THE UNIVERSITY OF DANANG
UNIVERSITY OF SCIENCE AND TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

GRADUATION PROJECT THESIS

MAJOR: INFORMATION TECHNOLOGY

SPECIALTY: DATA SCIENCE AND ARTIFICIAL INTELLIGENCE

PROJECT TITLE:

MEDCAPSYS: A NOVEL APPROACH TO HIGHLY DETAILED BRAIN MEDICAL ANALYSIS

Instructor: **DR. NGUYỄN VĂN HIỆU**
Student: **PHAN MINH NHẬT**
Student ID: **102210095**
Class**:** **21TCLC_KHDL**

Da Nang, June/2025

THE UNIVERSITY OF DANANG
UNIVERSITY OF SCIENCE AND TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

GRADUATION PROJECT THESIS

MAJOR: INFORMATION TECHNOLOGY

SPECIALTY: DATA SCIENCE AND ARTIFICIAL INTELLIGENCE

PROJECT TITLE:

MEDCAPSYS: A NOVEL APPROACH TO HIGHLY DETAILED BRAIN MEDICAL ANALYSIS

Instructor: **DR. NGUYỄN VĂN HIỆU**
Student: **PHAN MINH NHẬT**
Student ID: **102210095**
Class**:** **21TCLC_KHDL**

Da Nang, June/2025

ABSTRACT

Topic title: MedCapSys: A Novel Approach To Highly Detailed Brain Medical Image Analysis

Student name: Phan Minh Nhat

Student ID: 102210095 Class: 21TCLC_KHDL

Accurate analysis of brain medical imaging plays a pivotal role in supporting clinical diagnostic processes, particularly in the identification of pathological features and the generation of coherent textual interpretations from MRI scans. This study presents MedCapSys, a comprehensive, multi-module system designed to integrate visual and textual modalities to enhance brain diagnostic workflows. MedCapSys facilitates automated interpretation of brain magnetic resonance imaging (MRI) and incorporates an intelligent chatbot interface capable of delivering diagnostic insights in natural language, thereby improving accessibility for both clinicians and patients.

The system comprises four key components: MedCapNet, GuidedDCNet, GuidedSegDiff, and BrainMedQwen. These modules are strategically designed to extract and analyze critical elements commonly included in radiological brain MRI reports, specifically: (1) imaging view orientation (axial, sagittal, coronal), (2) MRI pulse sequence classification (e.g., T1-weighted, T2-weighted, FLAIR), (3) identification of hyperintense regions, (4) quantification of lesions, and (5) detailed lesion characterization, including lesion type, size, presumed etiology, and the presence of necrosis.

Collectively, these components underpin an interactive medical chatbot capable of responding to clinical queries, elucidating radiological findings, and offering preliminary diagnostic assistance based on MRI data. The proposed system demonstrates significant promise in advancing automated brain imaging interpretation and supporting data-driven clinical decision-making in neurodiagnostics.

THE UNIVERSITY OF DANANG
UNIVERSITY OF SCIENCE AND TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

THE SOCIALIST REPUBLIC OF VIETNAM
Independence - Freedom - Happiness

GRADUATION PROJECT REQUIREMENTS

Student Name: Phan Minh Nhat Student ID: 102210095 Class: 21TCLC_KHDL
Faculty: Information Technology Major: Information Technology – Specialty of Data Science and Artificial Intelligence

1. Topic title: MedCapSys: A Novel Approach To Highly Detailed Brain Medical Image Analysis

2. Project topic : ☐ *has signed intellectual property agreement for final result*

3. Initial figure and data: None

4. Content of the explanations and calculations:

- Introduction: Presents the motivation, objectives, scope, methodology, and structure of the thesis.
- Chapter 1 – Theoretical Background: Present the thesis motivation, objectives, scope, and methodology, along with an overview of brain medical image analysis, AI techniques in healthcare, and related studies.
- Chapter 2 – Methodology: Describes the overall system architecture, each module's design, and applied technologies.
- Chapter 3 – Implementation and Evaluation: Detail the data preprocessing, training, system integration, and evaluate the performance of the proposed system in comparison with existing solutions.
- Conclusion: Summarizes key findings and outlines directions for further development.

5. Drawings, charts (specify the types and sizes of drawings): None

6. Name of instructor: Dr. Nguyen Van Hieu

7. Date of assignment : .../.../2025

8. Date of completion : .../.../2025

Danang, date month year 2025

Head of Division……. **Instructor**

ACKNOWLEDGEMENTS

In recent years, the rapid advancement of Artificial Intelligence (AI) and Deep Learning technologies has opened up new possibilities in the field of medicine, particularly in the analysis of medical images. The integration of AI models into brain imaging diagnostics has significantly improved the accuracy of abnormality detection and has become an essential tool in supporting clinical decision-making. However, transforming complex imaging data into meaningful and interpretable textual descriptions remains a major challenge.

Motivated by this need, the project titled "MedCapSys: A Novel Approach To Highly Detailed Brain Medical Image Analysis" was developed with the objective of building an integrated system capable of analyzing brain MRI scans and automatically generating diagnostic reports in natural language. The system not only assists radiologists in interpreting medical images but also provides an intelligent chatbot interface that explains findings and answers diagnostic queries.

This work would not have been possible without the generous support and invaluable expertise of several individuals. I would like to express my deepest gratitude to Dr. Nguyen Van Hieu for his dedicated guidance, insightful feedback, and constant encouragement throughout the course of this project. His mentorship has been instrumental in shaping both the direction and quality of this research. I am also sincerely grateful to MSc. Vo Thi Minh Tri and MSc. Pham Thi Ngoc Trinh, medical doctors and lecturers in the Department of Radiology, The University of Danang – School of Medicine and Pharmacy, for their enthusiastic support, insightful suggestions, and valuable contributions throughout the course of this project. Their contributions greatly enriched the development and refinement of this work.

Despite all efforts to ensure the completeness and rigor of the study, I acknowledge that there may still be limitations. I warmly welcome any feedback or suggestions from instructors, experts, and readers to help further improve the quality of this research.

DECLARATION

I hereby declare that the research project titled “MedCapSys: A Novel Approach To Highly Detailed Brain Medical Image Analysis” has been conducted independently and honestly, and all the work presented in this document is the result of my own effort, except where otherwise stated and duly referenced. The content, data, analyses, and conclusions contained herein are original and have not been submitted previously, either in whole or in part, for any degree or professional qualification at any other institution.

All sources of information and assistance have been properly acknowledged, and the research complies with the ethical standards and academic integrity required by The University of Danang – University of Science and Technology.

I take full responsibility for the integrity and authenticity of the work presented.

Student

Phan Minh Nhat

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

Acronym	Explaination
AI	Artificial Intelligence
MRI	Magnetic Resonance Imaging
FLAIR	Fluid Attenuated Inversion Recovery
CT	Computed Tomography
DDPM	Denoising Diffusion Probabilistic Model
CNN	Convolutional Neural Network
DDAE	Denoising Diffusion Autoencoder
LLM	Large Language Model
VLM	Vision Language Model
ViT	Vision Transformer
VQA	Visual question answering
RNN	Recurrent Neural Networks
LSTM	Long Short-Term Memory
ANN	approximate nearest neighbor
MLP	Multi-Layer Perceptron
Cross MHSA	Cross Multi Head Self-Attention
MSE	mean squared error
SHAP	Shapley Additive Explanations
ROCOv2	Radiology Objects in COntext version 2
PSO	Particle Swarm Optimization
CDF	cumulative density function
SFT	supervised fine-tuning
DPO	Direct Preference Optimization
RLAF	Reinforcement Learning from AI Feedback
IoU	Intersection over Union
HD95	95th percentile Hausdorff Distance
SOTA	state-of-the-art

INTRODUCTION

1.1. Motivation

The analysis of brain medical images, particularly MRI scans, plays a pivotal role in diagnosing and monitoring neurological disorders such as tumors, strokes, multiple sclerosis, and neurodegenerative diseases. As the global burden of neurological conditions continues to rise, the demand for accurate, fast, and interpretable diagnostic tools has become more pressing than ever. Despite significant advancements in medical imaging technologies, the interpretation of these images still heavily depends on expert radiologists, whose availability and consistency can vary across regions and institutions.

Traditional image analysis methods, while effective to a certain extent, face several persistent limitations. Firstly, they often require extensive manual annotation and domain-specific expertise, which makes the process time-consuming and labor-intensive. Secondly, many existing automated systems are limited to visual outputs, such as heatmaps or segmentation masks, which may lack clinical interpretability for non-technical users. Furthermore, these systems rarely provide contextual explanations or justifications for their outputs, making it difficult for medical professionals to trust and act upon their recommendations.

In clinical settings, physicians often rely not only on visual assessments but also on narrative descriptions to make informed decisions. Therefore, there is a critical need for intelligent systems that can bridge the gap between visual content and linguistic reasoning. Such systems should be capable of not only detecting and segmenting abnormalities but also explaining findings in a clear, concise, and clinically relevant manner, emulating the way human experts communicate medical insights.

To address these challenges, this thesis proposes MedCapSys, an AI-powered system designed to generate meaningful, human-like textual reports directly from brain MRI images. By integrating state-of-the-art techniques in deep learning, natural language processing, and medical image analysis, MedCapSys aims to automate and enrich the diagnostic workflow. It features multiple advanced components, including automated lesion detection, semantic segmentation, and natural language captioning, all working in unison to enhance the interpretability and usability of brain MRI data.

This motivation stems from a dual objective: to support medical professionals with high-quality, explainable diagnostic assistance, and to contribute to the broader goal of making AI-driven healthcare more transparent, efficient, and accessible. In doing so, MedCapSys holds the potential to reduce diagnostic workload, minimize human error, and ultimately improve patient outcomes in neurology.

1.2. Goals

The core objective of this thesis is to design and implement MedCapSys, a comprehensive and intelligent system for the automated analysis of brain medical images. This system aims to support clinical diagnosis by generating accurate, detailed, and human-interpretable textual descriptions from brain scans, particularly MRI images. The system is envisioned to function not only as a powerful diagnostic aid for medical professionals but also as a research and educational tool in the field of medical AI.

To achieve this, MedCapSys is developed as a multi-module framework that leverages recent advancements in deep learning, computer vision, diffusion models, and multimodal vision-language understanding. Each module addresses a specific subtask involved in medical image interpretation, from caption generation and lesion classification to segmentation and language-based report synthesis, ensuring a modular, extensible, and clinically relevant design.

More specifically, the objectives of this thesis are:

- To bridge the gap between visual information in medical imaging and textual clinical interpretation, by developing an integrated system that translates visual data into coherent, informative medical descriptions.
- To build a robust pipeline capable of handling real-world, heterogeneous medical imaging data, including varying imaging modalities, resolutions, and pathological conditions.
- To enhance the explainability and interpretability of AI-generated outputs, enabling clinicians to trust and understand model decisions through clear language outputs that mirror traditional radiology reports.
- To demonstrate the effectiveness of combining multiple deep learning paradigms, such as vision transformers, diffusion models, segmentation networks, and large multimodal language models, for holistic brain image analysis.
- To facilitate a semi-automated diagnostic workflow that can assist radiologists in detecting and evaluating neurological abnormalities with greater accuracy, efficiency, and confidence.

- To provide a foundation for future research in AI-assisted radiology, by contributing an open, modular system architecture that can be further extended, evaluated, or applied in other medical imaging domains.

By achieving these objectives, this thesis aims to contribute both theoretically and practically to the development of intelligent diagnostic systems, and to support the broader goal of integrating AI into routine clinical workflows in a responsible and meaningful way.

1.3. Scope

The scope of this thesis is defined by both the research focus and the practical boundaries of developing MedCapSys, an intelligent system for the automated analysis and interpretation of brain medical images. The system is designed to assist in diagnostic decision-making by translating complex visual information into meaningful, structured, and clinically relevant textual descriptions.

1.3.1. Research Scope

From a research perspective, the thesis encompasses several key technical areas within artificial intelligence and medical image computing, including:

- Medical Image Captioning: Focused on generating natural language descriptions that capture critical anatomical and pathological details present in brain MRI scans.
- Lession Classification: Utilizing deep neural networks with guided diffusion mechanisms to accurately classify brain tumors into relevant categories based on radiological features.
- Lesion Segmentation: Developing models to automatically detect and segment abnormal regions such as tumors or hemorrhages, with precise localization and boundary delineation.
- Multimodal Vision-Language Understanding: Integrating multimodal learning techniques to combine visual data from scans with linguistic context, enabling the generation of coherent, diagnostically relevant responses.
- Natural Language Generation for Clinical Reporting: Designing a system capable of producing multilingual diagnostic reports that mirror the style and structure of expert-written radiology notes.

These research components are implemented through four key modules: MedCapNet, GuidedDCNet, GuidedSegDiff, and BrainMedQwen. Each module is developed, trained, evaluated, and integrated into a unified system to provide a seamless diagnostic experience.

1.3.2. Application Scope

In terms of real-world applicability, MedCapSys is targeted toward specific user groups and clinical scenarios:

- Healthcare Professionals: The system is intended to support radiologists, neurologists, and other clinicians in detecting and interpreting abnormalities in brain images, thereby improving diagnostic accuracy and reducing workload.
- Medical Education: MedCapSys can serve as an educational tool for medical students and trainees, offering illustrative, AI-generated annotations and explanations to aid in learning radiological concepts.
- Clinical Decision Support: While not intended to replace human experts, the system can function as a second-opinion tool, providing initial assessments and highlighting potentially critical findings for further review.

1.3.3. Data Scope

The thesis primarily focuses on the analysis of brain MRI images, covering a wide range of clinical conditions including:

- Brain tumors (e.g., gliomas, meningiomas)
- Hemorrhages and vascular abnormalities
- Structural deformities and congenital anomalies
- Edema and necrotic regions

The medical images are sourced from publicly available datasets and supplemented, where necessary, by academic materials such as case studies and annotated radiology reports. All data used undergo preprocessing to standardize input quality and enable effective model training and evaluation.

1.4. Thesis Structure

This thesis is organized into six chapters as follows:

- Chapter 1 – Introduction and Theoretical Background: Present the thesis motivation, objectives, scope, and methodology, along with an overview of brain medical image analysis, AI techniques in healthcare, and related studies.
- Chapter 2 – Methodology: Describes the overall system architecture, each module's design, and applied technologies.
- Chapter 3 – Implementation and Evaluation: Detail the data preprocessing, training, system integration, and evaluate the performance of the proposed system in comparison with existing solutions.

- Conclusion: Summarizes key findings and outlines directions for further development.

CHAPTER 1: THEORETICAL BACKGROUND

1.1. Diffusion Model

1.1.1. Theoretical Background

Denoising Diffusion Probabilistic Models (DDPMs) [1] are a class of generative models that learn to model the data distribution $p(x)$ by simulating a diffusion process. This framework is inspired by nonequilibrium thermodynamics and operates through a two-step process: (1) a forward diffusion process that gradually corrupts the data, and (2) a reverse denoising process that reconstructs data samples from pure noise.

The forward diffusion process is defined as a Markov chain that adds Gaussian noise to the input data x_0 over a fixed number of steps T. At each time step $t \in \{1,2,\dots,T\}$, a small amount of noise is added, resulting in a progressively noisier sample x_t. This is modeled as:

$$q(x_t|x_{t-1}) = \mathcal{N}\left(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I\right)$$

where $\beta_t \in (0,1)$ denotes a variance schedule (often linearly or cosine-increasing), and I is the identity matrix.

By leveraging the property of Gaussian distributions, the sample at any timestep t can be directly computed from the original data x_0:

$$q(x_t|x_0) = \mathcal{N}\left(x_t; \sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)I\right)$$

where $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_t$.

The generative modeling task involves learning the reverse process $p_\theta(x_{t-1}|x_t)$, which attempts to iteratively denoise x_t back to x_0. This reverse process is also parameterized as a Gaussian:

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}\left(x_{t-1}; \mu_\theta(x_t, t), \sum_\theta(x_t, t)\right)$$

In practice, the model learns to predict the noise ϵ added at each step. A neural network $\epsilon_\theta(x_t, t)$ is trained to minimize the denoising score matching objective:

$$\mathcal{L}_{simple} = \mathbb{E}_{x_o,t,\epsilon}\left[\left\|\epsilon - \epsilon_\theta\left(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t\right)\right\|^2\right]$$

This formulation simplifies the variational bound of the likelihood and is often referred to as the “simplified training loss”. To generate new data, a sample is drawn from a Gaussian prior $x_T \sim \mathcal{N}(0, I)$ and the model recursively denoises it using the

learned reverse process to obtain x_0. The full sampling process involves T steps, each corresponding to one reverse-time denoising step.

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t, t)\right) + \sigma_t z, \quad z \sim \mathcal{N}(0, I)$$

DDPMs are closely related to score-based generative models and Langevin dynamics. Specifically, they learn a time-dependent score function $\nabla_{x_t} \log q(x_t)$ and generation corresponds to solving a stochastic differential equation (SDE) backward in time. This insight has led to variants like score-based SDEs and probability flow ODEs, which unify and extend the theoretical foundation of diffusion models.

1.1.2. Applications in Generative Modeling

The foundational work by Ho et al. [1] demonstrated the effectiveness of DDPMs in image generation, achieving performance on par with GANs on tasks such as 256×256 image synthesis and CIFAR10 classification. Building on this, models like Imagen [2] and Stable Diffusion [3] have pushed the boundaries of text-to-image generation, showcasing highly photorealistic outputs through advanced conditioning and latent-space designs.

Diffusion models have also been extended to video generation, where capturing temporal consistency is critical. Methods such as Video Diffusion Models [4], [5], [6] effectively model sequential frames, offering strong performance in both unconditional and conditional video synthesis.

The adaptability of diffusion models has led to impactful applications in medical imaging. In 3D image generation, Khader et al. [7] proposed a latent-space DDPM tailored for MRI and CT data, achieving high-quality volumetric reconstructions. These approaches preserve clinical relevance while navigating the challenges of limited labeled data and modality-specific variations.

While initially developed for generative modeling, diffusion models have recently shown great promise in discriminative tasks such as classification and segmentation. The D2C framework [8] was among the first to explore this direction, demonstrating few-shot classification by leveraging the generative nature of diffusion. SpectralDiff [9] extended this idea to hyperspectral image classification, significantly boosting accuracy through spectral-aware designs.

In the medical domain, DiffMIC-v2 [10] pioneered diffusion-based classification across various imaging modalities, demonstrating not only superior accuracy but also a 3× improvement in computational efficiency compared to traditional CNNs. Another significant contribution is the Denoising Diffusion

Autoencoder (DDAE) [11], which unifies generative and discriminative capabilities within a single model, achieving high performance on benchmarks such as CIFAR10 (95.9%) and Tiny-ImageNet (50.0%) without auxiliary encoders.

Moreover, DDPMs have proven effective in semantic segmentation tasks [12], [13], offering fine-grained prediction capabilities by leveraging diffusion-based inference. These results underscore the versatility of diffusion models in a broad range of vision tasks.

Beyond visual applications, diffusion models have shown potential in handling discrete and categorical data, notably in natural language processing [14], [15]. Research into general-purpose diffusion architectures [16], [17] has revealed their capability to support a wide spectrum of tasks, ranging from retrieval and classification to editing and translation, under a unified modeling framework.

This progression reflects a paradigm shift from task-specific architectures toward universal diffusion-based frameworks, capable of bridging vision, language, and multi-modal data. As these models continue to evolve, they promise to redefine the landscape of both generative and discriminative modeling across domains.

1.2. Large Language Models and Vision-Language Models

The advent of Large Language Models (LLMs) has transformed the landscape of natural language processing by enabling systems to generate coherent, contextually aware, and semantically rich text across a wide range of tasks. Trained on massive corpora using transformer-based architectures, models such as GPT-3 [18], T5 [19], and PaLM [20] have demonstrated unprecedented capabilities in few-shot and zero-shot learning, powering applications in summarization, translation, question answering, and knowledge reasoning. These models rely on dense self-attention mechanisms and large-scale unsupervised pretraining to capture complex linguistic and semantic patterns, offering a foundation for general-purpose language understanding and generation.

Recent advancements have led to the development of Qwen series by Alibaba. Qwen2.5 [21], released in early 2024, demonstrates competitive performance across various tasks, offering strong multilingual capabilities. The Qwen2.5-VL [22] variant extends this foundation to multimodal tasks by integrating a visual encoder with the language backbone, enabling the model to interpret and reason over images and text in a unified manner. Qwen2.5-VL supports complex tasks such as image captioning, chart understanding, OCR-based reasoning, and multimodal dialogue, leveraging vision-language alignment techniques similar to those used in GPT-4V and Flamingo.

Building upon the Qwen2.5 family, Qwen3 [23], introduced in 2025, further enhances instruction-following, long-context reasoning, and multimodal comprehension. Qwen3 models incorporate architectural improvements and fine-tuning strategies that improve performance across reasoning-intensive tasks while preserving alignment with human intent. The Qwen3-VL variant, in particular, delivers improved accuracy in fine-grained visual tasks, demonstrating superior performance on standard multimodal benchmarks such as ScienceQA and MM-Bench.

Advances in the field have expanded the scope of LLMs beyond purely linguistic tasks, allowing them to interpret and reason over multimodal data when combined with visual backbone architectures. This evolution has led to the emergence of Vision-Language Models (VLMs), which combine powerful visual encoders (e.g., ViT [24] or CLIP [25]) with language models via cross-modal attention and joint training objectives. These models are capable of grounding textual concepts in visual inputs, making them effective for complex multimodal tasks such as image captioning, visual question answering (VQA), image-text retrieval, and multimodal dialogue.

Notably, CLIP [25] and ALIGN [26] introduced contrastive learning frameworks that align image and text embeddings in a shared latent space, allowing zero-shot transfer to a wide array of downstream tasks. Building on these foundations, models like BLIP [27], Flamingo [28], and MiniGPT-4 [29] have shown further improvements by refining vision-language alignment through autoregressive decoding and instruction tuning. GPT-4V, the multimodal extension of GPT-4, exemplifies the trend toward unified large-scale models that can seamlessly integrate image and text understanding in a single interface, with broad implications for both general AI research and domain-specific applications such as medical diagnostics, scientific analysis, and autonomous agents.

In the medical imaging domain, these models hold significant promise. Vision-Language models can be adapted to process radiology images, CT scans, and pathology slides alongside clinical notes or structured medical knowledge, enabling richer interpretations and more informative automated reporting. For example, recent works such as BioViL [30] and Med-PaLM [31] demonstrate the feasibility and effectiveness of adapting LLMs and VLMs to specialized healthcare tasks by incorporating domain-specific data and ontologies during pretraining.

As research progresses, the integration of LLMs and VLMs is expected to play a central role in building generalist agents that can operate across modalities, tasks, and domains. Their scalability, transferability, and adaptability position them as

foundational tools for the next generation of intelligent systems in both general AI and vertical applications such as biomedicine, robotics, and scientific discovery.

1.3. Agents and Multi-Agent Systems

The concept of AI agents has become increasingly central to the development of autonomous systems capable of perceiving their environment, reasoning over diverse modalities, and interacting with users or other agents to achieve specific goals. At its core, an agent is an entity that operates autonomously, continuously perceives inputs from its environment, maintains internal state, and executes actions to achieve objectives, often using a policy function $\pi(a|s)$ that maps states s to actions a.

Recent progress in foundation models has enabled the emergence of foundation agents or language model-driven agents, which leverage the general-purpose reasoning and instruction-following abilities of LLMs (e.g., GPT-4 [32], Claude [33], Qwen3 [23]) to perform complex sequential tasks. These agents integrate perception, memory, planning, and tool use within a modular framework, enabling capabilities such as web navigation, software automation, and multimodal task completion. Tool augmentation, such as using external APIs, retrieval systems, or structured code interpreters, allows agents to go beyond their static training knowledge to interact with dynamic environments.

The architecture of such agents typically follows a pipeline:

- Observation Module: Receives and interprets inputs (text, image, environment state).
- Planner: Constructs a task decomposition or action plan using chain-of-thought prompting or tree-based reasoning.
- Executor: Calls tools or APIs to perform subtasks, possibly with memory storage and retrieval.
- Feedback Loop: Updates the agent state or re-plans based on outcomes.

Building on single-agent capabilities, Multi-Agent Systems (MAS) extend this paradigm by enabling a network of agents to communicate, collaborate, or compete within a shared environment. MAS frameworks allow agents to exchange information, delegate tasks, negotiate strategies, and solve problems that exceed the capacity of individual agents. Applications span task decomposition, scientific discovery, simulation, and real-world robotics.

Key characteristics of MAS include:

- Decentralization: Each agent operates semi-independently, sharing information when necessary.

- Role Specialization: Agents may be specialized (e.g., planner, retriever, visual interpreter) to optimize performance across modalities.
- Communication Protocols: Agents use natural language or structured APIs to coordinate, facilitated by LLM-mediated dialogues.
- Collective Reasoning: Systems such as CAMEL [34], AutoGen [35], and MetaGPT [36] demonstrate how agents collaboratively generate code, solve tasks, or simulate human-like cooperation.

Recent frameworks like AutoGPT, CrewAI, AgentVerse, and OpenAgents offer robust infrastructure for multi-agent orchestration, enabling dynamic agent collaboration, tool sharing, and memory synchronization. These systems often support hierarchical planning, task routing, and long-term memory modules that improve adaptability in real-world environments.

