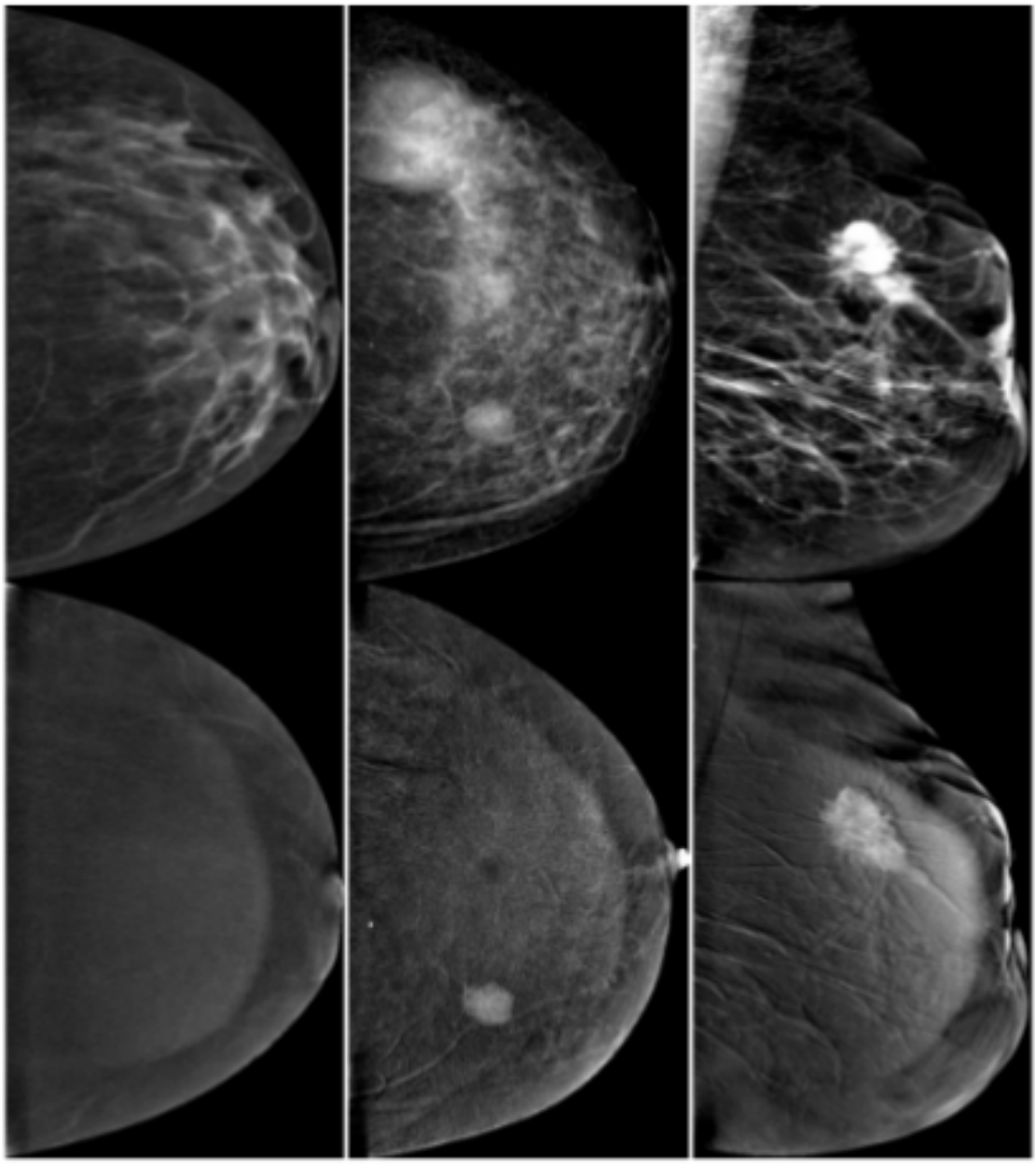

Figure 4.2: Sample of low energy and subtracted CESM images from the dataset.

Table 4.1: Overview of the CDD-CESM dataset

Characteristic	**Details**
Duration	2019-2021
Source	National Cancer Institute
Number of Participants	320
Total Images	2012
Normal Images	762 (37.8%)
Benign Images	580 (28.8%)
Malignant Images	670 (33.4%)
Age Distribution	
Under 40	55 (17.2%)
40-49	98 (30.6%)
50-59	97 (30.3%)
60-69	60 (18.8%)
70 and Above	10 (3.1%)
Cancer Types	
Invasive Ductal Carcinoma	440 (67.2%)
Invasive Lobular Carcinoma	40 (6.3%)
Mixed Invasive Ductal Carcinoma	30 (4.6%)
Pure Ductal Carcinoma In Situ	15 (2.3%)
Inflammatory Breast Cancer	35 (5.4%)
Other Types	85 (13.2%)

4.2 Data Preparation

I annotate the 2006 images in the dataset with painstaking precision per the American College of Radiology Breast Imaging Reporting and Data System (ACR BI-RADS) 2013 lexicon, which standardizes the descriptors used [12]. As tabulated in Table 4.2, the annotations include breast composition, mass shape, margin and density, architectural distortion, asymmetry, types and location of calcifications, enhancement patterns for the mass, non-mass enhancement patterns, and overall BIRADS assessment (1 to 6). Furthermore, both follow-up and pathologic results are included, as pathological findings are the gold standard for suspicious or malignant lesions, while follow-up results are the gold standard for benign lesions. Each case also includes comprehensive medical reports prepared by a team of radiologists, as well as manual segmentation annotations for abnormalities in each image. Omitted have been the annotations not related to lesion identification and classification, such as patient name, identification number, study date, and image series. A single comma-separated values (CSV) file for each image and each respective annotation is compiled. The dataset provides distinct reports for the CESM and DM images within it. The findings of each report are reported separately for each breast side, using descriptors defined in the ACR BIRADS 2013 lexicon and corresponding to each case in the annotated BIRADS category. To avoid violating patient confidentiality, all personally identifiable information has been removed.

Table 4.2: Details of Annotations for the CDD-CESM Dataset

Annotation	**Description**	**Method**	**Format**
Patient's Age	Age of the patient at the time of examination.	Derived from the date of birth.	Numerical
Breast Side	Indication of the breast (right or left).	Annotated by hand.	Categorical
Breast Composition	ACR classification of breast density indicating the proportion of fibroglandular tissue to fat.	Evaluated by two radiologists independently.	Categorical: a: Mostly fatty b: Scattered fibroglandular tissue c: Heterogeneously dense d: Extremely dense
Radiological Lexicon	Standard descriptors for breast imaging findings assessment.	Independent evaluation by two radiologists.	Includes mass shape, margin, density, architectural distortion, asymmetries, calcification type and distribution, mass enhancement patterns, and non-mass enhancement characteristics.
Overall BIRADS	Final categorization for breast imaging findings.	Blinded evaluation by two radiologists.	BIRADS scale from 1 to 6
Image View Type	Standard imaging perspectives in breast mammography.	Manually annotated.	Categorical: MLO: Most breast tissues depicted CC: Medial and external lateral portions revealed
Tags	Labels assigned as follows: - Standardized ACR BIRADS 2013 descriptors - Likely diagnosis - Classification types	Manually assigned and documented by a radiologist.	Set of approximately 140 tags
Machine Label	Identification of the mammography equipment used.	Annotated manually.	Machine number: 1 or 2

Continued on next page

Annotation	Description	Method	Format
Pathology Results / Follow-Up	Classification of cases into normal, benign, and malignant categories.	Manually annotated.	Categorical: - Normal - Benign - Malignant

It is crucial to release comprehensive medical reports because radiology reports often fall short of details as in larger datasets. DICOM images were extracted using the RadiAnt DICOM viewer application and then saved in JPEG format. Halfway through that, I performed manual cropping removing unnecessary borders from each image. Additionally, images are given systematic names in the form of patient number breast side image type image view; e.g. "P1 L CM MLO".

