University of Bonn

Master's thesis for obtaining the academic degree
„Master of Science (M.Sc.)“

Detection of Focal Cortical Dysplasia Type II Using Text Descriptions

Author:
German Mikhelson

First Examiner:
Prof. Dr. Thomas Schultz

Second Examiner:
Dr. Nils Goerke

Advisor:
Annalena Lange

Submitted: January 15, 2026

Declaration of Authorship

I declare that the work presented here is original and the result of my own investigations. Formulations and ideas taken from other sources are cited as such. It has not been submitted, either in part or whole, for a degree at this or any other university. During the preparation of the text, I used AI-based language models as writing assistance tools.

Location, Date

Signature

Abstract

Numerous methods have been developed for tumor detection in medical images, demonstrating the effectiveness of deep learning across a wide range of diagnostic tasks. However, the detection of Focal Cortical Dysplasia (FCD) is considerably more challenging. In contrast to tumors, these lesions do not exhibit progressive growth, often lack clear visual boundaries, and are characterized by subtle and heterogeneous imaging patterns. As a result, the sensitivity of existing automated detection methods remains relatively low, even for state-of-the-art approaches such as MELD, which currently represents one of the best-performing frameworks in this domain. In addition, publicly available annotated datasets for FCD are highly limited.

To address these limitations, this thesis proposes a novel multimodal segmentation framework that combines surface-based visual features with text-based inputs that guide the segmentation process. Experimental results showed that the proposed approach improves both detection accuracy and sensitivity compared to the MELD baseline. To further illustrate the practical utility of the proposed method, a web-based interface was implemented, enabling access to all model variants and their corresponding predictions.

Contents

1 Introduction

This chapter first highlights the importance of epilepsy detection and discusses the limitations of existing methods. It outlines the main objectives of the thesis and its research contributions. Finally, the chapter concludes with an overview of the thesis structure.

1.1 Motivation

Automatic detection of tumors using medical imaging has been widely studied across different internal organs. Recent advances in deep learning and transformer-based approaches have led to impressive results in tumor detection tasks, including lung cancer [1], brain tumors [2], kidney tumors on Computed Tomography (CT) imaging [3], and breast cancer [4]. These studies demonstrate the potential of modern computer vision techniques to achieve clinically significant results in various diagnostic tasks.

However, the detection of epileptogenic lesions, in particular FCD, remains substantially more challenging.

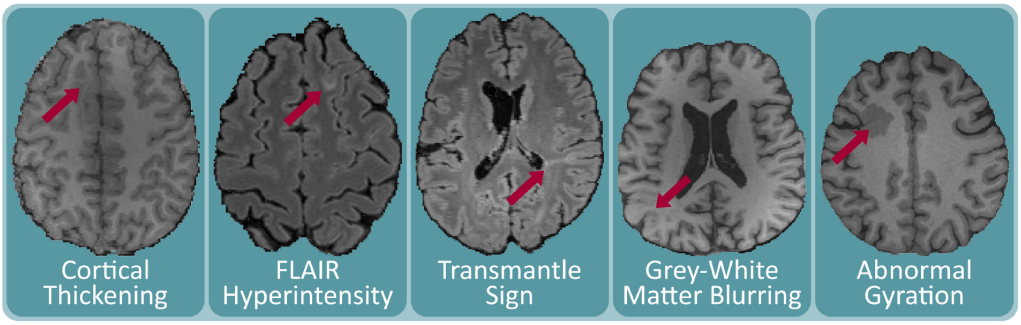

Figure 1.1: Typical features of focal cortical dysplasias (FCDs) visible in Magnetic Resonance Imaging (MRI) [5]

Unlike tumors, which often show progressive growth and well–defined boundaries, epileptogenic lesions typically remain stable in size and present with subtle imaging features, as illustrated in Fig. 1.1, complicating their identification even for trained neuroradiologists [5]. Earlier approaches [6–8] to automated FCD detection mainly used volumetric convolutional neural networks trained on 3D Magnetic Resonance Imaging (MRI) data, such as T1-weighted and FLAIR images. However, these methods were usually developed using relatively small datasets, often including only a few dozen to a few hundred patients, which limited their reliability and ability to generalize to unseen data. In addition, volumetric CNN-based approaches do not directly model the cortical surface or its shape, even though FCD-related abnormalities mainly affect the cortex. In contrast, the Multicentre Epilepsy Lesion Detection (MELD) project [9] uses a surface-based representation of the cortex and applies graph neural networks to model cortical structure and local relationships, allowing the use of large multi-center datasets and addressing important limitations of earlier volumetric methods. Although this represents

one of the most advanced solutions to date, performance remains limited, underscoring the continued complexity of the detection task.

At the same time, recent progress in text-guided and multimodal learning suggests that combining visual information with textual descriptions may help models focus on clinically relevant anatomical regions. In the context of epileptogenic lesion detection, such textual information can reflect prior clinical knowledge derived from examinations other than MRI, such as EEG findings or seizure semiology. For tasks with limited training data, textual annotations can therefore serve as an additional source of structure by guiding the model toward regions that are clinically suspected, even when MRI findings are subtle. This motivates investigating whether integrating such information, even in a coarse form (e.g., hemisphere or lobe labels), can help compensate for the scarcity of annotated datasets and improve detection performance.

In this thesis, we build upon the surface-based feature maps produced by the pretrained MELD model, which represent the cortical surface as a sparse graph. Given this graph-based representation, a natural next step is to investigate whether additional Graph Neural Network (GNN) layers can further improve lesion detection by aggregating information across larger neighborhoods on the cortical surface. Understanding the benefits and limitations of such graph-based aggregation is particularly important in sparse settings, where capturing long-range context may directly influence lesion detection.

1.2 Aims of the Thesis

The main objective of this thesis is to investigate how the integration of textual and visual information can improve the accuracy of epileptogenic lesion detection. Several research questions were formulated:

1. Do incorporating textual descriptions from anatomical atlases influence the accuracy of lesion segmentation?

2. Can competitive performance be achieved using only partial atlas information (e.g., hemisphere or lobe labels)?

3. What is the impact of integrating additional GNN blocks with MELD-based representations on segmentation quality?

We systematically evaluate the impact of different types of textual information, as well as alternative strategies for combining visual features, including the integration of additional GNN layers. We further analyze how these design choices influence segmentation performance, demonstrating that the joint integration of textual and visual features can enhance both the accuracy and the sensitivity of epileptogenic lesion detection.

1.3 Contributions

The main contributions of this thesis can be summarized as follows:

1. Introduced a state-of-the-art multimodal model for epileptogenic lesion detection that integrates surface-based visual features with textual information.

2. Conducted a systematic evaluation of design choices related to graph-based feature aggregation, including different configurations of GNN blocks.

3. Assessed the effect of multiple types of textual descriptions (full atlas annotations, hemisphere and lobe labels, etc.) on model accuracy and sensitivity.

4. Delivered comprehensive documentation and usage guidelines for the implemented methods.

5. Implemented a user-friendly interface that supports different pretrained models and enables interactive visualization of predictions.

1.4 Thesis structure

The structure of this thesis is organized as follows:

1. Chapter 2 reviews prior work on FCD detection and text-guided medical image segmentation, focusing on architectural choices, fusion strategies, and open limitations that motivate our approach.

2. Chapter 3 presents the proposed multimodal segmentation framework for epileptogenic lesion detection, detailing the surface-based visual features (from MELD preprocessing), the GNN block, LLM textual embeddings, and their integration within the GuideDecoder (with HexUnpool and SpiralConv adaptations).

3. Chapter 4 describes the Bonn and MELD datasets, FreeSurfer-based feature extraction and harmonization, the data-augmentation pipeline, and the variants of atlas-derived textual descriptions used in our study.

4. Chapter 5 details the experimental setup (losses, metrics, and implementation specifics) and reports the main results: baseline comparisons against MELD, the effect of linking different numbers of MELD feature stages to the GNN block, as well as experiments with mixed-text settings.

5. Chapter 6 discusses the findings, emphasizing trade-offs between sensitivity and precision, limitations of the current study, and implications for clinical applicability.

6. Finally, Chapter 7 summarizes key insights, outlines the limitations, and proposes directions for future research.

2 Related Works

This section firstly reviews the Transformer architecture, which is a fundamental component of modern vision models and large language models (LLMs). Then it introduces graph neural networks (GNNs) and LLMs, followed by an overview of existing approaches to FCD detection and text-guided medical image segmentation for tumor detection tasks, with a focus on their fusion strategies, and limitations.

2.1 Transformer Architecture

The transformer architecture, introduced in the seminal work by Vaswani et al. [10], is based entirely on attention mechanisms and does not rely on recurrence or convolution. Its core component is the ***self-attention layer***, which models long-range interactions between elements of a sequence. Given input embeddings X, the model computes query, key, and value matrices $Q = XW^Q$, $K = XW^K$, and $V = XW^V$, where W^Q, W^K, and W^V are learnable projections respectively. The self-attention operation is defined as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V, \tag{2.1}$$

where d_k is the dimensionality of the key vectors.

To enhance representational capacity, transformers employ ***multi-head attention***, which performs several attention operations in parallel:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)\, W^O, \tag{2.2}$$

where each head computes self-attention using its own learnable projections:

$$\text{head}_i = \text{Attention}(QW_i^Q,\, KW_i^K,\, VW_i^V), \tag{2.3}$$

with parameter matrices $W_i^Q \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^K \in \mathbb{R}^{d_{\text{model}} \times d_k}$, and $W_i^V \in \mathbb{R}^{d_{\text{model}} \times d_v}$. The output projection

$$W^O \in \mathbb{R}^{hd_v \times d_{\text{model}}}, \tag{2.4}$$

mixes information across all h heads and restores the dimensionality back to d_{model}.

Originally developed for natural language processing, transformer-based models have since been adapted to computer vision (e.g., ViT [11], Swin Transformer [12]) and to a variety of multimodal learning settings. In text-guided medical image segmentation, transformers are commonly used either as text encoders or as fusion modules, where cross-attention integrates semantic cues from textual descriptions with spatial image representations. This capacity to incorporate global contextual information makes transformers particularly suitable for multi-modal medical tasks, although their performance often depends on the availability of sufficiently

large and diverse training datasets.

2.2 Graph Neural Networks

Graph Neural Networks (GNNs) [13] provide a principled framework for learning on non-Euclidean data such as meshes, surfaces, and general irregular structures. In contrast to convolutional architectures that operate on grids, GNNs propagate information along graph edges, enabling feature integration across anatomically or structurally meaningful neighborhoods. Most modern GNNs follow the *message-passing* formulation [14], where node representations are iteratively updated by aggregating features from adjacent nodes. For a node v with neighborhood $\mathcal{N}(v)$ and current features h_v, the generic message-passing layer is expressed as:

$$m_v = \mathrm{AGG}\big(\{\, \phi(h_v, h_u, e_{uv}) : u \in \mathcal{N}(v) \,\}\big), \tag{2.5}$$

$$h'_v = \gamma(h_v, m_v), \tag{2.6}$$

where ϕ computes messages from neighbors, AGG is a permutation-invariant aggregation function (e.g., mean, sum, max), and γ is an update function (often implemented as an MLP). Different GNN architectures instantiate these components in distinct ways. For example, Graph Convolutional Networks (GCN) [15] use a normalized mean aggregator, while GraphSAGE [16] generalizes aggregation through mean, pooling, or LSTM-based operators, enabling inductive learning on unseen graphs.

Hierarchical GNN architectures extend this idea to multi-resolution settings, analogous to U-Net models [17] in Euclidean domains and graph U-Net architectures [18]. They downsample and upsample graphs using pooling and unpooling operators, which enable coarse-to-fine feature extraction on complex surfaces. This paradigm is particularly relevant for cortical-surface analysis, where the brain can be represented as a multi-resolution mesh.

Building on these principles, the MELD project [9] applies a GNN-U-Net architecture to multi-resolution cortical graphs and demonstrates that graph-based convolutions are effective for detecting subtle morphological abnormalities associated with FCD.

In this thesis, we build upon these principles to integrate cortical surface features with textual clinical information, enabling multimodal segmentation within a unified GNN-based framework.

2.3 Large Language Models

LLMs are typically based on transformer architectures and are trained on large text corpora to learn contextual semantic representations.

Given a sequence of token embeddings $X = (x_1, \ldots, x_T)$, an LLM encoder produces hidden states

$$H = (h_1, \ldots, h_T), \tag{2.7}$$

where each vector h_i is a context-dependent representation of token x_i. This means that h_i is computed not only from the embedding of x_i itself but also from information coming from other tokens in the sequence, which is achieved by the self-attention mechanism.

LLMs are typically trained using one of two objectives. In *causal* language modeling (used in

GPT-style models), the model predicts the next token based on the previously observed ones:

$$p(x_{i+1} \mid x_1, \ldots, x_i) = \text{softmax}(W^O h_i). \tag{2.8}$$

In *masked* language modeling (MLM), employed by BERT-style encoders, randomly selected tokens are masked and reconstructed from the surrounding context:

$$p(x_i \mid X_{\setminus i}) = \text{softmax}(W^O h_i), \tag{2.9}$$

where $X_{\setminus i}$ denotes the input sequence with token x_i masked.

LLMs trained with these objectives produce rich, high-level semantic embeddings that capture lexical, syntactic, and domain-specific patterns in text [19]. In medical imaging, domain-adapted LLMs such as BioBERT [20] provide contextual text embeddings that can guide segmentation models through cross-attention or contrastive alignment [21], improving robustness and interpretability in multimodal medical applications.

2.4 State-of-the-Art Methods

Building on early vision-based approaches in medical imaging, initial automated methods for FCD detection relied on convolutional neural networks operating directly on MRI data. Dev et al. [7] proposed a fully convolutional neural network for the automatic detection and localization of focal cortical dysplasia (FCD) using volumetric MRI. Their approach performs voxel-wise segmentation on multimodal MRI data, enabling direct lesion identification without manual feature engineering. However, the method was trained on a relatively small, single-center dataset, which may limit its robustness and generalizability across different populations and imaging protocols.

Along similar lines, House et al. [8] introduced a clinically integrated 3D convolutional neural network for the automated detection and segmentation of FCD lesions from routine T1-weighted and FLAIR MRI sequences. A key contribution of this work is its prospective validation within a real-world clinical workflow. Nevertheless, the model exhibits substantially lower detection performance for subtle or non-sclerotic FCD subtypes compared to more pronounced Taylor-type FCD, suggesting a bias toward more conspicuous lesion patterns and limited generalization across the full pathological spectrum.

A more recent and substantial advancement is represented by the MELD Graph model [9], which departs from volumetric representations by modeling the cerebral cortex as a multi-resolution graph and applying a GNN-U-Net architecture for lesion segmentation. By explicitly capturing cortical surface morphology, the MELD approach achieves high sensitivity and specificity, supported by saliency-based peak analysis and calibration assessment using the expected calibration error. Despite these strengths, the model has not been evaluated on patients with multiple FCD lesions, does not incorporate cross-attentional mechanisms for richer feature integration, and does not leverage textual clinical information across FCD subtypes.

The integration of multimodal data, as demonstrated by models like Ariadne's Thread [22], offers to close this gap. Its approach leverages a cross-attention mechanism to align and combine features from visual and textual encoders, yielding robust performance even with limited training data, a critical advantage for FCD detection task. Inspired by these developments, we introduce a novel text-integrated framework for FCD detection. Specifically, we adopt the pre-

trained MELD Graph as a robust visual backbone and extend it by incorporating atlas-derived textual descriptions within an architecture inspired by Ariadne's Thread.

3 Method

As mentioned earlier, the design of our architecture is inspired by the work of [22]. Our implementation differs in two key aspects:

- we increased the depth of the feature extraction pathway, and
- we introduced a GNN-based block to more effectively aggregate visual features at higher resolutions.

The overall architecture is shown in Figure 3.1, and each component is described in detail below.

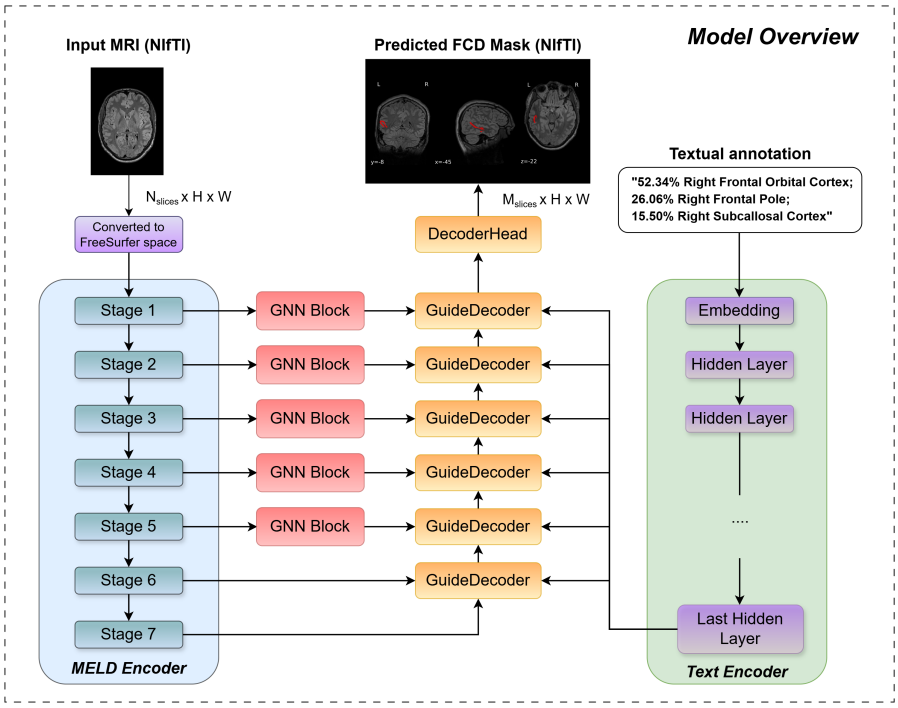

Figure 3.1: Overview of the proposed multimodal segmentation architecture. The input to the MELD encoder is an MRI scan in NIfTI format, which is first transformed into FreeSurfer space, a standardized surface-based cortical representation obtained by reconstructing the cerebral cortex from volumetric MRI. In this space, surface-based visual features include cortical thickness, curvature, sulcal depth, as well as intensity-based measures derived from T1-weighted and FLAIR images. The resulting surface-based visual features from each stage of the MELD encoder are then processed by their corresponding GNN block. In parallel, atlas-derived textual anatomical annotations are encoded by a pretrained language model, and the visual and textual representations are fused through the GuideDecoder blocks. Each stage has its own GuideDecoder, which integrates the multimodal information and passes it to the next level via upsampling. Finally, the fused features are decoded into the predicted FCD mask, which is output in NIfTI format. *Note.* The number of GNN blocks shown is illustrative and does not reflect a fixed architectural requirement; it is included solely for visualization purposes.