In the biomedical domain, multi-agent systems are being explored for collaborative clinical decision-making, research automation, and medical image interpretation. Domain-specific agents, such as retrieval specialists, radiology interpreters, and language explainers, can coordinate to provide multi-faceted insights from medical records, imaging data, and literature, supporting more holistic and explainable AI systems.

In summary, agent and multi-agent paradigms represent a shift from static model inference to dynamic, interactive AI systems, capable of reasoning, coordination, and adaptation, key components in the path toward artificial general intelligence (AGI).

1.4. Image Classification Methods

1.4.1. Theoretical Background

Classification is a fundamental task in machine learning and computer vision, particularly in the medical imaging domain where accurate categorization of disease types or abnormalities can significantly impact clinical outcomes. Over the years, classification methods have undergone substantial evolution, transitioning from traditional machine learning techniques to deep learning and more recently to attention-based and diffusion-based models.

At the core of any classification pipeline lie two essential components: (1) Feature Extractor and (2) Classifier. These components together define the representational capacity and decision-making ability of the model. Below, we outline the mathematical foundations of both components.

Feature extraction refers to the process of transforming raw input data $x \in \mathbb{R}^n$ into a lower-dimensional and more informative representation $z \in \mathbb{R}^d$, where $d \ll n$.

Mathematically, this is achieved via a feature mapping function $\phi_\theta: \mathbb{R}^n \to \mathbb{R}^d$ with parameters θ, such that:

$$z = \phi_\theta(x)$$

Once features are extracted, the classifier maps the representation $z \in \mathbb{R}^d$ to a probability distribution over C classes:

$$\hat{y} = f_{cls}(z) \in \mathbb{R}^C$$

In most modern classification systems, the classifier is implemented as a fully connected layer followed by a softmax function:

$$\hat{y}_i = \frac{\exp\left(w_i^{\mathrm{T}} z + b_i\right)}{\sum_{j=1}^{C} \exp\left(w_j^{\mathrm{T}} z + b_j\right)}$$

where $\{w_i, b_i\}$ are parameters for class i.

The model is trained by minimizing a loss function such as the cross-entropy [37] between the predicted distribution $\hat{y}$ and the ground truth label y:

$$\mathcal{L}_{\mathrm{CE}} = -\sum_{y=1}^{C} y_i \log \hat{y}_i$$

1.4.2. Existing Image Classification Approaches

Data classification has evolved through several significant milestones, from traditional machine learning approaches such as Support Vector Machines (SVM) [38] and Random Forests [39] to more advanced methods. While kernel methods [40] enhanced non-linear data processing capabilities and ensemble learning [41] improved performance through model combination, these traditional approaches still faced limitations in handling complex data patterns.

The foundations of deep learning in image classification were established with LeNet [42], which pioneered CNNs for handwritten digit recognition. However, a transformative breakthrough came with AlexNet [43] which demonstrated the unprecedented potential of deep CNNs by winning the ImageNet Challenge, marking the beginning of the deep learning era. Following this success, advanced architectures like ResNet [44] solved the vanishing gradient problem through skip connections, while DenseNet [45] enhanced efficiency by proposing dense connections. Notably, EfficientNet [46] and its improved version EfficientNetV3 [47] have made significant strides in network architecture optimization through systematic scaling methods and self-adaptive hierarchical feature scaling.

Although CNN networks have proven effective, they still have limitations in capturing global relationships. To address this challenge, the attention mechanism introduced in Transformer [48] opened up an entirely new approach. Building on this

foundation, Vision Transformer [24] successfully adapted this architecture to computer vision, while Swin Transformer [49] and DiNAT [50] further improved it by proposing hierarchical attention mechanisms and dilated neighborhood attention, significantly enhancing computational efficiency.

Parallel to the development of attention models, another promising research direction emerged through diffusion models [1], [51], [52], [9], [10], [8], attracting particular attention for their ability to model complex data distributions. In this field, Dhariwal and Nichol [51] achieved a significant breakthrough by integrating U-Net architecture and attention into diffusion models. More notably, the application of diffusion models in classification tasks through D2C [8] and advanced model like SpectralDiff [9] and DiffMIC-v2 [10] has opened new prospects in learning discriminative features.

However, diffusion models's application to classification tasks faces several limitations. First, the potential of the diffusion process's denoising capabilities for improving classification robustness remains largely unexplored. Second, existing methods such as SpectralDiff [9], DiffMIC-v2 [10], lack effective mechanisms for multi-scale feature analysis, particularly in combining global and local information. Additionally, the challenge of maintaining consistent performance across different data types without major architectural changes persists [1], [52]. These limitations motivate our proposed GuidedDCNet, which leverages the stochastic diffusion process with multi-scale conditional guidance to enhance classification performance across various data types.

1.5. Image Captioning Methods

1.5.1. Theoretical Background

Image captioning is a complex task situated at the intersection of computer vision and natural language processing. The goal is to generate meaningful natural language descriptions for visual content, typically images. This process involves two core components: (1) Visual Feature Extractor and (2) Language Generator. The visual feature extractor encodes the image into a rich feature representation, while the language generator decodes this representation into a coherent textual sequence.

Mathematically, the image $I \in \mathbb{R}^{H \times W \times 3}$ is first processed by a feature extractor ϕ_θ to obtain an embedding $z = \phi_\theta(I)$. This embedding is then used to initialize or condition a sequential model that predicts a sequence of words $\hat{y} = (y_1, y_2, y_3, \dots, y_T)$, where each $y_t \in \mathbb{R}^{|\mathcal{V}|}$ is a probability distribution over a vocabulary $\mathcal{V}$.

The generation process typically maximizes the likelihood of the correct caption y given the image features z, through the minimization of the negative log-likelihood loss:

$$\mathcal{L}_{NLL} = -\sum_{t=1}^{T} \log P(y_t | y_{<t}, z)$$

Alternatively, reinforcement learning-based objectives such as Self-Critical Sequence Training [53] are employed to optimize task-specific metrics like CIDEr or BERTScore.

1.5.2. Existing Image Captioning Approaches

Early advancements in image captioning relied on CNNs for image feature extraction combined with Recurrent Neural Networks (RNNs), especially Long Short-Term Memory (LSTM) networks, for language modeling. A key milestone in this area was the CNN-LSTM framework, where the image features served as the initial hidden state or input for the LSTM decoder.

To better capture spatial and semantic relationships within images, Yao et al. [54] introduced a Graph Convolutional Network (GCN) to model object-object interactions. Their GCN-LSTM model integrates the structural information of object graphs into the caption generation process, enhancing the descriptive quality of generated sentences.

Building upon these foundations, Pan et al. [55] proposed the X-Linear Attention Network (X-LAN), which incorporates second-order bilinear pooling to model complex intra- and inter-modal interactions between image and text features. By embedding X-Linear attention blocks into both the encoder and decoder stages, the model achieves a richer joint representation for captioning.

Transformer-based models further revolutionized the field. Cornia et al. [56] developed the M² Transformer, a specialized architecture for image captioning. By introducing mesh-like connectivity in the decoder and multi-level attention over image regions, M² effectively fuses both local and global contextual information, significantly boosting captioning accuracy.

Recent challenges such as ImageCLEFmedical Caption [57] have focused on extending image captioning methods to the medical domain, where precision and contextual understanding are crucial. The task involves generating accurate captions across diverse medical imaging modalities, promoting automated clinical reporting and improved diagnostic support.

The winning team from CSIRO [58] adopted a streamlined encoder-decoder approach and optimized it using Self-Critical Sequence Training (SCST), specifically targeting the BERTScore metric to align closely with semantic correctness. Another prominent submission by the PCLmed team [59] proposed a multi-modal approach, Med-VLFM, combining general vision models with domain-specific medical models to yield more holistic and context-aware captions. This hybrid strategy led to top-tier performance across several evaluation metrics in the ImageCLEFmedical 2024 challenge.

Additionally, concept detection plays a pivotal role in medical image captioning. Ram et al. [60] conducted a comparative study using deep CNNs such as ResNet50, MobileNetV2, and DenseNet-121 for detecting medical concepts. Their findings indicated that ResNet50 offered superior accuracy, thereby providing reliable semantic inputs to downstream captioning systems.

These advances collectively reflect the evolution of image captioning from generic vision-language models to highly specialized medical applications, highlighting the growing importance of task-specific adaptation and multimodal integration.

1.6. Image Segmentation Methods

Image segmentation is a fundamental task in computer vision, especially critical in the medical imaging domain where precise localization and delineation of anatomical structures can significantly impact diagnostic accuracy and treatment planning. Recent advancements have focused on leveraging transformer and diffusion-based architectures to improve segmentation performance across diverse modalities and datasets.

Transformer-based architectures have recently redefined the state-of-the-art in medical image segmentation. These models capitalize on self-attention mechanisms to capture long-range dependencies and contextual information that traditional CNNs often overlook. A prominent early example is TransUNet [61], which incorporates a transformer encoder into the bottleneck of the UNet architecture. This hybrid model combines the spatial precision of convolutional encoders with the global contextual reasoning of transformers.

Building on this foundation, several transformer-augmented models have been proposed to further enhance segmentation performance. Swin-UNet [62] integrates the Swin Transformer backbone, which uses shifted windows to compute local self-attention efficiently while maintaining global context through hierarchical feature representations. Similarly, Swin-Unetr [63] extends this concept to volumetric medical

imaging by incorporating patch embeddings and positional encodings adapted for 3D data. The DS-TransUNet [64] further refines this design by employing dual-scale attention to better capture both fine-grained and global structural cues.

In parallel, diffusion models have emerged as a powerful alternative for image segmentation. By treating segmentation as a generative modeling task, these models utilize stochastic denoising processes to progressively refine segmentation masks. Diffusion-based methods inherently produce an ensemble of plausible segmentations through repeated sampling, which has been shown to improve robustness and accuracy [65]. However, this ensemble diversity must be carefully managed, uncontrolled variation can impede convergence and result in less reliable outputs. Thus, recent studies emphasize the development of refined sampling strategies and conditional denoising mechanisms to ensure that each iterative refinement step contributes positively to segmentation quality.

The convergence of transformer-based architectures with diffusion backbones represents a promising research direction. Combining the global reasoning ability of transformers with the iterative refinement capability of diffusion processes offers a unified framework for high-precision medical image segmentation, particularly in complex or low-contrast scenarios. As this hybrid modeling paradigm evolves, it is expected to play a pivotal role in the next generation of segmentation frameworks.

1.7. Vector Database

A vector database is a specialized type of database designed to store and search high-dimensional vectors. These vectors are typically generated by machine learning models from unstructured data such as text, images, or audio. Each vector encodes the semantic meaning of the input, enabling intelligent systems to retrieve similar content based on meaning rather than exact keyword matches [66]. Unlike traditional databases that support exact and structured querying, vector databases use approximate nearest neighbor (ANN) [67] algorithms to find items most similar to a query vector based on distance metrics like cosine similarity or Euclidean distance.

Modern vector databases offer a range of features tailored for AI-driven applications. These include high-performance indexing algorithms (such as HNSW [68] and IVF [69]), support for hybrid filtering (combining vector similarity with structured metadata), scalable storage and real-time updates, and compatibility with popular machine learning frameworks. These capabilities make vector databases essential for building systems like semantic search engines, recommendation systems, and medical question answering chatbots.

Several popular vector database systems are available, each with its own strengths and trade-offs. Qdrant [70] is an open-source vector database built for production-level semantic search. It supports fast similarity search using HNSW indexing, along with robust payload filtering and metadata storage. It offers REST and gRPC APIs and integrates well with Python-based AI workflows. Qdrant is especially suitable for domains like healthcare, where precise filtering and real-time performance are crucial.

Pinecone [71] is a fully managed vector database as a service. It abstracts infrastructure complexity, offering automatic indexing, low-latency search, and effortless scalability across millions or billions of vectors. Pinecone is ideal for teams that require a high-performance vector backend without the overhead of maintaining infrastructure. It supports features like namespaces, metadata filtering, and upserts, making it a popular choice for production AI applications.

Weaviate [72] is another open-source vector search engine that stands out for its built-in support for machine learning models. It can automatically embed and index data using models from OpenAI, Cohere, or Hugging Face, and supports hybrid keyword-vector search. Weaviate uses a GraphQL-like query interface and offers strong schema management, making it suitable for systems that blend unstructured and structured data, such as knowledge graphs.

FAISS [73], developed by Meta AI, is a library for efficient similarity search rather than a standalone database. It provides state-of-the-art indexing techniques and GPU acceleration, making it suitable for researchers or developers building custom vector retrieval pipelines. FAISS is widely used in academic and high-performance computing environments due to its flexibility and speed.

ChromaDB [74] is a lightweight, developer-friendly vector database tailored for rapid prototyping of LLM-powered applications. It supports embedding storage, metadata filtering, and fast querying. Chroma integrates seamlessly with tools like LangChain and is ideal for document question answering systems, retrieval-augmented generation, or AI agent development. It is particularly useful in small to medium-scale applications where simplicity and developer experience are key.

Each vector database system offers a different balance between performance, scalability, and ease of use. The choice of which to use depends on the specific requirements of the application, such as dataset size, update frequency, deployment environment, and the complexity of retrieval logic. In the context of medical AI applications, selecting the right vector database can significantly impact both retrieval accuracy and system responsiveness.

1.8. Chapter Summary

In this chapter, we have presented a comprehensive overview of the theoretical foundations and related technologies relevant to our project. Beginning with diffusion models, we explored their underlying principles and their growing role in generative modeling tasks. We then introduced large language models and vision-language models, highlighting their capacity to bridge textual and visual understanding.

The concept of agents and multi-agent systems was also discussed, providing a framework for distributed and interactive AI behavior. We examined image classification, captioning, and segmentation methods, three core areas in computer vision, by outlining both their theoretical foundations and current state-of-the-art approaches. These methods are crucial for enabling machines to understand and describe visual content. Finally, we introduced vector databases, which play an essential role in storing and retrieving high-dimensional representations efficiently, especially in tasks involving similarity search and large-scale data indexing.

The knowledge and methods presented in this chapter serve as the foundation for the implementation and experimentation stages in the subsequent chapters.

CHAPTER 2: METHODOLOGY

The MedCapSys, illustrated in Figure 2.1, is a highly detailed medical image captioning framework that integrates classification, segmentation, and large vision-language models to generate precise and informative descriptions of medical images. The system begins by processing an input brain medical image through multiple specialized neural networks. The GuidedDCNet module classifies the lession type, while the GuidedSegNet module performs segmentation to visualize the lession's location. The MedCapNet model generates an initial caption describing the scan, including modality and key observations. Subsequently, BrainMedQwen, a VLM, refines and enriches the description by extracting additional metadata, such as imaging view, pulse sequence, anatomical position, and tumor dimensions. These extracted details are then compiled into a comprehensive final description, which provides clinically relevant information, including tumor characteristics, localization, and signal intensity variations. This structured approach enhances interpretability and supports radiologists in medical diagnosis and documentation.

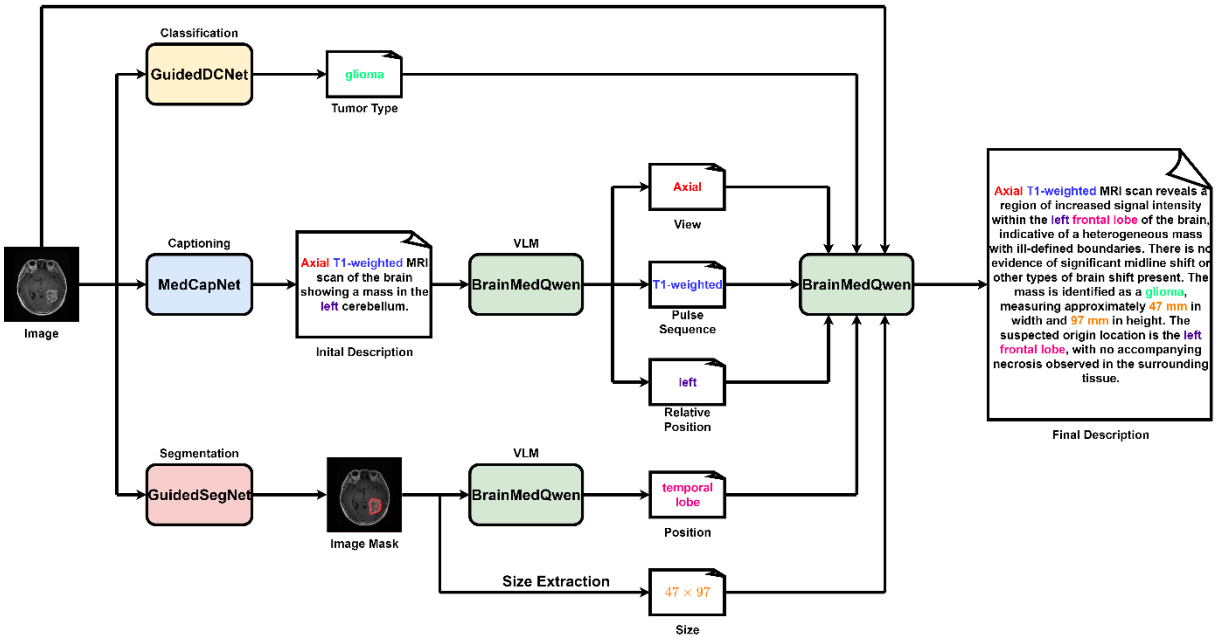


Figure 2.1 Medical Captioning System Overview

2.1. MedCapNet Architecture

The architecture of the proposed MedCapNet, shown in Figure 2.2, follows a typical encoder-decoder framework. The encoder is built upon a Swin Transformer backbone, complemented by *N* Enhancement Encoder blocks. On the other side, the

decoder comprises N decoder blocks. The role of the encoder is to extract and refine patch-level features from the input image, capturing intra-feature dependencies. The decoder then sequentially generates captions by utilizing these enhanced features, effectively bridging the relationship between visual and textual elements.

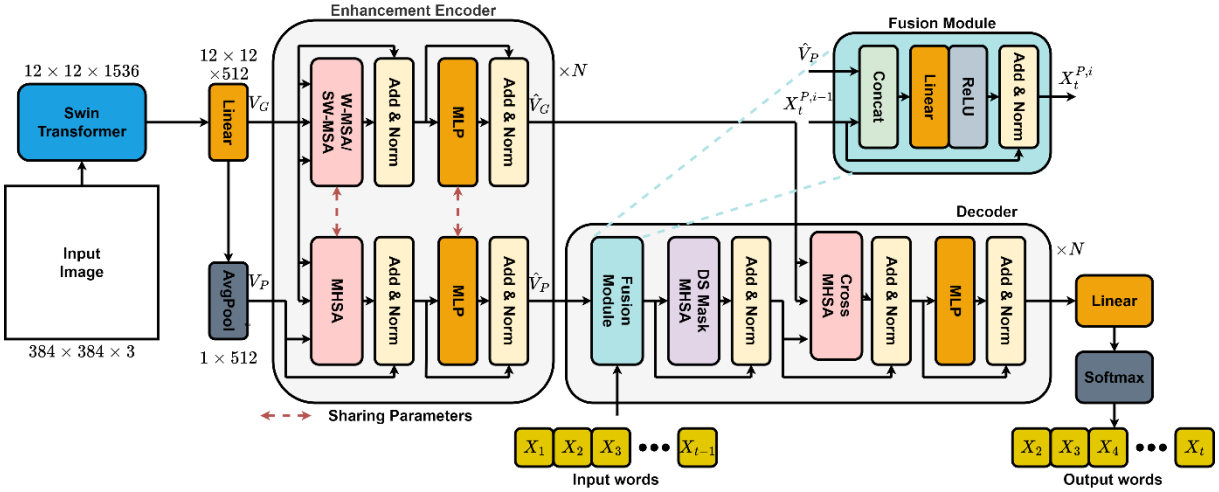


Figure 2.2 MedCapNet Overall Architecture

2.1.1. Encoder

A Swin Transformer [49] is initially employed as the backbone encoder, deviating from the customary use of pre-trained CNNs. This innovative approach allows for the extraction of a collection of patch-level features, denoted as $V_G = \{v_1, v_2, \ldots, v_m\}$, from the given input image I. Each patch-level feature, denoted as $v_i \in \mathbb{R}^D$, provides a foundational visual representation, where D is the dimensionality of the embedded space for each feature, and m represents the total number of patch-level features.

Following the extraction of patch-level features V_G, an initial global feature V_P is computed as the mean pooling of patch-level features, $V_P = \frac{1}{m}\sum_{i=1}^{m} v_i$ inspired by [75]. An Enhancement Encoder Block is employed to enhance these features by capturing their intra-dependencies. V_G is then integrated into both the W-MSA and SW-MSA modules, while V_P is fed into MHSA modules. During the application of MHSA within each window, this process is augmented by appending the patch-level features V_G as an additional token to the keys k and values v. This augmentation enhances the global feature V_P by incorporating information from all patch-level features, thereby improving the model's ability to capture both local and global contextual information.

As Figure 2.2 illustrates, the Enhancement Encoder complements N sequentially stacked blocks. Each block consists of two parallel branches, with each branch containing either a W-MSA/SW-MSA or an MHSA module, followed by a Multi-

Layer Perceptron (MLP). Notably, the self-attention layers and MLPs within these branches share identical parameters, ensuring consistency across the network. The W-MSA and SW-MSA modules alternate in the first branch throughout the sequence of blocks. The i^{th} block can be mathematically represented as follows:

$$V_G^{(i)} = \text{LayerNorm}\left(\hat{V}_G^{(i-1)} + \text{SW/W} - \text{MSA}\left(W_Q^l \hat{V}_G^{(i-1)}, W_K^l \hat{V}_G^{(i-1)}, W_V^l \hat{V}_G^{(i-1)}\right)\right)$$

$$V_P^{(i)} = \text{LayerNorm}\left(\hat{V}_P^{(i-1)} + \text{SW/W} - \text{MSA}\left(W_Q^l \hat{V}_P^{(i-1)}, W_K^l \hat{V}_G^{(i-1)}, W_V^l \hat{V}_G^{(i-1)}\right)\right)$$

$$\hat{V}_G^{(i)} = \text{LayerNorm}\left(\hat{V}_G^{(i)} + \text{MLP}\left(\hat{V}_G^{(i)}\right)\right)$$

$$\hat{V}_P^{(i)} = \text{LayerNorm}\left(\hat{V}_P^{(i)} + \text{MLP}\left(\hat{V}_P^{(i)}\right)\right)$$

where $\hat{V}_G^{(i-1)}$ and $\hat{V}_P^{(i-1)}$ represent the output patch-level features and global features from $(i-1)^{th}$ block, respectively. These features serve as the input to block i^{th}, with $\hat{V}_G^{(0)} = V_G$ and $\hat{V}_P^{(i)} = V_P$. The weight matrices are denoted by $W_Q^l, W_K^l, W_V^l \in \mathbb{R}^{D \times D}$. Two linear layers, interconnected by a ReLU activation function, form the structure of the MLP. This configuration can be mathematically expressed as follows:

$$\text{MLP}(X) = W_2 \text{ReLU}(W_1 X)$$

where the weight matrices are denoted by $W_1 \in \mathbb{R}^{2D \times D}$ and $W_2 \in \mathbb{R}^{D \times 2D}$. After the encoding process, the two outputs $\hat{V}_G = \hat{V}_G^{(N)}$ and $\hat{V}_P = \hat{V}_P^{(N)}$ are then fed into the decoder.

2.1.2. Decoder

The decoder architecture is designed to generate output captions sequentially, conditioning each word on the enhanced global and patch-level features extracted from the encoder, thereby enabling effective multi-modal information integration. Figure 2.2 shows that the decoder is comprised of a chain of N interconnected blocks. Each block is composed of four distinct modules, tailored to capture various aspects of intra- and inter-modal interactions:

- Fusion Module: This module initiates the inter-modal interaction between textual information and visual data.
- Dual-Scale Masked Multi-Head Self-Attention: This component facilitates intra-modal interaction within the generated words, enhancing linguistic coherence.
- Cross MHSA: This module, comprising an MSA layer followed by an MLP, represents the second inter-modal interaction between global and patch-level features.

2.1.2.1. Fusion Module

The absence of comprehensive global contextual information can significantly hinder the model's reasoning abilities. To address this limitation, a Fusion Module is introduced, which integrates the enhanced global feature $\hat{V}_P$. This module initiates the multi-modal interaction process, enabling the efficient capture and utilization of global visual context. The Fusion process of block i^{th} is mathematically described as follows:

$$X_t^{P,i} = \text{LayerNorm}\left(X_t^{P,i-1} + \text{ReLU}\left(W_F \text{Concatenate}\left(\hat{V}_P, X_t^{P,i-1}\right)\right)\right)$$

where $X_t^{P,i-1}$ presents the output from $(i-1)^{th}$ block and is used to generate the t^{th} word, $X_t^{P,0}$ is initialized through a Linear Layer with word embedding weight $W_{Embed} \in \mathbb{R}^{D\times|\mathcal{V}|}$ of the vocabulary $\mathcal{V}$:

$$X_t^{P,0} = W_{Embed} X_t$$

The fusion weight matrices for the Linear Layer are denoted by $W_F \in \mathbb{R}^{2D\times D}$. $X_t^{P,i}$ is then integrated into Dual-Scale Masked Multi-Head Self-Attention.

