NOVA IMS Information Management School

MDSAA

Master Degree Program in

Data Science and Advanced Analytics

MULTIMODAL LEARNING FOR LUNG CANCER DIAGNOSIS AND MANAGEMENT

A Deep Learning Pipeline for Classification, TNM Staging, and Treatment Protocol Generation

Catarina Costa Pereira Nascimento da Silva

Master Thesis

presented as partial requirement for obtaining a Master's Degree in Data Science and Advanced Analytics

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

Multimodal Learning for Lung Cancer Diagnosis and Management

A Deep Learning Pipeline for Classification, TNM Staging, and Treatment Protocol Generation

by

Catarina Costa Pereira Nascimento da Silva

Master Thesis presented as partial requirement for obtaining the Master's degree in Data Science and Advanced Analytics, with a specialization in Data Science.

Supervised by

Mauro Castelli, PhD, NOVA Information Management School

Bruno Jardim, PhD, NOVA Information Management School

July, 2025

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration. I further declare that I have fully acknowledged the Rules of Conduct and Code of Honor from the NOVA Information Management School.

[Lisbon, 15 of July of 2025]

ACKNOWLEDGMENTS

I would like to express my deep gratitude to my two supervisors, Professor Mauro Castelli and Professor Bruno Jardim, who were always available and offered invaluable support throughout this project. From the very beginning, their guidance made all the difference. Without their involvement, this work would not have been possible. I am especially thankful to both professors for believing in this project and in me every step of the way.

To my parents, who have unconditionally supported me and believed in me for as long as I can remember, thank you for making all this possible. Your faith in me, your strength, and your constant encouragement made every dream attainable. Thank you for celebrating every accomplishment of mine as if it were your own. Your belief in me and your love have meant more than words can express. To my mother, thank you for your tenderness and patience, especially during the moments when I felt everything was falling apart. To my father, I am deeply grateful for your constant care. Even without fully understanding the technical details of this work, you always searched for a thousand ways to help me solve every problem. You read, learned, and took an interest just to be there for me, and even when I doubted myself, you never did.

With all my heart, I thank Francisco - my boyfriend, my greatest support, and the one who stood by me through every doubt, always being my constant source of love and strength. Your unwavering support has meant everything to me. You patiently listened as I shared my worries for days on end, never letting me lose confidence in myself. Through countless hours of stress, you were always there to take care of me and give me the strength I couldn't find on my own. Thank you for always believing I could do anything, and most of all, for being by my side in every moment I needed you. I hope you'll always be by my side, and that I'll always have the chance to be by yours.

A special thank you to my brothers, David and Pedro, whom I care for so deeply. You've always been by my side whenever I needed you, offering support, laughter, and love in your own way. I am so grateful to have you both in my life, and your trust in me continues to inspire every path I follow.

Thank you also to my dear friends Bia and Luisa, who supported me through every difficult moment and listened endlessly to my thoughts and concerns. Your presence meant more than I can express.

I am also grateful to all my other friends and family members who have been, and continue to be, an incredible source of strength and support.

Finally, i would like to thank all the professors at Nova IMS who have been part of my academic journey. Your knowledge, guidance, and encouragement have made this work possible.

ABSTRACT

Accurate lung cancer diagnosis and staging are critical for personalized treatment and improved patient outcomes. Traditional diagnostic methods, such as manual image examination, are prone to variability and inefficiency. Recent advancements in Deep Learning (DL) have demonstrated potential in automating lung cancer classification and staging, thereby enhancing diagnostic accuracy and efficiency. However, existing solutions often address only isolated aspects of the diagnostic process, such as tumor detection, without offering a unified system for multi-target classification that integrates clinical tools for both clinicians and patients. This study presents a unified framework for lung cancer analysis that combines medical imaging, clinical data, and Large Language Models (LLMs) to support three key tasks: tumor type classification, TNM staging, and automated treatment protocol recommendation. Image-based classification was performed using YOLOv8n, trained on two CT datasets, achieving a maximum mean average precision (mAP50) of 0.418 and an F1 score of 0.44. TNM staging was addressed through a multimodal classifier, combining the ResNet50 model with a multilayer perceptron, which fused imaging and demographic inputs. This approach yielded an average F1 score of 0.389, with the M component showing the strongest performance. For treatment generation, a Retrieval-Augmented Generation (RAG) approach was employed, combining clinical prompts with relevant documents to produce personalized protocols using the Gemini 2.0 Flash LLM. The best-performing configuration achieved a stage match of 0.54 and a BERTScore of 0.83, along with high contextual fidelity across RAGAS metrics (Faithfulness: 0.68, Answer Relevancy: 0.81, Context Precision: 0.99, and Context Recall: 0.93). This framework demonstrates the potential to support both diagnosis and treatment planning within a single integrated system, contributing to more personalized and effective clinical decision-making in oncology.

KEYWORDS

Deep Learning (DL); Large Language Models (LLMs); Lung Cancer; Retrieval-Augmented Generation (RAG); Treatment Recommendation

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

ACS	American Cancer Society
ADC	Adenocarcinoma
AJCC	American Joint Committee on Cancer
AUC	Area Under the Curve
BERT	Bidirectional Encoder Representations from Transformers
CAD	Computer-Aided Diagnosis
CNN	Convolutional Neural Network
CoT	Chain of Thought
CT	Computed Tomography
DCNN	Deep Convolutional Neural Network
DDx	Differential Diagnosis
DL	Deep Learning
DNN	Deep Neural Network
DenseNet	Densely Connected Convolutional Network
EHR	Electronic Health Record
ESMO	European Society for Medical Oncology
GAI	Generative Artificial Intelligence
GPT	Generative Pre-trained Transformer
GTV	Gross Tumor Volume
HGG	High-Grade Glioma
ICD	International Classification of Diseases
KDIGO	Kidney Disease: Improving Global Outcomes
LCC	Large Cell Carcinoma
LGG	Low-Grade Glioma
LLM	Large Language Model

LLaMa	Large Language Model Meta
MAE	Mean Absolute Error
mAP	Mean Average Precision
Med-PaLM	Medical Pathways Language Model
MIMIC-III	Medical Information Mart for Intensive Care III
ML	Machine Learning
MPS	Metal Performance Shaders
NCCN	National Comprehensive Cancer Network
NCI	National Cancer Institute
NLP	Natural Language Processing
NIH	National Institutes of Health
NOS	Not Otherwise Specified
NSCLC	Non-Small Cell Lung Cancer
OG-RAG	Ontology-Grounded Retrieval-Augmented Generation
PCA	Principal Component Analysis
PET-CT	Positron Emission Tomography - Computed Tomography
RAG	Retrieval-Augmented Generation
ReLU	Rectified Linear Unit
RLHF	Reinforcement Learning from Human Feedback
RNN	Recurrent Neural Network
ROI	Region of Interest
ROUGE	Recall-Oriented Understudy for Gisting Evaluation
RTSTRUCT	Radiotherapy Structure Sets
ResNet	Residual Network
SCC	Squamous Cell Carcinoma
SCLC	Small Cell Lung Cancer

TCIA	The Cancer Imaging Archive
TNM	Tumor, Node, Metastasis
WHO	World Health Organization
WSI	Whole Slide Image
YOLO	You Only Look Once

1. INTRODUCTION

1.1. BACKGROUND AND RATIONALE

Lung cancer remains a leading cause of cancer-related deaths globally, with incidence and mortality rates reflecting its impact on public health (Bray et al., 2022). In the United States, lung cancer is expected to cause approximately 125,070 deaths in 2024 (Siegel, Giaquinto, & Jemal, 2024), accounting for about 18% of all cancer-related mortality. The high mortality rate associated with lung cancer is mainly due to late diagnosis, which restricts treatment options and, therefore, reduces survival rates (Siegel, Giaquinto, & Jemal, 2024). Although advances in early detection and treatment have contributed to a decline in lung cancer mortality over the past several decades, it still results in more deaths annually than colorectal, breast, and prostate cancers combined (Bray et al., 2022; Siegel, Giaquinto, & Jemal, 2024), underscoring the need for improvements in both early diagnosis and treatment efficacy.

Early diagnosis of lung cancer significantly improves survival outcomes (Huang et al., 2019); however, traditional diagnostic approaches, which are predominantly based on manual examination of imaging data, present several limitations. Traditional methods of lung cancer screening, such as chest radiography and visual CT scan analysis, are proven to be time-consuming and depend heavily on radiologists' expertise, which may vary and can lead to inconsistencies in diagnosis (Huang et al., 2019; El-Baz et al., 2013). Furthermore, traditional methods may fail to detect subtle indicators of early-stage cancer, particularly when examining large imaging volumes or when nodules are small and lack well-defined boundaries (El-Baz et al., 2013; Atmakuru et al., 2024).

The application of Deep Learning (DL) to lung cancer diagnosis presents a potential solution to several of these limitations (Atmakuru et al., 2024). DL algorithms can automate the analysis of large volumes of imaging data, identifying features that may be difficult to detect through manual examination, thereby enhancing diagnostic precision (El-Baz et al., 2013). Studies have shown (Atmakuru et al., 2024; Huang et al., 2019) that deep learning models can detect lung nodules with a sensitivity and accuracy comparable to that of radiologists, while also providing faster analysis. Beyond detection, DL models have shown high accuracy in classifying lung cancer subtypes, which are essential for determining disease prognosis and treatment strategies (Davri et al., 2023; Silva et al., 2022). Integrating DL into pathology improves clinical workflow by automating tasks that traditionally require manual annotation. Bray et al. (2022) and Davri et al. (2023) note that by enabling early detection, subtype classification, and workflow automation, DL significantly contributes to advances in lung cancer diagnosis. They also discuss that its integration into clinical and pathology workflows has the potential to reduce lung cancer mortality and improve treatment planning.

1.2. PROBLEM STATEMENT AND RESEARCH GAP

Despite recent advancements (Atmakuru et al., 2024; Huang et al., 2019, Bray et al., 2022, Davri et al., 2023), existing deep learning solutions in lung cancer diagnosis often address isolated aspects, such as tumor detection or staging, rather than providing a unified system. Current methods frequently neglect the integration of patient clinical information and tools for direct clinician and patient use. For instance, while some studies explore multi-modality fusion for image segmentation (Zhou et al., 2019), few incorporate demographic data or clinical history, which are crucial for personalized diagnostics. Additionally, research efforts like those by Ke et al. (2021) and Coudray et al. (2018) focus on interpreting imaging data or predicting genetic mutations but fail to deliver reporting systems or interactive tools to further engage patients in the diagnostic and treatment process. This gap highlights the need for a solution that integrates imaging data with patient demographic information to enhance model performance while improving patient understanding of their condition, fostering better engagement. Additionally, there is also a need to incorporate advanced methodologies to enhance the interpretability of these models, ensuring their practical application in clinical settings.

The incorporation of Large Language Models (LLMs) into oncology expands on DL advancements, particularly in medical imaging and patient communication, addressing the gaps in diagnostic workflows. LLMs (Jia et al., 2024; Geantă et al., 2024), have been shown to improve doctor-patient communication by translating complex medical information into accessible language while personalizing communication strategies to align with patient needs. Together, LLMs and DL enhance diagnostic precision and make medical knowledge more accessible, supporting informed decisions by simplifying complex medical information (Jia et al., 2024; Ahmed et al., 2022).

1.3. RESEARCH AIM AND OBJECTIVES

This thesis aims to leverage DL and LLMs by combining them into a unified system creating an interpretable and efficient diagnostic framework specific to lung cancer detection and management. By addressing some of the main limitations in traditional diagnostic methods, these technologies provide more effective, accessible tools for early detection, accurate diagnosis, and efficient workflows in lung cancer care (Jia et al., 2024; Ahmed et al., 2022), improving patient outcomes with the aim of reducing the global impact of lung cancer. To achieve this, the research focuses on the following objectives:

1. **Develop a Multi-Target Image Classification System**: Develop and implement a DL model capable of accurately classifying lung cancer types and clinical stages—Tumor, Node, Metastasis (T, N, M)—using CT scans from The Cancer Imaging Archive (TCIA) (Clark et al., 2013). Leverage techniques such as YOLO for rapid and precise tumor localization and classification, and ResNet for TNM stage prediction.

2. **Integrate Imaging and Clinical Data**: Utilize multi-modality fusion techniques to combine imaging features with patient information, such as demographic data, to improve model performance and diagnostic accuracy in predicting TNM cancer stages.

3. **Create a Personalized Diagnostic Reporting System**: Utilize LLMs to predict the overall cancer stage based on the T, N, and M classifications, and generate diagnostic reports that include detailed explanations of the treatment protocol. These reports should be tailored to the patient's demographic profile, cancer type, and stage, and designed to be understandable to both clinicians and patients.

1.4. RESEARCH CONTRIBUTIONS

Building on these objectives, this research expects to contribute to the field of lung cancer diagnosis and management by introducing a framework that integrates DL and LLMs. It seeks to address the limitations of traditional methods by providing an interpretable and efficient solution for multi-target classification, including tumor type and clinical staging. Additionally, it incorporates personalized reporting through fine-tuned LLMs to generate diagnostic summaries and treatment plans, aiming to improve communication and patient engagement. The overall goal of the research is to improve diagnostic accuracy, workflow efficiency, and accessibility, reducing healthcare disparities, and contributing to better patient outcomes, with the final aim of reducing the global impact of lung cancer.

1.5. THESIS STRUCTURE

This thesis is organized into six chapters, each addressing a distinct aspect of the research.

- **Chapter 1** (Introduction) defines the research context, problem, objectives, and contributions.
- **Chapter 2** (Literature Review) covers clinical aspects of lung cancer, limitations of current diagnostic methods, and the use of DL and LLMs in staging and treatment planning.
- **Chapter 3** (Methodology) describes the datasets, preprocessing steps, models, and evaluation metrics used for tumor image classification, TNM staging, and treatment recommendation.
- **Chapter 4** (Results/Discussion) presents the outcomes of the experiments, including classification performance, staging accuracy, and treatment generation quality.
- **Chapter 5** (Limitations and Future Works) outlines the main constraints encountered during the study and proposes directions for improvement.
- **Chapter 6** (Conclusion) summarizes the findings and highlights the configurations that showed the most reliable results across tasks.

2. LITERATURE REVIEW

This chapter reviews literature from 2016 to 2025, with a primary focus on studies from 2018 onward, examining lung cancer, DL, and LLMs in medical diagnostics, treatment planning, and oncology. The sources were primarily retrieved from peer-reviewed journals, the arXiv preprint repository, and academic publishers such as Elsevier and IEEE as well as platforms like PubMed and Google Scholar. The first section gives an overview of lung cancer staging systems, such as the TNM classification, and explains the clinical importance of accurate diagnosis. It identifies the limitations of traditional diagnostic methods, including challenges in accuracy and scalability, highlighting the need for advanced tools like DL and LLMs.

The second section explores DL applications, particularly convolutional neural networks (CNNs), in medical imaging. The studies reviewed demonstrate the potential of models like YOLO (Muhammad Yaseen, 2024) to improve diagnostic precision. Evaluation of DL scalability and applicability is discussed, with specific relevance to lung cancer staging and diagnosis.

The third section addresses LLMs in healthcare, focusing on their use in diagnostic support and treatment planning through techniques like prompt engineering and retrieval-augmented generation (RAG). The chapter concludes by synthesizing findings, identifying gaps in integrated DL-LLM approaches, and establishing the research foundation for the subsequent chapters.

2.1. LUNG CANCER

Lung cancer is primarily categorized into two major histological groups (American Cancer Society, 2024): non-small cell lung cancer (NSCLC) and small cell lung cancer (SCLC). NSCLC accounts for approximately 85% of all lung cancer cases and is further subtyped into adenocarcinoma (ADC), squamous cell carcinoma (SCC), and large cell carcinoma (LCC). On the other hand, SCLC comprises about 10-15% of cases and is known for its aggressive growth and early metastatic spread.

Among NSCLC subtypes, ADC is the most prevalent. This subtype is more common in women and younger individuals and is frequently observed in both smokers and non-smokers. SCC, often linked to smoking, typically originates in the central regions of the lung. LCC is less common but notable for its fast progression and poor prognosis (American Cancer Society, 2024). SCLC, while less prevalent, is characterized by a fast metastatic progression and a high initial response rate to chemotherapy and radiation therapy. However, recurrence is common, limiting long-term survival outcomes (American Cancer Society, 2024). Advancements in molecular pathology emphasize the heterogeneous nature of lung cancer (Inamura, 2017), even within the same histological subtypes, highlighting the complexity of this disease.

2.1.1. Staging Systems (TNM classification)

Lung cancer staging follows the TNM (Tumor, Node, Metastasis) classification, which describes

the tumor's anatomic extent based on three components (International Association for the Study of Lung Cancer [IASLC], 2024): T (extent of the primary tumor), N (lymph node involvement), and M (presence of distant metastases) (see Table II.1, Annex II). The TNM system is universally applied to both NSCLC and SCLC; however, it is only applied to tumors following a definitive lung cancer diagnosis. Each component is further divided into categories and subcategories based on specific descriptors that define the size, extent, and spread of the tumor.

2.1.2. Lung Cancer Staging

Lung cancer staging (National Cancer Institute, 2025; American Cancer Society, 2025) is a process that determines the extent of disease spread and guides treatment decisions. For NSCLC, staging follows the TNM system. These components are combined to assign an overall stage from I to IV (see Table II.2, Annex II), with early stages indicating localized disease and later stages reflecting advanced or metastatic cancer. In contrast, SCLC is typically staged using a simpler two-tier system: limited stage (confined to one hemithorax and suitable for a single radiation field) and extensive stage (disease that has spread beyond the thorax). Although some recent guidelines (American Cancer Society, 2025) also allow TNM staging for SCLC, the traditional two-stage system remains more widely used in clinical practice. Treatment strategies are closely tied to these staging systems, as early-stage NSCLC may be managed with surgery and adjuvant therapies, while advanced stages require systemic treatment. Similarly, SCLC treatment protocols vary significantly between limited and extensive stages, underlining the importance of accurate and stage-specific classification.

2.1.3. Impact of Correct Diagnosis

Accurate diagnosis, staging, and grading are crucial in the management of lung cancer, as they directly influence treatment strategies and patient outcomes (Navani et al., 2019; Silva et al., 2022). Early detection is especially critical, given that most lung cancer cases are diagnosed at advanced stages, limiting treatment options and resulting in poor survival rates. Over 75% of patients present with stage III or IV disease, correlating to a five-year survival rate of less than 5%, compared to approximately 60% for those diagnosed with stage IA (Nooreldeen & Bach, 2021). Delays in diagnosis often lead to disease progression, resistance to standard therapies, and reduced chances for curative interventions.

Therefore, the TNM classification has a significant impact on lung cancer management by reflecting the disease's anatomical extent, thereby guiding treatment strategies and survival predictions (International Association for the Study of Lung Cancer [IASLC], 2024). As discussed by Silva et al. (2022) and Nooreldeen and Bach (2021), accurate staging determines appropriate treatment, such as surgery for early-stage disease, chemoradiotherapy for advanced stages, and for optimizing patient outcomes. Misclassification in lung cancer staging can result in insufficient treatment or exclusion from potentially curative surgical options, thereby compromising outcomes. Navani et al. (2019) demonstrated that clinically

understaged patients often experience worse survival rates due to inadequate preoperative assessments that fail to detect metastatic disease.

Tumor grading further complements staging by providing insights into cancer cell aggressiveness, with higher grades indicating faster growth and a greater likelihood of metastases. This information, when integrated with cancer stage, genetic markers, patient age, and overall health, supports the development of personalized treatment plans. High-grade tumors often require immediate and aggressive interventions to avoid rapid disease progression. Furthermore, tumor grade is a vital prognostic indicator, aiding clinicians and patients in understanding the disease's trajectory and optimizing management strategies (National Cancer Institute, 2022). Together, accurate staging and grading ensure that treatment approaches are both precise and personalized, ultimately improving patient outcomes.

2.2. Limitations of Traditional Diagnostic Approaches

Traditional visual analysis of scans depends highly on radiologist expertise, which introduces variability and limits scalability. Radiologists often focus on specific regions of interest (ROIs), neglecting the complete chest volume, increasing the risk of missed malignancies (Huang et al., 2019). Additionally, visual interpretation lacks sensitivity to subtle nodular changes that may signify early-stage cancer. The subjectivity of the manual analysis and evaluation constrains the reproducibility and reliability of diagnostic outcomes.

Furthermore, traditional methods are hindered by technical and operational challenges. For example, standard diagnostic tools like chest radiography and sputum cytology, which is a medical test used to examine cells found in mucus and other material coughed up from the lungs, have shown limited efficacy in clinical trials and are not suitable for mass screening (Nooreldeen & Bach, 2021). The absence of automated and standardized criteria for risk assessment leads to inconsistencies, while manual image analysis consumes excessive time, reducing overall efficiency. These limitations collectively decrease the effectiveness of traditional approaches in early lung cancer detection, highlighting the need for more advanced and scalable tools, such as DL.

2.3. Deep Learning and Convolutional Neural Networks

DL is a subfield of machine learning that leverages hierarchical architectures to progressively process data through multiple layers, each extracting complex and abstract features or patterns (Guo et al., 2016). Unlike traditional machine learning methods, which often rely on manual feature extraction, DL automates this process using deep neural networks (DNNs). These networks consist of multiple processing layers that allow the modeling of complex, non-linear functions (Shrestha & Mahmood, 2019). The foundational structure of a neural network aims to mirror the human nervous system, with interconnected nodes organized across input, hidden, and output layers. Each of these nodes processes data through weighted connections,

and activation functions such as ReLU or sigmoid are applied to enable the learning of patterns within the data (Shrestha & Mahmood, 2019).

Guo et al. (2016) suggest that the fast growth and widespread adoption of DL is due to three key factors: the exponential increase in computational power (e.g., GPUs), the reduction in hardware costs, and the significant advancements in machine learning algorithms. Its applications extend across a wide range of fields, including natural language processing (NLP), predictive analytics, and particularly, computer vision, a field where DL has enabled significant advancements in recent years (Guo et al., 2016).

Among the different architectures in DL, CNNs have become essential in computer vision tasks. CNNs are specialized feedforward neural networks designed to process image data. Unlike traditional methods that rely on manually extracted features (Li et al., 2022), CNNs extract features automatically through convolutional operations, improving learning efficiency.

The architecture of a CNN, as shown in Figure 2.1, is inspired by the biological visual system, where artificial neurons simulate the function of biological ones. Li et al. (2022) explain that the key architectural elements such as local connections, weight sharing, and dimensionality reduction through pooling layers (Figure 2.2) contribute to CNNs' efficiency and adaptability. These mechanisms allow CNNs to reduce the number of parameters, focus on the most relevant features, and speed up the learning process during training.

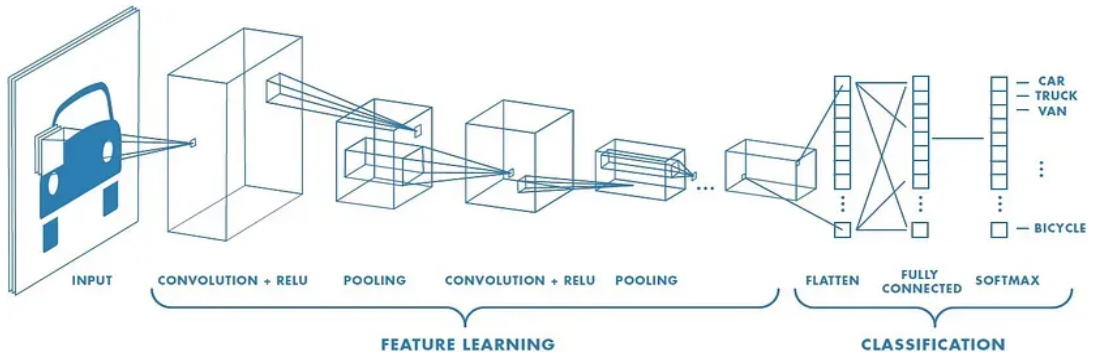


Figure 2.1 - Example of the architecture of a CNN (Towards Data Science, 2020)

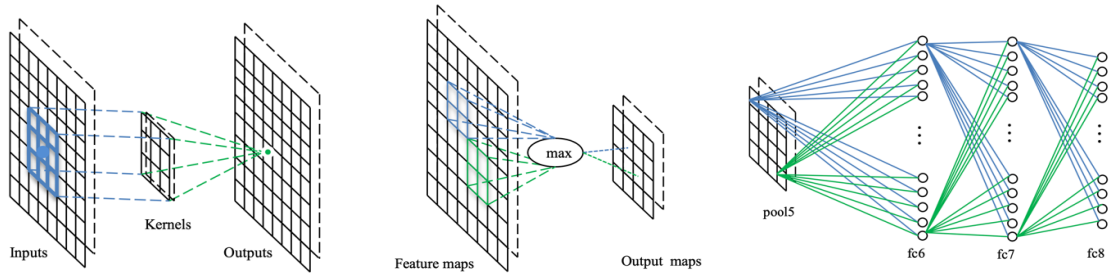


Figure 2.2 - Operations of a CNN architecture performed at each stage in sequence (Guo et al., 2016) - (a) operation of the convolutional layer; (b) operation of the max pooling layer; (c) operation of the fully-connected layer.

CNNs have shown exceptional performance in various computer vision tasks, including image classification, object detection, and semantic segmentation, largely due to their ability to learn and generalize from hierarchical representations of data. The ability of CNNs to automatically identify and focus on relevant image features has improved their reliability as one of the most used DL algorithms (Guo et al., 2016).

2.3.1. The Potential of Deep Learning in Healthcare

In the medical field, CNNs have demonstrated a significant impact, especially in radiology. They have achieved considerably high performance (Yamashita et al., 2018), even compared to professionals, in several medical fields such as skin lesion classification, lymph node metastasis detection, among many others. Beyond these, CNNs are widely used for detecting, classifying, and segmenting lesions in medical imaging, making them invaluable tools for tasks like the classification of lung nodules into benign or malignant categories using CT scans. Their ability to automate and enhance traditional diagnostic processes has drastically reduced the time and effort required for manual analysis while improving diagnosis accuracy and consistency (Yamashita et al., 2018).

The use of DL has been shown to address some of the main challenges in traditional diagnostic methods, such as variability in manual analysis and the inability to process large volumes of data efficiently (Malathy et al., 2024; Wang et al., 2019). By enabling early detection and personalized treatment, DL has shown its potential across various medical fields (Atmakuru et al., 2024; Javed et al., 2024), including oncology, cardiology, neurology, and ophthalmology.

One of the most impactful contributions of DL is its role in image segmentation and interpretation. Malathy et al. (2024) highlight that DL-enhanced image segmentation significantly improves the identification and delineation of anatomical structures. Using CNNs trained on annotated datasets, DL surpasses traditional manual segmentation methods, which are often slow and inconsistent. This technology can provide radiologists with fast and precise insights, aiding in diagnoses such as tumor boundary detection and brain lesion analysis.

As discussed previously, DL's ability to address limitations inherent in traditional diagnostic methods is another of its strengths. Atmakuru et al. (2024) highlight the capacity of DL models to process intra-nodular heterogeneity within lung nodules, providing a more detailed understanding of complex medical details. This understanding facilitates accurate diagnoses, allowing radiologists to make informed decisions. Additionally, DL automates complex visual analyses, reducing diagnostic times and enhancing the consistency of results while enabling clinicians to focus on more complex cases that require their expertise.

The field of cancer diagnostics, particularly lung cancer, has significantly benefited from all these advancements in DL. Wang et al. (2019) demonstrate that DL facilitates tasks such as tumor region identification, prognosis prediction, and microenvironment characterization. These advancements help improve diagnostic precision and efficiency. Building on this, Javed

et al. (2024) also show how techniques such as Deep Convolutional Neural Networks (DCNNs) automate disease classification and staging.

Expanding furthermore on these contributions, Davri et al. (2023) detail the high performance of advanced DL models like Deep Embedding-based Logistic Regression (DELR) and EfficientNet-B3, which can achieve high accuracy (AUC > 0.95) across diverse datasets. These models enable the classification of lung cancer subtypes, such as ADC, SCC, and SCLC, which, as mentioned earlier, is essential to determining the appropriate course of treatment.

Together, these advancements underscore the huge impact DL has been having in the past several decades on healthcare and especially on medical diagnosis. DL's models (Javed et al., 2024; Davri et al., 2023; Wang et al., 2019) have been shown to overcome the challenges of feature selection and variability in imaging data, delivering high accuracy on diagnosis, highly impacting early intervention and treatment planning.

Table 2.1 - Summary of the findings from the studies highlighting the impact of DL in healthcare

Author(s) and Year	Objective	Conclusions	Methods
Yamashita et al. (2018)	Overview of CNNs and their application in radiology, highlighting the basic concepts, methodologies involved, and impact.	CNNs have achieved remarkable success in various fields, particularly in medical research and radiology. DL has become a leading method for complex tasks like image classification and object detection	The paper discusses the application of CNNs in radiology, highlighting their ability to learn spatial hierarchies of features without the need for manual feature extraction techniques.
Malathy et al. (2024)	Discuss the recent advancements in enhancing the accuracy and stability of image segmentation in medical diagnostics through DL techniques, using architectures like U-Net and CNNs.	DL, particularly CNNs, significantly enhances the accuracy and efficiency of medical image segmentation, allowing for precise identification of anatomical features and pathological regions.	Compare different DL architectures on different datasets are selected based on the complexity of the segmentation task. (accuracy of the models (%) between 88.3 and 93.2)
Wang et al. (2019)	Provide an overview of current and potential	DL methods offer significant advantages, such as simplified feature definition,	The paper discusses various image processing and

Author(s) and Year	Objective	Conclusions	Methods
	applications for DL methods in lung cancer pathology image analysis.	enhanced recognition of complex objects, and time efficiency, over traditional learning methods in analyzing pathology images, particularly in lung cancer.	machine learning (ML) methods, with a focus on DL techniques for lung cancer image analysis.
Atmakuru et al. (2024)	Provide a systematic review of DL techniques specifically applied to lung cancer diagnostics, focusing on classification, segmentation, and predictive modeling.	The review highlights the strengths and limitations of various DL models, including CNNs, DNNs, and transfer learning, while addressing challenges like reproducibility and the need for multimodal data integration in LC diagnostics.	The paper analyzes 153 selected studies from an initial pool of 589 publications. It categorizes the studies based on imaging modalities, DL model types, and practical applications.
Javed et al. (2024)	Present a systematic literature review on the application of DL techniques, particularly CNNs, for the detection and classification of lung cancer.	DCNN consistently outperforms other algorithms in terms of accuracy, sensitivity, specificity, and precision across various imaging modalities, including CT scans and MRIs. It underscores the potential of DL to improve lung cancer detection and classification.	The paper explores the literature on lung cancer classification using DL approaches. The study analyzes various medical imaging modalities and evaluates the performance of different DL algorithms.
Davri et al. (2023)	Present a systematic review of DL methods for lung cancer diagnosis, prognosis, and prediction using histological and cytological images.	High-performance rates, with some models achieving accuracy over 97% and AUC values exceeding 0.95 across multiple datasets. Superiority of DL methods over traditional ML techniques, particularly in feature extraction and classification tasks.	Systematic review, following PRISMA guidelines, identified 357 articles, of which 96 met the eligibility criteria, using specific algorithms to filter relevant studies.

2.3.2. YOLO and Neural Network-Based Models for Medical Diagnosis

Among the different deep learning models, "You Only Look Once" (YOLO) has emerged as a significant development in object detection, bridging the gap between detection efficiency and real-time application. YOLO (Redmon et al., 2016), introduced in 2016 by Joseph Redmon,

is an object detection model that treats detection as a regression problem, avoiding the complexities of multi-stage pipelines. Its ability to process full images in a single evaluation while simultaneously predicting bounding boxes and class probabilities positions it as a robust tool for real-time applications in diverse fields, including medicine (Redmon et al., 2016).