3.1 Visual Feature Extraction

As input to the vision model, structural MRI scans are first mapped into the FreeSurfer surface space. From the MELD preprocessing pipeline, we obtain a multi-resolution set of surface-based features across seven hierarchical levels. Each level is represented as a tensor of shape (H, N, C), where:

- H – number of hemispheres, with $H = 2$ corresponding to the left and right hemispheres,
- N – number of vertices on the cortical mesh per hemisphere,
- C – number of features per vertex.

To aggregate higher-order geometric information, we process the selected feature layers individually through a dedicated **GNN Block**. This design choice is based on empirical findings (see Chapter 5), where incorporating the lowest-resolution layers led to oversmoothing and diluted discriminative information, while using only the higher-resolution layers provided the best trade-off between expressivity and stability.

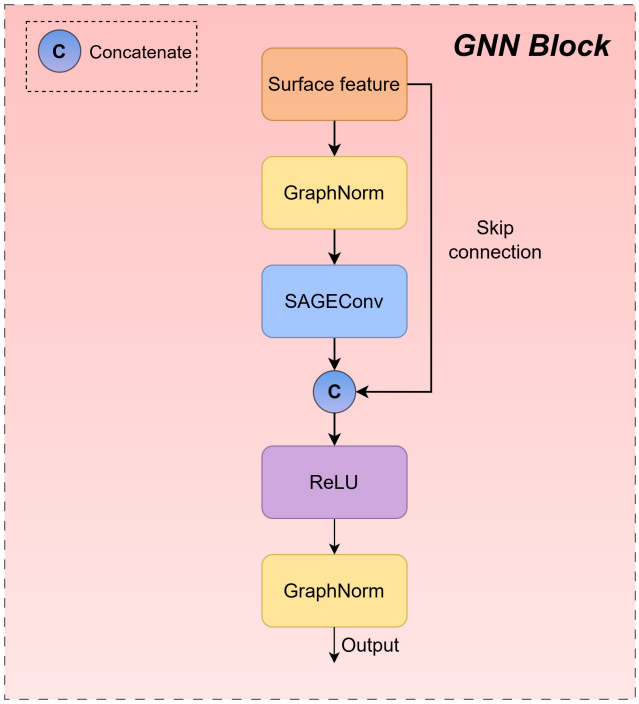

Figure 3.2: Structure of the GNN Block used in the MELD encoder. Each surface-based feature is normalized with GraphNorm, processed through a SAGEConv layer, and concatenated with the original feature via a skip connection. The result is passed through ReLU activation and another GraphNorm layer to produce the output representation.

3.2 GNN Block

Formally, let $\mathbf{X}^{(l)} \in \mathbb{R}^{(B \cdot H \cdot N_l) \times C_l}$ denote the input feature matrix at layer l, where B is the batch size, H is the number of hemispheres, N_l is the number of vertices per hemisphere at layer l, and C_l is the number of input channels.

Each *GNN Block* then applies the following sequence of operations:

$$\mathbf{H}_0 = \mathrm{GraphNorm}\big(\mathbf{X}^{(l)}, \texttt{batch}\big)\,, \tag{3.1}$$

$$\mathbf{H}_1 = \mathrm{SAGEConv}(\mathbf{H}_0, \texttt{edge\_index}_l)\,, \tag{3.2}$$

$$\mathbf{H}_2 = \mathbf{H}_1 + \mathbf{H}_0 \quad \text{(residual connection)}, \tag{3.3}$$

$$\mathbf{H}_3 = \mathrm{ReLU}(\mathbf{H}_2)\,, \tag{3.4}$$

$$\mathbf{Z}^{(l)} = \mathrm{GraphNorm}(\mathbf{H}_3, \texttt{batch})\,. \tag{3.5}$$

where:

- $\texttt{edge\_index}_l \in \mathbb{N}^{2\times|E_l|}$ — adjacency structure of the cortical surface graph;
- `batch` — batch vector;
- $\mathbf{Z}^{(l)} \in \mathbb{R}^{(B\cdot H\cdot N_l)\times C_l}$ — resulting representation for level l;
- **SAGEConv** implements the GraphSAGE [16] update rule

$$\mathbf{x}_i' = \mathbf{W}_1\mathbf{x}_i + \mathbf{W}_2 \cdot \underset{j\in\mathcal{N}(i)}{\mathrm{mean}}\, \mathbf{x}_j,$$

  where $\mathcal{N}(i)$ is the neighborhood of node i;
- **GraphNorm**$(x) = \gamma \cdot \dfrac{x - \mu_g}{\sqrt{\sigma_g^2 + \epsilon}} + \beta$, where μ_g, σ_g^2 are mean and variance per graph, and γ, β are learnable parameters;
- **ReLU**$(x) = \max(0, x)$ — rectified linear activation function;

The choice of **GraphNorm** is motivated by its ability to normalize node features on a per-graph basis. In practice, individual graphs can vary substantially in size, topology, and feature distribution, and normalizing across an entire batch may therefore yield unstable or inconsistent statistics. By computing normalization statistics separately for each graph, GraphNorm preserves the overall feature distribution within that graph and provides more stable activations, which is particularly important when training on heterogeneous cortical graphs. In addition, we adopt the pretrained **SAGEConv** operator as the main aggregation mechanism, since it is widely used and has proven effective for combining information from neighboring nodes in graph-based models.

After passing through the GNN Block, the feature dimensionality remains unchanged, ensuring consistency with subsequent blocks. The resulting features are then passed to the GuideDecoder for multimodal fusion.

3.3 Textual Feature Extraction

The text branch of our architecture leverages the pretrained **RadBERT** model [23], which was selected as an initial choice due to its specialization in radiology reports. In Chapter 5, we

additionally evaluate alternative pretrained text encoders to assess whether domain adaptation or model size affects segmentation performance.

The output of RadBERT's final hidden layer is passed to each GuideDecoder to enable multimodal fusion with cortical graph features.

Although previous studies suggest that selectively unfreezing the last transformer layers may improve downstream performance [24], we do not explore this direction in the present work and leave it for future investigations.

3.4 GuideDecoder

We adopt the GuideDecoder architecture (Figure 3.3) proposed by Zhong et al. [22], which fuses visual and textual features in a multimodal manner. The decoder first receives the text embeddings produced by the pretrained language model and projects them into the visual feature space so that their dimensionality matches that of the image tokens. Positional encodings are then added to both modalities to preserve their ordering and support the attention mechanisms operating within the decoder.

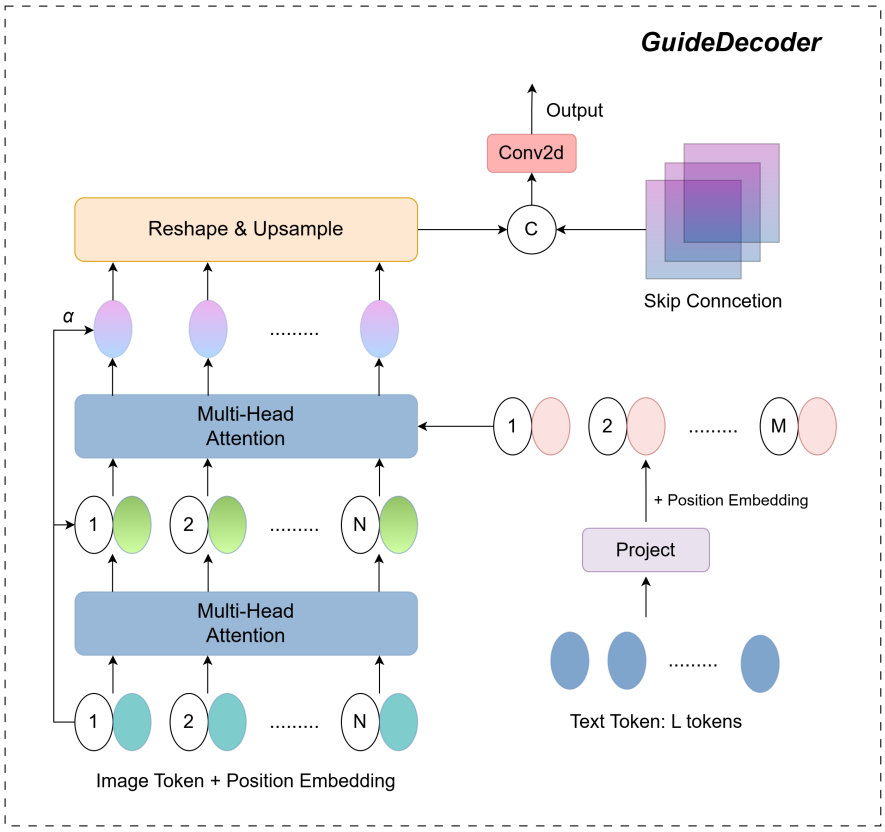

Figure 3.3: Structure of the GuideDecoder block. At each stage, the block receives image tokens from the visual encoder and text embeddings projected into the visual feature space. The visual tokens are refined through multi-head attention and fused with the projected text tokens via cross-attention. The fused multimodal features are then upsampled and combined with the skip connection before the final convolution produces the stage-specific output.

The visual tokens are processed through a stack of *multi-head attention layers*, which iteratively refine their representations before interacting with the text stream. The projected text

tokens serve as keys and values in a *cross-attention module*, allowing the decoder to incorporate semantically relevant information from the textual description into the visual features.

The resulting multimodal representations are subsequently reshaped and upsampled to the spatial resolution required by the decoder stage and fused with the corresponding skip connections from the visual encoder. A final convolutional layer transforms the fused features into the output prediction map.

In our implementation, the overall design of the GuideDecoder is preserved, but we introduce two modifications to adapt the architecture to surface-based representations. First, we replace the standard 2D upsampling with the `HexUnpool` operator (see Appendix HexUnpool), which performs mean unpooling on the icosphere mesh. Second, instead of the original `DynUNetBlock`[1] from the MONAI framework, which consists of convolution, normalization, and activation layers, the standard convolution is substituted with a `SpiralConv` operator [25], enabling convolutional feature aggregation directly on the cortical surface mesh after upsampling and fusion with skip connections.

3.5 Loss function

Following best practices for medical image segmentation and to ensure a fair comparison with the MELD model, we employed a composite loss. The loss function is defined as

$$L = L_{ce} + L_{dice} + L_{dist} + L_{class} + \sum_{i \in I_{ds}} w_{ds}^{i} \cdot (L_{ce}^{i} + L_{dice}^{i} + L_{dist}^{i}) \tag{3.6}$$

The individual components are detailed below.

Cross-Entropy Loss

For the segmentation task, we employ the binary cross-entropy loss, which is commonly used in medical image segmentation. Here, y_i denotes the ground-truth label, $\hat{y}_i$ the predicted probability, and n the number of vertices:

$$L_{ce} = -\frac{1}{n} \sum_{i=1}^{n} \left[y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \right]. \tag{3.7}$$

Dice Loss

The Dice Loss L_{dice} directly optimizes for the overlap between predicted and ground-truth lesion masks. This loss is less sensitive to class imbalance and encourages the network to predict coherent lesion regions. It is defined as

$$L_{dice} = 1 - \frac{2 \sum_{i=1}^{n} y_i \hat{y}_i}{\sum_{i=1}^{n} y_i^2 + \sum_{i=1}^{n} \hat{y}_i^2 + \epsilon} \tag{3.8}$$

It is important to note that we considered two different implementations of this loss. In the MONAI library, the Dice score is computed for each sample and then averaged over the batch (macro–averaging), whereas in the MELD implementation the aggregation is performed

[1] https://docs.monai.io/en/0.3.0/_modules/monai/networks/blocks/dynunet_block.html

immediately over the entire batch (micro–averaging). In our experiments, we observed that under strong class imbalance between background and lesion voxels, the micro–averaged version shifts the loss contribution towards the background, which reduces the gradient signal for rare lesion voxels and leads to training instability. Based on these empirical observations, we used the MONAI implementation, which provided higher metric values and enabled the model to detect more FCDs.

Distance Loss

To provide the network with additional contextual information and reduce false positives, we include a distance regression loss L_{dist}. The model is trained to predict the normalized geodesic distance d_i from each vertex to the lesion boundary. Here, $\hat{y}_{i,0}$ denotes the model's output corresponding to the *non-lesional* class for vertex i. We employ a mean absolute error loss weighted by $(d_i + 1)^{-1}$, which reduces the contribution of distant non-lesional vertices to the loss. Since the weight decreases with distance from the lesion boundary, the network focuses more on regions located closer to the lesion [9]:

$$L_{dist} = \frac{1}{n} \sum_{i=1}^{n} \frac{|d_i - \hat{y}_{i,0}|}{d_i + 1}. \tag{3.9}$$

Classification Loss

And in the last case, we add a weakly-supervised classification head to mitigate the uncertainty between lesion masks and actual lesions. Each subject is labeled as positive if any vertex belongs to a lesion. The classification head aggregates features across the deepest level (level 1) and predicts a subject-level label $\hat{c}$. The classification loss is then computed as binary cross-entropy [9]:

$$L_{class} = -\sum_{i=1}^{n} c_i \log(\hat{c}_i) + (1 - c_i) \log(1 - \hat{c}_i). \tag{3.10}$$

Deep Supervision

To encourage gradient flow and stabilize training, we adopt deep supervision at intermediate decoder levels $I_{ds} = \{6, 5, 4, 3, 2, 1\}$. At each level i, auxiliary predictions are generated and the same combination of cross-entropy, dice, and distance losses is applied. These auxiliary losses are weighted by w^i_{ds} and added to the total objective [9]:

$$\sum_{i \in I_{ds}} w^i_{ds} (L^i_{ce} + L^i_{dice} + L^i_{dist}). \tag{3.11}$$

3.6 Metrics

To evaluate the performance of our model, we follow the metrics used by the MELD authors for convenient comparison: Dice score, Positive Predictive Value (PPV), Intersection over Union (IoU), specificity, and sensitivity. Below we briefly describe each of them.

Dice score

The Dice similarity coefficient (DSC) is defined as the harmonic mean between precision and recall. For two sets of predicted positives P and ground truth positives G, it is given by

$$\text{Dice}(P, G) = \frac{2\,|P \cap G|}{|P| + |G|} = \frac{2TP}{2TP + FP + FN}, \tag{3.12}$$

where TP, FP and FN denote true positives, false positives and false negatives, respectively.

Positive Predictive Value

Also known as precision, PPV measures the proportion of correctly identified positive samples among all predicted positives:

$$\text{PPV} = \frac{TP}{TP + FP}. \tag{3.13}$$

In this work, we report PPV at two granularities:

- **Pixel-level PPV** evaluates precision over individual vertices of the cortical surface and therefore reflects how accurately the predicted mask matches the ground-truth lesion at a fine spatial scale.
- **Cluster-level PPV** treats each connected predicted region as a single detection and counts it as a true positive only if it overlaps with the manual lesion mask (or, following the MELD protocol, lies within 20 mm of it).

Intersection over Union

The IoU (also called the Jaccard index) quantifies the overlap between predicted and ground truth labels:

$$\text{IoU}(P, G) = \frac{|P \cap G|}{|P \cup G|} = \frac{TP}{TP + FP + FN}. \tag{3.14}$$

Specificity

Specificity quantifies the model's ability to avoid false-positive detections in healthy controls. It is defined as the proportion of control subjects for whom the model predicts no lesion clusters. Let $y_i = 0$ denote control subjects and $\hat{y}_i = 0$ indicate that the model produced no predictions for subject i. Then specificity is given by

$$\text{Specificity} = \frac{\sum_{i=1}^{N} \mathbb{1}(y_i = 0 \wedge \hat{y}_i = 0)}{\sum_{i=1}^{N} \mathbb{1}(y_i = 0)}. \tag{3.15}$$

Higher specificity indicates fewer false alarms in subjects without FCD.

Sensitivity

Sensitivity measures the ability of the model to correctly detect lesions in patients with FCD. Following the MELD evaluation protocol, a prediction is counted as correct if at least one predicted cluster overlaps with the manual lesion mask or lies within a 20 mm distance from it.

Formally, for a set of patients with ground-truth labels $y_i \in \{0, 1\}$ indicating the presence of FCD and binary prediction outcomes $\hat{y}_i \in \{0, 1\}$, sensitivity is defined as

$$\text{Sensitivity} = \frac{\sum_{i=1}^{N} \mathbb{1}(y_i = 1 \wedge \hat{y}_i = 1)}{\sum_{i=1}^{N} \mathbb{1}(y_i = 1)}. \tag{3.16}$$

A higher value indicates that the model successfully identifies a larger proportion of true FCD cases.

Confidence Intervals

To quantify the uncertainty of the reported metrics, we compute 95% confidence intervals following the procedure used by the MELD authors. For each metric, we report the sample median together with a confidence interval estimated via a non-parametric percentile bootstrap.

Given scores $x_1, \ldots, x_N$, we draw $B = 10{,}000$ bootstrap resamples of size N with replacement and compute the median for each resample, yielding the bootstrap distribution $\{\tilde{x}^{(b)}\}_{b=1}^{B}$. The 2.5th and 97.5th percentiles of this distribution are taken as the lower and upper bounds of the 95% interval.

This method makes no assumptions about the underlying distribution; missing values (NaN) are removed prior to resampling. When $N = 1$, the interval collapses to the observed value. A fixed random seed is used for reproducibility.

4 Dataset

We gratefully acknowledge the **MELD Project** for providing access to the dataset used in this work. Without this resource, it would not have been possible to conduct a systematic evaluation of our model.

4.1 Bonn Dataset

The Bonn scientific group has published a presurgical MRI dataset on **OpenNeuro**, entitled "*An open presurgery MRI dataset of people with epilepsy and focal cortical dysplasia type II*" [26]. This dataset comprises **170 participants**, including **85 individuals with FCD type II** and **85 healthy controls**. For each participant, the following data are available:

- High-resolution **3D T1-weighted** MRI scans (isotropic voxels, $1\,\mathrm{mm}^3$ or $0.8\,\mathrm{mm}^3$ resolution, depending on the subject);
- Corresponding **isotropic 3D FLAIR** imaging;
- **Manually delineated regions of interest (ROIs)** identifying FCD lesions, provided for patients only;
- A set of **clinical and demographic variables** (including age, sex, lesion laterality and location, histopathological subtype IIa/IIb, MRI-negative status, and postoperative outcome according to Engel classification).