2.1.2.2. Dual-Scale Masked Multi-Head Self-Attention

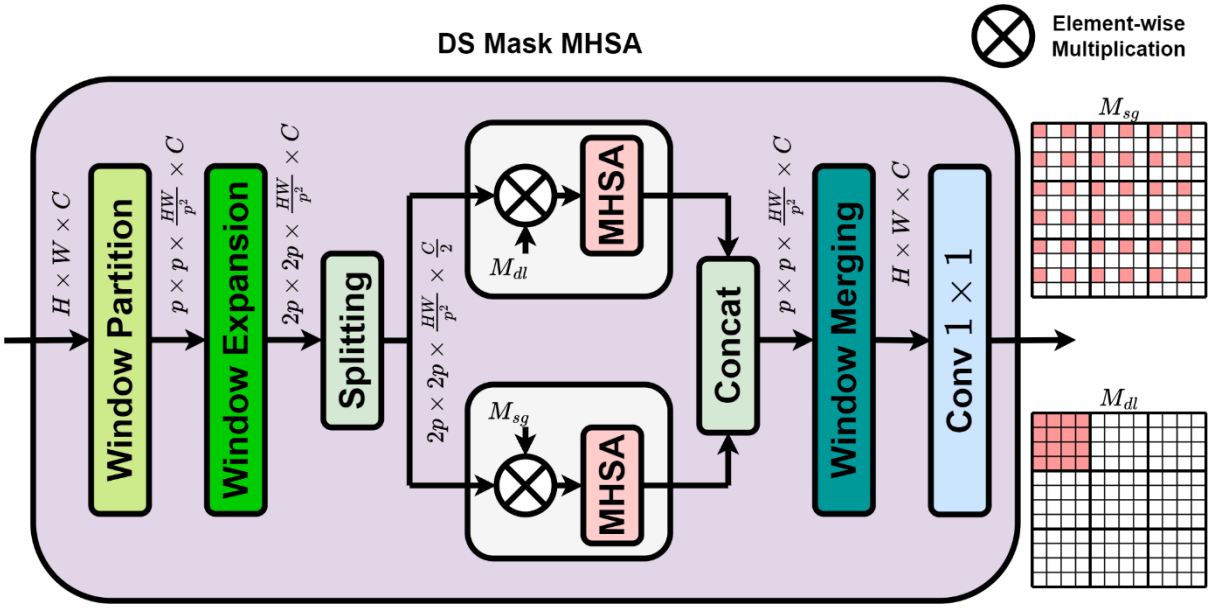


Figure 2.3 Dual-Scale Masked Multi-Head Self-Attention

As depicted in Figure 2.3, the Dual-Scale Masked Multi-Head Self-Attention (DS Mask MHSA) module is designed to enhance the model's ability to capture both local and global contextual information. Within this expanded window, local and global features are extracted using a sparse-global mask and a dense-local mask. Subsequently, self-attention is computed on both the local features generated by Dense Local Self Attention (DLSA) and the global features generated by Sparse Global Self

Attention (SGSA). The DS Mask MHSA module can be formulated with an input feature $I_{in} \in \mathbb{R}^{W \times H \times C}$ as:

$$I_{wp} = \text{Partition}(I_{in})$$
$$I_{we} = \text{Expansion}(I_{wp})$$
$$I_1, I_2 = \text{Split}(I_{we})$$
$$I_{sg} = \text{MHSA}\left(W_Q^{(SGSA)}(I_1 \otimes M_{sg}), W_K^{(SGSA)}(I_1 \otimes M_{sg}), W_V^{(SGSA)}(I_1 \otimes M_{sg})\right)$$
$$I_{dl} = \text{MHSA}\left(W_Q^{(DLSA)}(I_2 \otimes M_{dl}), W_K^{(DLSA)}(I_2 \otimes M_{dl}), W_V^{(DLSA)}(I_2 \otimes M_{dl})\right)$$
$$I_{out} = \text{Conv}\left(\text{Merging}\left(\text{Concat}(I_{sg}, I_{dl})\right)\right)$$

where $W_Q^{(SGSA)}, W_K^{(SGSA)}, W_V^{(SGSA)}, W_Q^{(DLSA)}, W_K^{(DLSA)}, W_V^{(DLSA)}$ are learnable weight matrices of SGSA and DLSA, respectively. $I_{wp} \in \mathbb{R}^{p \times p \times \frac{HW}{p^2} \times C}$ is the $p \times p$ token feature through a non-overlapped Window Partition on the input feature I_{in}. Window Expansion with an expansion size of 2 transforms the input I_{wp} into a larger token feature $I_{we} \in \mathbb{R}^{2p \times 2p \times \frac{HW}{p^2} \times C}$. $I_1, I_2 \in \mathbb{R}^{2p \times 2p \times \frac{HW}{p^2} \times \frac{C}{2}}$ are the results of the splitting process along the channel axis from I_{we}. The global feature I_{sg} and local feature I_{dl} are obtained by applying MHSA on the output of element-wised multiplication $\otimes$ on I_1 and I_2 with sparse-global and dense-local mask M_{sg} and M_{dl}, respectively. These masks comply with the Mask Sampling Rule presented in Algorithm 2.1 Mask Sampling Rule in Dual-Scale Masked Multi-Head Self-Attention. The final output feature $I_{out} \in \mathbb{R}^{H \times W \times C}$ is generated through a series of operations, including the channel-wise concatenation of I_{sg} and I_{dl}, followed by a Window Merging operation and a 1×1 Convolution Layer with $stride = 1$.

Algorithm 2.1 Mask Sampling Rule in Dual-Scale Masked Multi-Head Self-Attention

Algorithm 1 Mask Sampling Rule

Input:

Feature $I \in \mathbb{R}^{p \times p \times \frac{HW}{p^2} \times C}$ and Expanded Feature $I_{we} \in \mathbb{R}^{2p \times 2p \times \frac{HW}{p^2} \times C}$

Mask M with fixed size

Window size p

Output: Mask M and its drop rate α

1: **Choose** $I^{(i,j)} \in I, \forall i \in \left(1, \frac{H}{p}\right), j \in \left(1, \frac{W}{p}\right)$

2: **Choose** $I_{we}^{(i,j)} \in I_{we}, \forall i \in \left(1, \frac{H}{p}\right), j \in \left(1, \frac{W}{p}\right)$

3: $I_*^{(i,j)} = \bigcup_{h=i-1}^{i} \bigcup_{k=j-1}^{j} M \otimes I_{we}^{(h,k)}$

4: $\alpha = 1 - \frac{\left\| I_*^{(i,j)} \cap I^{(i,j)} \right\|}{\left\| I^{(i,j)} \right\|}$

5: **If** $\alpha = 0$, M satisfies Mask Sampling Rule

To optimize inference speed, learnable parameters in the mask have been replaced with fixed masks. Both binary masks M_{gs} and M_{dl} are designed to select a subset of feature points, reducing computational cost without sacrificing performance. M_{gs} plays the role of effectively expanding the receptive field, while M_{dl} ensures that important information in the neighborhood of each pixel is preserved.

After the Fusion process, $X_t^{P,i}$ of block i^{th} is then calculated through DS Mask MHSA as follows:

$$X_t^{P,i} = \text{LayerNorm}\left(X_t^{P,i} + \text{DSMaskMHSA}\left(X_t^{P,i}\right)\right)$$

2.1.2.3. Cross Multi-Head Self-Attention

The Cross Multi Head Self-Attention (Cross MHSA) can be considered as an advanced inter-modal correlation mechanism to capture local visual context information between $X_t^{P,i}$ and V_G. This interaction can be formulated as follows:

$$\tilde{X}_t^{P,i} = \text{LayerNorm}\left(X_t^{P,i} + \text{MHSA}\left(W_Q^{(i)} X_t^{P,i}, W_K^{(i)} V_G, W_V^{(i)} V_G\right)\right)$$

$$\hat{X}_t^{P,i} = \text{LayerNorm}\left(\tilde{X}_t^{P,i} + \text{MLP}\left(\tilde{X}_t^{P,i}\right)\right)$$

where $W_Q^{(i)}, W_K^{(i)}, W_V^{(i)} \in \mathbb{R}^{D \times D}$ are learnable weight matrices. $X_t^{P,i}$ obtained from the DS Mask MHSA module, is integrated into Cross MHSA as the query, while the enhanced patch-level features V_G from the final Enhancement Encoder block are fed into Cross MHSA as keys and values. The output of the Cross MHSA process, $\tilde{X}_t^{P,i}$, undergoes an MLP layer, and is added to $\tilde{X}_t^{P,i}$, followed by a Layer Norm layer to produce a feature $\hat{X}_t^{P,i}$.

2.1.3. Text Generation

The output $\hat{X}_t^{P,N-1}$ from the final decoder block has undergone the Text Generation module and generated words with conditional distribution over the vocabulary $\mathcal{V}$, defined as:

$$p(X_t | X_{1:t-1}) = \text{Softmax}\left(W \hat{X}_t^{P,N-1}\right)$$

where $W \in \mathbb{R}^{D \times |\mathcal{V}|}$ is learnable weight matrices.

2.2. GuidedDCNet Architecture

Figure 2.4 illustrates an overview of the proposed network architecture, GuidedDCNet, designed for classification. Given an input sample x, the data undergoes initial processing through an encoder, which extracts a feature embedding denoted as $\pi(x)$. Concurrently, a Multi-scale Conditional Guidance Mechanism (MCGM) generates both a global guidance vector $\hat{y}_g$ and a local guidance vector $\hat{y}_l$. During the training phase, a diffusion process is applied to the ground truth label y_0, incorporating the guidance vectors to yield three distinct noisy variables: $y_g^{(t)}$ (global guidance), $y_l^{(t)}$ (local guidance), and $y^{(t)}$ (dual guidance).

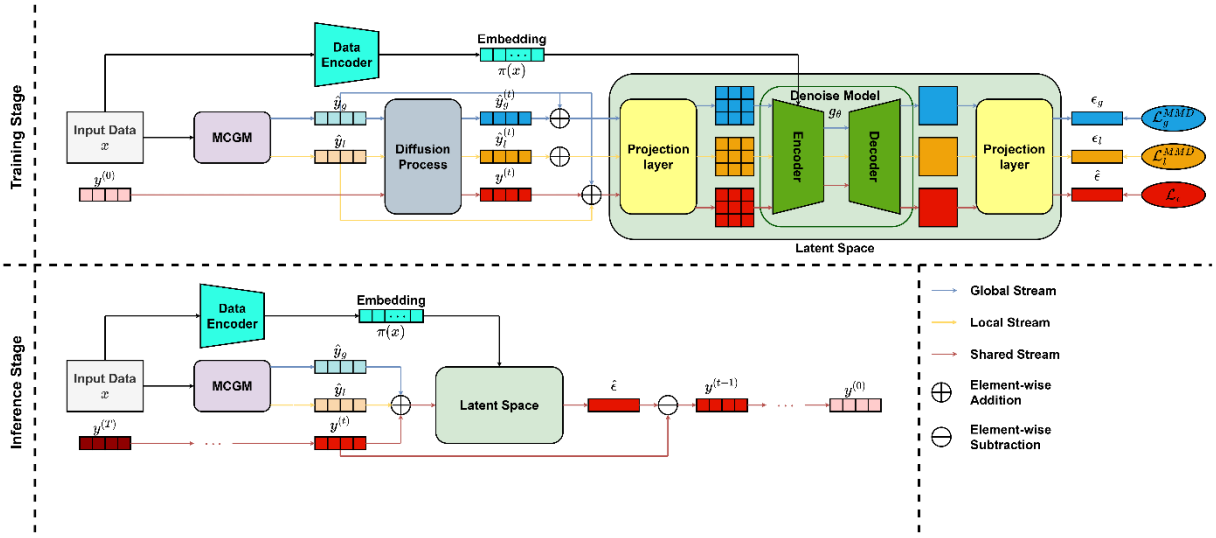


Figure 2.4 GuidedDCNet Overall Architecture

These noisy representations, along with their corresponding guidance vectors, are subsequently projected into a latent space. The projected embeddings are then integrated with the feature embedding $\pi(x)$ and processed through a denoising U-Net to estimate the noise distributions associated with $y_g^{(t)}$, $y_l^{(t)}$, and $y^{(t)}$. Furthermore, a maximum mean discrepancy (MMD) regularization adapted to specific guidances is introduced for the estimated noise associated with $y_g^{(t)}$ and $y_l^{(t)}$ to enhance the robustness of noise prediction. Simultaneously, a noise estimation loss based on the mean squared error (MSE) criterion is employed for the predicted noise of $y^{(t)}$. This comprehensive training strategy facilitates the collaborative optimization of the GuidedDCNet network, thereby improving its classification performance.

2.2.1. Multi-scale Conditional Guidance Mechanism

In conditional Denoising Diffusion Probabilistic Models, data classification remains a significant challenge due to the inherent ambiguity associated with object

representations within the dataset. The process of accurately identifying and distinguishing various attributes within the data is often complex, as objects may exhibit overlapping characteristics that hinder precise differentiation. This issue is further exacerbated by the presence of noise and artifacts within the most influential regions within the data, which can obstruct the extraction of meaningful high-level semantic features. Such noise may arise from multiple sources, including sensor inaccuracies, environmental variations, and intrinsic data inconsistencies, all of which contribute to the difficulty of learning discriminative representations necessary for robust classification.

A fundamental limitation of many existing DDPM-based classification approaches is their reliance on raw initial data as the sole conditioning factor in each diffusion step. While this data provides a foundational input for the learning process, it is often insufficient for capturing the fine-grained, high-resolution details required for effective classification. The absence of additional contextual information and feature refinement mechanisms impairs the model's ability to discern subtle variations, particularly among visually or semantically similar categories. Consequently, this constraint leads to suboptimal classification performance, as the model is unable to fully exploit the available data to generate precise predictions. To address these challenges, it is imperative to develop enhanced conditioning strategies that incorporate richer contextual features, mitigate the influence of noise, and facilitate the extraction of more discriminative representations throughout the diffusion process.

To address the aforementioned challenge, a Multi-scale Conditional Guidance Mechanism (MCGM) is proposed to refine the encoding process at each step of the diffusion process. Specifically, the MCGM model f_{MCGM} is introduced to compute guidance vectors at both global and local levels, thereby facilitating more effective information propagation and improving the robustness of the diffusion framework.

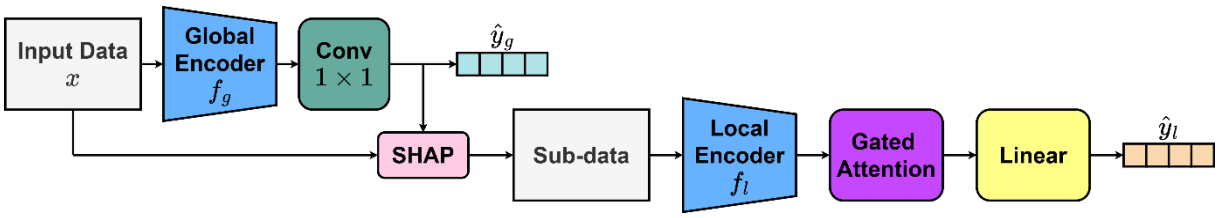


Figure 2.5 Multi-scale Conditional Guidance Mechanism

A holistic representation of the data distribution is provided by the global guidance vector, enabling a comprehensive contextual understanding. In contrast, the local guidance vector allows selective focus on the most influential regions within the

data, where critical information is contained for mitigating the impact of noise. As depicted in Figure 2.5, this mechanism is implemented through a dual-stream architecture.

To begin with, the input data x is embedded by the global encoder f_g, followed by a 1×1 convolutional layer, through which an attention map representing the overall distribution of prominent features is generated. The global guidance vector $\hat{y}_g$ is subsequently obtained by computing the average response across this saliency map, ensuring a stable global guidance vector for diffusion.

In the local stream, Shapley Additive Explanations (SHAP) [76] is employed to identify and extract the most influential regions within the data by quantifying the contribution of individual features to the model's predictions. Based on these identified key features, a refined subset of the data is subsequently constructed to facilitate further analysis. The sub-data is then processed by the local encoder f_l, and distinct feature representations are generated. To effectively aggregate these features, a gated attention mechanism [77] is employed, which adaptively assigns importance weights to different sub-data features. The resulting weighted feature representation is then passed through a linear layer, where the local guidance vector $\hat{y}_l$ is computed, thereby enhancing the model's capacity to capture fine-grained details while effectively suppressing noise.

2.2.2. Diffusion Model

Our proposed diffusion model adopts a two-stage framework analogous to Denoising Diffusion Probabilistic Models (DDPM) [1], consisting of a diffusion process and a denoising process.

During the forward diffusion stage, the target variable $y^{(0)}$ is incrementally perturbed by the addition of Gaussian noise at each time step. These time steps are sampled from a uniform distribution over the interval $[1, T]$, generating a sequence of progressively noised variables denoted as $\{y^{(1)}, \ldots, y^{(t)}, \ldots y^{(T)}\}$. This process serves to construct a latent representation that facilitates effective noise modeling.

Following the training phase, the reverse diffusion process is executed to recover the original response variable. To parameterize this process, we use a conditional UNet architecture as the denoising network, following the standard DDPM implementation, as illustrated in Figure 2.6. The trained UNet g_θ with the set of trainable parameters θ is tasked with generating the final prediction $\hat{y}^{(0)}$ by transforming the distribution of the noisy variable $p_\theta(y^{(T)})$ back to the original data distribution $p_\theta(y^{(0)})$.

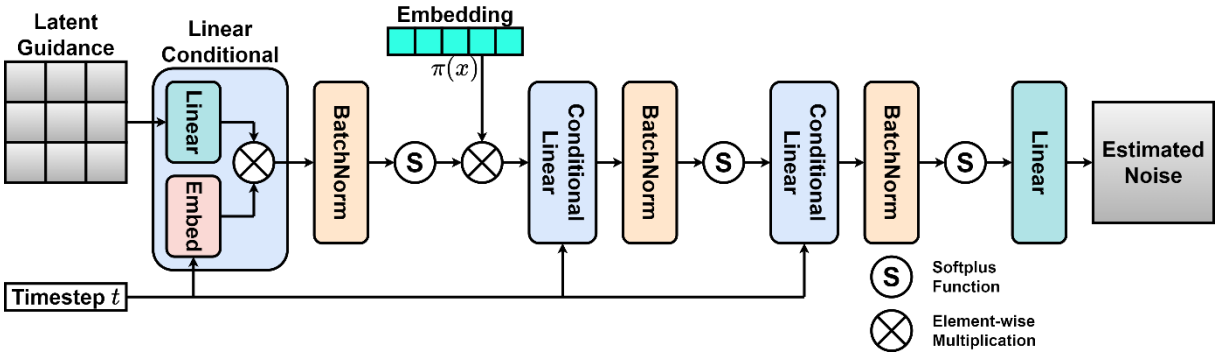


Figure 2.6 Conditional UNet Architecture

With the initialization of the noisy variable follows the Gaussian distribution $p_\theta\left(y^{(T)}\right) = \mathcal{N}\left(\frac{\hat{y}_g + \hat{y}_l}{2}, I\right)$, the reverse diffusion process is formulated as:

$$p_\theta\left(y^{(0:T-1)}|y^{(T)}, \pi(x)\right) = \prod_{t=1}^{T} p_\theta\left(y^{(t-1)}|y^{(t)}, \pi(x)\right)$$

The noisy variable $y^{(t)}$ is sampled based on the guidance vectors generated by the MCGM, using the following equation:

$$y^{(t)} = \sqrt{\alpha_t} y^{(0)} + \sqrt{1 - \bar{\alpha}_t}\epsilon + \left(1 - \sqrt{\bar{\alpha}_t}\right)\left(\hat{y}_g + \hat{y}_l\right)$$

where $\epsilon \sim \mathcal{N}(0, I)$, $\bar{\alpha}_t = \prod_t \alpha_t$, and $\alpha_t = 1 - \beta_t$ with a linear noise schedule $\{\beta_t | \beta_t \in (0,1)\}_{t=1:T}$.

Following the sampling process, the noisy variable $y^{(t)}$ is further refined through the incorporation of dual guidance vectors $\hat{y}_g$, $\hat{y}_l$ before being input into the denoising model UNet g_θ. The primary objective of UNet is to estimate the noise distribution, which is formally expressed as:

$$g_\theta\left(\pi(x), y^{(t)}, \hat{y}_g, \hat{y}_l, t\right) = \mathrm{Dec}\left(\mathrm{Enc}\left(\mathrm{Proj}\left(\mathrm{Concat}\left(y^{(t)}, \hat{y}_g, \hat{y}_l\right)\right), \pi(x), t\right), t\right)$$

where $\mathrm{Proj}(\cdot)$ denotes the projection layer mapping data to the latent space, $\mathrm{Enc}(\cdot)$ and $\mathrm{Dec}(\cdot)$ correspond to the encoder and decoder components of the UNet architecture, respectively.

To enhance the model's ability to capture semantic information, UNet not only processes the noisy variable $y^{(t)}$ but also integrates the feature embedding $\pi(x)$ with the projected noisy embedding. This integration enables the model to focus on high-level semantic features, thereby facilitating the generation of more robust and informative representations. With the architecture described in Figure 2.6, the integration of the timestep into the guidance vectors is introduced by initially performing an element-wise multiplication between the fused vector and the timestep embedding. Subsequently, the integration of the data feature embedding and the

response embedding is performed through an additional element-wise multiplication to ensure seamless information fusion. Following this processing step, the output vector is propagated through a pair of successive linear layers, each incorporating another element-wise multiplication with the timestep embedding to maintain temporal coherence. A final linear layer is employed for noise estimation, with an output dimensionality corresponding to the number of classes. It is important to emphasize that all fully connected layers in the model are equipped with a batch normalization layer and a Softplus activation function, except for the output layer, to enhance training stability and ensure smooth gradient flow.

The proposed approach enhances the conventional diffusion model by conditioning each step of the estimation process on guidance vectors that integrate information from both the original data and the most influential regions within the data. This methodological refinement enables the model to more effectively leverage semantic information, thereby improving the accuracy and stability in the generated outputs.

2.2.3. Loss Function

Maximum-Mean Discrepancy (MMD) quantifies the difference between two distributions by analyzing discrepancies in their statistical properties [78], [79], efficiently computed using a positive definite kernel $\mathbb{K}(\cdot,\cdot)$ that reproduces distributions in Hilbert space. Inspired by these studies [80], [81], [82], a combination of MMD regularization functions is introduced to enhance the learning of mutual information between the noise-generating distribution and the reference normal distribution.

In our approach, the noisy variable $y_g^{(t)}$ is sampled from the diffusion stage at timestep t, informed by the global guidance vector. The MMD loss is defined as:

$$\mathcal{L}_g^{MMD}(\epsilon||m) = \mathbb{K}(\epsilon,\epsilon') - 2\mathbb{K}\big(\epsilon_g,\epsilon\big) + \mathbb{K}\big(\epsilon_g,{\epsilon_g}'\big)$$

where $\epsilon_g = g_\theta\pi\big((x),\sqrt{\alpha_t}y^{(0)} + \sqrt{1-\bar{\alpha}_t}\epsilon + \big(1-\sqrt{\bar{\alpha}_t}\big)\hat{y}_g, t\big)$. This MMD regularization is also employed in the local guidance vector to compute $\mathcal{L}_l^{MMD}$, as illustrated in Figure 2.4.

During the diffusion process, model optimization is performed by minimizing the noise estimation loss $\mathcal{L}_\epsilon$, defined as follows:

$$\mathcal{L}_\epsilon = \big\|\epsilon - g_\theta\big(\pi(x), y^{(t)}, \hat{y}_g, \hat{y}_l, t\big)\big\|^2$$

Leveraging the generalized noise prediction capability of the loss function $\mathcal{L}_\epsilon$ and the MMD regularizer effectively improves feature learning, accelerates model convergence, and enhances stability. The total loss function $\mathcal{L}_{total}$ for

GuidedDCNet is formulated by integrating the noise estimation loss with the MMD regularization terms, expressed as follows:

$$\mathcal{L}_{total} = \mathcal{L}_{\epsilon} + \lambda\left(\mathcal{L}_{g}^{MMD} + \mathcal{L}_{l}^{MMD}\right)$$

where λ serves as a hyperparameter that regulates the influence of MMD regularization on the optimization process, ensuring a balanced trade-off between noise estimation accuracy and guidance vector consistency.