The YOLO framework excels in generalizability, enabling effective application in new domains such as tumor localization in medical imaging. This capability is derived from YOLO's efficient encoding of contextual and spatial information, resulting in fewer false detections compared to conventional detection systems like Fast R-CNN (Redmon et al., 2016). Integration of YOLO into medical imaging can leverage its high speed and accuracy. The model's ability to process up to 45 frames per second for its base version, and 155 frames per second for its faster variants (Redmon et al., 2016), makes it particularly suited for applications requiring real-time analysis, such as the identification and classification of lung cancer subtypes.

Recent advancements in the YOLO architecture, as demonstrated by Wehbe et al. (2024), specifically YOLOv8, have significantly contributed to tumor localization and classification. In this paper, YOLOv8 was utilized for identifying lung cancer subtypes, including SCC, ADC, and SCLC, achieving exceptionally high-performance metrics. The use of this model enabled a precise and fast identification of these subtypes in CT images, with a mean Average Precision (mAP) of 97.1%. The model's high precision of 96.1% and its detection speed of 0.22 seconds per image further emphasize its suitability for real-time medical applications. The study also integrated YOLOv8 with a TNMClassifier, a neural network designed for the TNM stage classification. The features extracted from YOLOv8 were reduced using Principal Component Analysis (PCA) to enhance efficiency and fed into the classifier, and the TNMClassifier achieved 98% accuracy in staging. This combined approach, validated on additional datasets with a recall of 0.91, underscores its potential to enhance lung cancer detection and staging, improving diagnostic speed and accuracy.

Building on the advancements of YOLO architectures applied to oncology, Elshahawy et al. (2023) also developed a hybrid DL model combining YOLOv5 and ResNet50 to address the limitations in melanoma detection, such as the small number of classes used, low segmentation accuracy, and overfitting issues in earlier studies. YOLOv5 was employed as the primary framework for detecting and classifying skin lesions, utilizing its end-to-end convolutional architecture to predict both the class and position of lesions. ResNet50 was integrated into the system to address gradient explosion issues and improve classification across the seven lesion classes. Its role as the image classification network allowed the model to handle the complexity of multiple scales and class similarities within dermoscopic images. The model's methodology included preprocessing, hyperparameter optimization, and classification. Features were extracted using YOLOv5 and ResNet50, with bounding boxes and probability scores used for lesion classification. The system achieved high-performance metrics, including 99.0% precision, 98.6% recall, 99.5% accuracy, and mAP scores of 98.3%

(mAP@0.5) and 98.7% (mAP@[0.5-0.95]). It processed images in 0.4 milliseconds, outperforming traditional methods in efficiency and accuracy.

Similarly, Elazab et al. (2024) introduced a hybrid DL model that integrates YOLOv5 and ResNet50 to enhance the classification and grading of gliomas using histopathological Whole Slide Images (WSIs), which are high-resolution digital scans. YOLOv5 was used to localize tumors within large WSIs by providing bounding boxes and initial classifications, while ResNet50 was incorporated into the YOLOv5 framework to improve feature extraction. This integration leveraged YOLOv5's strength in efficient and precise tumor detection and ResNet50's capability to handle complex patterns and high-dimensional data. Additionally, the inclusion of ResNet50 helped stabilize training dynamics and addressed issues like gradient explosion, ensuring a more robust performance during training. The model further incorporated an extreme gradient boosting (XGBoost) classifier to grade gliomas into four categories, effectively distinguishing between low-grade gliomas (LGG) and high-grade gliomas (HGG). The hybrid framework achieved considerably high performance, with 97.2% accuracy, 97.8% precision, 98.6% sensitivity, and a Dice similarity coefficient of 97%. These results outperform existing methods, highlighting the effectiveness of combining YOLOv5's detection capabilities with ResNet50's feature extraction for tumor classification and grading.

Table 2.2 - Summary of the findings from studies combining YOLO and Neural Network-Based Models for Medical Diagnosis

Author(s) and Year	Objective	Methodology	Performance metrics	Findings
Wehbe et al. (2024)	Developing a method for lung cancer subtype classification (SCC, ADC, SCLC) and TNM staging.	YOLOv8 + TNMClassifier (NN) + PCA (for feature reduction).	YOLO: Precision = 96.1%, Recall = 0.91, mAP = 97.1%, detection speed of 0.22 seconds TNMClassifier: Accuracy = 98%	The system enhances lung cancer detection and staging, improving diagnostic speed and accuracy.
Elshahawy et al. (2023)	Develop a model for early melanoma detection, aimed at improving the accuracy and efficiency of skin lesion classification.	YOLOv5 + ResNet50	Precision = 99.0%, Recall = 98.6%, Accuracy = 99.5%, DSC= 98.8%, mAP@0.5 = 98.3%, mAP@[0.5-0.95] = 98.7%	Effectiveness of the proposed model or lesion classification, achieving high-performance metrics.

Author(s) and Year	Objective	Methodology	Performance metrics	Findings
Elazab et al. (2024)	Develop a hybrid model for localization and predictive grading of brain tumors in histopathological images.	YOLOv5 + ResNet50 + XGBoost (used as a classifier to handle high-dimensional features and non-linear relationships)	Accuracy = 97.2%, Precision = 97.8%, Sensitivity = 98.6%, Dice = 97%	The model successfully identified brain tumors from histopathological images, however, it showed struggles with atypical tumor forms.

2.4. LARGE LANGUAGE MODELS

The development of LLMs builds upon decades of progress in language modeling, which began with statistical approaches and evolved into neural network-based models. Early models (Chang et al., 2023), such as n-grams, estimated word probabilities based on local context but struggled to capture long-range dependencies and complex linguistic structures. Neural language models address these limitations by developing universal representations (Zhao et al., 2024), which capture general language patterns. These representations can then be adapted to specific tasks, allowing pre-trained models to learn context-aware features for a wide range of applications.

The leap to modern LLMs was driven by groundbreaking advances in ML, particularly DL techniques. DNNs, with their ability to learn complex patterns from data, significantly enhanced the capabilities of language models. Innovations such as Recurrent Neural Networks (RNNs), CNNs, and, most importantly, the Transformer architecture, introduced by Vaswani et al. (2023) in 2017, provided the foundation for LLMs.

The Transformer architecture revolutionized NLP by replacing traditional sequence-aligned recurrence and convolutional methods with a novel mechanism called self-attention. Self-attention allows the model to weigh the importance of different words in a sequence based on their relevance to one another, enabling it to effectively capture long-range dependencies (Hagos et al., 2024; Vaswani et al., 2017). relevance to one another, enabling it to effectively capture long-range dependencies

Transformers consist of encoder-decoder structures (Vaswani et al., 2017), where the encoder generates a contextual representation of the input sequence, and the decoder produces the output sequence based on this context. Additional innovations, such as multi-head attention and positional encoding, enhance the model's ability to process complex relationships across the input data. These features make Transformers highly efficient and scalable (Wolf et al., 2020), capable of handling massive datasets and supporting parallelized training. Due to these

advancements, LLMs have become essential for a wide range of applications, from machine translation and question answering to text summarization and code generation.

Their ability to generate contextually relevant, high-quality outputs has also driven innovations in interactive AI systems, such as chatbots and virtual assistants. This progress is driven by advancements in computational power, the availability of extensive datasets, and transfer learning, enabling pre-trained models to be fine-tuned for specific tasks with minimal additional training (Hagos et al., 2024). Techniques like Reinforcement Learning from Human Feedback (RLHF) further enhance their performance (Chang et al., 2023), enabling these models to learn from user interactions and improve their conversational abilities.

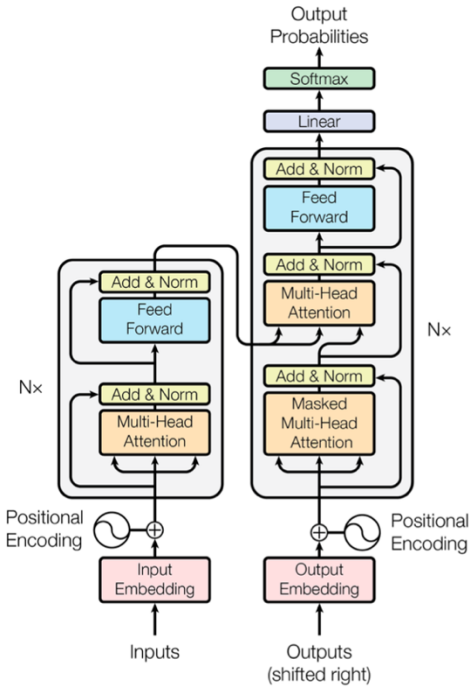


Figure 2.3 - The transformer - model architecture (Vaswani et al., 2017)

The Transformer architecture (Figure 2.3) has had a great impact due to its scalability and adaptability. It serves as the foundation for LLMs like Generative Pre-trained Transformer (GPT) -3 (Brown et al., 2020) and GPT-4 (OpenAI et al., 2024), which scale to hundreds of billions of parameters and excel across diverse tasks (Zhao et al., 2024). These models, including GPT and Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019), leverage vast datasets and computational power to handle complex tasks such as text generation, summarization, and translation (Hagos et al., 2024).

The ongoing growth of LLMs is further supported by tools like the Transformers library, which simplifies the deployment of pre-trained models. These tools enhance the adaptability and accessibility of these models, allowing experimentation with advanced architectures for diverse tasks (Wolf et al., 2020). Moreover, the fast integration of LLMs into real-world applications, such as speech recognition, personalized customer service, and educational tools, highlights their huge impact on technology and society (Hagos et al., 2024).

2.4.1. The Potential of Large Language Models in Healthcare

Across various fields, LLMs have shown success, and healthcare is no exception. They have shown a wide range of applications in healthcare, such as supporting diagnostic processes, administrative tasks, patient engagement and support, and treatment planning. Their ability to process large volumes of data and generate relevant insights allows them to address various operational and patient needs.

A key use of LLMs is in supporting clinical decision-making and radiology practices. Shen et al. (2024) highlight their role in assisting clinicians with imaging recommendations, protocol automation, and differential diagnoses. By synthesizing data from unstructured Electronic Health Records (EHRs), which are digital versions of a patient's medical history, as well as lab results and clinical notes, LLMs can accelerate diagnostic workflows and improve accuracy on diagnosis. Additionally, their capacity to summarize findings, detect errors, and format reports complements their application in generating clinical documentation, as described by Wang and Zhang (2024). These features can reduce workloads and enable healthcare professionals to focus on patient care.

In patient engagement and personalized medicine, LLMs also have been proven to excel in delivering targeted health information and providing detailed advice to patients Hemasri, Vijayalakshmi, and Jyotheesh (2024) explain their role in conversational AI agents that interact directly with patients, answering queries and offering insights relevant to specific health profiles. LLMs can assist in issuing medication reminders, tracking health data, and suggesting lifestyle modifications, further strengthening their impact on patient engagement. In oncology, these capabilities extend to providing patient-centered support, with Iannantuono et al. (2023) describing their use as "virtual assistants" that deliver specific and relevant information to both patients and physicians. This aligns with Shen et al.'s (2024) findings on LLMs' ability to simplify radiology reports into accessible language, improving patients' understanding of their conditions and enabling them to make informed decisions. These tools enhance operational efficiency while also improving communication between healthcare providers and patients, leading to better outcomes due to personalized support (Hemasri, Vijayalakshmi, & Jyotheesh, 2024). However, their application in fields like oncology highlights the need for verification processes because of risks such as outdated or inaccurate information.

LLMs are also applied in predictive modeling and clinical decision support. As mentioned by Hemasri, Vijayalakshmi, and Jyotheesh (2024) transformer-based models, such as GPT-4, can aid in tasks like patient triage, diagnosis, and treatment planning. Their integration with advanced AI technologies, including CNNs, advances the field of medical image analysis, especially in disease detection while also enhancing clinical workflows. These developments not only improve diagnostic accuracy but also enable the creation of more personalized treatment plans. Ongoing research is focused on enhancing model interpretability, integrating

diverse data types, and addressing challenges like data privacy and algorithm bias to further expand their applications in healthcare.

Table 2.3 - Summary of the findings from the studies highlighting the impact of LLMs on healthcare

Author(s) and Year	Objective	Conclusions	Methods
Shen et al. (2024)	Provide an overview of the capabilities and potential impact of LLMs and multi-modal large language models (MLLMs) in radiology.	LLMs and MLLMs have great potential in radiology, particularly in abdominal imaging. It emphasizes the ability of these models to improve workflows, including image analysis, report generation, image interpretation, and differential diagnosis (DDx) generation.	Review of significant achievements in LLM development and the exploration of their potential impact on abdominal radiology.
Wang and Zhang (2024)	Systematic review of the applications of LLMs in the medical and healthcare fields, focusing on their usage across various scenarios and tasks.	Highlights applications in various areas such as medical question-answering, clinical decision support, and medical education. LLMs can enhance communication, improve documentation, support clinical decisions, and facilitate research.	Systematic literature review methodology, searching for relevant papers across multiple databases, focusing on research from January 2022 to January 2024.
Hemasri, Vijayalakshmi, and Jyotheesh (2024)	Explore the impact of generative AI (GAI), especially LLMs in modern healthcare.	LLMs significantly enhance diagnostic accuracy, accelerate drug discovery, and improve clinical decision-making in healthcare. LLMs can enhance diagnostic accuracy and optimize treatment plans.	Discusses GAI models, particularly Transformers, and DL frameworks, to enhance healthcare applications.
Iannantuono et al. (2023)	Describe the current applications and potential impact of LLMs in oncology, particularly focusing on their	The review concludes that LLMs, particularly ChatGPT, have potential applications in oncology as virtual assistants for both patients and professionals. Also, the incorporation of LLMs into	A literature search was conducted on PubMed, covering articles from its inception to July

Author(s) and Year	Objective	Conclusions	Methods
	use in cancer care, and investigate the accuracy of responses generated by LLMs queries.	medical practice is necessary, but guidelines must be established to maximize benefits and minimize risks.	12, 2023, without applying filters.

2.4.1.1. Diagnostic Support and Validation through LLMs

LLMs have become increasingly relevant in assisting disease diagnosis by analyzing clinical data, identifying patterns, and generating DDx with minimal human intervention. According to Zhou et al. (2024), these models offer significant advantages, such as enhancing diagnostic accuracy, being capable of improving efficiency in clinical workflows, and supporting clinicians when managing complex cases or large patient panels. Their ability to process diverse data modalities, such as clinical notes, imaging studies, and time-series data, makes LLMs versatile tools. In their review, Zhou et al. (2024) analyzed existing literature on the application of LLMs for disease diagnosis, focusing on the methods, data requirements, and evaluation approaches employed in prior studies. Their research brings attention to the potential of LLMs in enhancing diagnostic accuracy and generating potential diagnoses, highlighting their applicability across diverse clinical specialties and disease types.

An example of this potential is the research by McDuff et al. (2023), which assessed the application of an LLM optimized for generating DDx, evaluating its performance and its role alone and then as an assistive tool for clinicians. The study aimed to assess whether LLMs could improve the accuracy and appropriateness of DDx lists, particularly in real complex medical cases. The research design involved the assessment of 302 complex medical cases. Twenty clinicians participated in the study, initially using traditional diagnostic tools such as search engines and medical references, and in the second part of the study using these tools supplemented with the LLM. The LLM demonstrated notable performance on its own, achieving a top-10 accuracy of 59.1%, significantly outperforming unassisted clinicians (33.6%), and surpassing the GPT-4 model in automated evaluations. When used as an assistive tool, the LLM enhanced the diagnostic capabilities of clinicians, resulting in higher quality and more accurate DDx lists. Clinicians assisted by the LLM achieved a top-10 accuracy of 51.7%, compared to 44.4% for those relying solely on traditional resources and 36.1% for unassisted clinicians. The inclusion of correct diagnoses in DDx lists, comprehensiveness scores, and overall quality improved significantly with LLM assistance. The study also explored clinicians' interactions with the LLM through qualitative interviews, revealing its potential to accelerate the diagnostic process and broaden the range of diagnoses considered. Participants highlighted the LLM's utility in training and education, emphasizing its ability to support the development of diagnostic reasoning skills.

Ríos-Hoyo et al. (2024) provide another example of LLMs in diagnostic applications, assessing the performance of OpenAI's GPT-3.5 and GPT-4 in generating DDx for complex clinical cases. The study aimed to evaluate the diagnostic accuracy of these models and identify limitations, including misdiagnosis patterns, aiming to improve their use in clinical decision support. The study methodology involved analyzing 75 complex clinical cases. These cases were presented to GPT-3.5 and GPT-4 using a standardized prompt designed, the same for both models, to produce DDx, prioritizing the most likely explanation for the symptoms. The models' performance was measured by metrics such as the accuracy of listing the correct diagnosis in the top positions, similarity to expert-generated lists, and correlation with disease representation in the medical literature.

The findings revealed that GPT-4 outperformed GPT-3.5 in several metrics. GPT-4 identified the correct diagnosis in 68% of cases, compared to 48% for GPT-3.5, and had greater alignment with DDx lists generated by experts (Jaccard Similarity Index of 0.22 vs. 0.12). Additionally, GPT-4 achieved higher accuracy when diagnoses were grouped by medical specialty, listing the correct diagnosis among the top three in 42% of cases compared to 24% for GPT-3.5. Both models showed a tendency to rely mainly on the representation of conditions in medical literature rather than actual disease incidence.

Their performance was moderate, neither exceptional nor inadequate, demonstrating some capability in generating potential diagnoses but with several limitations in consistency and accuracy. These results point to significant limitations that might explain why the models did not achieve better outcomes, highlighting the need for a cautious interpretation of the findings when understanding the potential of these models. A key weakness was the reliance on a small dataset from the Massachusetts General Hospital Case Records, which includes mainly complex and uncommon cases. This limits the results' applicability to more common clinical scenarios where LLMs might perform better. Additionally, the GPT models were trained on data collected before September 2021, reducing their effectiveness for conditions with changing epidemiology and highlighting the importance of continuously updating training data. Another issue was the use of a single prompt to evaluate model performance, potentially limiting the exploration of alternative prompt designs that could improve accuracy. Addressing these limitations by using larger, more diverse datasets, ensuring training data reflects current trends, and optimizing prompt designs could significantly enhance the performance and applicability of LLMs in clinical diagnostics.

2.4.1.2. LLMs for Treatment Planning and Support

Besides diagnostic tasks, LLMs have shown remarkable potential in transforming treatment planning and personalized healthcare strategies. Baig et al. (2024) explain the capability of LLMs to integrate diverse patient data, such as demographics, comorbidities, prior treatment responses, and preferences, to design highly personalized care plans. This approach surpasses traditional methods by providing treatment plans to the individual needs of patients, leading to more effective and centered care. LLMs can also enhance predictive accuracy in treatment

outcomes by analyzing large datasets and finding patterns that may not be evident to clinicians. As Baig et al. (2024) note, this predictive ability enables healthcare providers to improve treatment strategies, ensuring more accurate clinical decisions. Furthermore, these models can automate repetitive tasks, such as data analysis and the generation of preliminary care plans, allowing clinicians to have the highest focus on patient care.

In addition to general applications, domain-specific LLMs, such as RadOnc-GPT, developed by Liu et al. (2023), highlight the potential of fine-tuned models trained on specialized datasets to enhance clinical decision-making in fields like oncology. RadOnc-GPT was designed to perform three key tasks: generating radiotherapy treatment regimens, recommending optimal treatment modalities, and providing diagnostic descriptions with International Classification of Diseases (ICD) codes - a standardized system by the World Health Organization (WHO) for classifying and coding diseases, health conditions, and related procedures. Each task was evaluated separately to assess the model's performance. For the first task, RadOnc-GPT generated detailed radiotherapy regimens based on patient data, integrating diagnostic assessments, tumor staging, and treatment procedures. It achieved ROUGE-1, ROUGE-2, and ROUGE-L scores of 0.4341, 0.2250, and 0.4271, respectively, significantly outperforming Llama2, which couldn’t achieve scores higher than 0.0739. In treatment modality selection, the model was able to recommend appropriate radiation therapies, achieving ROUGE-1 and ROUGE-L scores of 0.7903, far exceeding Llama2's scores, which were near zero. For diagnostic descriptions and ICD code predictions, RadOnc-GPT attained ROUGE-1 and ROUGE-2 scores of 0.7050 and 0.6203, demonstrating superior accuracy and efficiency in administrative tasks surpassing Llama2’s scores that couldn’t reach 0.012. These results show the model's ability to improve these processes, reduce errors, and align treatment strategies with patient needs. RadOnc-GPT exemplifies the potential of domain-specific LLMs to improve workflows and support decisions in specialized medical contexts, although further validation is still needed for broader clinical adoption.

Similarly, other tools, as highlighted by Zhang and Kamel Boulos (2023), have proven valuable by providing healthcare professionals with detailed, evidence-based support that enhances clinical decision-making. For example, tools like Glass AI (Board of Innovation, n.d.) assist in generating DDx and suggesting treatment plans personalized to individual patient needs. When faced with symptoms, the system can identify potential conditions and guide clinicians through diagnostic tests and treatment options. Similarly, platforms like Regard (Regard, n.d.) work with EHR systems, using patient data to recommend treatments and improve diagnostic accuracy. This allows clinicians to consider new therapeutic approaches and optimize existing plans, making care delivery faster and more precise.

Automated clinical documentation tools like Suki Assistant (Suki.ai, n.d.) and Nuance AI (Nuance, n.d.), as noted by Zhang and Kamel Boulos (2023), also play a key role in supporting treatment strategies. These tools transcribe patient-clinician conversations and organize key information into EHRs, freeing up clinicians’ time for treatment planning. Tools like Corti

(Corti, n.d.) also extract important details, such as symptoms, medications, and treatment recommendations, from patient interactions. This ensures that no important information is missed and that treatment strategies are based on accurate and specific patient data.

Zhang and Kamel Boulos (2023) also discuss the role of these tools in personalized care. Platforms like Kahun (Kahun, n.d.) assess symptoms and suggest ranked diagnoses along with customized workup plans, helping clinicians develop individualized treatment strategies. Similarly, tools like Google Bard's Med-PaLM 2 (Rahaman et al., 2023; Google Research, n.d.) assist in diagnosis and recommend treatments by drawing on diverse medical datasets. This broad knowledge base supports clinicians in addressing both common and complex cases, such as managing blood disorders.

Table 2.4 - Summary of the findings from studies leveraging LLMs for diagnosis and treatment planning

Author(s) and Year	Objective	Methodology	Performance metrics	Findings
Zhou et al. (2024)	Review the application of LLMs in disease diagnosis, summarizing various clinical specialties, data modalities, techniques used, and evaluation methods.	Reviews various methods employed in LLMs for disease diagnosis, categorizing them into four main types: prompt-based methods, RAG, fine-tuning, and pre-training.	Not specified	LLMs have advantages, such as enhancing diagnostic accuracy, being capable of improving efficiency in clinical workflows, and supporting clinicians when managing complex cases or large patient panels.
McDuff et al. (2023)	Evaluate the performance of an LLM optimized for diagnostic reasoning in generating DDx for challenging medical cases.	Evaluation of an LLM on 302 complex cases, compared alone and as a clinician assistive tool.	LLM for DDx: top-10 accuracy = 59.1% compared to 33.6% for clinicians without assistance. Clinicians assisted by the LLM: top-10 accuracy of 51.7% versus 36.1% for those without	LLMs for DDx serve as a valuable assistive tool for clinicians, enhancing the quality of DDx generation. LLM for DDx outperformed unassisted clinicians. Clinicians assisted by the LLM produced higher-quality DDx lists.

Author(s) and Year	Objective	Methodology	Performance metrics	Findings
			LLM assistance.	
Ríos-Hoyo et al. (2024)	Evaluate the performance and limitations of LLMs in providing correct diagnoses for complex clinical cases.	Use OpenAI's GPT-3.5 and GPT-4, as diagnostic aids for complex medical cases using 75 clinical cases from Massachusetts General Hospital.	GPT-4 outperformed GPT -3.5 in accuracy (68% vs. 48%) and alignment with expert lists (Jaccard Index: 0.22 vs. 0.12).	OpenAI's GPT-4 model outperformed GPT-3.5 in diagnosing complex clinical cases, although misdiagnosis was still prevalent. These models could serve as decision aids rather than replacements.
Baig et al. (2024)	Investigate the opportunities, challenges, and barriers to implementing LLMs in personalized patient care plans.	PRISMA review methodology to systematically analyze the literature of 13 articles that were included in the final review.	Not specified	LLMs have a significant potential in creating personalized treatment care plans by analyzing various patient factors, thus enhancing personalized care.
Liu et al. (2023)	Present RadOnc-GPT, a LLM specifically fine-tuned for radiation oncology tasks.	Fine-tune RadOnc-GPT on a large dataset of radiation records, for radiotherapy regimen generation, treatment modality selection, and ICD code predictions.	ROUGE-1 = 0.4341, ROUGE-L = 0.4271 (vs. near-zero scores for Llama2)	RadOnc-GPT demonstrated superior performance in generating clinically coherent assessments compared to general LLMs. RadOnc-GPT can enhance the speed, accuracy, and quality of radiation therapy decision-making.
Zhang & Kamel Boulos (2023)	Present a review of examples of LLMs applications in medicine and healthcare.	Examples include Glass AI, Regard, Suki Assistant, Nuance AI, and Google Bard's Med-PaLM 2 for diagnosis, and treatment plans.	Not specified	LLMs have great potential to better workflows, enhance diagnostic accuracy, and create treatment plans, allowing more precise and faster care delivery.

2.4.2. Techniques for Optimizing LLMs

RAG (Lewis et al., 2021; Zhao et al., 2024) addresses the limitations of LLMs in handling real-time or specialized domain knowledge, which is particularly important in fields like oncology. Unlike standard LLMs, which rely on static internal knowledge, RAG integrates external information sources, such as domain-specific knowledge bases or the internet, to improve the relevance and accuracy of its outputs. The process involves retrieving contextually relevant information, incorporating it into the model's input prompt, and generating responses informed by the retrieved content. This approach minimizes factual errors and allows for more targeted outputs. RAG also serves as a customized prompting strategy by incorporating external information into the model's input. This approach allows LLMs to dynamically adapt to specific information requirements, ensuring their responses are accurate and contextually relevant (Fan et al., 2024; Zhao et al., 2024).

In addition to RAG, prompt engineering improves LLMs' responses by structuring and customizing input instructions. This method (Zhang et al., 2024) ensures outputs are accurate, specific to the task in question, and overall aligned with the intended purpose. Techniques include zero-shot prompting, where the model performs tasks without examples; one-shot prompting, in which a single example is provided to guide responses; and few-shot prompting, which consists of providing the model with multiple examples to establish patterns. Other strategies involve providing clear instructions, specifying context, applying explicit constraints, and assigning roles. While no single best approach exists, iterative testing and refinement of prompts can significantly improve their performance. In fields like medical diagnostics, where accuracy and interpretability are critical, structured prompts have been shown to improve reliability and outcomes. The sections below provide different examples of how both techniques were used and their capabilities in improving the overall performance of LLMs.

2.4.2.1. Prompt Engineering

Zhang et al. (2024) analyzed the impact of prompt engineering on the performance of different LLMs in identifying metastatic cancer from discharge summaries. The study demonstrated that structured prompts significantly enhance model performance, often surpassing domain-specific models like PubMedBERT (Gu et al., 2021), emphasizing the importance of prompt engineering in achieving precision and interpretability in biomedical tasks. For their research, a dataset of 1,873 discharge summaries from MIMIC-III, annotated by medical fellows was used. The study evaluated models across different prompt engineering techniques such as zero-shot, one-shot, and fine-tuning strategies. Six prompts were designed and tested, ranging from baseline to refined versions incorporating chains of thought and structured instructions. The evaluation included various input token lengths (1,500 and 3,000) and metrics such as F1 scores, precision, and recall. The results showed that structured prompts with clear instructions significantly improved performance, where the prompt that performed the best achieved the highest F1 score of 0.941 and precision of 1.000 in GPT-4. In their research they also conclude that simple and concise prompts also performed

comparably, demonstrating that clarity is as effective as advanced techniques. Prompt engineering proved critical for leveraging models like GPT-4, enabling them to outperform even specialized models.

Input token length also influenced outcomes. While shorter inputs (1,500 tokens) worked better for GPT-3.5 Turbo, GPT-4 handled longer inputs (3,000 tokens) effectively, maintaining high precision and recall. This ability to process extended contexts differentiates GPT-4 from its predecessors and underscores the potential of prompt engineering to optimize model utility in real-world scenarios. The study concluded that effective prompt engineering allows LLMs like GPT-4 to rival or exceed specialized models in biomedical applications. By focusing on structured, clear prompts, the need for resource-intensive strategies such as fine-tuning is reduced, making LLMs a scalable solution for complex tasks in healthcare.

Similarly, Most et al. (2024) conducted a study to evaluate the performance of LLMs in clinical pharmacy tasks. The study compared five LLMs on 219 multiple-choice questions derived from pharmacy curricula. By assessing their accuracy on knowledge-based and skill-based questions, the research aimed to explore the role of prompt engineering in enhancing LLMs' reasoning and response capabilities, particularly in the context of clinical pharmacy. The findings revealed that GPT-4 consistently outperformed the other models, achieving an average accuracy of 71.6%, with the highest performance observed on knowledge-based questions (87%) compared to skill-based questions (67%). Prompt engineering emerged as one of the main factors for improving LLM performance, particularly for GPT-4. Without prompt engineering, GPT-4 achieved an accuracy of 84% on knowledge-based questions, comparable to the performance of third-year pharmacy students. However, the use of advanced prompt engineering techniques, such as Chain-of-Thought (CoT) prompting and self-consistency approaches, significantly enhanced accuracy. With these techniques, GPT-4 achieved up to 93% accuracy, surpassing the baseline performance of pharmacy students.

Another example of the impact of prompt engineering is presented by Wang et al. (2025), who explored the feasibility of using ChatGPT as a cost-effective alternative to standardized patients in medical training, specifically for history-taking tasks. The study evaluated ChatGPT’s performance, emphasizing the role of prompt engineering in improving outputs for medical assessments. Conducted in two phases, it analyzed the chatbot’s accuracy, adaptability, and anthropomorphism in simulating standardized patients. In the first phase, ChatGPT's responses to inquiries about inflammatory bowel disease (IBD) were assessed across three quality groups—good, medium, and bad—over 30 runs each. Responses were evaluated for relevance and accuracy. In the second phase, prompt engineering was applied to enhance anthropomorphism, clinical accuracy, and adaptability. Responses from original prompts (OP) and revised prompts (RP) were compared across 300 runs. The study also examined the effect of varying language structures on ChatGPT's performance. Prompt engineering significantly improved the models’ performance by addressing limitations such as unstable patient mimicry and overly broad responses that included unasked information,

inflating scores inaccurately. By refining prompts to impose constraints and guide outputs, the chatbot's realism, clinical accuracy, and adaptability improved. Scoring discrepancies dropped from 29.83% to 6.06%, with the standard deviation decreasing from 0.55 to 0.068, and scoring accuracy improved nearly fivefold, highlighting the effectiveness of prompt optimization.

Many other studies highlight the importance and impact of prompt engineering in improving LLM performance. For example, the studies by Sakai et al. (2024) and Musa et al. (2024). Sakai et al. utilized prompt engineering techniques such as zero-shot learning, in-context learning, and CoT prompting to conduct Multi-label Text Classification (MLTC) on sensitive inpatient comments. Their results showed that GPT-4 Turbo, when paired with these techniques, significantly outperformed traditional methods and pre-trained language models (PLMs), achieving a high F1-score and weighted F1-score. By structuring prompts to interpret patient feedback, the study showed how this approach provides useful insights for healthcare practitioners, improving patient care and experience analysis.

Similarly, Musa et al. (2024) explored prompt engineering in the context of medical diagnosis, employing a systematic, multi-step algorithm to refine and optimize prompts for diagnostic insights. This approach involved structured processes such as input processing, trigger token matching, and template selection, enabling the model to deliver more accurate and relevant responses. The study found that engineered prompts covered clinical presentations, diagnostic criteria, treatments, and complications more effectively than manual prompts, improving standardization and diagnostic utility.

Together, these studies exemplify the impact prompt engineering has in significantly improving the capabilities of LLMs across diverse healthcare contexts.