4.3 Medical Image Fine-Tuning

Any pixel would receive a score between 0 and 1 in the Grad-CAM generated highlights, indicating the pixel's importance in the final prediction. A threshold is applied to these highlighted pixels to make precise segmentation for each image as to accurately identify exactly what areas are included in the segment. In particular, all images collectively, rather than individually, provide the top intensities from which the top 25% are selected to make up the segment.
To further identify, an additional threshold is applied to suppress the low-intensity pixels in the segment. This step is worthwhile since tumors appear in medical imaging in tissue regions with high luminosity, which is enlarged by this filtering of abnormal tissue. As a result, the pixel intensities within each segment belonging to the top 15% of the pixel intensities of which there are in the dataset are kept, again based on all images in the dataset. Figure 4.3 visualizes the entire process and results of our final output in comparison to the ground truth segmentation.

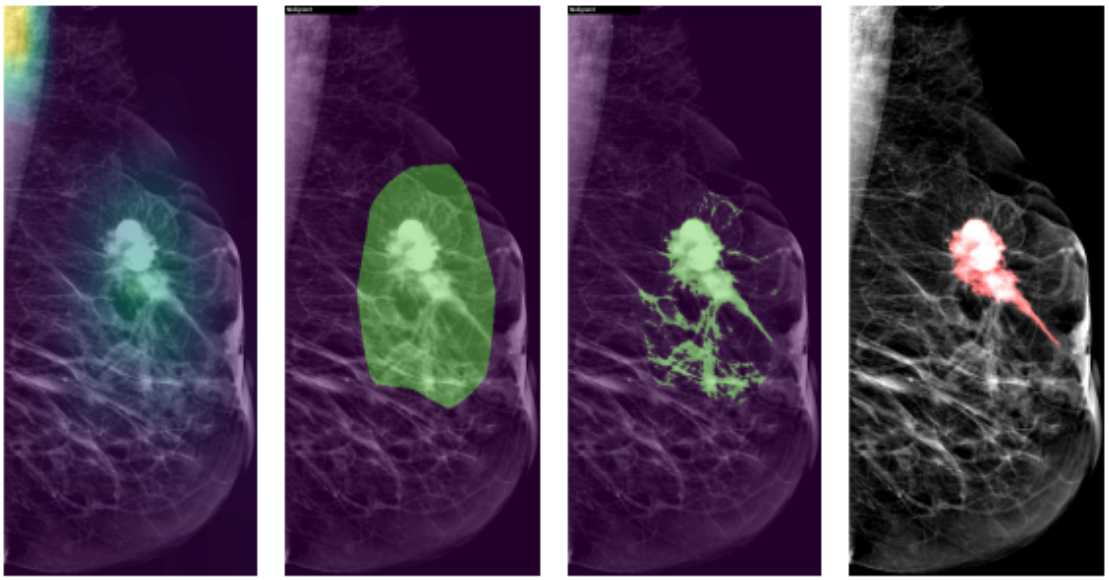

Figure 4.3: (a) Example showing the deep learning-based Grad-CAM highlights, (b) Segmentation generated by applying a threshold to these highlights, (c) Final segmentation after applying an intensity threshold on the brightest pixels, and (d) Manually annotated segmentation for comparison.

Chapter 5

Analysis of Experimental Results and Discussion

For this study, the experimental setup was run with the aid of a standard machine learning workstation, whose hardware had been detailed as an NVIDIA RTX 3080 Ti GPU, a 4-core Intel i7 CPU at 2.8GHz, 16 GB RAM, and 512 GB SSD storage. On the software side, the experiments were conducted by running on an Ubuntu 20.04 Linux operating system and Python version 3.8 as well as other libraries used for model development and evaluation. Specifically, deep learning model building was done using TensorFlow 2.4.0 and Keras 2.4.3, language modeling with Transformers 3.0.0 [31], image processing using OpenCV 4.2.0, data augmentation using ImgAug 0.4.0, natural language evaluation using NLG-Eval [11], text processing using NLTK [36], and implementing EfficientNet 1.1.0 [8] for the EfficientNet The configuration of this was such that the model can be trained and tested in a robust and efficient manner.

5.1 Dataset Preprocessing

In this section we describe how the IU-Xray dataset and newly created CDD-CESM dataset were prepared.

5.1.1 IU-Xray Dataset

Only a few public datasets exist in medical imaging with associated radiology reports. There are few that provide professionally annotated reports for chest X-ray images, such as the Indiana University chest X-ray dataset (IU-Xray) [31].
The reports are structured into the following sections:

- Impression: A brief summary or title.
- Findings: Observations conducted in a detailed narrative form.
- Manual Tags: The reports coded tags.

There are two types of tags available in this dataset:

Impression: Negative chest x-XXXX.

Findings: Cardiac and mediastinal contours are within normal limits. The lungs are clear. Bony structures are intact.

Manual Tags: Normal

Figure 5.1: Example chest X-ray image and report from the IU-Xray dataset. The dataset includes 7,200 images (frontal and lateral views), covering approximately 3,800 patients along with their respective reports.

- MTI Tags: Now extracted through the MTI indexer based on Medical Subject Headings (or MeSH) to easily index them, a practice first adopted by the Indexing Initiative in 2002.
- Manual Tags: Trained medical coders add on MeSH and RadLex lexicons. For this work, I set the report target to be the concatenation of Impressions and Findings as in [36]. The rest was excluded, and I picked 102 tags that occurred more than 20 times as tags with sufficient frequency. Compressed confidential data appeared as "XXXX." An experiment was built from selecting 520 randomly selected images from the dataset input.

5.1.2 CDD-CESM Dataset

Mammography and contrast-enhanced mammography-based images comprise 2000 images in the CDD-CESM dataset. The images are categorized into three primary classes:

- Normal
- Benign
- Malignant

Tags are given to each image according to ACR's BI-RADS lexicon (2013). I kept 48 tags out of 140 tags: they were used enough to keep the model trained consistently. I concatenated the title of each report to its Findings (contents of each report) and used this as the report target. A randomly selected 310 images for evaluation were used as the test set.