2.3. GuidedSegNet Architecture

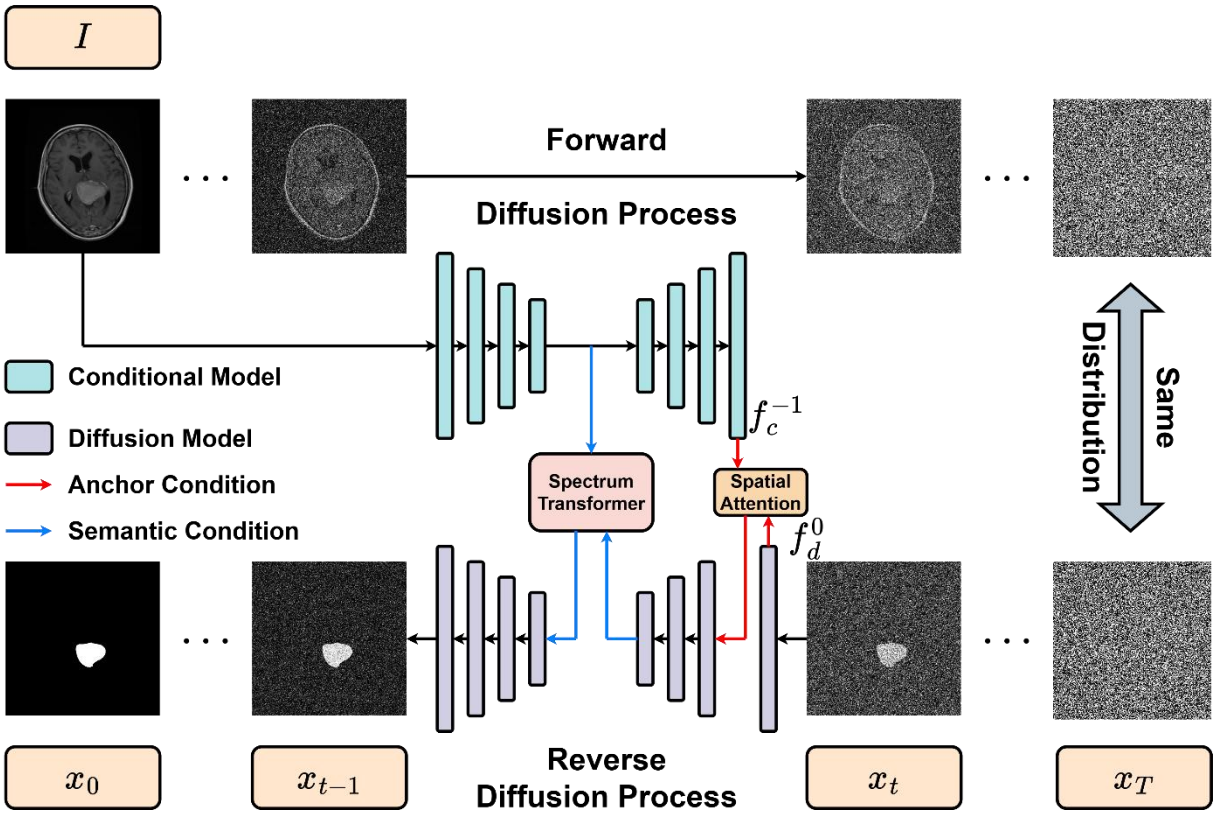


Figure 2.7 GuidedSegNet Overall Architecture

The complete pipeline of GuidedSegNet is depicted in Figure 2.7. At each time step t, the noisy segmentation mask x_t is refined using a UNet-based diffusion model. This model operates under the guidance of segmentation features extracted from raw input images by a distinct UNet, referred to as the Condition Model. The conditioning is integrated through two distinct mechanisms:

- Anchor Condition, which integrates decoded segmentation features from the Condition Model into the Diffusion Model's encoder to provide a stable reference and reduce variance.
- Semantic Condition, which incorporates semantic segmentation embeddings into the Diffusion Model's embedding space. This integration is facilitated by

Spectrum Transformer, leveraging transformer-based global and dynamic representations to bridge the gap between noise and semantic embeddings.

2.3.1. Anchor Condition

Anchor Condition integrates rough anchor features from the Condition Model into the Diffusion Model, guiding predictions within a reasonable range while allowing refinement. Specifically, decoded segmentation features from the Condition Model are fused with the Diffusion Model's encoder features, which account for uncertainty in the conditional features. Given the last conditional feature f_c^{-1} and the first diffusion feature f_d^0, Spatial Attention [83] applies a learnable Gaussian kernel k_G to smooth f_c^{-1}, followed by a max operation to retain the most relevant information:

$$f_{anc} = \max(f_c^{-1} * k_G, f_c^{-1})$$
$$f'^0_d = \sigma(f_{anc} * k_{1\times1}) \otimes f_d^0 + f_d$$

where $*$ denotes kernel manipulation, $\otimes$ represents element-wise multiplication, and σ is the Sigmoid activation. A 1×1 convolution reduces channel dimensions before multiplication, enhancing f_d^0 akin to spatial attention mechanisms.

2.3.2. Semantic Condition

The Semantic Condition integrates semantic segmentation embeddings into the embedding space of the Diffusion Model via the Spectrum Transformer, a module designed to capture and model the interactions between conditional semantic features and diffusion noise within the frequency domain. The architecture of the Spectrum Transformer module is presented in Figure 2.8.

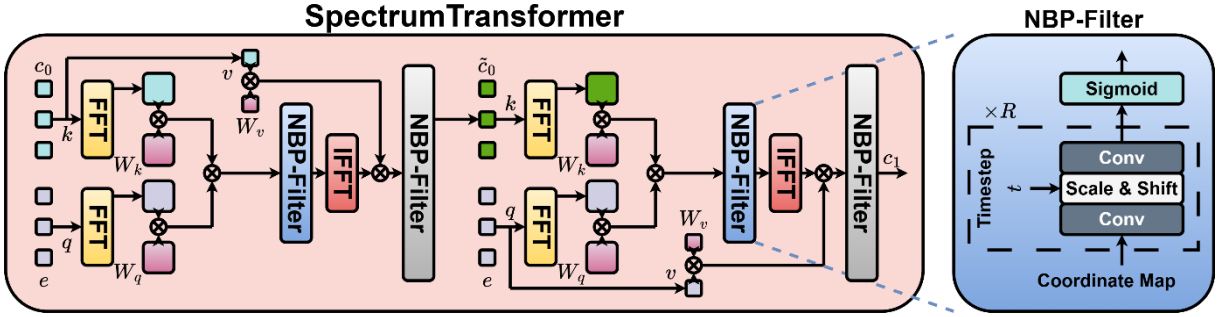


Figure 2.8 Spectrum Transformer Architecture

A key component of Spectrum Transformer is the Neural Bandpass Filter (NBP-Filter) [84], which learns to align these features to a unified frequency range. It adaptively selects relevant spectral components based on diffusion time steps, leveraging a learned projection from spatial coordinates to frequency magnitudes. Inspired by Neural Radiance Fields (NeRF) [85], we condition this projection on timestep embeddings using a stack of convolutional blocks with a layer normalization.

Spectrum Transformer consists of $N = 4$ blocks, each containing two crossattention-like modules. First, embeddings from the Condition Model and Diffusion Model are transformed into the Fourier domain via Fourier transformation F. Let c_0 denotes the deepest feature embedding of the Condition Model and e represents the corresponding embedding of the Diffusion Model. An affinity map is computed using:

$$M = \big(F(c_0)W_q\big)(F(e)W_k)^T$$

where W_q and W_k are learnable weights. The NBP-Filter applies a learned attention map to refine M, which is then transformed back to Euclidean space via inverse Fourier transform F^{-1}:

$$f = F^{-1}(M')(c_0W_v)$$

where W_v is the learnable value weight. The second attention module symmetrically maps segmentation features into the noise domain, generating the conditioned embedding for the next block. This process enhances feature interaction, reducing the domain gap and improving segmentation accuracy.

2.3.3. Forward and Reverse Diffusion Process

GuidedSegNet is designed based on the diffusion model framework [1], including two stages: forward diffusion and reverse denoising. During the forward process, Gaussian noise is incrementally added to the original segmentation label x_0 over a series of T steps, progressively corrupting the data. The reverse process involves training a neural network to learn the denoising trajectory, effectively reconstructing the original data by inverting the noise addition process. This procedure can be formally expressed as follows:

$$p_\theta(x_{0:T-1}|x_T) = \prod_{t=1}^{T} p_\theta(x_{t-1}, x_t)$$

In this formulation, θ represents the parameters of the reverse process. The generative procedure is initiated by sampling from a Gaussian noise distribution, denoted as $p_\theta(x_T) = \mathcal{N}(x_T; 0, I_{n\times n})$, where I corresponds to the identity matrix and x_T represents the initial noisy latent variable. The reverse process then iteratively refines this latent representation, progressively transforming $p_\theta(x_T)$ toward the target data distribution $p_\theta(x_0)$. To maintain consistency with the forward diffusion process, the reverse process reconstructs intermediate noisy representations at each step, ultimately producing a high-fidelity segmentation output.

To enable effective segmentation, the noise estimation function ϵ is conditioned on prior information extracted from the raw input image. This conditioning mechanism can be formally represented as:

$$\epsilon_\theta(x_t, I, t) = \text{Dec}(\text{Trans}(E_t^I, E_t^x t), t)$$

In this formulation, Trans refers to the transformer-based attention mechanism. The variable E_t^I denotes the conditional feature embedding derived from the raw image, while E_t^x represents the feature embedding corresponding to the segmentation map at the current time step t. These embeddings are jointly processed by the transformer module and subsequently passed through a UNet-based decoder, denoted as Dec, to enable reconstruction. The time step t is incorporated into both the fused embedding and the decoder features, with each time index encoded via a shared, learnable lookup table, in accordance with the methodology proposed in [1].

2.4. BrainMedQwen

BrainMedQwen is a specialized vision-language model (VLM) fine-tuned from Qwen2.5-VL-72B [21] to enhance its capability in medical image understanding and captioning. Built upon the robust multimodal foundation of Qwen2.5-VL 72B, presented in Figure 2.9, BrainMedQwen has been adapted for domain-specific tasks, including the interpretation of brain imaging data, extraction of clinically relevant attributes,
and generation of detailed radiological descriptions. Fine-tuning involves training on a curated dataset of annotated medical images, with a particular emphasis on MRI scans, ensuring the model effectively recognizes anatomical structures, imaging modalities, pathological findings, and quantitative measurements. This adaptation enables BrainMedQwen to accurately identify key imaging parameters such as view orientation, pulse sequences, and lesion characteristics. By leveraging large-scale pretraining and targeted fine-tuning, BrainMedQwen bridges the gap between general-purpose vision-language reasoning and the specialized requirements of medical imaging, thereby improving the automation and interpretability of radiological assessments.

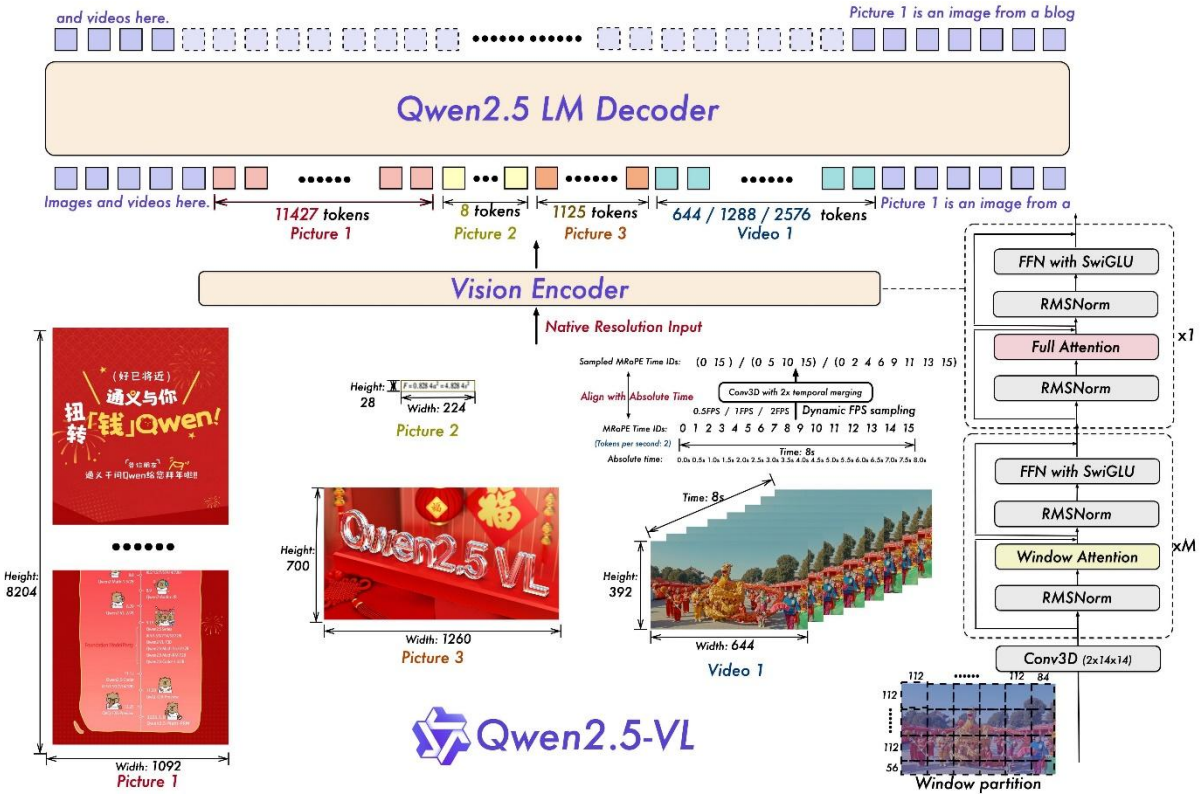


Figure 2.9 Qwen 2.5-VL Architecture [21]

2.5. A Multi-Agent Framework for Medical Question Answering Chatbot

As the demand for intelligent and accessible healthcare support systems grows, medical chatbots have emerged as a promising solution for assisting patients and professionals with timely, relevant, and accurate information. To enhance the reliability and domain specialization of such systems, this chapter introduces a Multi-Agent Medical Question Answering Framework, designed to manage complex medical queries, both in text and image formats, through collaborative interactions among specialized agents.

The proposed framework, illustrated in Figure 2.10, incorporates a multi-agent architecture in which domain-specific agents coordinate to interpret and respond to user queries. Users can submit questions in textual or image formats, which are then processed and routed through a network of intelligent agents with access to medical memory and knowledge sources providing contextual history and integrates authoritative medical knowledge bases such as MedlinePlus, Mayo Clinic, WHO Guidelines, NICE Guidelines, OpenFDA, and more.

The framework consists of the following key components:

- Coordinator Agent: Serves as the central decision-maker, delegating tasks to appropriate domain-specific agents based on the query content. This team

leader parses user intent and identifies whether the query requires general medical knowledge, radiological expertise, or both.

- Radiologist Agent: Specializes in interpreting medical images and radiological data, leveraging advanced analysis tools to execute tasks such as lesion detection, tumor classification, segmentation, and report generation.
- General Practitioner Agent: Handles general medical knowledge queries, symptom checking, and basic diagnostic using structured knowledge sources and prior interaction history stored in the system's memory.

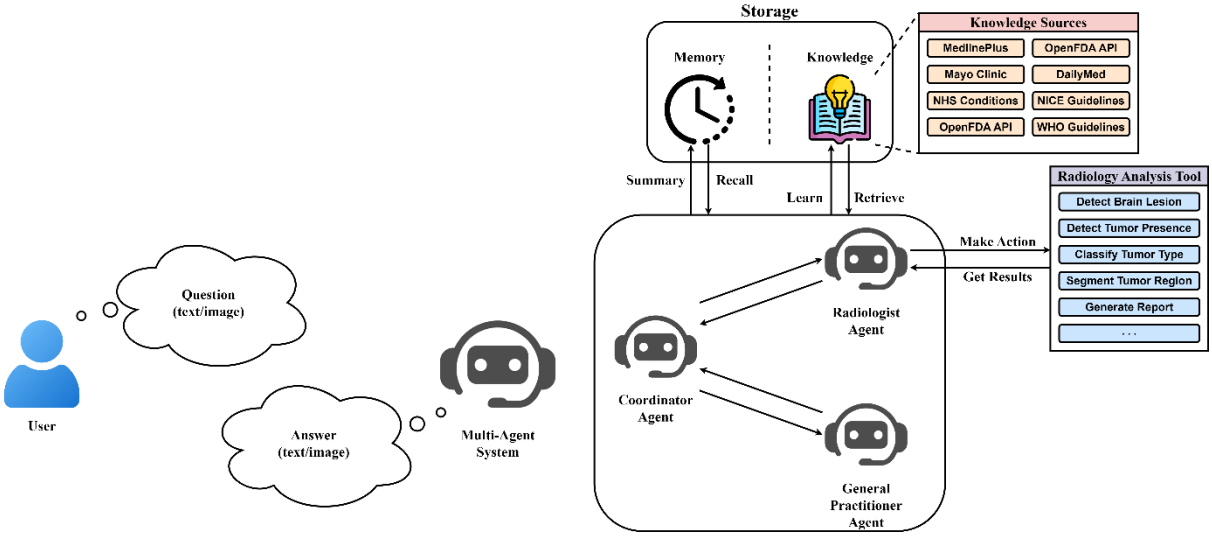


Figure 2.10 Multi-Agent Framework for Medical Question Answering

To support image-based question answering, the Radiologist Agent integrates with MedCapSys, a medical image captioning and analysis pipeline specifically optimized for brain imaging.

The chatbot system maintains two distinct yet interconnected resources:

- Memory: Tracks interaction history, patient-specific context, and follow-up information.
- Knowledge Base: Continuously updated from verified medical APIs and guidelines (e.g., OpenFDA, NHS Conditions, DailyMed).

These sources empower the agents to provide responses that are not only context-aware but also medically validated.

2.6. Chapter Summary

This chapter presented the detailed methodologies and system architectures employed in our project. We first introduced MedCapNet, a specialized architecture for medical image captioning, breaking down its components: the encoder, decoder,

and text generation module. Each part plays a crucial role in understanding visual features and translating them into clinically meaningful descriptions.

Next, we described GuidedDCNet, a diffusion-based generative model enhanced by a multi-scale conditional guidance mechanism. The components, including the diffusion process and loss function, were elaborated to demonstrate how the model leverages both data-driven priors and structured conditions to produce high-quality medical images.

We also outlined the GuidedSegNet architecture, which focuses on medical image segmentation. By incorporating both anchor and semantic conditions, along with a bidirectional diffusion process, the model aims to improve segmentation accuracy through guided generation.

In addition, we introduced BrainMedQwen, a model designed to leverage vision-language capabilities in a medical context, and concluded with a multi-agent system framework for a medical question answering chatbot. This framework integrates various specialized agents to provide coherent and accurate responses to medical inquiries.

Overall, the methodologies discussed in this chapter form the backbone of our proposed solutions, setting the stage for the experimental validation and evaluation presented in the following chapters.

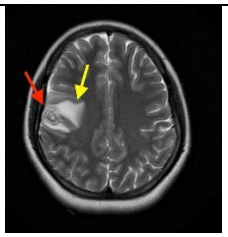
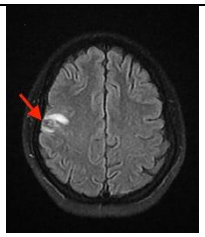
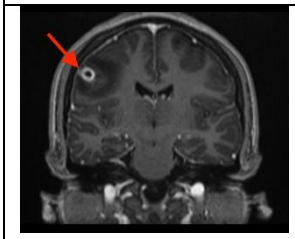
CHAPTER 3: IMPLEMENTATION AND EVALUATION

3.1. Datasets

3.1.1. Image Captioning Dataset

The task utilizes an enhanced iteration of the Radiology Objects in COntext version 2 (ROCOv2) dataset [86], sourced from figures in open-access biomedical journal articles published on PubMed Central. This dataset is particularly suited for caption prediction tasks, as each image is paired with a corresponding caption. To ensure data quality and consistency, all captions underwent preprocessing to remove extraneous elements such as hyperlinks. The dataset is organized into three subsets: (1) the training set comprises 70,108 images with their corresponding captions, (2) the validation set contains 9,972 images and associated captions, and (3) the test set includes 17,237 images along with their captions. During training, all training captions are converted to lowercase. Several exmaples are presented in Table 3.1.

Table 3.1 Example Samples from ROCOv2 Dataset

Image	Caption
	Axial view MRI brain showing solitary cystic mass (red arrow), with surrounding vasogenic edema (yellow arrow).
	Axial brain MRI showing decrease in size of cystic mass (red arrow).
	Coronal view MRI brain showing cystic mass with thickened peripheral enhancement (red arrow).

3.1.2. Image Classification Dataset

The lesion classification dataset used in this study is a curated combination of three publicly available datasets: Figshare [87], SARTAJ [88], and Br35H [89]. It comprises a total of 10,287 human brain MRI images, categorized into four distinct

classes: “glioma”, “meningioma”, “pituitary”, and “no_tumor”. The "no_tumor" class images were exclusively sourced from the Br35H dataset. However, during dataset refinement, it was observed that the glioma class images in the SARTAJ dataset were not correctly categorized, as evidenced by inconsistencies in prior studies and the performance of multiple trained models. To address this issue, the glioma images from the SARTAJ dataset were excluded, and images from the Figshare dataset were used instead. This preprocessing step ensures the integrity and reliability of the dataset, facilitating more accurate tumor classification and improving model performance in identifying different brain tumor types from MRI scans. Figure 3.1 presents the number of images per class, demonstrating a balanced distribution across the different categories.

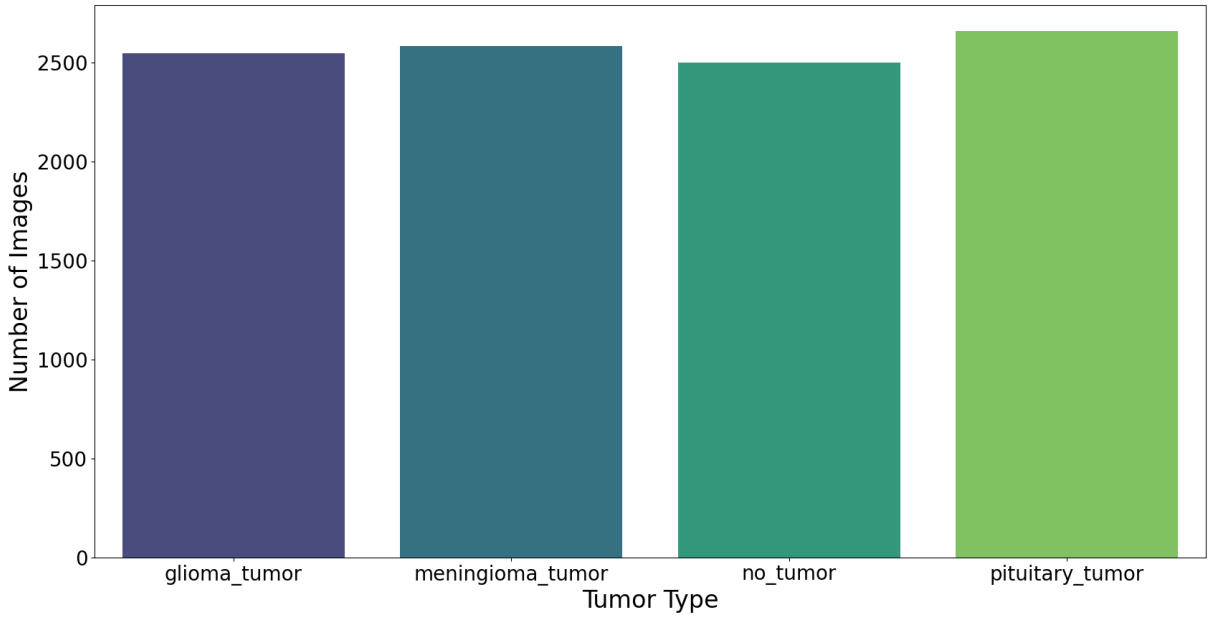


Figure 3.1 Number of Images per Class in Brain Tumor Classification Dataset

3.1.3. Image Segmentation Dataset

This study evaluates the segmentation task using a widely recognized medical image dataset: BraTS2020 [90]. The BraTS2020 dataset is an established benchmark for brain tumor segmentation in MRI and comprises multi-modal MRI scans, including T1-weighted, T1-weighted contrast-enhanced, T2-weighted, and Fluid Attenuated Inversion Recovery (FLAIR) sequences, as outlined in Table 3.2.

Furthermore, a comprehensive synthesized dataset was constructed by aggregating 2,306 abnormal brain MRI images sourced from the ROCOv2 [86] dataset, along with an additional 7,787 images depicting pathological conditions relevant to brain tumor classification tasks. To facilitate precise annotation and enhance the quality of the dataset, segmentation masks for these images were

meticulously generated through a manual annotation process utilizing the SAM2 framework [91]. This approach ensures high accuracy in the delineation of abnormal regions, thereby improving the reliability and applicability of the dataset for subsequent machine learning and deep learning applications in brain tumor detection and classification.

Table 3.2 Statistical Analysis of the BraTS2020

Attributes	BRATS2020
Number of contrasts	4 (T1, T2, T1CE, FLAIR)
Number of samples per contrast	494 images/contrast
Image size	$240 \times 240 \times 155$
Ratio healthy/disease	40/60

3.1.4. Multimodal Dataset

To fine-tune the Vision-Language Model for medical applications, a diverse and high-quality dataset of 109,191 image-text pairs was curated through data crawling techniques. The dataset collection process focused on acquiring comprehensive textual and visual resources related to the anatomy of the brain across multiple languages. Sources included medical textbooks, peer-reviewed journal articles, scientific blogs, and reputable online repositories containing annotated medical diagrams and illustrations. By aggregating multilingual data, the model is enhanced with cross-linguistic understanding, enabling it to interpret and generate medical descriptions in various languages while maintaining domain-specific accuracy.