2.4.2.2. RAG

As discussed previously, RAG is designed to improve the accuracy and reliability of responses generated by LLMs. An example of its broad applications in several fields is the study of Sharma et al. (2024), which introduced Ontology-Grounded Retrieval Augmented Generation (OG-RAG). This method integrates RAG with domain-specific ontologies to improve the accuracy and relevance of LLM-generated responses, especially in specialized domains. Ontologies provide structured frameworks that define entities and their relationships within a domain, organizing concepts and their interconnections systematically.

Unlike traditional retrieval-augmented models, which often fail to account for the complex relationships required for specialized tasks like healthcare, OG-RAG employs a hypergraph representation of domain documents to capture factual knowledge. In this approach, hyperedges connect related pieces of information. Using a greedy algorithm, the system retrieves the minimal set of hyperedges required to construct a precise context for a query, enabling efficient retrieval without computational overhead.

OG-RAG can be applied in various domains, including healthcare, legal, agriculture, and other tasks like journalism and research, where specific and accurate knowledge is essential. OG-RAG demonstrated significant improvements over other retrieval methods, including a 55% increase in the recall of accurate facts and a 40% improvement in the correctness of generated responses. Additionally, the method enabled 30% faster attribution of responses to context and improved fact-based reasoning accuracy by 27%. The study concluded that OG-RAG's ability to structure domain knowledge through ontologies and enhance LLM responses makes it highly effective in specialized workflows, offering more reliable and contextually accurate answers. These results highlight the adaptability of RAG when applied to specific domains, showing its ability to improve LLM performance across diverse fields.

Building on this, research in the medical field has further explored the potential of RAG to address limitations in LLMs, such as hallucinations. Xu et al. (2024) conducted a study to evaluate whether RAG could improve the accuracy, empathy, and relevance of responses and provide a theoretical basis for its application in nursing practice and education. The researchers conducted a study involving two groups. The control group used GPT-4 to directly answer questions, while the experimental group (RAG-GPT) used GPT-4 combined with RAG technology. For the experimental group, the RAG process integrated a knowledge base focused on breast cancer nursing care, including resources like textbooks, clinical guidelines, and traditional Chinese therapy. Semantic matching was performed using an embedding model, which retrieved the most relevant knowledge from the database based on a cosine similarity threshold. The relevant information was then integrated with the input query before being processed by GPT-4 to generate a response.

The findings revealed that the integration of RAG significantly improved overall satisfaction and accuracy in responses. The RAG-GPT group achieved higher scores in both satisfaction (8.4 ± 0.84) and accuracy (8.6 ± 0.69) compared to the GPT-4 group (satisfaction: 5.4 ± 1.27; accuracy: 5.6 ± 0.96). However, there was no significant difference in empathy between the two groups, with the RAG-GPT group scoring 8.4 ± 0.85 and the GPT-4 group scoring 7.8 ± 1.22. These results indicate that RAG can enhance the reliability of LLM responses without compromising their empathetic tone, making it a valuable tool in medical contexts.

Expanding on RAG's medical applications, Ranjit et al. (2023) introduced CXR-RepaiR-Gen, a retrieval-augmented method for generating radiology reports from chest X-rays, aiming to improve report accuracy and relevance. This approach combined multimodal embeddings from a vision-language model with OpenAI's LLMs to generate detailed reports. Aligned text and image embeddings retrieved relevant impressions from radiology datasets, which were used as input to LLMs. Prompt engineering ensured the generated reports were customized for clinical use, addressing common issues like irrelevant details and repetitive content. The evaluation showed a 25.88% improvement in BERTScore, a 6.31% improvement in Semb, and maintained parity in RadGraph F1, demonstrating reports closer to reference standards without compromising clinical accuracy. By combining domain-specific retrieval with LLMs,

the study also highlights the value of RAG in improving radiology workflows by producing concise and contextually accurate reports suited to clinical requirements.

Similarly, the studies by Long et al. (2024) and Miao et al. (2024) also address the potential of integrating RAG into LLMs for medical applications. Long et al. (2024) introduced Bailicai, a framework designed to overcome limitations in open-source LLMs, such as hallucinations and insufficient domain knowledge. Bailicai outperformed existing models, including GPT-3.5, across medical benchmarks, improving performance by 20.72% compared to other RAG systems. It also reduced hallucinations and irrelevant document noise, making it efficient for medical settings. Miao et al. (2024) focused on nephrology, integrating RAG with a customized GPT-4 model to enhance accuracy in chronic kidney disease (CKD) treatment. By grounding responses in the KDIGO 2023 guidelines, which provide recommendations for the management of kidney diseases, the model provided more precise and specialized medical advice than the general GPT-4.

Together, these studies highlight the great impact and capabilities of RAG in enhancing LLM performance, particularly in the medical field. By grounding responses in structured, domain-specific knowledge, RAG improves the accuracy, reliability, and specificity of generated outputs. This makes it a valuable tool in healthcare, where accurate information can directly impact patient outcomes.

Table 2.5 - Summary of the findings from studies that use RAG and Prompt engineering to enhance LLMs' performance

Author(s) and Year	Objective	Methodology	Performance metrics	Findings
Zhao et al. (2024)	Review recent advances in LLMs.	Mentions: RAG for integrating external knowledge bases to improve outputs.	Not specified	RAG minimized factual errors, improved contextual relevance, and enabled dynamic adaptation of LLM responses.
Zhang et al. (2024)	Evaluate prompt engineering and fine-tuning strategies in LLMs.	Prompt engineering techniques: zero-shot, one-shot, and few-shot methods. Evaluation of different LLMs.	GPT-4 outperformed GPT-3.5 Turbo and Llama-7B models. F1 = 0.941 (GPT-4); Precision = 1 (best prompt)	Clear prompts improved model performance significantly. One-shot learning and fine-tuning showed no incremental benefit.
Most et al. (2024)	Compare the performance	Comparison of five LLMs performance.	GPT-4 achieved the	Prompt engineering techniques can

Author(s) and Year	Objective	Methodology	Performance metrics	Findings
	of LLMs, and assess their reasoning using prompt engineering techniques, on pharmacotherapy.	Implementation of zero-shot CoT approach.	highest accuracy at 71.6%, but with advanced prompts, GPT-4 achieved 93%.	significantly improve LLMs' accuracy.
Wang et al. (2025)	Explore ChatGPT as a substitute for standardized patients	Assessed ChatGPT responses for tasks before and after prompt refinement.	Accuracy improved ~5x (post-prompt refinement); SD = 0.068.	Prompt engineering improved ChatGPT's clinical accuracy, adaptability, and scoring consistency, reducing discrepancies from 29.83% to 6.06%.
Sakai et al. (2024)	Conduct Multi-label Text Classification (MLTC) of inpatient comments.	Used zero-shot, in-context learning, and CoT prompting for sensitive inpatient feedback analysis.	GPT-4 Turbo F1-score = 76.12% ±0.021, Weighted F1-score = 73.61% ± 0.006	Prompt engineering is valuable in improving patient care and experience analysis.
Musa et al. (2024)	Enhance medical diagnosis using prompt engineering in LLMs.	Utilized a systematic, multi-step algorithm for prompt construction.	Not specified	Prompt engineering enhances model performance in medical diagnosis. Reprompting yields superior responses on medical topics.
Sharma et al. (2024)	Improve LLM responses using domain-specific ontologies.	OG-RAG	Recall: + 55%, Correctness: + 40%, Fact-based reasoning: +27%.	OG-RAG improves domain adaptation for LLMs using ontologies. Enhances factual accuracy in LLMs responses.

Author(s) and Year	Objective	Methodology	Performance metrics	Findings
Xu et al. (2024)	Evaluate LLM performance in breast cancer nursing care.	Two groups were established: control (GPT-4) and experimental (RAG-GPT).	Accuracy: 8.6 ± 0.69; Satisfaction: 8.4 ± 0.84.	RAG significantly improves LLM performance for nursing care. Increases answers' accuracy without reducing empathy.
Ranjit et al. (2023)	Radiology report generation with RAG LLMs.	Combined multimodal embeddings and RAG to generate reports.	BERTScore: +25.88%; RadGraph F1: Parity maintained, Semb score: +3.86 points.	RAG reduces hallucinations in generated radiology reports. The approach improves clinical metrics significantly.
Long et al. (2024)	Develop the Bailicai for medical applications. Enhance the performance LLMs' performance by integrating RAG.	Integration of RAG and medical knowledge injection for domain-specific fine-tuning.	Bailicai Performance: +20.72% vs. GPT-3.5.	Bailicai improves medical outcomes through RAG. Reduces hallucinations and document noise issues.
Miao et al. (2024)	Integrate LLMs with RAG systems in nephrology, enhancing the accuracy and reliability of medical information.	Prompt engineering, including CoT approach + RAG for enhanced accuracy.	Not specified	Integration of LLMs with RAG enhances nephrology applications, reducing outdated or incorrect information in responses.

2.5. RELATED WORK

Within the scope of the reviewed research, two studies stood out as highly relevant for integrating multiple methodologies into a unified framework: Azurmendi et al. (2024) and Wang et al. (2023). These works were the only ones identified in the literature that explicitly

attempted to combine classification models and LLMs attempting to create a more “complete” diagnostic tool. Both studies utilize distinct yet complementary techniques, highlighting the potential for integrating different frameworks within the same project to address complex clinical challenges.

Azurmendi et al. (2024) studied the use of DL algorithms to detect postural asymmetries in individuals aged 3 to 20 years using pressure platforms. The goal was to improve the diagnosis, treatment, and management of postural disorders and motor disabilities through continuous, non-invasive monitoring.

The study applied CNN for classification, hemibody segmentation (dividing the body into symmetrical halves), and recognition of specific body parts, along with YOLOv8 for object detection. Pressure mat data were analyzed to identify postural anomalies by segmenting the body into regions like the head, trunk, and limbs. YOLOv8 localized body parts within the pressure images and handled noisy, low-resolution data effectively. The trained model accurately analyzed new images, assigning pressures to body regions and providing specialists with detailed pressure distribution insights. Additionally, a pre-trained LLaMa3 model generated automated clinical reports based on the processed data. Guided by clinical prompts, it summarized results from the pressure mat analysis and outputs from the DL algorithms, enabling quick report generation for clinical records. While the use of LLaMa3 was not directly compared to other language models, it was evaluated by health specialists and considered acceptable.

The framework achieved 100% accuracy in classification, a mean absolute error (MAE) of 7 for hemibody segmentation, and 70% accuracy for object detection. The study concluded that integrating DL algorithms with pressure mat data could enhance the monitoring of postural anomalies, support personalized care, improve patient monitoring, and assist clinical decision-making.

Wang et al. (2023) investigated integrating LLMs with computer-aided diagnosis (CAD) systems to improve medical image interpretation, such as X-rays. The goal was to create a user-friendly system that combined CAD models' visual analysis capabilities with the reasoning and medical knowledge of LLMs, addressing the challenge of making CAD outputs more interpretable for clinicians. The methodology of this research involved several steps. First, medical images, such as X-rays, were processed using trained CAD networks, including diagnosis networks, lesion segmentation networks, and report generation networks. These networks generated initial outputs, in the form of probabilities or segmentation maps. The next step involved translating these outputs, which were in the form of raw tensors, into a natural language format that could be interpreted by the LLM. This was done by converting the results into a grading system that categorized diseases based on their likelihood (e.g., "No sign," "Small possibility," etc.). The LLM, specifically GPT-3 (text-davinci-003), was then used to generate a medical report by summarizing the results and give a conclusion. The model was

guided by various prompts designed to format the outputs in a way that was able to engage in a conversation regarding the results.

The results showed that the integration of LLMs into CAD networks significantly improved the interpretability and accuracy of generated reports, achieving superior recall and F1 scores compared to baseline methods. The reports were also better aligned with clinical language, making them more useful for healthcare professionals and patients. However, limitations included occasional over-reliance on raw CAD outputs in the reports.

The study concluded that combining LLMs with CAD networks improves medical image interpretation by making diagnostic outputs more understandable. It also emphasized the importance of refining models, designing better prompts, and developing more advanced LLMs to enhance diagnostic accuracy and report quality in clinical settings.

Table 2.6 - Summary of the findings from studies that use DL models and LLMs as a unified framework

Author(s) and Year	Objective	Methodology	Performance metrics	Findings
Azurmendi et al. (2024)	Improve detection of postural asymmetries and motor disabilities in children and adolescents.	Combined DL techniques (CNNs, hemibody segmentation, YOLOv8) and LLMs (LLaMa3) to process pressure platform data and generate clinical reports.	Classification Accuracy: 100%; MAE: ~7; Object Detection: 70%.	DL Algorithms can effectively assist in early detection of postural anomalies and monitor postural disabilities.
Wang et al. (2023)	Enhance CAD systems for medical image interpretation using LLMs.	ChatCAD: Integrated CAD networks (diagnosis, segmentation, report generation) with GPT-3 (text-davinci-003) for medical reporting.	Superior recall and F1-score compared to baseline models.	ChatCAD enhances interactive medical image diagnosis using LLM, improving radiology report quality.

2.6. DISCUSSION

The reviewed works, analyzed throughout this chapter, demonstrate significant advancements in tumor detection, classification, and staging using YOLO, hybrid DL models, and LLMs. However, they reveal several limitations. For instance, studies by Wehbe et al. (2024), Elshahawy et al. (2023), and Elazab et al. (2024) primarily focus on imaging data, overlooking demographic features such as age, gender, or even habits such as smoking or drinking, which are vital for a personalized approach to diagnosis. Although these models

achieve high accuracy and precision, they lack mechanisms to validate predictions or provide interpretable outputs to directly support clinical decision-making.

Moreover, current models often specialize in isolated tasks, such as TNM staging or histological grading, without offering broader solutions or incorporating tools for generating treatment plans alongside the diagnosis.

The reviewed research highlights significant contributions of LLMs and DL models in healthcare but reveals gaps limiting broader applications. Studies like McDuff et al. (2023) and Ríos-Hoyo et al. (2024) focus on specific diagnostic tasks, such as DDx or radiotherapy planning, but fail to integrate multimodal data, like patient demographics and clinical notes, essential for personalized care. Additionally, the absence of RAG techniques reduces the reliability and applicability of the outputs in medical settings.

Another gap lies in the lack of systems that combine diagnosis with treatment planning. While LLMs have demonstrated potential for summarizing findings and improving DDx, they remain limited in their ability to synthesize diverse data and generate specific treatment plans. Models like RadOnc-GPT (Liu et al., 2023) and platforms like Glass AI and Med-PaLM 2 focus on narrow, domain-specific tasks such as radiation therapy recommendations or managing common cases. However, they lack the flexibility to address the complexity of personalized oncology care, particularly across varying stages and grading levels of diseases like lung cancer.

Azurmendi et al. (2024) and Wang et al. (2023) explored integrating DL and LLMs into unified frameworks, highlighting both advancements and limitations in the field. Both studies demonstrated the potential of combining DL for data analysis with LLMs for generating readable outputs, such as clinical reports. However, they share significant gaps that limit their applicability in clinical contexts.

Neither framework fully integrates multimodal data, such as combining demographic and imaging information, which is essential for personalized diagnostics and treatment planning. Additionally, both rely on LLMs for output generation but do not utilize them to validate DL or CAD model outputs, reducing the reliability and accuracy of outputs. Both studies also fall short in employing advanced techniques like RAG or even prompt optimization. These tools are critical for grounding LLM responses in reliable, up-to-date data and enhancing model adaptability to specialized datasets or diseases. While Wang et al. and Azurmendi et al. recognized the potential of LLMs to improve interpretability and usability, neither addressed the broader challenges of adapting their frameworks for complex clinical scenarios, such as managing diverse data sources or generating treatment plans.

This research aims to address these gaps by integrating a DL framework designed for lung cancer imaging with LLMs to create a unified workflow for prediction, validation, and treatment planning. Unlike previous studies, this framework uses LLMs to infer cancer stage and generate appropriate treatment protocols. Additionally, it incorporates multimodal data,

such as lung cancer imaging and demographic features, to predict cancer type and TNM staging. The predictions, along with the patient data, will be fed into LLMs, which, through RAG and optimized prompts, aim to generate specific and accurate outputs. These outputs include treatment plans that provide recommendations to the needs of individual patients. By combining customized DL models, LLM, and techniques like RAG, the proposed research aims to provide a robust, personalized solution for lung cancer care, addressing gaps in current methods and improving diagnostic frameworks in oncology.

3. METHODOLOGY

The methodology chapter is divided into two main parts, each covering a distinct component of the system: the image classification model and the treatment recommendation model. The objective is to develop a unified system that receives only a CT image and the demographic data of a patient as input and can classify the cancer type, predicting the TNM stages, inferring the overall cancer stage, and generating a treatment recommendation aligned with international guidelines for lung cancer care (Figure 3.1).

The first part of the methodology explains the development of the image classification model. It begins with a description of the datasets used, namely LUNG PET-CT-DX and NSCLC-Radiomics, and describes the image extraction process along with the inclusion of the demographic patient data for the classification of cancer types and TNM stages. This section also explains the pre-processing steps applied to both image and non-image data, followed by the procedures used for data splitting and augmentation. Additionally, it introduces the models used, YOLOv8 and ResNet50, and presents the experimental setup for their implementation.

The second part focuses on the report generation system, responsible for predicting the overall cancer stage and recommending a treatment plan. It starts by describing the construction of the knowledge base and its integration into the RAG pipeline, followed by the pre-processing of the textual data obtained through web scraping and PDF extraction. The section then explains the embedding models and retrieval strategies tested, as well as the language models used for generating the final outputs.

Each part concludes with an explanation of the metrics used to evaluate the respective models.

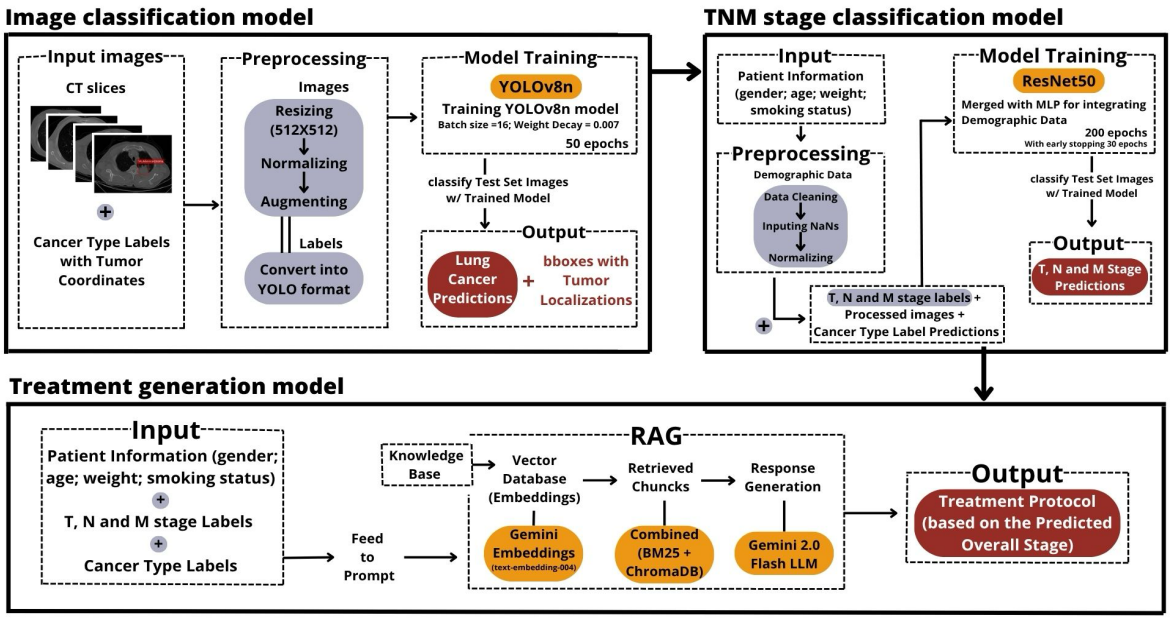


Figure 3.1 - Proposed Methodology for Lung Cancer Classification, Staging, and Treatment Recommendation

3.1. IMAGE CLASSIFICATION MODEL

The first part of the methodology, detailed below, focuses on the image classification model. It describes the processing of imaging and clinical data for cancer type and TNM stage prediction, including data preparation, model implementation, and evaluation methods.

3.1.1. Data: LUNG PET-CT-DX

The LUNG PET-CT-DX dataset (Li et al., 2020) consists of CT and PET-CT DICOM images of 355 lung cancer patients, with corresponding XML files indicating tumor locations through bounding boxes (Figure 3.2). The images were collected from patients who underwent PET/CT imaging and lung biopsies for suspected lung cancer. Patients were categorized by histopathological diagnosis: ADC (IDs containing 'A'), consisting of 251 patients; SCLC ('B'), consisting of 38 patients; LCC ('E'), comprising 5 patients and SCC ('G'), representing a total of 61 patients. The tumor annotations were performed by five thoracic radiologists specializing in lung cancer. These annotations are provided in PASCAL VOC XML format. Each study includes a CT volume, PET volume, and fused PET/CT images. CT images have a resolution of 512 × 512 pixels at 1 mm × 1 mm, and PET images have a resolution of 200 × 200 pixels at 4.07 mm × 4.07 mm. Both imaging modalities are reconstructed with a 1 mm slice thickness. The dataset also includes clinical information for all subjects and is publicly available through The Cancer Imaging Archive's (Clark et al., 2013) Lung-PET-CT-Dx collection.

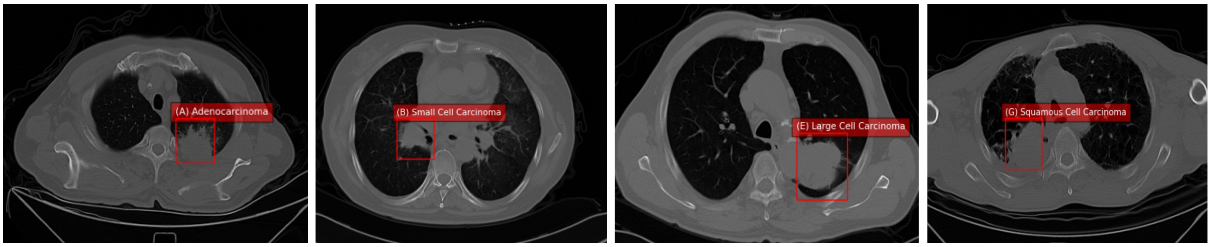


Figure 3.2 - Sample of Lung Cancer CT Images with Tumor Annotations from the Lung-PET-CT-Dx dataset: (A) Adenocarcinoma; (B) Small Cell Carcinoma; (E) Large Cell Carcinoma; (G) Squamous Cell Carcinoma

3.1.1.1. Image Extraction and Sampling Process

For this study, only CT scans from the dataset were considered. The original dataset consisted of 251,135 images (127.18 GB), making it infeasible to process in its entirety due to memory constraints. To address this, patients were sampled systematically, ensuring the final subset remained representative of the original data across the four target variables: T-stage, M-stage, N-stage, and histopathological diagnosis.

The patient selection was guided by a metadata CSV file containing information about each image. The sampling process accounted for the number of images per patient and preserved the distribution of the target variable (histopathological diagnosis) to avoid class imbalance, while also ensuring that no T-stage, M-stage, or N-stage labels were excluded from the final dataset. Since class sizes varied, all patients except one (explained in section 3.1.2.2) from

class B (SCLC) and all patients from class E (LCC) were included in the final sample due to their smaller populations (5 and 37 patients, respectively). For classes A (ADC) and G (SCC), patients were sampled until the total number of images reached approximately that of class B (3 021 images), in effort to maintain class balance. Class E was not considered in this process due to its limited number of available images (201 images). After sampling, the final subset consisted of 5 patients from class E, 37 from class B, 77 from class A, and 37 from class G.

It's important to note that each patient contributed multiple CT slices to the dataset, with the number of images varying by patient. Not all slices contained annotated bounding boxes for tumor locations; only annotated images were included in the final analysis.

This approach ensured that the final dataset was both manageable in size and representative of the original target distributions, avoiding the loss of target data. Table 3.1 summarizes the distribution of patients and images before and after sampling.

Table 3.1 - Patient and Image Distribution Before and After Sampling for Lung Cancer Groups: A (ADC); B (SCLC); E (LCC); G (SCC)

	Total Before Sampling			**Total After Sampling**	
Group	Patients	Images	Annotated Images*	**Patients**	**Annotated Images***
A	251	85.011	20.894	**77**	**3.529**
B	38	16.946	3.116	**37**	**3.021**
E	5	808	201	**5**	**201**
G	61	48.830	7.351	**38**	**3.476**

*Number of images that had corresponding bounding boxes with tumor localization

3.1.1.2. Demographic Patient Data

The clinical data, obtained from the TCIA LUNG PET-CT-DX dataset (Li et al., 2020), was provided in an Excel file, where each row represented a unique patient identified by the NewPatientID column. This identifier combined the cancer type and a patient number (e.g., A0001, where A represents ADC and 0001 is the patient number). The dataset included both demographic characteristics and clinical diagnosis information. The demographic variables consisted of Sex, Age, Weight (kg), and Smoking History, a binary column where 0 indicates a non-smoker and 1 indicates a smoker. The clinical diagnosis variables included T-Stage, N-Stage, M-Stage, and Histopathological Grading, providing details on tumor staging and classification for each patient.

For the 157 sampled patients, age ranged from 28 to 83 years, with a mean of 60.08 years (±10.18). The 25th percentile was 54 years, the median was 61 years, and the 75th percentile was 67 years, indicating a distribution skewed towards older patients. Weight data was

available for 155 patients, ranging from 35.5 kg to 98 kg, with a mean of 65.05 kg (±11.68). The median weight was 65 kg, with most patients falling between 57 kg (25th percentile) and 71 kg (75th percentile). Smoking history was recorded as a binary variable (0 indicates a non-smoker and 1 indicates a smoker), where approximately 47.8% of patients had a history of smoking (mean value of 0.48). In terms of categorical variables, all NewPatientIDs were unique, and the sex distribution was split into two categories, with males being the most common (91 male patients, compared to 66 females).

To minimize the impact of missing data, the sampling strategy aimed to exclude patients with incomplete records. However, to ensure the retention of all patients from certain histopathological classes and maintain the integrity of target variables, two missing values remained in the weight variable (representing 1.27% missing data).

3.1.2. Data: NSCLC-Radiomics

The addition of a new dataset was necessary due to the model's underperformance when trained exclusively with data from the LUNG PET-Dx dataset, particularly in distinguishing underrepresented classes. To address this issue and enhance model generalization, an additional dataset was incorporated to improve class representation. This complementary dataset included CT scans from 422 patients diagnosed with NSCLC. These patients were categorized into four histological subtypes: SCC (152 patients), LCC (114 patients), Not Otherwise Specified (NOS) (63 patients), and ADC (51 patients). Since the objective was to improve the performance of the model on underperforming classes, only patients from the LCC (class E) and SCC (class G) groups were selected, as these classes demonstrated poor predictive performance on the initial dataset.

Each scan in this dataset included a manual delineation of the gross tumor volume (GTV) performed by a radiation oncologist. In addition to the scans, the dataset contained DICOM Radiotherapy Structure Sets (RTSTRUCT) and DICOM Segmentation files, which provided detailed annotations of tumors and surrounding anatomical structures, including the lungs, heart, and esophagus. Furthermore, it included clinical data, allowing for the analysis of tumor characteristics and patient prognosis. As part of a broader study on radiomics, this dataset is publicly available on TCIA (Clark et al., 2013) under the name NSCLC-Radiomics (Aerts et al., 2019).

3.1.2.1. Image Extraction and Sampling Process

Due to computational constraints, it was not feasible to incorporate all images from the NSCLC-Radiomics dataset, as doing so would have resulted in an excessively large dataset. To ensure a manageable dataset size while improving model performance, a sampling process was conducted. For class E (LCC), all available patients from the NSCLC-Radiomics dataset were included, as this class had the poorest performance in the initial dataset. The original LUNG PET-CT-DX dataset contained only 201 annotated images for this class, making it significantly underrepresented. By adding all 114 patients and their corresponding images from the NSCLC-

Radiomics dataset, the total number of annotated images for this class increased significantly, aiming to improve the model's predictive capabilities.

For class G (SCC), the initial dataset already contained a relatively good number of images but lacked patient diversity, and the model struggled to differentiate this class from the others, leading to frequent misclassifications. To address this, an additional 512 images from 56 patients were incorporated from the NSCLC-Radiomics dataset. This selection introduced greater variation among patients aiming to help the model generalize better while remaining within computational limitations. Table 3.2 presents the total number of patients and images with bounding boxes available in the original NSCLC-Radiomics dataset, as well as those incorporated into the final dataset after sampling.

Table 3.2 - Patient and Image Distribution Before and After Sampling for Lung Cancer Groups E (LCC) and G (SCC) in the NSCLC-Radiomics Dataset

	Total Before Sampling		**Total After Sampling**	
Group	Patients	Annotated Images*	**Patients**	**Annotated Images***
E	114	2.026	**114**	**2.026**
G	152	2.735	**56**	**512**

*Number of images that had corresponding bounding boxes with tumor localization

3.1.2.2. Demographic Patient Data

The additional patient data incorporated from the NSCLC-Radiomics dataset included 10 demographic and clinical variables, namely Patient ID, age, clinical T-stage, clinical N-stage, clinical M-stage, overall stage, histology, gender, survival time, and death status event. Among these, age had 5 missing values (2.94%), and Overall Stage had 1 missing value (0.59%).

From the 170 sampled patients, the age distribution ranged from 33.7 to 91.7 years, with a mean age of 67.7 years and a median of 68.1 years. The clinical staging variables (T, N, and M stages) were recorded for all patients, with T-stage ranging from 1 to 5, N-stage from 0 to 4, and M-stage from 0 to 3. The most frequent overall stage was IIIB, occurring in 76 patients.

The survival time ranged from 10 to 4,328 days, with a mean of 1,071 days and a median of 637 days. While survival time was not used as a predictive variable for diagnosis, analysis of the death status event revealed that 87.1% of patients had a recorded death event. Lastly, the gender distribution consisted of 106 males and 64 females.

3.1.3. Data Pre-Processing

After sampling the imaging and clinical data, preprocessing steps were applied to prepare the datasets for model training. Image data were processed for compatibility with the YOLO object detection model through format conversion, resizing, normalization, and annotation

structuring. Clinical and demographic data were cleaned, imputed, encoded, and standardized to be used together with image features in a multimodal architecture based on ResNet for TNM stage classification.

3.1.3.1. Image Data

For the LUNG PET-CT-DX dataset, the preprocessing pipeline was designed to prepare DICOM CT images and their corresponding tumor annotations for compatibility with YOLO.

The DICOM images were converted to JPEG format, as YOLO requires images in standard formats. To ensure consistency and compatibility, all images were resized to 512 × 512 pixels, to balance image detail with computational efficiency.

Bounding box annotations, extracted from XML files, were scaled to match the resized image dimensions. The bounding boxes were normalized, with coordinates converted to relative values by dividing them by the target image size. Each bounding box was then formatted in the YOLO annotation format, which includes the class ID, normalized center coordinates, and normalized width and height. The target groups were assigned numerical class labels: ADC class A in the dataset (0), SCLC class B (1), LCC class E (2), and SCC class G (3). These annotations were saved as text files, with one annotation file per image.

The preprocessed images were stored in an "images" folder, while the corresponding annotation files were placed in a "labels" folder. Each file was named using the patient ID along with the image ID, maintaining traceability to the original dataset.

The NSCLC-Radiomics dataset required a slightly different preprocessing approach, as its bounding boxes were provided in a different format. Tumor annotations were extracted from RTSTRUCT DICOM files, which contained manual tumor delineations performed by radiation oncologists.

CT scan slices were organized in the correct anatomical order by sorting them based on their instance number metadata. Since CT scans consist of cross-sectional images forming a 3D representation of the patient's anatomy, they are not always stored sequentially. Sorting ensured the spatial relationships between consecutive slices were preserved, preventing misalignment issues. This step was critical for accurately mapping tumor annotations from the RTSTRUCT files to their corresponding images, as incorrect ordering could result in misplaced bounding boxes and inaccurate training data.