5.2 Metrics of Evaluation

I discuss the evaluation metrics I used to rate the performance of the visual model, the report generation model, and the segmentation model in this section.

5.2.1 Classification Metrics

As evaluated using the Receiver Operator Characteristic (ROC-AUC) curve [11], the visual model was trained to learn to classify multiple tags per image. This curve is then used to discriminate between signal and noise as a function of threshold via comparison of true positive rate vs false positive rate. There is a summary measure for this curve provided by the Area Under the Curve, which is the Area Under the Curve (AUC) based on the model's ability to distinguish classes. An AUC value of 1 signifies a model which can perfectly separate all positive from the negative class points, while 0 gives an AUC value for a model that improperly classifies all points. Since AUC is between 0.5 and 1, the model has a nice probability of telling which of the class points are positive or negative, since there are more true positives and true negatives than false positives and false negatives. For comparison, an AUC of less than 0.5 indicates that the classifier cannot differentiate between those positive and negative classes, and thus predicts data points as if it's being operated on a random or constant class. This means that the higher the AUC, the better that model will be able to distinguish between classifications.
There are several approaches to averaging the performance of different classifiers:

- Macro: Calculates the unweighted mean of each label's metrics, disregarding labelling imbalance.
- Weighted: Weight metrics by support, and averages metrics for each label while considering class imbalance.
- Micro: This method treats each entry in the label indicator matrix as a label then calculates the metrics at a global level.
- Samples: It averages the metrics for an instance and calculates metrics.

In this study I applied weighted average approach to handle class imbalance.

5.2.2 Metrices for Word-Overlapping

Image captioning models are often evaluated using word-overlap metrics. I had some reservations about their effectiveness in evaluating sensitive medical systems but included them to ensure a fair comparison with previous studies. The following metrics were applied:

- BLEU Score [13]: Simply, the Bilingual Evaluation Understudy (BLEU) score compares the output candidate text translation to one or more reference translations and results in a score of 1.0 being a perfect match and 0.0 being a perfect mismatch. It is language agnostic, easy to compute, and shows strong correlation with human judgments. It consists of n-gram matching between candidate and reference text: each token being unigram, each word pair bigram. With BLEU, we adjust it to limit the frequency a word can match with reference text, rewarding the candidate for correct matches versus throwing out a hill of high probability phrase paraphrases. While obtaining a BLEU score of perfect is hard because you need to exactly match the reference, differences in the number and quality of reference translations make it difficult to compare across datasets.

- ROUGE Score [23]: When used for summarization evaluation, the principal report of the Recall-Oriented Understudy for Gisting Evaluation (ROUGE) is based on recall. ROUGE measures the co-occurrence statistics using the longest common subsequence (LCS), weighted LCS and skip bigram and give an F score based on the harmonic mean of precision and recall. Equations for ROUGE-L, where candidate and reference sentences are represented as A and B with lengths m and n, respectively, are given as follows:

$$P = \frac{\text{LCS}(A, B)}{m} \tag{5.1}$$

$$R = \frac{\text{LCS}(A, B)}{n} \tag{5.2}$$

The weighted harmonic mean of precision and recall then yields the F-score:

$$F = \frac{(1 + \beta^2) \cdot RP}{R + \beta^2 \cdot P} \tag{5.3}$$

- METEOR Score [13]: A metric for Evaluation and Translation with Explicit Ordering (METEOR) using a weighted F score based on unigram mappings and a penalty function on incorrect word order, outperforms BLEU. In METEOR, the largest groups of mappings are identified between candidate and reference translations based on unigrams aligned via exact matches, Porter stemmed and Wordnet synonymy. The METEOR F-score is calculated as:

$$F = \frac{PR}{\alpha \cdot P + (1 - \alpha) \cdot R} \tag{5.4}$$

METEOR also includes a penalty function to account for word order in the candidate sentence:

$$\text{Penalty} = \lambda \left(\frac{C}{m}\right)^{\beta} \tag{5.5}$$

where C represents the number of matching chunks and m is the total number of matches.

- CIDEr Score [20]: Synonyms for that one are Consensus based Image Description Evaluation(CIDEr) score, which my experience is tailored for image captioning. CIDEr is a metric based upon cosine similarity between candidate and reference sentences, maximizing precision and recall at once. This process is represented by the CIDErn score:

$$\text{CIDEr}_n(c_i, S_i) = \frac{1}{m} \sum_j \frac{g_n(c_i) \cdot g_n(s_{ij})}{\|g_n(c_i)\| \cdot \|g_n(s_{ij})\|} \tag{5.6}$$

In the following equations, $g_n(c_i)$ represents a TF-IDF vector for all n-grams of length n. The CIDEr score combines the scores across different n-gram lengths as shown below:

$$\text{CIDEr}(c_i, S_i) = \sum_{n=1}^{N} w_n \cdot \text{CIDEr}_n(c_i, S_i) \tag{5.7}$$

These evaluation metrics offer a comprehensive framework for analyzing both the qualitative and quantitative outcomes of the models used in this study.

5.2.3 Metrics for Semantic Similarity

In order to evaluate semantic similarity, several word overlap metrics such as BLEU [13], METEOR [23], ROUGE-L [13] and CIDEr [20] were used. Yet, as explained in previous works [26], [44] these metrics may be inadequate measures of sensitive captioning systems, if only a single reference sentence is provided, as in the case of IU-Xray and CDD-CESM. For instance, radiology reports would express "Normal" and "No findings" to have the same meaning, yet score zero on word overlap metrics. To address this issue, additional semantic similarity approaches were adopted:

- SkipThought Cosine Similarity [43]: A neural network called Skip thought learns fixed sentence-length representations of natural language without the use of supervised or tagged input. It uses as its training signal the natural sentence sequence in a corpus and encodes any sentence in a fixed-length vector. The encoding makes it possible to carry out simple calculations to see whether two sentences mean the same thing or not. The Skip-thought model consists of three main components: A fixed-length representation is generated by an encoder network z from sentence x(i) at position embeddings and serves previously as a decoder network. z(i) to reconstruct the sentence It also includes a next decoder network that is fed by To generate the following sentence, use z(i). x(i+1). In general, both encoder and decoder networks are used with GRU or LSTM models. At its final output, Skipping Thought gives us a trained encoder that can generate fixed-length sentence representations that can be used in tasks like semantic similarity and sentiment classification.

- Embedding Average Cosine Similarity: At the word2vec [9] level, this metric computes similarity between vectors of all words in the reference sentence and averages them. We create the average cosine similarity between the embeddings of the two sentences, as we also do for the predicted sentence. Unlike Skip-thought, this approach is at a word level rather than looking for spatial or order information between words.

- Vector Extrema [46]: In the vector extrema method, for each sentence, each dimension of the word vector is filtered out and a maximum extreme value over the entire sentence is selected. A measure of semantic similarity is computed as cosine similarity between the vector extrema of the predicted and reference sentences.