Dataset Structure

The dataset follows the BIDS (Brain Imaging Data Structure) convention, with each participant stored in an individual directory. A typical subject folder has the following structure:

```
sub-00001/
    anat/
        sub-00001_XXX-XXXXX_T1w.nii.gz
        sub-00001_XXX-XXXXX_T1w.json
        sub-00001_XXX-XXXXX_FLAIR.nii.gz
        sub-00001_XXX-XXXXX_FLAIR.json
        sub-00001_XXX-XXXXX_FLAIR_roi.nii.gz     (patients only)
```

Each NIfTI image is accompanied by a corresponding JSON sidecar file containing DICOM-derived metadata, including voxel size, TR/TE/TI parameters, image orientation, and scanner information.

At the dataset root, several metadata files are provided:

```
participants.tsv
participants.json
participants_with_scanner.tsv
dataset_description.json
```

The file `participants.tsv` contains demographic and clinical information (e.g., age, sex, diagnosis, lesion laterality, histopathological subtype), while `participants.json` provides descriptions of these variables. The file `participants_with_scanner.tsv` includes additional scanner-related metadata such as field strength and vendor. The standard BIDS file `dataset_description.json` documents the dataset-level metadata, and `meld_bids_config.json` contains configuration parameters required for MELD preprocessing and quality control.

Manually delineated lesion masks (`*_roi.nii.gz`) are available only for patients and are spatially aligned with the anatomical T1-weighted MRI.

4.2 MELD Dataset

The **MELD Project** provides a large-scale neuroimaging dataset comprising MRI scans and clinical data of patients with FCD, as well as healthy controls. In total, the dataset includes **1185 participants** from **23 international epilepsy surgery centers**, including the Bonn datasets described earlier. Due to missing or corrupted files in the obtained release, we used a subset of **1130 participants** in our experiments. The dataset is not publicly available; access must be requested directly from the study authors [9].

For each participant, the dataset includes:

- Structural MRI: **3D T1-weighted** (acquired for all participants) and **3D FLAIR**, which was included only for participants for whom this sequence was available;
- Lesion annotations: **manually delineated Region of Interest (ROI)** identifying FCD lesions (patients only). For MRI-negative cases, i.e., patients in whom no visible lesion is detected on conventional MRI despite histologically confirmed FCD, **postsurgical resection cavities** were used to guide ROI definition;
- Demographic and clinical metadata;
- Predefined splits for training, validation, and test cohorts.

Dataset Structure

To reduce inter-site variability arising from differences in scanners and acquisition protocols across centers, the MELD dataset provides surface-based features that have been harmonized using the ComBat method [27]. The provided ComBat-harmonized `.hdf5` files do not contain the multiscale representations used in the MELD pipeline. Instead, they store only per-vertex cortical attributes extracted by FreeSurfer (e.g., cortical thickness, curvature, sulcal depth, and T1/FLAIR intensity features), together with their asymmetry maps and the corresponding lesion masks.

Clinical and demographic metadata are distributed separately in CSV/TSV files. The available metadata fields include: *participant ID, scanning site, diagnostic group, age of onset,*

epilepsy duration, age at presurgical MRI, sex, MRI-negative status, Engel outcome, histopathology, lesion hemisphere and lobe, presence of FLAIR hyperintensity, seizure outcome, and scanner information.

The dataset release also includes predefined train/validation/test splits, which we follow to ensure comparability with prior MELD studies.

4.3 FreeSurfer Processing

Each image was processed with the FreeSurfer framework [28], from which **11 core surface-based features** were extracted:

- **Morphometric features:** cortical thickness, sulcal depth, curvature, and intrinsic curvature;
- **Intensity features:** gray–white matter intensity contrast, and FLAIR intensity sampled at 6 intracortical and subcortical depths.

In addition to the raw measurements, features were further processed into **raw values**, **control-normalized features**, and **asymmetry features** (left vs. right hemisphere). Cortical thickness was additionally adjusted by regressing out curvature. Altogether, this yielded **34 input features per participant**, computed at **163,842 vertices** on a bilaterally symmetrical cortical surface template [9]. Scanner-related variability was addressed through the ComBat harmonization applied in the released MELD features.

For conversion from volume space to FreeSurfer surface space, detailed instructions are available in the MELD documentation [29]. In practice, however, it is often more practical to request preprocessed files directly from the authors, since converting a single image to surface space is computationally expensive: approximately 6–7 hours per scan (even with FastSurfer [30] on an NVIDIA A100 GPU, the process required 3–4 hours).

4.4 Data Augmentation

In the original study MELD authors applied augmentations in three stages:

- **Lesion–mask augmentation**: deforming the lesion region together with its corresponding lesion mask and surface-based features to produce anatomically coherent augmented samples;
- **Mesh–space transforms**: applying geometric transformations to the cortical surface mesh, consistently affecting both features and lesion masks;
- **Intensity transforms**: modifying the intensity features at each vertex.

Each transform is applied independently with probability p defined in the experiment configuration. The order is fixed: lesion–mask augmentation $\rightarrow$ mesh transforms $\rightarrow$ intensity transforms.

Lesion–mask augmentation

Given a geodesic distance map D on the cortical surface (negative inside the lesion), we first normalise it by $|\min D|$ and add low–frequency noise, which is generated on a low–resolution icosphere (level 2) and then upsampled to the target resolution using the predefined unpool operators:

$$\tilde{D} \;=\; \texttt{Unpool}\Big(\tfrac{D}{|\min D|} \;+\; \mathcal{N}(0,\sigma^2)\Big)\,, \qquad L' \;=\; \mathbb{1}\{\tilde{D} \le 0\}.$$

In our implementation, we use $\sigma = 0.5$ by default. And as `Unpool` we use the `HexUnpool` operator (see Appendix HexUnpool). After modifying the binary lesion mask L, the geodesic distances and smoothed labels are **recomputed**:

$$D' = \texttt{fast\_geodesics}(L'), \qquad \widetilde{L} = \texttt{smoothing}(L', \text{iteration} = 10).$$

This procedure is applied only if the lesion mask is non-empty. After augmentation, the geodesic distances and smoothed labels are recomputed to ensure a consistent lesion shape. However, in our training setup, only the resulting binary lesion mask is used for supervision, while the recomputed distances and smoothed labels are not used directly in the loss function.

Mesh–space transforms (icosphere re–indexing)

We use precomputed vertex index mappings provided by the MELD authors, which define fixed permutations of vertices on the icosphere mesh. These mappings are generated offline for each transformation type (spinning, warping, and flipping) and are applied uniformly to all vertex-wise tensors (features, labels, distances, etc.) to preserve anatomical correspondence (GitHub repository). The MELD framework defines three types of such maps:

- **Spinning** — index remapping that corresponds to a rigid rotation of the icosphere.
- **Warping** — smooth non-rigid remapping that locally compresses or stretches the mesh.
- **Flipping** — mirror reflection of the mesh, e.g. exchanging left and right hemispheres.

Per-vertex feature intensity transforms

All per-vertex intensity transformations are applied to all feature channels. For each transformation, parameters are sampled per channel; unless stated otherwise, the same parameters are applied consistently across all vertices within a channel:

- **Gaussian noise**: add $\epsilon \sim \mathcal{N}(0,\sigma^2)$ with $\sigma^2 \sim \mathcal{U}(0, 0.1)$.
- **Brightness scaling**: multiply each channel by $m_c \sim \mathcal{U}(0.75, 1.25)$.
- **Contrast adjustment**: for each channel c, sample $f \sim \mathcal{U}(0.65, 1.5)$ and apply

$$x_c \;\leftarrow\; (x_c - \mu_c)\, f + \mu_c,$$

$$min_c \coloneqq \min(x_c), \quad max_c \coloneqq \max(x_c),$$

$$x_c[x_c < min_c] \;=\; min_c, \quad x_c[x_c > max_c] \;=\; max_c.$$

- **Gamma**: for $\gamma \sim \mathcal{U}(0.7, 1.5)$,

$$x' \leftarrow \Big(\frac{x - \min(x)}{\max(x) - \min(x) + \varepsilon}\Big)^{\gamma} (\max(x) - \min(x) + \varepsilon) + \min(x);$$

$$x'' \leftarrow \frac{x' - \mu}{\sigma + \varepsilon}, \quad \text{where } \varepsilon = 10^{-7},\ \mu = \text{mean}(x),\ \sigma = \text{std}(x).$$

We also use an *inverted* variant by applying the same operation to $-x$ and negating back.

Disabled/placeholder transforms

The current implementation contains placeholders for `Gaussian blur` and `low-resolution downsampling`. These operations are not implemented in the original authors' code and were not used in our reported results.

4.5 Types of Atlas Descriptions

We generated textual descriptions with `AtlasReader` [31]. Given each ROI, it is first registered to template space and anatomical atlases are overlaid to quantify its anatomical context. For every atlas region, we compute the percentage of the ROI volume that overlaps that region and report the resulting region–percentage table. When visual inputs are augmented, the corresponding textual descriptions are regenerated from the transformed lesion location to remain anatomically consistent, ensuring alignment between visual and textual modalities. In this work, we used the Harvard–Oxford (cortical and subcortical) probabilistic atlas [32].

However, strong validation scores obtained using full anatomical descriptions do not guarantee robustness at deployment: in real-world scenarios, neither precise overlap percentages nor complete region names are typically available. Therefore, in Chapter 5 we also evaluate several reduced-text settings:

I. *hemisphere only* (Left / Right);

II. *fine-grained lobe regions* (e.g., Precentral Gyrus, Frontal Pole, Subcallosal Cortex);

III. *coarse lobe labels* (e.g., Frontal, Temporal, Parietal, Occipital, Insular);

IV. *hemisphere + lobe / lobe regions*;

V. *mixed* (randomly sampling one of {hemisphere only, lobe region only, hemisphere + lobe, full text, no text}; this scheme is described in more detail in the next chapter).

The anatomical names of lobes and their respective regions are taken from *Automated Anatomical Labeling of Activations in SPM Using a Macroscopic Anatomical Parcellation of the MNI MRI Single-Subject Brain*.

5 Experiments

This chapter consists of several components. We begin by outlining key implementation details of the proposed model, followed by a description of the web interface developed to facilitate interaction with the different model variants. Finally, we present a series of experiments, including systematic ablation studies and an analysis of how various design choices influence model performance on both the main training cohort and an independent validation dataset.

5.1 Implementation Details

In this work, we aimed to use the same training parameters as the MELD authors to ensure a fair and consistent comparison. The proporsed framework was trained with the following hyperparameters. The training batch size was fixed at 8 and the validation batch size at 4. The initial learning rate was set to 3×10^{-4} and optimized using the OneCycleLR scheduler (maximum learning rate 3×10^{-3}). The schedule included a warm-up phase during the first 10% of training steps, followed by cosine annealing. Training was performed for up to 100 epochs (minimum 20 epochs), with early stopping applied if the validation performance did not improve for 20 epochs. For evaluation, we trained an ensemble of five independently initialized models (seeds 42–46). At test time, we averaged their per-vertex predictions to form the final output.

The model integrated surface-based graph features and contextual text embeddings. Feature channel dimensions across the encoder stages were $[32, 32, 64, 64, 128, 128, 256]$, with corresponding text sequence lengths $[128, 64, 64, 32, 32, 16, 16]$, and a maximum text length of 256 tokens. Deep supervision was employed with levels $I_{ds} = [6, 5, 4, 3, 2, 1]$ and associated weights $w_{ds} = [0.5, 0.25, 0.125, 0.0625, 0.03125, 0.0150765]$ [9]. The text encoder was initialized from RadBERT, with a projection dimension of 768.

To address class imbalances, non-lesional hemispheres were undersampled, ensuring that approximately one third of training examples contained a lesion. Data augmentation strategies included:

Table 5.1: Data augmentation strategies applied during training [9].

Transformation	**Probability**
Random flipping	0.5
Gaussian blur	0.2
Spinning	0.2
Warping	0.2
Brightness/contrast/gamma	0.15 each
Gaussian noise	0.15
Low-resolution simulation	0.25

Hardware

All experiments were performed on the *Bender* high-performance computing cluster at the University of Bonn, equipped with NVIDIA A100 GPUs (80 GiB each) and AMD EPYC CPUs. A detailed description of the system can be found in the official documentation[1]. Our implementation used PyTorch 2.1.0+cu121, TorchVision 0.16.0+cu121, TorchAudio 2.1.0+cu121, Torch Geometric 2.5.3, Torch Scatter 2.1.2, TorchMetrics 0.11.4, and Python 3.9.18.

5.2 Web Interface

To make it easier to use different pretrained models, we developed a web interface that offers an accessible and intuitive way for users to interact with the lesion detection system.

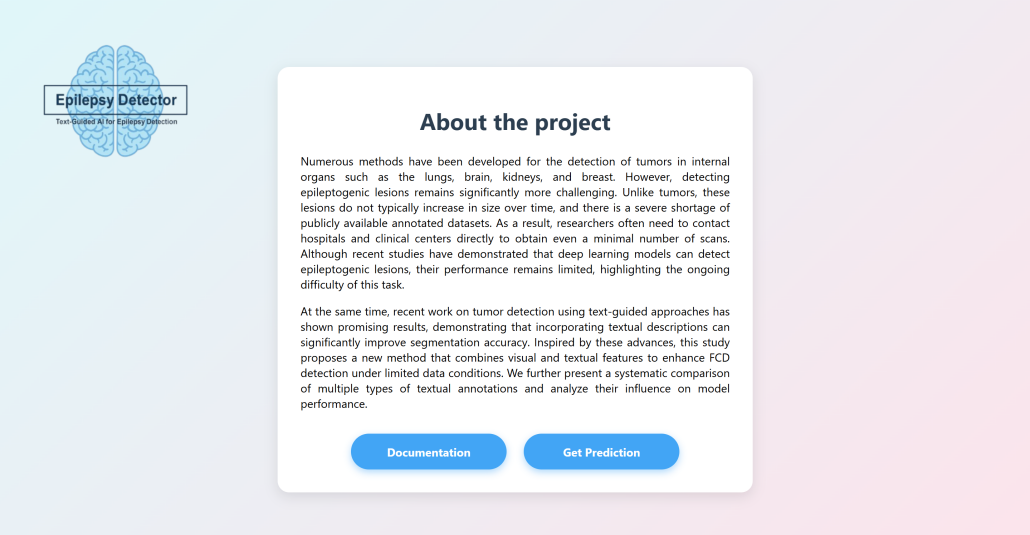

Figure 5.1: Main page of the web interface with a brief project description and navigation buttons.

The main page presents a brief overview of the project and includes navigation controls that let users explore the documentation or move on to the prediction page for processing a specific patient.

[1] https://www.hpc.uni-bonn.de/en/systems/bender

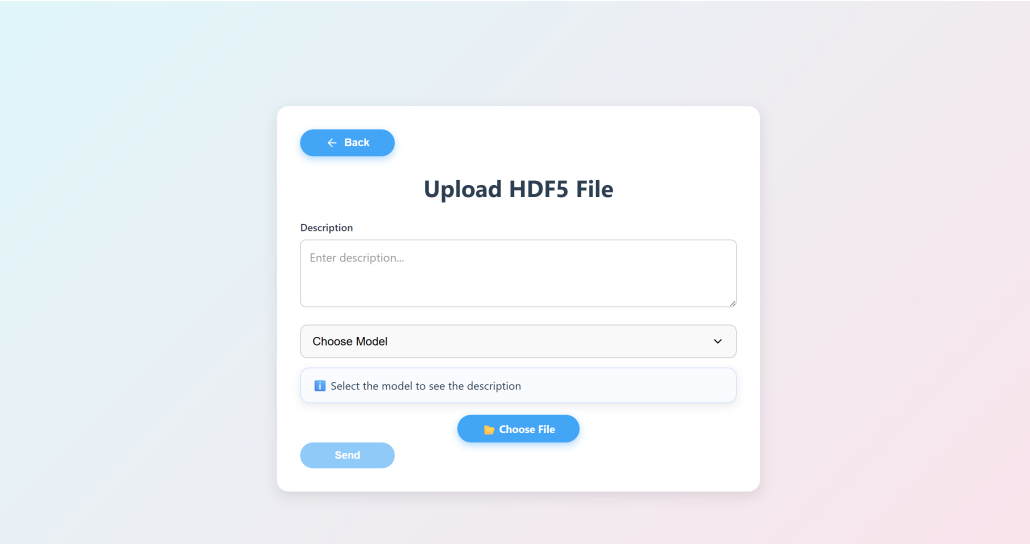

Figure 5.2: Upload page where the user selects a model, adds a description, and uploads an HDF5 file for analysis. *Note: the current prototype accepts only a single HDF5 file.*

The workflow begins on the upload page, where the user selects one of the trained models, optionally adds a short textual description, and uploads an HDF5 file for analysis. The current prototype supports the upload of a single HDF5 file at a time.

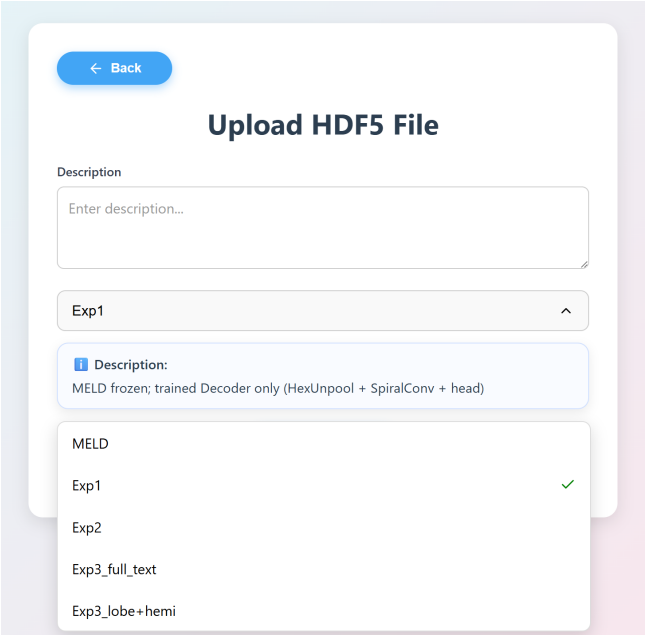

Figure 5.3: Dropdown menu for selecting among trained models.

To support different use cases and experimental settings, the interface includes a model selection menu listing all available trained variants. These include the baseline MELD model, two vision-only architectures (Exp1 and Exp2) with minor architectural differences, and the multimodal Exp3 model trained on various textual inputs. Each option is accompanied by a short explanation—similar in style to modern LLM model pickers – helping users choose between faster or more accurate configurations.