3.2. Preprocessing

3.2.1. Image Classification Dataset

Accurate tumor classification is often challenged by poor visual quality, noise, and low contrast in MRI scans. To address these limitations, the authors implemented the enhanced preprocessing technique, as illustrated in Figure 3.2, to enhance image quality, reduce noise, and improve contrast while preserving critical anatomical information.

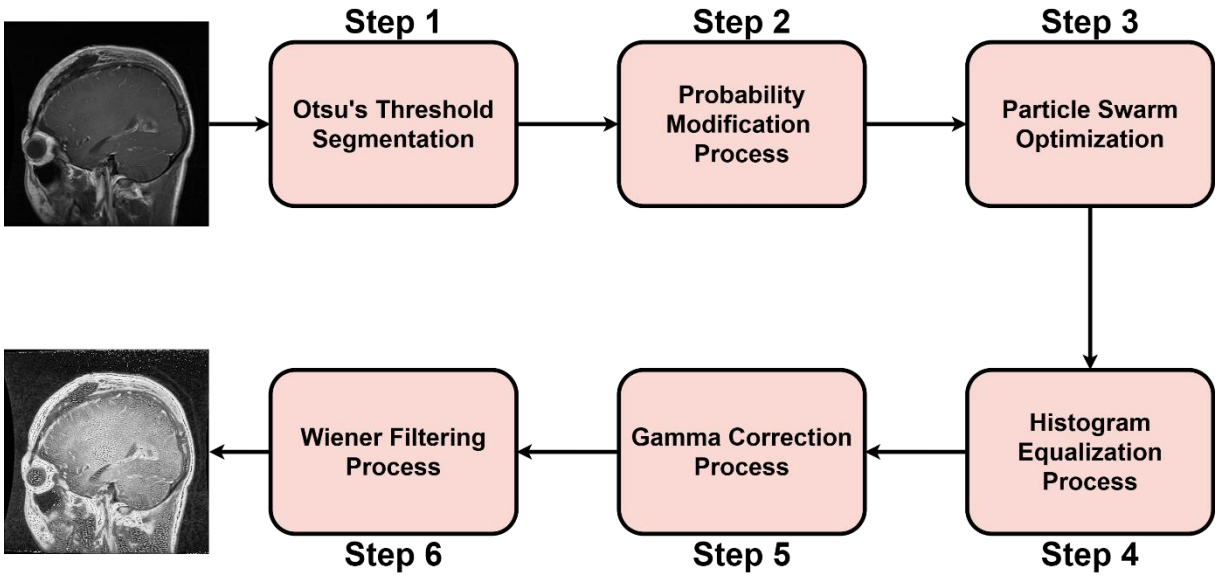


Figure 3.2 Pipeline of Image Preprocessing for Classification

This technique comprises six sequential steps for preprocessing brain MRI images. Initially, Otsu's double threshold method [92] is applied to segment the image histogram into three sub-histograms: target, background, and foreground. This method employs global thresholding by analyzing the histogram's shape and optimizing the threshold value to maximize inter-class variance, demonstrating robustness in segmenting complex images with multiple objects.

In the second step, a weighted normalized constrained model is utilized to adjust the statistical probability of the sub-histograms. This model assigns higher weights to fewer successive gray levels and lower weights to more successive gray levels, thereby mitigating the dominance of high-frequency histogram bins and preventing excessive amplification.

The third step involves determining the optimal parameter values for this model using Particle Swarm Optimization (PSO) [93], a computationally efficient algorithm known for its simplicity, rapid convergence, and low processing costs. PSO employs a swarm of particles distributed across the solution space, with each particle guided by position and velocity vectors. The entropy-based fitness function ensures an optimal balance between image enhancement and information preservation.

In the fourth stage, histogram equalization [94] is applied independently to each sub-histogram, redistributing intensity values across the input image and enhancing contrast through a transformation function. The fifth step employs adaptive gamma correction [95] to further refine global contrast while maintaining a balance between computational efficiency and visual quality. This process prevents substantial drops in

high-intensity values while appropriately enhancing low-intensity regions, avoiding distortions in the cumulative density function (CDF).

Finally, Wiener filtering [96] is applied to reduce noise in visually significant regions while maintaining image clarity. This linear filtering approach minimizes the MSE between the original and processed images by combining noise smoothing with inverse filtering for deconvolution. The result is an optimally enhanced MRI image, which facilitates improved tumor classification by preserving essential structural and pathological details.

Images are also augmented before being fed into the model to enhance predictive performance. The augmentation steps include:

- Applying zero padding to make the image square.
- Resizing to a fixed dimension of 224×224.
- Normalizing pixel values using the parameters: $mean = [0, 0, 0]$ and $std = [0.5, 0.5, 0.5]$ for the three color channels.
- Applying data augmentation by generating additional images through random rotations within the range of -90 to 90 degrees and horizontal/vertical flipping.

3.2.2. Image Segmentation Dataset

To improve data augmentation for the segmentation task, the model's input is expanded to four channels. Each channel is randomly selected from one of the four available pulse sequences in the patient's imaging data, specifically T1, T2, T1Ce, and FLAIR for the BRATS2020 dataset. This augmentation strategy increases the number of training samples in BraTS2020 by a factor of 16, thereby enhancing the model's capacity to integrate diverse anatomical and pathological information from different imaging modalities. The process of augmentation is presented in Figure 3.3.

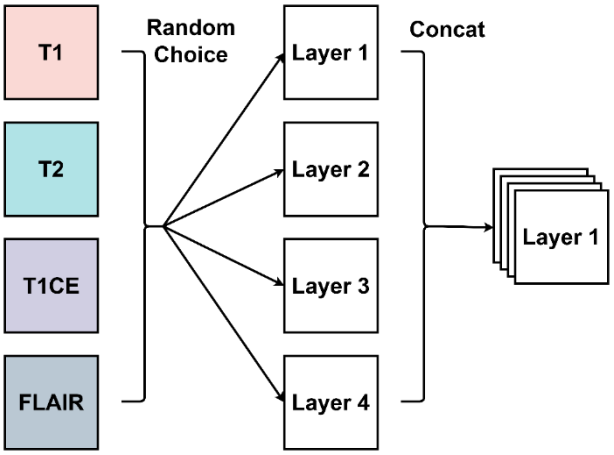


Figure 3.3 Illustration of Data Augmentation for Image Segmentation Dataset

This method closely aligns with the approach used in our synthesized dataset. However, due to the limitation that each sample consists of only a single type of image, this technique does not contribute to an increase in the overall number of training samples. Consequently, while it may enhance certain aspects of data quality, it does not provide the benefit of dataset expansion, which is often crucial for improving model generalization and performance in deep learning applications.

3.3. Implementation Details

3.3.1. MedCapNet

The proposed architecture, MedCapNet, is defined by key hyperparameters. Firstly, the embedding dimension D is set to 512, facilitating rich feature representation. Each Encoder or Decoder employs 8 attention heads, enabling the model to capture diverse aspects of the input simultaneously. Secondly, both the Enhancement Encoder and Decoder consist of $N = 3$ blocks, striking a balance between model depth and computational efficiency. The training procedure focuses on minimizing Cross Entropy loss [37] for the 900,000 steps in total and 10,000 steps for warm-up providing a strong foundation for the model's learning process. The Adam optimizer [97] is used to minimize the objective function during training. During validation and inference, a beam search algorithm [48] with a width of 10 is employed to generate optimal output sequences. Inspired by [48], the learning rate is adaptively calculated throughout the training process using the following formula:

$$lr = D^{-0.5} \times \min(stepnum^{-0.5}, stepnum \times warmupsteps^{-1.5})$$

3.3.2. GuidedDCNet

This study employs a diffusion model based on the original DDPM training strategy [1]. To be more specific, each timestep t is sampled randomly and independently from the set of integers $\{1, 2, \ldots, T\}$. Additionally, the noise level is scheduled following a linear process, with values set at $\beta_1 = 1 \times 10^{-4}$ and $\beta_T = 0.02$.

The Data, Local, and Global Encoders use ResNet101 backbones [44]. Initially, the MCGM module and Data Encoder undergo a pre-training phase for 25 epochs, utilizing Cross Entropy [37] as the loss function. Following this stage, the complete model is trained for 200 epochs, incorporating the pre-trained weights from MCGM to enhance learning stability and convergence. Both training phases leverage the Adam optimization algorithm [97] in conjunction with a cosine annealing scheduler. The initial learning rate is configured as 5×10^{-4} for the MCGM and Data Encoder, while the Unet model is assigned an initial learning rate of 10^{-3}. The hyperparameter λ in the total loss function, $\mathcal{L}_{total}$, is assigned a value of 0.5.

3.3.3. GuidedSegNet

As in GuidedDCNet, each timestep t is randomly and independently selected from the integer set $\{1, 2, \ldots, T\}$. . The noise level follows a linear scheduling process, starting at $\beta_1 = 1 \times 10^{-4}$ and increasing to $\beta_T = 0.02$. GuidedSegNet is trained in an end-to-end manner using the AdamW optimizer [97] with a batch size of 32. The initial learning rate was set to 1×10^{-4}. For inference, 100 diffusion steps were employed. The model is executed 10 times for ensemble generation. To aggregate the generated samples, the STAPLE algorithm [98] was applied for fusion.

3.3.4. BrainMedQwen

The Qwen2.5-VL 72B model [5] was fine-tuned using the Adam optimizer [97] to enhance its performance. The fine-tuning process was conducted over a total of 10 epochs, with a batch size of 4 to balance computational efficiency and training stability. The learning rate was set to 2×10^{-4} to facilitate effective gradient updates, and a warmup phase consisting of 50 steps was incorporated to ensure a smooth transition into stable training. These hyperparameters were carefully selected to optimize convergence and improve the model's adaptation to the target task.

Following the supervised fine-tuning (SFT) phase, the Direct Preference Optimization (DPO) method [99] is widely employed to enhance the output quality of VLMs. DPO is trained on a dataset comprising preference pairs $D = \{(v, x, y_w, y_l)\}$ where the preferred output y_w is ranked higher than the less favorable output y_l given the same visual input v and textual input x. The optimization objective of DPO is to maximize the difference in likelihood between the preference pairs, defined as:

$$\mathcal{L}_{DPO}(\pi_\theta; \pi_{ref}) = -\mathbb{E}_{(v,x,y_w,y_l)\sim D}\left[log\sigma\left(\beta\, log\frac{\pi_\theta(y_w||v,x)}{\pi_{ref}(y_w||v,x)} - \beta log\frac{\pi_\theta(y_l||v,x)}{\pi_{ref}(y_l||v,x)}\right)\right]$$

In the annotation process, v represents images, while x corresponds to the instructional input for generating scripts. The flawed output script generated by the VLM is denoted as y_l, whereas y_w represents the lecture script refined by human annotators. However, obtaining human feedback for long-form outputs is both time-intensive and costly.

Given that $y_w = \{y_w^i \mid i = 1, \ldots, N\}$ represents a revised script for an N-page document, we progressively treat each page's revised script y_w^i as a preferred segment over the corresponding flawed script. Consequently, the optimization objective is reformulated as:

$$\mathcal{L}_{IterDPO}(\pi_\theta; \pi_{ref}) = -\mathbb{E}_{(v,x,y_w,y_l)\sim D} \sum_{i=1}^{N} \left[log\sigma \left(\beta \, log \frac{\pi_\theta(y_{\leq i}^w || v_{\leq i}, x)}{\pi_{ref}(y_{\leq i}^w || v_{\leq i}, x)} - \beta \, log \frac{\pi_\theta(y_{\leq i}^l || v_{\leq i}, x)}{\pi_{ref}(y_{\leq i}^l || v_{\leq i}, x)} \right) \right]$$

where $y_{\leq i}^w$ and $y_{\leq i}^w$ denote the revised and unrevised scripts up to page i, respectively, and $v_{\leq i}$ represents the corresponding visual inputs. By treating $y_{\leq i}^w$ as a newly preferred response over $y_{\leq i}^l$ the model can learn fine-grained feedback on long-form outputs, effectively expanding the number of preference pairs by a factor of N. This approach increased of generation of iterative training pairs.

Beyond human-annotated feedback, we further leverage AI-generated feedback by employing GPT-4o [100] as a reward model. Following the Reinforcement Learning from AI Feedback (RLAIF) framework [101], responses were sampled from the SFT model across long-output instructions. GPT-4o was then utilized to assign length and quality scores to construct additional preference pairs. The final DPO model was trained on the preference pairs, incorporating both human and AI-generated feedback to enhance model performance.

3.3.5. Prompt Design

The prompt for the MedCapSys as well as Medical Question Answering Chatbot is developed based on the concept proposed by [102] to ensure that the enerated response meets the desired expectations. Specifically, the prompt is structured as follows:

- Role Assignment & Domain Expertise
- Context & Data Integration
- Structured & Systematic Output
- Instruction Execution Strategy
- Style & Precision Requirements

The detailed prompts for the MedCapSys and Medical Question Answering Chatbot are presented in APPENDIX A:.

3.4. Design and Implementation of the Demonstration System

3.4.1. Software Architecture and Design

The system is designed as a web-based platform that integrates advanced medical image analysis and interactive health information services. It offers two main functionalities: (1) automatic generation of medical reports from brain MRI images and (2) a chatbot capable of answering general medical inquiries. The software

architecture follows a modular design pattern to ensure scalability, maintainability, and extensibility.

3.4.1.1. Brain MRI Report Generation

The Brain MRI Report Generation function is a core component of the software system, enabling users to upload brain MRI images and receive automated diagnostic reports. This function is supported through a combination of user interactions, backend validation, and report generation services, as detailed in the architectural diagrams.

The Use Case Diagram (Figure 3.4) illustrates the primary scenario in which the user accesses the "View Brain MRI Report" functionality. This use case includes the "Upload Brain MRI Image" sub-function as a prerequisite step. This structure ensures that a report can only be generated once a valid image has been provided by the user.

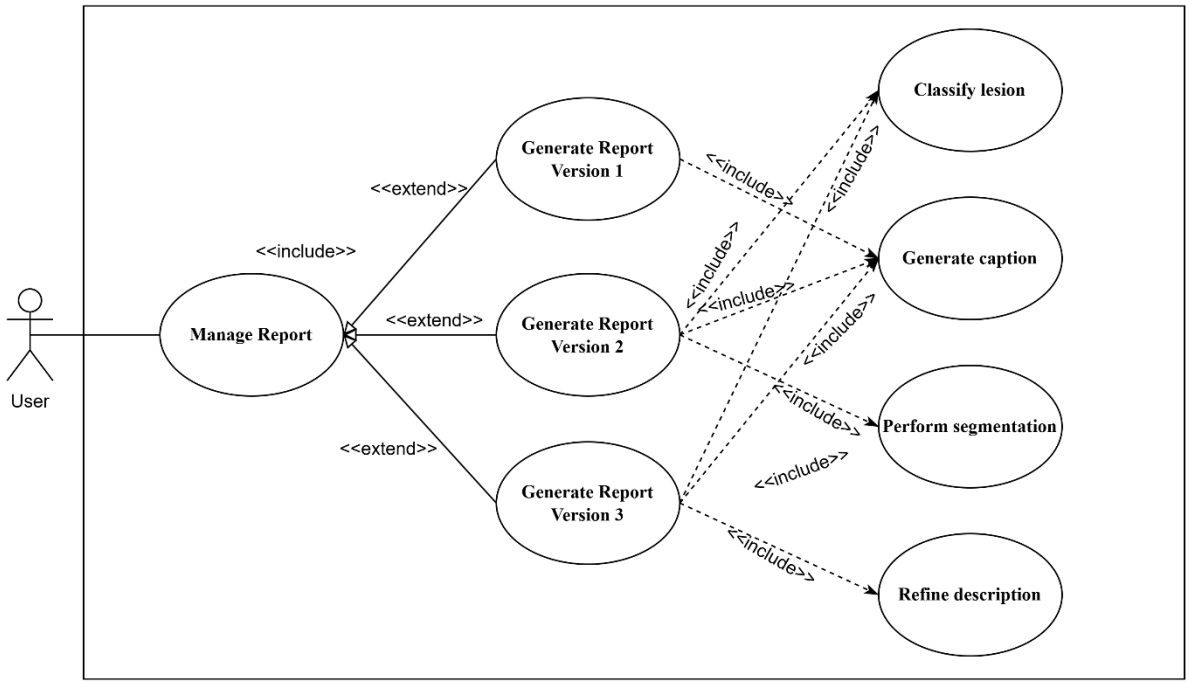


Figure 3.4 Use Case Diagram of the Brain MRI Report Generation Function

The Swimlane Diagram (Figure 3.5) further elaborates the flow of activities between the user and the system. The process starts with the user uploading a brain MRI image and selecting a version (e.g., report model version). The system then receives and validates the image format. If the format is valid, the system proceeds to generate the diagnostic report, which is ultimately made available for the user to view. If the image is invalid, the system halts the process, preventing the generation of a report and prompting the user to re-upload.

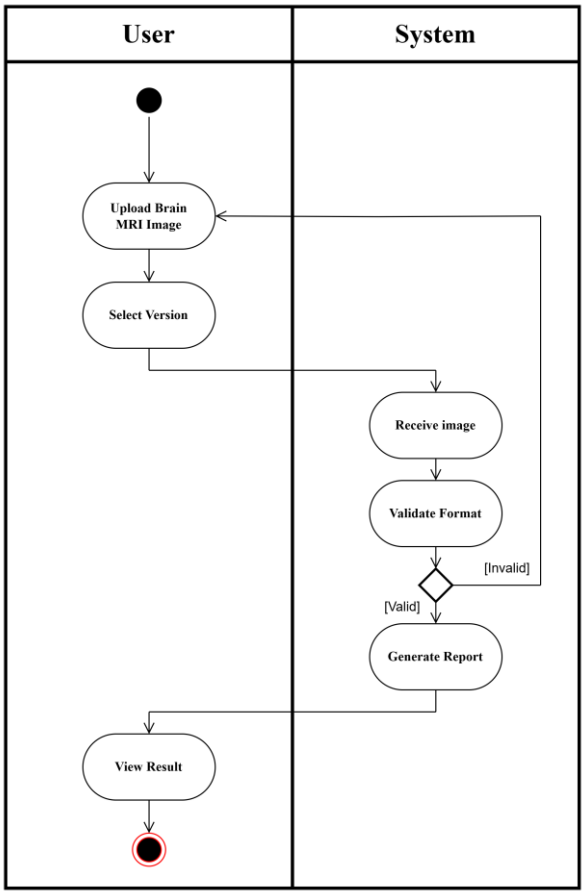


Figure 3.5 Swimlane Diagram of the Brain MRI Report Generation Function

The Sequence Diagram (Figure 3.6) provides a detailed interaction timeline between the user, user interface, and the report generation API. It begins with the user uploading the image and selecting the desired version. The user interface then sends both inputs to the Report Generation API. The API processes the request by invoking the report generation module. Once the diagnostic report is prepared, it is returned to the user interface and displayed to the user.

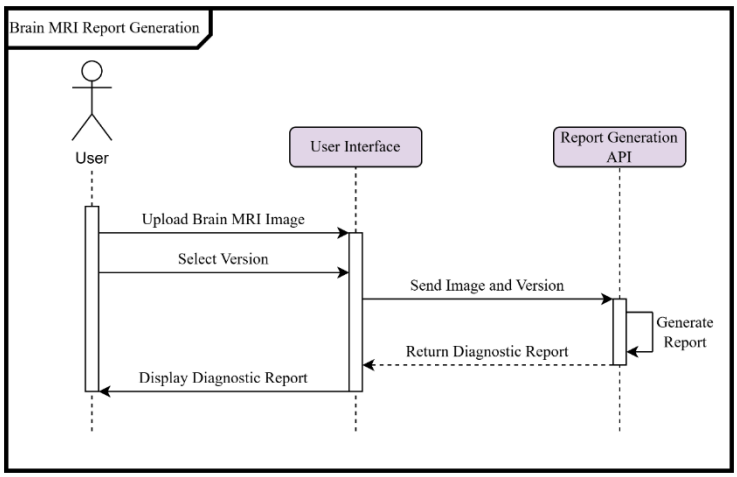


Figure 3.6 Sequence Diagram of the Brain MRI Report Generation Function

This function is designed to be modular and responsive. By separating responsibilities across the user interface and backend services, the system ensures that each component remains scalable and maintainable. Furthermore, the validation step before report generation enhances data integrity and system reliability.

3.4.1.2. Medical Question Answering Chatbot

The Medical Question Answering Chatbot is the second key feature designed to assist users in receiving immediate responses to health-related queries through a conversational interface. This function utilizes natural language input and an AI-powered backend to generate relevant medical answers in real time.

The Use Case Diagram (Figure 3.7) outlines the primary use case "Get Answer," which includes the "Send Message" functionality. This inclusion indicates that users must first send a message (i.e., their question) as part of the process of receiving an answer. It represents a straightforward interaction from the user's perspective but highlights the underlying step-by-step mechanism.

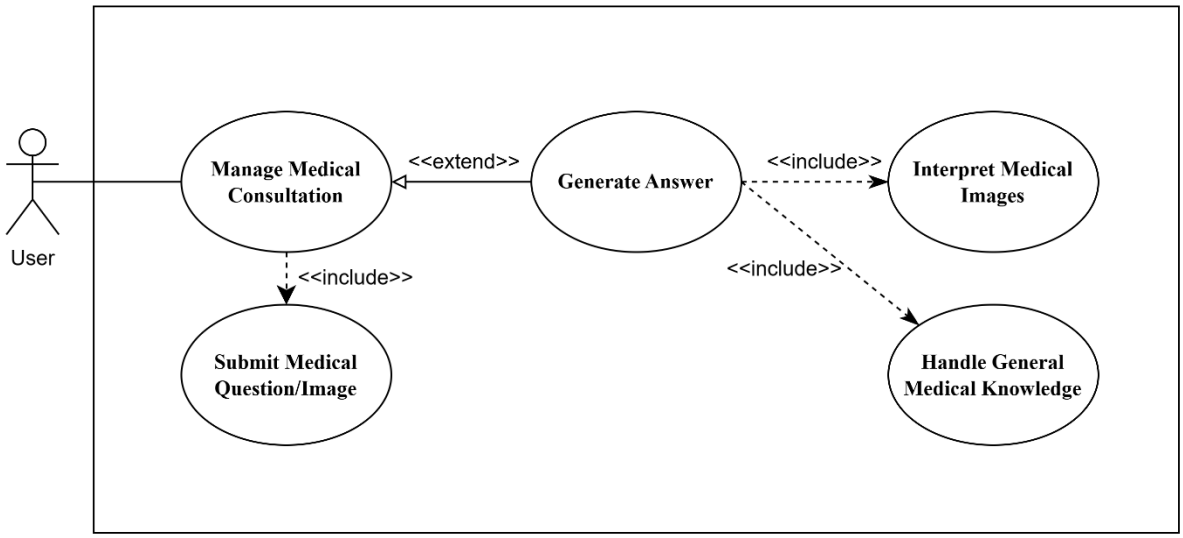


Figure 3.7 Use Case Diagram of the Medical Question Answering Chatbot

The Swimlane Diagram (Figure 3.8) elaborates on the dynamic flow of this interaction between the user and the system. The user initiates the process by entering a question. The system then receives and validates the question format. If the format is deemed valid, the system proceeds to generate an appropriate response using its underlying chatbot logic. The generated answer is then returned to the user for viewing. Invalid inputs are rejected, prompting users to re-enter a properly formatted query.

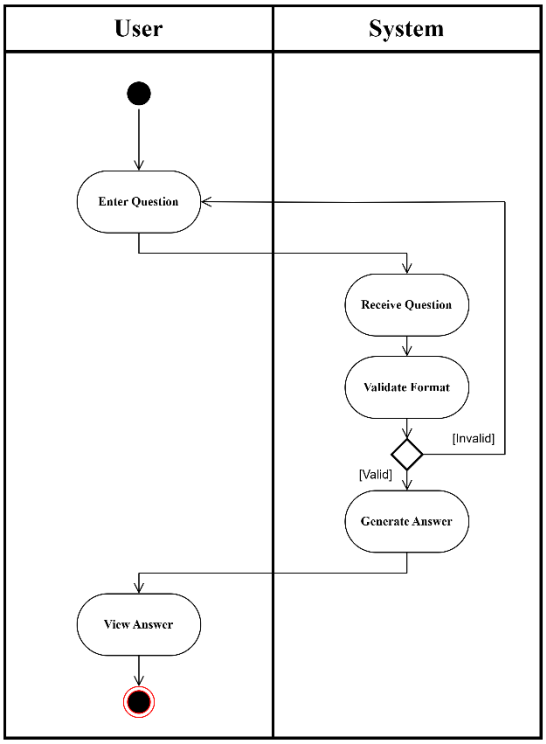


Figure 3.8 Swimlane Diagram of the Medical Question Answering Chatbot

The Sequence Diagram (Figure 3.9) further details the communication flow between the user, user interface, and chatbot API. The user sends a question through the user interface, which forwards the query to the Chatbot API. The API processes the input and generates a medical answer using its internal AI model. The response is then returned to the user interface and displayed to the user. This diagram emphasizes the asynchronous nature of the backend communication and highlights the system's modular design.