To ensure consistency with the LUNG PET-CT-DX dataset, all images were normalized and resized to 512 × 512 pixels. The number of classes remained unchanged to maintain compatibility between datasets. Each image was converted to JPEG format, and tumor bounding boxes were extracted, scaled to match the resized images, and formatted in the YOLO annotation format (.txt) with class ID and normalized coordinates.

The processed images and labels were stored in separate directories, following the same structure as the LUNG PET-CT-DX dataset, allowing for seamless integration into the training pipeline.

3.1.3.2. Data Cleaning (Demographic Patient Data)

To prepare the clinical data for integration with imaging data in the TNM stage prediction task, both datasets used in this study underwent data cleaning, imputation, encoding, and feature standardization.

In the LUNG-PET-CT-Dx dataset, three patients had invalid entries in the M-stage variable: two patients had an M-stage of 3, and one patient had an M-stage of 2. These values are not defined in the American Joint Committee on Cancer (AJCC) 8th edition (Amin et al., 2017) of the TNM staging system, which includes only M0 and M1 to indicate the absence or presence of metastasis. As a result, these patients were removed from the dataset to maintain consistency with clinically valid staging classifications.

In the NSCLC-Radiomics dataset, a similar issue was identified. Two patients presented inconsistencies in the M-stage variable. However, unlike the LUNG-PET-CT-Dx dataset, the NSCLC-Radiomics dataset included the overall stage classification for each patient. By cross-referencing the T and N stages with the provided overall stage, it was possible to infer the correct M-stage using the AJCC 8th edition staging criteria (Amin et al., 2017). The two patients were labeled with M3 but were classified at stage III overall. Based on the AJCC staging system, stage III corresponds to M0. Therefore, the M values for these patients were corrected to M0, rather than removed from the dataset.

Across both datasets, some demographic variables contained missing data. In the LUNG-PET-CT-Dx dataset, 1.27% of the weight values were missing, while in the NSCLC-Radiomics dataset, 2.94% of age values were missing. No missing data were observed in any other demographic or clinical variables. To address these gaps, missing values were imputed using the mean of the patient's corresponding cancer type group (e.g., ADC, SCC, SCLC), rather than using the overall dataset mean. This was done to preserve internal consistency within each class and avoid introducing bias, as age and weight distributions can differ across cancer types.

After handling missing values, variables were encoded and standardized for model training. Smoking status was encoded as 0 for non-smokers, 1 for smokers, and 2 for "not specified." The value 2 was used for patients from the NSCLC-Radiomics dataset, where smoking history was not provided. Numerical variables such as age and weight were standardized using z-score normalization to bring all values to a comparable scale.

To prepare the target variables for prediction, the T, N, and M stage labels were also processed. Due to insufficient representation across all possible subcategories, especially for T-stage labels, it was necessary to aggregate subcategories into broader classes. T1a, T1b, and T1c were all mapped to T1; T2a and T2b to T2; and so on. This reduced the T-stage classes to

T1, T2, T3, and T4. The M-stage variable was binarized, with M0 mapped to 0 and all M1 labels (M1a; M1b, and M1c) to 1. For the N-stage, no aggregation was needed, as it already consisted of four discrete numeric levels: N0 through N3.

This allowed both datasets to be merged and used for training the TNM stage prediction model, ensuring that input features and target labels were properly cleaned, imputed, and standardized. The final merged dataset included the target columns (T-stage, M-stage, and N-stage) along with the variables gender, smoking status, weight (kg), and age.

3.1.4. Data Splitting

To train YOLO for cancer-type detection and classification, the dataset was split into training, validation, and test sets to ensure proper evaluation and model training. The ratios for splitting were set at 70% for training, 15% for validation, and 15% for testing. The splitting process was designed to maintain a 70-15-15 ratio not only in terms of the total number of images but also in the number of patients. Additionally, the split ensured a balance across the four target classes, in all subsets. This approach guaranteed that the training, validation, and test sets were representative of the original dataset in both class distribution and patient diversity.

To achieve this, images were grouped by patient ID and class. Each patient's images were assigned to a single subset to avoid data leakage and to preserve statistical independence between the training, validation, and test sets. The training set was specifically designed to include a large variety of patients, ensuring that it captured the full spectrum of the dataset. This aimed to reduce the risk of overfitting by exposing the model to diverse data during training. The validation and test sets were structured to ensure balanced patient and image distributions across all four classes while maintaining comparable sample sizes.

When the NSCLC-Radiomics dataset was incorporated, the original LUNG-PET-CX dataset split was maintained, ensuring that no changes were made to its structure. However, the newly incorporated patients from the NSCLC-Radiomics dataset (E and G classes) were split separately, following the same logic as before—all images from a single patient remained within the same dataset split. This ensured that the E and G selected patients were divided into training, validation, and test sets using the same 70-15-15 ratio.

In the first trials, which only included the LUNG-PET-CX dataset, the training set contained 7494 images from 118 patients, while the validation and test sets contained 1375 and 1358 images, respectively, with 20 and 19 patients in each subset.

For the final dataset incorporating both LUNG-PET-CX and NSCLC-Radiomics data, table 3.3 summarizes the dataset split. Comprising a total of 13,765 images from 327 patients, the dataset was divided into 9,239 images for training (236 patients), 1,839 images for validation (46 patients), and 1,687 images for testing (45 patients) while ensuring a balanced representation of all four target classes within each set.

Table 3.3 - Distribution of Patients and Images Across Train, Validation, and Test Sets for Lung Cancer Groups A (ADC), B(SCLC), E (LCC), and G (SCC)

	Train		**Validation**		**Test**	
Group	**Patients**	**Images**	**Patients**	**Images**	**Patients**	**Images**
A	59	2.546	10	508	8	475
B	27	2.303	5	364	5	354
E	82 (3 from (1) 79 from (2))	1.455	19 (1 from (1) 18 from (2))	470	18 (1 from (1) 17 from (2))	302
G	68 (29 from (1) 39 from (2))	2.935	12 (4 from (1) 8 from (2))	497	14 (5 from (1) 9 from (2))	556
TOTAL	**236**	**9.239**	**46**	**1.839**	**45**	**1.687**

(1) LUNG-PET-CX dataset (2) NSCLC-Radiomics dataset

3.1.5. Data Augmentation

Data augmentation was applied to artificially expand the training and validation datasets by transforming existing images. This process aimed to introduce variations in the data while preserving the anatomical and diagnostic integrity of the CT scans. The augmentation helped balance class distributions and mitigate overfitting by exposing the model to a wider variety of image conditions. The same augmentation strategy was applied both in the initial experiments using only the LUNG-PET-CX dataset and in the final dataset incorporating also the NSCLC-Radiomics dataset. The transformations used remained consistent across both cases. Large rotations, extreme brightness adjustments, or contrast modifications were avoided to prevent distortions that could alter anatomical features.

For the final dataset incorporating both datasets, data augmentation was applied to the training and validation sets to ensure that the number of images per target class reached a value close to the class with the most images in each subset. This approach preserved class balance not by matching all classes to the size of the largest one, but by approximating a practical value that was sufficiently close to prevent imbalance. In the training set, all classes were augmented until they reached approximately 2950 images, close to the largest class (class G), which had 2935 images. In the validation set, augmentation was applied until all classes had approximately 510 images, aligning with Class A, which had 508 images in the validation set. This ensured that all classes had nearly equal representation without unnecessary over-expansion.

In the initial experiments, before incorporating the second dataset (NSCLC-Radiomics), augmentation was applied aiming to preserve anatomical features while addressing class

imbalance. Class E, which had only 201 images, was particularly underrepresented compared to other classes with over 3,000 images. To mitigate this, two augmentation pipelines were used: a standard pipeline for all classes and an intensive pipeline exclusively for Class E. The standard pipeline introduced minimal variations to maintain anatomical features as much as possible. For Class E, a more intensive augmentation pipeline with larger translations, scaling variations, and rotations were applied to increase sample diversity while ensuring the images remained diagnostically valid.

However, initial experiments revealed significant overfitting, indicating that the model was overly reliant on specific patterns in the training data rather than learning generalizable features. To address this, the final dataset incorporated more extensive augmentations to introduce greater variability in training images, aiming to reduce overfitting and improve generalization. As a result, the intensive augmentation pipeline previously used only for Class E was applied to all classes. The transformations remained within the limits necessary to preserve CT scan characteristics, ensuring the images resembled realistic medical data while still introducing diversity to improve generalization, as described in Table 3.4.

For the validation dataset, minimal augmentations were applied to maintain similarity with real-world conditions. This approach was used consistently across all experiments, including both datasets. The validation augmentation pipeline included only minor transformations such as small horizontal flips, slight brightness and contrast changes, and some affine transformations, as described in Table 3.5. No augmentation was applied to the test set, as it was kept unaltered to provide an unbiased evaluation of model performance.

Table 3.4 - Summary of the Data Augmentation Parameters Applied to the Training Set

Transformation	**Range/Details**	**Purpose**
Larger translations	5%-10% translation	Introduce spatial variation without distorting anatomy
Scaling variations	80%-120% scaling	Simulate different imaging scales while preserving structures
Rotations	-10° to 10° rotation	Avoid unrealistic spatial changes while adding a slight variation
Shear transformations	-5° to 5° shear	Ensure minor structural shifts without unnatural distortions
Blur adjustments	Increased blur to simulate scanner noise	Mimic imaging artifacts and scanner noise
Contrast adjustments	Adjustments to match imaging modality differences	Maintain contrast levels realistic to CT imaging

Table 3.5 - Summary of the Data Augmentation Parameters Applied to the Validation Set

Transformation	Range/Details	Purpose
Horizontal Flip	5% probability	Introduce minor variations in validation while maintaining real-world resemblance
Random Brightness/Contrast	3%-5% variation	Simulate slight imaging condition variations in the validation set
Affine Transformations	1%-2% translation, 95%-105% scaling, -2° to 2° rotation	Preserve anatomical integrity while adding small transformations

3.1.6. Models

Initially, the model selected for cancer-type classification was YOLO due to its efficiency in real-time object detection tasks. However, early experimental results did not meet performance expectations. As a result, further experimentation was conducted to enhance the model architecture. Among the strategies explored, one of the most promising was the integration of ResNet50 into the YOLO backbone, aiming to improve feature extraction and overall predictive accuracy. In parallel, the TNM stage prediction was conducted exclusively using the ResNet50 architecture, which was adopted for all experiments related to this task. The following sections describe the two main models tested in this study: YOLOv8 and ResNet50.

3.1.6.1. Yolov8

YOLOv8, developed by Ultralytics in 2023 (Muhammad Yaseen, 2024), is an improvement over YOLOv5, designed to enhance accuracy and efficiency in real-time object detection. It retains the core architecture of previous versions (see Annex I, Figure I.1), integrating localization and classification in a single model. The model consists of three key components: backbone, neck, and head. The backbone extracts hierarchical features from input images using CSPDarknet or depthwise separable convolutions, optimizing both speed and accuracy. These extracted features are then processed by the neck, which utilizes a Path Aggregation Network (PANet) to refine and integrate multi-scale information, improving detection across objects of different sizes. Finally, the head predicts bounding box coordinates, confidence scores, and class labels, employing an anchor-free approach that simplifies the model and adapts better to varying object shapes and sizes.

This study employed YOLOv8n (nano) and YOLOv8s (small) for their efficient balance of speed and accuracy. YOLOv8n (~2 MB in INT8 format) is suited for real-time applications with limited processing power, while YOLOv8s (~9 million parameters) enhances feature extraction through spatial pyramid pooling and PANet, making it ideal for tasks requiring higher accuracy

on standard hardware. Larger models, YOLOv8m, YOLOv8l, and YOLOv8x, offer better detection but require advanced GPUs due to their 25-90 million parameters. Given the available resources, YOLOv8n and YOLOv8s provide the best trade-off between performance and efficiency.

3.1.6.2. ResNet50

The ResNet-50 architecture (see Annex I, Figure I.2) is a DCNN designed to address the optimization challenges associated with training very deep networks. The core innovation of ResNet-50, as described by He et al. (2015), is the introduction of residual learning, where the network is constructed to learn residual functions regarding the layer inputs rather than directly learning the desired function. This is achieved through residual blocks, which incorporate identity shortcut connections that bypass one or more layers, allowing the network to maintain gradient flow and prevent the degradation problem that arises in deep architectures. The architecture consists of 50 layers, structured with convolutional layers, batch normalization, ReLU activations, and fully connected layers. A distinctive feature of ResNet-50 is its use of bottleneck residual blocks, which replace traditional convolutional layers with a three-layer structure, reducing computational cost while maintaining representational capacity.

ResNet-50 is built with an initial convolutional layer followed by four stages of residual blocks, each increasing in depth and feature dimensionality while reducing spatial resolution. The ResNet-50 model significantly improves accuracy and training efficiency, outperforming conventional architectures such as VGG-16 (Simonyan & Zisserman, 2015) while requiring fewer parameters and computational resources (He et al., 2015).

3.1.7. Implementation and Experimental Settings

Initially, the YOLOv8n and YOLOv8s models were tested on the LUNG PET-Dx dataset. However, both models exhibited convergence issues and did not achieve the expected performance. To address these issues, ResNet50 (He et al., 2015), and EfficientNetB1 (Tan & Le, 2020) were considered as alternative backbone architectures to improve performance.

For both backbone models, a hyperparameter search was conducted using Optuna (Akiba et al., 2019), focusing on different optimizers, schedulers, and hyperparameter ranges. The search process involved 50 trials to explore a wide range of hyperparameter configurations. After identifying the best-performing optimizer and scheduler, an additional 30 trials were conducted to refine the selection and determine the optimal hyperparameters. Once the optimal hyperparameters were identified, each model was trained for 200 epochs with an early stopping patience of 30 epochs.

ResNet50 yielded the best results among the two tested models and was subsequently integrated into the YOLO architecture. Due to computational constraints, the first four layers of YOLO were frozen, making only the stage 4 layer of ResNet50 trainable. Further

experiments to unfreeze additional layers could not be conducted due to resource limitations. Attempts were also made to integrate only stage 4 of ResNet50 into the YOLO structure (as this layer focuses on class-specific features essential for classification); however, this approach was computationally infeasible due to excessive processing time (since all layers were unfrozen). These integration experiments were performed exclusively with YOLOv8n (YOLO Nano) since the number of parameters and gradients in YOLOv8s made it impractical to run given the available computational resources.

For the Experiments with ResNet50 as YOLOnv8's backbone, batch size was limited to 8 due to memory constraints. The small batch size may have impacted model performance by reducing the stability of gradient updates and increasing variance during training. Despite the modifications, the improvements in YOLO's performance were marginal, and one class (class E) presented zero predictions in the test set.

To address this issue, additional data from a dataset containing NSCLC images was incorporated. Specifically, classes E and G, which performed the worst, were supplemented with additional images from the new dataset. This approach aimed to balance performance improvements with computational constraints. While limited by the number of additional images available, this experiment resulted in the best performance among all tested configurations.

Due to time constraints, experiments incorporating ResNet50 into the YOLO structure with the expanded dataset were not conducted. Additionally, batch size limitations would have remained a constraint in these experiments. The final set of experiments, incorporating additional data, was conducted using both YOLOv8n and YOLOv8s. Attempts were made to acquire additional data for the B class (SCLC); however, no suitable images matching the existing dataset were available. Consequently, performance for this class remained suboptimal.

For TNM stage classification, the data split used in the cancer type prediction task was reused to ensure a consistent distribution of cancer types across the training, validation, and test sets. Preserving this split aimed to facilitate the model's learning of stage-related patterns without introducing bias, while also maintaining a relatively balanced distribution of TNM stages across the subsets.

However, since each image contributed to all three TNM labels (T, N, M), the dataset remained imbalanced across classes within each label. To address this, the image augmentation pipeline from the cancer type prediction task was reused. In addition, selective augmentation and image removal were applied to reduce class overrepresentation. This process was guided by the distribution of all three labels associated with each image. Priority was given to augmenting data from patients with fewer available images to reduce the influence of individuals with larger image counts. As complete class balance could not be achieved, class weights were computed separately for T, N, and M based on their frequencies in the

augmented training set. These weights were incorporated into the loss function to increase the contribution of underrepresented classes during model training.

The final training set included 11,439 images, while 1,809 images were used for validation. The test set, consisting of 1,687 images, remained unchanged from the cancer type prediction task. The class distribution for each TNM target in the training, validation, and test sets is summarized in Appendix C, Tables C.4, C.5, and C.3, respectively.

The model employed for this task followed a multimodal ResNet architecture. The image data were processed using a ResNet50 backbone pretrained on ImageNet, while the demographic and clinical information, including gender, age, body weight, smoking history, and cancer type, was processed through a multilayer perceptron. The outputs from the image and clinical branches were then concatenated and passed through a fully connected layer that was shared across the network, followed by separate classification heads responsible for predicting the T, N, and M stages.

To optimize training, 50 Optuna trials were run with different hyperparameter combinations (see Appendix C, Table C.6) using the ResNet50-based model, evaluated on the validation set. The best configuration was then used to retrain the model on the full training set for up to 200 epochs, with early stopping (patience = 30) and checkpointing of the best-performing weights. The optimal hyperparameters for TNM prediction are summarized in Appendix C, Table C.6.

3.1.8. Evaluation Metrics

This section presents the evaluation metrics used to assess the performance of the YOLO-based object detection model and the ResNet50 model for the classification of four distinct object classes. The selected metrics, precision, recall, F1-score, specificity, accuracy, Intersection over Union (IoU), and mean Average Precision (mAP), were chosen due to their relevance in evaluating classification and object detection tasks. While precision, recall, F1-score, specificity, and accuracy apply to both models, IoU and mAP are exclusively used for evaluating object detection models such as YOLO due to their focus on localization accuracy and bounding box performance.

3.1.8.1. True Positives, False Positives, True Negatives, and False Negatives

Fundamental to all evaluation metrics are the following components:

- True Positive (TP): The number of samples correctly classified as belonging to the positive class (Hicks et al., 2022).
- False Positive (FP): The number of samples incorrectly classified as belonging to the positive class (Hicks et al., 2022).

- True Negative (TN): The number of samples correctly classified as not belonging to the positive class (Hicks et al., 2022).
- False Negative (FN): The number of samples incorrectly classified as not belonging to the positive class (Hicks et al., 2022).

3.1.8.2. Precision

Precision (PREC) is defined as the proportion of correctly classified positive samples relative to all samples predicted as positive. It measures the reliability of positive predictions by quantifying how many of the detected objects truly belong to the target class.

$$\mathrm{PREC} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}$$

Precision values range between 0 and 1, where 1 indicates that all predicted positive samples are correctly classified, and 0 indicates no correct positive predictions (Hicks et al., 2022).

3.1.8.3. Recall

Recall (REC), also known as sensitivity or the True Positive Rate (TPR), quantifies the ability of the model to correctly identify all positive instances within the dataset. It is defined as the ratio of correctly classified positive samples to the total number of actual positive samples.

$$\mathrm{REC} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}$$

The recall metric is especially critical in contexts where minimizing missed detections is important, such as medical and safety applications (Hicks et al., 2022).

3.1.8.4. F1 score

The F1-score (F1) is the harmonic mean of precision and recall, providing a balanced measure of a model's performance by considering both false positives and false negatives. It is particularly useful when there is an imbalance between classes, as it penalizes models with extreme values of precision or recall.

$$F1 = \frac{2 \times \mathrm{PREC} \times \mathrm{REC}}{\mathrm{PREC} + \mathrm{REC}} = \frac{2 \times \mathrm{TP}}{2 \times \mathrm{TP} + \mathrm{FP} + \mathrm{FN}}$$

The F1 score ranges from 0 to 1, where 1 indicates perfect precision and recall, and 0 indicates that the model fails to detect any relevant instances (Hicks et al., 2022).

3.1.8.5. Accuracy

Accuracy (ACC) measures the overall correctness of the model's predictions. It is defined as the proportion of correctly classified samples (both positive and negative) to the total number of samples in the dataset.

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{TN} + \mathrm{FN}}$$

Accuracy is one of the most reported metrics in machine learning applications but is known to be less reliable when dealing with imbalanced classes, where predictions favoring the majority class can skew the results (Hicks et al., 2022).

3.1.8.6. Intersection over Union

The IoU quantifies the overlap between the predicted bounding box and the ground-truth bounding box, as defined by:

$$\mathrm{IoU} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}$$

IoU values range between 0 and 1, where a higher value indicates better overlap. In object detection, an IoU threshold (typically 0.5) determines whether a predicted box is considered a correct detection (Padilla et al., 2020; Rainio et al., 2024).

3.1.8.7. Mean Average Precision

The mAP evaluates object detection by considering both the precision-recall tradeoff and localization accuracy. Specifically, mAP@0.5 calculates the mean precision across classes when the IoU threshold is set at 0.5, while mAP@[0.5:0.95] computes the average precision over multiple IoU thresholds ranging from 0.5 to 0.95 in increments of 0.05.

$$\mathrm{mAP} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{AP}_i$$

Where N is the number of classes, and APi is the Average Precision for class $\boldsymbol{i}$. This metric is useful for ensuring that both detection accuracy and precise localization are assessed (Padilla et al., 2020; Rainio et al., 2024).

3.2. REPORT GENERATION MODEL

This section presents the development of the report generation model, including the construction of the knowledge base, preprocessing of textual data, and implementation of the RAG pipeline. It also describes the embedding methods, retrieval strategies, language models, and evaluation procedures.

3.2.1. Knowledge Base

The knowledge base was gathered to support the RAG system, designed to generate treatment recommendations based on cancer type, TNM staging, and some demographic patient data. To achieve this, data sources were selected from internationally recognized and

certified entities to ensure coverage of lung cancer characteristics, risk factors, diagnostic criteria, staging systems, and treatment protocols.

The primary sources of information included medical organizations such as the American Cancer Society (ACS) and the National Cancer Institute (NCI) (American Cancer Society, 2025; and the National Cancer Institute, 2025). Web pages from these sources were collected to gather information on lung cancer classification, symptoms, diagnostic methods, staging criteria, survival rates, and treatment approaches. The collected data covered both NSCLC and SCLC, along with available treatment modalities such as surgery, chemotherapy, radiation therapy, targeted therapy, immunotherapy, and palliative care. Since lung cancer treatment protocols are determined by the stage of the disease, defined using the TNM staging system, the data collected primarily focused on stage-specific treatment guidance.

In addition to web page data, treatment guidelines from the National Comprehensive Cancer Network (NCCN) and the European Society for Medical Oncology (ESMO) (National Comprehensive Cancer Network, 2025; European Society for Medical Oncology, 2025) were also included. These documents were only available in PDF format and were processed separately (as described in section 3.2.2.1). These files contained clinical recommendations and treatment protocols for NSCLC and SCLC, also structured according to disease stage.

The final knowledge base consisted of 34 web pages and 3 PDF documents.

3.2.2. Data Pre-processing

To prepare the collected documents for use in the RAG system, a preprocessing pipeline was applied. This included the extraction of structured text from web pages and PDF files, followed by cleaning procedures to remove irrelevant elements, standardize content formatting, and ensure compatibility with downstream embedding and retrieval components.

3.2.2.1. Web Scraping and PDF Processing

Web scraping was performed using Selenium WebDriver (Selenium Contributors, 2024) to extract data from the selected websites. Selenium was chosen because it allows dynamic interaction with webpages, making it suitable for extracting content from sites that require JavaScript execution. Other libraries like BeautifulSoup (Richardson, 2023) were not sufficient for this task, as they only parse static HTML and cannot handle dynamically loaded content.

For scraping the data, ChromeDriver was used to control the Chrome browser for loading and interacting with the web pages. The browser was run in headless mode with non-essential features disabled to optimize performance. Pages were loaded with timeout handling, and metadata such as publication date and author information was extracted.

A total of 34 web pages were scraped. Their content was extracted and organized to preserve the original hierarchical structure and flow of information as presented in the original sources. The text content was structured hierarchically by extracting headings, paragraphs, and lists

while maintaining contextual relationships. The hierarchy was preserved by tracking the order of headings (h1 to h6) and associating each content element with its preceding heading level, ensuring that the document structure remained intact. Image captions were included to preserve relevant visual data as some pages provided image descriptions that had relevant information. Extracted data was stored in CSV and JSON formats, with CSV containing key page metadata and JSON containing the extracted document information for further analysis. The scraping process iterated through all URLs, applying these steps systematically. Error handling was also included to ensure that missing elements or loading failures did not interrupt execution.

In addition to web scraping, the three PDF files also underwent through preprocessing. The extraction was performed using pdfplumber, which allowed retrieval of the text while preserving the structure of the document. Each PDF was opened and processed page by page, extracting text while skipping empty pages. The processed text was then stored in JSON format, ensuring that each page's content was associated with its corresponding page number. This ensured that the information was retained in the original order and structure as it appeared in the original PDF files, maintaining consistency with the rest of the knowledge base.

3.2.2.2. Data Cleaning

Once the data was collected and stored in JSON files, a cleaning process was applied to remove irrelevant, duplicate, or inconsistent content.

The first step involved removing HTML tags from extracted text to ensure that only meaningful content remained. This was done using parsing techniques that detected and stripped HTML elements while preserving the underlying textual information. Excess whitespace and special characters were also removed to ensure uniform formatting.

To further improve data quality, navigation elements such as menu labels and footer links were removed. These were identified by comparing extracted text against a predefined list of common terms like "home," "about," and "contact." As such elements do not contribute to the meaningful interpretation of content, their removal helped retain only relevant information.

Short text fragments were also filtered out based on a word count threshold. These fragments were often incomplete, consisting of isolated words or phrases that lacked context or meaningful information. A threshold of three words was applied to exclude entries that were too brief to provide meaningful insights. These were typically artifacts of formatting, such as stray labels, list items, or broken sentences.

To maintain encoding consistency and ensure compatibility with embedding models and retrieval strategies, non-ASCII characters (e.g., accented letters or special symbols) were removed. Additionally, repeated headings that were often introduced by structural patterns

during extraction, were eliminated to reduce redundancy while preserving the original hierarchy and logical flow of the content. These steps ensured that paragraphs remained contextually linked to their respective sections, preventing the text from being fragmented or overly repetitive.

Finally, a manual verification was conducted to ensure that no irrelevant information remained. Citations and references, which frequently appeared within the body of the text, were manually removed due to their inconsistent formatting, which limited the effectiveness of automated filtering. No additional manual cleaning was performed beyond the removal of these references and citations.

3.2.3. Implementation and Experimental Settings

This section presents the implementation details and experimental settings of the different models evaluated within the RAG framework, including the embedding generation, retrieval strategies, and language models employed.

3.2.3.1. Embedding

To enable retrieval and ranking in the RAG system, text data was first converted into numerical vector representations known as embeddings. This process involved segmenting the cleaned text into chunks, generating embeddings using pre-trained models, and storing the resulting vectors in ChromaDB (Chroma, 2025), an open-source vector database for scalable retrieval.

The cleaned content scraped from each webpage was stored in separate JSON files. For the chunking process, each of the JSON files was split into multiple documents. Each document corresponded to a section of the webpage, defined by a heading (or title) and the content that followed it. This approach was used because webpage content was typically organized into distinct sections under headings, where the content across titles didn't fully interrelate. By treating each section as a separate document, the structure of the original page was better preserved, maintaining a clearer context in each segment.

Chunking was applied only to documents within each JSON file that exceeded the token limit of the embedding model. Documents shorter than the limit were kept as single chunks. This avoided unnecessary fragmentation of short sections and maintained their original coherence. For longer documents, overlapping chunking was applied with a 20% overlap to ensure continuity between chunks. Chunk sizes were set based on the input capacity of each embedding model: 1,000 tokens for the OpenAI and Google models, and 500 tokens for MiniLM.

For the JSON files containing content extracted from PDFs, this split into multiple documents based on titles was not possible. The structure of the text in PDFs was more continuous and not clearly separated by headings in a consistent way. Therefore, the entire content in each PDF-derived JSON file was treated as a single document and chunked directly based on token

length. The same chunking logic and model-specific token limits were applied, but without dividing the content into title-based sections.

This approach aimed at chunking the content in a way that best preserved the original context, making it easier to retrieve relevant information.

In this study, three embedding models were selected to evaluate how their use, in combination with different retrieval strategies, influenced the overall performance of the RAG system. OpenAI's text-embedding-ada-002 (OpenAI, 2024) was selected for its effectiveness in capturing semantic relationships over long sequences. Google's text-embedding-004 (Google DeepMind, 2024) was included for its public API availability and suitability for retrieval tasks. All-MiniLM-L6-v2 (Hugging Face, 2024) was used for local inference due to its compact size, which allowed it to run efficiently on limited hardware. Although smaller in capacity, MiniLM has demonstrated reliable performance in semantic similarity and retrieval tasks, offering a practical balance between speed and accuracy. These models were chosen based on a combination of accessibility, computational efficiency, and empirical evidence of strong performance in retrieval-based applications. More complex models, such as those based on BERT-large or other transformer architectures, were excluded due to their higher computational demands and latency, which were not compatible with the hardware and efficiency constraints of the system.

To improve processing efficiency, embeddings were generated in batches. Special handling was required for OpenAI's model due to strict API rate limits of 2,500 requests and 250,000 tokens per minute. Token counts were validated before each call, and a backoff strategy with exponential delays was implemented when rate limits were exceeded. Google's model was also accessed via an API, while MiniLM ran locally using PyTorch with hardware acceleration via MPS (Metal Performance Shaders). Once generated, all embeddings were stored in ChromaDB for use in retrieval operations.

3.2.3.2. Retrieval

For retrieval, three distinct methods were also implemented and tested: semantic retrieval with ChromaDB (Chroma, 2025), lexical retrieval using BM25 (Robertson & Zaragoza, 2009), and a hybrid approach that combines both.

For semantic retrieval, ChromaDB was used to compare queries against stored embeddings based on cosine similarity. This approach enabled the identification of documents with similar contextual meaning, even when the exact terminology varied. It was particularly suitable for retrieving clinical guidelines and research literature where different terms may refer to the same underlying concepts.

In contrast, BM25 was used for lexical retrieval. The text corpus was first tokenized, and BM25 scores were calculated by comparing query terms to the content of each document. Documents were then ranked by relevance based on these scores. This method favored exact

keyword matches and was especially effective for retrieving structured content such as medical protocols, diagnostic criteria, and standardized procedures where consistent terminology is typically used.

A hybrid method was also implemented to combine the strengths of both approaches. BM25 was used to retrieve documents with high lexical similarity, while ChromaDB returned results based on semantic relevance. The two sets were merged to ensure unique entries, and the final list was ranked based on relevance. This combined method aims to support the retrieval of treatment protocols by integrating precise keyword matching with a broader contextual understanding.

3.2.3.3. Language Models (GPT-4o Mini & Gemini 2.0 Flash)

For this study, two models, GPT-4o Mini (OpenAI, 2024) and Gemini 2.0 Flash (Google DeepMind, 2024), were tested within the RAG system to infer the cancer stage and generate the treatment protocols. Additional testing was performed using LLaMA 3.2 (Rozière et al., 2024) and DeepSeek (Qu et al., 2025) (*llama-3.2-1B*, *deepseek-vl-1.3b-chat*, and *deepseek-coder-1.3b-instruct*); however, due to the requirement for local execution, these experiments could not be fully completed within the available time and computational resources.

GPT-4o Mini (OpenAI, 2024) is a language model with a 128K token context window and support for outputs of up to 16K tokens per request. It was selected for this study due to its ability to process long sequences of text and perform reasoning tasks that are essential to clinical applications. These include interpreting medical content in context, understanding conditional relationships, and applying multi-step logical inference, which are capabilities necessary for following clinical guidelines and generating appropriate treatment recommendations. Within the RAG system, GPT-4o Mini was used to generate treatment protocols by combining retrieved content with clinical inputs, such as cancer type and TNM stage. Its balance of performance and resource efficiency also makes it suitable for healthcare applications.

Alongside GPT-4o mini, Gemini 2.0 Flash was also tested as part of the RAG system used to generate treatment protocols. Gemini 2.0 Flash (Google DeepMind, 2024) is a language model that supports text-based tasks and integration with external tools via function calling. It can process inputs such as plain text and code and is optimized for low-latency responses. The model can execute queries that involve both information retrieval and generation. Gemini 2.0 Flash was selected due to its accessibility through publicly available APIs, which allow integration without requiring local model deployment. Although it is a smaller variant compared to Gemini 1.5 Pro, it supports essential capabilities such as handling structured queries, maintaining conversational context, and applying basic reasoning over inputs. These features make it a practical choice for healthcare applications, particularly in resource-constrained environments where cost is a limiting factor.