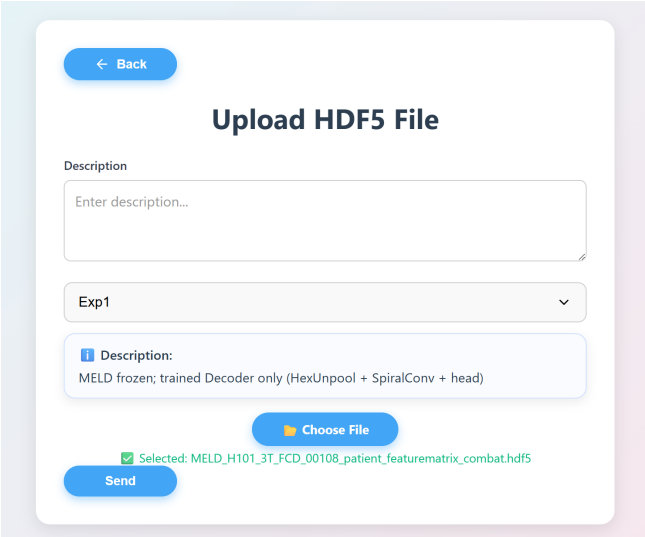

Figure 5.4: Example of a successfully uploaded file ready for processing.

Once a file is uploaded successfully, the interface confirms that the case is ready for processing and allows the user to initiate inference. After the prediction is complete, the results page displays the detected lesion regions and provides two export formats: PNG images for quick visual inspection and NIfTI volumes for more advanced analysis in 3D medical imaging software. This enables both general users and developers to efficiently inspect, validate, and further process the model outputs.

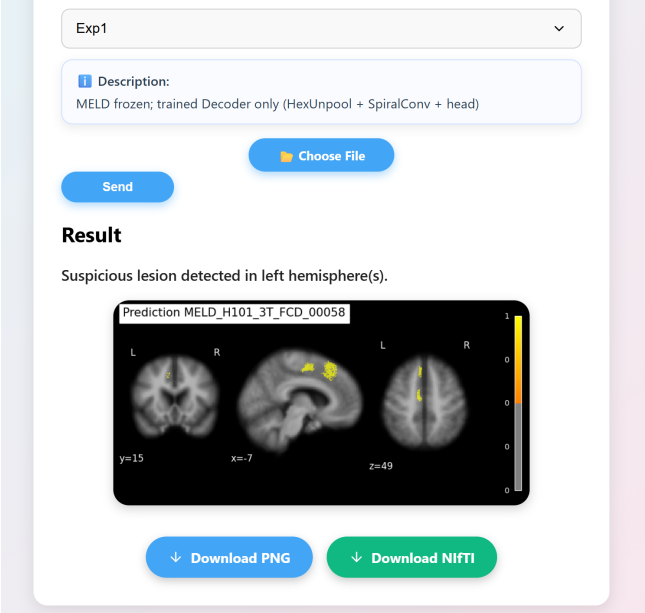

Figure 5.5: Prediction result with detected lesion regions, with download options for PNG (for general users) and NIfTI (for developers; viewable in 3D medical imaging tools for further analysis).

5.3 Basic Experiments

To systematically assess the contribution of each component of the proposed multimodal architecture, we gradually enabled individual modules of the decoder — geometry-based upsampling, self-attention, and finally text-conditioned cross-attention. This stepwise experimental

design allows isolating the effect of textual guidance and ensuring that any improvement in lesion detection quality can be attributed to a specific architectural modification rather than confounding changes.

In all options, we freezed both encoders: the visual encoder (the **MELD** backbone) and the text encoder. Although **RadBERT** was used as the initial language model during the early stages of this work, simply because it was the first domain-specific model that integrated well into the pipeline, the choice of text encoder was later examined more systematically. After observing that textual signals substantially influence lesion detection quality, we conducted an additional study comparing several biomedical and radiology-oriented language models (e.g., BlueBERT, PubMedBERT). This analysis (see Appendix Semantic similarity analysis) motivated the inclusion of multiple text-encoder variants in the experimental section.

Across all variants, the geometry-based upsampling path (`HexUnpool` + `SpiralConv`) and the final segmentation head are always trained, while the encoders remain frozen (Fig. 5.8). The experimental variants differ only in (i) whether the `GuideDecoder` is inserted before the upsampling path and (ii) which textual features are incorporated. The pretrained `Exp1` model (described below) is used to initialize the decoder for all variants except the MELD-only baseline, which is trained from scratch.

We consider the following configurations:

- **MELD.** Serves as the baseline model.
- **Exp1 (Unpool + Spiral, no text).** In this setting, the self- and cross-attention mechanisms of the `GuideDecoder` are disabled, while the remaining reshaping and upsampling components are kept (see Fig. 5.6). The vision features from the MELD encoder are reshaped and passed directly into the geometry-based upsampling path (`HexUnpool` + `SpiralConv`) and then to the segmentation head. No textual input is used in this variant, so the model relies purely on image-derived features.

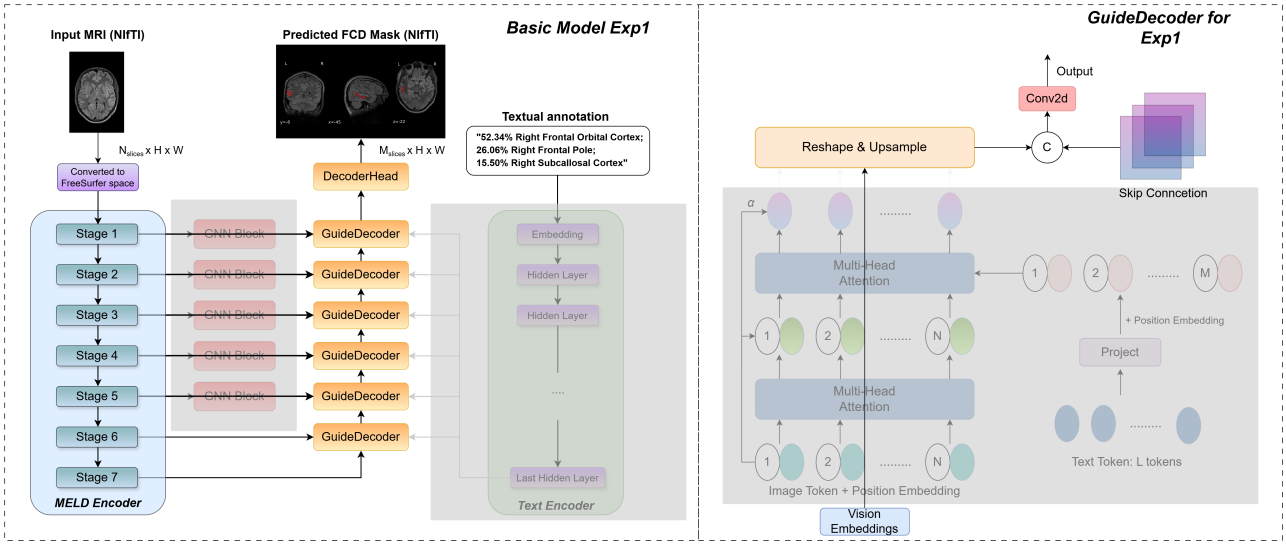

Figure 5.6: Architecture of Exp1: the text branch and the self-/cross-attention modules in the `GuideDecoder` are disabled, and MELD vision features are fed directly into the geometry-based upsampling path and segmentation head.

- **Exp2 (GuideDecoder: self-attention only).**

A stack of `GuideDecoder` blocks is inserted into the decoder before each upsampling stage. Since the MELD U-Net decoder has a depth of 7 (i.e., 6 upsampling levels), we include six `GuideDecoder` blocks at stages D6–D1, from the coarsest to the finest resolution. The upsampling path and segmentation head remain identical to Exp1. In this experiment the text branch is disabled, so each `GuideDecoder` performs only self-attention–based refinement of the visual features before they are upsampled (see Fig. 5.7).

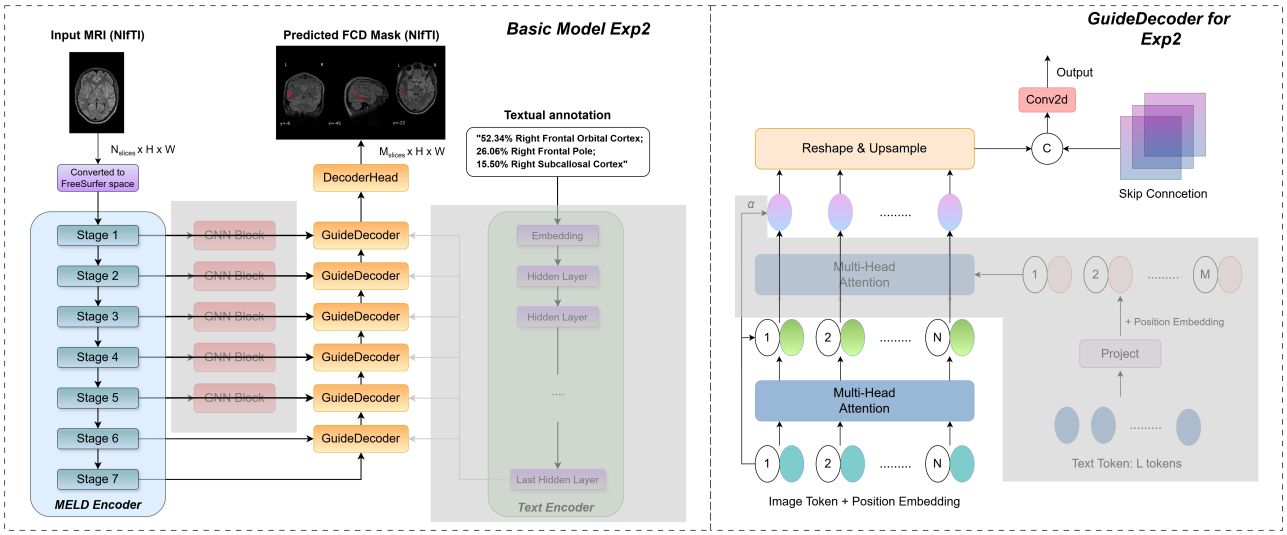

Figure 5.7: Architecture of Exp2: each `GuideDecoder` block contains a self-attention head, while the text branch remains disabled.

- **Exp3 (GuideDecoder + Text).** The full `GuideDecoder`, incorporating self-attention on image tokens and cross-modal attention between image and text embeddings, is used (see Fig. 5.8). The only difference across variants is the textual conditioning:

  - *full* (complete Atlas annotations): *48.86% Left Middle Frontal Gyrus; 30.63% Left Precentral Gyrus; 20.51% Left Superior Frontal Gyrus*

  - *hemi* (hemisphere only): *Left Hemisphere* ;

  - *lobe_regions* (lobe regions only): *Left Middle Frontal Gyrus; Precentral Gyrus; Superior Frontal Gyrus*;

  - *hemi+lobe_regions* (hemisphere + lobe regions): *Left Hemisphere; Middle Frontal Gyrus; Precentral Gyrus; Superior Frontal Gyrus* ;

  - *hemi+lobe* (hemisphere + lobe): *Left Hemisphere; Frontal lobe.*

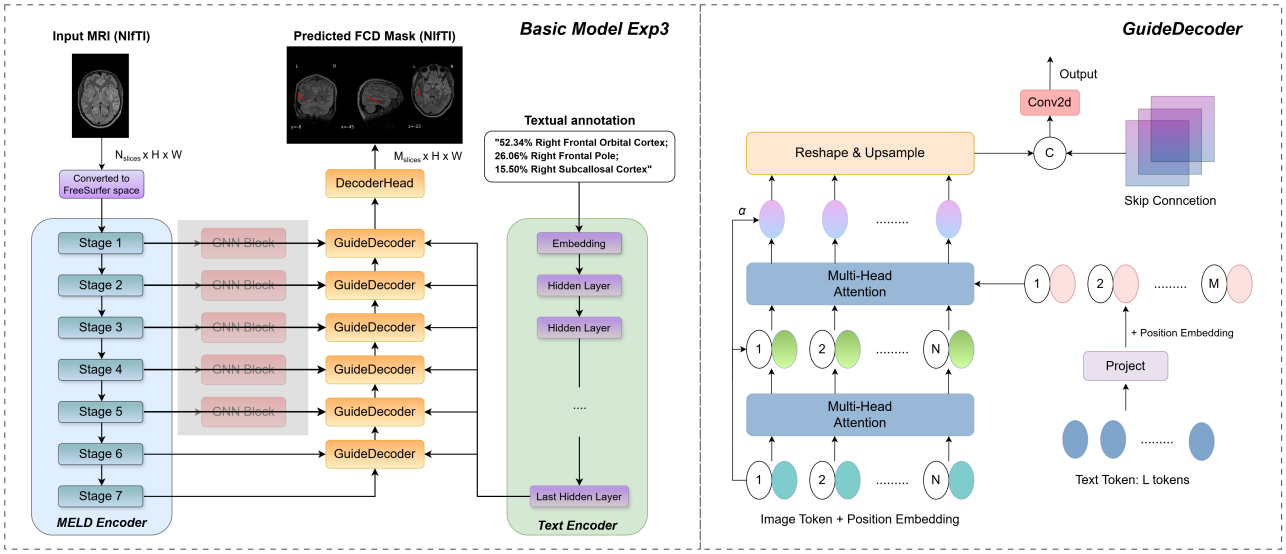

Figure 5.8: Architecture of Exp3: the full `GuideDecoder` is used, incorporating both self-attention on visual tokens and cross-attention between visual features and text embeddings.

- **Exp3_mixed (GuideDecoder + Text).** Same as **Exp3**, but Atlas descriptions were randomly sampled from one of {*hemisphere only*, *lobe_regions only*, *hemisphere + lobe_regions*, *full text*, *no text*}.

  For each subject, we first generated a full Atlas-based description and then derived all partial textual variants. During training, a single variant was randomly selected at every data loading step, so that the same subject could appear with different textual inputs across epochs, while the visual augmentations remained fixed.

Main cohort

Table 5.2 presents the median performance on the main cohort (i.e., data from the same distribution as used during training).

In the RadBERT group, defined as all variants using the default RadBERT language encoder (without alternative LLMs such as BlueBERT or PubMedBERT), the prompt combining *hemisphere and lobe-region* terms detected the most lesions, though with only slightly higher Dice, PPV_{pixels}, and IoU. Across all models, the PubMedBERT variant achieved the highest number of detected lesions, likely due to better domain-context understanding (see Table 1 in the Appendix). This clearly demonstrates that selecting an appropriate language model can materially improve performance, underscoring the substantial impact of the language encoder.

Table 5.2: Median performance on the main cohort (with 95% confidence intervals).

Model	Dice	PPV pixels	(mean) PPV clusters	(median) PPV clusters	IoU	Specificity, % (N = 193)	Sensitivity, % (N = 259)
MELD	0.230 (0.107–0.320)	0.166 (0.086–0.220)	0.515	1.000 (1.000-1.000)	0.130 (0.057–0.190)	58 (n=112) (51.3–64.8)	65.6 (n=170) (59.8–71.4)
Exp1	0.238 (0.125–0.313)	0.198 (0.130–0.318)	0.361	1.000 (0.500-1.000)	0.135 (0.066–0.186)	31.1 (n=60) (24.9–37.8)	69.1 (n=179) (63.3–74.9)
Exp2	0.256 (0.125–0.326)	0.204 (0.122–0.322)	0.333	1.000 (0.500-1.000)	0.147 (0.067–0.195)	24.9 (n=48) (19.2–31.1)	70.7 (n=183) (65.3–76.1)
Exp3: hemi	0.231 (0.148–0.285)	0.188 (0.131–0.298)	0.483	0.667 (0.500-1.000)	0.131 (0.080–0.166)	100.0 (n=193) (100.0–100.0)	71.0 (n=184) (65.6–76.4)
Exp3: lobe_regions	0.254 (0.159–0.332)	0.231 (0.168–0.320)	0.404	0.667 (0.500-1.000)	0.145 (0.086–0.199)	100.0 (n=193) (100.0–100.0)	72.6 (n=188) (67.2–78.0)
Exp3: hemi + lobe_regions	0.264 (0.180–0.342)	0.238 (0.160–0.327)	0.333	0.500 (0.500-1.000)	0.152 (0.099–0.206)	99.5 (n=192) (98.4–100.0)	74.5 (n=193) (69.1–79.9)
Exp3: hemi + lobe	0.239 (0.113–0.302)	0.242 (0.166–0.353)	0.456	0.667 (0.500-1.000)	0.136 (0.060–0.178)	99.5 (n=192) (98.4–100.0)	71.4 (n=185) (66.0–76.8)
Exp3: hemi + lobe + BlueBERT	0.230 (0.125–0.305)	0.231 (0.146–0.330)	0.443	0.667 (0.500-1.000)	0.130 (0.067–0.180)	99.0 (n=191) (97.4–100.0)	71.8 (n=186) (66.4–77.2)
Exp3: hemi + lobe + PubmedBERT	0.233 (0.139–0.330)	0.242 (0.153–0.342)	0.334	0.500 (0.500-0.667)	0.132 (0.074–0.198)	100.0 (n=193) (100.0–100.0)	76.4 (n=198) (71.4–81.5)
Exp3: full_desc	0.246 (0.145–0.330)	0.259 (0.178–0.360)	0.478	0.667 (0.500-1.000)	0.141 (0.078–0.197)	94.3 (n=182) (90.7–97.4)	72.6 (n=188) (67.2–78.0)

Note. Colors denote rank within each column: best , second , third .
Exp labels. **Exp3: hemi** – hemisphere conditioning; **lobe_regions** – lobe- and region-level prompts; **hemi+lobe** – hemisphere + coarse lobe name; **hemi+lobe+BlueBERT** – same as previous but with BlueBERT instead of RadBERT; **full_desc** – full free-text description; **Exp1** and **Exp2** are ablation baselines.

In real-world settings, detailed subregion information may be unavailable. We therefore evaluated prompts with only general *lobe names* (e.g., *Frontal lobe*, *Temporal lobe*, *Insular lobe*). For the RadBERT-based models, performance was very close to the *hemi + lobe_regions* setup: Dice decreased by ∼2.5%, PPV_pixels by ∼0.4%, IoU by ∼1.6%, and only **8** fewer lesions were detected. Likewise, using only *hemisphere* descriptions yielded results close to *hemi + lobe*, indicating that even a minimal textual prompt can help identify additional lesion regions.

Also, when using the full textual description (e.g., "52.34% Right Frontal Orbital Cortex; 26.06% Right Frontal Pole; 15.50% Right Subcallosal Cortex"), the model performed worse than *hemi + lobe_regions*. One possible reason is the presence of redundant information, such as percentage overlaps of regions, which may not contribute to the prediction task. Since RadBERT was trained on radiology reports, it may not benefit from such structured numeric content. For comparison, we also tested PubMedBERT, which was trained on a larger corpus of brain scan–related reports. Although its quality metrics were slightly lower than those of the *hemi + lobe* configuration with RadBERT, it nevertheless detected **13 additional lesion clusters**.