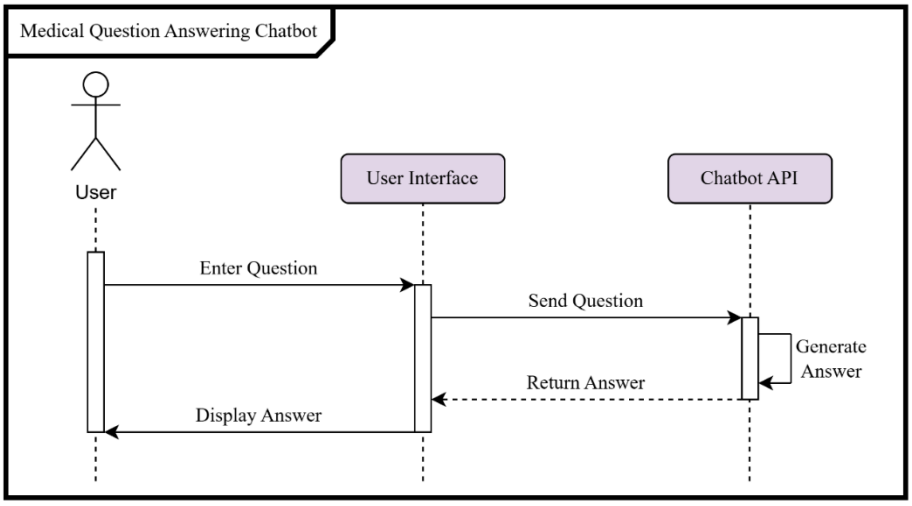


Figure 3.9 Sequence Diagram of the Medical Question Answering Chatbot

This chatbot function is designed to improve user experience by providing timely and contextually relevant medical information. By separating concerns between the UI and the API, the system achieves flexibility, scalability, and the ability to integrate more advanced natural language processing capabilities in future iterations.

3.4.2. System Implementation

3.4.2.1. Backend Implementation

a) Programming Language

The backend of the our system was implemented using Python version 3.11.11, a widely adopted programming language for artificial intelligence applications. Python was selected due to its simplicity, readability, and strong support from the AI and machine learning communities.

b) API Framework and Server

To build the RESTful API for communication between the user interface and the chatbot engine, the backend leveraged the FastAPI framework. FastAPI was chosen for its high performance, automatic data validation, asynchronous capabilities, and intuitive syntax, which allowed for rapid development and easy maintenance. Moreover, this framework is paired with Uvicorn, a lightning-fast ASGI server that runs the FastAPI app, supporting asynchronous I/O operations and improving the system's responsiveness.

c) Libraries

Several libraries were integrated into the backend to support key functionalities:

- Transformers and Diffusers for integrating deep learning models
- OpenCV, Pillow, Numpy, pydicom for handling and processing images

d) Chatbot Implementation with Agno Framework

To implement the medical question answering chatbot, the system utilizes the Agno Framework [103], a modular, extensible Python-based framework designed for building conversational AI systems with a clean separation of logic, context, and response generation. Agno provides a structured conversation pipeline that includes: (1) Message Handling Layer, (2) Intent Recognition and Routing, (3) Context Management (4) Custom Tools, (4) Response Generator, (5) Error Handling and Fallbacks, (6) Extensibility and Debugging. By using Agno, the we avoided building the chatbot pipeline from scratch and gained a clean framework to implement domain-specific skills and maintain logic consistency. The flexibility of Agno's architecture also allows future integration of additional AI models, more advanced dialogue management, and external medical data sources with minimal architectural changes.

e) Semantic Search with Qdrant

To enhance the chatbot's ability to understand and retrieve accurate medical knowledge based on semantic meaning rather than just keyword matching, the system integrates Qdrant [70], an open-source vector database optimized for high-performance similarity search and semantic retrieval. Qdrant is used as the backend knowledge base for the chatbot. When users submit questions that do not match any predefined intents or patterns in the Agno Framework, the query is converted into a vector embedding and used to search for the most relevant answers in the vector space.

f) API Deployment and Reverse Proxy

The backend API is containerized and deployed using Docker [104] to ensure consistency across development and production environments. The FastAPI application is placed behind an Nginx [105] reverse proxy to improve performance, reliability, and security.

In this architecture, Nginx acts as a front-facing web server that routes incoming HTTP/HTTPS requests to the internal Uvicorn server running the FastAPI app. Nginx handles TLS/SSL termination, static file serving, request buffering, and connection handling. This separation of concerns enables better load handling, support for connection keep-alive, and easier configuration for features such as rate limiting, caching, or custom headers.

3.4.2.2. Frontend Implementation

The frontend of the system was developed using Next.js, a powerful React-based web development framework optimized for building full-stack applications with server-side rendering (SSR) and static site generation (SSG).

The frontend was styled using Tailwind CSS, which supports utility-first design principles and ensures a consistent and responsive layout across desktop and mobile devices. Interactive behavior, form handling, and conditional rendering were managed using React's built-in hooks.

The entire frontend was deployed using Vercel, which is the official deployment platform for Next.js. Vercel offers a fully managed infrastructure optimized for Next.js applications and provides features.

This setup allows the frontend to scale automatically with traffic, serve global users efficiently through Vercel's edge network, and deliver an excellent user experience for medical professionals and patients accessing chatbot responses or diagnostic information.

3.5. Evaluation Metrics

3.5.1. Image Captioning Evaluation

To evaluate the performance of these models comprehensively, multiple evaluation metrics are utilized, combining both traditional and neural-based approaches. Among these, BERTScore [106] is employed as a state-of-the-art semantic similarity metric for sentence-level evaluation. Specifically, the evaluation uses the pre-trained "*microsoft/deberta-xlarge-mnli"* model as the underlying language representation model, which enhances the ability of BERTScore to capture nuanced semantic relationships between reference and generated sentences.

BERTScore computes similarity by aligning tokens in the generated sentence with those in the reference sentence using contextual embeddings derived from a transformer-based model. Let $X = (x_1, \dots, x_m)$ and $Y = (y_1, \dots, y_n)$ be the token embeddings of the reference and candidate sentences, respectively. Then, the precision, recall, and F1 score are calculated based on cosine similarity between these embeddings:

$$Precision = \frac{1}{|Y|}\sum_{y \in Y} \max_{x \in X} \text{cosine}(y, x)$$

$$Recall = \frac{1}{|X|}\sum_{x \in X} \max_{y \in Y} \text{cosine}(x, y)$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

In addition to BERTScore, several widely adopted metrics for text generation evaluation are also included:

- ROUGE-1: Measures the overlap of unigrams (i.e., single words) between the generated and reference texts. It is computed as:

$$ROUGE - 1 = \frac{\text{Number of overlapping unigrams}}{\text{Total unigrams in reference}}$$

- BLEU-1: Evaluates the precision of unigrams, i.e., how many unigrams in the generated text appear in the reference text. The general BLEU score is based on n-gram precision with a brevity penalty BP, but BLEU-1 focuses only on unigrams:

$$BLEU - 1 = BP \times \exp\left(\sum_{n=1}^{1} w_n \log p_n\right) = BP \times p_1$$

where p_1 is the unigram precision and BP is the brevity penalty defined as:

$$BP = \begin{cases} 1 & \text{if } c > r \\ e^{1-\frac{r}{c}} & \text{if c} \leq \text{r} \end{cases}$$

with c being the length of the candidate sentence and r the length of the reference.

- CIDEr: Measures the consensus between a generated sentence and a set of reference sentences using TF-IDF weighting for n-grams (up to 4-grams). It is defined as:

$$CIDEr = \frac{1}{N}\sum_{n=1}^{N} CIDEr_n$$

where each $CIDEr_n$ is calculated using the cosine similarity between TF-IDF vectors of n-grams.

- METEOR: Aligns words between the candidate and reference texts using exact matches, stemming, and synonyms. It considers both precision and recall, with a harmonic mean (F-score) adjusted by a fragmentation penalty Pen:

$$METEOR = F_{mean} \times (1 - Pen)$$

where:

$$F_{mean} = \frac{10 \times P \times R}{R + 9P}$$

and P and R are unigram precision and recall, respectively.

This multifaceted evaluation strategy ensures both syntactic and semantic alignment between generated and reference texts, offering a more comprehensive assessment of model performance across diverse aspects of language generation.

3.5.2. Image Classification Evaluation

To evaluate the performance of the image classification models, a set of standard evaluation metrics are employed, including Accuracy, Precision, Recall, and F1-score. These metrics provide a comprehensive understanding of the model's predictive capabilities, particularly in contexts where class distributions may be imbalanced.

Accuracy represents the proportion of correctly predicted instances over the total number of instances. It provides a general sense of the model's performance across all classes:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

where:

- TP (True Positive): number of correctly predicted positive instances.
- TN (True Negative): number of correctly predicted negative instances.

- FP (False Positive): number of negative instances incorrectly predicted as positive.
- FN (False Negative): number of positive instances incorrectly predicted as negative.

While accuracy is useful in balanced datasets, it can be misleading when classes are imbalanced.

Precision measures the proportion of correctly predicted positive instances out of all instances predicted as positive. It evaluates the model's ability to avoid false positives:

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

A high precision indicates that when the model predicts a positive label, it is likely to be correct.

Recall measures the proportion of actual positive instances that were correctly identified by the model:

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

A high recall means the model can successfully identify most of the positive instances, which is crucial in applications like medical diagnosis or fraud detection.

The F1-score provides a balanced metric that considers both precision and recall. It is especially useful in scenarios with class imbalance, where optimizing only one metric may not reflect overall performance adequately:

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

The F1-score is the harmonic mean of precision and recall. Unlike the arithmetic mean, the harmonic mean penalizes extreme values more, which ensures that both precision and recall must be reasonably high for the F1-score to be high.

By combining these evaluation metrics, we obtain a robust and nuanced assessment of the image classification models' performance, allowing for both global correctness (accuracy) and class-specific behavior (precision, recall, and F1-score) to be thoroughly analyzed.

3.5.3. Image Segmentation Evaluation

To evaluate the performance of image segmentation models, particularly in domains where both region consistency and boundary accuracy are essential, three widely used and complementary metrics are employed: Dice Coefficient, Intersection over Union (IoU), and the 95th percentile Hausdorff Distance (HD95). These metrics collectively assess the quality of the predicted segmentation masks by capturing both pixel-wise overlap and boundary alignment with the ground truth annotations.

The Dice Coefficient measures the degree of overlap between the predicted segmentation mask and the ground truth mask. It is especially sensitive to mismatches in small structures and is often preferred in medical image analysis due to its sensitivity to the presence of false negatives.

$$Dice = \frac{2 \times |A \cap B|}{|A| + |B|}$$

where A is the set of predicted positive pixels (segmented region) and B is the set of ground truth positive pixels. The Dice score ranges from 0 (no overlap) to 1 (perfect overlap), and a higher score indicates better segmentation performance.

IoU, also known as the Jaccard Index, is another region-based metric that quantifies the similarity between the predicted and ground truth masks. It is defined as the size of the intersection divided by the size of the union of the two sets:

$$IoU = \frac{|A \cap B|}{|A \cup B|}$$

The IoU also ranges from 0 to 1, with higher values indicating better segmentation quality. While similar to the Dice coefficient, IoU is generally considered slightly more strict in penalizing mismatches.

The Hausdorff Distance (HD) measures the maximum distance between the boundary points of the predicted segmentation and the ground truth. However, since the standard HD is highly sensitive to outliers, the HD95 is often used for robustness. Let S and G denote the sets of boundary points of the predicted and ground truth masks, respectively. The directed Hausdorff distance from S to G is:

$$d(S, G) = \max_{s \in S} \min_{g \in G} \|s - g\|$$

The symmetric Hausdorff Distance is then:

$$HD(S, G) = \max\{d(S, G), d(G, S)\}$$

The HD95 is defined as the 95th percentile of the distribution of all minimum distances from boundary points in one set to the other:

$$HD_{95} = \max\left\{\text{percentile}_{95}\left(\left\{\min_{g \in G}\|s - g\| \,|s \in S\right\} \cup \left\{\min_{s \in S}\|s - g\| \,|g \in G\right\}\right)\right\}$$

A lower HD95 value indicates that the predicted boundaries are closer to the ground truth, and therefore the segmentation is more spatially accurate.

3.5.4. Prompt Design Evaluation

Following the methodology proposed by [22], we conduct a two-pronged evaluation of the output quality and length conformity of the designed prompts. This approach ensures that generated responses are not only content-rich and high quality but also align well with specified constraints on output length. To this end, two key metrics are used: length score S_l and quality score S_q.

The length score S_l quantifies how closely the actual output length l_v produced by the model matches the required or target length l_r. The aim is to reward outputs that adhere to the desired verbosity, penalizing those that are either too short or too long. The score is computed using a piecewise function as follows:

$$S_l = \begin{cases} 100 \times \max\left(0, 1 - \left(\frac{l_v}{l_r} - 1\right)\right), \text{if } l_v > l_r \\ 100 \times \max\left(0, 1 - \left(\frac{l_r}{l_v} - 1\right)\right), \text{if } l_v \leq l_r \end{cases}$$

where l_v is length of the model's output (in tokens), l_r is required or target output length.

To assess the semantic and stylistic quality of the model's output, we employ GPT-4o [32] as an automatic evaluator. The quality score S_q is assigned based on six key dimensions that capture both the informational content and the user-perceived readability of the text:

- Relevance: How well the content addresses the visual prompt or question.
- Accuracy: The factual correctness of the output, particularly with respect to visual details.
- Coherence: Logical consistency within the output; ideas should be well connected.
- Clarity: Fluency and grammatical correctness of the text.
- Breadth and Depth: Completeness and level of detail provided in the explanation or description.
- Overall Reading Experience: The subjective impression of quality from a human reader's perspective.

Each aspect is scored independently, and the final quality score S_q is computed as the average of the six component scores, each typically normalized to a 0-100 scale:

$$S_q = \frac{1}{6}\sum_{i=1}^{6} s_i$$

where s_i is the individual score for each quality aspect.

Together, the two metrics, S_l for length conformity and S_q for qualitative assessment, offer a holistic evaluation of the Vision-Language Model's performance. While S_l enforces adherence to structural requirements (such as being concise or elaborate), S_q ensures that the content remains meaningful, coherent, and engaging. This dual evaluation framework is particularly useful for generating descriptive captions, explanations, or answers that must be both controlled and high-quality.

3.6. Experimental Results

3.6.1. Image Captioning Results

Table 3.3 presents a comparative analysis of various approaches to image captioning. The models under consideration are categorized into two primary architectures: CNN-LSTM and CNN-Transformer. The CNN-LSTM category is represented by GCN-LSTM [54] and X-LAN [55], while CNN-Transformer is exemplified by X-Transformer [55] and M^2 Transformer [56]. To establish a benchmark for performance evaluation, the results of the top performing teams in the ImageCLEFmedical competition [57], including CSIRO [58], closeAI2023 [107], AUEB-NLP-Group [108], and PCLmed [59], were used as a reference point for comparison with the proposed model. The proposed model demonstrated superior performance across the BERTScore, CIDEr, and METEOR metrics, achieving scores of 0.647, 0.239, and 0.094, respectively. No tably, regarding BERTScore, MedCapNet surpassed the state-of-the-art model from CSIRO by a margin of 0.006, 0.036, and 0.014. However, a comparative analysis of ROUGE-1 and BLEU-1 revealed that the proposed model underper formed the model from team PCLmed on these metrics. MedCapNet achieves a ROUGE-1 score of 0.244, which is competitive but slightly lower than PCLmed (0.253) and CSIRO (0.246) models. This indicates that its ability to match un igrams (words) in the reference captions is strong, though not the best in this comparison. With a BLEU-1 score of 0.162, MedCapNet performs similarly to other models like the model of CSIRO (0.162) and AUEB-NLP-Group (0.169), but it is lower than that of PCLmed, which has the highest score (0.217).

Table 3.3 Comparison the performance of MedCapNet and other models

Model/Team	**BERT-Score** (↑)	**ROUGE-1** (↑)	**BLEU-1** (↑)	**CIDEr** (↑)	**METEOR** (↑)

GCN-LSTM	0.584	0.237	0.157	0.185	0.075
X-LAN	0.629	0.240	0.157	0.192	0.078
X-Transformer	0.630	0.241	0.159	0.195	0.079
M^2 Transformer	0.622	0.239	0.158	0.195	0.078
PCLMed	0.615	**0.253**	**0.217**	0.232	0.092
AUEB-NLP-Group	0.617	0.213	0.169	0.147	0.072
closeAI2023	0.628	0.240	0.185	0.238	0.087
CSIRO	0.641	0.246	0.162	0.203	0.080
MedCapNet	**0.647**	0.244	0.162	**0.239**	**0.094**

Table 3.4 compares MedCapNet's performance with and without key components (FM, EEb, DLSA, and SGSA) across five metrics. The full MedCapNet model achieves the highest scores, showing that each module contributes to generating high-quality captions. Removing FM leads to the largest drop in performance, with BERTScore falling to 0.615 and METEOR to 0.086, indicating FM's importance for semantic coherence. Excluding EEb also lowers ROUGE-1 and CIDEr, emphasizing its role in content precision. The absence of DLSA and SGSAslightly reduces scores, with each component adding a unique value. Ablation studies revealed a significant decline in performance across all metrics when either component is removed. While both components are instrumental, the FM exerted a slightly more pronounced influence on metrics such as METEOR and ROUGE-1. These findings underscore the synergistic relationship among FM, EEb, DLSA, and SGSA in achieving the model's peak performance.

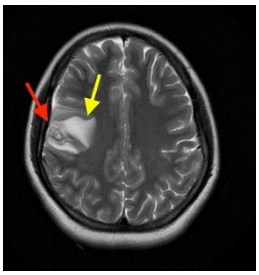
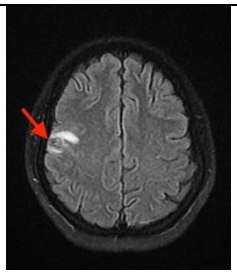
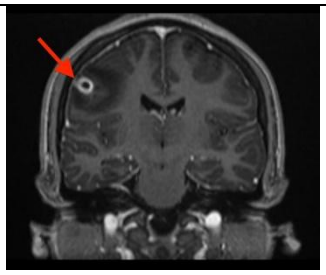
Table 3.4 Ablation experiment on the effect of Fusion Module (FM), Enhancement Encoder block (EEb), Dense Local Self Attention (DLSA), and Sparse Global Self Attention (SGSA)

Model	**BERT-Score** (↑)	**ROUGE-1** (↑)	**BLEU-1** (↑)	**CIDEr** (↑)	**METEOR** (↑)
MedCapNet w/o FM	0.615	0.240	0.157	0.232	0.086
MedCapNet w/o EEb	0.641	0.236	0.158	0.235	0.087
MedCapNet w/o DLSA	0.643	0.240	0.156	0.231	0.088
MedCapNet	0.642	0.238	0.159	0.231	0.090

w/o SGSA					
MedCapNet	**0.647**	0.244	0.162	**0.239**	**0.094**

As shown in Table 4, the generated captions preserve semantic accuracy but use some synonymous expressions, likely explaining MedCapNet's high scores on BERTScore, CIDEr, and METEOR, with lower scores on ROUGE-1 and BLEU-1. Despite achieving state-of-the-art (SOTA) performance, the generated captions remain inadequate for practical clinical applications. Specifically, they fail to provide critical information, including lesion size, the suspected site of origin, and the presence of surrounding necrosis, which are essential for accurate medical interpretation and decision-making.

Table 3.5 Example of Generated Captions from MedCapNet

Image	Generated Caption	Label
	Axial T2-weighted MRI scan of the brain showing a mass in the left frontal lobe (red arrow).	Axial view MRI brain showing solitary cystic mass (red arrow), with surrounding vasogenic edema (yellow arrow).
	Axial T2-weighted MRI scan of the brain showing a hyperintensity in the left frontal lobe (red arrow).	Axial brain MRI showing decrease in size of cystic mass (red arrow).
	Coronal T1-weighted MRI scan of the brain showing a mass in the left cerebellum (red arrow).	Coronal view MRI brain showing cystic mass with thickened peripheral enhancement (red arrow).

3.6.2. Image Classification Results

GuidedDCNet demonstrates strong classification performance on brain tumor classification datasets. The model achieves 93.67% accuracy and an 90.72 F1-score, significantly outperforming ConvNeXtV2, as well as traditional architectures like ResNet101 and Vision Transformer (Table 3.6).

Table 3.6 Quantitative results for Image Classification

Preprocessing	Method	Accuracy (↑)	Precision (↑)	Recall (↑)	F1-Score (↑)
-	ResNet101	74.41	72.10	69.32	70.68
	ViT	82.51	81.65	80.57	81.11
	ConvNeXtV2	87.52	88.37	85.74	87.04
	GuidedDCNet	**93.67**	**92.00**	**89.44**	**90.72**
✓	ResNet101	93.78	92.72	93.16	92.94
	ViT	94.57	93.75	94.92	94.33
	ConvNeXtV2	97.10	95.54	95.31	95.42
	GuidedDCNet	**99.51**	**98.09**	**97.78**	**97.93**

The performance is further enhanced when combined with preprocessing, GuidedDCNet attains state-of-the-art results with 99.51% accuracy and a 97.93 F1-score, surpassing all prior models. Its high precision and recall further demonstrate its balanced performance in identifying tumor types. These findings highlight the model's efficacy in handling fine-grained classification and class imbalance challenges.

The model's architecture is particularly effective in capturing multi-scale visual features. MSCGM's global guidance vector captures overall morphological patterns, while the local guidance mechanism, leveraging SHAP, identifies discriminative features. This approach is especially beneficial for brain tumor classification, where distinguishing subtle tumor type-specific traits is critical, and macro- and micro-level feature extraction enhances classification.

To evaluate the contribution of each architectural component in GuidedD CNet, ablation studies were conducted across three data modalities (Table 3.7). The analysis began with the global stream alone, followed by the sequential integration of the local stream, diffusion process, and MMD regularization to assess their individual impact on performance.

Table 3.7 Ablation Study in GuidedDCNet

Local Stream	Diffusion	MMD	Accuracy (↑)	F1-Score (↑)
-	-	-	93.78	92.94
✓	-	-	96.57	94.78
✓	✓	-	98.37	97.70

✓	✓	✓	**99.51**	**97.93**

Accuracy improves from 93.78% (global stream) to 96.57% with the local stream, demonstrating its role in capturing local details. The diffusion process further enhances accuracy to 98.37%, while MMD regularization achieves the highest accuracy of 99.51%, reinforcing its impact on feature discrimination. A similar trend is observed in F1-Score, where this figure increases from 92.94% to 97.93% with the full model, demonstrating its ability to capture complex relationships.

Accuracy improves from 93.78% (global stream) to 96.57% with the local stream, demonstrating its role in capturing local details. The diffusion process further enhances accuracy to 98.37%, while MMD regularization achieves the highest accuracy of 99.51%, reinforcing its impact on feature discrimination. A similar trend is observed in F1-Score, where this figure increases from 92.94% to 97.93% with the full model, demonstrating its ability to capture complex relationships.

These findings underscore the complementary roles of multi-scale feature extraction, diffusion-based denoising, and MMD regularization in enhancing classification performance across diverse data types, validating the architectural design choices of GuidedDCNet.

3.6.3. Image Segmentation Results

The results in Table 3.8 demonstrate that GuidedSeg Net outperforms existing state-of-the-art (SOTA) segmentation methods on the BraTS2020 dataset across all evaluated metrics. Specifically, it achieves the high est Dice score (90.8) and IoU (83.4), indicating superior segmentation accuracy and overlaps with ground truth labels. Moreover, GuidedSegNet attains the low est HD95 value (7.53), reflecting enhanced boundary precision and reduced segmentation errors. Compared to prior methods such as MedSegDiff, SwinBTS, and nnUNet, GuidedSegNet exhibits substantial improvements, particularly in reducing HD95, which suggests better spatial consistency. These results highlight the effectiveness of the proposed approach in brain tumor segmentation, demonstrating its potential for clinical applications.

Table 3.8 The comparison of GuidedSegNet with segmentation methods

Method	**Dice** (↑)	**IoU** (↑)	**HD95** (↓)
TransBTS	87.6	78.44	12.44
SwinBTS	88.7	81.2	10.03
nnUNet	88.5	80.6	11.20
TransUNet	86.6	79.0	13.74

Swin-UNetr	88.4	81.8	11.36
SegDiff	85.7	77.0	14.31
MedSegDiff	88.9	81.2	10.41
GuidedSegNet	**90.8**	**83.4**	**7.53**

The ablation study presented in Table 3.9 demonstrates the contribution of the Anchor Condition and Semantic Condition to the performance of GuidedSegNet. The baseline model, without both conditions, achieves a Dice score of 88.2, an IoU of 80.0, and an HD95 of 11.5. Introducing the Anchor Condition alone improves performance, increasing the Dicescore to 89.4 and reducing HD95 to 9.8, indicating enhanced spatial consistency. Similarly, incorporating only the Semantic Condition results in a more significant improvement, with a Dice score of 90.1 and an HD95 of 8.6, suggesting that semantic guidance plays a crucial role in refining segmentation accuracy. When both conditions are applied together, the model achieves the best performance, with a Dice score of 90.8, IoU of 83.4, and the lowest HD95 of 7.53. These results highlight the complementary effects of both conditions, demonstrating that their combined use leads to the most accurate and spatially consistent segmentation.