3.2.3.4. Prompt

As part of the evaluation pipeline, a structured prompt was developed to simulate clinical reasoning for predicting lung cancer stages and generating treatment protocols (see Appendix D, Table D.1).

The objective of the prompt was to guide the system in performing two tasks: (1) to determine the clinical cancer stage based on the TNM classification and cancer type, and (2) to generate a corresponding treatment recommendation. The prompt was designed to process the output from the YOLO model (indicating cancer types, such as NSCLC or SCLC) alongside patient data, including age, gender, smoking status, and TNM stages. Since the system was not intended for typical question-answer interactions, but rather for automated reasoning based on inputs, the prompt had to be fixed and standardized to be correctly integrated into the pipeline.

Each prompt began by assigning the role of a lung cancer specialist to the model and clearly stating its task. The instruction was divided into two parts (see Appendix D, Table D.1). First, the model was required to determine the clinical stage using the AJCC 8th Edition lung cancer staging system, based on the provided TNM values. It is important to note that there is no complete consensus across all clinical guidelines when defining disease stages; different organizations may interpret specific T, N, and M combinations differently. To ensure consistent evaluation, the prompt strictly followed the AJCC 8th Edition as the reference framework. The model was not only expected to assign a stage (e.g., Stage IIB, IIIA) but also to justify its classification using staging logic consistent with this guideline.

In the second part, the prompt directed the model to generate a structured treatment plan appropriate to the identified stage and cancer type. The instructions reflected standard treatment pathways that vary by stage and differ between NSCLC and SCLC. For NSCLC, the prompt outlined stage-specific strategies, including surgical and non-surgical options, clinical trial opportunities, and palliative care. For SCLC, the model classified cases as limited or extensive stage and recommended standard regimens such as etoposide/platinum-based chemoradiation, with immunotherapy when appropriate. It also included considerations for older adults and detailed follow-up and supportive care. In all cases, the prompt required strict adherence to internationally accepted guidelines, including those from the NIH, NCCN, ESMO, and ASCO. The prompt concluded with a final instruction to enforce the technical question:

“Based on the patient data and TNM staging, what is the exact stage of the cancer and the indicated course of treatment?”

This prompt format was designed to ensure consistent behavior across all test cases and model configurations. Its structure aimed to support reproducibility across the different retrieval methods, embedding models, and language models tested during evaluation. Additionally, it was intended to ensure a uniform output format, allowing the treatment guidelines to be presented consistently for each disease stage and cancer type, enabling proper evaluation.

3.2.4. Evaluation

The evaluation pipeline was developed to assess the performance of the RAG system by systematically testing different combinations of embedding models, retrieval strategies, and language models to identify which configuration produced the most accurate and clinically relevant results. The evaluation focused on two main tasks: (1) predicting the correct cancer stage based on TNM descriptors and demographic patient data, and (2) generating a treatment recommendation appropriate for the predicted stage using information retrieved from clinical documents.

The evaluation used two separate datasets. The first was a patient dataset, consisting of individual records where each entry included a unique combination of TNM descriptors for each cancer type (NSCLC and SCLC), along with patient features such as age, gender, and smoking status. These records were used as a test set to construct the input prompts, simulating real patients to evaluate the system.

The second dataset was a reference CSV file, created manually, and used exclusively for evaluation. This file mapped each TNM combination and cancer type to the corresponding clinical cancer stage (e.g., Stage IIB, IIIA) and its associated standard treatment protocol, based on established lung cancer guidelines. It served as ground truth against which the outputs generated by the system were compared.

For each patient entry, a prompt was constructed using the TNM descriptors, cancer type, and demographic information. This prompt was passed to the document retrieval module, which returned the top-ranked documents relevant to the case. The three selected retrieval methods were tested. The retrieved documents were appended to the prompt to form the complete input for the language model, which then generated a predicted stage and treatment plan.

To support the retrieval process, the three selected embedding models were also tested. Each embedding model was paired with each of the three retrieval strategies, resulting in nine distinct retrieval configurations. These were then tested using one of the two language models (Google's Gemini 2.0 Flash or GPT-4o mini) to produce the final outputs.

Each unique configuration (defined as a combination of embedding model, retrieval method, and language model) was evaluated over ten independent runs for each patient entry in the dataset. This was done to reduce the impact of response variability and ensure statistical robustness. In each run, the predicted stage and treatment recommendation were extracted and compared to the ground-truth entries in the reference CSV file.

All run results were stored in a structured CSV file. These results were grouped and analyzed by embedding model, retrieval method, and language model to determine which configurations performed best. For the top-performing configurations, further experiments

were conducted to fine-tune system behavior. These included varying the number of retrieved documents (k) and adjusting the generation temperature to control randomness.

3.2.4.1. Evaluation Set

To evaluate the outputs generated by the system, a test set was manually created. This file served as the reference for verifying both the predicted cancer stage and the corresponding treatment recommendation. Its content was based on the AJCC 8th Edition lung cancer staging system (Amin et al., 2017) and standard treatment protocols associated with each disease stage.

The test set included all clinically valid combinations of TNM descriptors, as well as cancer type, covering both NSCLC and SCLC. Each row in the file represents a unique combination of these values and maps them to the appropriate consolidated stage classification, such as Stage IIA or IIIB. For each stage, a treatment recommendation is also provided, corresponding to standard care approaches as outlined in oncology guidelines.

For SCLC, the test set included age-based treatment stratification, as clinical guidelines provide distinct treatment recommendations for patients above and below the age of 70. While age can influence treatment decisions across all lung cancer types, only in SCLC do guidelines define specific alternatives based on this threshold. Accordingly, each of the two main SCLC stages (limited and extensive) was represented by two entries: one treatment reference for patients under 70 and one for patients 70 or older. These were the only stages where age-based treatment variation was included in the test set.

The test set was developed for two purposes: first, to serve as the ground truth against which the system's outputs could be compared; and second, to act as a reference for verifying whether the model's predicted stage and treatment aligned with expected clinical decisions. Its construction was guided by established oncology guidelines from sources including the NCCN, ESMO, NIH, and ACS. These same sources were incorporated into the retrieval component of the system, ensuring that the model was evaluated against the same body of knowledge available to it during generation.

Although care was taken to follow these guidelines closely, the dataset was created solely through the interpretation of publicly available materials, without clinical understanding or domain-specific knowledge of medical terminology and protocols. Therefore, while every effort was made to ensure that all available information was included and correctly interpreted, there may be omissions, simplifications, or minor errors resulting from limitations in clinical knowledge. Treatment decisions in oncology often depend on factors beyond TNM staging, such as biomarker status, other medical conditions, or individual patient factors. Therefore, the test set represents a simplified version of the treatment protocols and should be used only as a technical reference for evaluation. In addition, due to these limitations, the evaluation results may reflect certain biases.

Although the current version of the test set enables the evaluation of the RAG system, is not intended for clinical use. To ensure its validity for real-world applications or clinical studies, the test set would require review, correction, and approval by certified specialists in thoracic oncology or medical oncology.

3.2.4.2. Evaluation Metrics

To evaluate the system's performance, different metrics were used to assess both cancer-stage prediction and treatment recommendation outputs.

For stage classification, accuracy, precision, recall, and F1 scores were used to assess overall prediction quality. Cohen's Kappa coefficient was included to measure agreement between predicted and ground-truth stage labels while accounting for agreement expected by chance. This was relevant due to the large number of possible stage classes and class imbalance.

For treatment recommendations, the evaluation focused on both text similarity and meaning. ROUGE-L (Lin, 2004) was used to measure the overlap between the generated and reference treatments by identifying the longest common subsequence. It was chosen over BLEU (Papineni et al., 2001) because it better handles flexible sentence structures, which are important for clinical text where exact wording can vary. BERTScore (Zhang et al., 2020) was used to assess semantic similarity through contextual embeddings. Although ROUGE-L was also considered, BERTScore was the preferred metric for evaluating treatment recommendations, as it allowed greater flexibility in assessing responses that may have been clinically correct but phrased differently. This was especially important given potential gaps or simplifications in the test set, which may not have captured all valid treatment options or clinical scenarios for each cancer stage.

To assess the impact of retrieval on generation quality, RAGAS (Es et al., 2023; RAGAS, 2025) metrics were applied. Faithfulness measured whether the generated treatment was supported by retrieved content. Answer relevancy checked whether the treatment addressed the question in the prompt. Context precision evaluated the proportion of relevant information among retrieved content, while context recall assessed whether all necessary information was included.

3.3. HARDWARE

All experiments were conducted on a system equipped with an Apple M1 chip and 8GB of memory. For model training and optimization, the Metal Performance Shaders (MPS) backend was utilized to leverage hardware acceleration. However, YOLO models could not be trained using MPS due to compatibility issues, as YOLO's implementation relies on operations that are not fully supported by the MPS backend, leading to errors and unstable training behavior. As a result, YOLO models were trained using the CPU, which significantly increased training time and computational load.

4. RESULTS AND DISCUSSION

This section presents and discusses the main results obtained across the three core tasks of the study: cancer type classification, TNM staging, and treatment report generation.

4.1. IMAGE CLASSIFICATION MODEL

Table 4.1 summarizes the per-class bounding box performance metrics for each model evaluated on the test dataset in the lung cancer classification task. In the initial experiments using only the LUNG PET Dx dataset, performance across all configurations was limited. The YOLOv8n model achieved a test precision of 0.322, recall of 0.312, mAP50 of 0.271, and an F1 score of 0.28. YOLOv8s yielded comparable results with a slightly lower recall (0.298) and mAP50 (0.249). These outcomes indicated that the models were unable to effectively classify the four lung cancer classes in this dataset.

Table 4.1 - Performance Metrics of the Evaluated Models on the Testing Set

Model	Dataset	Batch size	Precision (B)	Recall (B)	mAP50 (B)	mAP50-95(B)	F1 score
YOLOV8n	LUNG-PET-Dx	16	0.322	0.312	0.271	0.137	0.28
YOLOV8s	LUNG-PET-Dx	16	0.325	0.298	0.249	0.116	0.29
YOLOV8n + ResNet50	LUNG-PET-Dx	8	0.317	0.239	0.239	0.11	0.27
YOLOV8n	**LUNG-PET-Dx + NSCLC-Radiomics**	**16**	0.483	**0.425**	**0.418**	**0.213**	**0.44**
YOLOV8n	LUNG-PET-Dx + NSCLC-Radiomics	32	**0.485**	0.374	0.383	0.195	0.41
YOLOV8s	LUNG-PET-Dx + NSCLC-Radiomics	16	0.455	0.387	0.382	0.2	0.41
YOLOV8s	LUNG-PET-Dx + NSCLC-Radiomics	32	0.432	0.326	0.338	0.175	0.37

Class G (SCC) was frequently confused with other classes, likely due to overlapping visual features since this class resemble a lot the remaining classes, especially class B (SCLC). Class E was never detected in any of the initial trials and was consistently predicted as background. This was not due to object size as class E tumors were generally large and visually distinct, but rather due to extremely limited representation: only 43 test samples and 201 total images of class E were available in the dataset. The combination of this imbalance and the small dataset size likely caused the model to ignore this class during training. As a result, predicted bounding boxes for this class showed confidence scores near zero.

The ResNet50 backbone was evaluated with YOLOv8n to enhance feature representation. While training showed more stable learning behavior, the model did not converge, and test performance did not improve. It achieved a test precision of 0.317, recall of 0.239, and mAP50 of 0.239. The decline in recall and sustained low mAP suggest that the increased model capacity may have led to overfitting, particularly given the small and imbalanced training data. Due to computational and time constraints, further experiments with stronger regularization strategies, such as higher weight decay or dropout, were not conducted but could potentially mitigate overfitting effects in future work. Notably, the confidence scores of predicted bounding boxes remained low, indicating poor generalization beyond the training set.

Subsequent experiments incorporated a combined dataset comprising LUNG PET Dx and NSCLC Radiomics. This data augmentation improved class representation, particularly for the classes that showed the weakest performance previously. For example, the YOLOv8n model trained on the combined dataset with batch size 16 reached a precision of 0.483, recall of 0.425, mAP50 of 0.418, and F1 score of 0.44. These improvements indicate that broader class diversity and increased sample size allowed the model to better distinguish class-specific patterns and reduce misclassification, especially for class E. Recall and mAP50 improved by more than 10 percentage points relative to the single-dataset trials, confirming that limited training data had been a significant bottleneck.

Batch size also influenced performance. Increasing the batch size from 16 to 32 generally reduced recall and mAP. For YOLOv8n, recall dropped from 0.425 to 0.374 and mAP50 from 0.418 to 0.383. YOLOv8s showed a similar trend, with recall decreasing from 0.387 to 0.326 and mAP50 from 0.382 to 0.338. This suggests that while larger batches may stabilize gradients during training, they may also reduce the frequency with which minority or difficult samples are seen per update step, resulting in lower sensitivity to those classes.

For the experiments with both datasets, Weight decay was systematically adjusted to address overfitting in both nano and small model variants. Experiments compared decay values of 0.0005 (default), 0.0007, and 0.003. The value of 0.0007 consistently yielded the best results across models trained on the combined dataset, as it provided effective regularization without constraining the learning capacity. A decay of 0.003 resulted in underfitting, while 0.0005 was insufficient to prevent overfitting. All models presented in Table 4.1 using the combined datasets were trained with a weight decay of 0.0007, as this configuration produced the most balanced and stable outcomes.

In summary, the YOLOv8n model trained on the combined dataset with a batch size of 16 and weight decay of 0.0007 obtained the highest performance, with a test mAP50 of 0.418 and an F1 score of 0.44. This configuration also showed low processing times, with 0.4 ms for preprocessing, 127.8 ms for inference, 0 ms for loss computation, and 0.2 ms for postprocessing per image.

Table 4.2 - Classification performance report for the YOLOv8n model (batch size = 16) trained using both datasets. Results include per-class bounding box metrics.

Class	Images	Precision (B)	Recall (B)	mAP50 (B)	mAP50-95(B)
A (ADC)	475	0.534	0.6	0.499	0.219
B (SCLC)	354	0.429	0.206	0.317	0.163
E (LCC)	302	0.519	0.45	0.488	0.28
G (SCC)	556	0.451	0.444	0.367	0.19

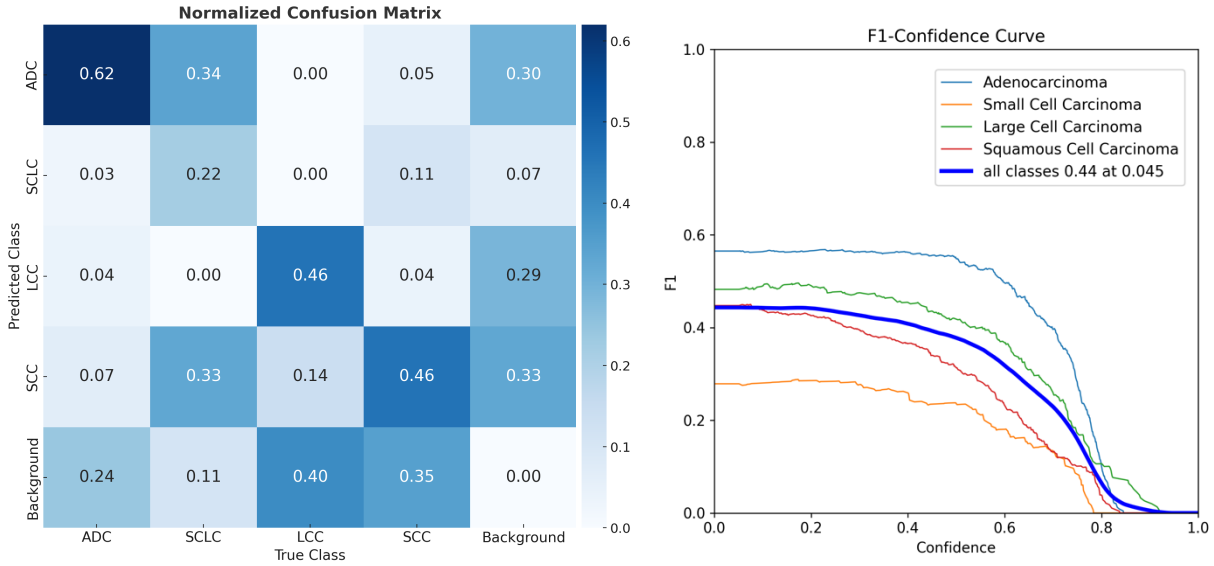


Figure 4.1 - Confusion matrix and F1-confidence curves for the YOLOv8n model (16 batch size) trained on the combined dataset. The confusion matrix (left) shows per-class normalized detection accuracy. The F1-confidence curves (right) indicate F1-score variation by confidence threshold, with the best global F1 of 0.44 achieved at 0.045.

Table 4.2 presents the per-class performance metrics and provides further insight into the behavior of the best-performing model (YOLOv8n, combined dataset, batch size 16, weight decay 0.0007) in distinguishing between lung cancer classes in the test set.

ADC achieved the highest performance among all classes, with a precision of 0.534, recall of 0.6, and mAP50 of 0.499. According to the normalized confusion matrix (Figure 4.1), 62% of ADC samples were correctly classified, while the most frequent misclassifications occurred towards background (24%), with very limited confusion with other classes, all below 10%, the highest being towards SCC (7%). The F1-confidence curve shows that predictions for this class remained stable and robust across a wide range of confidence thresholds.

LCC reached a precision of 0.519, recall of 0.45, and mAP50 of 0.488. The model correctly classified 46% of LCC samples, with the most frequent misclassifications being into background (40%) and SCC (14%), as shown in Figure 4.1.

SCC achieved a precision of 0.451, recall of 0.444, and mAP50 of 0.367. The confusion matrix (Figure 4.1) indicates that 46% of SCC samples were correctly classified, while a considerable proportion of errors were towards background (35%) and LCC (14%). The F1-confidence curve shows a reduction in F1-score at higher confidence thresholds, indicating lower prediction certainty for this class.

SCLC exhibited the lowest performance among all classes, with a precision of 0.429, recall of 0.206, and mAP50 of 0.317. Only 22% of SCLC samples were correctly classified, while most errors were towards background (40%) and SCC (33%). This low performance is consistent with the F1-confidence curve (Figure 4.1), where the F1-score remains well below 0.3 across all thresholds. The limited representation of SCLC in the training data likely explains this result, as no additional SCLC images were included during dataset augmentation, reducing the model's exposure to relevant patterns of this class.

Overall, the F1-confidence curves shown in Figure 4.1 support these findings. ADC and LCC achieved the highest F1-scores, particularly at lower and intermediate thresholds, reflecting the model's ability to distinguish these classes effectively when allowing for lower confidence predictions. However, it is possible to observe a gradual reduction in F1-score as the confidence threshold increases, especially for ADC and LCC. This behavior suggests that, although the model was able to detect these classes well, many of the correct predictions are associated with lower confidence values. This is likely a consequence of high intra-class variability and the presence of challenging or borderline samples within these classes, which leads the model to assign lower confidence scores even when the prediction is correct.

For SCC, the F1-score remains moderate but shows a deeper decline as the confidence threshold increases, indicating that many of its correct detections are also linked to lower confidence levels. This can be explained by the visual similarity and overlapping characteristics of SCC with other classes, which introduces ambiguity in the feature space learned by the model.

In the case of SCLC, the F1-score remained consistently low across all confidence thresholds. This is indicative of the insufficient representation of SCLC during training, preventing the model from learning distinctive and reliable features for this class. Consequently, the few correct predictions made for SCLC tend to have low confidence.

Thus, the overall shape of the F1-confidence curves provided important insights into the model's behavior: classes with more training data and clearer distinguishing features maintain higher F1-scores across a broader range of confidence thresholds, while classes affected by data imbalance or high similarity to other classes exhibited sharper declines in performance as stricter confidence levels are applied.

In conclusion, the best-performing model showed a good capacity to distinguish ADC and LCC, due to their greater representation in the training dataset and more distinct characteristics. The performance for SCC was lower due to class overlap and prediction uncertainty. SCLC was

the most challenging class, primarily due to its low number of training samples and feature similarity with other classes.

These findings reinforce the importance of data quantity, diversity, and class balance in supervised object detection tasks. In particular, the results highlight that increasing the variety and number of images used for training had a greater impact on model performance than modifications in architecture or hyperparameter tuning. The exposure of the model to more and varied images proved fundamental to improving its ability to detect and distinguish between the different lung cancer types.

4.1.1. TNM Staging Model

For the TNM staging classification task, training results indicated rapid learning, with high F1 values on the validation set. However, overfitting appeared early, as the model reached near-perfect performance on the training set within only a few epochs. Regularization techniques were applied to both image and demographic inputs, including dropout and weight decay. Dropout was applied more intensively to demographic features because each patient had only one set of demographic values, which were repeated across all corresponding image slices. Despite these measures, overfitting persisted, likely due to class imbalance across all TNM targets and the uneven representation of patients in the dataset.

The final model achieved a mean F1-score of 0.3890 on the test set, with significant variation in performance across the T, N, and M components. The M component yielded the highest performance, with an accuracy of 0.6959 and an F1-score of 0.5815. This can be attributed to the binary nature of the M classification (M0 vs. M1), where class 0 (M0) was dominant. Moreover, the clinical and morphological differences between M0 (non-metastatic) and M1 (metastatic) tumors are generally more distinct, which likely contributed to clearer separability in imaging data.

In contrast, the T target achieved the lowest performance, with an F1-score of 0.2358 and an accuracy of 0.2893. The confusion matrix showed frequent misclassifications across T stages. Particularly, class 2 presented a recall of only 0.02 (see Appendix C, Table C.7), which is likely due to a combination of small sample size and high intra-class variability. Unlike the M classes, the visual and anatomical characteristics between T stages tend to be more similar and less distinct (see Appendix C, Table C.8), making classification more difficult.

The N component yielded intermediate results, with an F1-score of 0.3497. While the model was able to differentiate between some N stages, confusion between adjacent classes remained common. This may be explained by the overlapping spatial and pathological features associated with lymph node involvement. The N classification refers to the extent of regional lymph node spread, which can present gradual rather than discrete differences across stages.

Overall, these results suggest that the model's performance was strongly influenced by the complexity of class definitions, the balance of training samples across target stages, and the degree of visual separability between categories. The persistent overfitting and inconsistent performance suggest the need for several methodological improvements.

4.2. TREATMENT GENERATION MODEL

Table 4.3 presents the evaluation metrics for cancer stage prediction and treatment generation across different combinations of embedding models, retrieval methods, and LLMs. Metrics include Stage Match (mean), BERTScore F1, and four RAGAS components: Faithfulness, Answer Relevancy, Context Precision, and Context Recall.

When comparing the LLMs independently, GPT-4o consistently achieved the highest BERTScore F1 values, reaching up to 0.843, indicating strong semantic similarity between generated and reference responses. Gemini, while slightly lower in BERTScore (ranging from 0.824 to 0.832), outperformed GPT-4o in cancer stage prediction, achieving a maximum stage match mean of 0.606 in the configuration using Gemini embeddings and cosine retrieval. By contrast, the best GPT-4o configuration reached a stage match mean of 0.506 with Gemini embeddings and a combined retriever. Despite this high alignment in predicted stages, the Gemini embeddings + Cosine + Gemini LLM setup performed poorly in other metrics, including RAGAS Faithfulness (0.317), Context Precision (0.752), and Context Recall (0.345), limiting its overall usefulness. As stage match can be influenced by prompt design, greater importance was placed on factual grounding and contextual relevance. Therefore, this configuration was not included among the top-performing combinations.

The most balanced overall results were achieved using Gemini embeddings, the combined retriever, and the Gemini LLM. This configuration yielded a Stage Match mean of 0.556, BERTScore F1 of 0.827, Context Precision of 0.994, and Context Recall of 0.911, with strong RAGAS Faithfulness (0.552) and Answer Relevancy (0.807). Another well-performing configuration was OpenAI embeddings with the BM25 retriever and the Gemini LLM, which achieved a Stage Match mean of 0.550, Faithfulness of 0.566, Answer Relevancy of 0.793, and Context Precision and Recall of 0.994 and 0.885 respectively. These results reflect strong alignment between predicted treatments and their source contexts.

Overall, the Gemini LLM produced more accurate outputs in terms of treatment-stage alignment, while GPT-4o produced more semantically fluent responses. Cosine-based retrieval improved semantic fluency and stage match, especially when paired with OpenAI or Gemini embeddings, but often led to reduced Context Recall and Faithfulness. The combined retriever offered the most consistent results across all metrics and was therefore considered better suited for treatment generation.

Based on performance across all evaluation dimensions (stage prediction, semantic similarity, factual consistency, and contextual grounding), the three best configurations selected for further testing were: (1) Gemini embeddings + combined retriever + Gemini LLM;

(2) OpenAI embeddings + BM25 retriever + Gemini LLM; (3) OpenAI embeddings + combined retriever + Gemini LLM.

Table 4.3 - Evaluation metrics for treatment generation across combinations of Embedding Model, Retrieval Method, and LLM (Top-K = 7; Temperature =7), with the three configurations achieving the highest overall performance highlighted.

LLM	Embbe-ding Model	Retrieval method	Stage match *	BERT-Score F1	RAGAS Faith-fullness	RAGAS Answer Relevancy	RAGAS Context Precision	RAGAS Context Recall
Gemini	Gemini	combined	0,556	0,827	0,552	0,807	0,994	0,911
Gemini	OpenAI	BM25	0,550	0,827	**0,566**	0,793	0,994	0,885
Gemini	OpenAI	combined	0,539	0,827	0,544	0,800	0,994	0,899
Gemini	Gemini	BM25	0,528	0,826	0,544	0,803	0,994	0,892
Gemini	OpenAI	cosine	0,547	0,832	0,504	0,774	0,962	0,958
Gemini	MiniLM	combined	0,572	0,830	0,456	0,797	**0,998**	0,489
Gemini	MiniLM	BM25	0,558	0,829	0,441	0,780	**0,998**	0,479
Gemini	MiniLM	cosine	0,517	0,824	0,349	0,781	0,755	0,469
Gemini	Gemini	cosine	**0,606**	0,824	0,317	0,774	0,752	0,345
GPT-4o	OpenAI	combined	0,494	0,842	0,476	0,859	0,994	0,924
GPT-4o	Gemini	BM25	0,497	0,842	0,476	0,860	0,993	0,921
GPT-4o	Gemini	combined	0,506	0,842	0,476	0,855	0,993	0,901
GPT-4o	OpenAI	BM25	0,486	0,841	0,464	0,860	0,993	0,905
GPT-4o	OpenAI	cosine	0,450	0,843	0,267	**0,863**	0,964	**0,976**
GPT-4o	MiniLM	BM25	0,483	0,842	0,292	0,862	**0,998**	0,486
GPT-4o	MiniLM	combined	0,464	0,842	0,278	0,860	**0,998**	0,471
GPT-4o	MiniLM	cosine	0,453	**0,844**	0,211	**0,863**	0,759	0,487
GPT-4o	Gemini	cosine	0,519	0,843	0,192	0,858	0,756	0,322

* Average proportion of correct cancer stage predictions, computed as a binary value per instance (1 for a correct match, 0 otherwise), averaged over 10 runs per model, retrieval, and language model configuration

These combinations were subsequently evaluated under different Top-K and temperature settings to identify the best configurations. Top-K (the number of documents retrieved per

query) was tested at values of 5, 7, 10, and 15. Temperature (which controls randomness in the model's output) was varied at 0.0, 0.3, 0.5, and 0.7 (see Appendix E).

For the Gemini embeddings + combined retriever + Gemini LLM configuration (Combination 1, Table 4.4), the best results were achieved at Top-K = 15 and temperature = 0.0, with a Stage Match of 0.54, BERTScore F1 of 0.83, RAGAS Faithfulness of 0.68, and Context Recall of 0.93, yielding an average score of 0.796 (calculated as the mean of all evaluated metrics). The same configuration with lower Top-K settings (e.g., 10) resulted in comparable semantic alignment but reduced factual grounding, particularly in Faithfulness (see Appendix E, Tables E.4 and E.5).

In the OpenAI embeddings + combined retriever + Gemini LLM configuration (Combination 2, Table 4.4), the optimal setting was also Top-K = 15, but with a temperature of 0.7. This setup yielded a Stage Match of 0.54, Faithfulness of 0.65, and Context Recall of 0.96, with the same mean score of 0.796. While semantic and contextual performance remained strong, Faithfulness was slightly lower, and the higher temperature introduced more variation in outputs.

The OpenAI embeddings + BM25 retriever + Gemini LLM configuration (Combination 3, Table 4.4) performed best at Top-K = 15 and temperature = 0.7, reaching a Stage Match of 0.51, Faithfulness of 0.66, Context Recall of 0.98, and a mean score of 0.795. While factual and contextual metrics were high, stage alignment remained slightly below that of the combined retriever setups.

Although Combinations 1 and 2 achieved equal mean scores (0.796), they reflect different trade-offs. Combination 1 used a lower temperature (0.0) and produced higher Faithfulness with equivalent Stage Match, which may be preferred in tasks requiring deterministic outputs and strict factual adherence, such as treatment protocol generation. In contrast, Combination 2 showed stronger context retrieval but at a higher temperature (0.7) and slightly reduced factual reliability. Given the application context, where output consistency and factual accuracy are prioritized, Combination 1 (Gemini embeddings + combined retriever + Gemini LLM, Top-K = 15, temperature = 0.0) offers the most balanced and reliable configuration.

Table 4.4 - Performance Comparison of RAG Configurations Under Varying Top-K and Temperature Settings

Combination	**Top-K**	**Temperature**	**Stage match**	**BERT-Score F1**	**Faithfullness**	**Answer Relevancy**	**Context Precision**	**Context Recall**
(1)	15	0.0	0,54	0,83	0,68	0,81	0,99	0,93
(2)	15	0.7	0,54	0,83	0,65	0,81	0,99	0,96
(3)	15	0.7	0,51	0,83	0,66	0,80	0,99	0,98

5. CONCLUSION AND FUTURE WORKS

This study developed and evaluated an integrated framework for lung cancer analysis across three tasks: cancer type classification, TNM staging, and treatment recommendation Appendix F for an example output of the full pipeline). The results showed that combining imaging and structured data can support automated analysis, though several limitations were identified throughout.

In the classification task, the YOLOv8n model trained on the combined LUNG PET Dx and NSCLC Radiomics datasets (batch size = 16, weight decay = 0.0007) yielded the strongest results, with an F1 score of 0.44 and mAP50 of 0.418. ADC and LCC were more accurately identified, while SCLC remained the most difficult to detect, likely due to low representation and visual overlap with other tumor types.

In TNM staging, the model reached a mean F1 score of 0.389. Performance was highest for the M component, reflecting its binary structure and clearer imaging distinctions. The T and N targets showed lower results, influenced by class imbalance, visual similarity among stages, and the method used to integrate demographic and image data. Overfitting and data redundancy also limited generalization.

In the treatment generation task, the top-performing configuration combined Gemini embeddings, a combined retriever, and the Gemini LLM with Top-K = 15 and temperature = 0.0. It achieved a stage match of 0.54, BERTScore F1 of 0.83, Faithfulness of 0.68, and Context Recall of 0.93, resulting in the highest overall score (0.796). While GPT-4o showed better semantic similarity, it was less consistent in aligning treatments with clinical stage. Notably, combined retrieval methods and higher Top-K values enhanced both factual grounding and contextual relevance, while lower temperatures (0.0–0.3) ensured more stable and faithful outputs which is essential for clinical applications.

Throughout this study, several limitations were identified across the model development, data processing, and evaluation stages. These issues impacted the performance and generalizability of the system across the three main tasks: lung cancer classification, TNM staging, and treatment protocol generation.

The following section outlines key limitations observed during model development, data handling, and evaluation, and proposes directions for future research to address these constraints and strengthen the system's clinical applicability.