We also report specificity, defined as the proportion of healthy patients with no predicted clusters. Across all *Exp3 models*, specificity was ∼36–42% higher than that of MELD. This indicates that the text-conditioned models distinguish healthy from non-healthy cases substantially better.

Throughout this section, we did not focus on the median $\text{PPV}_{\text{clusters}}$, the primary metric used in the original MELD study. In our experiments, the median $\text{PPV}_{\text{clusters}}$ is equal to 1.0 for nearly all models and therefore provides no discrimination; as a result, it is not informative and does not help us interpret the results. The mean $\text{PPV}_{\text{clusters}}$, in contrast, reveals more structure. As shown in Table 5.2, MELD achieves the highest mean $\text{PPV}_{\text{clusters}}$ on the main cohort, with a small margin of approximately 2–3% over the next two models and a much larger gap to the remaining ones. On the independent cohort Table 5.3, however, the situation is reversed: the

proposed architecture shows improvements of 2–8% compared with the two best-performing baselines. These results are more informative and reflect model differences more reliably. For these reasons, in the following experiments we report only the mean $PPV_{clusters}$, as it allows us to distinguish between models in this study, whereas the median version does not.

Independent cohort

Table 5.3: Median performance on the independent cohort (with 95% confidence intervals).

Model	Dice	PPV pixels	(mean) PPV clusters	(median) PPV clusters	IoU	Specificity, % (N = 83)	Sensitivity, % (N = 82)
MELD	0.358 (0.209–0.465)	0.261 (0.142–0.428)	0.464	1.000 (1.000–1.000)	0.218 (0.117–0.303)	55.4 (n=46) (44.6–66.3)	69.5 (n=57) (59.8–79.3)
Exp1	0.376 (0.238–0.529)	0.443 (0.243–0.619)	0.350	1.000 (1.000–1.000)	0.232 (0.138–0.359)	39.8 (n=33) (28.9–50.6)	73.2 (n=60) (63.4–82.9)
Exp2	0.342 (0.155–0.494)	0.381 (0.177–0.651)	0.379	1.000 (0.667–1.000)	0.206 (0.084–0.328)	37.3 (n=31) (27.7–48.2)	69.5 (n=57) (59.8–79.3)
Exp3: hemi	0.372 (0.261–0.515)	0.380 (0.181–0.601)	0.468	1.000 (1.000–1.000)	0.228 (0.150–0.347)	100.0 (n=83) (100.0–100.0)	74.4 (n=61) (64.6–84.1)
Exp3: lobe_regions	0.373 (0.283–0.507)	0.407 (0.233–0.589)	0.257	1.000 (0.667–1.000)	0.229 (0.165–0.340)	100.0 (n=83) (100.0–100.0)	78.0 (n=64) (68.3–86.6)
Exp3: hemi+ lobe_regions	0.363 (0.190–0.513)	0.434 (0.209–0.657)	0.359	1.000 (0.500–1.000)	0.222 (0.105–0.345)	97.6 (n=81) (94.0–100.0)	74.4 (n=61) (64.6–84.1)
Exp3: hemi+lobe	0.327 (0.164–0.512)	0.411 (0.236–0.667)	0.545	1.000 (0.667–1.000)	0.196 (0.090–0.344)	98.8 (n=82) (96.4–100.0)	72.0 (n=59) (62.2–81.7)
Exp3: hemi+lobe+ BlueBERT	0.362 (0.260–0.498)	0.469 (0.267–0.620)	0.356	1.000 (0.750–1.000)	0.221 (0.150–0.331)	96.4 (n=80) (91.6–100.0)	76.8 (n=63) (67.1–85.4)
Exp3: hemi+lobe+ PubmedBERT	0.347 (0.195–0.529)	0.457 (0.202–0.730)	0.258	1.000 (0.500–1.000)	0.210 (0.109–0.360)	100.0 (n=83) (100.0–100.0)	80.5 (n=66) (72.0–89.0)
Exp3: full_desc	0.423 (0.267–0.506)	0.484 (0.315-0.693)	0.460	1.000 (1.000–1.000)	0.268 (0.154-0.339)	90.4 (n=75) (83.1–96.4)	73.2 (n=60) (63.4–82.9)

Note. Colors denote rank within each column: best , second , third .
Exp labels. **Exp3: hemi** – hemisphere conditioning; **lobe_regions** – lobe- and region-level prompts; **hemi+lobe** – hemisphere + coarse lobe name; **hemi+lobe+BlueBERT** – same as previous but with BlueBERT instead of RadBERT; **full_desc** – full free-text description; **Exp1** and **Exp2** are ablation baselines.

Following the MELD paper, we used the dataset obtained from Bonn as an independent test cohort. This evaluation allows assessing the generalization capability of the models to unseen data from a different site and acquisition setting.

The models that detected the largest number of lesion clusters were those trained with *lobe_regions* prompts in the RadBERT group, and the *hemi + lobe + PubMedBERT* variant across all architectures. When PubMedBERT was used instead of RadBERT in the *hemi + lobe* configuration, the model detected more lesions (sensitivity: 80.5% vs. 76.8%), indicating improved lesion—level recall. However, this came at the cost of slightly lower segmentation quality, as reflected by slightly decreases in Dice (–1.5%), PPV_{pixels} (–1.2%), and IoU (–1.1%). This trade-off is consistent with PubMedBERT's pretraining on a substantially larger corpus of biomedical and brain-related clinical text, which enhances its ability to recognize lesion-related terminology but does not necessarily improve spatial localization. Overall, the *hemi + lobe + PubMedBERT* model provides the strongest sensitivity (66/82), a key objective in this study, while maintaining acceptable segmentation quality.

5.4 Mixed Text

We investigated how different types of textual descriptions affect segmentation quality (**Exp3 mixed**). During training, one available description was randomly sampled for each patient, whereas evaluation used a fixed prompt type. We also assessed robustness to *incorrect* prompts. All experiments employed **RadBERT** as the text encoder.

We compared two model options:

1. **Decoder open from the start**: the GuideDecoder is initialized with pre-trained weights from *Exp1* and trained from the first epoch.

2. **Decoder warm-up (5 epochs)**: during the first five epochs, only the GuideDecoder is trained. The choice of five epochs is pragmatic: each experiment uses an ensemble of models, a single model trains for roughly 8 hours, and typical runs span 20–30 epochs; therefore, five warm-up epochs were considered sufficient.

Based on Tables 5.4 and 5.5, introducing a 5-epoch warm-up improves Specificity on average by approximately 13.6% and Sensitivity by about 4%, while Dice, PPV, and IoU decrease slightly by 1–2%. We interpret this as evidence that the warm-up phase allows the decoder to initially focus on learning the structure of the textual descriptions and forming stable cross-attention links to the visual embeddings, instead of immediately altering the MELD-derived visual features. This helps make the early training more stable and causes the model to predict more cautiously, which reduces over-segmentation and therefore increases Specificity.

Table 5.4: Different text types for **Exp3_mixed** models (values in parentheses indicate 95% confidence intervals). *Decoder open from the start* — GuideDecoder trained from the first epoch. Main cohort — correct descriptions.

Model	Text type	Dice	PPV pixels	(mean) PPV clusters	IoU	Specificity, % (N = 193)	Sensitivity, % (N = 259)
MELD	–	0.230 (0.107–0.320)	0.166 (0.086–0.220)	0.515	0.130 (0.057–0.190)	58 (n=112) (51.3–64.8)	65.6 (n=170) (59.8–71.4)
Exp3_mixed	hemi	0.235 (0.107–0.294)	0.222 (0.144–0.303)	0.511	0.133 (0.057–0.172)	79.3 (n=153) (73.6–85.0)	69.1 (n=179) (63.3–74.9)
Exp3_mixed	lobe_regions	0.243 (0.081–0.323)	0.246 (0.151–0.345)	0.491	0.138 (0.042–0.193)	79.3 (n=153) (73.6–85.0)	66.4 (n=172) (60.6–72.2)
Exp3_mixed	hemi+ lobe_regions	0.241 (0.097–0.312)	0.235 (0.152–0.346)	0.492	0.137 (0.051–0.185)	79.3 (n=153) (73.6–85.0)	68.7 (n=178) (62.9–74.1)
Exp3_mixed	hemi+lobe	0.241 (0.095–0.321)	0.233 (0.145–0.335)	0.505	0.137 (0.050–0.191)	79.3 (n=153) (73.6–85.0)	67.2 (n=174) (61.4–73.0)
Exp3_mixed	full_desc	0.251 (0.124–0.300)	0.256 (0.167–0.351)	0.479	0.144 (0.066–0.177)	79.3 (n=153) (73.6–85.0)	69.9 (n=181) (64.1–75.3)

Table 5.5: Different text types for **Exp3_mixed freeze 5 epochs** models (values in parentheses indicate 95% confidence intervals). *Decoder warm-up for 5 epochs* — GuideDecoder unfrozen after epoch 5. Main cohort — correct descriptions.

Model	Text type	Dice	PPV pixels	(mean) PPV clusters	IoU	Specificity, % (N = 193)	Sensitivity, % (N = 259)
MELD	–	0.230 (0.107–0.320)	0.166 (0.086–0.220)	0.515	0.130 (0.057–0.190)	58 (n=112) (51.3–64.8)	65.6 (n=170) (59.8–71.4)
Exp3_mixed	hemi	0.225 (0.131–0.294)	0.215 (0.137–0.325)	0.401	0.127 (0.070–0.172)	88.1 (n=170) (83.4–92.2)	73.7 (n=191) (68.3–79.2)
Exp3_mixed	lobe_regions	0.207 (0.123–0.330)	0.244 (0.160–0.307)	0.419	0.116 (0.065–0.197)	88.1 (n=170) (83.4–92.2)	71.8 (n=186) (66.4–77.2)
Exp3_mixed	hemi+ lobe_regions	0.219 (0.148–0.323)	0.216 (0.154–0.291)	0.392	0.123 (0.080–0.192)	88.1 (n=170) (83.4–92.2)	73.0 (n=189) (67.6–78.4)
Exp3_mixed	hemi+ lobe	0.228 (0.129–0.321)	0.247 (0.180–0.330)	0.482	0.129 (0.069–0.180)	88.1 (n=170) (83.4–92.2)	71.0 (n=184) (65.6–76.4)
Exp3_mixed	full_desc	0.230 (0.140–0.297)	0.204 (0.141–0.275)	0.342	0.130 (0.075–0.175)	88.1 (n=170) (83.4–92.2)	74.5 (n=193) (69.1–79.9)

Wrong textual descriptions (Main cohort)

We now evaluate models with *incorrect* prompts, including the `no_text` setting (i.e., the model receives the placeholder string *full_ brain* as input).

Across both Tables 5.6 and 5.7, models that were trained on correct textual descriptions but tested with intentionally incorrect ones still follow a consistent trend. While performance naturally decreases when the text contradicts the visual input, the extent of this drop varies across different types of "wrong" prompts. In many cases the degradation is modest, and in some settings the results remain comparable to–or even slightly better than–those obtained with correct descriptions. This behavior is plausible for several reasons:

1. **Partial but useful signal.** Even an inexact prompt still contains structure (e.g., hemisphere or a coarse lobe). This partial cue narrows the search region and reduces the need to scan the whole cortex.

2. **Robustness of the decoder.** Thanks to pretraining, the decoder tolerates mild text–image mismatch. When the text is unhelpful, it relies more on visual embeddings and uses the prompt only as a weak guide for attention.

3. **Regularization effect.** Slightly noisy text prevents the model from memorizing specific wording and forces it to rely on more general text–image relations. This acts as mild regularization: predictions become more conservative, reducing false positives (FP). Consequently, precision metrics (e.g., PPV) tend to increase, while recall and Dice typically remain stable.

Removing the text encoder does not substantially degrade performance, but the metrics do become consistently worse. Compared with models trained on correct descriptions, **Specificity** drops by roughly 27%. A similar pattern is observed for **Sensitivity**: the model without text detects approximately 1–2% fewer lesions relative to models using wrong prompts, and about 6–7% fewer relative to models using correct prompts.

Without any linguistic prior, the attention becomes broad and less informative. Under strong class imbalance, the model behaves more conservatively and fires only at very high confidence.

Table 5.6: Different text types with *wrong* descriptions (without freezing). Main cohort — wrong descriptions

Model	Text type	Dice	PPV pixels	(mean) PPV clusters	IoU	Specificity, % (N = 193)	Sensitivity, % (N = 259)
MELD	–	0.230 (0.107–0.320)	0.166 (0.086–0.220)	0.515	0.130 (0.057–0.190)	58 (n=112) (51.3–64.8)	65.6 (n=170) (59.8–71.4)
Exp3_mixed	wrong_hemi	0.234 (0.109–0.304)	0.217 (0.144–0.296)	0.509	0.133 (0.057–0.179)	79.3 (n=153) (73.6–85.0)	69.5 (n=180) (63.7–74.9)
Exp3_mixed	wrong_hemi + correct_lobe_regions	0.240 (0.097–0.312)	0.241 (0.152–0.346)	0.486	0.136 (0.051–0.185)	79.3 (n=153) (73.6–85.0)	67.6 (n=175) (61.8–73.4)
Exp3_mixed	wrong_hemi + correct_lobe	0.239 (0.088–0.317)	0.228 (0.146–0.332)	0.508	0.135 (0.046–0.188)	79.3 (n=153) (73.6–85.0)	67.6 (n=175) (61.8–73.4)
Exp3_mixed	wrong_hemi + wrong_lobe_regions	0.241 (0.098–0.320)	0.222 (0.147–0.343)	0.489	0.137 (0.052–0.190)	79.3 (n=153) (73.6–85.0)	68.0 (n=176) (62.2–73.7)
Exp3_mixed	wrong_hemi + wrong_lobe	0.243 (0.083–0.317)	0.234 (0.148–0.331)	0.498	0.139 (0.044–0.188)	79.3 (n=153) (73.6–85.0)	67.2 (n=174) (61.4–73.0)
Exp3_mixed	no_text	0.249 (0.080–0.296)	0.192 (0.127–0.289)	0.443	0.142 (0.042–0.174)	52.3 (n=101) (45.1–59.1)	66.0 (n=171) (60.2–71.8)

A similar effect is observed in Table 5.7. Even when the model is trained on deliberately incorrect textual descriptions, freezing the decoder for the first 5 epochs again leads to a performance gain. This can be clearly seen in all cases except for the `no_text` baseline, where **Sensitivity**

decreases by about 14%, while for the wrong-text models **Sensitivity** still improves by roughly 2%.

Table 5.7: Different text types with *wrong* descriptions (with 5 frozen epochs). Main cohort — wrong descriptions

Model	Text type	Dice	PPV pixels	(mean) PPV clusters	IoU	Specificity, % (N = 193)	Sensitivity, % (N = 259)
MELD	–	0.230 (0.107–0.320)	0.166 (0.086–0.220)	0.515	0.130 (0.057–0.190)	58 (n=112) (51.3–64.8)	65.6 (n=170) (59.8–71.4)
Exp3_mixed	wrong_hemi	0.231 (0.119–0.287)	0.207 (0.139–0.303)	0.403	0.131 (0.063–0.167)	88.1 (n=170) (83.4–92.2)	73.7 (n=191) (68.3–79.2)
Exp3_mixed	wrong_hemi + correct_lobe_regions	0.216 (0.153–0.322)	0.216 (0.153–0.291)	0.395	0.121 (0.083–0.192)	88.1 (n=170) (83.4–92.2)	73.0 (n=189) (67.6–78.4)
Exp3_mixed	wrong_hemi + correct_lobe	0.222 (0.120–0.318)	0.243 (0.166–0.329)	0.479	0.125 (0.064–0.189)	88.1 (n=170) (83.4–92.2)	70.7 (n=183) (65.3–76.1)
Exp3_mixed	wrong_hemi+ wrong_lobe_regions	0.224 (0.135–0.330)	0.207 (0.149–0.298)	0.378	0.126 (0.072–0.198)	88.1 (n=170) (83.4–92.2)	74.1 (n=192) (68.7–79.5)
Exp3_mixed	wrong_hemi+ wrong_lobe	0.219 (0.114–0.331)	0.243 (0.164–0.346)	0.470	0.123 (0.060–0.198)	88.1 (n=170) (83.4–92.2)	71.0 (n=184) (65.6–76.4)
Exp3_mixed	no_text	0.245 (0.102–0.295)	0.205 (0.132–0.292)	0.384	0.140 (0.054–0.173)	38.3 (n=74) (31.6–45.1)	68.3 (n=177) (62.5–74.1)

Thus, the warm-up phase provides a consistent benefit: it prevents the decoder from immediately overfitting to noisy textual cues and encourages the model to extract whatever useful structural information is present, even when the text itself is incorrect.

Wrong textual descriptions (Independent cohort)

Table 5.8: Different text types for **Exp3_mixed** models (values in parentheses indicate 95% confidence intervals). *Decoder open from the start* — GuideDecoder trained from the first epoch. Independent cohort — correct descriptions.

Model	Text type	Dice	PPV pixels	(mean) PPV clusters	IoU	Specificity, % (N = 83)	Sensitivity, % (N = 82)
MELD	–	0.358 (0.209–0.465)	0.261 (0.142–0.428)	0.464	0.218 (0.117–0.303)	55.4 (n=46) (44.6–66.3)	69.5 (n=57) (59.8–79.3)
Exp3_mixed	hemi	0.430 (0.067–0.515)	0.429 (0.181–0.610)	0.604	0.274 (0.035–0.347)	89.2 (n=74) (81.9–95.2)	68.3 (n=56) (57.3–78.0)
Exp3_mixed	lobe_regions	0.374 (0.177–0.496)	0.486 (0.224–0.670)	0.656	0.230 (0.097–0.330)	89.2 (n=74) (81.9–95.2)	68.3 (n=56) (57.3–78.0)
Exp3_mixed	hemi+ lobe_regions	0.400 (0.222–0.501)	0.515 (0.242–0.668)	0.616	0.250 (0.125–0.334)	89.2 (n=74) (81.9–95.2)	69.5 (n=57) (59.8–79.3)
Exp3_mixed	hemi+lobe	0.380 (0.119–0.498)	0.515 (0.073–0.655)	0.613	0.234 (0.064–0.331)	89.2 (n=74) (81.9–95.2)	65.9 (n=54) (54.9–75.6)
Exp3_mixed	full_desc	0.402 (0.271–0.486)	0.515 (0.311–0.682)	0.626	0.251 (0.156–0.321)	89.2 (n=74) (81.9–95.2)	70.7 (n=58) (61.0–80.5)

On the independent cohort (Tables 5.8–5.11), for both correct and incorrect prompts, the model with a 5-epoch decoder freeze generally detects approximately 5% more lesions (higher **Sensitivity**). For most text–description settings, this increase in detected lesions does not substantially degrade segmentation quality: several segmentation metrics (**Dice**, **IoU**, and **PPV**) remain stable or even improve slightly compared to the non-frozen model. For the `hemi + lobe_regions` and `full_desc` prompts, we observe a noticeable decrease in segmentation quality (5–11% in Dice and 5–7% in IoU). However, these decreases are offset by a consistent improvement in lesion detection, which remains the primary objective in this setting.

