# MedSegText – A Cross-Modal, Multi-Task Framework for Automated Chest CT Interpretation

Tamilarasee Sethuraj
*Department of Computer Science*
*Illinois Institute of Technology*
Chicago, Illinois, USA
tsethuraj@hawk.iit.edu

Shouvik Roy
*Department of Computer Science*
*Illinois Institute of Technology*
Chicago, Illinois, USA
sroy20@iit.edu

***Abstract*—This project implements MedSegText, a deep learning framework designed for simultaneously delineating pulmonary abnormalities and generating concise radiological findings from chest CT scans. The architecture leverages shared visual feature representations from a common encoder, enabling joint training of both segmentation and text generation tasks with the aim of improving diagnostic efficiency for interpreting volumetric CT data. The model employs a ConvNeXt encoder to extract hierarchical visual features, which serve as input to two parallel decoder branches: 1) A U-Net-style segmentation decoder utilizing skip connections for precise pixel-level lesion mapping. 2) A Transformer-based text decoder generating single-sentence findings via cross-attention to the encoded visual features. MedSegText produces aligned segmentation maps and textual findings in a single inference step. This unified approach seeks to streamline radiology workflows and provide grounded diagnostic descriptions, forming a basis for future analysis of joint learning effects and the implementation of enhanced cross-task interaction mechanisms for large-scale, explainable CT analysis.The source code for this project is publicly available at https://github.com/Tamilarasee/MMI_Unet_Lesion_Segmentation/tree/MedSegText.**

***Index Terms*—Medical Imaging, CT Scan, Image Segmentation, Report Generation, Multi-Task Learning, Multi-Modal, Deep Learning, ConvNeXt, Transformer, U-Net**

## I. Introduction

Medical imaging, particularly Computed Tomography (CT), plays a pivotal role in diagnosing a wide array of conditions. Interpreting these complex volumetric scans, however, is a demanding task for radiologists, often involving meticulous examination to identify subtle abnormalities and the subsequent composition of detailed descriptive reports. This process is not only time-consuming, contributing to significant workload, but also susceptible to inter-observer variability. The challenge is compounded by the need for both accurate spatial localization of findings (e.g., lesions, opacities) and the generation of precise, standardized textual descriptions for clinical communication and record-keeping. Automating aspects of this workflow holds immense potential for improving efficiency, consistency, and throughput in clinical practice, especially for widespread conditions like those observed during the COVID-19 pandemic, often analyzed using datasets like MosMedData [1].

This project explores the development of an automated system, termed MedSegText, designed to address both critical aspects of CT interpretation: segmentation and text generation. We investigate a multi-task deep learning approach where a single model learns to simultaneously delineate relevant regions within a CT scan and generate corresponding concise textual findings (often single-sentence descriptions). The core hypothesis is that training these tasks jointly, using shared underlying visual representations derived from a common encoder, can lead to synergistic benefits, potentially improving both segmentation accuracy and the contextual relevance of the generated text compared to single-task models.

Over the course of this project, we implemented and evaluated several architectural approaches to understand the complexities involved:

- A baseline traditional U-Net for segmentation using only image input.
- An exploration based on the MMI-UNet architecture [3], investigating the use of both image and text input to potentially guide segmentation.
- A text-generation-only model inspired by architectures like RATCHET [8], focusing on generating findings from image input alone using a Transformer decoder.
- The proposed MedSegText framework, combining a shared ConvNeXt encoder with parallel U-Net-style segmentation and Transformer-based text decoders for joint output generation from image input.

This report details the motivation, related literature, implementation journey through these architectures, experimental setup, results obtained for each approach, and concludes with a discussion of the findings and potential future work.

## II. Related Work

The automated analysis of medical images, especially CT scans, has seen significant advancements fueled by deep learning. Our work builds upon research in several related areas: medical image segmentation, automated radiological report generation, and multi-modal learning in medicine.

### A. Medical Image Segmentation

Deep convolutional neural networks (CNNs), particularly U-Net and its numerous variants (e.g., UNet++, Attention U-Net), have become the standard for various medical image segmentation tasks. These encoder-decoder architectures, utilizing

skip connections to preserve fine-grained spatial information, excel at pixel-level delineation of anatomical structures and pathological findings in modalities like CT and MRI. Modern CNN designs like ConvNeXt [4], which we utilize as our encoder backbone, have further pushed performance boundaries. Some research has explored incorporating attention mechanisms or, as seen in MMI-UNet [3], leveraging textual information via visual-textual matching attention within the encoder pathway to potentially improve segmentation accuracy, particularly by focusing the model on text-relevant visual features.