- Greedy Matching [49]: This is the only embedding based method without sentence-level embedding. Instead, instead of having each word in one sentence greedily matched against the closest word in the other sentence by cosine similarity and then averaging the overall score across all words, each word

is greedily matched against the closest word in the other sentence by cosine similarity, and the combined score is averaged over all words. In particular, the asymmetry of this approach is addressed by averaging scores in each direction. Greedy matching can be used to identify essential phrases from the generated response that semantically match to the phrases that were present in ground truth.

5.2.4 Segmentation Metrics

To evaluate the automated segmentation approach, several quantitative metrics were used to compare generated segmentation against ground-truth data:

- Intersection over Union (IoU): One of the most commonly used segmentation evaluation metric (IoU) is defined as area of overlap between predicted mask and ground truth mask divided by the union area between them. For binary or multi class segmentation, mean IoU is calculated by averaging IoU calculated per class.
- F1 Score (Dice Coefficient): The F1 score, or Dice coefficient, is defined as twice the area of intersection between predicted and true classes divided by the total area of both classes.
- Overlap50: Radiologists at the National Cancer Institute introduced this metric. Each image is assigned a binary score of 0 or 1, depending if the automatic segmentation cover at least 50% of the ground truth pixels or not.

5.3 Visual Model Experiment

Generating complete textual reports or segmentation annotations is built upon the visual model. In addition, it can serve as an independent model to predict existing tags of a medical image. It further describes the training experiments that we conducted, and evaluates the model's performance.

5.3.1 Experiment

I then conducted several experiments running various model architectures and hyperparameters to help in developing the final visual model we use in CDGPT3.5 and VSGRU. Architectures such as CheXNet [31], DenseNet121, DenseNet201 [27] and EfficientNetB4 through EfficientNetB7 [17], were tried during these trials. We pre-trained each model on ImageNet except for CheXNet, which was pre-trained only on chest X-ray images of the ChestXray14 dataset [31] shown in Table 5.1. For image input size, we kept the size fixed at 224x224 throughout all models which were used the same size as the pre trained models. To lessen overfitting, I replaced the last layer with a tag-specific layer, and added a Dropout layer of drop probability 0 with elu act (set through the act hyperparameter of the Activations class). The new classification layer was thus initialized with weights randomized using the Xavier initializer.
I set the learning rate to 1e-3 initially, with the decay factor being 0.1 whenever the validation loss didn't change for three epochs. A minimum allowed learning rate of

Table 5.1: The results of the visual models experiments conducted. BCE stands for binary cross-entropy, and Nadam stands for Nesterov Adam.

Dataset	Visual Model	Loss Function	Optimizer	Train AUC	Test AUC
IU-Xray	Chexnet	BCE	Adam	86%	**77%**
	Chexnet	BCE	Nadam	84%	75%
	Chexnet	Focal Loss	Adam	85%	76%
	Densenet121	BCE	Adam	83%	75%
	Densenet201	BCE	Adam	87%	65%
	EfficientNetB7	BCE	Adam	85%	78%
CDD-CESM	Chexnet	BCE	Adam	89%	**82%**
	Densenet121	BCE	Adam	85%	78%
	Densenet201	BCE	Adam	**91%**	71%
	EfficientNetB7	BCE	Adam	90%	81%

1e-7 was chosen. The training was made to converge without overly stretching the training duration via learning rate decay. Equation 5.8 shows that I also experimented with both our binary cross entropy as well as focal loss. It is an enhanced form of cross entropy loss that strengthens the ability to focus more on difficult or misclassified cases.

$$\mathrm{FL}(p_t) = -\alpha_t(1-p_t)^{\gamma} \cdot \log(p_t) \tag{5.8}$$

I used Adam [2] and Nesterov Adam [39] as that combines Adam with the Nesterov accelerated gradient method [14] for optimization. In the experiments, batch size is 28. Furthermore, I also trained the model on IU-XRay and CDD-CESM dataset separately.

5.3.2 Discussion

In Table 5.1, we present the results of training several different visual models on these datasets using different optimizers and loss functions. I tested the DenseNet121 model, pre-trained on ImageNet, on the IU-XRay dataset, scoring 72% accuracy on the test set, compared to 79% for CheXNet, pre-trained on chest x-ray data. Though DenseNet201 scored 86% on the training set, the test set was unlucky, scoring only 63%, meaning it most likely overfit due to high parameter count and model complexity. That resulted in a big difference between training and testing. CheXNet is the only model to come close to the task with 76% on the test set, second only to the EfficientNetB7 model. On the CDD-CESM dataset, where the input is mammograms instead of x-ray images, results were similar with CheXNet having the highest accuracy (81%) on the test set. We attribute this outcome to visual similarities between these medical image types and natural images.
Based on those, I have concluded that CheXNet, along with the binary cross entropy loss and the Adam optimizer showed the best test performance and was selected for the tasks of extraction of visual features and of semantic tag prediction. The way the pre-trained CheXNet model learns and converges faster than other architectures is shown in Figure 5.2.

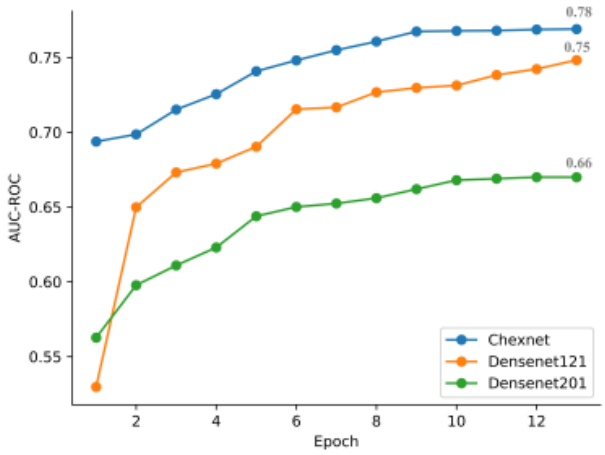

Figure 5.2: Various experiments were conducted to predict the 105 tags from the X-ray images. The AUC-ROC values represent the average AUC-ROC scores calculated on the test set.

5.4 Experiment: Report Generation

In this section, we explain and assess the report generation experiments by means of means of qualitative and quantitative metrics.

5.4.1 CDGPT3.5 Experiment

The CDGPT3.5 model relies on a pre-trained GPT3.5 base model, so I trained my model with the full 50,000+ GPT3.5 vocabulary including all English words as representations. During training we applied a maximum length of 190 tokens which covers all ground truth reports and prevents generated reports from ending early. A learning rate of $10^{-}3$ and a batch size of 15 was used with the Adam optimizer. The IU-XRay and CDD-CESM datasets were trained separately using the model. Two additional experiments were conducted to assess the importance of conditioning the model on both visual and semantic features, where instead the model was conditioned on visual or semantic features. The training started with a "startseq" token marking the start of a sentence, and a standard GPT3.5 end-of-sentence token "$< |endoftext| >$" remained, and "<" acted as the padding token. I used a beam search width of 7 for inference, beyond which I did not see much difference. To avoid the tendency of high score tokens to lead to repeats sequences, the current word was only selected from the top 45 tokens in confidence score. Finally, we chose a minimum output length of size 3 tokens for generated reports. Reports end at whenever the end token is chosen or at maximum length. No length or repetition penalties were imposed.