We also compare two key text–description types, `hemi+lobe` and `hemi+lobe_regions`, under both training strategies (with and without decoder warm-up).

For models trained *without* decoder warm-up, the `hemi+lobe_regions` setting consistently produces higher segmentation quality: Dice and IoU improve by approximately 2-3%, and

Table 5.9: Different text types for **Exp3_mixed freeze 5 epochs** models (values in parentheses indicate 95% confidence intervals). *Decoder warm-up for 5 epochs* — GuideDecoder unfrozen after epoch 5. Independent cohort — correct descriptions.

Model	Text type	Dice	PPV pixels	(mean) PPV clusters	IoU	Specificity, % (N = 83)	Sensitivity, % (N = 82)
MELD	–	0.358 (0.209–0.465)	0.261 (0.142–0.428)	0.464	0.218 (0.117–0.303)	55.4 (n=46) (44.6–66.3)	69.5 (n=57) (59.8–79.3)
Exp3_mixed	hemi	0.368 (0.248–0.505)	0.406 (0.218–0.596)	0.489	0.225 (0.141–0.338)	94.0 (n=78) (88.0–98.8)	74.4 (n=61) (64.6–84.1)
Exp3_mixed	lobe_regions	0.335 (0.144–0.474)	0.338 (0.168–0.543)	0.486	0.201 (0.077–0.311)	94.0 (n=78) (88.0–98.8)	72.0 (n=59) (62.2–81.7)
Exp3_mixed	hemi+ lobe_regions	0.332 (0.183–0.477)	0.343 (0.174–0.576)	0.474	0.199 (0.101–0.313)	94.0 (n=78) (88.0–98.8)	75.6 (n=62) (65.9–84.1)
Exp3_mixed	hemi+lobe	0.400 (0.162–0.507)	0.462 (0.239–0.672)	0.586	0.250 (0.088–0.340)	94.0 (n=78) (88.0–98.8)	74.4 (n=61) (64.6–84.1)
Exp3_mixed	full_desc	0.305 (0.210–0.478)	0.326 (0.164–0.651)	0.306	0.180 (0.117–0.314)	94.0 (n=78) (88.0–98.8)	78.0 (n=64) (68.3–86.6)

Table 5.10: Different text types with *wrong* descriptions (without freezing). Independent cohort — wrong descriptions

Model	Text type	Dice	PPV pixels	(mean) PPV clusters	IoU	Specificity, % (N = 83)	Sensitivity, % (N = 82)
MELD	–	0.358 (0.209–0.465)	0.261 (0.142–0.428)	0.464	0.218 (0.117–0.303)	55.4 (n=46) (44.6–66.3)	69.5 (n=57) (59.8–79.3)
Exp3_mixed	wrong_hemi	0.421 (0.067–0.520)	0.432 (0.179–0.609)	0.604	0.267 (0.035–0.352)	89.2 (n=74) (81.9–95.2)	68.3 (n=56) (57.3–78.0)
Exp3_mixed	wrong_hemi + correct_lobe_regions	0.392 (0.214–0.503)	0.515 (0.240–0.675)	0.625	0.244 (0.120–0.336)	89.2 (n=74) (81.9–95.2)	69.5 (n=57) (59.8–79.3)
Exp3_mixed	wrong_hemi + correct_lobe	0.378 (0.119–0.492)	0.509 (0.073–0.658)	0.613	0.233 (0.063–0.326)	89.2 (n=74) (81.9–95.2)	65.9 (n=54) (54.9–75.6)
Exp3_mixed	wrong_hemi + wrong_lobe_regions	0.433 (0.258–0.515)	0.475 (0.256–0.676)	0.660	0.276 (0.148–0.346)	89.2 (n=74) (81.9–95.2)	69.5 (n=57) (59.8–79.3)
Exp3_mixed	wrong_hemi + wrong_lobe	0.388 (0.147–0.501)	0.481 (0.154–0.672)	0.617	0.241 (0.079–0.334)	89.2 (n=74) (81.9–95.2)	67.1 (n=55) (56.1–76.8)
Exp3_mixed	no_text	0.397 (0.138–0.502)	0.507 (0.151–0.647)	0.549	0.248 (0.074–0.335)	68.7 (n=57) (59.0–78.3)	65.9 (n=54) (54.9–75.6)

Sensitivity increases by approximately 2-4% compared to `hemi+lobe`. The same trend is seen in the evaluation of the incorrect text condition, where we compare `wrong_hemi + wrong_lobe` with `wrong_hemi + wrong_lobe_regions`.

However, for models trained *with* 5-epoch warm-up, the situation is the opposite. In this case, the `hemi+lobe` model provides higher segmentation accuracy, outperforming `hemi+lobe_regions` by approximately 3-7% in Dice and IoU, although with a slight decrease in Sensitivity (typically 1-2%).

Table 5.11: Different text types with *wrong* descriptions (with 5 frozen epochs). Independent cohort — wrong descriptions

Model	Text type	Dice	PPV pixels	(mean) PPV clusters	IoU	Specificity, % (N = 83)	Sensitivity, % (N = 82)
MELD	–	0.358 (0.209–0.465)	0.261 (0.142–0.428)	0.464	0.218 (0.117–0.303)	55.4 (n=46) (44.6–66.3)	69.5 (n=57) (59.8–79.3)
Exp3_mixed	wrong_hemi	0.368 (0.273–0.499)	0.382 (0.220–0.558)	0.471	0.225 (0.158–0.333)	94.0 (n=78) (88.0–98.8)	74.4 (n=61) (64.6–84.1)
Exp3_mixed	wrong_hemi + correct_lobe_regions	0.321 (0.163–0.456)	0.336 (0.152–0.545)	0.471	0.192 (0.089–0.296)	94.0 (n=78) (88.0–98.8)	74.4 (n=61) (64.6–84.1)
Exp3_mixed	wrong_hemi + correct_lobe	0.401 (0.129–0.507)	0.448 (0.240–0.676)	0.578	0.253 (0.069–0.340)	94.0 (n=78) (88.0–98.8)	74.4 (n=61) (64.6–84.1)
Exp3_mixed	wrong_hemi + wrong_lobe_regions	0.325 (0.196–0.458)	0.338 (0.205–0.619)	0.444	0.194 (0.109–0.297)	94.0 (n=78) (88.0–98.8)	74.4 (n=61) (64.6–84.1)
Exp3_mixed	wrong_hemi + wrong_lobe	0.353 (0.137–0.489)	0.374 (0.245–0.622)	0.551	0.214 (0.074–0.324)	94.0 (n=78) (88.0–98.8)	72.0 (n=59) (62.2–81.7)
Exp3_mixed	no_text	0.345 (0.108–0.453)	0.349 (0.156–0.567)	0.406	0.209 (0.058–0.292)	47.0 (n=39) (36.1–57.8)	64.6 (n=53) (53.7–74.4)

These results suggest that the warm-up strategy is preferable for practical use. In real clinical workflows, coarse anatomical labels are far more common than precise region-level descriptions,

and in such scenarios the warm-up model provides more reliable segmentation accuracy, even if this comes with a slight reduction in the number of detected lesions.

5.5 Linking MELD to GNN

In these experiments we investigated the effect of connecting different numbers of MELD feature stages to the GNN block. The rationale was that higher MELD stages may produce sparse representations, while lower stages provide richer local detail. By progressively adding stages from top to bottom, we aimed to evaluate how multi-stage integration influences model performance. To isolate this effect, the text encoder and GuideDecoder were disabled.

On the main cohort (Tables 5.12), the configuration with **three GNN layers** provides the most balanced performance across Dice, PPV_{pixels}, and IoU, while also detecting **one** additional lesion compared with the no-GNN baseline. This is the only configuration that consistently appears among the top performers for all primary metrics, which supports selecting it as the best overall trade-off.

Connecting **all seven** MELD stages to the GNN yields the highest lesion count (sensitivity increased approximately by 3%: 188/259 vs. 180/259), with only a modest drop in Dice and PPV_{pixels} (about 1–4%). This gain is consistent with SAGEConv's neighborhood aggregation: increasing depth expands each node's receptive field and multi-stage inputs inject complementary local/global context, enabling detection of additional lesion clusters that are missed with purely local features.

Table 5.12: Different number of MELD stages connected to the GNN block (values in parentheses indicate 95% confidence intervals). Main cohort

Experiment	Dice	PPV pixels	(mean) PPV clusters	IoU	Sensitivity, % (N = 259)
Exp1: 0 layers	0.240 (0.129–0.324)	0.217 (0.151–0.319)	0.532	0.136 (0.069–0.193)	69.5 (n=180) (63.7–74.9)
Exp1: 1 layer	0.235 (0.096–0.313)	0.235 (0.148–0.362)	0.603	0.133 (0.051–0.186)	67.2 (n=174) (61.4–73.0)
Exp1: 2 layers	0.234 (0.126–0.334)	0.219 (0.136–0.371)	0.573	0.132 (0.067–0.200)	68.3 (n=177) (62.5–74.1)
Exp1: 3 layers	0.249 (0.114–0.332)	0.224 (0.142–0.341)	0.479	0.142 (0.060–0.199)	69.9 (n=181) (64.1–75.3)
Exp1: 4 layers	0.231 (0.097–0.304)	0.180 (0.128–0.283)	0.540	0.130 (0.051–0.179)	68.3 (n=177) (62.5–74.1)
Exp1: 5 layers	0.223 (0.081–0.312)	0.172 (0.090–0.261)	0.533	0.125 (0.042–0.185)	66.0 (n=171) (60.2–71.8)
Exp1: 6 layers	0.208 (0.121–0.312)	0.219 (0.147–0.336)	0.539	0.116 (0.065–0.185)	69.1 (n=179) (63.3–74.9)
Exp1: 7 layers	0.206 (0.119–0.293)	0.219 (0.135–0.341)	0.436	0.115 (0.063–0.172)	72.6 (n=188) (67.2–78.0)

On the independent (Bonn) cohort Table 5.13, the **5-layer** configuration gives the best overall trade-off: it attains the *second-highest* Dice and IoU while detecting the most lesions (63/82). Compared with the no-GNN baseline (0 layers), the gap is ≈3% in Dice and ≈5% in IoU, highlighting the benefit of the proposed GNN block.

For subsequent experiments we keep the **3-layer** and **5-layer** variants, and will compare them in the final study to determine the optimal number of connected MELD stages.

Table 5.13: Different number of MELD stages connected to the GNN block (values in parentheses indicate 95% confidence intervals). Independent cohort (Bonn Dataset)

Experiment	Dice	PPV pixels	(mean) PPV clusters	IoU	Sensitivity, % (N = 82)
Exp1: 0 layers	0.371 (0.289–0.520)	0.438 (0.246–0.660)	0.680	0.227 (0.169–0.351)	73.2 (n=60) (63.4–82.9)
Exp1: 1 layer	0.342 (0.083–0.515)	0.528 (0.242–0.757)	0.819	0.206 (0.043–0.347)	65.9 (n=54) (54.9–75.6)
Exp1: 2 layers	0.383 (0.216–0.506)	0.432 (0.243–0.663)	0.729	0.237 (0.123–0.339)	70.7 (n=58) (61.0–80.5)
Exp1: 3 layers	0.390 (0.302–0.517)	0.503 (0.309–0.732)	0.619	0.242 (0.178–0.349)	75.6 (n=62) (65.9–84.1)
Exp1: 4 layers	0.458 (0.279–0.600)	0.491 (0.186–0.582)	0.653	0.297 (0.162–0.428)	73.2 (n=60) (63.4–82.9)
Exp1: 5 layers	0.432 (0.303–0.549)	0.410 (0.249–0.606)	0.555	0.275 (0.179–0.378)	76.8 (n=63) (67.1–85.4)
Exp1: 6 layers	0.382 (0.292–0.544)	0.584 (0.342–0.687)	0.644	0.236 (0.171–0.373)	73.2 (n=60) (63.4–82.9)
Exp1: 7 layers	0.402 (0.284–0.493)	0.375 (0.232–0.634)	0.451	0.252 (0.165–0.327)	74.4 (n=61) (64.6–84.1)

5.6 Final Experiments

By summarizing the results from the previous chapters, we conclude that the most appropriate text model is PubMedBERT, the optimal configurations for the visual branch are GNN encoders with 3 and 5 blocks, and the best training strategy is to unfreeze the pretrained Exp1 decoder after 5 epochs. All these parameters will be used in the following experiments to observe the resulting improvements.

For these experiments, we will use the following types of text prompts: "hemisphere" and "hemisphere + lobe". Using "lobe regions" and "full text" is not meaningful, as discussed earlier. We have also included a "no text" column, where instead of an empty string, this column will contain a general description: "full brain". Additionally, we will train the model with mixed text, primarily to reduce the overall experimental runtime, since our goal here is to demonstrate relative improvements under optimal settings, rather than to obtain the best possible model.

The architecture used in this experiment for Exp1 is shown below:

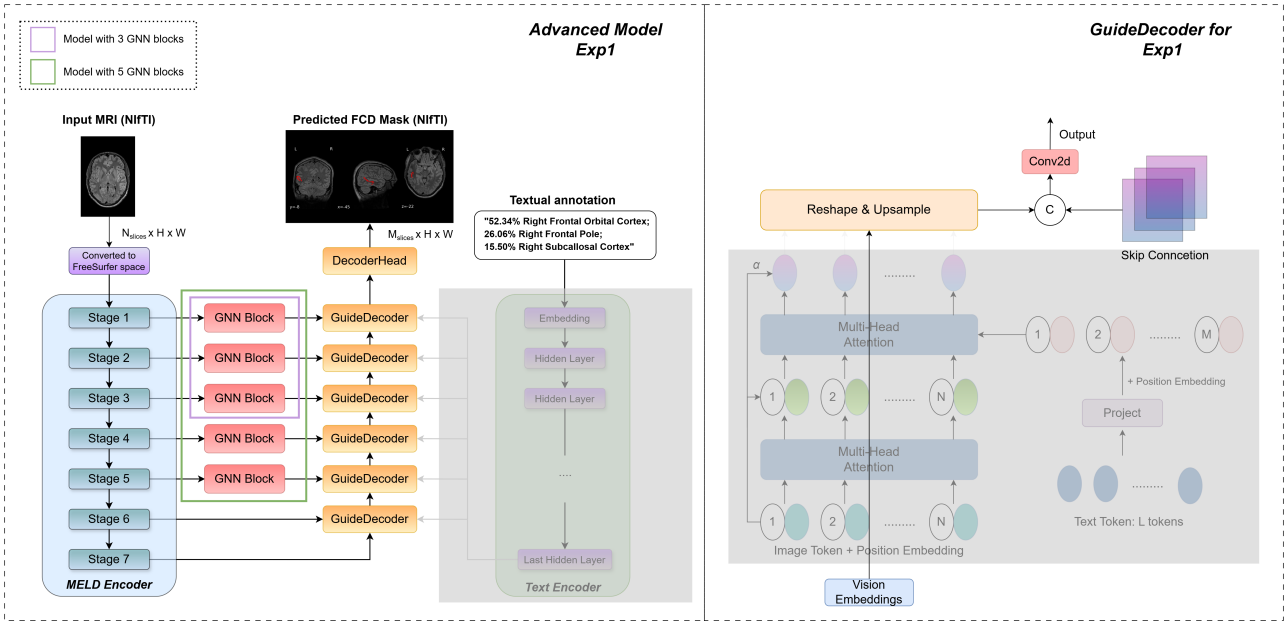

Figure 5.9: Architecture of Advanced Exp1: MELD surface-based features are extracted through a multi-stage GNN encoder (3- or 5-block variants) and passed through a hierarchical GuideDecoder with cross-attention to text embeddings. At each decoder stage, visual features are fused with linguistic cues, reshaped, and upsampled through geometry-aware operations, before producing the final segmentation via the DecoderHead. The right panel illustrates the internal structure of the GuideDecoder layer, including self- and cross-attention over text and vision tokens, skip connections, and spatial upsampling.

On the main cohort (Table 5.14), in the *hemisphere + lobe* and *no_ text* settings, the models with GNN blocks indeed detect a larger number of lesions. However, in most cases the model without a GNN block exhibits higher values across all primary metrics and clearly dominates in terms of *Specificity*.

The $\text{PPV}_{\text{clusters}}$ metric explicitly indicates that GNN-based models produce substantially more false-positive clusters, which negatively affects the overall prediction quality. Therefore, based on the obtained results, no definitive advantage of information aggregation in sparse graphs can be concluded.

Table 5.14: Median performance on the main cohort (with 95% confidence intervals).