### B. Automated Report Generation

Generating textual descriptions from medical images shares similarities with the broader field of image captioning [9]. Early methods often used CNNs for image features and Recurrent Neural Networks (RNNs) like LSTMs for text generation. More recently, Transformer-based architectures [7], leveraging self-attention and cross-attention mechanisms, have shown superior performance in capturing long-range dependencies and generating more coherent text. RATCHET [8] specifically adapted a CNN-Transformer architecture (DenseNet-121 encoder, Transformer decoder) for generating chest X-ray reports, demonstrating the effectiveness of this approach in the medical domain. These models typically focus solely on generating the text report from the image input.

### C. Multi-Modal Learning in Medical AI

Integrating information from multiple modalities, such as images and text (radiology reports), is an active research area. Approaches like MMI-UNet [3] utilize text embeddings (e.g., from CXR-BERT [5]) alongside image features, often employing cross-modal attention mechanisms (like their Image-Text Matching module) within the encoder to enhance tasks like segmentation. The core idea is that text can provide valuable context or guidance for interpreting the visual data. Conversely, our MedSegText project explores the direction of *jointly generating* both segmentation maps and textual findings from a single image input within a unified framework. While sharing an encoder allows for implicit cross-modal learning (visual features must be good for both tasks), our current iteration focuses on parallel decoders rather than direct cross-attention between the decoder tasks themselves, aiming for interpretable and aligned multi-modal outputs.

## III. Implementation

This section details the architectures and implementation specifics of the models explored throughout the project. All models were implemented using PyTorch [13].

### A. Baseline U-Net (Segmentation)

Our baseline segmentation model employs a standard U-Net architecture, based on the foundational work by Ronneberger et al. [2], but adapted with common modern deep learning practices as illustrated in Fig.1.This model follows the typical encoder-decoder structure with skip connections, utilizing blocks containing 3x3 convolutions, Batch Normalization, and ReLU activations. Max pooling is used for downsampling in the encoder, while transpose convolutions handle upsampling in the decoder. A final 1x1 convolution maps features to the single output segmentation class. This model takes only the CT image as input and outputs a segmentation map.Further details regarding hyperparameters and evaluation results are presented in Results Section.

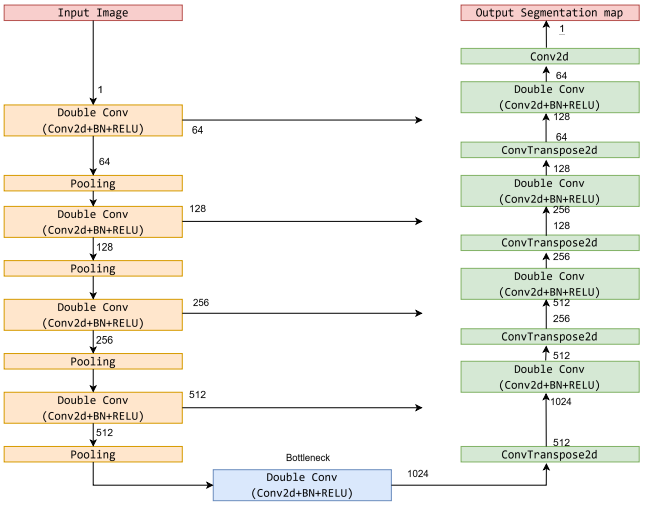


Fig. 1. Architecture of the baseline U-Net model used for segmentation. Blocks show Double Convolution (Conv2d+BN+ReLU), Pooling, Transpose Convolution, and skip connections.

### B. Multi-Modal Segmentation using MMI-UNet

For multi-modal segmentation, we implemented the MMI-UNet architecture presented by Bui et al. [3]. This model leverages both visual (CT image) and textual (report/description) information.As detailed in the original paper [3], the architecture as shown in Fig. 2 employs a ConvNeXt-Tiny [4] image encoder and a frozen CXR-BERT [5] text encoder. Fusion occurs via Image-Text Matching (ITM) modules at multiple encoder levels, with the resulting visual features passed via skip connections to a U-Net-style decoder featuring Efficient Channel Attention (ECA) [6]. The model accepts an image-text pair as input and produces a segmentation map. Further implementation details, hyperparameters, and evaluation results are presented in Results Section.