5.1. LIMITATIONS

The image classification model based on YOLOv8n encountered several limitations related to data availability, model design, and computational resources. The primary constraint was the imbalance in the dataset, particularly for SCLC (class B) and LCC (class E). These classes were significantly underrepresented, which impaired the model’s ability to generalize across all

cancer types. Although additional samples were integrated, the number of images for SCLC remained limited due to the absence of suitable public datasets with the necessary imaging features. As a result, SCLC became one of the lowest-performing classes in the final model. Additionally, not all available images for SCC (class G) could be used, as hardware limitations restricted dataset size. This further affected model performance, especially for a class known to share visual characteristics with other cancer types, increasing the likelihood of misclassification.

Another issue involved the presence of small bounding boxes in slices taken from tumor margins. These regions often lacked sufficient anatomical context, leading to errors in classification or omission during detection. These slices were retained to simulate real clinical settings where all available patient data must be processed, but their inclusion introduced detection challenges.

From a training perspective, convergence issues were observed when trying to modify the model's structure. While integrating ResNet50 with YOLOv8n improved training stability, only the final stage of ResNet50 could be fine-tuned due to memory limitations. The batch size was constrained to 8, reducing gradient stability and training efficiency. More complex backbones, such as DenseNet, could not be tested due to hardware constraints. EfficientNetB1 was evaluated but underperformed relative to ResNet50. Deeper EfficientNet variants were not tested for similar memory-related reasons.

For the TNM staging task, both data and modeling limitations reduced performance. Key demographic variables such as age and weight were missing for a subset of patients and had to be imputed, which may have introduced bias. Moreover, several TNM classes were underrepresented, particularly the subcategories within each stage. Due to the lack of sufficient samples, detailed classes, for example, T1a and T1b, were grouped into a single category (T1) to ensure enough training data per class. This merging improved class balance but reduced the level of detail in the model's predictions, limiting the precision of the staging output.

The model also exhibited clear signs of overfitting. Test performance did not reflect the high accuracy observed on the validation set, indicating poor generalization. Extensive regularization, including dropout and weight decay, was applied to both image and demographic inputs, with even stronger dropout on demographic features due to their repetition across all slices from a given patient. Despite these efforts, overfitting remained a persistent issue. One contributing factor was the repeated use of the same demographic vector across multiple image slices for a single patient, which introduced redundancy and possibly overemphasized non-imaging features. As each slice could contribute differently to the T, N, and M targets, data augmentation could not be applied uniformly across patients. This further amplified class imbalance and input redundancy, which compromised model generalization.

In the treatment generation task, limitations arose from dataset construction, content inconsistency, and model deployment. The evaluation set was manually created without expert clinical validation, increasing the risk of content inaccuracies. This was further compounded by the absence of lung cancer specialists, who could have supported both dataset development and output evaluation. Without expert input, it was not possible to fully assess the clinical accuracy or relevance of the final model's recommendations.

An additional limitation involved cancer staging consistency in the retrieved documents used for treatment generation. While the project adopted the AJCC 8th (Amin et al., 2017) edition as the reference standard for TNM staging, many retrieved documents, such as clinical websites or manuals, used alternative staging conventions or aggregation rules. These inconsistencies likely introduced errors in stage attribution, with the model sometimes generating treatments based on incorrect or mismatched staging schemes. This variability undermined the alignment between predicted treatments and the correct TNM stage classification. Moreover, evaluation relied primarily on token-level metrics such as BERTScore and RAGAS components (Faithfulness, Relevancy, Context Precision, Context Recall), which may not capture true clinical utility.

Language models such as DeepSeek and LLaMA were also considered for the treatment generation pipeline. However, they were excluded from experimentation due to the need for local deployment and significant computational resources. As such, their potential contribution could not be evaluated in this study.

Finally, the use of LLMs, including GPT-4o and Gemini, introduced transparency and interpretability challenges. These models operate as black boxes with undocumented internal mechanisms (Hemasri et al., 2024; Zhang, Shi, & Kamel Boulos, 2024). Known issues such as factual inconsistencies, hallucinated content, and lack of explainability remain unresolved. Recent literature supports these concerns. Hemasri et al. (2024) emphasized the ethical and security challenges of using LLMs in healthcare and highlighted the need for transparent validation protocols. Zhou et al. (2024) reported on bias and integration complexities in clinical applications. Zhang, Shi, and Kamel Boulos (2024) discussed how prompt sensitivity, and proprietary architectures affect clinical reliability and model trustworthiness.

5.2. FUTURE WORKS

The limitations identified in this study suggest multiple directions for future development. A key priority is the expansion and diversification of the dataset, particularly to address the class imbalance observed in tumor classification. The model's performance was significantly limited by the underrepresentation of SCLC and had difficulty distinguishing SCC from other classes. Addressing these limitations will require the collection or sharing of larger, more balanced datasets, ideally through data-sharing agreements, federated learning frameworks, or institutional collaborations that preserve data privacy.

Another limitation concerns the processing of CT slices. In this pipeline, each slice was treated independently, without accounting for spatial continuity or anatomical structure across the full scan. This approach was particularly problematic for peripheral tumor slices, which often contained small or ambiguous bounding boxes. Without contextual information from adjacent slices, these regions were more prone to misclassification. A potential improvement is the use of tracking-based methods to maintain consistency across slices, as demonstrated by Whebe et al. (2023). In their work, YOLO was combined with ByteTrack (Zhang et al., 2022), a multi-object tracking algorithm originally developed for video analysis. When applied to CT scans, this method could improve detection accuracy by preserving object continuity across consecutive slices. Additionally, 3D models such as volumetric convolutional networks or spatially aware transformer architectures (Chen et al., 2021; Hatamizadeh et al., 2021; Milletari et al., 2016) could be explored to better capture inter-slice relationships.

Model architecture is another area for improvement. The YOLOv8n and ResNet50 configuration tested in this work was limited by hardware constraints, which allowed fine-tuning of only the final ResNet layer. This reduced training flexibility and affected convergence. Future work should consider alternative backbones such as ConvNeXt (Liu et al., 2022), MobileViT (Mehta & Rastegari, 2022), or EfficientNetV2 (Tan & Le, 2021), which offer more favorable performance-to-resource trade-offs. Techniques such as gradient checkpointing, mixed-precision training, and gradient accumulation could support larger batch sizes and more extensive training in constrained computational environments.

For the TNM staging task, modifications to the integration of multimodal inputs are also necessary. In the current setup, demographic vectors were repeated across all slices from the same patient, introducing redundancy and contributing to overfitting. Future pipelines should explore late fusion or patient-level feature aggregation to mitigate this issue. Stronger regularization, such as label smoothing, dropout, or architectural constraints, may also support improved generalization.

Balancing the distribution of classes across the T, N, and M targets is also essential. Some categories were underrepresented, limiting the model's ability to learn intra-class variation. Class weighting and focal loss should be applied more effectively, and data augmentation strategies should be revised to ensure consistent representation across patients, especially when different slices contribute differently to TNM components.

Multimodal transformer-based models (Bannur et al., 2023; Hayat et al., 2023; Chen et al., 2024) may also be explored. These models can jointly process image and structured data, directing attention to anatomically meaningful areas. While they offer potential benefits for staging, their impact on model interpretability should be carefully evaluated.

In the treatment generation task, the most immediate need is the development of a clinically validated evaluation dataset. The current test set was constructed manually without expert review, raising concerns about consistency and accuracy. Future datasets should include key

clinical variables (such as comorbidities, biomarker status, treatment history, and patient age considerations across all cancer stages) to support more personalized and appropriate treatment recommendations.

Another key challenge was the inconsistency in cancer stage predictions produced by the language model. Future pipelines should apply stricter document selection criteria to ensure alignment with a standardized staging reference, such as the AJCC 8th edition (Amin et al., 2017). Although prompt engineering was implemented to reduce staging mismatches, further refinement could help guide model outputs more effectively.

While general-purpose language models such as GPT-4o and Gemini performed well in generating responses, future work should explore domain-specific models like Med-PaLM (Rahaman et al., 2023; Google Research, n.d), BioGPT (Luo et al., 2023), or BioMedLM (Bolton et al., 2024). These models are trained on biomedical literature and clinical data, which may improve accuracy and context alignment in medical applications.

Additionally, future developments could incorporate a chatbot interface to enhance patient support. This tool could help communicate diagnostic findings and treatment options more clearly, assist patients in understanding their condition, and provide accessible explanations based on individual cancer stages and clinical profiles.

Finally, explainability should be addressed by implementing mechanisms for source attribution, allowing each treatment recommendation to be linked to a supporting document or guideline. Evaluation strategies should move beyond token-level metrics and instead assess clinically relevant aspects, such as staging-treatment alignment, adherence to official protocols, and inclusion of all necessary treatment components (e.g., surgery, systemic therapy, and follow-up), to ensure medical safety and applicability.

In conclusion, this study developed and evaluated an integrated framework for lung cancer classification, TNM staging, and treatment recommendation. Across tasks, dataset size and class imbalance were the main limiting factors, directly impacting model performance and generalizability. The findings suggest that using more representative training data, incorporating inter-slice relationships in CT analysis, and adopting clinically validated evaluation methods could help improve results in future work.

Alternative model architectures, late-fusion approaches, and domain-specific models may offer further improvements, although their effectiveness remains to be tested. Future development should also involve medical professionals throughout the pipeline, particularly in the evaluation and improvement of treatment recommendations. Clinical review is necessary to ensure the safety, appropriateness, and practical relevance of the generated outputs. In parallel, it is crucial to continue improving model transparency and exploring strategies that enhance explainability. With further testing, evaluation, and integration of clinical feedback, the pipeline may contribute to more reliable support tools in cancer care.

BIBLIOGRAPHICAL REFERENCES

Abas Mohamed, Y., Ee Khoo, B., Shahrimie Mohd Asaari, M., Ezane Aziz, M., & Rahiman Ghazali, F. (2024). Decoding the black box: Explainable AI (Xai) for cancer diagnosis, prognosis, and treatment planning-A state-of-the art systematic review. International Journal of Medical Informatics, 193, 105689. https://doi.org/10.1016/j.ijmedinf.2024.105689

Aerts, H. J. W. L., Wee, L., Rios Velazquez, E., Leijenaar, R. T. H., Parmar, C., Grossmann, P., Carvalho, S., Bussink, J., Monshouwer, R., Haibe-Kains, B., Rietveld, D., Hoebers, F., Rietbergen, M. M., Leemans, C. R., Dekker, A., Quackenbush, J., Gillies, R. J., Lambin, P. (2014). Data From NSCLC-Radiomics (version 4) [Data set]. The Cancer Imaging Archive. https://doi.org/10.7937/K9/TCIA.2015.PF0M9REI

Ahmed, S. B., Solis-Oba, R., & Ilie, L. (2022). Explainable-ai in automated medical report generation using chest x-ray images. Applied Sciences, 12(22), 11750. https://doi.org/10.3390/app122211750

Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2623–2631. https://doi.org/10.1145/3292500.3330701

American Cancer Society. (n.d.). Lung cancer. American Cancer Society. Retrieved March 2025, from https://www.cancer.org/cancer/types/lung-cancer.html

Amin, M. B., Greene, F. L., Edge, S. B., Compton, C. C., Gershenwald, J. E., Brookland, R. K., Meyer, L., Gress, D. M., Byrd, D. R., & Winchester, D. P. (2017). The Eighth Edition AJCC Cancer Staging Manual: Continuing to build a bridge from a population-based to a more «personalized» approach to cancer staging. CA: A Cancer Journal for Clinicians, 67(2), 93–99. https://doi.org/10.3322/caac.21388

Atmakuru, A., Chakraborty, S., Faust, O., Salvi, M., Datta Barua, P., Molinari, F., Acharya, U. R., & Homaira, N. (2024). Deep learning in radiology for lung cancer diagnostics: A systematic review of classification, segmentation, and predictive modeling techniques. Expert Systems with Applications, 255, 124665. https://doi.org/10.1016/j.eswa.2024.124665

Azurmendi, I., Gonzalez, M., García, G., Zulueta, E., & Martín, E. (2024). Deep learning-based postural asymmetry detection through pressure mat. Applied Sciences, 14(24), 12050. https://doi.org/10.3390/app142412050

Baig, M. M., Hobson, C., GholamHosseini, H., Ullah, E., & Afifi, S. (2024). Generative ai in improving personalized patient care plans: Opportunities and barriers towards its wider adoption. Applied Sciences, 14(23), 10899. https://doi.org/10.3390/app142310899

Bannur, S., Hyland, S., Liu, Q., Pérez-García, F., Ilse, M., Castro, D. C., Boecking, B., Sharma, H., Bouzid, K., Thieme, A., Schwaighofer, A., Wetscherek, M., Lungren, M. P., Nori, A., Alvarez-Valle, J., & Oktay, O. (2023). Learning to exploit temporal structure for biomedical vision-language processing (No. arXiv:2301.04558). arXiv. https://doi.org/10.48550/arXiv.2301.04558

Board of Innovation. (n.d.). Glass AI: Explore the power of generative AI in healthcare. Retrieved January 11, 2025, from https://healthcare.boardofinnovation.com/glass-ai/

Bolton, E., Venigalla, A., Yasunaga, M., Hall, D., Xiong, B., Lee, T., Daneshjou, R., Frankle, J., Liang, P., Carbin, M., & Manning, C. D. (2024). Biomedlm: A 2. 7b parameter language model trained on biomedical text (No. arXiv:2403.18421). arXiv. https://doi.org/10.48550/arXiv.2403.18421

Bray, F., Laversanne, M., Sung, H., Ferlay, J., Siegel, R. L., Soerjomataram, I., & Jemal, A. (2024). Global cancer statistics 2022: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. CA: A Cancer Journal for Clinicians, 74(3), 229–263. https://doi.org/10.3322/caac.21834

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., … Amodei, D. (2020). Language models are few-shot learners (No. arXiv:2005.14165). arXiv. https://doi.org/10.48550/arXiv.2005.14165

Chang, Y., Wang, X., Wang, J., Wu, Y., Yang, L., Zhu, K., Chen, H., Yi, X., Wang, C., Wang, Y., Ye, W., Zhang, Y., Chang, Y., Yu, P. S., Yang, Q., & Xie, X. (2023). A survey on evaluation of large language models (No. arXiv:2307.03109). arXiv. https://doi.org/10.48550/arXiv.2307.03109

Chaudhary, S. (2020, November 19). *The Annotated ResNet-50*. Towards Data Science. https://towardsdatascience.com/the-annotated-resnet-50-a6c536034758

Chen, C., Zhao, L.-L., Lang, Q., & Xu, Y. (2024). A novel detection and classification framework for diagnosing of cerebral microbleeds using transformer and language. Bioengineering, 11(10), 993. https://doi.org/10.3390/bioengineering11100993

Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., Lu, L., Yuille, A. L., & Zhou, Y. (2021). Transunet: Transformers make strong encoders for medical image segmentation (No. arXiv:2102.04306). arXiv. https://doi.org/10.48550/arXiv.2102.04306

Chroma. (n.d.). Introduction—Chroma docs. Retrieved March 26, 2025, from https://docs.trychroma.com/docs/overview/introduction

Clark, K., Vendt, B., Smith, K., Freymann, J., Kirby, J., Koppel, P., Moore, S., Phillips, S., Maffitt, D., Pringle, M., Tarbox, L., & Prior, F. (2013). The cancer imaging archive (Tcia): Maintaining and operating a public information repository. Journal of Digital Imaging, 26(6), 1045–1057. https://doi.org/10.1007/s10278-013-9622-7

Corti. (n.d.). Engage solutions: AI for enhanced patient engagement. Retrieved January 11, 2025, from https://www.corti.ai/solutions/engage

Coudray, N., Ocampo, P. S., Sakellaropoulos, T., Narula, N., Snuderl, M., Fenyö, D., Moreira, A. L., Razavian, N., & Tsirigos, A. (2018). Classification and mutation prediction from non–small cell lung cancer histopathology images using deep learning. Nature Medicine, 24(10), 1559–1567. https://doi.org/10.1038/s41591-018-0177-5

Davri, A., Birbas, E., Kanavos, T., Ntritsos, G., Giannakeas, N., Tzallas, A. T., & Batistatou, A. (2023). Deep learning for lung cancer diagnosis, prognosis and prediction using histological and cytological images: A systematic review. Cancers, 15(15), 3981. https://doi.org/10.3390/cancers15153981

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding (No. arXiv:1810.04805). arXiv. https://doi.org/10.48550/arXiv.1810.04805

El-Baz, A., Beache, G. M., Gimel'farb, G., Suzuki, K., Okada, K., Elnakib, A., Soliman, A., & Abdollahi, B. (2013). Computer-aided diagnosis systems for lung cancer: Challenges and methodologies. International Journal of Biomedical Imaging, 2013, 1–46. https://doi.org/10.1155/2013/942353

Elazab, N., Gab-Allah, W. A., & Elmogy, M. (2024). A multi-class brain tumor grading system based on histopathological images using a hybrid YOLO and RESNET networks. Scientific Reports, 14(1), 4584. https://doi.org/10.1038/s41598-024-54864-6

Elshahawy, M., Elnemr, A., Oproescu, M., Schiopu, A.-G., Elgarayhi, A., Elmogy, M. M., & Sallah, M. (2023). Early melanoma detection based on a hybrid yolov5 and resnet technique. Diagnostics, 13(17), 2804. https://doi.org/10.3390/diagnostics13172804

Es, S., James, J., Espinosa-Anke, L., & Schockaert, S. (2023). Ragas: Automated evaluation of retrieval augmented generation (No. arXiv:2309.15217). arXiv. https://doi.org/10.48550/arXiv.2309.15217

European Society for Medical Oncology. (n.d.). ESMO – European Society for Medical Oncology. Retrieved March 2025, from https://www.esmo.org/

Fan, W., Ding, Y., Ning, L., Wang, S., Li, H., Yin, D., Chua, T.-S., & Li, Q. (2024). A survey on rag meeting llms: Towards retrieval-augmented large language models (No. arXiv:2405.06211). arXiv. https://doi.org/10.48550/arXiv.2405.06211

Geantă, M., Bădescu, D., Chirca, N., Nechita, O. C., Radu, C. G., Rascu, S., Rădăvoi, D., Sima, C., Toma, C., & Jinga, V. (2024). The Potential Impact of Large Language Models on Doctor-Patient Communication: A Case Study in Prostate Cancer. Healthcare (Basel, Switzerland), 12(15), 1548. https://doi.org/10.3390/healthcare12151548

Google DeepMind. (2024). Gemini 2.0 Flash: Advancing real-time multimodal AI. https://cloud.google.com/vertex-ai/generative-ai/docs/gemini-v2

Google Research. (n.d.). Med-PaLM: Medical AI research and applications. Retrieved January 11, 2025, from https://sites.research.google/med-palm/

Gu, Y., Tinn, R., Cheng, H., Lucas, M., Usuyama, N., Liu, X., Naumann, T., Gao, J., & Poon, H. (2021). Domain-specific language model pretraining for biomedical natural language processing (No. arXiv:2007.15779). arXiv. https://doi.org/10.48550/arXiv.2007.15779

Gulum, M. A., Trombley, C. M., & Kantardzic, M. (2021). A review of explainable deep learning cancer detection models in medical imaging. Applied Sciences, 11(10), 4573. https://doi.org/10.3390/app11104573

Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S., & Lew, M. S. (2016). Deep learning for visual understanding: A review. Neurocomputing, 187, 27–48. https://doi.org/10.1016/j.neucom.2015.09.116

Hagos, D. H., Battle, R., & Rawat, D. B. (2024). Recent advances in generative ai and large language models: Current status, challenges, and perspectives (No. arXiv:2407.14962). arXiv. https://doi.org/10.48550/arXiv.2407.14962

Hatamizadeh, A., Tang, Y., Nath, V., Yang, D., Myronenko, A., Landman, B., Roth, H., & Xu, D. (2021). Unetr: Transformers for 3d medical image segmentation (No. arXiv:2103.10504). arXiv. https://doi.org/10.48550/arXiv.2103.10504

Hayat, N., Geras, K. J., & Shamout, F. E. (2023). MedFuse: Multi-modal fusion with clinical time-series data and chest X-ray images (No. arXiv:2207.07027). arXiv. https://doi.org/10.48550/arXiv.2207.07027

He, K., Zhang, X., Ren, S., & Sun, J. (2015). *Deep residual learning for image recognition* (No. arXiv:1512.03385). arXiv. https://doi.org/10.48550/arXiv.1512.03385

Hemasri, Chettim & Vijayalakshmi, M. & Jyotheesh, Vootukuru. (2024). Redefining Medicine: The Power of Generative AI in Modern Healthcare. 1293-1298. 10.1109/ICOSEC61587.2024.10722592.

Hicks, S. A., Strümke, I., Thambawita, V., Hammou, M., Riegler, M. A., Halvorsen, P., & Parasa, S. (2022). On evaluation metrics for medical applications of artificial intelligence. Scientific Reports, 12(1), 5979. https://doi.org/10.1038/s41598-022-09954-8

Huang, P., Lin, C. T., Li, Y., Tammemagi, M. C., Brock, M. V., Atkar-Khattra, S., Xu, Y., Hu, P., Mayo, J. R., Schmidt, H., Gingras, M., Pasian, S., Stewart, L., Tsai, S., Seely, J. M., Manos, D., Burrowes, P., Bhatia, R., Tsao, M.-S., & Lam, S. (2019). Prediction of lung cancer risk at follow-up screening with low-dose CT: A training and validation study of a deep learning method. The Lancet Digital Health, 1(7), e353–e362. https://doi.org/10.1016/S2589-7500(19)30159-1

Hugging Face. (2024, January 5). sentence-transformers/all-MiniLM-L6-v2. https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

Iannantuono, G. M., Bracken-Clarke, D., Floudas, C. S., Roselli, M., Gulley, J. L., & Karzai, F. (2023). Applications of large language models in cancer care: Current evidence and future perspectives. Frontiers in Oncology, 13, 1268915. https://doi.org/10.3389/fonc.2023.1268915

Inamura, K. (2017). Lung cancer: Understanding its molecular pathology and the 2015 who classification. Frontiers in Oncology, 7. https://doi.org/10.3389/fonc.2017.00193

International Association for the Study of Lung Cancer (IASLC). (2024). Staging Manual in Thoracic Oncology (3rd ed.). Springer.

Javed, R., Abbas, T., Khan, A. H., Daud, A., Bukhari, A., & Alharbey, R. (2024). Deep learning for lungs cancer detection: A review. Artificial Intelligence Review, 57(8), 197. https://doi.org/10.1007/s10462-024-10807-1

Jia, F., Liu, X., Deng, L., Gu, J., Pu, C., Bai, T., Huang, M., Lu, Y., & Liu, K. (2024). Oncogpt: A medical conversational model tailored with oncology domain expertise on a large language model meta-ai(Llama) (No. arXiv:2402.16810). arXiv. https://doi.org/10.48550/arXiv.2402.16810

Kahun. (n.d.). Technology: AI-driven clinical reasoning. Retrieved January 11, 2025, from https://www.kahun.com/technology

Ke, A., Ellsworth, W., Banerjee, O., Ng, A. Y., & Rajpurkar, P. (2021). CheXtransfer: Performance and parameter efficiency of ImageNet models for chest X-Ray interpretation. Proceedings of the Conference on Health, Inference, and Learning, 116–124. https://doi.org/10.1145/3450439.3451867

Lababede, O., & Meziane, M. A. (2018). The eighth edition of tnm staging of lung cancer: Reference chart and diagrams. *The Oncologist*, *23*(7), 844–848. https://doi.org/10.1634/theoncologist.2017-0659

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., & Kiela, D. (2021). Retrieval-augmented generation for knowledge-intensive nlp tasks (No. arXiv:2005.11401). arXiv. https://doi.org/10.48550/arXiv.2005.11401

Li, P., Wang, S., Li, T., Lu, J., HuangFu, Y., & Wang, D. (2020). A Large-Scale CT and PET/CT Dataset for Lung Cancer Diagnosis (Version 5) [Data set]. The Cancer Imaging Archive. https://doi.org/10.7937/TCIA.2020.NNC2-0461

Li, P., Wang, S., Li, T., Lu, J., HuangFu, Y., & Wang, D. (2020). A large-scale ct and pet/ct dataset for lung cancer diagnosis [Dataset]. The Cancer Imaging Archive. https://doi.org/10.7937/TCIA.2020.NNC2-0461

Li, Z., Liu, F., Yang, W., Peng, S., & Zhou, J. (2022). A survey of convolutional neural networks: Analysis, applications, and prospects. IEEE Transactions on Neural Networks and Learning Systems, 33(12), 6999–7019. https://doi.org/10.1109/TNNLS.2021.3084827

Liam, C.-K., Liam, Y.-S., Poh, M.-E., & Wong, C.-K. (2020). Accuracy of lung cancer staging in the multidisciplinary team setting. Translational Lung Cancer Research, 9(4), 1654–1666. https://doi.org/10.21037/tlcr.2019.11.28

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. Text Summarization Branches Out, 74–81. https://aclanthology.org/W04-1013/

Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., & Xie, S. (2022). A ConvNet for the 2020s (No. arXiv:2201.03545). arXiv. https://doi.org/10.48550/arXiv.2201.03545

Liu, Z., Wang, P., Li, Y., Holmes, J., Shu, P., Zhang, L., Liu, C., Liu, N., Zhu, D., Li, X., Li, Q., Patel, S. H., Sio, T. T., Liu, T., & Liu, W. (2023). Radonc-gpt: A large language model for radiation oncology (No. arXiv:2309.10160). arXiv. https://doi.org/10.48550/arXiv.2309.10160

Long, C., Liu, Y., Ouyang, C., & Yu, Y. (2024). Bailicai: A domain-optimized retrieval-augmented generation framework for medical applications (No. arXiv:2407.21055). arXiv. https://doi.org/10.48550/arXiv.2407.21055

Luo, R., Sun, L., Xia, Y., Qin, T., Zhang, S., Poon, H., & Liu, T.-Y. (2023). Biogpt: Generative pre-trained transformer for biomedical text generation and mining (No. arXiv:2210.10341). arXiv. https://doi.org/10.48550/arXiv.2210.10341

Malathy, V & Maiti, Niladri & Kumar, Nithin & Lavanya, D. & Aswath, S. & Banu, Shaik. (2024). Deep Learning -Enhanced Image Segmentation for Medical Diagnostics. 1-6. 10.1109/ACCAI61061.2024.10602242.

McDuff, D., Schaekermann, M., Tu, T., Palepu, A., Wang, A., Garrison, J., Singhal, K., Sharma, Y., Azizi, S., Kulkarni, K., Hou, L., Cheng, Y., Liu, Y., Mahdavi, S. S., Prakash, S., Pathak, A., Semturs, C., Patel, S., Webster, D. R., … Natarajan, V. (2023). Towards accurate

differential diagnosis with large language models (No. arXiv:2312.00164). arXiv. https://doi.org/10.48550/arXiv.2312.00164

Mehta, S., & Rastegari, M. (2022). Mobilevit: Light-weight, general-purpose, and mobile-friendly vision transformer (No. arXiv:2110.02178). arXiv. https://doi.org/10.48550/arXiv.2110.02178

Miao, J., Thongprayoon, C., Suppadungsuk, S., Garcia Valencia, O. A., & Cheungpasitporn, W. (2024). Integrating retrieval-augmented generation with large language models in nephrology: Advancing practical applications. Medicina (Kaunas, Lithuania), 60(3), 445. https://doi.org/10.3390/medicina60030445

Milletari, F., Navab, N., & Ahmadi, S.-A. (2016). V-net: Fully convolutional neural networks for volumetric medical image segmentation (No. arXiv:1606.04797). arXiv. https://doi.org/10.48550/arXiv.1606.04797

Most, A., Hu, M., Yang, H., Liu, T., Chen, X., Li, S., Xu, S., Liu, Z., & Sikora, A. (2024). Evaluating accuracy and reproducibility of large language model performance in pharmacy education. medRxiv. https://doi.org/10.1101/2024.03.21.24304667

Musa, A. B., Fasina, E. P., Ojiako, C. P., Sawyerr, B. A., & Murainah, A.-A. (2024). Improving medical diagnosis with LLM reprompting. Machine Intelligence Research Group (MIRG), Department of Computer Sciences, University of Lagos, Lagos, Nigeria. https://api-ir.unilag.edu.ng/server/api/core/bitstreams/8f44db44-caa7-4a32-bc38-84b288b4a54a/content

National Cancer Institute. (n.d.). National Cancer Institute. Retrieved March 2025, from https://www.cancer.gov/

National Comprehensive Cancer Network. (n.d.). NCCN – National Comprehensive Cancer Network. Retrieved March 2025, from https://www.nccn.org/

Navani, N., Fisher, D. J., Tierney, J. F., Stephens, R. J., Burdett, S., Burdett, S., Rydzewska, L. H. M., Tierney, J. F., Auperin, A., Le Chevalier, T., Le Pechoux, C., Pignon, J.-P., Arriagada, R., Johnson, D. H., Van Meerbeeck, J., Parmar, M. K. B., Stephens, R. J., Stewart, L. A., Bunn, P. A., … Yang, X.-N. (2019). The accuracy of clinical staging of stage i-iiia non-small cell lung cancer. Chest, 155(3), 502–509. https://doi.org/10.1016/j.chest.2018.10.020

Nooreldeen, R., & Bach, H. (2021). Current and future development in lung cancer diagnosis. International Journal of Molecular Sciences, 22(16), 8661. https://doi.org/10.3390/ijms22168661

Nuance. (n.d.). Dragon Medical One: AI-powered clinical documentation. Retrieved January 11, 2025, from https://www.nuance.com/healthcare/dragon-ai-clinical-solutions/dragon-medical-one.html

OpenAI, Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., Avila, R., Babuschkin, I., Balaji, S., Balcom, V., Baltescu, P., Bao, H., Bavarian, M., Belgum, J., … Zoph, B. (2024). Gpt-4 technical report (No. arXiv:2303.08774). arXiv. https://doi.org/10.48550/arXiv.2303.08774

OpenAI. (2024). GPT-4o Mini: Advancing cost-efficient intelligence. https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/

OpenAI. (n.d.). Embedding models. Retrieved March 26, 2025, from https://platform.openai.com/docs/guides/embeddings#embedding-models

Padilla, Rafael & Netto, Sergio & da Silva, Eduardo. (2020). A Survey on Performance Metrics for Object-Detection Algorithms. 10.1109/IWSSIP48289.2020.