Model	Text type	Deocder type	Dice	PPV pixels	(mean) PPV clusters	IoU	Specificity, % (N = 193)	Sensitivity, % (N = 259)
MELD	–	Basice Exp1	0.230 (0.107–0.320)	0.166 (0.086–0.220)	0.515	0.130 (0.057–0.190)	58.0 (n=112) (51.3–64.8)	65.6 (n=170) (59.8–71.4)
Exp3_mixed	hemisphere	Exp1 with 3 GNN	**0.232** (0.141-0.312)	0.212 (0.146-0.330)	0.314	**0.131** (0.076-0.185)	8.3 (n=16) (4.7–12.4)	69.9 (n=181) (64.1–75.3)
		Exp1 with 5 GNN	0.211 (0.110–0.320)	**0.226** (0.154–0.394)	0.242	0.118 (0.058–0.191)	1.0 (n=2) (0.0–2.6)	72.2 (n=187) (66.8–77.6)
		Basic Exp1	0.225 (0.131–0.294)	0.215 (0.137–0.325)	**0.401**	0.127 (0.070–0.172)	**88.1 (n=170)** (83.4–92.2)	**73.7 (n=191)** (68.3–79.2)
Exp3_mixed	hemisphere + lobe	Exp1 with 3 GNN	0.224 (0.154-0.319)	0.200 (0.142-0.314)	0.310	0.126 (0.083-0.190)	8.3 (n=16) (4.7–12.4)	69.5 (n=180) (63.7–74.9)
		Exp1 with 5 GNN	0.186 (0.119–0.314)	0.241 (0.158–0.384)	0.246	0.103 (0.063–0.186)	1.0 (n=2) (0.0–2.6)	**71.8 (n=186)** (66.4–77.2)
		Basic Exp1	**0.228** (0.129–0.321)	**0.247** (0.180–0.330)	**0.482**	**0.129** (0.069–0.180)	**88.1 (n=170)** (83.4–92.2)	71.0 (n=184) (65.6–76.4)
Exp3_mixed	no text	Exp1 with 3 GNN	0.212 (0.120-0.311)	**0.262** (0.164-0.355)	0.206	0.119 (0.064-0.184)	0.0 (n=0) (0.0–0.0)	**72.2 (n=187)** (66.8–77.6)
		Exp1 with 5 GNN	0.229 (0.118–0.311)	0.259 (0.135–0.363)	0.235	0.130 (0.063–0.184)	1.0 (n=2) (0.0–2.6)	69.5 (n=180) (63.7–74.9)
		Basic Exp1	**0.249** (0.080–0.296)	0.192 (0.127–0.289)	**0.443**	**0.142** (0.042–0.174)	**52.3 (n=101)** (45.1–59.1)	66.0 (n=171) (60.2–71.8)

However, on the independent (Bonn) cohort (Tables 5.15), we observe that for the *hemisphere* and *hemisphere + lobe* settings, the model with **three GNN blocks** achieves the highest Dice, IoU, and Sensitivity. However, its $PPV_{clusters}$ is almost twice as low as that of the baseline without a GNN block, indicating a substantially larger number of false-positive clusters. This is further reflected in the Specificity, which is approximately three times lower than the baseline and nearly two orders of magnitude lower compared with the 5-GNN configuration.

These findings suggest that the number of GNN blocks must be selected with caution: deeper aggregation can cause feature oversmoothing, which may negatively affect cluster-level precision, as clearly seen in the 5-GNN case. In the *no-text* setting, GNN-based models primarily increase the number of detected lesions, but do so at the cost of all remaining metrics, indicating reduced prediction quality.

Table 5.15: Median performance on the independent cohort (with 95% confidence intervals).

Model	Text type	Deocder type	Dice	PPV pixels	(mean) PPV clusters	IoU	Specificity, % (N = 83)	Sensitivity, % (N = 82)
MELD	–	–	0.358 (0.209–0.465)	0.261 (0.142–0.428)	0.464	0.218 (0.117–0.303)	55.4 (n=46) (44.6–66.3)	69.5 (n=57) (59.8–79.3)
Exp3_mixed	hemisphere	Exp1 with 3 GNN	**0.446** (0.289–0.553)	0.404 (0.228–0.605)	0.294	**0.288** (0.169–0.382)	33.7 (n=28) (24.1–44.6)	**78.0 (n=64)** (68.3–86.6)
		Exp1 with 5 GNN	0.378 (0.274–0.549)	**0.418** (0.237–0.667)	0.186	0.233 (0.159–0.379)	1.2 (n=1) (0.0–3.6)	75.6 (n=62) (65.9–84.1)
		Basic Exp1	0.368 (0.248–0.505)	0.406 (0.218–0.596)	**0.489**	0.225 (0.141–0.338)	**94.0 (n=78)** (88.0–98.8)	74.4 (n=61) (64.6–84.1)
Exp3_mixed	hemisphere + lobe	Exp1 with 3 GNN	**0.412** (0.302–0.565)	0.388 (0.217–0.567)	0.299	**0.260** (0.178–0.394)	33.7 (n=28) (24.1–44.6)	**78.0 (n=64)** (68.3–86.6)
		Exp1 with 5 GNN	0.379 (0.259-0.541)	0.408 (0.220-0.666)	0.188	0.233 (0.148-0.371)	1.2 (n=1) (0.0–3.6)	75.6 (n=62) (65.9–84.1)
		Basic Exp1	0.400 (0.162–0.507)	**0.462** (0.239–0.672)	**0.586**	0.250 (0.088–0.340)	**94.0 (n=78)** (88.0–98.8)	74.4 (n=61) (64.6–84.1)
Exp3_mixed	no text	Exp1 with 3 GNN	0.299 (0.161-0.431)	0.385 (0.193-0.711)	0.202	0.176 (0.088-0.274)	1.2 (n=1) (0.0–3.6)	72.0 (n=59) (62.2–81.7)
		Exp1 with 5 GNN	0.342 (0.242-0.464)	0.376 (0.215-0.676)	0.191	0.206 (0.138-0.302)	0.0 (n=0) (0.0–0.0)	**74.4 (n=61)** (64.6–84.1)
		Basic Exp1	**0.397** (0.138–0.502)	**0.507** (0.151–0.647)	**0.549**	**0.248** (0.074–0.335)	**68.7 (n=57)** (59.0–78.3)	65.9 (n=54) (54.9–75.6)

Overall, the results show that additional message passing can indeed increase sensitivity for sparse graphs by capturing more lesion candidates, but this benefit comes with a pronounced increase in false positives. Therefore, while deeper aggregation offers potential advantages, it also highlights the need for further refinement of GNN-based feature integration to avoid sacrificing precision.

5.7 General conclusions

- From the **Basic Experiments** chapter, we conclude that:
  - **Exp3_hemi+lobe_regions** provides the most balanced performance and achieves the **highest sensitivity**. Incorporating hemisphere and lobe-region information constrains the search space and improves localization.
  - **Exp3_hemi+lobe+PubMedBERT** achieves the best overall trade-off on both cohorts. The concise prompt without redundant context, combined with PubMedBERT's domain-specific pretraining, consistently improves all metrics, highlighting its influence and importance for the final results.
  - **Exp3_full_desc** underperforms relative to the hemi+lobe variants, suggesting that long atlas-style descriptions introduce noise and redundancy. In contrast, shorter and more targeted prompts generalize better.

- From the **Mixed Text** chapter, we conclude that training the model on mixed-type descriptions yields better results than the **MELD** baseline, but remains slightly worse than models trained on a single, well-specified prompt type. Notably, replacing correct descriptions with incorrect ones does not collapse performance, demonstrating the robustness and stability of the proposed architecture.

  Furthermore, freezing the decoder for the first several epochs allows the model to focus on aligning vision and text embeddings before full end-to-end training. This warm-up phase helps the model learn the contextual structure of textual prompts more reliably, ultimately leading to more detected lesions and a higher number of correctly classified healthy patients.

- From the **Linking MELD to GNN** chapter, we conclude that adding a GNN block that aggregates information from neighboring nodes helps the model accumulate contextual information more effectively. This leads to improved performance and higher sensitivity. In future work, integrating this block into the final architecture may further enhance results.

- From the **Final Experiments** chapter, we conclude that adding a GNN block ultimately helps the model detect more lesions. However, it also increases the number of false-positive predictions, which negatively affects the overall quality. Therefore, the idea of aggregating information within sparse graphs requires further refinement.

- From the **Text Distribution** chapter (see Appendix), we conclude that conditioning on fine-grained region names as textual inputs tends to induce overfitting, because the underlying label distribution is heavily imbalanced and long-tailed. Frequent categories dominate the learning signal, while numerous rare classes lack sufficient variability for robust generalization.

- From the **Semantic Similarity Analysis of Frontal Lobe Regions** chapter (see Appendix), we conclude that the choice of text encoder is crucial. Biomedical language models pretrained on domain-specific corpora (e.g., PubMedBERT, RadBERT, BioClinicalBERT) produce higher semantic coherence between anatomically related brain regions, whereas general-purpose models exhibit inconsistent similarity patterns. This shows that

semantically aligned text embeddings can improve multimodal fusion and ultimately enhance the model's interpretability and performance.

6 Discussion

The experiments conducted in this thesis demonstrate that incorporating textual information into surface-based GNN architectures can substantially improve the detection of FCD type II. At the same time, the results highlight important trade-offs that need to be considered when designing multimodal models.

A more detailed analysis of the GNN experiments reveals a nuanced picture. Integrating additional MELD feature stages into the GNN block does increase the model's ability to detect lesions: sensitivity and lesion-level coverage improved as more stages were incorporated, indicating that neighborhood aggregation can amplify subtle abnormal patterns that may be missed by purely local representations. This benefit, however, comes at the cost of reduced precision and a higher rate of false positives.

Importantly, the results also show that adding more GNN layers does not monotonically improve performance. The effectiveness of GNN propagation depends strongly on the structure of the underlying MELD features. For high-dimensional stages, where the surface graph is relatively sparse, message passing helps stabilize features and enrich contextual information, making GNN layers particularly beneficial. In contrast, for low-dimensional stages (such as stages 6–7), where the graph becomes denser and the representations more compressed, additional propagation tends to oversmooth node features. This blurs spatial distinctions and can ultimately degrade performance rather than enhance it.

Experiments with different forms of textual input demonstrated that even coarse labels (hemisphere or lobe only) meaningfully improved segmentation compared to purely vision-based baselines. Full atlas descriptions increased sensitivity further but produced unstable predictions, generating many false positive clusters. Thus, reduced-text settings may provide the best balance for clinical application.

Several limitations must be acknowledged. The dataset size remains modest compared to other medical imaging benchmarks, and domain shifts across scanners and sites may limit generalizability. Furthermore, the analysis was restricted to type II FCD, and it remains unclear whether the same conclusions extend to other subtypes. Finally, the model relies on atlas-based text generated automatically; the integration of free-form radiology reports may further improve results.

In this study, we did not explore the effect of unfreezing different numbers of LLM layers or varying the number of cross-modal connections to the GuideDecoder. However, previous work (Huemann et al., 2024 [33]) shows that selectively unfreezing specific layers of a language model and fine-tuning them for a segmentation task can yield consistent performance gains. Consequently, systematic experiments on partial LLM fine-tuning, on alternative strategies for connecting the text encoder to the GuideDecoder, and on increasing the architectural capacity of the GuideDecoder itself represent promising directions for future research.

Another conceptual direction concerns the representation of anatomical descriptions. In this thesis, atlas information was encoded via textual prompts and processed by a pretrained language model. An alternative would be to express the same information as a numerical feature

vector with one dimension per cortical region. In such a vector, the percentages mentioned in the textual description (e.g., 52.34% Right Frontal Orbital Cortex; 26.06% Right Frontal Pole) would directly populate the corresponding entries, while all other regions would be set to zero. For coarse labels such as lobe-only or hemisphere-only descriptions, the corresponding weight could be uniformly distributed across all constituent regions.

While this vector-based formulation is simple, interpretable, and avoids the computational overhead of a pretrained LLM, it is also substantially less expressive. A model trained solely on this numerical representation has no inherent knowledge of anatomical relationships: each vector dimension is treated as an independent feature, and the network does not know which regions belong to the same lobe or hemisphere, which ones are spatially adjacent, or how they relate hierarchically. As a result, this representation reflects explicit numeric values, but does not reflect the underlying anatomical semantics. In contrast, language models encode prior knowledge about anatomical terminology, regional similarity, and hierarchical structure within the cortex, enabling them to generalise even when the textual descriptions are coarse or partially specified. Thus, although a numerical vector representation is a possible baseline, it is not clear in advance whether it would perform better or even as well as the LLM-based approach used in this work, making it a useful direction for future research.

Overall, the discussion of results highlights that textual guidance is a promising direction for FCD detection. However, the balance between sensitivity and precision remains a key design choice that must be adapted depending on whether the clinical task prioritizes coverage or specificity.

7 Conclusion

This thesis introduced a novel multimodal segmentation framework for detecting focal cortical dysplasia type II, combining surface-based visual representations with textual priors derived from anatomical atlases. The main findings are summarized below:

- Incorporating textual features consistently improved segmentation performance over vision-only baselines, demonstrating the value of multimodal fusion in low-data medical imaging scenarios.
- Aggregating MELD features using a GNN block increased the number of detected lesions but also introduced a higher rate of false positives. While this indicates that additional message passing can enhance sensitivity, it also shows that the current aggregation strategy requires further refinement to avoid degrading overall quality.
- Even coarse textual labels such as hemisphere- or lobe-level descriptions yielded substantial improvements. This suggests that fine-grained region names are not strictly necessary, and that lightweight, structured prompts may offer a more robust and generalizable alternative to detailed atlas-style descriptions.

Overall, the results demonstrate that multimodal architectures integrating language and surface-based vision are both feasible and beneficial for epileptogenic lesion detection under limited-data conditions. This work represents an initial step toward leveraging anatomical language priors in surface-based segmentation models.

Future research should explore training on larger and more heterogeneous cohorts, incorporating clinical radiology reports as richer textual supervision, and investigating alternative fusion strategies or graph-based aggregation mechanisms to improve robustness, reduce false positives, and enhance generalizability across clinical centers.

Appendix

HexUnpool

We implement surface upsampling with a custom `HexUnpool` operator. Given features $X \in \mathbb{R}^{B\times H\times N_{\text{from}}\times C}$ defined on a coarse icosphere with N_{from} vertices, the operator produces an upsampled tensor $X' \in \mathbb{R}^{B\times H\times N_{\text{to}}\times C}$ on a denser icosphere ($N_{\text{to}} > N_{\text{from}}$).

The icosphere hierarchy used in MELD is generated in a deterministic manner: each refinement step subdivides the triangular mesh, creating new vertices at midpoints of edges while preserving the spherical topology. For every fine-level vertex, the MELD preprocessing pipeline provides a precomputed list of its *parent* vertices on the coarser level. These mappings are stored in the accompanying icosphere files (e.g., `coords`, `neighbours`, `spirals`, `t_edges`) and define how features should be propagated across resolutions. We directly use these mappings to ensure geometrically consistent upsampling.

During upsampling, two types of vertices must be handled:

1. **Vertices inherited from the coarse mesh.** Their features are copied directly to the corresponding positions in X'.

2. **Newly introduced vertices.** Each fine-level vertex v corresponds to one or more coarse-level *parent* vertices. Let $\mathcal{I}(v)$ denote the (predefined) set of indices of parent vertices from which v was generated during mesh refinement. These are not geometric neighbors on the fine mesh but a fixed upsampling correspondence determined by the icosphere construction. The feature of v is computed as the mean of its parents:

$$X'_v \;=\; \frac{1}{|\mathcal{I}(v)|}\sum_{u\in\mathcal{I}(v)} X_u.$$

This yields a smooth and topology-consistent interpolation of features from the coarse surface to the finer one. Since the mapping $\mathcal{I}(v)$ is deterministically defined by the MELD icosphere hierarchy, `HexUnpool` performs no learnable interpolation and does not introduce artifacts or distortions into the surface structure.

Semantic similarity analysis

Table 1 reports cosine similarities between the term "frontal lobe" and several related or distinct brain-region terms across multiple biomedical language models. In general, models pretrained on large biomedical or radiological corpora (e.g., BioClinicalBERT, PubMedBERT, RadBERT-RoBERTa) show more coherent behaviour across semantically related regions.

When comparing regions that are subsets of the frontal lobe ("prefrontal cortex", "inferior frontal gyrus") with lateral or orientation-only terms ("right", "right frontal"), we observe the following pattern:

Table 1: Cosine similarity between "frontal lobe" and related region terms across different biomedical language models.

Model	Prefrontal cortex	Inferior frontal gyrus	Right	Right frontal	Temporal lobe	Parietal lobe
bionlp/bluebert_pubmed_mimic_uncased_L-12_H-768_A-12	0.882	0.812	0.792	0.904	0.968	0.892
emilyalsentzer/Bio_ClinicalBERT	0.909	0.877	0.781	0.941	0.982	0.932
microsoft/BiomedNLP-PubMedBERT-base-uncased-abstract-fulltext	0.974	0.972	0.931	0.971	0.985	0.987
StanfordAIMI/RadBERT	0.504	0.431	0.369	0.654	0.816	0.599
zzxslp/RadBERT-RoBERTa-4m	0.971	0.946	0.921	0.960	0.971	0.976
microsoft/BiomedVLP-CXR-BERT-general	0.489	0.623	0.572	0.680	0.770	0.810
cambridgeltl/SapBERT-from-PubMedBERT-fulltext	0.762	0.587	0.360	0.689	0.577	0.703
allenai/scibert_scivocab_uncased	0.903	0.923	0.748	0.917	0.946	0.976
intfloat/e5-base	0.893	0.899	0.771	0.892	0.892	0.889

- frontal-lobe subsets show high similarity to "frontal lobe" (average 0.81 for "prefrontal cortex" and 0.79 for "inferior frontal gyrus");
- the direction-only term "right" is consistently lower (around 0.69);
- the compound term "right frontal"—which shares the lexical component "frontal"—shows higher similarity (around 0.85).

These results indicate that biomedical language models primarily capture lexical overlap and contextual co-occurrence rather than true anatomical relationships. Terms that share components or frequently appear in similar scientific contexts tend to form closely clustered embeddings, even when they refer to distinct anatomical regions. For example, any expression containing "frontal" is placed near "frontal lobe" due to shared vocabulary and usage patterns. Similarly, other major cortical lobes (e.g., temporal, parietal) also show relatively high similarity to "frontal lobe", reflecting a tendency of biomedical transformers to group broadly defined cortical structures. This behaviour is typical across models and should not be interpreted as anatomical grounding.

When selecting a model for our experiments, our aim is not to recover the underlying anatomical hierarchy but to identify a model that handles domain terminology in a stable and predictable manner. In particular, we prioritise models that capture semantic relatedness between terms that occur in similar linguistic contexts and that distinguish compound expressions from more general ones. Such stability in semantic patterns is sufficient for our downstream pipeline.

Based on these observations, **PubMedBERT**, **BioClinicalBERT** and **BlueBERT** demonstrate the most consistent and task-relevant behaviour and are therefore suitable for generating text embeddings in our framework. In contrast, **StanfordAIMI/RadBERT**, **BiomedVLP-CXR-BERT-general**, and **SapBERT** exhibit lower or less stable similarities among related terms and are consequently less suitable for our purposes.

Text distribution

To characterise potential sources of bias, we analyse the distribution of text-derived region labels at three anatomical levels: hemispheres, lobes, and lobe-regions. This reveals whether particular regions or hemispheres are overrepresented, which could drive overfitting or induce a preference for frequent anatomical terms during training.

Unless stated otherwise, all statistics are computed on the *entire* dataset (train, validation, and test). The label *No lesion detected* denotes healthy controls and is excluded from validation and test to avoid artificially inflating evaluation metrics. After this exclusion its effective frequency is roughly halved, although the overall imbalance across regions persists.