### C. Text Generation (RATCHET Adaptation)

To explore text generation independently, we adapted the RATCHET approach [8]. The primary modification involved using a pre-trained ConvNeXt-Tiny model [4] as the image encoder, differing from RATCHET's original DenseNet-121. The final feature map (`f4`) extracted by the ConvNeXt encoder was flattened to serve as the 'memory' context for a standard Transformer decoder [7]. The decoder consisted of an embedding layer for input tokens, positional encoding to inject sequence information, and multiple (*num_decoder_layers*) Transformer decoder layers. For these layers, we utilized

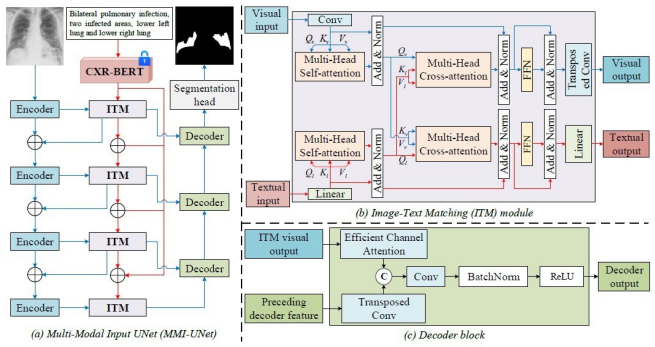


Fig. 2. Architecture of the MMI-UNet model implemented for multi-modal segmentation (Adapted from Bui et al. [3]). Shows (a) overall architecture, (b) Image-Text Matching (ITM) module, and (c) decoder block.

the standard PyTorch 'nn.TransformerDecoderLayer' module, where each layer includes the necessary components: masked multi-head self-attention, multi-head cross-attention (attending to the image 'memory'), and a feed-forward network, along with residual connections and layer normalization. A final linear layer projected the decoder output sequence to the vocabulary size, yielding the next token probabilities. This model takes only the image as input and autoregressively generates the output text sequence. Further details on hyperparameters and evaluation metrics are provided in Results Section.

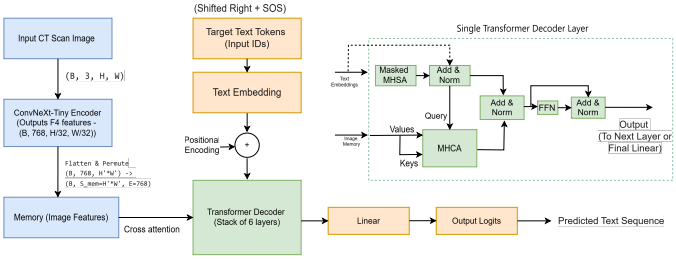


Fig. 3. Architecture for text generation (adapted from RATCHET [8]). A ConvNeXt-Tiny encoder extracts image features, which serve as memory for a standard Transformer decoder generating the text sequence. The inset shows the components within a single Transformer decoder layer.

### *D. MedSegText (Joint Segmentation and Text Generation)*

Building upon the insights gained from the single-task models, the final MedSegText architecture was developed to explore the potential of multi-task learning for simultaneous segmentation and text generation. The goal was to create a unified model capable of producing aligned visual and textual outputs from a single CT image input. This architecture leverages successful components identified earlier: the ConvNeXt backbone for strong visual feature extraction and the Transformer-based decoder for effective text generation.

A core design principle is the use of a **shared visual encoder**. The pre-trained `ConvNeXtEncoder` [4] (identical to the one used in the text-only model) processes the input image through its hierarchical stages, producing multi-scale feature maps (`f1` to `f4`). Sharing the encoder is parameter-efficient and hypothesizes that learning representations beneficial for both segmentation (requiring spatial detail) and text generation (requiring semantic understanding of visual content) can lead to more robust features compared to single-task training.

Two parallel decoder branches operate on the features extracted by the shared encoder, as illustrated in Fig. 4:

- **Segmentation Decoder:** To ensure effective spatial localization, a standard **U-Net style convolutional decoder** was implemented, closely mirroring the structure of the baseline U-Net (Fig. 1). This decoder takes the bottleneck features (`f4`) and progressively upsamples them using transpose convolutions. Crucially, at each upsampling stage, it concatenates the feature map with high-resolution features provided via **skip connections** from the corresponding encoder level (`f3`, `f2`, `f1`). These skip connections allow the decoder to recover fine-grained details lost during encoding. Standard `DoubleConv` blocks (Convolution-BatchNorm-ReLU pairs) refine the features at each stage. A final 1x1 convolution maps the features to the output channel dimension (1 for binary segmentation), producing logits which are then interpolated to the original image resolution.
- **Text Decoder:** This branch directly reuses the architecture from the RATCHET-inspired text-only model (Fig. 3), which demonstrated promising results. The bottleneck visual features (`f4`) are flattened to serve as the **visual context ('memory')** for a standard **Transformer decoder** [7]. This decoder employs masked self-attention to process the sequence of input text embeddings (combined with positional encodings) and, critically, uses **cross-attention** to query the visual context ('memory') at each step. This cross-attention mechanism allows the model to condition the generated text on the relevant visual information extracted by the shared encoder. The decoder operates autoregressively, predicting one token at a time, with a final linear layer mapping the output states to vocabulary logits. Key hyperparameters (such as *num_decoder_layers*=6, *nhead*=8) were adopted from the text-only implementation and are detailed in Results Section.