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2001). BLEU: A method for automatic evaluation of machine translation. Proceedings of the 40th Annual Meeting on Association for Computational Linguistics - ACL ’02, 311. https://doi.org/10.3115/1073083.1073135

Qu, X., Li, Y., Su, Z., Sun, W., Yan, J., Liu, D., Cui, G., Liu, D., Liang, S., He, J., Li, P., Wei, W., Shao, J., Lu, C., Zhang, Y., Hua, X.-S., Zhou, B., & Cheng, Y. (2025). A survey of efficient reasoning for large reasoning models: Language, multimodality, and beyond (No. arXiv:2503.21614). arXiv. https://doi.org/10.48550/arXiv.2503.21614

RAGAS. (n.d.). RAGAS documentation. Retrieved March 26, 2025, from https://docs.ragas.io/en/stable/

Rahaman, M. S., Ahsan, M. M. T., Anjum, N., Rahman, M. M., & Rahman, M. N. (2023). The ai race is on! Google’s bard and openai’s chatgpt head to head: An opinion article (SSRN Scholarly Paper No. 4351785). https://doi.org/10.2139/ssrn.4351785

Rainio, O., Teuho, J., & Klén, R. (2024). Evaluation metrics and statistical tests for machine learning. Scientific Reports, 14(1), 6086. https://doi.org/10.1038/s41598-024-56706-x

Ranjit, M., Ganapathy, G., Manuel, R., & Ganu, T. (2023). Retrieval augmented chest x-ray report generation using openai gpt models (No. arXiv:2305.03660). arXiv. https://doi.org/10.48550/arXiv.2305.03660

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 779–788. https://doi.org/10.1109/CVPR.2016.91

Regard. (n.d.). Regard: AI-driven clinical insights platform. Retrieved January 11, 2025, from https://regard.com

Richardson, L. (2023). beautifulsoup4 (Version 4.x) [Python package]. Python Package Index. https://pypi.org/project/beautifulsoup4/

Ríos-Hoyo, A., Shan, N. L., Li, A., Pearson, A. T., Pusztai, L., & Howard, F. M. (2024). Evaluation of large language models as a diagnostic aid for complex medical cases. Frontiers in Medicine, 11. https://doi.org/10.3389/fmed.2024.1380148

Robertson, S., & Zaragoza, H. (2009). The probabilistic relevance framework: Bm25 and beyond. Foundations and Trends® in Information Retrieval, 3(4), 333–389. https://doi.org/10.1561/1500000019

Rozière, B., Gehring, J., Gloeckle, F., Sootla, S., Gat, I., Tan, X. E., Adi, Y., Liu, J., Sauvestre, R., Remez, T., Rapin, J., Kozhevnikov, A., Evtimov, I., Bitton, J., Bhatt, M., Ferrer, C. C., Grattafiori, A., Xiong, W., Défossez, A., … Synnaeve, G. (2024). Code llama: Open foundation models for code (No. arXiv:2308.12950). arXiv. https://doi.org/10.48550/arXiv.2308.12950

Sakai, H., Lam, S. S., Mikaeili, M., Bosire, J., & Jovin, F. (2024). Large language models for patient comments multi-label classification (No. arXiv:2410.23528). arXiv. https://doi.org/10.48550/arXiv.2410.23528

Selenium Contributors. (2024). selenium (Version 4.x) [Python package]. PyPI. https://pypi.org/project/selenium/

Sharma, K., Kumar, P., & Li, Y. (2024). Og-rag: Ontology-grounded retrieval-augmented generation for large language models (No. arXiv:2412.15235). arXiv. https://doi.org/10.48550/arXiv.2412.15235

Shen, Y., Xu, Y., Ma, J., Rui, W., Zhao, C., Heacock, L., & Huang, C. (2024). Multi-modal large language models in radiology: Principles, applications, and potential. Abdominal Radiology (New York). https://doi.org/10.1007/s00261-024-04708-8

Shrestha, A., & Mahmood, A. (2019). Review of deep learning algorithms and architectures. IEEE Access, 7, 53040–53065. https://doi.org/10.1109/ACCESS.2019.2912200

Siegel, R. L., Giaquinto, A. N., & Jemal, A. (2024). Cancer statistics, 2024. CA: A Cancer Journal for Clinicians, 74(1), 12–49. https://doi.org/10.3322/caac.21820

Silva, F., Pereira, T., Neves, I., Morgado, J., Freitas, C., Malafaia, M., Sousa, J., Fonseca, J., Negrão, E., Flor De Lima, B., Correia Da Silva, M., Madureira, A. J., Ramos, I., Costa, J. L., Hespanhol, V., Cunha, A., & Oliveira, H. P. (2022). Towards machine learning-aided lung cancer clinical routines: Approaches and open challenges. Journal of Personalized Medicine, 12(3), 480. https://doi.org/10.3390/jpm12030480

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition (No. arXiv:1409.1556). arXiv. https://doi.org/10.48550/arXiv.1409.1556

Suki.ai. (n.d.). Technology. Retrieved January 11, 2025, from https://www.suki.ai/technology/

Tan, M., & Le, Q. V. (2020). Efficientnet: Rethinking model scaling for convolutional neural networks (No. arXiv:1905.11946). arXiv. https://doi.org/10.48550/arXiv.1905.11946

Tan, M., & Le, Q. V. (2021). Efficientnetv2: Smaller models and faster training (No. arXiv:2104.00298). arXiv. https://doi.org/10.48550/arXiv.2104.00298

Ultralytics. (2023). *YOLOv8 anchor-free bounding box prediction – Issue #189* [Illustration]. GitHub. https://github.com/ultralytics/ultralytics/issues/189

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2023). Attention is all you need (No. arXiv:1706.03762). arXiv. https://doi.org/10.48550/arXiv.1706.03762

Wang, C., Li, S., Lin, N., Zhang, X., Han, Y., Wang, X., Liu, D., Tan, X., Pu, D., Li, K., Qian, G., & Yin, R. (2025). Application of large language models in medical training evaluation-using chatgpt as a standardized patient: Multimetric assessment. Journal of Medical Internet Research, 27, e59435. https://doi.org/10.2196/59435

Wang, D., & Zhang, S. (2024). Large language models in medical and healthcare fields: Applications, advances, and challenges. Artificial Intelligence Review, 57(11), 299. https://doi.org/10.1007/s10462-024-10921-0

Wang, S., Yang, D. M., Rong, R., Zhan, X., Fujimoto, J., Liu, H., Minna, J., Wistuba, I. I., Xie, Y., & Xiao, G. (2019). Artificial intelligence in lung cancer pathology image analysis. Cancers, 11(11), 1673. https://doi.org/10.3390/cancers11111673

Wang, S., Zhao, Z., Ouyang, X., Wang, Q., & Shen, D. (2023). Chatcad: Interactive computer-aided diagnosis on medical image using large language models (No. arXiv:2302.07257). arXiv. https://doi.org/10.48550/arXiv.2302.07257

Wehbe, A., Dellepiane, S., & Minetti, I. (2024). Enhanced lung cancer detection and tnm staging using yolov8 and tnmclassifier: An integrated deep learning approach for ct imaging. IEEE Access, 12, 141414–141424. https://doi.org/10.1109/ACCESS.2024.3462629

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., Platen, P. von, Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., … Rush, A. M. (2020). Huggingface's transformers: State-of-the-art natural language processing (No. arXiv:1910.03771). arXiv. https://doi.org/10.48550/arXiv.1910.03771

Xu, R., Hong, Y., Zhang, F., & Xu, H. (2024). Evaluation of the integration of retrieval-augmented generation in large language model for breast cancer nursing care

responses. Scientific Reports, 14(1), 30794. https://doi.org/10.1038/s41598-024-81052-3

Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: An overview and application in radiology. Insights into Imaging, 9(4), 611–629. https://doi.org/10.1007/s13244-018-0639-9

Zhang, P., & Kamel Boulos, M. N. (2023). Generative ai in medicine and healthcare: Promises, opportunities and challenges. Future Internet, 15(9), 286. https://doi.org/10.3390/fi15090286

Zhang, P., Shi, J., & Kamel Boulos, M. N. (2024). Generative AI in Medicine and Healthcare: Moving Beyond the 'Peak of Inflated Expectations'. Future Internet, 16(12), 462. https://doi.org/10.3390/fi16120462

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2020). Bertscore: Evaluating text generation with bert (No. arXiv:1904.09675). arXiv. https://doi.org/10.48550/arXiv.1904.09675

Zhang, X., Talukdar, N., Vemulapalli, S., Ahn, S., Wang, J., Meng, H., Murtaza, S. M. B., Leshchiner, D., Dave, A. A., Joseph, D. F., Witteveen-Lane, M., Chesla, D., Zhou, J., & Chen, B. (2024). Comparison of prompt engineering and fine-tuning strategies in large language models in the classification of clinical notes. medRxiv, 2024.02.07.24302444. https://doi.org/10.1101/2024.02.07.24302444

Zhang, Y., Sun, P., Jiang, Y., Yu, D., Weng, F., Yuan, Z., Luo, P., Liu, W., & Wang, X. (2022). Bytetrack: Multi-object tracking by associating every detection box (No. arXiv:2110.06864). arXiv. https://doi.org/10.48550/arXiv.2110.06864

Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., Du, Y., Yang, C., Chen, Y., Chen, Z., Jiang, J., Ren, R., Li, Y., Tang, X., Liu, Z., … Wen, J.-R. (2024). A survey of large language models (No. arXiv:2303.18223). arXiv. https://doi.org/10.48550/arXiv.2303.18223

Zhou, S., Xu, Z., Zhang, M., Xu, C., Guo, Y., Zhan, Z., Ding, S., Wang, J., Xu, K., Fang, Y., Xia, L., Yeung, J., Zha, D., Melton, G. B., Lin, M., & Zhang, R. (2024). Large language models for disease diagnosis: A scoping review (No. arXiv:2409.00097). arXiv. https://doi.org/10.48550/arXiv.2409.00097

Zhou, T., Ruan, S., & Canu, S. (2019). A review: Deep learning for medical image segmentation using multi-modality fusion. Array, 3–4, 100004. https://doi.org/10.1016/j.array.2019.100004

APPENDIX A: EVALUATION METRICS ON THE TRAIN SET FOR THE YOLOV8N MODEL (BATCH SIZE: 16, COMBINED DATASET)

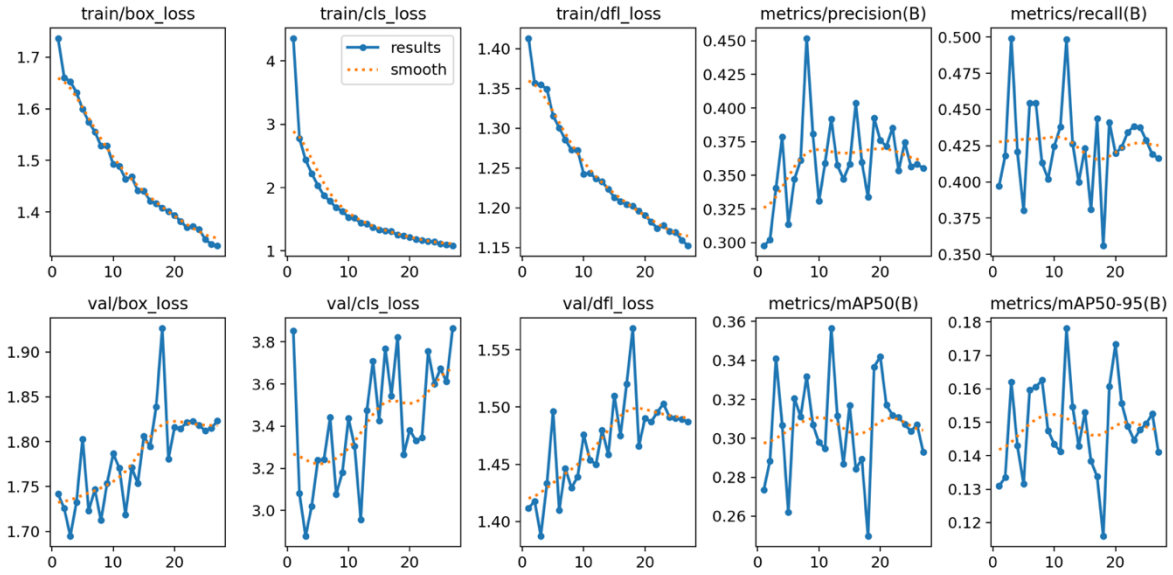


Figure A.1 -Training and Validation Loss & Metric Curves

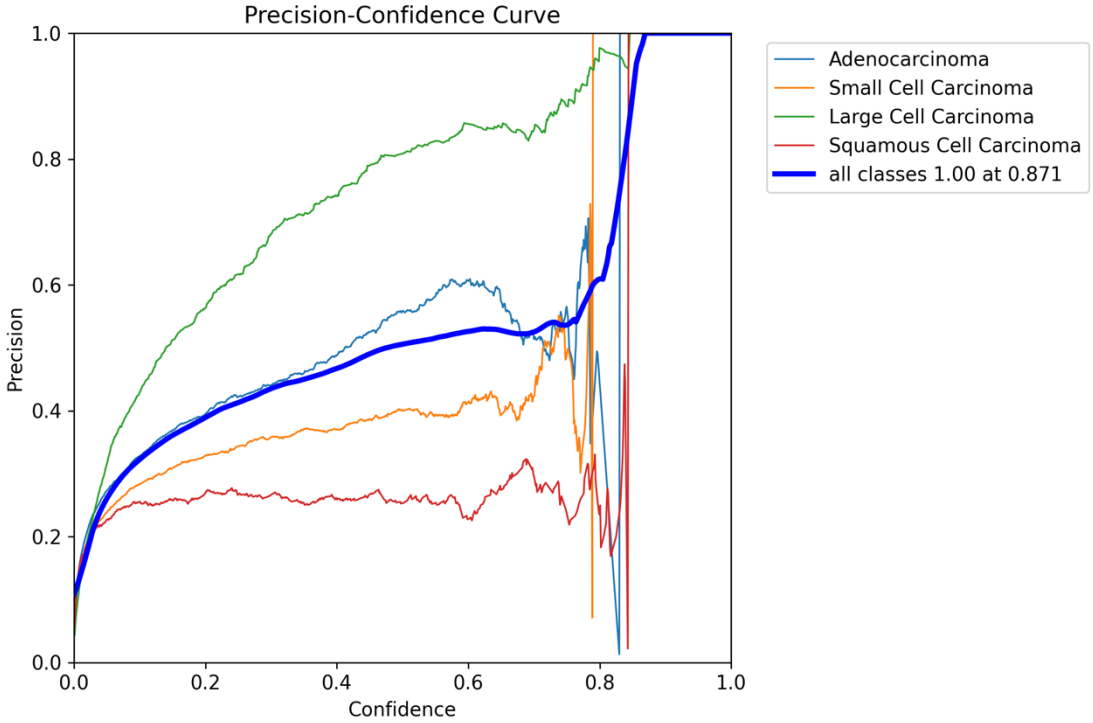


Figure A.2 -Precision-Confidence Curve (Training Set)

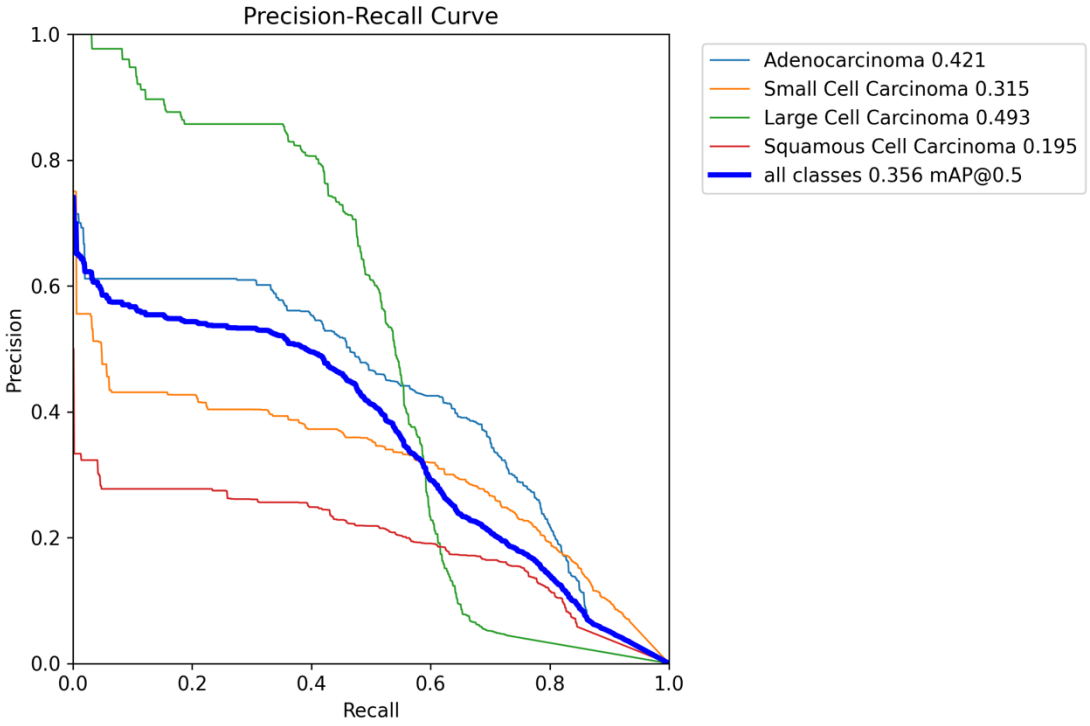


Figure A.3 - Precision-Recall Curve (Training Set)

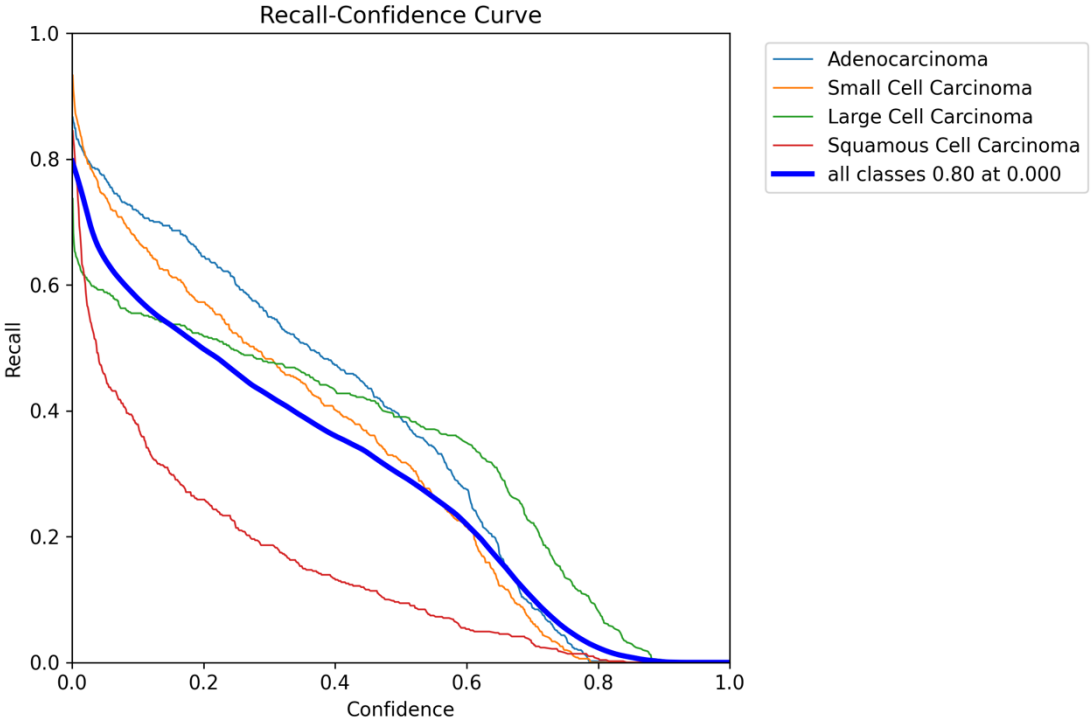


Figure A.4 - Recall-Confidence Curve (Training Set)

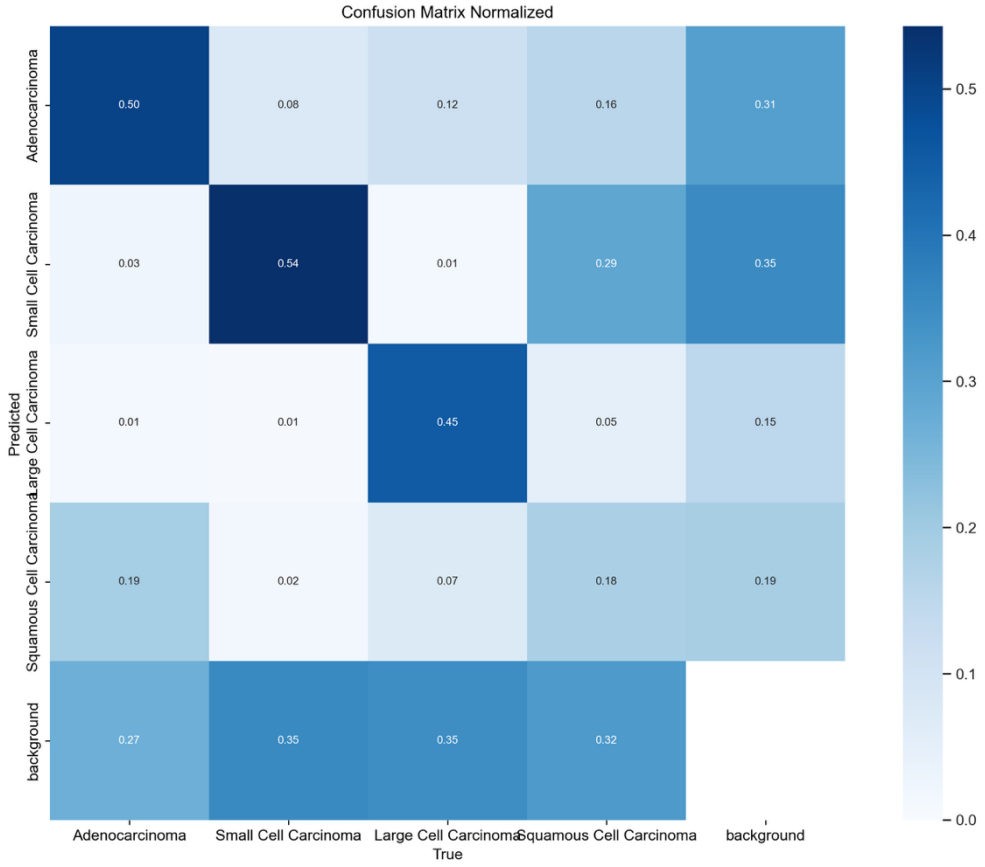


Figure A.5 - Normalized Confusion Matrix (Training Set)

APPENDIX B: EVALUATION METRICS ON THE TEST SET FOR THE YOLOV8N MODEL (BATCH SIZE: 16, COMBINED DATASET)

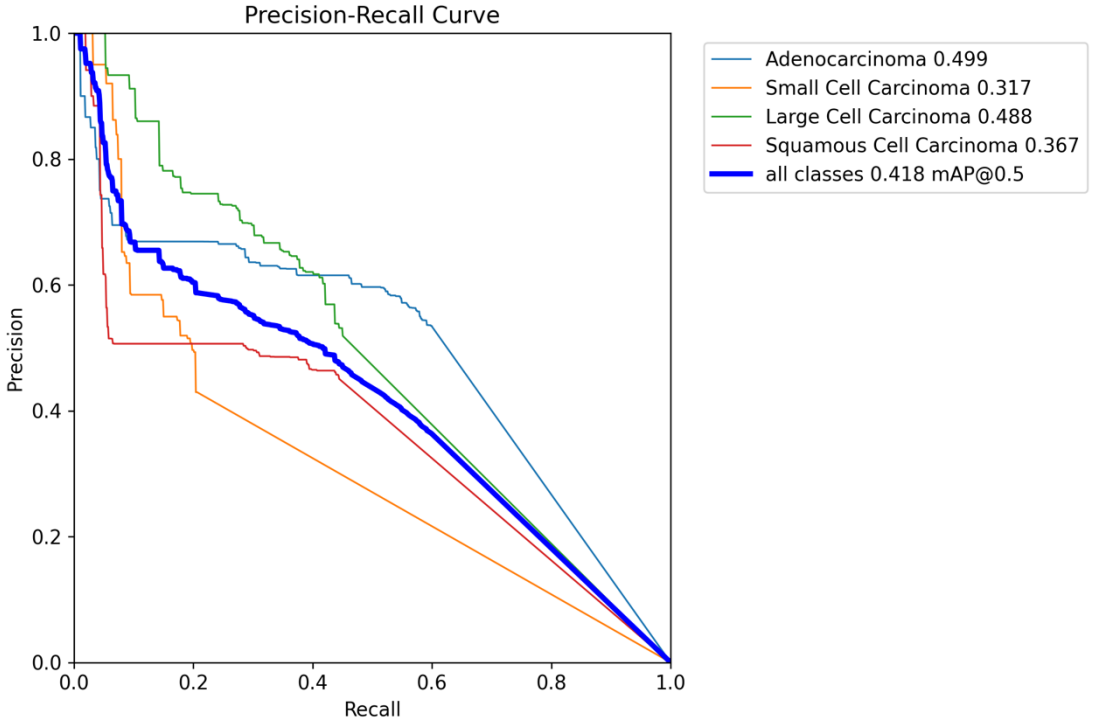


Figure B.1 - Precision-Recall Curve

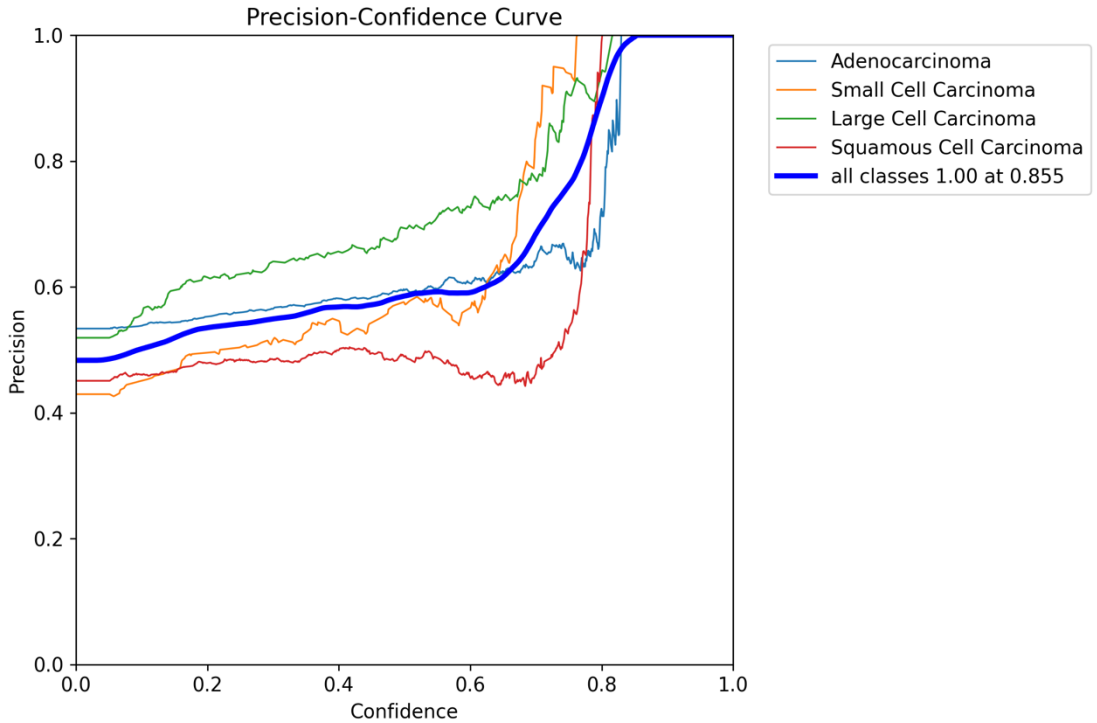


Figure B.2 - Precision-Confidence Curve

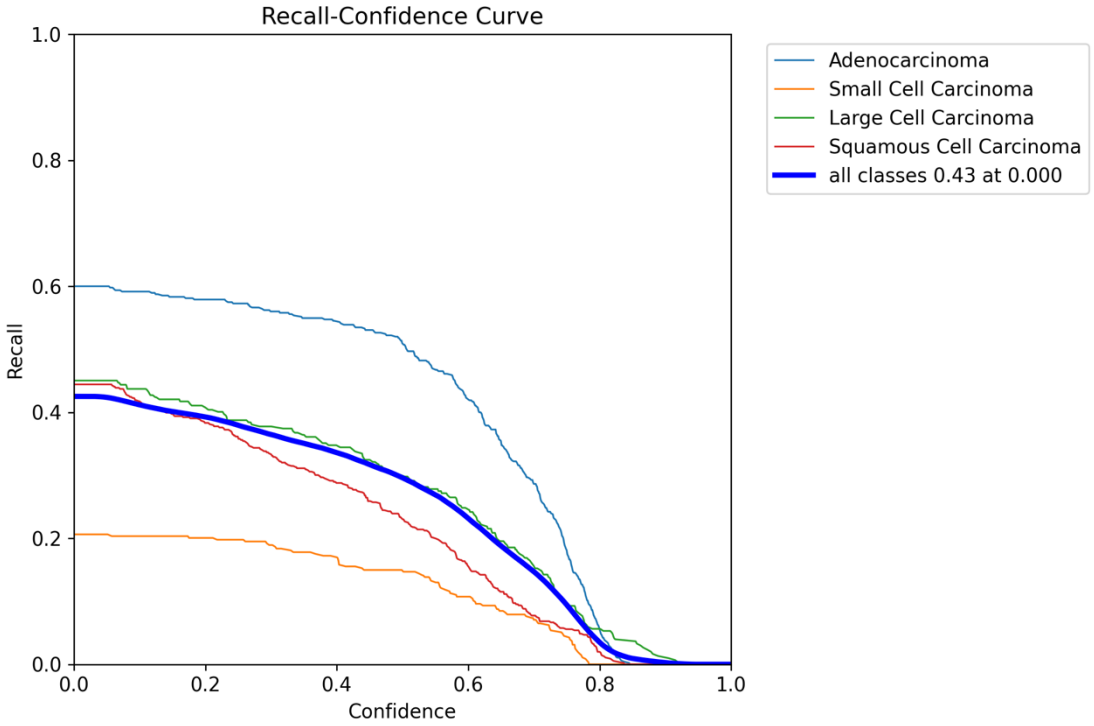


Figure B.3 - Recall-Confidence Curve

APPENDIX C: TMN STAGE CLASSIFICATION MODEL

Table C.1 - Class Distribution per TNM Target in the Training Set (Before Augmentation)

Target class	T-Stage	N-Stage	M-Stage
0	1802	3543	6368
1	3562	2684	2866
2	2441	1040	-
3	1429	1967	-

Table C.2 - Class Distribution per TNM Target in the Validation Set (Before Augmentation)

Target class	T-Stage	N-Stage	M-Stage
0	362	797	1373
1	683	604	464
2	542	168	-
3	250	268	-

Table C.3 - Class Distribution per TNM Target in the Test Set

Target class	T-Stage	N-Stage	M-Stage
0	752	577	1364
1	642	726	323
2	126	169	-
3	167	215	-

Table C.4 - Class Distribution per TNM Target in the Training Set (After Augmentation)

Target class	T-Stage	N-Stage	M-Stage
0	2802	2843	5868
1	2862	2984	5566
2	2941	2640	-
3	2829	2967	-

Table C.5 - Class Distribution per TNM Target in the Validation Set (After Augmentation)

Target class	T-Stage	N-Stage	M-Stage
0	422	437	943
1	403	504	864
2	532	428	-
3	450	438	-

Table C.6 - Hyperparameter Search Space and Best Values for TMN Classification Model (Optuna Optimization)

Hyperparameter	Search Space	Best Value Found
Learning rate	1e-6 to 5e-4 (log scale)	3.55e-06
Batch size	[16, 32]	32
Weight decay	1e-4 to 1e-1 (log scale)	0.00205
Trainable layers	1 to 4	4
Dropout Rate 1	0.3 to 0.9	0.3028
Dropout Rate 2	0.1 to 0.5	0.3177
Optimizer	['Adam', 'SGD', 'RMSprop']	RMSprop
Scheduler	['StepLR', 'CosineAnnealingLR', 'ReduceLROnPlateau', 'OneCycleLR']	ReduceLROnPlateau
Label smoothing	0.1 to 0.3	0.2443
Hidden units	[64, 128, 256]	256
Number of fc layers	1 to 3	2
Factor	0.1 to 0.5	0.1813
Patience	2 to 6	2

Table C.7 - Classification Report: Target T (Test set)

Class	Precision	Recall	F1-score	Support
0	0.30	0.15	0.20	752
1	0.37	0.50	0.43	642
2	0.01	0.02	0.01	126
3	0.30	0.31	0.30	167
Accuracy	-	-	**0.29**	1687
Macro avg	0.25	0.25	0.24	1687

Table C.8 - Confusion Matrix: Target T (Test set)

	Pred 0	Pred 1	Pred 2
True 0	110	514	126
True 1	155	324	76
True 2	86	7	3

Table C.9 - Classification Report: Target N (Test set)

Class	Precision	Recall	F1-score	Support
0	0.38	0.37	0.38	577
1	0.57	0.38	0.46	726
2	0.36	0.42	0.39	169
3	0.13	0.27	0.18	215
Accuracy	-	-	**0.37**	1687
Macro avg	0.36	0.36	0.35	1687

Table C.10 - Classification Report: Target N (Test set)

	Pred 0	Pred 1	Pred 2
True 0	215	209	29
True 1	174	277	64
True 2	54	1	71

Table C.11 - Classification Report: Target M (Test set)

Class	Precision	Recall	F1-score	Support
0	0.85	0.75	0.80	1364
1	0.30	0.45	0.36	323
Accuracy	-	-	**0.70**	1687
Macro avg	0.58	0.60	0.58	1687
Weighted avg	0.75	0.70	0.72	1687

Table C.12 - Confusion Matrix: Target M (Test set)

	Pred 0	Pred 1
True 0	1028	336
True 1	177	146

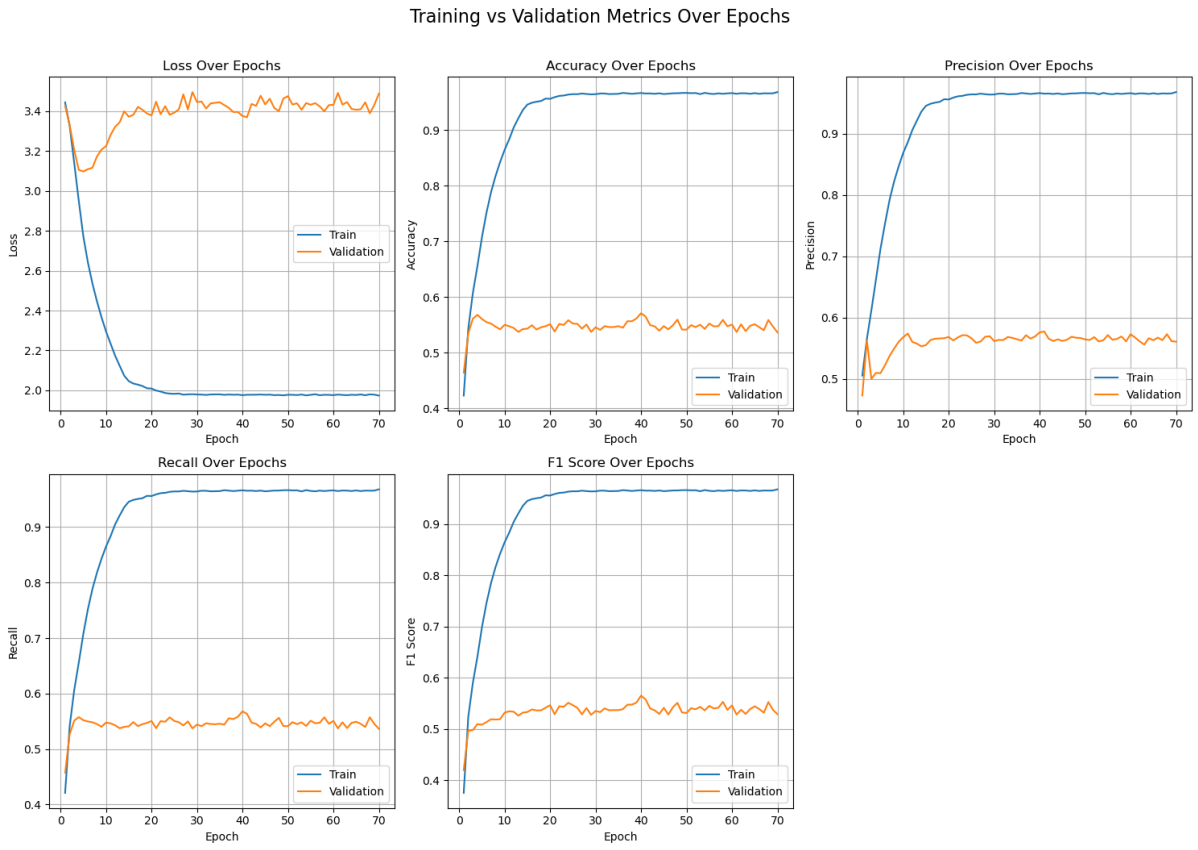


Figure C.1 - Training and Validation Performance Metrics Over Epochs

APPENDIX D: PROMPT DESIGN FOR CLINICAL REASONING BASED ON TNM STAGING

This appendix provides a breakdown of the prompt used to generate clinical recommendations based on TNM staging for lung cancer. Each row, from table D.1 presents a transcribed segment of the original prompt (column "Prompt Segment"), along with an explanation of its intended function (column "Reasoning").This structure was developed to ensure the assistant follows clinical logic, avoids unsupported reasoning, and presents treatment recommendations aligned with established oncology guidelines.