5.4.2 VSGRU Experiment

The VSGRU model was trained with a vocabulary of 1,480 words including all the words used in the target datasets. After the average pooling layer in the visual model, we extracted visual features. The entire feature vector (size 47×1024) is not

used, but rather this approach, which significantly reduces the training time. The size of the vector of visual features was reduced to 1024 after pooling. Additionally, like for the semantic features, these were averaged across, making their size from *tags size* $\times 400$ to 400 for optimal training. To combat overfitting an additional dropout layer was added to the concatenated visual and semantic features with a rate of 0.4.
Pre trained Word2Vec embeddings were used for initializing the embedding layer of the GRU cell and they were fine tuned. And with a constant learning rate of 1×10^{-3} as the Adam optimizer, we kept a batch size of 15. All reports in the dataset covered less than 165 words, so the model was trained up to that length. Each report generation began with a start sentence token during inference. Then it sampled the word w_{i+1}, given a subsequent word, w_i, based on the predicted output vector distribution using softmax. In case of w, it was allowed to get smaller than 3 words; in this case we threw away the last sentence of the report and resampled w_{i+1} The report otherwise ended once the end token had been chosen.
For technical reasons (e.g. beam search width), we also used as here a beam search width of 7, which retains multiple high-probability paths and indeed improves results relative to the suboptimal greedy search.

5.4.3 Baseline Models

The word-overlap metrics obtained from the trained models were compared with four different baseline model types:

- Non-Hierarchical Recurrent Models: Works like [9], [18], [49] have employed the decoder as a single layer recurrent network.
- Hierarchical Recurrent Models: Following models suggested in [17], [44], where the two layer recurrent network used as the decoder features one layer dedicated to the generation of sentences and the other to the generation of words.
- Retrieval-Based Models: Similarly to [34], models that retrieve similar reports from the dataset.
- Transformer-Based Models: Such as in transformer structure models as decoders used in [40].

I then compared the CDGPT3.5 model with the VSGRU model for experiments on a new dataset (CDD-CESM) for which there was no baseline. I also compared the CDGPT3.5 and VSGRU models using semantic-based metrics the first time such metrics were used for report evaluation.

5.4.4 Discussion on Word-overlap Results

For the IU-XRay dataset, word overlap scores of the CDGPT3.5 generalize a little better than most non-hierarchical models (first section of Table 5.2).
The improved RNCM model [32] showed higher Bleu average due to an increased Bleu-1 score at the cost of Bleu-2 and Bleu-3 scores, as we predict tags to generate a controlled, bounded report. For the transformer-based type (fourth part

of Table 5.2), CDGPT3.5 outperformed the model presented in [43] in all measures but Cider. Despite some inherent advantages, however, the model used in [42] slightly outperformed the transformer by containing an internal recurrent cell within the transformer meaning that some of these advantages are counteracted. Finally, CDGPT3.5 did not outperform hierarchical recurrent models (second section in Table 5.2) or the method of cross-modal retrieval in [42] (third section in Table 5.2). On the other hand, I have also found that my VSGRU is a custom recurrent model giving the highest Cider score out of all other techniques.

Table 5.2: Quantitative results are presented using word-overlap metrics. Here, CDGPT3.5-vis and CDGPT3.5-sem denote the method when employing only visual or semantic features, respectively.

Dataset	Method	Bleu-1	Bleu-2	Bleu-3	Meteor	Rouge	Cider
IU-XRay	CNN-RNN [11]	0.321	0.216	0.135	0.161	0.263	0.118
	LRCN [57]	0.373	0.223	0.153	0.153	0.273	0.185
	ATT-RK [12]	0.362	0.229	0.144	0.176	0.318	0.161
	RNCM [68]	0.791	0.147	0.051	-	-	-
	VSGRU*	0.352	0.215	0.161	0.157	0.248	0.409
	Co-ATTN [8]	0.510	0.380	0.311	0.220	0.440	0.320
	MvH [10]	0.524	0.365	0.320	0.347	0.447	-
	SentSAT [72]	0.446	0.295	0.199	-	0.362	0.309
	CVSE [73]	0.195	-	0.038	0.078	0.150	-
	FFL+CFL [74]	0.565	0.518	0.492	0.551	0.585	-
	RTMIC [78]	0.345	0.239	0.148	-	-	0.328
	RM+MCLN [79]	0.474	0.309	0.221	0.189	0.367	-
	CDGPT3.5-vis*	0.344	0.213	0.141	0.157	0.276	0.235
	CDGPT3.5-sem*	0.359	0.229	0.148	0.151	0.271	0.263
	CDGPT3.5*	0.382	0.250	0.169	0.167	0.283	0.252
CDD-CESM	VSGRU*	0.568	0.533	0.498	0.342	0.647	4.389
	CDGPT3.5-vis*	0.583	0.535	0.491	0.331	0.617	3.812
	CDGPT3.5-sem*	0.580	0.528	0.482	0.319	0.595	3.532
	CDGPT3.5*	0.579	0.537	0.510	0.349	0.657	4.102

Moreover, I evaluated the models utilizing visual or semantic alone and found that combining visual and semantic features enhanced word-overlap metrics, but improved overall performance.
CDGPT3.5 surpassed VSGRU on all word-overlap metrics in the CDD-CESM dataset except for the Cider score. However, word-overlap metrics alone may not appropriately measure quality-generated reports.

5.5 Discussion on Semantic Similarity

The results of the CDGPT3.5 and VSGRU models are shown in Table 5.3 on semantic similarity metrics. On the IU-XRay dataset, VSGRU was outrun by CDGPT3.5 in all semantic similarity metrics, except vector extrema. CDGPT3.5 also outperform VSGRU on the CDD-CESM dataset on all metrics. These results demonstrate that what the CDGPT3.5 model can produce are reports with more semantically aligned semantic representations. Moreover, I tested how well each model performed using visual or semantic features only and discovered that, when combined, these types of features greatly increased word overlap and semantic similarity scores.

Table 5.3: Evaluation of test set performance based on semantic similarity metrics.