This complete MedSegText model takes the CT image as input (along with target text sequences during training) and produces two distinct outputs: segmentation logits and text logits. It was trained end-to-end using a **combined loss function** (`JointLoss`), weighting the contributions of the segmentation loss (e.g., Dice + BCE) and the text generation loss (Cross-Entropy) to update both the decoders and the shared encoder parameters. This approach aims to balance the learning objectives of both tasks, as explored in the experiments detailed in Section Results.

## IV. Experimental Evaluation

### *A. Dataset*

This project utilized the publicly available MosMedData+ dataset [1], which contains 2729 CT scan slices focused on

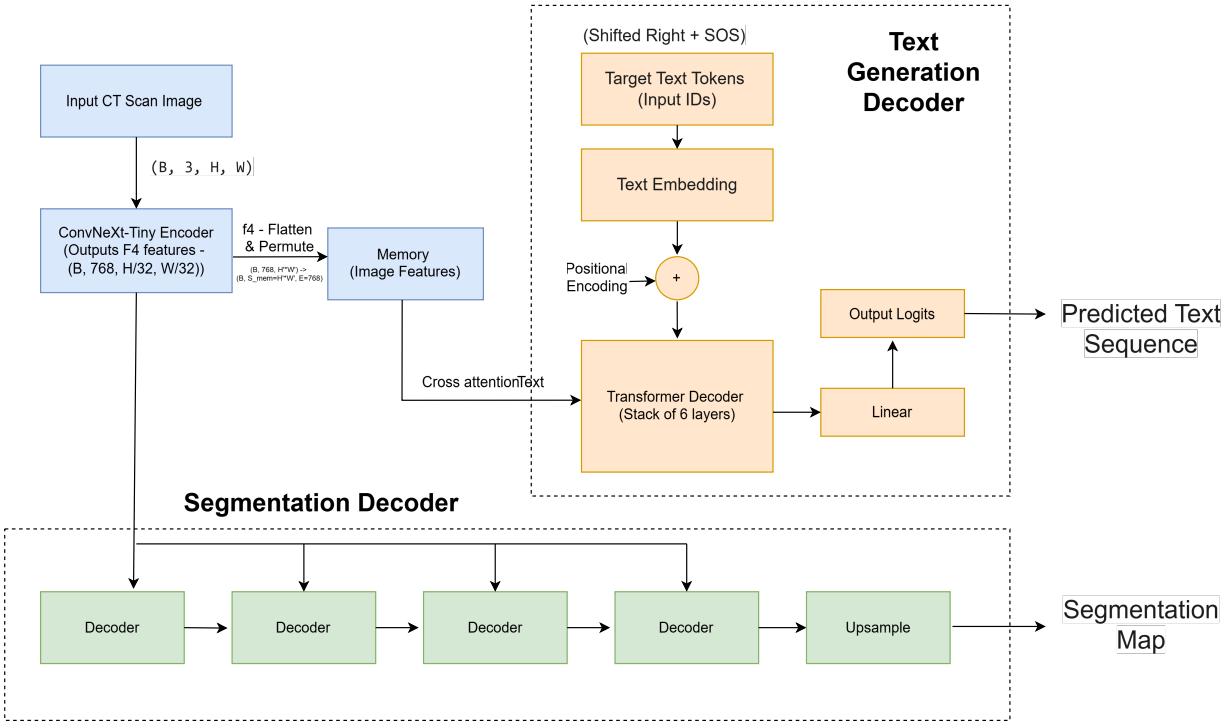


Fig. 4. Overall architecture of the proposed MedSegText model. A shared ConvNeXt-Tiny encoder extracts multi-scale visual features. These features feed two parallel decoders: a U-Net style Segmentation Decoder using skip connections (F1-F4) and a Transformer-based Text Generation Decoder using flattened F4 features as memory for cross-attention.

lung infections, often associated with COVID-19. Each slice is paired with a textual annotation describing findings such as infection laterality, number of lesions, and approximate location. The dataset was randomly split into training (2183 samples), validation (273 samples), and testing (273 samples) sets. All images were resized to 224x224 pixels. Standard data augmentation techniques, including random zoom (10%, p=0.1), were applied to the training set. Input images were normalized to the [0, 1] range. For text processing, the `microsoft/BiomedVLP-CXR-BERT-specialized` tokenizer [5] was used, with special tokens ([SOS], [EOS], [PAD]) added, resulting in a final vocabulary size of 30524. Text sequences were padded to a maximum length of 50 tokens.

