An Interpretable Alzheimer's Disease Diagnosis Model via Gray Matter Attention-Guided Counterfactual Reasoning

Supplementary Material

1. Introduction

This supplementary document provides additional details to support the findings of the main paper, including data preprocessing, brain tissue segmentation, and the implementation and training procedures of the proposed model. These materials are provided to enhance the reproducibility and clarity of our approach.

2. Data Processing Details

This section describes the overall data processing pipeline, which consists of three main stages: data preprocessing, brain tissue segmentation, and data augmentation. Data preprocessing aims to standardize the input MRI scans and remove irrelevant structures, brain tissue segmentation extracts gray matter (GM), white matter (WM), and cerebrospinal fluid (CSF) regions for further analysis, and data augmentation enhances model generalization by introducing controlled variations into the training data. Detailed procedures for each stage are described below.

Data Preprocessing. In the data preprocessing stage, we first used FLIRT [59] provided by FSL 6.0 to register all T1-weighted sMRI scans in the MNI152 standard space. A 12-degree-of-freedom affine transformation was applied, with the correlation ratio employed as the similarity metric, in order to reduce spatial discrepancies caused by multi-center data and heterogeneous scanning protocols. Subsequently, skull stripping was performed using the HD-BET tool, which has demonstrated superior preservation of cortical structures in multi-center MRI data. To correct intensity inhomogeneities induced by magnetic field non-uniformities, we applied the N4BiasFieldCorrection algorithm from ANTs (four levels of iteration, 50 iterations per level, convergence threshold = 1×10^{-5}). Finally, all images were resampled to isotropic 1 mm^3 voxels and adjusted to a uniform volume size of $128\times128\times128$ to ensure spatial consistency of the input data during the gray matter segmentation process.

Brain Tissue Segmentation. To analyze the significant gray matter atrophy associated with AD pathology, we performed tissue segmentation on sMRI images using the FSL-FAST. This tool classifies brain tissues into three categories: gray matter (GM), white matter (WM), and cerebrospinal fluid (CSF). Our primary focus was on gray matter structures. The segmentation was executed with the following parameters: `fast -t 1 -n 3 -H 0.2 -I 5 -l 25.0 -g`, where `-t 1` specifies T1-weighted images, `-n 3` defines the number of tissue types, `-H` controls the bias field smoothness, and `-I` indicates the number of bias field iterations. After segmentation, a median filter with a radius of 1 was applied to remove noise, followed by Gaussian smoothing with a 4-mm full width at half maximum (FWHM) kernel to refine edges and enhance structural features of the gray matter. These preprocessing steps ensured higher-quality structural information for downstream modeling tasks.

Data Augmentation. During the training phase, we designed different data augmentation strategies for the diagnosis and explanation components to enhance their generalization capability. For the diagnosis component, a composite lightweight augmentation pipeline was employed, including random rotation (±10°), translation (up to 5% of the image size), scaling (ranging from [0.9, 1.1]), resizing to 110% of the original resolution followed by random cropping to the target size, and brightness/contrast perturbations (both adjusted within ±0.1, applied with a probability of 0.8). All images were finally converted into tensors and normalized to the range of [-1, 1]. For the explanation component, a relatively simplified augmentation scheme was used. The input images were first resized to the target dimensions and then subjected to brightness and contrast perturbations (within ±0.4, with a probability of 0.8), followed by tensor conversion and normalization to [-1, 1]. No random perturbations were introduced during the testing phase. Only resizing, center cropping, and normalization were applied to ensure stability and consistency throughout the evaluation process.

3. Implementation Environment

All experiments were conducted on a workstation equipped with two NVIDIA TITAN RTX GPUs (24 GB each), an Intel Xeon W-2133 CPU, and 128 GB of RAM, running Ubuntu 20.04 LTS. The deep learning framework used was PyTorch 1.12 with CUDA 12.2. Each experiment was repeated five times with different random seeds for statistical reliability, and the dataset split seed was fixed to 42 to ensure reproducibility.

4. Model Architecture and Training Details

The proposed model consists of two components: a diagnosis component and an explanation component. Both were trained using the AdamW optimizer, with hyperparameters β_1 and β_2 set to 0.5 and 0.99, respectively. The learning rate

scheduling followed a cosine annealing strategy, and early stopping with a patience value of 50 was employed to prevent overfitting. All experimental results are reported as the mean and standard deviation over five independent runs.

Diagnosis Component. Built upon a 3D ResNet-18 [15] backbone, the diagnostic component consists of two diagnostic branches, which are formed by separately incorporating the GMA and CFD modules into the backbone, respectively. During training, the batch size was set to 32, and the total number of training epochs was 200. The loss function combined a cross-entropy classification loss and a supervised contrastive (SupCon [21]) loss, with weights of 1.0 and 0.8, respectively. The initial learning rate was set to 1×10^{-4}, and the weight decay coefficient was 1×10^{-5}.

Explanation Component. The explanation component comprises a generator (G) and a discriminator (D). The generator G adopts a 3D conditional U-Net [40] architecture inspired by CycleGAN [61]. To enhance robustness, it follows the principles of diffusion models [17, 45] through multi-scale gaussian noise augmentation and a single-step denoising training strategy. The discriminator D is implemented as a 3D conditional Patch-GAN [18] and is trained using a standard adversarial loss to distinguish between real and generated images. During training, the batch size was set to 8, and training was conducted for 100 epochs. The total loss consisted of three components: adversarial loss, identity loss, and ACCC loss, weighted at 1.0, 1.0, and 10.0, respectively. Gradient clipping was applied with a threshold of 10.0 to ensure stable training. The initial learning rates for the generator and discriminator were set to 2×10^{-4} and 5×10^{-5}, respectively.