Dataset	Method	SkipThoughtCS	Emb. Avg	VExt.	Greedy Match
IU-XRay	VSGRU	0.503	0.818	0.459	0.688
	CDGPT3.5-vis	0.591	0.859	0.526	0.698
	CDGPT3.5-sem	0.627	0.855	0.522	0.681
	CDGPT3.5	0.634	0.861	0.517	0.712
CDD-CESM	VSGRU	0.761	0.832	0.718	0.815
	CDGPT3.5-vis	0.724	0.823	0.746	0.812
	CDGPT3.5-sem	0.655	0.728	0.601	0.676
	CDGPT3.5	0.796	0.840	0.778	0.827

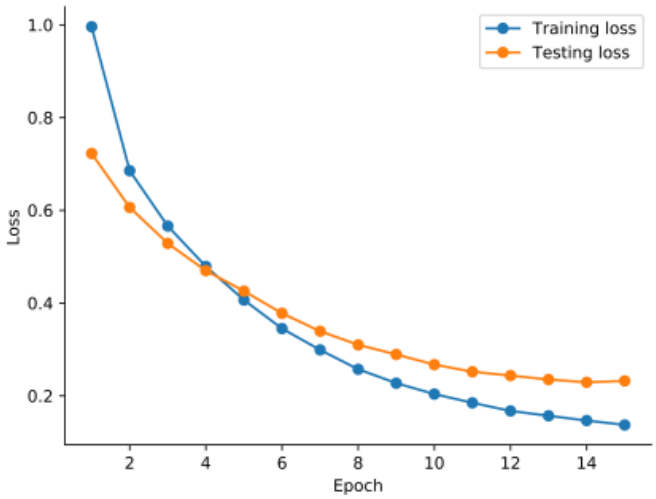

Figure 5.3: The loss values during both training and testing phases for the CDGPT3.5 model.

5.6 Time performance evaluation

Figure 5.3 shows the training durations for report generation using a recurrent-based model as well as a transformer-based model. Successively we trained a non-hierarchical recurrent model and train DistilGPT3.5 as the decoder, we find that they require around twice the time to train. While the VSGRU model uses averaged visual and semantic features, the CDGPT3.5 leverages all feature data. Additionally, the exponential running time was also achieved in less than one second by CDGPT3.5, run on the pre-trained transformer, compared to 3.9 seconds needed for VSGRU. Furthermore, we find that on average, it takes CDGPT3.5 slightly over 4.5 seconds to fully report on a single image using a 7 beam width, while VSGRU requires over 19.8 seconds.

5.7 Automated Segmented Annotation

In this section we describe the experiment for which segmentation annotations where generated for the CDD-CESM images and the evaluation of the results.

5.7.1 Experiment

For this experiment, I fine-tuned an EfficientNetB0 model, pre-trained on the ImageNet dataset, to classify images into three categories: Normal, Benign, or Malignant. We trained the model on the whole CDD-CESM dataset to produce segmentations for all the images. The images were resized to have dimensions of 224x224 using interpolation, anti aliasing to match input size of the pre trained model. The mean and standard deviation of them were then subtracted and divided by them. With training, I applied random augmentations, such as cropping, zooming, and horizontal flipping, to increase robustness.
During training, all model weights were fine tuned. We used a batch size of 30 and decayed initial learning rate to 1e-3. In addition, the final visual features were fed to a dropout layer with a drop probability of 0.75 prior to the classifier. Heatmap intensities greater than the top 23% were used to define the abnormal segment for segmentation. I refined these segmentations by intersecting the identified segments with the top 14% of white pixel intensities in the image. The thresholds were tuned experimentally and selected.

5.7.2 Discussion

Compared to ground truth segmentations, the model achieved an average Intersection over Union (IoU) of 65.4%, overlap50 of 84.5%, and an average F1 score of 72.3%. These metrics are shown for different image groups in table 5.4. Mass enhancement had the highest overlap50 of 92%, while in cases postoperative we see the lowest overlap50 of 78%. The model may not be able to fully capture changes that occur postoperatively: e.g. edema and skin thickening.
Patients aged 70 years or older had the highest overlap50 of 95%, patients 40 or younger had the lowest overlap50 of 79%. However, visualization accuracy was reduced as breast density increased. In subtracted images, an overlap50 of 82% compared to 87% as attained with low energy images. This difference is due, in part, to the dense adenotic tissue, which overlies abnormalities and obscures them in low-energy images. Suppressing this dense tissue allows abnormalities to become clearer by removing it.
I suggest radiologists provide both low-energy and subtracted images for each patient in each view, to increase reliability in the drawing of conclusions. In addition, benign findings had a 50 overlap of 77% compared to malignant findings of 91%. The non-enhancing nature of most benign lesions in subtracted images explains this disparity. In low-energy images, dense breast tissue frequently overlaid benign lesions or heavily shadowed them, or the result density and orientation mimicked appropriate parenchymal tissue.
Additionally, I effectively detected highly cellular benign findings. Nevertheless, we found a significant reduction in accuracy for cases of multiplicity and retroareolar locations.
In general, some halo (breast within breast) or ripple artifacts inhibited the model from achieving better detection accuracy in some subtracted images. In figure 5.4 shows the sample outputs of the model.