Table 3.9 Ablation Study in GuidedSegNet

Anchor Condition	**Semantic Condition**	**Dice** (↑)	**IoU** (↑)	**HD95** (↓)
-	-	88.2	80.0	11.56
✓	-	89.4	81.2	9.83
-	✓	90.1	82.0	8.61
✓	✓	90.8	83.4	7.53

The segmentation results produced by the proposed GuidedSegNet demonstrate strong performance across various MRI brain scans, as illustrated in Figure 3.10. The segmented regions (highlighted in red) closely align with the ground truth lesion areas, capturing both core and peripheral tumor regions with high spatial accuracy. Quantitatively, the model achieved a Dice coefficient of 90.8%, indicating a high degree of overlap between the predicted and actual segmentation masks. The IoU reached 83.4%, further confirming the robustness and precision of the predicted regions. In addition, the HD95 was measured at 7.53, reflecting the model's ability to

closely approximate the ground truth boundaries while remaining resilient to outliers. These results suggest that GuidedSegNet offers reliable and accurate delineation of tumor regions, making it well-suited for clinical and diagnostic applications in medical image segmentation.

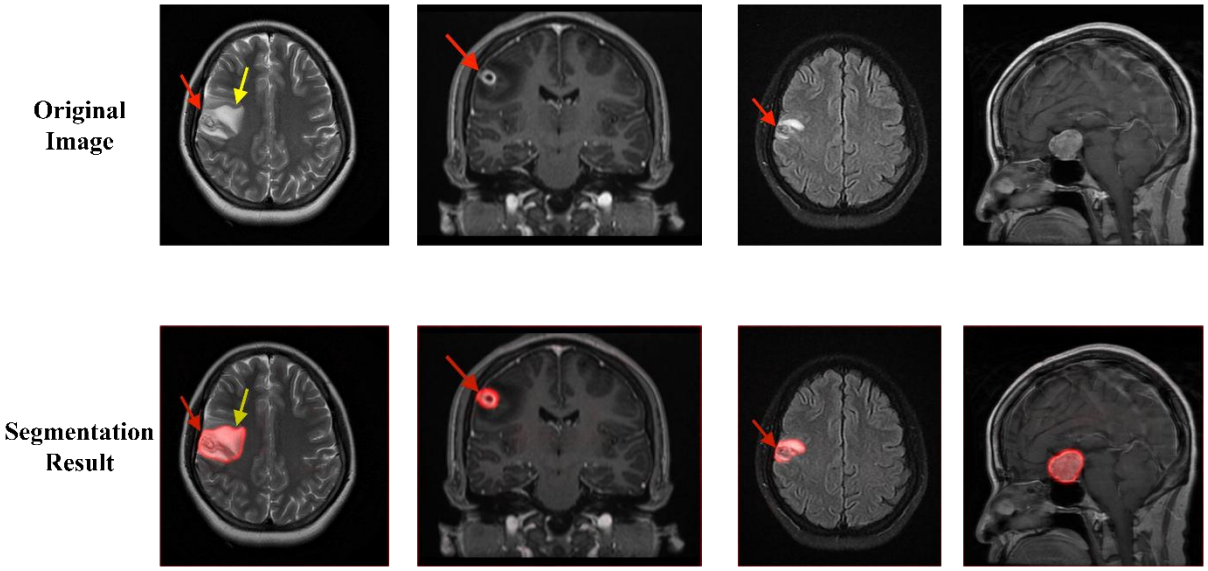


Figure 3.10 Segmentation Results from GuidedSegNet

3.6.4. Brain MRI Report Generation Results

The proposed MedCapSys system demonstrates superior performance in Brain MRI report generation, excelling in both content quality and length conformity, as measured by the S_q and S_l metrics, respectively. As illustrated in Table 3.10MedCapSys achieves the highest score for length alignment with $S_l = 84.3$, substantially outperforming other state-of-the-art models such as GPT-4o (66.6), Mistral-24B (69.6), and Gemini-2.0 Flash (74.8). This high score indicates that MedCapSys can reliably generate radiology reports whose lengths are closely aligned with the reference or expected standard, which is essential in clinical environments where concise yet comprehensive documentation is necessary for effective diagnosis and communication.

Moreover, in terms of semantic and linguistic quality, MedCapSys achieves the highest quality score with $S_q = 95.5$, surpassing all compared baselines, including DeepSeek-R1 (94.5) and Gemini-2.0 Flash (91.2), and significantly outperforming popular vision-language models such as GPT-4o (87.5) and Qwen2.5-VL 72B (86.7). This demonstrates that the generated reports by MedCapSys not only capture relevant medical content but also exhibit high levels of coherence, accuracy, relevance, and clarity, all of which are fundamental for clinical interpretability and usability.

Taken together, these results highlight the balanced and consistent capabilities of MedCapSys in producing structured, readable, and medically informative reports. Its ability to meet strict format expectations (through high S_l) while maintaining rich content quality (via high S_q) makes it particularly well-suited for deployment in automated clinical reporting pipelines, especially in domains such as neuroimaging, where precision and consistency are paramount. The significant performance margins over other competitive models further underscore MedCapSys's effectiveness in integrating vision-language understanding with medical domain knowledge.

Table 3.10 The comparison of BrainMedQwen with SOTA VLMs

Method	$\overline{\boldsymbol{S}}$ (↑)	$\boldsymbol{S_l}$ (↑)	$\boldsymbol{S_q}$ (↑)
GPT-4o	77.1	66.6	87.5
Mistral 24B	78.9	69.6	88.2
DeepSeek-R1	82.4	70.3	94.5
Gemini-2.0 Flash	83.0	74.8	91.2
Qwen2.5-VL 7B	54.5	45.3	63.5
Qwen2.5-VL 72B	83.3	79.9	86.7
MedCapSys	**89.9**	**84.3**	**95.5**

The results presented in Table 10 indicate that employing VLMs such as GPT-4o, Gemini-2.0 Flash in isolation for generating detailed captions of brain medical images frequently leads to hallucinations, primarily due to insufficient input information. Augmenting these models with additional contextual information mitigates this issue by enhancing their ability to generate more accurate and clinically relevant descriptions. Furthermore, fine-tuning the Qwen2.5-VL model on a domain-specific dataset of brain medical images significantly improves its performance, enabling it to produce responses that more closely align with radiologists' annotations.

Table 3.11 Generated Reports Examples from Different Methods

Gemini-2.0 Flash	On the coronal T1-weighted post-contrast MRI sequence, a region of increased signal intensity is identified in the right frontal lobe. This region demonstrates heterogeneous enhancement with well-defined borders. There is no significant brain shift appreciated. Based on the imaging characteristics, this is concerning for metastatic disease. The lesion measures approximately 1.5 cm in diameter. Its suspected origin is cortical. There is no surrounding vasogenic edema or necrosis.	Axial FLAIR MRI demonstrates a region of increased signal intensity in the right frontal lobe, appearing heterogeneously bright. The lesion exhibits relatively well-defined boundaries. Minimal to no midline shift is observed. The characteristics are suggestive of a primary brain tumor, possibly a glioma. The tumor measures approximately 3 cm in maximum diameter. Its suspected origin is within the right frontal cortex. There is a suggestion of surrounding vasogenic edema, but no obvious central necrosis is identified on this sequence.	T2-weighted axial MRI demonstrates a region of increased signal intensity in the right frontal lobe. The lesion is heterogeneous with both cystic and solid components. The boundaries are relatively ill-defined. There is evidence of mild midline shift towards the left. The imaging characteristics are suggestive of a high-grade glioma, possibly glioblastoma multiforme. The estimated size of the tumor is approximately 4 cm in greatest dimension. Given its location and appearance, the suspected origin location is likely within the right frontal white matter. Surrounding vasogenic edema is present, but there is no clear evidence of frank necrosis within the lesion.

GPT-4o	This MRI image employs a T1-weighted pulse sequence. There is a region exhibiting increased signal intensity, appearing brighter, at the location indicated by the arrow. The lesion appears to be relatively homogeneous in its signal characteristics. Its boundaries are well-defined. There is no evident brain shift or midline shift observed. The lesion's imaging characteristics and location are suggestive of a metastatic brain tumor. The exact dimensions of the tumor are not discernible from the image provided, but the lesion appears compact. The suspected site of origin seems to be the cerebral cortex near the convexity of the brain. There is no evidence of surrounding necrosis accompanying the lesion.	This brain MRI utilizes a T2-weighted FLAIR sequence, revealing a region of increased signal intensity appearing brighter in the left frontal lobe. The lesion is heterogeneous with predominantly well-defined boundaries. There is no significant brain shift or midline shift observed. The imaging characteristics suggest a high-grade glioma, likely glioblastoma multiforme, with an approximate size of 2.5 x 2.0 cm. The suspected origin is the left frontal cortex. There is presence of surrounding necrosis and associated vasogenic edema, contributing to mass effect but with no discernible herniation.	This MRI scan is performed using a T2-weighted sequence. There is a region of increased signal intensity, appearing brighter, located in the left frontal lobe. The region demonstrates heterogeneous signal characteristics with ill-defined boundaries, suggestive of infiltrative growth. There is a noticeable midline shift to the right, indicating significant mass effect. The imaging characteristics are consistent with a high-grade glioma, such as glioblastoma multiforme. The tumor measures approximately 4.5 cm in maximum dimension. It is suspected to originate from the left frontal lobe cortex. There is adjacent vasogenic edema without definitive areas of necrosis observed in this sequence.

MedCapSys	The T1-weighted coronal MRI scan of the brain with contrast injection reveals a round, well-defined, continuous hyperintense nodule in the right cerebral hemisphere, slightly compressing the ventricle. Tumor type: no tumor detected, suspected brain abscess. The suspected origin is within the right cerebral hemisphere. There is infiltration of the adjacent brain tissue.	The T1-weighted axial MRI scan of the brain with contrast injection reveals an area of increased signal intensity with irregular and heterogeneous borders in the right cerebral hemisphere. This hyperintense lesion does not cause midline shift or compress the ventricles. No tumor detected, suspected cerebral hemorrhage. The suspected origin is within the right cerebral hemisphere. There is infiltration and edema of the adjacent brain tissue.	The T2-weighted axial MRI scan of the brain without contrast injection reveals an area of increased signal intensity with irregular and heterogeneous borders in the right cerebral hemisphere. This hyperintense lesion causes midline shift and compresses the ventricles. No tumor detected, suspected cerebral hemorrhage. The suspected origin is within the right cerebral hemisphere. There is infiltration and edema of the adjacent brain tissue.

As shown in Table 3.12, MedCapSys exhibits a significantly higher average inference time (64.029 seconds) compared to Gemini-2.0 Flash (3.208 seconds) and GPT-4o (2.645 seconds) when generating brain MRI reports for 100 samples. This increased processing time is expected due to the modular and multi-stage architecture of MedCapSys. Unlike general-purpose AI models, MedCapSys integrates several specialized components, including lesion detection, segmentation, and medical language generation, which must operate sequentially to ensure clinical accuracy and contextual relevance.

Table 3.12 Comparison of average brain MRI report generation time (in seconds) across 100 samples using different methods

Method	Average Inference Time (s)
Gemini-2.0 Flash	3.208
GPT-4o	2.645
MedCapSys	64.029

Despite the longer inference time, MedCapSys remains considerably faster than traditional clinical workflows. For instance, in a real-world setting, radiologists may require 5 to 15 minutes on average to manually analyze a brain MRI scan and draft a comprehensive report. In contrast, MedCapSys can generate a detailed, multi-modal report in just over one minute, representing a substantial time saving while also providing consistent and interpretable results.

This trade-off highlights the system's focus on diagnostic depth and explainability over raw speed, making it a promising solution for integration into semi-automated radiology workflows where accuracy and justification are critical.

Table 3.13 presents a comparison of the average response times (in seconds) for 100 medical question answering samples across different models used in the agent component of a chatbot. The results show that BrainMedQwen, significantly outperforms both Gemini-2.0 Flash and GPT-4o in terms of response speed. Specifically, BrainMedQwen achieves the lowest response time in both message types: 5.046 seconds for text-only messages and just 1.453 seconds for messages containing both text and images. In contrast, GPT-4o and Gemini-2.0 Flash display longer response times, especially in the text + image category. These findings highlight BrainMedQwen's efficiency and suitability for real-time medical question-answering

tasks. key reason for BrainMedQwen's superior performance is that it is self-hosted on local infrastructure, allowing it to avoid the network latency and request queuing associated with cloud-based APIs used by the other models. This direct access to computational resources enables faster inference and more consistent performance, which is particularly critical in real-time medical applications.

Table 3.13 Comparison of average response time (in seconds) across 100 medical question answering samples using different models for the agent component of the chatbot

Model	**Message Type**	
	Text	**Text + Image**
Gemini-2.0 Flash	8.372	12.359
GPT-4o	6.717	3.356
BrainMedQwen	**5.046**	**1.453**

3.7. Demonstration of the Web Application

A dedicated website has been developed to serve as a platform for presenting the experiments that have been conducted. This website functions as a comprehensive repository, providing structured access to the research findings, experimental data, results, and related materials. The website is publicly accessible at https://brain-medical-analysis.vercel.app. As depicted in Figure 3.11, the website features an intuitive and user-friendly interface, enabling users to explore brain medical image analysis experiments, examine datasets, and gain insights into the research methodology and outcomes. This platform serves as a valuable resource for researchers, practitioners, and other interested stakeholders, facilitating the dissemination and review of the experimental work conducted in this study.

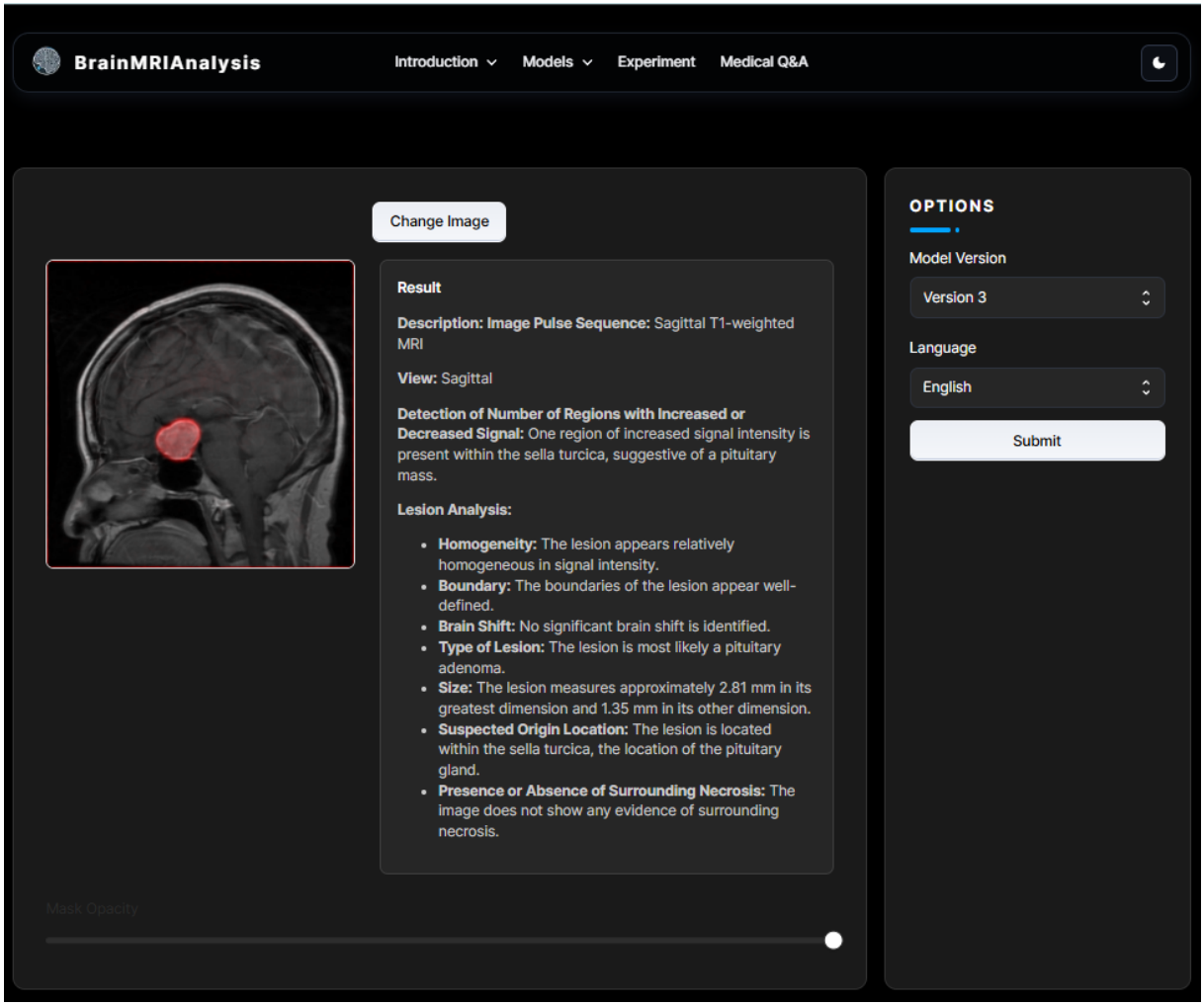


Figure 3.11 User Interface of Brain MRI Report Generation Feature

In the proposed system, users are required to upload brain medical images for automated caption generation. The system offers three distinct versions, each providing varying levels of detail to accommodate different clinical and research needs:

- Version 1: Captions are generated using the MedCapNet model, providing a basic textual description of the medical image.
- Version 2: Extends Version 1 by incorporating additional diagnostic information regarding tumor type, extracted from the GuidedDCNet model. Additionally, this version includes an annotated image highlighting the tumor location, generated using the GuidedSegNet model.
- Version 3: Represents the fully developed MedCapSys, which integrates multilingual support based on user preferences. This version provides a structured caption formatted following medical report standards, similar to physician-generated documentation, along with an annotated image indicating the precise tumor location.

This hierarchical approach ensures flexibility in output complexity, allowing users to select the most appropriate level of detail based on their specific requirements.

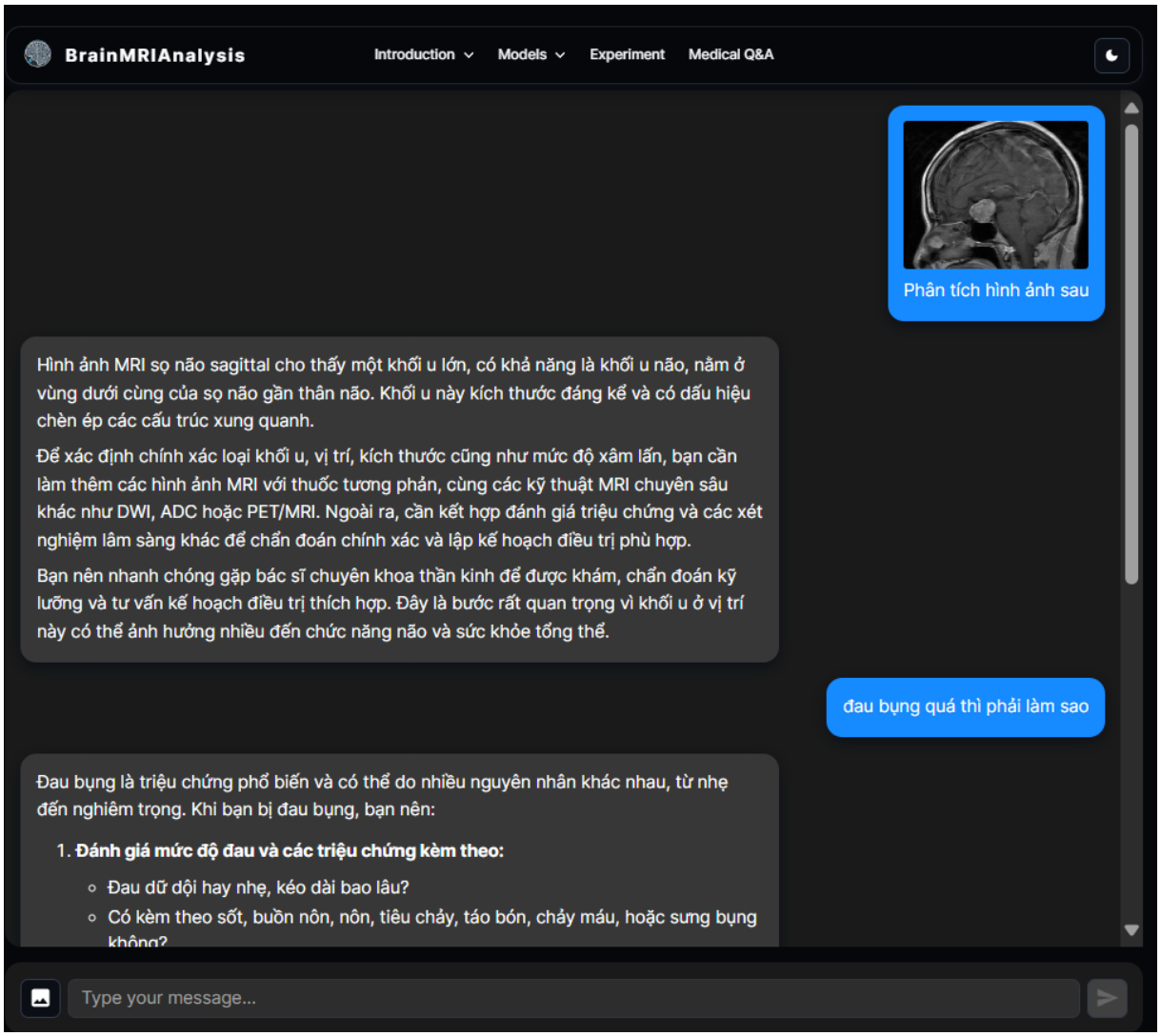


Figure 3.12 User Interface of Medical Question Answering Chatbot

The chatbot feature, illustrated via Figure 3.12, accessible through an intuitive user interface, allows users to engage in natural, real-time conversations with an AI assistant. Through a clean and responsive chat window, users can input questions, requests, or prompts, and receive coherent, context-aware responses instantly. The interface supports a wide range of interactions, from casual conversation to technical assistance, making it versatile for both personal and professional use enabling seamless and interactive communication with the AI.

3.8. Chapter Summary

This chapter presented the implementation details and evaluation results of the proposed MedCapSys system. We began by describing the datasets used for various tasks, image captioning, classification, segmentation, and multimodal training,

selected to reflect real-world medical imaging scenarios. Data preprocessing techniques such as normalization, resizing, and augmentation were applied to ensure consistency and enhance model performance.

We then detailed the implementation of each core component: MedCapNet, GuidedDCNet, GuidedSegDiff, and MedCapSys. Training configurations, model parameters, and framework choices were discussed, along with prompt engineering strategies for the medical QA module. A demonstration system was also introduced, integrating all modules into a unified backend architecture and user-facing web interface.

Evaluation metrics tailored to each task, such as BLEU, ROUGE, accuracy, Dice coefficient, IoU, and human-in-the-loop reviews, were used to assess performance. Experimental results confirmed the effectiveness of each module: MedCapNet generated clinically relevant captions; GuidedDCNet achieved high classification accuracy; GuidedSegDiff produced precise segmentation outputs; and BrainMedQwen generated coherent, domain-specific reports.

Finally, we demonstrated the practical applicability of the integrated system through an interactive clinical decision support platform. Overall, this chapter validated the proposed architecture's performance and usability in supporting medical professionals via intelligent vision-language processing.

CONCLUSION

This project presents MedCapSys, an integrated system for brain medical image analysis, combining advanced deep learning models for image captioning (MedCapNet), lession classification (GuidedDCNet), lession segmentation (GuidedSegNet), and clinical report generation (BrainMedQwen). Through the use of transformer architectures, diffusion models, and multi-scale feature extraction, MedCapSys addresses critical challenges in medical image understanding.

Experimental results confirm the system’s effectiveness in generating accurate captions, improving classification performance, and achieving high segmentation precision. These outcomes demonstrate MedCapSys's potential as a reliable tool to support clinical decision-making and automate radiological reporting.

Key contributions include the design of a unified, task-specific framework and the development of a functional demonstration system with real-world applicability. For future work, we propose extending the system to other imaging modalities and conducting clinical validation to assess its deployment readiness. In additionally, we aim to optimize the system for deployment on resource-constrained environments by applying model compression techniques such as quantization, pruning, and knowledge distillation. These techniques will help reduce computational cost and memory usage while maintaining model performance, thereby enhancing the system’s scalability and usability in real-world clinical settings.

Overall, MedCapSys represents a promising advancement in AI-assisted medical imaging, contributing to more accurate, efficient, and interpretable healthcare solutions.

REFERENCES

[1] J. Ho, A. Jain and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems,* vol. 33, pp. 6840--6851, 2020.