### B. Metrics

The performance of the implemented models was evaluated using standard metrics appropriate for each task:

- **Segmentation:** Dice Similarity Coefficient (Dice) and Intersection over Union (IoU / Jaccard Index) were used to measure the overlap between the predicted segmentation masks (thresholded at 0.5 after sigmoid activation) and the ground truth binary masks.
- **Text Generation:** BLEU (BLEU-1 to BLEU-4) [10], METEOR [11], and ROUGE-L (F1-score) [12] were calculated using the NLTK [14] and rouge-score libraries to assess the quality of the generated text findings against the ground truth references. Standard tokenization and lowercasing were applied before metric calculation.

### C. Implementation Details

All models were implemented using PyTorch and trained on an RTX 6000 GPU. The AdamW optimizer [15] was used for all training phases. For the text-only phase, a learning rate of 1e-4 was used for 30 epochs with a batch size of 8. For the initial joint training experiment (Exp1), a learning rate of 1e-4 was used for 50 epochs with loss weights Seg=1.0, Text=0.5. For the second joint training experiment (Exp2), the learning rate was reduced to 1e-5 for 50 epochs with loss weights Seg=1.0, Text=1.0. The segmentation task used a combined Dice loss and Binary Cross-Entropy with Logits loss (‘nn.BCEWithLogitsLoss‘). The text generation task used Cross-Entropy loss (‘nn.CrossEntropyLoss‘) with the PAD token ignored. The ‘JointLoss‘ function combined these weighted losses for joint training.

## V. Results

This section presents the quantitative and qualitative results obtained from evaluating the different implemented architectures on the MosMedData+ test set. The evaluation aims to demonstrate the capabilities of each model stage, leading up to the joint MedSegText framework.

### A. Experimental Baselines

This subsection details the performance of the foundational models developed for individual tasks: segmentation-only and text-generation-only.

*1) Baseline U-Net for Segmentation:* The standard U-Net architecture served as an initial baseline for the segmentation task.

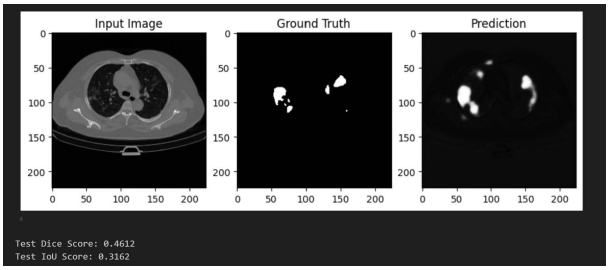


Fig. 5. Baseline U-Net: Sample qualitative output and metrics. (Epoch 3, LR 0.001, Dice: 0.4612, IoU: 0.3162)

TABLE I
BASELINE U-NET: HYPERPARAMETERS

| Parameter | Value |
|---|---|
| Epochs | 3 |
| Learning Rate | 0.001 |
| Optimizer | Adam |
| Loss Function | Dice + BCE Loss |
| Batch Size | 10 |

TABLE II
BASELINE U-NET: SEGMENTATION METRICS

| Metric | Value |
|---|---|
| Average Dice Score | 0.4612 |
| Average IoU Score | 0.3162 |

The baseline U-Net achieved a modest performance, providing an initial benchmark for segmentation accuracy. The qualitative results (Fig. 5) show its capability to identify larger lesion areas but with room for improvement in boundary precision.

*2) MMI-UNet Inspired Model for Segmentation:* This model incorporated a ConvNeXt encoder with a U-Net style decoder, drawing inspiration from the MMI-UNet encoder structure but applied to a segmentation-only task.

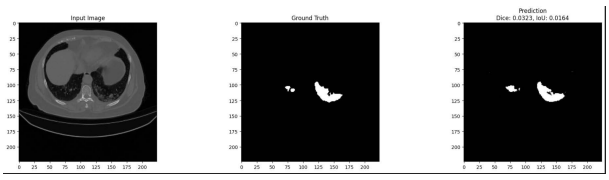


Fig. 6. MMI-UNet Inspired: Sample qualitative output and metrics. (Epoch 60, LR 1e-4, Dice: 0.0323, IoU: 0.0164)

TABLE III
MMI-UNET INSPIRED: HYPERPARAMETERS

| Parameter | Value |
|---|---|
| Epochs | 60 |
| Learning Rate | 1e-4 |
| Optimizer | Adam |
| Loss Function | Dice + BCE Loss |
| Batch Size | 1 |

TABLE IV
MMI-UNET INSPIRED: SEGMENTATION METRICS

| Metric | Value |
|---|---|
| Average Dice Score | 0.0323 |
| Average IoU Score | 0.0164 |

The MMI-UNet inspired model, using the ConvNeXt encoder, was trained for an extended period. The quantitative results (Table IV) indicate challenges in this specific run, with lower scores compared to the baseline U-Net. Figure 6 shows a sample output. This particular experiment with the MMI-UNet encoder for segmentation alone did not yield improved results over the simpler U-Net, potentially due to optimization challenges or the need for further hyperparameter tuning for this specific configuration.