Hemisphere. The distribution of hemisphere labels is shown in Figure 1. In the table *"No lesion detected"* means these are healthy patients.

In the tables, *"No lesion detected"* denotes **healthy control** scans with no radiologically/annotationally confirmed FCD.

The number of cases for the left and right hemispheres is approximately equal, with only a small, statistically insignificant difference. This indicates that the *entire dataset* is well balanced across hemispheres. Consequently, a model trained on these data is unlikely to learn a systematic bias towards either hemisphere, which would otherwise reduce its ability to generalise.

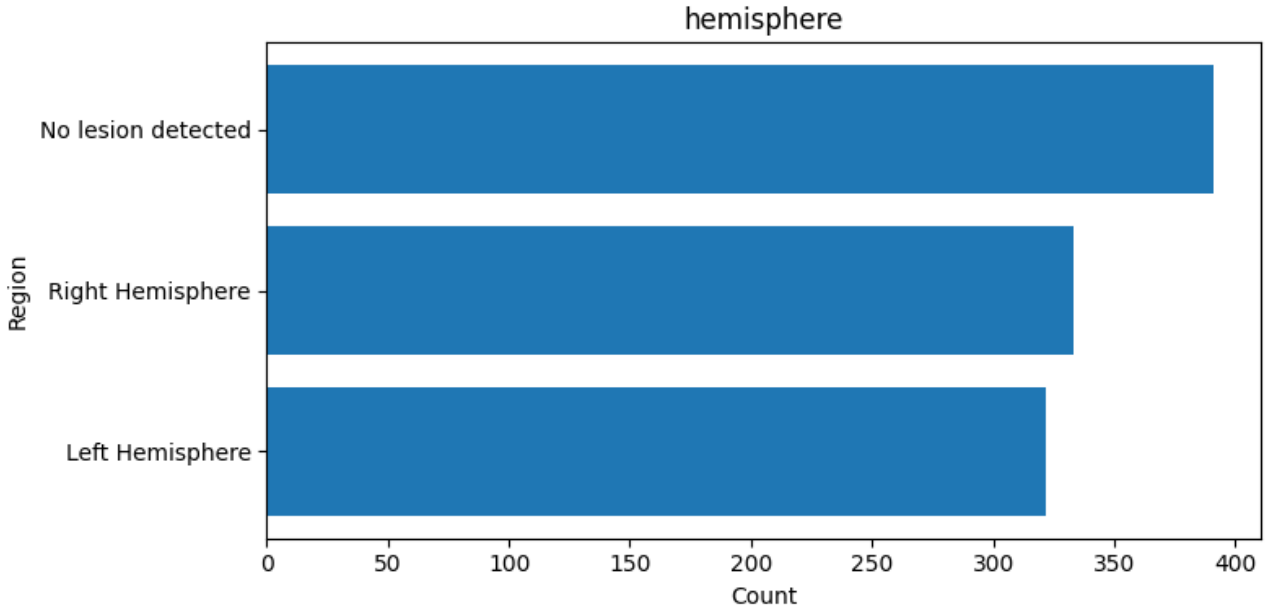

Figure 1: Hemisphere distribution

Lobes. The lobe-level distribution, presented in Figure 2, demonstrates a moderate imbalance. The *frontal*, *limbic*, and *parietal* lobes occur much more frequently than the *occipital*, *insular*, *subcallosal*, and especially the *brainstem*, which are relatively rare. Such imbalance may cause the model to "memorise" common patterns such as *frontal lobe* while underperforming on underrepresented categories like *brainstem* or *insula*. However, compared to the following plot, the lobe regons imbalance can be considered moderate and still suitable for training, especially if loss weighting or other balancing techniques are used.

Lobe regions. The third plot (Figure 3) provides the distribution for specific anatomical regions within the lobes. Here, the imbalance becomes much more pronounced. The label *No*

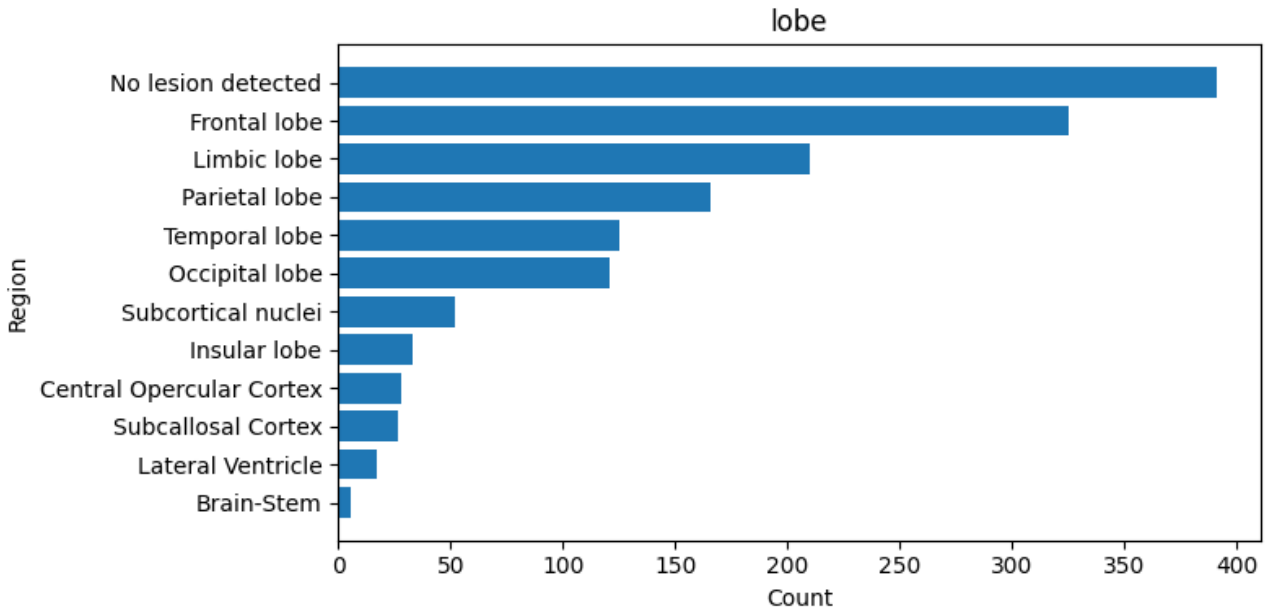

Figure 2: Lobe distribution

lesion detected dominates the dataset, with a frequency exceeding 400, whereas many regions appear only one to three times. This extremely long-tailed distribution suggests a risk of overfitting to the most frequent labels. As a result, the model may bias its *spatial* predictions toward regions whose textual labels are frequent in the training data (e.g., the *frontal gyrus*), assigning higher lesion probabilities there. This imbalance poses a risk for models relying on textual embeddings, since they may learn frequency-driven rather than semantically meaningful representations.

Discussion. From these findings, we conclude that using fine-grained region names directly as text input would likely lead to model overfitting, given the strong imbalance and the large number of rare categories. The extreme class skew among fine-grained lobe–region labels makes direct use of their verbatim names as text inputs ill-suited: models tend to memorise frequent terms and underfit rare ones, which harms generalisation. To further mitigate imbalance, one could consider data augmentation or class-weighted loss functions, at the cost of longer training and increased computational complexity. Nonetheless, such strategies would likely improve model robustness and generalisation performance.

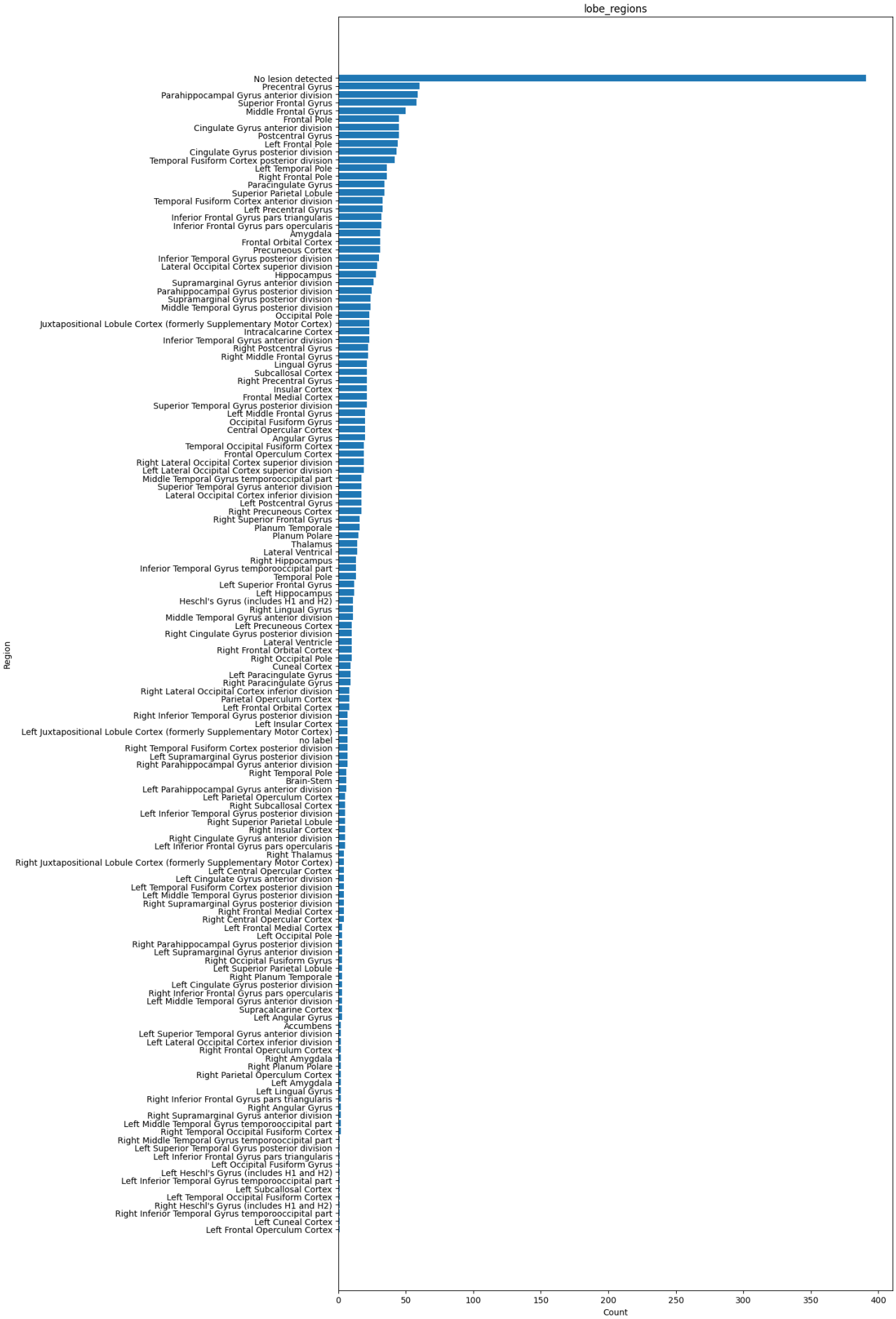

Figure 3: Lobe–region distribution

List of Acronyms

CT Computed Tomography

FCD Focal Cortical Dysplasia

GNN Graph Neural Network

IoU Intersection over Union

LLM Large Language Model

MELD Multicentre Epilepsy Lesion Detection

MRI Magnetic Resonance Imaging

PPV Positive Predictive Value

ROI Region of Interest

References

[1] R. Durgam, B. Panduri, V. Balaji, A. O. Khadidos, A. O. Khadidos, and S. Selvarajan. "Enhancing lung cancer detection through integrated deep learning and transformer models". In: *Scientific Reports* 15.1 (2025), p. 15614 (cit. on p. 1).

[2] A. B. Abdusalomov, M. Mukhiddinov, and T. K. Whangbo. "Brain tumor detection based on deep learning approaches and magnetic resonance imaging". In: *Cancers* 15.16 (2023), p. 4172 (cit. on p. 1).

[3] K. Sharma, Z. Uddin, A. Wadal, and D. Gupta. "Hybrid Deep Learning Framework for Classification of Kidney CT Images: Diagnosis of Stones, Cysts, and Tumors". In: *arXiv preprint arXiv:2502.04367* (2025) (cit. on p. 1).

[4] A. Mehmood, Y. Hu, and S. H. Khan. "A Novel Channel Boosted Residual CNN-Transformer with Regional-Boundary Learning for Breast Cancer Detection". In: *arXiv preprint arXiv:2503.15008* (2025) (cit. on p. 1).

[5] L. Walger et al. "Artificial intelligence for the detection of focal cortical dysplasia: Challenges in translating algorithms into clinical practice". In: *Epilepsia* 64.5 (2023), pp. 1093–1112 (cit. on p. 1).

[6] J. Wagner et al. "Morphometric MRI analysis improves detection of focal cortical dysplasia type II". In: *Brain* 134.10 (2011), pp. 2844–2854 (cit. on p. 1).

[7] K. M. B. Dev et al. "Automatic detection and localization of focal cortical dysplasia lesions in MRI using fully convolutional neural network". In: *Biomedical Signal Processing and Control* 52 (2019), pp. 218–225 (cit. on pp. 1, 6).

[8] P. M. House et al. "Automated detection and segmentation of focal cortical dysplasias (FCDs) with artificial intelligence: Presentation of a novel convolutional neural network and its prospective clinical validation". In: *Epilepsy Research* 172 (2021), p. 106594 (cit. on pp. 1, 6).

[9] M. Ripart, H. Spitzer, L. Z. Williams, L. Walger, A. Chen, A. Napolitano, et al. "Detection of epileptogenic focal cortical dysplasia using graph neural networks: a MELD study". In: *JAMA Neurology* 82.4 (2025), pp. 397–406 (cit. on pp. 1, 5, 6, 13, 17, 18, 21).

[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. "Attention Is All You Need". In: *NeurIPS*. 2017 (cit. on p. 4).

[11] A. e. a. Dosovitskiy. "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale". In: *ICLR* (2021) (cit. on p. 4).

[12] Z. e. a. Liu. "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows". In: *ICCV*. 2021 (cit. on p. 4).

[13] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini. "The Graph Neural Network Model". In: *IEEE Transactions on Neural Networks* 20.1 (2008), pp. 61–80 (cit. on p. 5).

[14] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl. "Neural Message Passing for Quantum Chemistry". In: *Proceedings of the 34th International Conference on Machine Learning*. 2017, pp. 1263–1272 (cit. on p. 5).

[15] T. N. Kipf and M. Welling. "Semi-Supervised Classification with Graph Convolutional Networks". In: *International Conference on Learning Representations (ICLR)* (2017) (cit. on p. 5).

[16] W. L. Hamilton, R. Ying, and J. Leskovec. "Inductive Representation Learning on Large Graphs". In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2017 (cit. on pp. 5, 10).

[17] O. Ronneberger, P. Fischer, and T. Brox. "U-Net: Convolutional Networks for Biomedical Image Segmentation". In: *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. Springer, 2015, pp. 234–241 (cit. on p. 5).

[18] H. Gao and S. Ji. "Graph U-Net". In: *Proceedings of the 36th International Conference on Machine Learning (ICML)*. Vol. 97. Proceedings of Machine Learning Research. PMLR, 2019, pp. 2083–2092 (cit. on p. 5).

[19] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *NAACL-HLT* (2019) (cit. on p. 6).

[20] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang. "BioBERT: a pre-trained biomedical language representation model for biomedical text mining". In: *Bioinformatics* 36.4 (2019), pp. 1234–1240 (cit. on p. 6).

[21] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. "Learning Transferable Visual Models From Natural Language Supervision". In: *Proceedings of the 38th International Conference on Machine Learning (ICML)*. Vol. 139. PMLR, 2021, pp. 8748–8763 (cit. on p. 6).

[22] Y. Zhong et al. "Ariadne's thread: Using text prompts to improve segmentation of infected areas from chest x-ray images". In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2023*. Cham: Springer Nature, 2023 (cit. on pp. 6, 8, 11).

[23] B. H. Park, M. J. Kim, B. S. Kim, H. S. Kim, Y. S. Kim, H. J. Kang, and H. Kim. "Artificial Intelligence for Diagnosis of Focal Cortical Dysplasia: A Narrative Review". In: *Radiology: Artificial Intelligence* 4.6 (2022), e210258 (cit. on p. 10).

[24] M. E. Peters, S. Ruder, and N. A. Smith. "To Tune or Not to Tune? Adapting Pretrained Representations to Diverse Tasks". In: *arXiv preprint arXiv:1911.03090* (2019) (cit. on p. 11).

[25] S. Gong, Y. Zhao, M. Liu, and J. Peng. "SpiralNet: A Fast and Robust Mesh Convolution Operator". In: *arXiv preprint arXiv:1905.00160* (2019) (cit. on p. 12).

[26] T. Rüber and et al. "An open presurgery MRI dataset of people with epilepsy and focal cortical dysplasia type II". In: *Scientific Data* 10.1 (2023), p. 475 (cit. on p. 16).

[27] J.-P. Fortin, N. Cullen, Y. I. Sheline, W. D. Taylor, I. Aselcioglu, P. A. Cook, P. Adams, C. Cooper, M. Fava, P. J. McGrath, et al. "Harmonization of cortical thickness measurements across scanners and sites". In: *NeuroImage* 167 (2018), pp. 104–120 (cit. on p. 17).

[28] B. Fischl. "FreeSurfer". In: *NeuroImage* 62.2 (2012), pp. 774–781 (cit. on p. 18).

[29] *MELD Graph Documentation*. https://meld-graph.readthedocs.io/en/latest/ (cit. on p. 18).

[30] L. Henschel, S. Conjeti, S. Estrada, K. Diers, B. Fischl, and M. Reuter. "FastSurfer: Fast and accurate deep learning based neuroimaging pipeline". In: *NeuroImage* 219 (2020), p. 117012 (cit. on p. 18).

[31] M. P. Notter, D. Gale, P. Herholz, R. Markello, M.-L. Notter-Bielser, and K. Whitaker. "AtlasReader: A Python package to generate coordinate tables, region labels, and informative figures from statistical MRI images". In: *Journal of Open Source Software* 4.34 (2019), p. 1257 (cit. on p. 20).

[32] *Harvard–Oxford Cortical and Subcortical Structural Atlases*. RRID:SCR_001476. FMRIB, University of Oxford & Harvard Center for Morphometric Analysis. Available via SciCrunch RRID resolver. 2014 (cit. on p. 20).

[33] Z. Huemann, X. Tie, J. Hu, and T. J. Bradshaw. "ConTEXTual net: a multimodal vision-language model for segmentation of pneumothorax". In: *Journal of Imaging Informatics in Medicine* 37.4 (2024), pp. 1652–1663 (cit. on p. 41).