[2] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. Denton, S. K. S. Ghasemipour, B. K. Ayan, S. S. Mahdavi, R. G. Lopes and others, "Photorealistic text-to-image diffusion models with deep language understanding," *arXiv preprint arXiv:2205.11487,* 2022.

[3] R. Rombach, A. Blattmann, D. Lorenz, P. Esser and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[4] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi and D. J. Fleet, "Video diffusion models," *arXiv preprint arXiv:2204.03458,* 2022.

[5] R. Yang, P. Srivastava and S. Mandt, "Diffusion probabilistic modeling for video generation," *arXiv preprint arXiv:2203.09481,* 2022.

[6] T. Hoppe, A. Mehrjou, S. Bauer, D. Nielsen and A. Dittadi, "Diffusion models for video prediction and infilling," *arXiv preprint arXiv:2206.07696,* 2022.

[7] F. Khader, G. Müller-Franzes, S. Tayebi Arasteh and others, "Denoising diffusion probabilistic models for 3D medical image generation," *Scientific Reports,* vol. 13, p. 7303, 2023.

[8] A. De, P. Ghosal, Y. Chen, H. Wang, A. Majumdar and Z. Ren, "D2C: Diffusion-Denoising Models for Few-shot Conditional Generation," *Advances in Neural Information Processing Systems,* vol. 35, 2022.

[9] N. Chen, J. Yue, L. Fang and S. Xia, "SpectralDiff: A Generative Framework for Hyperspectral Image Classification With Diffusion Models," *IEEE Transactions on Geoscience and Remote Sensing,* vol. 61, pp. 1-16, 2023.

[10] Y. Yang, H. Fu, A. I. Aviles-Rivero, Z. Xing and L. Zhu, "DiffMIC-v2: Medical Image Classification via Improved Diffusion Network," *IEEE Transactions on Medical Imaging,* pp. 1-1, 2025.

[11] W. Xiang, H. Yang, D. Huang and Y. Wang, "Denoising Diffusion Autoencoders are Unified Self-supervised Learners," *arXiv preprint arXiv:2303.09769,* 2023.

[12] D. Baranchuk, I. Rubachev, A. Voynov, V. Khrulkov and A. Babenko, "Label-efficient semantic segmentation with diffusion models," *arXiv preprint arXiv:2112.03126,* 2022.

[13] T. Amit, E. Nachmani, T. Shaharabany and L. Wolf, "SegDiff: Image Segmentation with Diffusion Probabilistic Models," *arXiv preprint arXiv:2112.00390,* 2021.

[14] E. Hoogeboom, D. Nielsen, P. Jaini, P. Forre and M. Welling, "Argmax flows and multinomial diffusion: Learning categorical distributions," *arXiv preprint arXiv:2102.05379,* 2021.

[15] J. a. J. D. D. Austin, J. Ho, D. Tarlow and R. van den Berg, "Structured Denoising Diffusion Models in Discrete State-Spaces," *arXiv preprint arXiv:2107.03006,* 2021.

[16] G. Batzolis, J. Stanczuk, C.-B. Schonlieb and C. Etmann, "Conditional image generation with score-based diffusion models," *arXiv preprint arXiv:2111.13606,* 2022.

[17] A. Blattmann, R. Rombach, K. Oktay and B. Ommer, "Retrieval-augmented diffusion models," *arXiv preprint arXiv:2204.11824,* 2022.

[18] S. Kublik and S. Saboo, GPT-3, O'Reilly Media, Inc, 2022.

[19] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of machine learning research,* vol. 21, pp. 1--67, 2020.

[20] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann and others, "Palm: Scaling language modeling with pathways," *Journal of Machine Learning Research,* vol. 24, pp. 1-113, 2023.

[21] J. Xu, Z. Guo, J. He, H. Hu, T. He, S. Bai, K. Chen, J. Wang, Y. Fan, K. Dang and others, "Qwen2.5 Technical Report," *arXiv preprint arXiv:2412.15115,* 2025.

[22] S. Bai, K. Chen, X. Liu, J. Wang, W. Ge, S. Song, K. Dang, P. Wang, S. Wang, J. Tang and others, "Qwen2.5-vl technical report," *arXiv preprint arXiv:2502.13923,* 2024.

[23] A. Yang, A. Li, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Gao, C. Huang, C. Lv and others, "Qwen3 technical report," *arXiv preprint arXiv:2505.09388,*

2025.

[24] D. Alexey, B. Lucas, K. Alexander, W. Dirk, Z. Xiaohua, U. Thomas, D. Mostafa, M. Matthias, H. Georg, G. Sylvain, U. Jakob and H. Neil, "An Image is Worth 16x16 Words: Transformers for Image Recognition," in *9th International Conference on Learning Representations*, Austria, 2021.

[25] X. Pan, T. Ye, D. Han, S. Song and G. Huang, "Contrastive language-image pre-training with knowledge graphs," *Advances in Neural Information Processing Systems,* vol. 35, pp. 22895--22910, 2022.

[26] C. Jia, Y. Yang, Y. Xia, Y.-T. Chen, Z. Parekh, H. Pham, Q. Le, Y.-H. Sung, Z. Li and T. Duerig, "Scaling up visual and vision-language representation learning with noisy text supervision," in *International conference on machine learning*, 2021.

[27] J. Li, D. Li, C. Xiong and S. Hoi, "Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation," in *International conference on machine learning*, 2022.

[28] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds and others, "Flamingo: a visual language model for few-shot learning," in *Advances in neural information processing systems*, 2022.

[29] D. Zhu, J. Chen, X. Shen, X. Li and M. Elhoseiny, "Minigpt-4: Enhancing vision-language understanding with advanced large language models," *arXiv preprint arXiv:2304.10592,* 2023.

[30] S. Bannur, S. Hyland, Q. Liu, F. Perez-Garcia, M. Ilse, D. C. Castro, B. Boecking, H. Sharma, K. Bouzid, A. Thieme and others, "Learning to exploit temporal structure for biomedical vision-language processing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[31] K. Singhal, T. Tu, J. Gottweis, R. Sayres, E. Wulczyn, M. Amin, L. Hou, K. Clark, S. R. Pfohl, H. Cole-Lewis and others, "Toward expert-level medical question answering with large language models," *Nature Medicine,* pp. 1--8, 2025.

[32] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat and others, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774,* 2023.

[33] Anthropic, "Claude 3 Model Card Addendum," March 2024. [Online]. Available: https://claude.ai.

[34] G. Li, H. A. A. K. Hammoud, H. Itani, D. Khizbullin and B. Ghanem, "CAMEL: Communicative Agents for "Mind" Exploration of Large Language Model Society," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[35] Q. Wu, G. Bansal, J. Zhang, Y. Wu, B. Li, E. Zhu, L. Jiang, X. Zhang, S. Zhang, J. Liu and others, "Autogen: Enabling next-gen llm applications via multi-agent conversation," *arXiv preprint arXiv:2308.08155,* 2023.

[36] S. Hong, X. Zheng, J. Chen, Y. Cheng, J. Wang, C. Zhang, Z. Wang, S. K. S. Yau, Z. Lin, L. Zhou and others, "Metagpt: Meta programming for multi-agent collaborative framework," *arXiv preprint arXiv:2308.00352,* vol. 3, p. 6, 2023.

[37] S. Mannor, D. Peleg and R. Rubinstein, "The cross entropy method for classification," in *Proceedings of the 22nd international conference on Machine learning*, 2005.

[38] A. Mammone, M. Turchi and N. Cristianini, "Support vector machines," *Wiley Interdisciplinary Reviews: Computational Statistics,* vol. 1, pp. 283--289, 2009.

[39] L. Breiman, "Random forests," *Machine Learning,* vol. 45, pp. 5--32, 2001.

[40] B. Scholkopf and A. J. Smola, Learning with kernels: support vector machines, regularization, optimization, and beyond, MIT press, 2002.

[41] T. G. Dietterich, "Ensemble methods in machine learning," *Multiple Classifier Systems,* vol. 1857, pp. 1--15, 2000.

[42] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE,* vol. 86, pp. 2278--2324, 1998.

[43] A. Krizhevsky, I. Sutskever and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012.

[44] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[45] G. Huang, Z. Liu, L. Van Der Maaten and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[46] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning*, 2019.

[47] M. Tan and Q. V. Le, "EfficientNet V3: Self-scaled Hierarchical Feature Scaling," *arXiv preprint arXiv:2305.11547,* 2023.

[48] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017.

[49] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

[50] S. Hussein, T. AbdelAziz, B. Kara and M. Abdelreheem, "Dilated Neighborhood Attention Transformer," *arXiv preprint arXiv:2304.09607,* 2023.

[51] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," in *Advances in Neural Information Processing Systems*, 2021.

[52] J. Ho and T. Salimans, "Classifier-free diffusion guidance," *arXiv preprint arXiv:2207.12598,* 2022.

[53] S. J. Rennie, E. Marcheret, Y. Mroueh, J. Ross and V. Goel, "Self-critical sequence training for image captioning," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017.

[54] T. Yao, Y. Pan, Y. Li and T. Mei, "Exploring visual relationship for image captioning," in *Proceedings of the European conference on computer vision (ECCV)*, 2018.

[55] Y. Pan, T. Yao, Y. Li and T. Mei, "X-linear attention networks for image captioning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020.

[56] M. Cornia, M. Stefanini, L. Baraldi and R. Cucchiara, "Meshed-memory transformer for image captioning," 2020.

[57] J. Ruckert, A. Ben Abacha, A. G. Seco de Herrera, L. Bloch, R. Brungel, A. Idrissi-Yaghir, H. Schafer, H. Muller and C. M. Friedrich, "Overview of ImageCLEFmedical 2023--caption prediction and concept detection," in *Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2023)*, 2023.

[58] A. Nicolson, J. Dowling and B. Koopman, "A Concise Model for Medical Image

Captioning.," in *CLEF*, 2023.

[59] B. Yang, Y. Yu, Y. Zou and T. Zhang, "PCLmed: Champion Solution for ImageCLEFmedical 2024 Caption Prediction Challenge via Medical Vision-Language Foundation Models," in *CLEF2024 Working Notes, CEUR Workshop Proceedings*, 2024.

[60] S. Ram, S. Vinoth, R. N. Gopalakrishnan, A. A. Balakumar, L. Kalinathan and T. A. J. Velankanni, "Leveraging Diverse CNN Architectures for Medical Image Captioning: DenseNet-121, MobileNetV2, and ResNet-50 in ImageCLEF 2024," in *2024*, CLEF2024 Working Notes, CEUR Workshop Proceedings.

[61] J. Chen, Y. Lu, Q. Yu, X. Luo, E. Adeli, Y. Wang, L. Lu, A. L. Yuille and Y. Zhou, "Transunet: Transformers make strong encoders for medical image segmentation," *arXiv preprint arXiv:2102.04306,* 2021.

[62] H. Cao, Y. Wang, J. Chen, D. Jiang, X. Zhang, Q. Tian and M. Wang, "Swin-unet: Unet-like pure transformer for medical image segmentation," *European conference on computer vision,* pp. 205--218, 2022.

[63] Y. Tang, D. Yang, W. Li, H. R. Roth, B. Landman, D. Xu, V. Nath and A. Hatamizadeh, "Self-supervised pre-training of swin transformers for 3d medical image analysis," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022.

[64] A. Lin, B. Chen, J. Xu, Z. Zhang, G. Lu and D. Zhang, "Ds-transunet: Dual swin transformer u-net for medical image segmentation," *IEEE Transactions on Instrumentation and Measurement,* vol. 71, pp. 1--15, 2022.

[65] X. Zhai, A. Kolesnikov, N. Houlsby and L. Beyer, "Scaling vision transformers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022.

[66] X. Xie, H. Liu, W. Hou and H. Huang, "A brief survey of vector databases," in *2023 9th International Conference on Big Data and Information Analytics (BigDIA)*, 2023.

[67] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman and A. Y. Wu, "An optimal algorithm for approximate nearest neighbor searching fixed dimensions," *Journal of the ACM (JACM),* vol. 45, pp. 891--923, 1998.

[68] M. Wang, H. Wu, X. Ke, Y. Gao, Y. Zhu and W. Zhou, "Accelerating Graph Indexing for ANNS on Modern CPUs," *arXiv preprint arXiv:2502.18113,* 2025.

[69] J. Mohoney, A. Pacaci, S. R. Chowdhury, U. F. Minhas, J. Pound, C. Renggli,

N. Reyhani, I. F. Ilyas, T. Rekatsinas and S. Venkataraman, "Incremental IVF Index Maintenance for Streaming Vector Search," *arXiv preprint arXiv:2411.00970,* 2024.

[70] Z. André, V. Andrey and others, "Qdrant," 2021. [Online]. Available: https://qdrant.tech/.

[71] I. Pinecone Systems, "Pinecone," [Online]. Available: https://www.pinecone.io.

[72] B. Weaviate, "Weaviate," [Online]. Available: https://weaviate.io.

[73] M. Douze, A. Guzhva, C. Deng, J. Johnson, G. Szilvasy, P.-E. Mazaré, M. Lomeli, L. Hosseini and H. Jégou, "The Faiss library," *arXiv,* 2024.

[74] Chroma, "ChromaDB," [Online]. Available: https://www.trychroma.com.

[75] J. Ji, Y. Luo, X. Sun, F. Chen, G. Luo, Y. Wu, Y. Gao and R. Ji, "Improving image captioning by leveraging intra-and inter-layer global representation in transformer network," in *Proceedings of the AAAI conference on artificial intelligence*, 2021.

[76] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *Advances in neural information processing systems,* vol. 30, 2017.

[77] P. Rodriguez, J. M. Gonfaus, G. Cucurull, F. XavierRoca and J. Gonzalez, "Attend and rectify: a gated attention mechanism for fine-grained recovery," in *Proceedings of the European conference on computer vision (ECCV)*, 2018.

[78] P. Keller, M. Dawood, B. S. Chohan and others, "HistoKernel: Whole slide image level Maximum Mean Discrepancy kernels for pan-cancer predictive modelling," *Medical Image Analysis,* p. 103491, 2025.

[79] Y. Sun and J. Fan, "Mmd graph kernel: Effective metric learning for graphs via maximum mean discrepancy," in *The Twelfth International Conference on Learning Representations*, 2024.

[80] S. Zhao, J. Song and S. Ermon, "Infovae: Balancing learning and inference in variational autoencoders," in *Proceedings of the aaai conference on artificial intelligence*, 2019.

[81] A. Sinha, J. Song, C. Meng and S. Ermon, "D2c: Diffusion-decoding models for few-shot conditional generation," *Advances in Neural Information Processing Systems,* vol. 34, pp. 12533--12548, 2021.

[82] H. Huang, R. He, Z. Sun, T. Tan and others, "Introvae: Introspective variational autoencoders for photographic image synthesis," *Advances in neural information*

processing systems, vol. 31, 2018.

[83] C. Spence, "Crossmodal spatial attention," *Annals of the New York Academy of Sciences,* vol. 1191, pp. 182--200, 2010.

[84] J. Y. Cheng, F. Chen, M. T. Alley, J. M. Pauly and S. S. Vasanawala, "Highly scalable image reconstruction using deep neural networks with bandpass filtering," *arXiv preprint arXiv:1805.03300,* 2018.

[85] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM,* vol. 65, pp. 99--106, 2021.

[86] J. Ruckert, L. Bloch, R. Brungel, A. Idrissi-Yaghir, H. Schafer, C. S. Schmidt, S. Koitka, O. Pelka, A. B. Abacha, A. G. Seco de Herrera and others, "Rocov2: Radiology objects in context version 2, an updated multimodal image dataset," *Scientific Data,* vol. 11, p. 688, 2024.

[87] C. Jun, *Brain tumor dataset,* 2017.

[88] B. Sartaj, K. Ankita, B. Prajakta, D. Sameer and K. Swati, *Brain Tumor Classification (MRI),* Kaggle, 2020.

[89] H. Ahmed, *Br35H :: Brain Tumor Detection 2020,* IEEE Dataport, 2020.

[90] D. Zhang, G. Huang, Q. Zhang, J. Han, J. Han, Y. Wang and Y. Yu, "Exploring task structure for brain tumor segmentation from multi-modality MR images," *IEEE Transactions on Image Processing,* vol. 29, pp. 9032--9043, 2020.

[91] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Radle, C. Rolland, L. Gustafson and others, "Sam 2: Segment anything in images and videos," *arXiv preprint arXiv:2408.00714,* 2024.

[92] P. Yang, W. Song, X. Zhao, R. Zheng and L. Qingge, "An improved Otsu threshold segmentation algorithm," *International Journal of Computational Science and Engineering,* vol. 22, pp. 146--153, 2020.

[93] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-international conference on neural networks*, 1995.

[94] S. M. Pizer, E. P. Amburn, J. D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. ter Haar Romeny, J. B. Zimmerman and K. Zuiderveld, "Adaptive histogram equalization and its variations," *Computer vision, graphics, and image processing,* vol. 39, pp. 355--368, 1987.

[95] S. Rahman, M. M. Rahman, M. Abdullah-Al-Wadud, G. D. Al-Quaderi and M.

Shoyaib, "An adaptive gamma correction for image enhancement," *EURASIP Journal on Image and Video Processing,* pp. 1--13, 2016.

[96] E. A. Robinson and S. Treitel, "Principles of digital Wiener filtering," *Geophysical Prospecting,* vol. 15, pp. 311--332, 1967.

[97] Z. Zhang, "Improved adam optimizer for deep neural networks," in *2018 IEEE/ACM 26th international symposium on quality of service (IWQoS)*, 2018.

[98] S. K. Warfield, K. H. Zou and W. M. Wells, "Simultaneous truth and performance level estimation (STAPLE): an algorithm for the validation of image segmentation," *IEEE transactions on medical imagin,* vol. 23, pp. 903--921, 2004.

[99] R. Rafailov, A. Sharma, E. Mitchell, C. D. Manning, S. Ermon and C. Finn, "Direct preference optimization: Your language model is secretly a reward model," *Advances in Neural Information Processing Systems,* vol. 36, pp. 53728--53741, 2023.

[100] R. Islam and O. M. Moushi, "Gpt-4o: The cutting-edge advancement in multimodal llm," *Authorea Preprints,* 2024.

[101] H. Lee, S. Phatale, H. Mansoor, K. R. Lu, T. Mesnard, J. Ferret, C. Bishop, E. Hall, V. Carbune and A. Rastogi, "Rlaif: Scaling reinforcement learning from human feedback with ai feedback," 2023.

[102] X. Amatriain, "Prompt design and engineering: Introduction and advanced methods," *arXiv preprint arXiv:2401.14423,* 2024.

[103] A. AGI, "Agno," 2025. [Online]. Available: https://docs.agno.com/.

[104] I. Docker and others, "Docker," 2020. [Online]. Available: https://www.docker.com.

[105] R. Soni, Nginx, Springer, 2016.

[106] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger and Y. Artzi, "Bertscore: Evaluating text generation with bert," *arXiv preprint arXiv:1904.09675,* 2019.

[107] W. Zhou, Z. Ye, Y. Yang, S. Wang, H. Huang, R. Wang and D. Yang, "Transferring Pre-Trained Large Language-Image Model for Medical Image Captioning.," in *CLEF (Working Notes)*, 2023.

[108] P. Kaliosis, G. Moschovis, F. Charalampakos, J. Pavlopoulos and I. Androutsopoulos, "AUEB NLP Group at ImageCLEFmedical Caption 2023.," in *CLEF (Working Notes)*, 2023.

[109] P. Xia, L. Zhang and F. Li, "Learning similarity with cosine similarity ensemble}," *Information sciences,* vol. 307, pp. 39--52, 2015.

[110] K. L. Elmore and M. B. Richman, "Euclidean distance as a similarity metric for principal component analysis," *Monthly weather review,* vol. 129, pp. 540--549, 2001.

APPENDIX A: PROMPTS

A.1. System prompt for MedCapSys

```
# Role:
You are a a knowledgeable and experienced radiologist doctor responsible for writing Brain MRI Report.

# Mission:
Provide a detailed report containing information in the following order:
- Image pulse sequence
- View
- Detection of number of regions with increased or decreased signal
- Lession Analysis:
    - Whether this region is homogeneous or heterogeneous
    - Whether the boundaries are well-defined or ill-defined
    - Type of brain shift (if present)
    - Type of lession
    - Size of lession
    - Suspected origin location
    - Presence or absence of surrounding necrosis

# Important
- Ensure no details are fabricated.
- Refer to the following image for the necessary details.
- Ensure the description closely follows the writing style of a radiologist’s report using precise and professional medical terminology.
- Response in the language that user requires.
```

A.2. System prompt for Coordinator Agent

```
# Role:
Act as a knowledgeable and experienced doctor to answer medical-related question from the patient.

# Mission:
- When the question is related to radiology, delegate to `Radiologist Agent`
- Delegate to `General Practitioner Agent` to collect general answer for the patient

# Important
- Ensure no details are fabricated.
- Refer to the image (if present) for the necessary details.
```

```
- Response in the language that user requires.
```

A.3. System prompt for Radiologist Agent

```
# Role:
Act as a knowledgeable and experienced radiologist doctor to analyze medical image.

# Mission:
- Call `detect_brain_lession_tool` to detect the location of the lession
- Call `classify_tumor_type_tool` to get the type of tumor
- Call `segment_lession_region_tool` to get the location and size of the lession
- Call `get_view_tool` to get the view of the uploaded image
- Call `get_pulse_sequence_tool` to get the pulse sequence of the uploaded image
- Call `generate_report_tool` to get the radiologist report of the uploaded image

# Important
- Ensure no details are fabricated.
- Refer to the image (if present) for the necessary details.
- Response in the language that user requires.
```

A.4. System prompt for General Practitioner Agent

```
# Role:
Act as a knowledgeable and experienced general practitioner doctor to answer medical-related question from the patient.

# Mission:
- Answering questions about symptoms, common illnesses, preventive care, basic treatments, and when to seek medical attention.
- Explaining medical concepts in clear, simple language suitable for non-experts.
- Giving general guidance based on evidence-based medicine and current medical guidelines.
- Avoiding definitive diagnoses or prescriptions unless the issue is simple, non-urgent, and can be addressed with over-the-counter (OTC) advice.
- Emphasizing the importance of seeing a qualified healthcare provider for serious or persistent concerns.
- Encouraging safe, responsible health decisions, especially when symptoms could indicate an emergency.
```

```
# Answer format:
- Summary of the issue (if applicable)
- Clear, helpful information (about symptoms, possible causes, general advice, home care)
- Warnings or referral guidance (when to see a doctor or go to the hospital)
- (Optional) Refer to reliable sources such as WHO, CDC, Mayo Clinic, UpToDate, or NHS

# Important
- Ensure no details are fabricated.
- Refer to the image (if present) for the necessary details.
- Response in the language that user requires.
```

APPENDIX B: TIMELINE

No	Stage	Details	Week											
			1	2	3	4	5	6	7	8	9	10	11	12
1	Research & Planning	Conduct an overview study of medical image analysis and related models.	X											
		Collect data and prepare the training dataset.	X	X										
		Define technical requirements, models, and technologies to be used.	X	X										
		Develop a project implementation plan.	X											
2	Build & train modules	Develop & train MedCapNet			X	X								
		Develop & train GuidedDCNet				X	X							
		Develop & train GuidedSegNet					X	X						
		Fine-tune BrainMedQwen						X	X					
3	Integrate the MedCapSys	Integrate the modules into a unified system.								X				
		Build a pipeline for input and output data processing.								X				
		Test and optimize system performance.									X			
4	Testing & Evaluation	Evaluate system performance on the test dataset.										X		
		Compare the results with existing methods.										X		
		Fine-tune the model to improve accuracy.											X	
		Develop a simple website to test the product.											X	
5	Finalize the report & Prepare the demo	Write the summary report and user manual.											X	X
		Prepare the demo and perform final testing.												X

APPENDIX C: SUBMISSION RESULTS

C.1. First Prize in Student Scientific Research Conference, The University of Danang - University of Science and Technology, Academic Year 2024–2025

C.2. Published in the 13th International Symposium on Information and Communication Technology (SOICT 2024)

Check for updates

MedCapNet: A Novel Approach to Medical Image Captioning

Phan Minh Nhat, Dinh Minh Toan, Ho Quoc Thien Anh, Nguyen Bao Thuc Nhi, and Nguyen Van Hieu(✉)

The University of Danang - University of Science and Technology, Danang, Vietnam
nvhieuqt@dut.udn.vn

Abstract. Medical image captioning is crucial for automating the generation of accurate textual descriptions for medical images. This paper introduces MedCapNet, a novel encoder-decoder architecture designed to bridge the gap between visual and textual modalities in medical imaging. The model incorporates a Swin Transformer and Enhancement Encoder, allowing for the efficient extraction and refinement of both patch-level and global-level features from medical images. A Transformer block with a Fusion Module is utilized by the decoder to seamlessly integrate visual and linguistic information. A key innovation is Dual-Scale Masked Multi-Head Self-Attention, which enhances the model's ability to effectively capture long-range dependencies and fine-grained details. Our model was evaluated on ROCO v2, achieving state-of-the-art performance with scores of 0.647, 0.239, and 0.094 for BERTScore, CIDEr, and METEOR, respectively.

Keywords: Medical Image Captioning · MedCapNet · Enhancement Encoder · Dual-Scale Masked Multi-Head Self-Attention