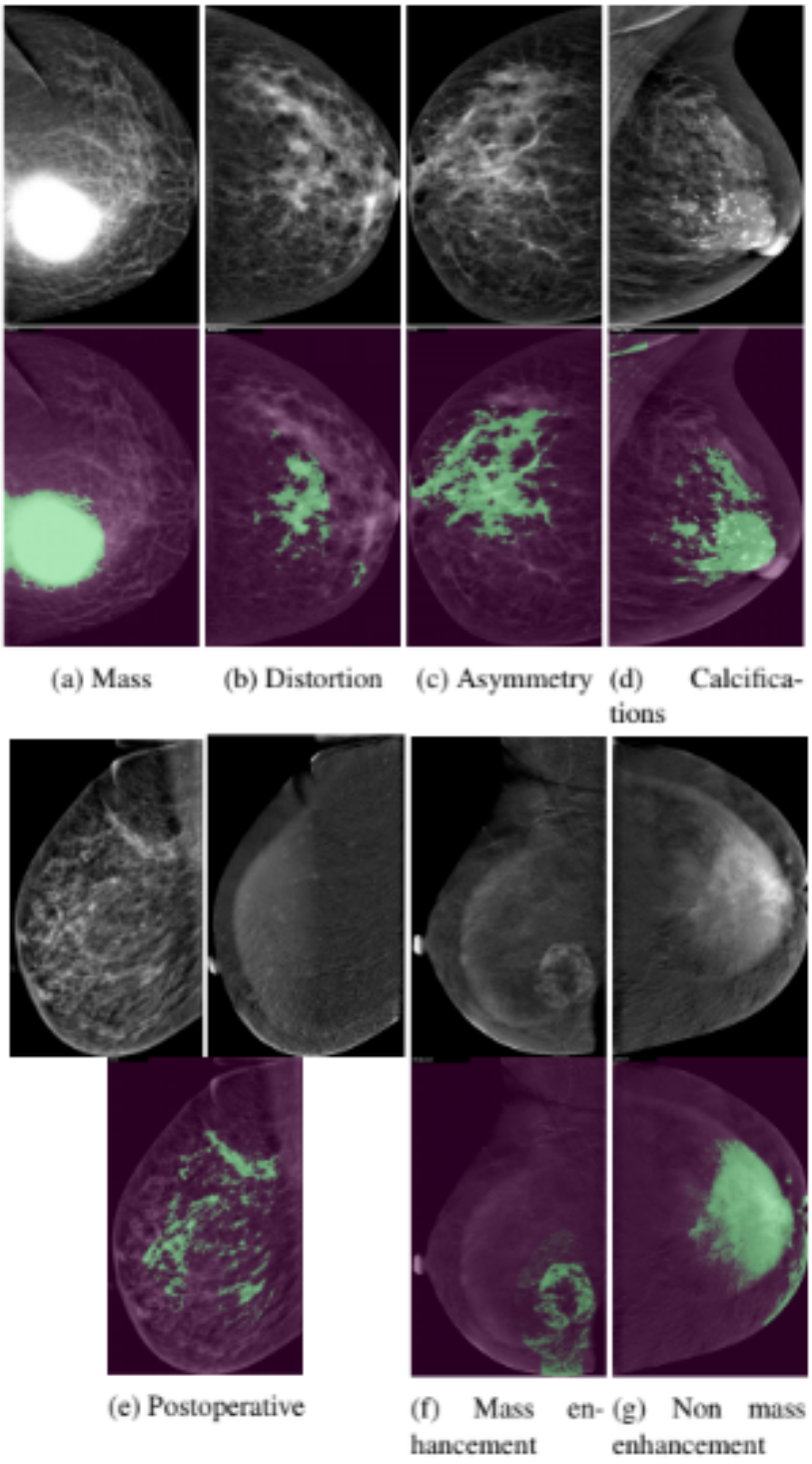

Figure 5.4: Examples of various cases and their respective automatically generated segmentations

Table 5.4: Detailed results of the segmentation model.

Category	**Details**	**Count**	**Overlap50**	**IOU**	**F1**
Findings	Mass	314	0.84	0.63	0.71
	Distortion	50	0.88	0.69	0.78
	Asymmetry	225	0.86	0.68	0.77
	Calcifications	242	0.80	0.60	0.69
	Postoperative	161	0.76	0.59	0.66
	Mass enhancement	330	0.90	0.67	0.72
	Non-mass enhancement	188	0.88	0.70	0.78
Type	DM	670	0.80	0.62	0.70
	CM	585	0.85	0.64	0.70
Pathology	Benign	590	0.73	0.57	0.62
	Malignant	665	0.89	0.67	0.75
View	MLO	638	0.82	0.62	0.70
	CC	625	0.82	0.63	0.70
Model	GE	1180	0.83	0.62	0.70
	Hologic	78	0.72	0.58	0.65
Age	<40	245	0.77	0.64	0.71
	40-69	955	0.82	0.63	0.69
	$\geq$ 70	60	0.93	0.72	0.79

Chapter 6

Conclusion and Future Work

Using this research, we introduce a novel approach for training DistilGPT3.5 on visual and semantic features and show that this model is able to generate comprehensive full text reports for chest X-ray scans. Compared to previous non-hierarchical recurrent models and transformer-based methods, conditioning a pre-trained transformer model demonstrated several advantages: By doing so, I (1) achieve faster training times compared to CPTM, (2) should in general not require specifying a particular vocabulary for which the model should generate, and (3) should perform better on word-overlap quantitative metrics. I am further the first to employ semantic similarity measures at the embedding level in analyzing medical reports, overcoming the shortcomings of word overlap approaches which are in general inadequate for vital systems including medical reporting.
I also compiled a dataset consisting of the National Cancer Institute mammograms and contrast-enhanced mammograms, their annotations and full text reports. Furthermore, I presented an innovative method for automatic annotation of mammogram images based solely on class labels from the annotations for the latter stage of the task, being aware of segmentation annotation being very time consuming.
However, the promising results are somewhat limited in that they need to be improved to bring the reliability level of the generated reports up. The main problem is that from time to time they neglect to include significant data in the reports. The model's generalization ability is limited by the available datasets which renders it vulnerable to overfitting. To deal with this, I strongly recommend creating and releasing larger datasets that contain multiple reports per image.
Finally, I describe CDGPT3.5, a deep learning model that is conditioned on the visual and semantic features of pre trained DistilGPT3.5 transformer model. For quantitative evaluation, semantic similarity metrics were incorporated alongside word-overlap metrics, and the model was tested on two datasets: on the IU-XRay dataset and the newly announced dataset CDD-CESM. To spark further investigation of pre trained transformers as decoders that generate medical reports from small datasets, I hope to provide these findings. Finally, I entreat researchers to approach the system critical to medical reporting with a blend of quantitative and qualitative methods. For further research and evaluation, the code and trained models are made publicly available.

6.1 Future Work

There are several ways that could enrich the work of the current. The lack of large, high-quality medical datasets together with the related radiology reports is one of the most important challenges. To my knowledge, CDD-CESM is only the second dataset that couples medical reports to medical images. Below are some areas where future research could lead to significant improvements:
Explore additional transformer models: Also experiment with other transformer based models, like BERT [43], GPT-4 [18], T5 [32], Linformer [46], or Reformer [46], as decoders using this proposed method of conditioning. These models might be tested and better generative capabilities may be revealed along with better results.
Utilize transformer-based semantic features: In this work, we extract semantic features from pre trained word2vec embeddings trained on medical corpora. In future research, transformer based embeddings trained on medical text datasets may be employed to supply richer, more context aware semantic features.
Leverage transformers for semantic similarity metrics: I also calculated semantic similarity between generated and ground truth reports using pre trained recurrent models, but transformer models can also be used to perform similar tasks. These models may actually generate more accurate similarity readings.
Incorporate multi-view models: As the proposed model processes a single image at a time, it increases the general usability, but may reduce the ability to detect some anomalies. As in [6], multi-view models utilize both frontal and lateral chest X rays and hence perform a workflow similar to that of radiologists. Performance could be improved by adapting this technique to the proposed methodology.
Augment training with real reports: There is a major limitation of the IU-XRay dataset in having multiple reports for a single image. This allow the model's learning to be limited and prone to overfiting. Additionally, prediction reports are limited to quantitative evaluations, because predicted reports must be rigorously matched to a single reference report. The model's robustness and evaluation can be greatly improved by addition of reports by more than on professional, or by use of paraphrasing techniques like [32].
By further refining the approach and expanding the boundaries of automated medical report generation these areas can be addressed in future studies.

Bibliography

[1] J. Suckling, "The mammographic image analysis society digital mammogram database," in *Digital Mammo*, 1994, pp. 375–386.

[2] Y. LeCun, P. Haffner, L. Bottou, and Y. Bengio, "Object recognition with gradient-based learning," in *Shape, Contour and Grouping in Computer Vision*, vol. 1681, 1999, p. 319.

[3] M. Heath, K. Bowyer, D. Kopans, R. Moore, and W. Kegelmeyer, "The digital database for screening mammography," in *Proceedings of the Fifth International Workshop on Digital Mammography*, 2001, pp. 212–218.

[4] J. M. Lewin, P. Isaacs, V. Vance, and F. Larke, "Dual-energy contrast-enhanced digital subtraction mammography: Feasibility," *Radiology*, vol. 229, no. 1, pp. 261–268, 2003.

[5] T. M. Lehmann, M. O. Guld, C. Thies, *et al.*, "Content-based image retrieval in medical applications," *Methods of Information in Medicine*, vol. 43, no. 04, pp. 354–361, 2004.

[6] F. Luisier, C. Vonesch, T. Blu, and M. Unser, "Fast interscale wavelet denoising of poisson-corrupted images," *Signal Processing*, vol. 90, no. 2, pp. 415–427, 2010.

[7] M. Zaoual, "Fda clearance for ge contrast enhanced spectral mammography cesm," 2011.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, vol. 25, Lake Tahoe, Nevada, United States, 2012, pp. 1106–1114.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *26th Annual Conference on Neural Information Processing Systems*, Lake Tahoe, Nevada, United States, 2012, pp. 1106–1114.

[10] R. Hupse, M. Samulski, M. Lobbes, *et al.*, "Computer-aided detection of masses at mammography: Interactive decision support versus prompts," *Radiology*, vol. 266, no. 1, pp. 123–129, 2013.

[11] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013*, Lake Tahoe, Nevada, United States, 2013, pp. 3111–3119.

[12] C. D'Orsi, *ACR BI-RADS Atlas: Breast Imaging Reporting and Data System, 5th ed.* 2014.

[13] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *3rd International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, 2015.

[14] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations (ICLR)*, 2015. [Online]. Available: https://arxiv.org/abs/1412.6980.

[15] U. Lalji, C. Jeukens, N. Houben, *et al.*, "Evaluation of low-energy contrast-enhanced spectral mammography images by comparing them to full-field digital mammography using euref image quality criteria," *European Radiology*, vol. 25, no. 10, pp. 2813–2820, 2015.

[16] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, May 2015.

[17] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *3rd International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, 2015.

[18] C. Szegedy, W. Liu, Y. Jia, *et al.*, "Going deeper with convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, 2015, pp. 1–9.

[19] A. Chetlen, J. Mack, and T. Chan, "Breast cancer screening controversies: Who, when, why, and how?" *Clinical Imaging*, vol. 40, no. 2, pp. 279–282, 2016.

[20] Ö. Çiçek, A. Abdulkadir, S. Lienkamp, T. Brox, and O. Ronneberger, "3d u-net: Learning dense volumetric segmentation from sparse annotation," in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, vol. 9901, Athens, Greece, 2016, pp. 424–432.

[21] L. Gondara, "Medical image denoising using convolutional denoising autoencoders," in *IEEE International Conference on Data Mining Workshops (ICDM Workshops)*, Barcelona, Spain, 2016, pp. 241–246.

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 770–778.

[23] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, Berlin, Germany, Aug. 2016, pp. 1715–1725.

[24] C. Bhimani, D. Matta, R. Roth, *et al.*, "Contrast-enhanced spectral mammography: Technique, indications, and clinical applications," *Academic Radiology*, vol. 24, no. 1, pp. 84–88, 2017.

[25] B. Erickson, "Machine learning: Discovering the future of medical imaging," *Journal of Digital Imaging*, vol. 30, p. 391, 2017.

[26] A. Howard, M. Zhu, B. Chen, *et al.*, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *CoRR*, vol. abs/1704.04861, 2017.

[27] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, pp. 2261–2269.

[28] G. Kang, K. Liu, B. Hou, and N. Zhang, "3d multi-view convolutional neural networks for lung nodule classification," *The Public Library of Science*, vol. 12, no. 11, pp. 1–21, 2017.

[29] R. S. Lee, F. Gimenez, A. Hoogi, K. K. Miyake, M. Gorovoy, and D. Rubin, "A curated mammography data set for use in computer-aided detection and diagnosis research," *Scientific Data*, vol. 4, no. 1, pp. 1–9, 2017.

[30] J. V. Manjon, "Mri preprocessing," in *Imaging Biomarkers*, 2017, pp. 53–63.

[31] P. Rajpurkar, J. Irvin, K. Zhu, *et al.*, "CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning," *CoRR*, vol. abs/1711.05225, 2017. [Online]. Available: http://arxiv.org/abs/1711.05225.

[32] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need," in *Advances in neural information processing systems*, vol. 30, Long Beach, CA, USA, 2017, pp. 5998–6008.

[33] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. Summers, "Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, pp. 3462–3471.

[34] Z. Zhang, Y. Xie, F. Xing, M. McGough, and L. Yang, "Mdnet: A semantically and visually interpretable medical image diagnosis network," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, Honolulu, HI, USA, 2017, pp. 3549–3557.

[35] B. Dadonaite and M. Roser, *Pneumonia*, Our World in Data, 2018. [Online]. Available: https://ourworldindata.org/pneumonia.

[36] B. Jing, P. Xie, and E. Xing, "On the automatic generation of medical imaging reports," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018*, Melbourne, Australia, 2018, pp. 2577–2586.

[37] R. McDonald, G. Brokos, and I. Androutsopoulos, "Deep relevance ranking using enhanced document-query interactions," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Brussels, Belgium, 2018, pp. 1849–1860.

[38] A. Rezvantalab, H. Safigholi, and S. Karimijeshni, "Dermatologist level dermoscopy skin cancer classification using different deep learning convolutional neural networks algorithms," *CoRR*, vol. abs/1805.02994, 2018.

[39] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "Bert: Pretraining of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019*, Minneapolis, MN, USA, 2019, pp. 4171–4186.

[40] W. Al-Dhabyani, M. Gomaa, H. Khaled, and A. Fahmy, "Deep learning approaches for data augmentation and classification of breast masses using ultrasound images," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 5, 2019.

[41] Y. Huang, J. Xu, Y. Zhou, T. Tong, and X. Zhuang, "Diagnosis of alzheimer's disease via multi-modality 3d convolutional neural network," *Frontiers in Neuroscience*, vol. 13, p. 509, 2019.

[42] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *CoRR*, vol. abs/1901.00726, 2019.

[43] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: Smaller, faster, cheaper and lighter," *CoRR*, vol. abs/1910.01108, 2019.

[44] M. Tan, B. Chen, R. Pang, *et al.*, "Mnasnet: Platform-aware neural architecture search for mobile," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, 2019, pp. 2820–2828.

[45] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *Proceedings of the 36th International Conference on Machine Learning (ICML)*, vol. 97, Long Beach, California, USA, 2019, pp. 6105–6114.

[46] J. Yuan, H. Liao, R. Luo, and J. Luo, "Automatic radiology report generation based on multi-view image fusion and medical concept enrichment," in *Medical Image Computing and Computer Assisted Intervention - MICCAI 2019*, vol. 11769, Shenzhen, China, 2019, pp. 721–729.

[47] Z. Ziegler, L. Kyriazi, S. Gehrmann, and A. Rush, "Encoder-agnostic adaptation for conditional language generation," *CoRR*, vol. abs/1908.06938, 2019.

[48] R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," *International Journal of Computer Vision*, vol. 128, no. 2, pp. 336–359, 2020.

[49] H. Sung, J. Ferlay, R. Siegel, *et al.*, "Global cancer statistics 2020: Estimates of incidence and mortality worldwide for 36 cancers in 185 countries," *CA: A Cancer Journal for Clinicians*, vol. 71, no. 3, pp. 209–249, 2021. DOI: 10.3322/caac.21660.