*3) Text-Only Generation Model (RATCHET-Inspired):* This model focused solely on generating textual findings from CT images, using a ConvNeXt encoder and a Transformer-based decoder.

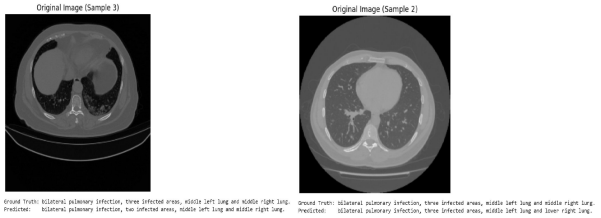


Fig. 7. Text-Only Model: Sample qualitative text generation. (Epoch 30, LR 1e-4)

TABLE V
TEXT-ONLY MODEL: HYPERPARAMETERS

| Parameter | Value |
|---|---|
| Epochs | 30 |
| Learning Rate | 1e-4 |
| Optimizer | Adam |
| Loss Function | Cross-Entropy Loss |
| Batch Size | 8 |
| Embedding Dimension | 768 |
| Transformer Heads | 8 |
| Transformer Decoder Layers | 6 |

The text-only model demonstrated strong performance on standard NLP metrics (Table VI), indicating its effectiveness in learning to generate relevant textual descriptions from the visual input. Qualitative examples (Fig. 7) show the model

TABLE VI
TEXT-ONLY MODEL: TEXT GENERATION METRICS

| Metric | Value |
|---|---|
| BLEU-1 | 0.9180 |
| BLEU-2 | 0.8863 |
| BLEU-3 | 0.8492 |
| BLEU-4 | 0.8036 |
| METEOR | 0.9279 |
| ROUGE-L | 0.9107 |

producing coherent findings that align well with typical radiological descriptions.

### *B. MedSegText Joint Model Experiments*

This subsection presents the results from the MedSegText framework, where segmentation and text generation tasks were trained jointly. Two experiments with different loss weighting and learning rates are compared.

*1) Experiment 1 (Seg Weight: 1.0, Text Weight: 0.5, LR: 1e-4):* The first joint training experiment prioritized segmentation loss slightly over text generation loss.

TABLE VII
MEDSEGTEXT EXP1: HYPERPARAMETERS

| Parameter | Value |
|---|---|
| Epochs | 50 |
| Learning Rate | 1e-4 |
| Optimizer | Adam |
| Loss Function | Joint Loss (Dice + Cross-Entropy) |
| Segmentation Loss Weight | 1.0 |
| Text Loss Weight | 0.5 |
| Batch Size | 8 |

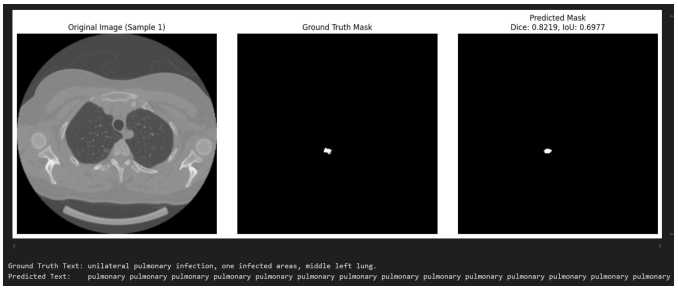


Fig. 8. MedSegText (Exp1): Sample qualitative output showing segmentation (Dice, IoU) and text generation. (Epoch 50, LR 1e-5, Seg Weight 1.0, Text Weight 0.5)

Experiment 1 yielded promising segmentation results (Avg Dice: 0.7473), outperforming the baseline U-Net. However, the text generation performance was significantly degraded compared to the text-only model (e.g., BLEU-4: 0.0035). This suggests that the segmentation task, with its higher effective loss contribution, dominated the learning process, hindering the text decoder's ability to learn effectively.

*2) Experiment 2 (Seg Weight: 1.0, Text Weight: 1.0, LR: 1e-5):* The second experiment aimed to balance the tasks by using equal loss weights and a reduced learning rate to facilitate more stable joint learning.

TABLE VIII
MEDSEGTEXT EXP1: JOINT TASK METRICS

| Metric | Value |
|---|---|
| *Segmentation* | |
| Avg Dice Score | 0.7473 |
| Avg IoU Score | 0.6348 |
| *Text Generation* | |
| BLEU-1 | 0.0189 |
| BLEU-2 | 0.0060 |
| BLEU-3 | 0.0044 |
| BLEU-4 | 0.0035 |
| METEOR | 0.0263 |
| ROUGE-L | 0.0310 |

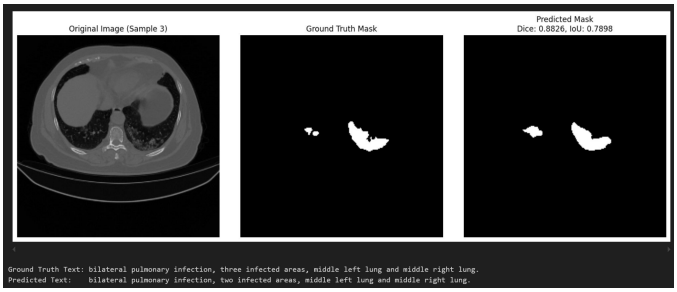


Fig. 9. MedSegText (Exp2): Sample qualitative output showing segmentation (Dice, IoU) and text generation. (Epoch 50, LR 1e-5, Seg Weight 1.0, Text Weight 1.0)

TABLE IX
MEDSEGTEXT EXP2: HYPERPARAMETERS

| Parameter | Value |
|---|---|
| Epochs | 50 |
| Learning Rate | 1e-5 |
| Optimizer | Adam |
| Loss Function | Joint Loss (Dice + Cross-Entropy) |
| Segmentation Loss Weight | 1.0 |
| Text Loss Weight | 1.0 |
| Batch Size | 8 |

TABLE X
MEDSEGTEXT EXP2: JOINT TASK METRICS

| Metric | Value |
|---|---|
| *Segmentation* | |
| Avg Dice Score | 0.7279 |
| Avg IoU Score | 0.6147 |
| *Text Generation* | |
| BLEU-1 | 0.9347 |
| BLEU-2 | 0.9097 |
| BLEU-3 | 0.8803 |
| BLEU-4 | 0.8449 |
| METEOR | 0.9405 |
| ROUGE-L | 0.9292 |

In Experiment 2, by balancing the loss contributions (both weights set to 1.0) and reducing the learning rate to 1e-5, the model achieved a remarkable improvement in text generation performance (e.g., BLEU-4: 0.8449, METEOR: 0.9405), comparable to the text-only model. The segmentation performance (Avg Dice: 0.7279, Avg IoU: 0.6147) remained strong, experiencing only a slight decrease compared to Experiment 1. This indicates that with appropriate hyperparameter tuning, the MedSegText framework can effectively learn both tasks simultaneously, showcasing a significant step towards a synergistic multi-task model. Figure 9 provides qualitative examples from this experiment, illustrating the model's ability to generate both accurate segmentations and relevant textual findings.

## VI. Conclusion

This project undertook a comprehensive exploration into the automated interpretation of chest CT scans, specifically addressing the intricate multi-task challenge of concurrent pulmonary lesion segmentation and the generation of concise radiological findings. The research journey involved the systematic development, implementation, and evaluation of four distinct architectural stages: an initial baseline U-Net for segmentation, an MMI-UNet inspired segmentation model utilizing a ConvNeXt encoder, a RATCHET-inspired Transformer-based model dedicated to text generation, and culminating in the proposed MedSegText framework. This final framework integrated a shared ConvNeXt visual encoder with parallel, specialized decoders for the segmentation and text generation tasks, aiming for a synergistic learning approach.

Our experimental results systematically benchmarked each stage. The MedSegText framework, leveraging the ConvNeXt encoder, demonstrated strong segmentation capabilities, achieving an average Dice score of 0.7473 and an IoU of 0.6348 in its initial joint configuration (Exp1), and a Dice score of 0.7279 with an IoU of 0.6147 in a more balanced configuration (Exp2). The dedicated text generation model, drawing inspiration from the RATCHET architecture, produced highly coherent and contextually relevant findings, achieving excellent n-gram based metrics, including a BLEU-4 score of 0.8036, METEOR score of 0.9279, and ROUGE-L of 0.9107. The primary challenge, characteristic of multi-task learning, lay in effectively balancing the two distinct learning objectives during joint training. The initial joint experiment (Exp1), with a higher relative weight for segmentation loss, maintained strong segmentation but led to a significant degradation in text generation quality (BLEU-4: 0.0035). However, by meticulously adjusting the loss weights to parity and reducing the learning rate (Exp2), we successfully mitigated this issue, achieving substantial improvements in text generation (BLEU-4: 0.8449, METEOR: 0.9405) while retaining commendable segmentation performance (Dice: 0.7279). This demonstrated the critical role of hyperparameter tuning in enabling effective multi-task learning.

The current MedSegText framework, while establishing the feasibility of joint lesion segmentation and radiological finding generation, serves as a foundational stepping stone. The journey highlighted the complexities of multi-task optimization and paved the way for several exciting research directions aimed at enhancing model synergy and clinical applicability. In summary, this project successfully navigated these complexities, and the MedSegText framework represents a significant step towards unified models capable of producing both precise lesion segmentations and clinically relevant textual findings, thereby holding the potential to enhance radiology workflows and support more comprehensive, explainable AI in medical imaging.

## VII. Future Work

Building upon the insights gained from this project, several promising avenues exist to further enhance the MedSegText framework. A primary direction involves implementing more sophisticated cross-modal attention mechanisms directly between the segmentation and text decoders. This would allow for a tighter coupling of the tasks, potentially enabling the segmentation output to more explicitly guide text generation concerning lesion characteristics, and conversely, for textual context to refine segmentation boundaries. Alongside architectural enhancements, exploring dynamic or adaptive loss weighting strategies, rather than static coefficients, could lead to more robust joint training and reduce sensitivity to initial hyperparameter choices.

Furthermore, extending the model's capabilities to generate more structured, hierarchical radiological reports beyond single findings would significantly increase its clinical utility. Incorporating external medical knowledge, such as ontologies or knowledge graphs, could improve the factual accuracy and clinical relevance of the generated text. Finally, continued work on evaluating the model with more semantically-aware text metrics (e.g., BERTScore) and through qualitative assessment by radiologists, alongside training on larger and more diverse datasets, will be crucial for validating and improving the system's real-world applicability and robustness. These future steps aim to evolve MedSegText into a more powerful, interpretable, and clinically impactful tool for automated CT scan analysis.

## References


[1] S.P. Morozov, A.E. Andreychenko, N.A. Pavlov, A.V. Vladzymyrskyy, N.V. Ledikhova, V.A. Gombolevskiy, I.A. Blokhin, P.B. Gelezhe, A.V. Gonchar, and V.Yu. Chernina, "MosMedData: Chest CT Scans With COVID-19 Related Findings Dataset," *Medical Physics*, vol. 47, no. 11, pp. 6137–6144, 2020.

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

[3] P.-N. Bui, D.-T. Le, and H. Choo, "Visual-Textual Matching Attention for Lesion Segmentation in Chest Images," in *Medical Image Computing and Computer Assisted Intervention – MICCAI 2023*, 2023.

[4] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A convnet for the 2020s," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022.

[5] B. Boecking, N. Usuyama, S. Bannur, D. C. Castro, A. Schwaighofer, S. Hyland, M. Wetscherek, T. Naumann, A. V. Nori, J. Alvarez-Valle, *et al.*, "Making the most of text semantics to improve biomedical vision-language processing," in *Eur. Conf. Comput. Vis.*, 2022

[6] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "Eca-net: Efficient channel attention for deep convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020.
[7] A. Vaswani et al., "Attention Is All You Need," in *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, 2017, pp. 5998–6008.
[8] B. Hou, G. Kaissis, R. M. Summers, and B. Kainz, "RATCHET: Medical Transformer for Chest X-ray Diagnosis and Reporting," in *Medical Image Computing and Computer Assisted Intervention – MICCAI 2021*, 2021
[9] G. Reale-Nosei, E. Amador-Domínguez, and E. Serrano, "From vision to text: A comprehensive review of natural image captioning in medical diagnosis and radiology report generation," *Medical Image Analysis*, vol. 97, p. 103264, 2024.
[10] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a Method for Automatic Evaluation of Machine Translation," in *Proc. 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2002, pp. 311–318.
[11] A. Lavie and A. Agarwal, "METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments," in *Proc. ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization*, 2005, pp. 228–231.
[12] C.-Y. Lin, "ROUGE: A Package for Automatic Evaluation of Summaries," in *Proc. ACL Workshop on Text Summarization Branches Out*, 2004, pp. 74–81.
[13] A. Paszke et al., "PyTorch: An Imperative Style, High-Performance Deep Learning Library," in *Advances in Neural Information Processing Systems 32 (NeurIPS 2019)*, 2019, pp. 8024–8035.
[14] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*. O'Reilly Media, Inc., 2009.
[15] I. Loshchilov and F. Hutter, "Decoupled Weight Decay Regularization," in *Proc. 7th Int. Conf. Learn. Represent. (ICLR)*, 2019.