Table D.1 - Structure and Purpose of Prompt Instructions Used for Generating the Treatment Protocols

Step	Prompt Segment	Reasoning
Role	You are a clinical oncology assistant specialized in lung cancer.	This instruction defines the assistant's identity and clinical expertise. It sets the context for domain-specific reasoning aligned with oncology practices.
Tasks	Your tasks: 1. Determine the clinical TNM stage (I-IV, including substages A, B, or C) based on the AJCC 8th Edition staging system. 2. Generate a structured, evidence-based treatment plan according to the stage, histology, and type of lung cancer. 3. Use only information derived from retrieved clinical guidelines and peer-reviewed literature to support your reasoning. 4. Do not assume facts outside the provided information. 5. Use specific medical terminology, and name all treatments, radiotherapy modalities, and chemotherapy/immunotherapy regimens explicitly when referenced in guidelines.	These step-by-step rules ensure the assistant follows clinical guidelines, avoids unsupported assumptions (hallucinations), and uses clinical terminology.

Patient Information	Patient Information - Type of Cancer: {cancer_type} - Age: {age} - Gender: {gender} - Tumor (T) Stage: {t_stage} - Lymph Node (N) Stage: {n_stage} - Metastasis (M) Stage: {m_stage} - Histopathological Grade: {histopath_grade} - Additional Clinical Factors: {additional_info} (optional)	This section inputs patient clinical data into the prompt. It ensures responses are specific to each case and that staging and treatment decisions are grounded in actual patient characteristics.
NSCLC - TNM Classification	TNM Staging Classification (NSCLC) - Classify the patient's cancer using the AJCC 8th Edition TNM system. - Always specify the substage letter (A, B, or C) when reporting the stage (e.g., Stage IIIB, IVA).	This segment guides the assistant when the cancer type is NSCLC. Including substages (A, B, or C) ensures that staging is detailed and not limited to general levels (I, II, III, IV), which is essential for accurate treatment recommendations.
NSCLC - Treatment Strategy	Evidence-Based Treatment Strategy (NSCLC) - Stage I-II: Guide curative options such as surgery, SBRT, and neoadjuvant/adjuvant chemotherapy. - Stage III: Distinguish between resectable and unresectable disease; outline multimodal treatment including chemoradiotherapy and surgery. - Stage IV: Systemic treatment plan based on line of therapy and biomarkers. - Non-Surgical Management: Define alternatives such as SBRT and EBRT. - Clinical Trials: Mention when appropriate. - Palliative Care and Follow-Up: Provide surveillance and symptom management protocols.	This response template organizes treatment strategies by stage and supports guideline-based recommendations. It provides clarity for constructing responses and serves to ensure completeness and medical relevance in NSCLC cases.

SCLC - TNM Classification	TNM and Traditional Stage Classification (SCLC) - Classify as Limited-Stage (LS-SCLC) or Extensive-Stage (ES-SCLC). - Justify using AJCC 8th Edition criteria and anatomical details.	SCLC uses a binary staging approach. This section directs the assistant to classify between Limited-Stage and Extensive-Stage SCLC based on established clinical criteria.
SCLC - Treatment Strategy	Evidence-Based Treatment Strategy (SCLC) - LS-SCLC: Recommend concurrent chemoradiation, PCI, and optional surgery. - ES-SCLC: Guide systemic treatment and radiation strategies. - Age ≥70: Recommend geriatric adaptations. - Palliative Care and Follow-Up: Mention protocols.	This part gives treatment strategies examples for each SCLC stage, including adjustments for older patients to guide the response.
Final Output Requirements	Final Structured Output 1. Clinical Stage: e.g., Stage IIIB NSCLC or ES-SCLC. 2. Treatment Plan: Evidence-based and stage-specific. 3. Therapies: Name drugs, radiotherapy modalities, immunotherapies. 4. Clinical Trials: Identify if appropriate. 5. Palliative/Supportive Care: Outline when to integrate. 6. Follow-Up Plan: Imaging and biomarker recommendations.	This enforces the structure pretended for the output. It makes sure all care components are covered and in a clear format.

APPENDIX E: TREATMENT PROTOCOL MODEL RESULTS

This appendix summarizes the evaluation results for the three best-performing RAG pipeline configurations used in treatment protocol generation. The results show the impact of varying Top-k and temperature parameters on model performance. Configurations are ranked from best to worst based on the mean performance across all evaluation metrics, including Stage Match mean, BERTScore F1, and key RAGAS metrics: Faithfulness, Answer Relevancy, Context Precision, and Context Recall. For the best-performing model (which uses Gemini embeddings (text-embedding-004), a hybrid retrieval method (BM25 + ChromaDB), and the Gemini LLM) additional evaluation metrics, including a confusion matrix and a classification report, are provided.

Table E.1 - Evaluation results for different configurations of the RAG pipeline using Gemini embeddings (text-embedding-004), a hybrid retrieval method (BM25 + ChromaDB), and the Gemini LLM

Top K	Tempe-rature	Stage match*	BERT-Score F1	RAGAS Faith-fullness	RAGAS Answer Relevancy	RAGAS Context Precision	RAGAS Context Recall
15	0.0	0,54	0,83	0,68	0,81	0,99	0,93
15	0.3	0,52	0,83	0,64	0,80	0,99	0,95
15	0.7	0,50	0,83	0,62	0,81	0,99	0,97
15	0.5	0,50	0,83	0,62	0,80	0,99	0,97
10	0.7	0,54	0,83	0,54	0,82	1,00	0,92
10	0.3	0,53	0,83	0,56	0,80	1,00	0,92
10	0.5	0,50	0,83	0,53	0,81	1,00	0,92
10	0.0	0,50	0,83	0,49	0,81	1,00	0,92
7	0.7	0,56	0,83	0,41	0,80	1,00	0,45
7	0.5	0,54	0,83	0,42	0,80	1,00	0,45
5	0.7	0,53	0,83	0,42	0,82	1,00	0,43
7	0.0	0,52	0,83	0,42	0,81	1,00	0,43
5	0.5	0,50	0,83	0,42	0,81	1,00	0,43
7	0.3	0,53	0,83	0,40	0,80	1,00	0,43
5	0.3	0,50	0,83	0,38	0,81	1,00	0,43
5	0.0	0,48	0,83	0,38	0,82	1,00	0,42

Table E.2 - Evaluation results for different configurations of the RAG pipeline using OpenAI embeddings(text-embedding-ada-002), a hybrid retrieval method (BM25 + ChromaDB), and the Gemini LLM

Top K	Tempe-rature	Stage match*	BERT-Score F1	RAGAS Faith-fullness	RAGAS Answer Relevancy	RAGAS Context Precision	RAGAS Context Recall
15	0.7	0,536	0,83	0,65	0,81	0,99	0,96

Top K	Tempe-rature	Stage match*	BERT-Score F1	RAGAS Faith-fullness	RAGAS Answer Relevancy	RAGAS Context Precision	RAGAS Context Recall
15	0.0	0,528	0,83	0,65	0,80	0,99	0,97
15	0.5	0,517	0,83	0,61	0,80	0,99	0,98
15	0.3	0,469	0,83	0,64	0,81	0,99	0,97
10	0.0	0,528	0,83	0,57	0,81	1,00	0,93
10	0.3	0,506	0,83	0,58	0,82	0,99	0,92
10	0.7	0,517	0,83	0,56	0,80	1,00	0,94
10	0.5	0,517	0,83	0,55	0,80	1,00	0,92
5	0.3	0,542	0,83	0,39	0,81	1,00	0,51
5	0.0	0,522	0,83	0,40	0,81	1,00	0,51
7	0.7	0,569	0,83	0,46	0,80	1,00	0,39
5	0.5	0,483	0,83	0,38	0,81	1,00	0,54
7	0.5	0,564	0,83	0,45	0,81	1,00	0,39
5	0.7	0,519	0,83	0,37	0,82	1,00	0,46
7	0.0	0,514	0,83	0,46	0,81	1,00	0,39
7	0.3	0,517	0,83	0,41	0,80	1,00	0,39

Table E.3 - Evaluation results for different configurations of the RAG pipeline using OpenAI embeddings(text-embedding-ada-002), BM25 retrieval method, and the Gemini LLM.

Top K	Tempe-rature	Stage match*	BERT-Score F1	RAGAS Faith-fullness	RAGAS Answer Relevancy	RAGAS Context Precision	RAGAS Context Recall
15	0.7	0,51	0,83	0,66	0,80	0,99	0,98
15	0.3	0,50	0,83	0,64	0,79	0,99	0,99
15	0.0	0,50	0,83	0,64	0,80	0,99	0,97
15	0.5	0,47	0,83	0,63	0,81	1,00	0,97
10	0.0	0,55	0,83	0,55	0,80	1,00	0,95
10	0.3	0,53	0,83	0,55	0,80	1,00	0,94
10	0.5	0,51	0,83	0,53	0,82	1,00	0,93
10	0.7	0,50	0,83	0,55	0,80	1,00	0,93
5	0.7	0,51	0,83	0,43	0,81	1,00	0,57
5	0.5	0,51	0,83	0,41	0,81	1,00	0,55
5	0.3	0,50	0,83	0,42	0,80	1,00	0,54
5	0.0	0,49	0,83	0,38	0,82	1,00	0,54
7	0.0	0,52	0,83	0,45	0,81	1,00	0,42
7	0.3	0,54	0,83	0,44	0,80	1,00	0,39
7	0.7	0,55	0,83	0,43	0,82	1,00	0,37
7	0.5	0,53	0,83	0,42	0,80	1,00	0,39

Table E.4 - Confusion Matrix for the Treatment Generation pipeline using Gemini embeddings (text-embedding-004), a hybrid retrieval method (BM25 + ChromaDB), and the Gemini LLM

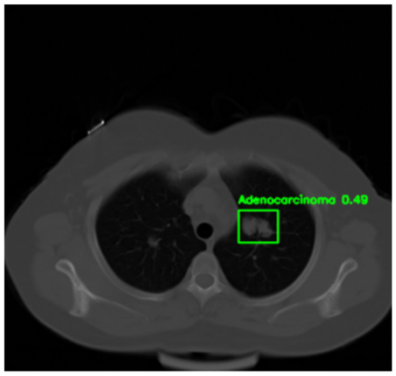
True stage	ES-SCLC	IA	IB	IIA	IIB	IIIA	IIIB	IIIC	IVA	IVB	IVC	LS-SCLC
ES-SCLC	20	0	0	0	0	0	0	0	0	0	0	0
IA	0	30	0	0	0	0	0	0	0	0	0	0
IB	0	0	10	0	0	0	0	0	0	0	0	0
IIA	0	1	7	2	0	0	0	0	0	0	0	0
IIB	0	0	0	36	18	6	0	0	0	0	0	0
IIIA	0	0	0	0	2	48	21	0	0	0	0	9
IIIB	0	0	0	0	0	1	32	25	0	0	0	12
IIIC	0	0	0	0	0	0	4	13	0	1	0	2
IVA	0	0	0	0	0	0	0	0	9	17	3	1
IVB	0	0	0	0	0	0	0	0	0	0	0	0
IVC	0	0	0	0	0	0	0	0	0	0	0	0
LS-SCLC	19	0	0	0	0	0	0	0	0	0	0	11

Table E.5 - Classification Report for the Treatment Generation pipeline using Gemini embeddings (text-embedding-004), a hybrid retrieval method (BM25 + ChromaDB), and the Gemini LLM

Class	Precision	Recall	F1-score	Support
ES-SCLC	0.51	1.0	0.68	20.0
IA	0.97	1.0	0.98	30.0
IB	0.59	1.0	0.74	10.0
IIA	0.05	0.2	0.08	10.0
IIB	0.9	0.3	0.45	60.0
IIIA	0.87	0.6	0.71	80.0
IIIB	0.56	0.46	0.50	70.0
IIIC	0.34	0.65	0.45	20.0
IVA	1.0	0.3	0.46	30.0
IVB	0.0	0.0	0.0	0.0
IVC	0.0	0.0	0.0	0.0
LS-SCLC	0.31	0.37	0.34	30.0
Accuracy	0.54	0.54	0.54	0.54
Macro avg	0.51	0.49	0.45	360.0
Weighted avg	0.71	0.54	0.55	360.0

APPENDIX F: EXAMPLE OUTPUT OF THE DEVELOPED DIAGNOSTIC AND TREATMENT PIPELINE

```
Speed: 2.2ms preprocess, 53.5ms inference, 0.5ms postprocess per image at shape (1, 3, 512, 512)
```




```
 TNM Prediction:
T: T0, N: N0, M: M1, Cancer Type: Adenocarcinoma

 Treatment Plan:
### 1. Clinical Stage:
Based on the provided information:
- T0: No evidence of primary tumor.
- N0: No regional lymph node involvement.
- M1: Distant metastasis present.

According to the AJCC 8th Edition staging system, the patient's clinical stage is **Stage IV**.

### 2. Treatment Plan:
Given the patient's Stage IV adenocarcinoma, the primary treatment approach is systemic therapy. The treatment plan should consider the patient's
performance status, PD-L1 expression, and the presence of any targetable mutations (EGFR, ALK, ROS1, BRAF, MET, RET, NTRK).

**First-Line Therapy:**
1.  **Molecular Testing:** The patient's tumor should undergo comprehensive molecular testing to identify actionable mutations such as EGFR, ALK,
ROS1, BRAF, MET, RET, and NTRK. According to the American Cancer Society, if any of these driver mutations are found, initial treatment with a tar
geted therapy (rather than chemotherapy) would be recommended.
2.  **PD-L1 Testing:** Assess PD-L1 expression on tumor cells.

    *   **If a targetable mutation is present (e.g., EGFR, ALK, ROS1, BRAF, MET, RET, NTRK):**
        *   Initiate treatment with the appropriate targeted therapy (e.g., EGFR inhibitor like Osimertinib, ALK inhibitor like Alectinib, ROS1 in
hibitor, BRAF inhibitor, MET inhibitor, RET inhibitor, or NTRK inhibitor).
    *   **If no targetable mutation is present and PD-L1 ≥ 50%:**
        *   Consider single-agent immunotherapy with Pembrolizumab.
    *   **If no targetable mutation is present and PD-L1 < 50%:**
        *   Consider combination chemoimmunotherapy with a platinum-based doublet (e.g., Cisplatin or Carboplatin plus Pemetrexed) and Pembrolizum
ab.

**Second-Line Therapy and Beyond:**
1.  **If the patient progresses on first-line targeted therapy:**
    *   Consider second-line chemotherapy (e.g., Pemetrexed if not previously used, Docetaxel, or Gemcitabine).
2.  **If the patient progresses on first-line chemoimmunotherapy:**
    *   Consider second-line chemotherapy (e.g., Docetaxel or Gemcitabine).
    *   If the patient did not receive immunotherapy in the first line and has PD-L1 expression ≥ 1%, consider single-agent immunotherapy with Niv
olumab or Atezolizumab.

### 3. Therapeutic Modalities:
-   **Chemotherapy Regimens:**
    -   Cisplatin + Pemetrexed
    -   Carboplatin + Pemetrexed
    -   Docetaxel
    -   Gemcitabine
-   **Targeted Therapies:**
    -   Osimertinib (EGFR inhibitor)
    -   Alectinib (ALK inhibitor)
-   **Immunotherapy:**
    -   Pembrolizumab (anti-PD-1)
    -   Nivolumab (anti-PD-1)
    -   Atezolizumab (anti-PD-L1)
-   **Radiotherapy:**
    -   Not typically used as a primary treatment modality in Stage IV NSCLC, but may be considered for palliative purposes to manage symptomatic
metastases (e.g., bone metastases, brain metastases).

### 4. Clinical Trial Considerations:
-   Patients with Stage IV NSCLC should be evaluated for enrollment in clinical trials. Trials may investigate novel targeted therapies, immunothe
rapeutic approaches, or combinations of therapies.

### 5. Palliative and Supportive Care:
-   Palliative care should be integrated early in the management of Stage IV NSCLC to address symptom control, improve quality of life, and provid
e psychosocial support.
-   Address pain management, dyspnea, fatigue, and other symptoms.
-   Provide counseling and support for patients and their families.
-   Advanced care planning and discussions about goals of care are essential.

### 6. Follow-Up Plan:
-   Regular follow-up appointments with the oncologist to monitor treatment response, assess for disease progression, and manage treatment-related
toxicities.
-   Imaging studies (e.g., CT scans, PET/CT scans) every 2-3 months to evaluate treatment response and monitor for disease progression.
-   Monitor for treatment-related side effects and provide appropriate supportive care.
```

Figure F.1 - Example Output of the Proposed Integrated Pipeline for Lung Cancer Detection, TNM Staging, and Treatment Planning

APPENDIX G: AI TOOLS AND RESEARCH PLATFORMS USED IN THIS STUDY

Table G.1 - Summary of Tools and Their Uses in the Research Process

Tool / Platform	Use in This Research
Scopus, **IEEE Xplore**, **ACM Digital Library**, **Web of Science**, **Consensus**	Used for searching and discovering relevant papers using smart keyword combinations. Consensus was particularly helpful to quickly understand paper content via summaries.
VOSviewer	Used to analyze relationships between papers (e.g., co-authorship, citations). A Scopus-exported dataset was uploaded to visualize clusters of related research**.**
Litmaps	Helped explore papers related to a key article by visualizing the citation network and finding seminal or follow-up studies**.**
Scite.ai	Used to check whether papers were supported or contradicted by others. Helpful to understand the strength or debate around a topic.
SciSpace (Typeset.io)	Uploaded papers to ask questions and clarify complex sections. The tool highlighted answers directly in the PDF, aiding comprehension.
Paperpal	Used to correct grammar, improve sentence clarity, and ensure proper academic tone during the writing and editing phases.
Mendeley	Reference management: storing, organizing, and inserting citations and bibliographies into the report.
ChatGPT (OpenAI)	Assisted in revising and improving writing, correcting grammar, structuring annexes, generating figure/table captions, and clarifying research summaries.

ANNEX I: YOLOV8 AND RESNET-50 MODEL ARCHITECTURES

This annex provides visual schematics of the two DL architectures used in this study: YOLOv8 for lung cancer detection and classification and ResNet-50 for classification.

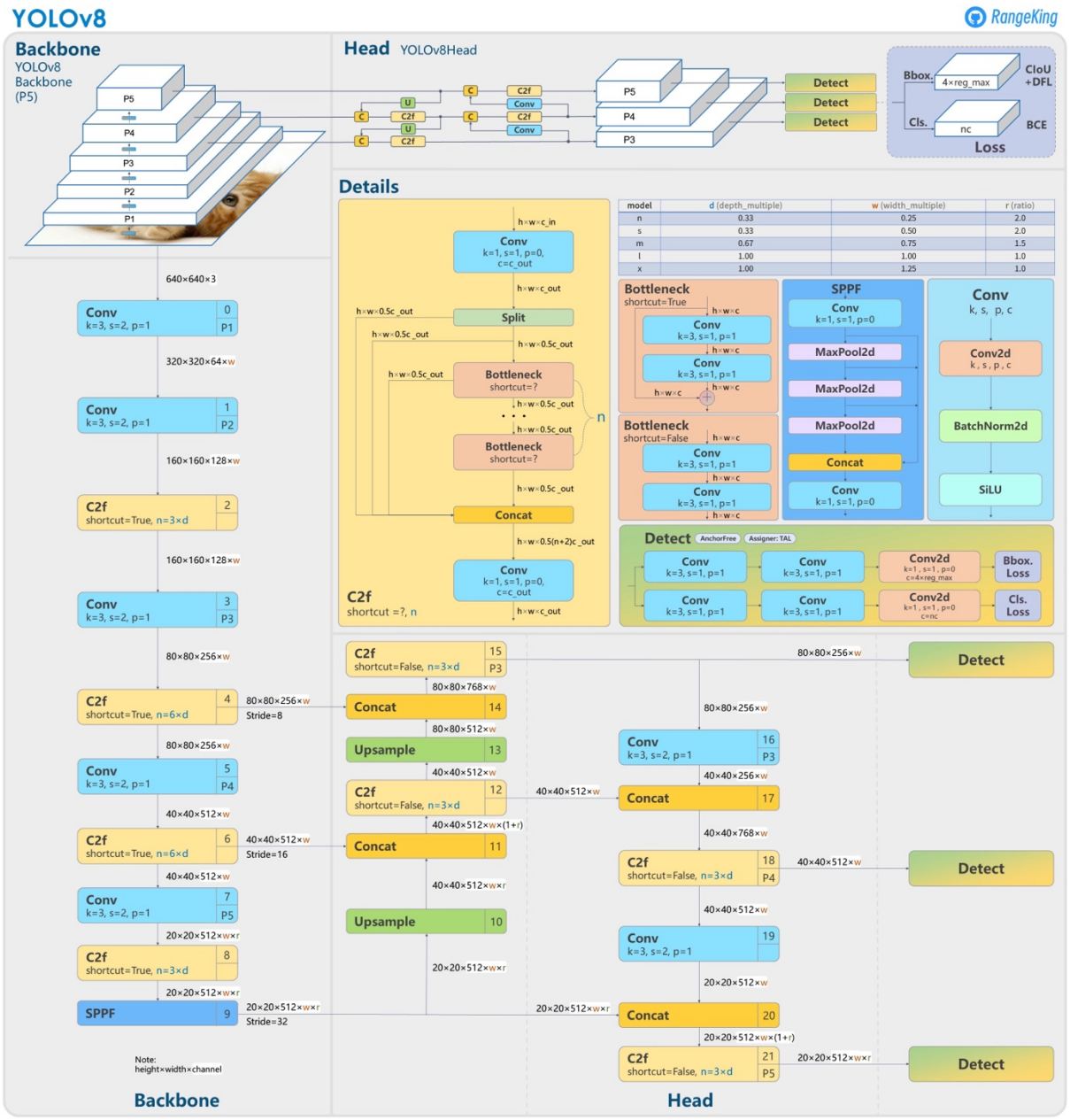


Figure I.1 - YOLOv8 Model Architecture (Backbone, Neck, Head)

Source: Ultralytics, 2023

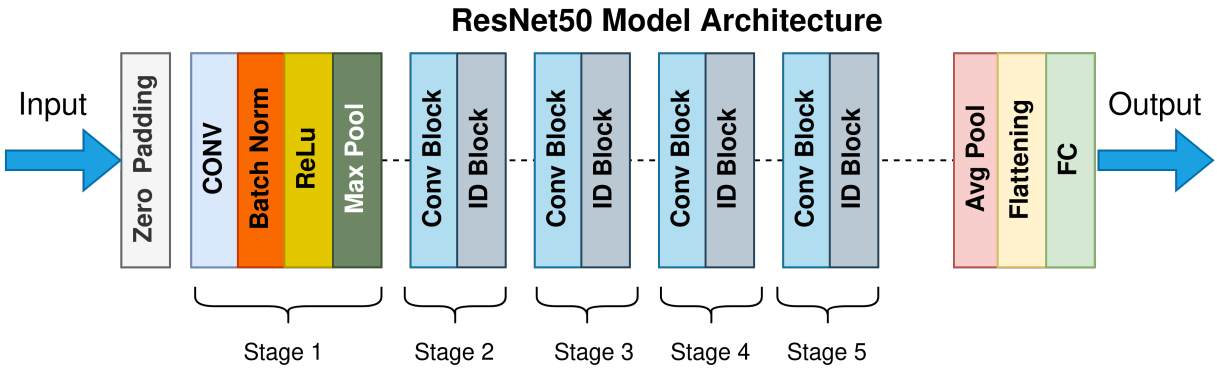


Figure I.2 - ResNet-50 Model Architecture

Source: Lababede & Meziane (2018)

ANNEX II: TNM CLASSIFICATION TABLES FOR LUNG CANCER (8TH EDITION)

This annex presents the official tables from the 8th edition of the TNM classification system for lung cancer, as published by the IASLC. These tables define the descriptors for Primary Tumor (T), Regional Lymph Nodes (N), and Distant Metastasis (M), as well as the combined Stage Groups (IA to IVB). These definitions form the basis for clinical staging and were used to interpret, label, and stratify the cases in the present study.

Table II.1 - Primary Tumor (T), Lymph Node (N), Metastasis (M) Descriptors (Eighth edition of TNM staging of lung cancer)

Primary tumor (T)	
T category	**Definition**
Tx	Tumor that is proven histopathologically (malignant cells in bronchopulmonary secretions/washings) but cannot be assessed or is not demonstrable radiologically or bronchoscopically.
T0	No evidence of primary tumor.
Tis	Carcinoma in situ: Squamous cell carcinoma in situ. Adenocarcinoma in situ (pure lepidic pattern and $\leq$3 cm in greatest dimension).
T1	Size: $\leq$3 cm. Airway location: in or distal to the lobar bronchus. Local invasion: none (surrounded by lung or visceral pleura). Subdivisions: T1mi: Minimally invasive adenocarcinoma (pure lepidic pattern, $\leq$3 cm in greatest dimension and $\leq$5 mm invasion)—T1a (size $\leq$1 cm)[a]—T1b (1 cm $<$ size $\leq$ 2 cm)—T1c (2 cm $<$ size $\leq$ 3 cm).
T2	Any of the following characteristics: Size: $>$3 cm but $\leq$5 cm. Airway location: invasion of the main bronchus (regardless the distance to the carina) or presence of atelectasis or obstructive. Pneumonitis that extends to hilar region (whether it is involving part or the entire lung). Local invasion: visceral pleura (PL1 or PL2). Subdivisions: T2a (3 cm $<$ size $\leq$ 4 cm or cannot be determined) and T2b (4 cm $<$ size $\leq$ 5 cm).
T3	Any of the following characteristics: Size: $>$5 cm but $\leq$7 cm. Local invasion: direct invasion of chest wall (including superior sulcus tumors), parietal pleura (PL3), phrenic nerve, or parietal pericardium. Separate tumor nodule(s) in the same lobe of the primary tumor.
T4	Any of the following characteristics: Size $>$7 cm. Airway location: invasion of the carina or trachea. Local invasion: diaphragm, mediastinum, heart, great vessels, recurrent laryngeal nerve, esophagus or vertebral body. Separate tumor nodule(s) in an ipsilateral different lobe of the primary tumor.
Lymph nodes (N)	
Descriptor	**Definition**
Nx	Regional lymph nodes cannot be evaluated.
N0	No regional lymph nodes involvement.
N1	Involvement of ipsilateral peribronchial and/or ipsilateral hilar lymph nodes (includes direct extension to intrapulmonary nodes).
N2	Involvement of the ipsilateral mediastinal and/or subcarinal lymph nodes.
N3	Involvement of any of the following lymph node groups: contralateral mediastinal, contralateral hilar, ipsilateral or contralateral scalene, or supraclavicular nodes.
Distant metastasis (M)	
Descriptor	**Definition**
M0	No distant metastasis.
M1	Presence of distant metastasis. Subdivisions: M1a (separate tumor nodule(s) in a contralateral lobe to that of the primary tumor or tumors with pleural or pericardial nodules or malignant effusion); M1b (single extrathoracic metastasis); M1c (multiple extrathoracic metastases to one or more organs).

Source: Lababede & Meziane (2018)

Table II.2 - Lung Cancer Stage Grouping (8th Edition)

Stage group	
Occult carcinoma	(TxN0M0)
Stage 0	(TisN0M0)
Stage IA1	(T1aN0M0) (T1(mi)N0M0)
Stage IA2	(T1bN0M0)
Stage IA3	(T1cN0M0)
Stage IB	(T2aN0M0)
Stage IIA	(T2bN0M0)
Stage IIB	(T (1–2)N1M0) (T3N0M0)
Stage IIIA	(T(1–2)N2M0) (T3N1M0) (T4N(0–1)M0)
Stage IIIB	(T(1–2)N3M0) (T(3–4)N2M0)
Stage IIIC	(T(3–4)N3M0)
Stage IVA	(Any T, Any N, M1a,b)
Stage IVB	(Any T, Any N, M1c)

Source: Lababede & Meziane (2018)

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa