Deep Networks for Behavioral Variant Frontotemporal Dementia Identification from Multiple Acquisition Sources

Marco Di Benedetto[a,1,*], Fabio Carrara[a,1], Benedetta Tafuri[b,c], Salvatore Nigro[b,d], Roberto De Blasi[e], Fabrizio Falchi[a], Claudio Gennaro[a], Giuseppe Gigli[d,f], Giancarlo Logroscino[b,c], Giuseppe Amato[a], for the Frontotemporal Lobar Degeneration Neuroimaging Initiative[2]

[a] *Institute of Information Science and Technologies "Alessandro Faedo" (ISTI), National Research Council (CNR), Pisa (PI), Italy*
[b] *Center for Neurodegenerative Diseases and the Aging Brain, University of Bari "Aldo Moro", Tricase (LE), Italy*
[c] *Department of Basic Medicine, Neuroscience, and Sense Organs, University of Bari 'Aldo Moro', Bari (BA), Italy*
[d] *Institute of Nanotechnology (NANOTEC), National Research Council (CNR), Lecce (LE), Italy*
[e] *Department of Radiology, "Pia Fondazione Cardinale G. Panico", Tricase, Lecce (LE), Italy*
[f] *Department of Mathematics and Physics "Ennio De Giorgi", University of Salento, Campus Ecotekne, Lecce (LE), Italy*

Abstract

Behavioral variant frontotemporal dementia (bvFTD) is a neurodegenerative syndrome whose clinical diagnosis remains a challenging task especially in the early stage of the disease. Currently, the presence of frontal and anterior temporal lobe atrophies on magnetic resonance imaging (MRI) is part of the diagnostic criteria for bvFTD. However, MRI data processing is usually dependent on the acquisition device and mostly require human-assisted crafting of feature extraction. Following the impressive improvements of deep architectures, in this study we report on bvFTD identification using various classes of artificial neural networks, and present the results we achieved on classification accuracy and obliviousness on acquisition devices using extensive hyperparameter search. In particular, we will demonstrate the stability and generalization of different deep networks based on the attention mechanism, where data intra-mixing confers models the ability to identify the disorder even on MRI data in inter-device settings, i.e., on data produced by different acquisition devices and without model fine tuning, as shown from the very encouraging performance evaluations that dramatically reach and overcome the 90% value on the AuROC and balanced accuracy metrics.

Keywords: Medical Imaging, Behavioral Variant Frontotemporal Dementia, bvFTD, Machine Learning, Deep Learning, Neural Networks, Classification, Logistic Regression, Multi-Layer Perceptron, 3D Convolution, Transformer

1. Introduction

Frontotemporal lobar degeneration is the second most frequent cause of early onset dementia [1]. Behavioral variant of frontotemporal dementia (bvFTD) represents the most frequent phenotype [2, 3] and is associated with progressive behavioral impairment and changes in personality [4]. In the past years, evidences of frontotemporal atrophy on Magnetic Resonance Imaging (MRI) have been proposed as a useful biomarker to improve the specificity of bvFTD diagnosis, often difficult due to the clinical overlap with other neurodegenerative conditions and/or psychiatric disorders.

Several machine learning techniques have been applied to distinguish bvFTD from healthy controls (HC) using MRI-based features to define new imaging biomarkers in diagnostic criteria. Although these investigations showed moderate or high accuracy in the identification of bvFTD patients, however, most of these studies were conducted in small samples [5] and only one work considered an independent validation cohort [6], limiting the generalizability of the results. Moreover, the input features used in the classification models are often obtained by non-trivial images analyses making difficult to translate the results into clinical practice.

Deep learning overcomes some limitations about the preprocessing steps deeling with raw or semi-raw data and enable to explore the complexity of sample as much as possible. Recent findings suggest that the problem of differential diagnosis in the field of neurodegenerative disease [7, 8, 9] can be solved using deep network architectures thanks to its capability to explore MRI features in terms of major depth, width and inter-layer connections of the networks, extracting hierarchical features that represent

[*] Corresponding author - marco.dibenedetto@isti.cnr.it

[1] Authors contribute equally

[2] Data used in preparation of this article were obtained from the Frontotemporal Lobar Degeneration Neuroimaging Initiative (FTLDNI) database (http://4rtni-ftldni.ini.usc.edu/). The investigators at FTLDNI contributed to the design and implementation of FTLDNI and/or provided data, but did not participate in analysis or writing of this report (unless otherwise listed).

different levels of abstraction in a data-driven manner.

In this work, we study the potentiality of different preprocessing pipelines and deep network architectures for bvFTD identification. By analyzing the input 3D image only, we show that deep networks, especially the kind that intra-mix image features, offer significant outcomes and insights to bvFTD identification, providing a methodology able to reach and overcome the 90% value in both AuROC and balanced accuracy in *inter-device generalization* settings, i.e., on data produced by different acquisition devices and without model fine tuning.

In summary, our contributions are:

- the analysis of volume cropping, voxel normalization, and per-ROI processing on the performance of machine learning pipelines,
- the comparison of several deep learning models, from simple baselines to modern attention-based and data intra-mixing architectures, for classification from MR imaging, and
- the inspection of generalization capabilities and convergence stability among different runs and hyperparameter setups using datasets coming from multiple acquisition scanners.

We start by introducing previous works in Section 2, then we overview the schema of deep networks we used in Section 3. In Section 4 the datasets are presented, while in Section 5 we show our approach to the problem and discuss our experiments in Section 6. Results are reported in Section 7, and conclude in Section 8 with an insight on future directions.

2. Related Work

Machine learning techniques based on morphometric analysis have been widely used in order to find diagnostic biomarkers for bvFTD showing great potential as demonstrated by several recent scientific developments [5, 10, 11]. In particular, Moller et al. [6] and Mayer et al. [12] applied support vector machine (SVM) classification to predict diagnosis of bvFTD with high accuracy, respectively 85% in a whole brain setting on a separate test set for the first paper and of up to 84.6% in a ROI approach focusing on frontotemporal, insular regions, and basal ganglia in comparison with the whole brain approach. Bachli et al. [13] used a logistic regression classifier based on multimodal features such as cognitive scores (executive functions and cognitive screening) and brain atrophy measures (VBM from fronto-temporo-insular regions in bvFTD) to identify the most relevant characteristics in predicting the incidence of bvFTD respect to normal subjects. Testing the algorithm on different cohorts, they achieved an accuracy of up to 90%. A multimodal computational approach was also used by Donnelly-Kehoe et al. [14] to identify patients with bvFTD by analyzing sMRI and resting-state functional connectivity from 44 patients with bvFTD and 60 healthy controls (across three imaging centers with different acquisition protocols). The approach used by the authors achieved classification accuracy of 91% across all centers by exploiting site normalization, native space feature extraction, and a random forest classifier.

Despite the optimal results obtained with classical machine learning model, the existing techniques of differential diagnosis of bvFTD rely on some manual preprocessing of data like features extraction and selection expert-dependent. In the recent years some researchers have tried new implementations with deep learning approach that allows to overcome these problems in the differential diagnosis of neurodegenerative diseases [15, 16, 17, 18, 19]. Specifically, Gong et al. [20] used a lightweight fully convolutional neural network architecture to predict age from brain MRI scans in the Predictive Analytic Challenge (PAC) 2019. The dataset consisted of label-known training/validation datasets (2,638 subjects in total) and a "true" test set of 660 subjects whose labels were unknown to the competition participants. Spasov et al. [7] presented a novel deep learning architecture aiming at identifying mild cognitive impairment (MCI) patients who have a high likelihood of developing AD within 3 years. In this work, the developed deep learning procedures combined structural MRI, demographic, neuropsychological, and APOe4 genetic data as input measures. The convolutional neural network (CNN) employed fewer parameters than other deep learning architectures which significantly limited data-overfitting (550,000 network parameters, which is orders of magnitude lower than other network designs). Basaia et al. [8] presented a CNN with similar aim using combination of two database (an international database (ADNI) and your institutional set) to validate the results. Their approach provided a powerful tool for the automatic individual patient diagnosis along the AD continuum. Deep learning techniques have been also applied by Hu et al. [9] to solve the differential diagnosis problem of FTD and AD. In this study, the authors trained a deep neural network directly using raw T1 images (from two publicly available databases, i.e., the ADNI and the NIFD) to classify FTD, AD and corresponding NCs (normal controls), yielding an accuracy of 91.83% based on the most common T1-weighted sequence.

The work we present here offers very encouraging outcomes on the subject of bvFTD identification, with results that reach and dramatically overcome the 91.0% value of AuROC and balanced accuracy in *inter-device generalization* settings, that is, on data produced by different acquisition devices and without any fine-tuned training of the proposed models.

3. Classification and Deep Network Architectures

The problem of identifying the presence of bvFTD disease from a set of MR images is what is called a *classification task.* In Computer Science, the topic of classification consists in assigning a certain label to a particular input instance. Due to its complexity, this task is one of the most studied problems to which artificial intelligence research, and in particular the machine learning branch, have been involved since its beginning. Although lot of working solutions that rely on human-crafted features extracted from input data have been proposed (e.g., support vector machines [21, 22] or random forests [23]), we decide to focus our study on the de-facto superior performance hereby shown by deep artificial neural networks [24], exploiting their ability to learn more accurate representations from raw data.

In general, the proposed methods follow the standard binary classification pipeline : first the data is preprocessed (e.g., ad-hoc and statistical normalization), then feed to a specific classifier (in our case, a neural network), from whose output the final data label is extracted (e.g., by thresholding). In our scenario, we consider *three-dimensional* input images representing the patient's head volume, each regularly structured as a 3D grid of *volume elements*, or *voxels*, as an analogy to pixels in 2D images.

As shown in Section 6.1, we take into account several architectures, from simple regressors to more complex solutions like convolutional or attention-based networks, as introduced hereafter.

Logistic Regressor. As baseline, we consider binary logistic regression directly from voxels in which we model the following relation:

$$p(y = \text{bvFTD}|X) = \text{LinReg}(X, \Theta) = \sigma\left(\sum_{v\in\text{Vol}} w_v x_v + b\right), \tag{1}$$

where $\{x_v\}_{v\in\text{Vol}} = X \in \mathbb{R}^{D\times H\times W}$ is the input volume, $\Theta = \{w_v, b\}$ are the model parameters, and σ is the *sigmoid* function. In the neural network framework, this model can be seen as a single-neuron, single-layer network (e.g., *perceptron*) with sigmoid activation operating on the array of the flattened volume voxels.

Multi-Layer Perceptron (MLP). Multi-layer networks are the foundation of deep representation learning, as building a hierarchy of representations improves the ability to express and learn high-level patterns in data [25, 26]. Multi-Layer Perceptron (MLP) models consist of multiple layers of perceptrons interleaved with non-linear activations; the last layer can be adapted to produce the desired output — $p(y = \text{bvFTD}|X)$ in our scenario. Formally,

$$p(y = \text{bvFTD}|X) = \text{MLP}(X, \Theta) = \sigma(\mathbf{w}_o \cdot \mathbf{h}^{(L)} + b_o) \tag{2}$$

$$\mathbf{h}^{(i)} = \sigma(W^{(i)}\mathbf{h}^{(i-1)} + \mathbf{b}^{(i)}), \quad i \in [1, L] \tag{3}$$

$$\mathbf{h}^{(0)} = X, \tag{4}$$

where L is the network depth i.e. number of layers, $\Theta = \{W_i, \mathbf{b}_i, \mathbf{w}_L, b_L\}$ are the model parameters, $\mathbf{h}_i$ is the output of the i-th layer, σ is a non-linear function applied element-wise, and X is the input.

3D Convolutional Network. Convolutional networks are multi-layer deep networks particularly suitable for modelling spatial local properties in data. Indeed, they shine in recognition tasks with grid-structured data like images, audio, video, and also volumetric data [27]. For volumetric data, networks comprise 3D convolutions — an operation we can summarize as a sliding-window dot product between a small $k \times k \times k$ cubic kernel and the input volume. Each 3D convolution applies multiple kernels and thus produces a multi-channel volume collecting the results for each kernel and for each kernel position in the input space. A 3D convolutional network is defined as a cascade of 3D convolution layers interleaved with non-linear activation functions. Formally,

$$p(y = \text{bvFTD}|X) = \text{ConvNet3D}(X, \Theta) = \sigma(\mathbf{w}_o \cdot \mathbf{H}^{(L)} + b_o) \tag{5}$$

$$\mathbf{H}^{(i)} = \sigma(\text{Conv3D}(\mathbf{H}^{(i-1)}, \theta^{(i)})), \quad i \in [1, L] \tag{6}$$

$$\mathbf{H}^{(0)} = X, \tag{7}$$

where X is the input volume, $\theta^{(i)}$ are the weights of the i-th convolutional layer, $\mathbf{H}^{(i)}$ is the i-th intermediate volume, and $\mathbf{w}_o, b_o$ are the weights of the final linear layer.

Vision Transformer. As occur in natural language processing (NLP), identifying dependencies among words in a phrase is a key requirement for understanding the underlying semantic. To this end, researchers tried to express this interconnection by introducing the concept of recurrent processing within neural architectures, especially for sequence analysis with Recurrent Neural Networks [28] and Long short-term memory [29]. However, passing state between successive computation blocks amplified the gradient vanishing issue, thus reducing dependency propagation. To solve this problem, the *attention* mechanism was introduced in the form of an encoder-decoder network called *Transformer* [30]. In the encoder part, the idea is to enrich every item (e.g., words or *tokens*) of the input sequence with information coming from all other items. This *context augmentation* is provided by a sequential group of encoding blocks. More in detail, every input n-dimensional token t_i will first produce *query*, *key*, and *value* vectors with learned linear operations:

$$X = \begin{bmatrix} t_1 \\ \vdots \\ t_k \end{bmatrix}, \quad Q = XW^Q, \quad K = XW^K, \quad V = XW^V .$$

Then, the attention matrix is calculated as

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{n}}\right) V ,$$

where the $softmax$ output represents the scaled *score* matrix of all possible token pairs. By translating from NLP to image processing, Dosovitskiy et al. [31] extrapolated the encoder part and designed the *Vision Transformer* (ViT), an architecture that reinterprets the concept of image tokens [32] by translating it on a sequence of embedded *image patches* obtained by partitioning the image into a uniform grid of rectangular pixel cells, which are then used to feed the network. Given the generalization introduced by patches, this procedure can naturally be extended to any number of input dimensions (i.e., 3 in our context).

MLP-Mixer. Starting from the idea of patches introduced with ViT, the *MLP-Mixer* architecture [33] begins with tokens partitioned from the input, and conveys them on a set of cascaded layers that, basically, intermix the data as it occurs in cross attention modules. Differently from convolutional and attention mechanisms, the MLP-Mixer uses only multi-layer perceptrons layers to operate on tokens *directly* and *across* them. In particular, each layer is composed of two MLPs, where the first processes each token independently, and the second intermixes previous output in a linear operation. Amongst them, layer normalization and residual skips keep controlling the gradient flow:

$$\begin{aligned} U_{*,i} &= X_{*,i} + W_2\sigma(W_1 LayerNorm(X)_{*,i}), \quad \text{for } i = 1...C \\ Y_{j,*} &= U_{j,*} + W_4\sigma(W_3 LayerNorm(X)_{j,*}), \quad \text{for } j = 1...S \end{aligned}$$

where S is the number of the partitioning patches of the input image, and C is their *channel* dimensionality after token projection. The complexity of the network is linear with the number of input patches, as opposed to the quadratic complexity of the transformer architecture. This network typology proved to be surprisingly efficient in both quality, touching the state of the art generated by convolutional and attention models, and quantity, by generating a significative speed-up in throughput.

gMLP. By following the observation that a static parametrization introduced by an MLP can represent arbitrary functions, the *gMLP* architecture [34] simplifies the complex structure of a transformer by replacing the attention mechanism with a linear operation on a spatial input projection. The model structure is similar to the vision transformer and the MLP-Mixer, that is, a series of identical (but with independent weights) encoder blocks enclosed by an input tokenization and an output classifier. Each encoding block is composed as

$$Z = \sigma(XU), \quad \tilde{Z} = s(Z), \quad Y = \tilde{Z}V ,$$

where $X \in \mathbb{R}^{n \times d}$ is the input, σ is an activation function, and s is the *spatial gating unit*, defined as

$$s(Z) = Z_1 \odot (WZ_2 + b)$$

with $\odot$ indicating element-wise multiplication, and Z_1, Z_2 represent two independent partition of Z along the channel dimension.

3.1. A note on Data Intra-Mixing

In general, we can classify logistic regressor, MLP, and 3D convolution networks as *whole-data* models, that is, architectures where data is computed as a single lump in the processing pipeline. On contrast, Visual Transformer, MLP-Mixer, and gMLP partition data (spatially and/or per-channel) in so called *patches.* By putting patches in relation to each other (e.g., the attention mechanism [30]), Dosovitskiy et al. [31] demonstrated that the model can achieve state-of-the-art performances in classification tasks. We call this kind of interleaving as data *intra-mixing*, as referred to internal correlation of single data parts.

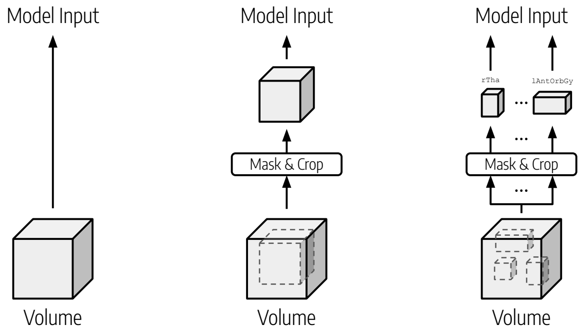
4. Multiple Source Datasets

One of the main goal of our research was to identify a deep learning model that was *robust* enough to identify bvFTD from data coming from *different* acquisition devices, so we focused our gathering of exemplars to two separate patient databases, each working with different MR scanners.

Participants. Data used in the preparation of this study were obtained from two different MRI datasets: the Frontotemporal Lobar Degeneration Neuroimaging Initiative (FTLDNI) database (for up-to-date information on participation and protocol, please visit [35]), and the Center for Neurodegenerative Diseases and the Aging Brain (CMND) database from the Department of Clinical Research in Neurology - University of Study "Aldo Moro" - Bari at Pia Foundation of Cult and Religion "Card. G.Panico".
The goals of the FTLDNI, funded through the National Institute of Aging, are to identify neuroimaging modalities and methods of analysis for tracking frontotemporal lobar degeneration (FTLD) and to assess the value of imaging versus other biomarkers in diagnostic roles. From this database we included 110 *healthy controls* (HC) and 50 bvFTD patients who had a valid structural T1-weighted MR images collected only at University of California, San Francisco (UCSF), the largest recruiting center, in order to avoid potential bias derived from different imaging protocol. MR images were acquired on a 3T Siemens Trio Tim

		FTLDNI		CMND	
		Train	*Test*	*Train*	*Test*
N°	HC	60	50	13	11
	bvFTD	30	20	16	14
Age (Years)	HC	62.4±7.7	63.2±7.1	63.21±5.91	
	bvFTD	61.3±7.5	61.3±6.8	68.23±7.65	
Sex (% Female)	HC	0.7	0.4	0.6	
	bvFTD	0.7	0.6	0.6	

Table 1: Datasets and splits. We used the *balanced average* metric to counter the imbalanced ratio of ill and healty patients. Statistical information is extracted from patients data at the source level.



(a) Whole-brain (b) Frontotemporal mask (c) ROIs

Figure 1: Preprocessing Pipelines. In (a) the whole volume is taken as input. In (b) and (c) regions are defined with fixed 3D binary masks: region volumes are cropped to their 3D bounding boxes, and the voxels outside the masked region(s) are cropped or set to zero. We name these three configurations as *None*, *FT*, and *ROI*, respectively.

system equipped with a 12-channel head coil at the UCSF Neuroscience Imaging Center, including whole-brain three-dimensional T1 MPRAGE (TR/TE = 2,300/2.9 ms, matrix = 240 × 256 × 160, isotropic voxels = 1 mm, slice thickness = 1 mm).

The second cohort was recruited between 2017 and 2019 at the CMND center. The dataset included 29 patients with bvFTD, diagnosed according to Rasckoscky [4] and 24 control subjects with valid MR images acquired on a 3T scanner (Philips Ingenia 3T) in the sagittal plane using a Fast-Field Echo (FFE) T1-weighted sequence. The FFE parameters were empirically optimized for gray-white contrast, with repetition time = 8.2 ms, echo time = 3.8 ms, flip angle = 8°, resolution = 256 × 256, slices = 200 and thickness = 1 mm.

The datasets have been stratifiedly splitted into subsets for training and test. Demographic information was reported in (Table 1). Before final training, we tuned model hyperparameters with a 5-fold evaluation process.

5. Method

Prior to classification, the structural MR imaging data were preprocessed with default settings of the CAT12 toolbox (Structural Brain Mapping Group, Jena University Hospital, Jena, Germany), including corrections for bias-field inhomogeneities, segmentation into gray matter (GM), white matter, and cerebrospinal fluid, followed by spatial normalization to the DARTEL template in MNI space (voxel size: 1.5 mm x 1.5 mm x 1.5 mm). Normalized images were modulated to guarantee that relative volumes were preserved following the spatial normalization procedure. Next, for Voxel-Based-Morphometry (VBM) purpose, the preprocessed GM data were smoothed with an 8mm full-width-half-maximum (FWHM) isotropic Gaussian kernel. An optimal gray matter mask was also generated from all smoothed images using the SPM12 Masking toolbox and the Luo–Nichols anti-mode method of automatic thresholding [36].

The 3D T1-weighted image for each subject were also segmented using the ROI analysis tool of CAT12 to extract regional masks, and a frontotemporal mask was created merging 60 Region-Of-Interest (ROI) from the Neuromorphometrics atlas (supplementary table SX) [37].

As stated before, our goal is to harness deep learning to build a predictor for bvFTD starting from voxel data coming from MRI. Given an input volume X, we model the probability of this volume belonging to the positive (bvFTD) group:

$$p(y = \text{bvFTD}|X) = f(X, \Theta)\,, \tag{8}$$

where f is learnable model parametrized by Θ.

We investigated several variations of the preprocessing pipeline depicted in Figure 1. We use two different preprocessed brain images as input for our models: normalized gray matter volumes (named *wm*), and modulated normalized gray matter volumes (named *mwp*). For each input volume, we explore three different setting for the definition of the voxels involved in the classification task: (i) whole-brain analysis in which all voxels in the gray matter volumes are considered as input to the network, (ii) a customised voxel-based analysis in which voxels belonging to the brain regions commonly affected in bvFTD are used as input to the network (named *frontotemporal mask*), and (iii) ROI analysis (see Table 2) in which each brain region associated with bvFTD neurodegeneration is considered as an independent input to the network. When ROI analysis is used, we modified our model such that the final prediction was obtained by fusing the information coming from each of the 60 regions independently processed. Formally,

$$p(y = \text{bvFTD}|X) = \sigma\left(\mathbf{w}_o\left[f^{(1)}(X^{(1)})|...|f^{(n)}(X^{(n)})\right] + \mathbf{b}_o\right)\,, \tag{9}$$

where $f^{(i)}(X^{(i)})$ is the output of the subnetwork applied to the i-th volume region, $[\cdot|\cdot]$ indicates concatenation, and $\{\mathbf{w}_o, \mathbf{b}_o\}$ are the parameters of a linear projection that fuses the subnetwork's outputs.

Region	vol. shape	Region	vol. shape
Whole Brain	$121 \times 145 \times 121$	Frontotemporal	$97 \times 96 \times 92$
Frontal Left (F_L)		Frontal Right (F_R)	
lAntOrbGy	$14 \times 21 \times 16$	rAntOrbGy	$14 \times 21 \times 16$
lCbr+Mot	$14 \times 30 \times 32$	rCbr+Mot	$15 \times 29 \times 35$
lCenOpe	$25 \times 24 \times 22$	rCenOpe	$25 \times 21 \times 22$
lFroOpe	$22 \times 18 \times 20$	rFroOpe	$21 \times 18 \times 18$
lFroPo	$24 \times 9 \times 34$	rFroPo	$25 \times 10 \times 33$
lInfFroGy	$23 \times 19 \times 23$	rInfFroGy	$23 \times 17 \times 27$
lInfFroOrbGy	$22 \times 21 \times 19$	rInfFroOrbGy	$22 \times 18 \times 18$
lMedFroCbr	$12 \times 28 \times 12$	rMedFroCbr	$10 \times 27 \times 12$
lMedOrbGy	$17 \times 39 \times 19$	rMedOrbGy	$18 \times 39 \times 20$
lMedPrcGy	$17 \times 17 \times 31$	rMedPrcGy	$16 \times 16 \times 31$
lMidFroGy	$27 \times 50 \times 56$	rMidFroGy	$26 \times 49 \times 54$
lParOpe	$29 \times 17 \times 13$	rParOpe	$24 \times 16 \times 16$
lPosOrbGy	$19 \times 20 \times 19$	rPosOrbGy	$18 \times 19 \times 19$
lPrcGy	$44 \times 34 \times 56$	rPrcGy	$44 \times 34 \times 55$
lRecGy	$10 \times 29 \times 15$	rRecGy	$10 \times 29 \times 16$
lSCA	$10 \times 11 \times 19$	rSCA	$10 \times 12 \times 19$
lSupFroGy	$21 \times 56 \times 63$	rSupFroGy	$22 \times 56 \times 61$
lSupMedFroGy	$12 \times 29 \times 47$	rSupMedFroGy	$14 \times 31 \times 50$
Subcortical Left (S_L)		Subcortical Right (S_R)	
lCau	$8 \times 28 \times 21$	rCau	$9 \times 28 \times 20$
lPut	$13 \times 23 \times 16$	rPut	$13 \times 23 \times 16$
lTha	$16 \times 23 \times 17$	rTha	$16 \times 22 \times 16$
Temporal Left (T_L)		Temporal Right (T_R)	
lAntIns	$16 \times 30 \times 28$	rAntIns	$16 \times 28 \times 27$
lFusGy	$27 \times 41 \times 34$	rFusGy	$21 \times 41 \times 35$
lInfTemGy	$33 \times 50 \times 36$	rInfTemGy	$34 \times 49 \times 38$
lPla	$22 \times 24 \times 23$	rPla	$17 \times 22 \times 22$
lPosIns	$12 \times 23 \times 28$	rPosIns	$12 \times 23 \times 29$
lSupTemGy	$24 \times 44 \times 33$	rSupTemGy	$25 \times 42 \times 32$
lTem	$24 \times 21 \times 15$	rTem	$20 \times 21 \times 20$
lTemPo	$33 \times 20 \times 35$	rTemPo	$33 \times 18 \times 36$
lTemTraGy	$20 \times 17 \times 12$	rTemTraGy	$19 \times 17 \times 13$

Table 2: Region definitions. Regions in the first line (Whole Brain and Frontotemporal) concern single-input processing, while the following lines describe regions used in per-ROI processing approaches.

6. Experiments

As preliminary step, we assessed fronto-temporal brain atrophy of bvFTD respect to HC with a VBM analysis on smoothed GM images(see Supplementary Figure). We used different network architectures, from the simplest to more complex and modern ones, and conducted several experiments based on different settings mostly related to extensive hyperparameter search for model configurations and data preprocessing, as described in the following.

6.1. Network Types

We investigated the six architectures introduced in Section 3, namely Logistic Regressors, MultiLayer Perceptrons, 3D Convolutional Networks, Visual Transformer, MLP-Mixer, and gMLP. In the following, we describe the implementation details of each tested architecture, as summarized in Figure 2.

Logistic Regression. This model is composed by a single linear projection from the voxel values to the logit (log-odds of the input belonging to the positive group).

In the ROI configuration, each region undergoes a separate linear projection with one output. The 60 outputs are concatenated and projected by an additional linear layer with sigmoid activation to obtain the final score.

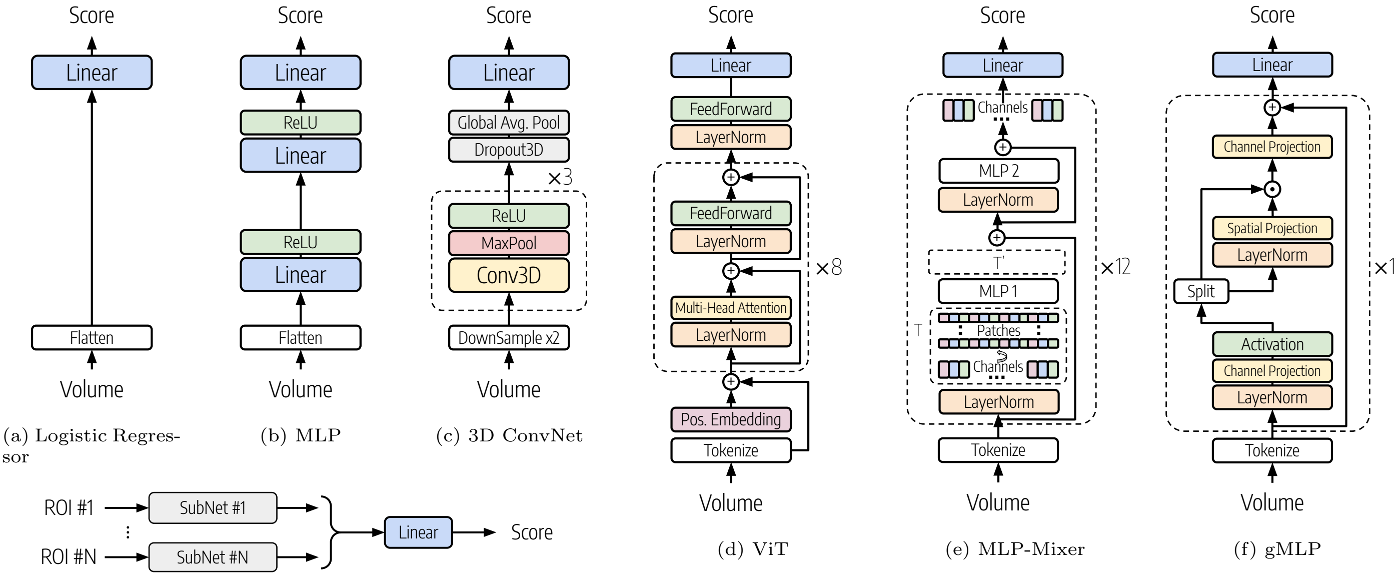
Multi-Layer Perceptron. For MLPs, we adopt two hidden layers ($L = 2$ in Eq. 2) with ReLU activations and with 100 and 50 output neurons respectively. The network processes the flattened array of voxels and produces the score using a single-output layer with sigmoid activation.

In the configuration using ROIs, we instead set the number of intermediate outputs of hidden layers to 100 and 10, thus obtaining a 600-dimensional final representation (10×60 ROIs) after concatenation. A final linear layer with sigmoid activation produces the final score from this concatenated representation.

3D Convolutional Network. We build the model with 3 convolutional layers ($L = 3$ in Eq. 5) with number of kernels 16, 64, and 256, respectively. All kernels are $3 \times 3 \times 3$ with stride of $1 \times 1 \times 1$. To lower the memory footprint, we first downsample the input volume using a $2 \times 2 \times 2$ average pooling operation. After each convolutional layer, the output is downsampled using a 3D max-pool operation that reduces its spatial extents, and then the ReLU activation is applied element-wise. The last output is mean-pooled over the three spatial dimensions, obtaining a single vector representation of the input volume with a number of dimensions equals to the number of kernels of the last convolution output. Finally, a linear layer with sigmoid activation produces $p(y = \text{bvFTD}|X)$. During training, we apply 3D spatial dropout to the output of the last convolutional layers with a probability of 0.5; this randomly set to zero the entire volume related to each kernel in the layer and helps avoiding kernel co-adaptation and overfitting.

The ROI configuration differs as follows. No initial input downsample is performed, and max pooling is applied only once after the first convolutional layer, as ROI volumes are smaller and can be processed without downsampling. We set the number of kernels in convolutional layers to 32, 64, and 10, respectively, to obtain a 600-dimensional final representation after concatenation as in the MLP model.

ViT, MLP-Mixer, gMLP. Similarly to the Vision Transformer, we exploited the same patch-based tokenization to build a general n-dimensional classifier to be applied on our three-dimensional MRI input. In our implementations of ViT, MLP-Mixer, and gMLP architectures, each module begins with a *tokenizer* part, where we used a volumetric patch of size 16^3, each linearly embedded in



(a) Logistic Regressor

(b) MLP

(c) 3D ConvNet

(d) ViT

(e) MLP-Mixer

(f) gMLP

(g) Multi-ROI meta-architecture

Figure 2: Architectures of the evaluated networks. Starting from the simple *Logistic Regressor* (a), we explored various neural models, considering the *Multi-Layer Perceptron* (b), *3D Convolution* (c), and the more recent *Visual Transformer* (d), *MLP-mixer* (e), and *gMLP* (f). In each case, the input 3D medical image is flattened or tokenized before entering the actual network. For region-based classification, each network is replicated (with an independent set of weights) after region extraction, and their output is then linearly processed for the final classification label (g).

128-dimensional token. Then follows the *encoder* part, composed of architecture-dependent encoding layers, finishing the module with a classifier made of dense layers. In each architecture we used a *Gaussian Error Linear Units* (*GELU*) activation function. In our ROI implementation, the above architectures are replicated for each region, extending the output to 8 dimensions, each then concatenated and fed through a *ReLU* non-linearity before the single-output linear layer. Note that per-ROI modules have independent weights.

The ViT encoder consisted of 8 layers, in which each multi-head attention submodule is composed of 8 heads of size 32 and a residual connection, followed by a feed-forward submodule with a 256-dimensional hidden layer and a gated non-linearity.

The MLP-Mixer architecture is composed of 12 encoding layers, with a 4× token expansion factor.

The gMLP architecture similarly has the encoding part made of of 12 encoding layers, with a 4× token expansion factor, and a final survival probability of 0.99.

Due to their analogy in explicit data intra-mixing, we call this group of networks as *transformer-based* models.

6.2. Training and Testing Sets

We train and test our networks on the FTLDNI dataset, and validate them on the CMND dataset. First, we perform hyperparameters optimization via grid search using 5-fold cross-validation on the train split of the FTLDNI dataset; we keep the hyperparameter setting that maximizes the mean balanced accuracy over the 5 test folds, and we refit the model on the entire train set with the selected hyperparameters. Then, we select the optimal threshold that maximizes the balanced accuracy on the test set of the FTLDNI dataset. Finally, we make predictions on the CMND dataset using the fitted model and the optimal threshold found. This procedure ensures a fair evaluation of the model performance, as it minimizes the risk of overfitting both model's parameters and the threshold value to the target dataset.

6.3. Support Vector Machine

For comparative purpose, we use the Pattern Recognition for Neuroimaging Toolbox (or PRoNTo) [38] to perform a binary SVM analysis to classify bvFTD respect to controls. In particular, in the training step smoothed GM images on FTLDNI dataset are treated as spatial patterns and a statistical learning model are used to identify statistical properties of the data that can be used to discriminate between the two groups of subjects [6]. We train a binary SVM to classify patients with bvFTD versus control subjects with leave-one out cross-validation and to construct voxel-wise discrimination maps. These maps of weights contain the model parameters learned by the SVM. Diagnostic prediction in the independent prediction set (CMND dataset) is performed as follows: single-subject smoothed GM densities is multiplied by the model weights computed from the linear SVM. The integral of this product define the class, which could be predicted by using a simple threshold.

7. Results

The conducted experiments were evaluated according to the most common metrics and performance measurements, revealing interesting behaviour according to network architecture complexity.

Evaluation Metrics. We evaluate our models using common metrics for binary classification evaluation, that is, the *Area under the ROC* curve (AuROC) as a threshold-independent metric, and the specificity (SS), sensitivity (SP) and balanced accuracy (Bal Acc) using the optimal threshold according to Youden's J statistic.

Networks Performances. We report metrics for each combination of data source (wm, mwp), data crop (None, FT, ROI, see Figure 1), data whitening (i.e., subtracting mean and dividing by standard deviation of the voxel distribution), and model, for a total of 72 configurations. For each metric, we report mean and standard deviation over 5 runs, training in total 360 models. Table 3 reports the mean AuROC for each configuration obtained on the FTLDNI test set and on the CMND set, whereas individual ROC curves for each configuration are depicted in Figure 3. In Table 4, we instead report metrics of the best performing configurations per data type and per model. Of note, SVM classification task reported a training AuROC over FTLDNI dataset of 95.5%, sensitivity of 90.0% and specificity of 96.4%. Diagnostic prediction in the independent CMND set of bvFTD and HC achieved an AuROC of 85.8% with sensitivity of 73.3% and specificity of 100%.

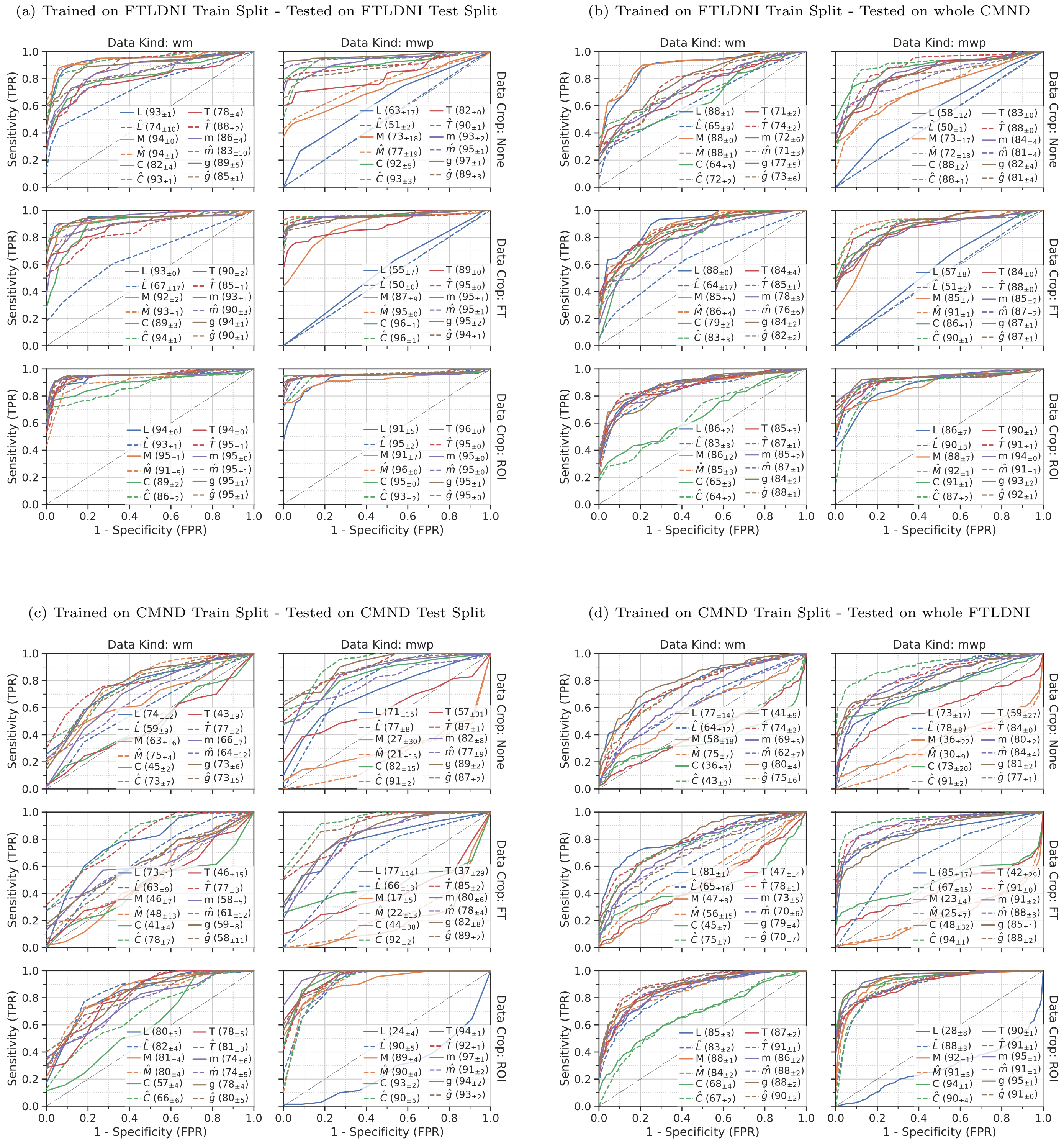
7.1. Discussion

The aim of this study was to investigate the diagnostic capability of different deep network architectures based on structural MRI in differentiating bvFTD patients from healthy controls. Our models were trained on a publicly available dataset and validated on a separated set to evaluate the *generalizability* of the achieved results. Respect to conventional machine learning investigations based on structural MRI data, deep learning methods showed higher performances in bvFTD classification [12, 6]. Moreover, our structural-based framework demonstrate a comparable predictive power respect to the most recent works that combined morphometric features with clinical outcomes or functional connectivity information [13, 14].
The results of our experimental analysis provided several insights on data, models, and the overall task.

Among the tested models, transformer-based models (ViT, MLP-Mixer, and gMLP) tend to be the most performing ones, overcoming the 90% AuROC value consistently in the best data processing configurations (per-ROI analysis of mwp data). They also appear more stable across runs and different data preprocessing, whereas simpler models may not converge or converge to sub-optimal solutions achieving higher standard deviation of metric values (see Table 3). Moreover, transformer-based models are the most promising models in terms of generalization abilities; whereas all the models can surpass the 90% values on most metrics on test data coming from the same MRI machine used for training (Tables 4a and 4c), the best performance when testing on data from different MRI machines is mostly achieved by transformer-based models (Tables 4b and 4d). Among them, ViT and gMLP models offer a better overall performance than the MLP-Mixer (one-sided paired t-test on AuROC values, p-value = 0.005 and 0.007 respectively), whereas there is no significant difference among the former. Multi-layer perceptrons also offer comparable performance while being less stable to parameter initialization, and convolutional models suffer the most from data shift. Linear logistic regressors often do not converge on less curated data.

Processing ROIs independently (Crop = ROI) tends to provide superior performance than processing the whole volume (Crop = {None, FT}) where simpler models (Logistic Reg., MLP) get confused more easily. However, the performance boost is payed with an increased model size and computational cost (see Table 6). Per-ROI processing also adds stability to model training as shown in Figure 4. We observed a strong correlation between predictions across different models and across multiple training runs when using per-ROI processing. This occurs also when using unmodulated (wm) data. On the other hand, when processing whole volumes, correlation between predictions of different models tends to decrease, sometimes even between different training runs of the same model. We deem that the per-ROI processing pipeline, adopting multiple submodels per ROI, balances the local convergence of each submodule and reduces the risk of global overfitting. Table 5 shows how the AuROC changes for the best model (gMLP on modulated and whitened data) when using different configurations of ROIs inside macro-regions. Note that ROIs in the frontal area lead to most performant classifiers. When processing whole volumes, frontotemporal masking (FT) should be adopted. As expected, modulated data (Kind = mwp) tend to increase performances in all models but the simpler ones (Logistic Reg., MLP) where we deem cleaner data, together with its scarcity, increase chances of overfitting. Obtaining high performances (AuROC > 85%) can be achieved also with unmodulated data (Kind = wm) but is more dependent on the specific data processing (in particular the data crop) used. Concerning data whitening, no strong pattern emerges, as performance only slightly improves or degrades depending on the specific configuration considered.

We deem the attention-based processing in transformer-like architectures, i.e., building finer representations by comparing different 3D parts of the input, can facilitate the learning of more scanner-independent features and improve the generalization of classifiers, especially in multiple source settings. Our experiments support this trend, but further validation with additional data sources should be carried out in future work.



(a) Trained on FTLDNI Train Split - Tested on FTLDNI Test Split

(b) Trained on FTLDNI Train Split - Tested on whole CMND

(c) Trained on CMND Train Split - Tested on CMND Test Split

(d) Trained on CMND Train Split - Tested on whole FTLDNI

Figure 3: ROC curves and AUC (%, mean$_{\pm\text{std}}$ in parenthesis in the legend) for different data preprocessing. Different data modulations are shown in columns, and different data cropping are shown in rows. Dashed lines and the hat symbol $\hat{\cdot}$ in the legend indicate results with data whitening. L = Linear Regressor; M = MLP; C = ConvNet3D; T = ViT; m = MLP-Mixer; g = gMLP.

Data Kind	wm						mwp					
Data Crop	None		FT		ROI		None		FT		ROI	
Whitening	✗	✓	✗	✓	✗	✓	✗	✓	✗	✓	✗	✓
Trained on FTLDNI Train Split - Tested on FTLDNI Test Split												
Logistic Regressor	$93_{\pm1}$	$74_{\pm11}$	$93_{\pm0}$	$67_{\pm19}$	$94_{\pm0}$	$93_{\pm1}$	$63_{\pm19}$	$51_{\pm2}$	$55_{\pm8}$	$50_{\pm0}$	$91_{\pm5}$	$95_{\pm2}$
MLP	$94_{\pm1}$	$94_{\pm1}$	$92_{\pm2}$	$93_{\pm1}$	$95_{\pm1}$	$91_{\pm6}$	$73_{\pm20}$	$77_{\pm21}$	$87_{\pm10}$	$95_{\pm0}$	$91_{\pm8}$	$96_{\pm0}$
ConvNet3D	$82_{\pm5}$	$93_{\pm1}$	$89_{\pm3}$	$94_{\pm1}$	$89_{\pm3}$	$86_{\pm2}$	$92_{\pm6}$	$93_{\pm4}$	$96_{\pm1}$	$96_{\pm1}$	$95_{\pm0}$	$93_{\pm3}$
ViT	$78_{\pm4}$	$88_{\pm2}$	$90_{\pm2}$	$85_{\pm1}$	$94_{\pm0}$	$95_{\pm1}$	$82_{\pm0}$	$90_{\pm1}$	$89_{\pm0}$	$95_{\pm0}$	$96_{\pm0}$	$95_{\pm0}$
MLP-Mixer	$86_{\pm4}$	$83_{\pm11}$	$93_{\pm1}$	$90_{\pm3}$	$95_{\pm0}$	$95_{\pm1}$	$93_{\pm3}$	$95_{\pm1}$	$95_{\pm1}$	$95_{\pm1}$	$95_{\pm0}$	$95_{\pm0}$
gMLP	$89_{\pm6}$	$85_{\pm2}$	$94_{\pm2}$	$90_{\pm1}$	$95_{\pm1}$	$95_{\pm1}$	$97_{\pm1}$	$89_{\pm4}$	$95_{\pm2}$	$94_{\pm1}$	$95_{\pm1}$	$95_{\pm0}$
Trained on FTLDNI Train Split - Tested on whole CMND												
Logistic Regressor	$88_{\pm1}$	$65_{\pm10}$	$88_{\pm0}$	$64_{\pm19}$	$86_{\pm2}$	$83_{\pm3}$	$58_{\pm14}$	$50_{\pm1}$	$57_{\pm8}$	$51_{\pm2}$	$86_{\pm7}$	$90_{\pm4}$
MLP	$88_{\pm0}$	$88_{\pm1}$	$85_{\pm5}$	$86_{\pm4}$	$86_{\pm2}$	$85_{\pm3}$	$73_{\pm19}$	$72_{\pm15}$	$85_{\pm8}$	$91_{\pm1}$	$88_{\pm7}$	$92_{\pm1}$
ConvNet3D	$64_{\pm3}$	$72_{\pm2}$	$79_{\pm2}$	$83_{\pm3}$	$65_{\pm3}$	$64_{\pm2}$	$88_{\pm3}$	$88_{\pm1}$	$86_{\pm2}$	$90_{\pm1}$	$91_{\pm1}$	$87_{\pm2}$
ViT	$71_{\pm2}$	$74_{\pm2}$	$84_{\pm4}$	$85_{\pm2}$	$85_{\pm3}$	$87_{\pm2}$	$83_{\pm0}$	$88_{\pm0}$	$84_{\pm0}$	$88_{\pm0}$	$90_{\pm2}$	$91_{\pm1}$
MLP-Mixer	$72_{\pm6}$	$71_{\pm3}$	$78_{\pm3}$	$76_{\pm7}$	$85_{\pm3}$	$87_{\pm1}$	$84_{\pm5}$	$81_{\pm4}$	$85_{\pm2}$	$87_{\pm2}$	$94_{\pm0}$	$91_{\pm1}$
gMLP	$77_{\pm5}$	$73_{\pm7}$	$84_{\pm3}$	$82_{\pm2}$	$84_{\pm2}$	$88_{\pm1}$	$82_{\pm4}$	$81_{\pm5}$	$87_{\pm1}$	$87_{\pm1}$	$93_{\pm2}$	$92_{\pm1}$
Trained on CMND Train Split - Tested on CMND Test Split												
Logistic Regressor	$74_{\pm13}$	$59_{\pm10}$	$73_{\pm1}$	$63_{\pm10}$	$80_{\pm3}$	$82_{\pm5}$	$71_{\pm17}$	$77_{\pm9}$	$77_{\pm16}$	$66_{\pm15}$	$24_{\pm4}$	$90_{\pm6}$
MLP	$63_{\pm18}$	$75_{\pm4}$	$46_{\pm8}$	$48_{\pm15}$	$81_{\pm4}$	$80_{\pm5}$	$27_{\pm33}$	$21_{\pm16}$	$17_{\pm6}$	$22_{\pm14}$	$89_{\pm4}$	$90_{\pm5}$
ConvNet 3D	$45_{\pm2}$	$73_{\pm8}$	$41_{\pm5}$	$78_{\pm8}$	$57_{\pm4}$	$66_{\pm6}$	$82_{\pm17}$	$91_{\pm2}$	$44_{\pm42}$	$92_{\pm3}$	$93_{\pm2}$	$90_{\pm6}$
ViT	$43_{\pm10}$	$77_{\pm2}$	$46_{\pm17}$	$77_{\pm4}$	$78_{\pm6}$	$81_{\pm3}$	$57_{\pm34}$	$87_{\pm1}$	$37_{\pm32}$	$85_{\pm2}$	$94_{\pm1}$	$92_{\pm1}$
MLP-Mixer	$66_{\pm8}$	$64_{\pm13}$	$58_{\pm6}$	$61_{\pm14}$	$74_{\pm7}$	$74_{\pm6}$	$82_{\pm9}$	$77_{\pm10}$	$80_{\pm7}$	$78_{\pm5}$	$97_{\pm1}$	$91_{\pm2}$
gMLP	$73_{\pm6}$	$73_{\pm6}$	$59_{\pm9}$	$58_{\pm12}$	$78_{\pm4}$	$80_{\pm6}$	$89_{\pm2}$	$87_{\pm2}$	$82_{\pm9}$	$89_{\pm2}$	$94_{\pm2}$	$93_{\pm2}$
Trained on CMND Train Split - Tested on whole FTLDNI												
Logistic Regressor	$77_{\pm15}$	$64_{\pm13}$	$81_{\pm1}$	$65_{\pm18}$	$85_{\pm3}$	$83_{\pm2}$	$73_{\pm19}$	$78_{\pm9}$	$85_{\pm19}$	$67_{\pm17}$	$28_{\pm9}$	$88_{\pm4}$
MLP	$58_{\pm21}$	$75_{\pm8}$	$47_{\pm9}$	$56_{\pm17}$	$88_{\pm1}$	$84_{\pm2}$	$36_{\pm24}$	$30_{\pm10}$	$23_{\pm5}$	$25_{\pm8}$	$92_{\pm1}$	$91_{\pm5}$
ConvNet 3D	$36_{\pm3}$	$43_{\pm4}$	$45_{\pm8}$	$75_{\pm8}$	$68_{\pm4}$	$67_{\pm2}$	$73_{\pm22}$	$91_{\pm2}$	$48_{\pm36}$	$94_{\pm1}$	$94_{\pm1}$	$90_{\pm4}$
ViT	$41_{\pm10}$	$74_{\pm2}$	$47_{\pm15}$	$78_{\pm1}$	$87_{\pm2}$	$91_{\pm1}$	$59_{\pm30}$	$84_{\pm1}$	$42_{\pm32}$	$91_{\pm0}$	$90_{\pm1}$	$91_{\pm1}$
MLP-Mixer	$69_{\pm5}$	$62_{\pm8}$	$73_{\pm6}$	$70_{\pm7}$	$86_{\pm2}$	$88_{\pm2}$	$80_{\pm2}$	$84_{\pm5}$	$91_{\pm2}$	$88_{\pm3}$	$95_{\pm1}$	$91_{\pm1}$
gMLP	$80_{\pm4}$	$75_{\pm6}$	$79_{\pm5}$	$70_{\pm8}$	$88_{\pm2}$	$90_{\pm2}$	$81_{\pm2}$	$77_{\pm1}$	$85_{\pm1}$	$88_{\pm2}$	$95_{\pm1}$	$91_{\pm1}$

Table 3: Area under the ROC curve (%, $\text{mean}_{\pm\text{std}}$) on FTLDNI and CMND datasets. None = Whole Volume; FT = Frontotemporal Masking; ROI = Per-ROI Processing (see Figure 1).

Table 4: Metrics (%, mean ± std) of best performing configurations (in terms of balanced accuracy) for each model and data kind. **Boldface values** indicates the highest values obtained among models for a specific metric.

(a) Trained on FTLDNI Train Split - Tested on FTLDNI Test Split

Data	Model	Crop	Whiten	Sensitivity	Specificity	Bal. Accuracy	AuROC
wm	Logistic Regression	FT	✗	90.00 ± 0.00	95.60 ± 0.89	92.80 ± 0.45	93.48 ± 0.26
	MLP	FT	✓	90.00 ± 0.00	94.40 ± 1.67	92.20 ± 0.84	93.46 ± 0.62
	ConvNet 3D	FT	✓	86.00 ± 2.24	95.60 ± 4.34	90.80 ± 1.10	93.70 ± 0.63
	Transformer	ROI	✓	93.00 ± 2.74	92.00 ± 3.74	92.50 ± 1.27	94.64 ± 0.74
	MLP-Mixer	ROI	✗	91.00 ± 4.18	**96.80** ± 1.10	93.90 ± 1.71	94.54 ± 0.38
	gMLP	ROI	✓	**94.00** ± 2.24	94.80 ± 1.79	**94.40** ± 1.14	**95.36** ± 0.61
mwp	Logistic Regression	ROI	✓	94.00 ± 2.24	96.80 ± 4.15	95.40 ± 1.88	94.55 ± 2.05
	MLP	ROI	✓	**95.00** ± 0.00	**100.00** ± 0.00	**97.50** ± 0.00	95.68 ± 0.11
	ConvNet 3D	ROI	✗	**95.00** ± 0.00	99.60 ± 0.89	97.30 ± 0.45	95.40 ± 0.39
	Transformer	ROI	✗	92.00 ± 2.74	98.80 ± 1.10	95.40 ± 1.56	95.56 ± 0.30
	MLP-Mixer	ROI	✗	92.00 ± 2.74	98.80 ± 1.79	95.40 ± 0.65	95.22 ± 0.16
	gMLP	None	✗	93.00 ± 2.74	99.20 ± 1.10	96.10 ± 1.56	**96.56** ± 1.13

(b) Trained on FTLDNI Train Split - Tested on whole CMND

Data	Model	Crop	Whiten	Sensitivity	Specificity	Bal. Accuracy	AuROC
wm	Logistic Regressor	FT	✗	**91.33** ± 1.83	74.17 ± 1.86	82.75 ± 0.81	**88.25** ± 0.32
	MLP	None	✗	88.00 ± 2.98	81.67 ± 3.73	84.83 ± 0.37	88.03 ± 0.42
	ConvNet 3D	FT	✓	78.00 ± 10.95	80.00 ± 6.85	79.00 ± 3.64	83.43 ± 3.17
	Transformer	ROI	✓	80.00 ± 4.08	90.83 ± 3.49	**85.42** ± 2.34	86.56 ± 1.66
	MLP-Mixer	ROI	✓	76.00 ± 5.96	**91.67** ± 5.10	83.83 ± 2.23	87.06 ± 1.43
	gMLP	ROI	✓	81.33 ± 6.91	87.50 ± 9.77	84.42 ± 2.14	87.75 ± 0.86
mwp	Logistic Regressor	ROI	✓	82.67 ± 6.41	91.67 ± 7.22	87.17 ± 2.44	89.75 ± 3.69
	MLP	FT	✓	88.67 ± 2.98	90.00 ± 3.73	89.33 ± 0.91	91.36 ± 0.94
	ConvNet 3D	ROI	✗	88.67 ± 3.80	90.00 ± 3.73	89.33 ± 0.70	91.19 ± 0.59
	Transformer	ROI	✓	83.33 ± 8.16	93.33 ± 7.57	88.33 ± 2.59	91.19 ± 0.85
	MLP-Mixer	ROI	✗	83.33 ± 6.67	**95.83** ± 5.10	89.58 ± 1.93	**93.56** ± 0.32
	gMLP	ROI	✗	**92.00** ± 1.83	90.00 ± 3.73	**91.00** ± 2.53	92.56 ± 2.15

(c) Trained on CMND Train Split - Tested on CMND Test Split

Data	Model	Crop	Whiten	Sensitivity	Specificity	Bal. Accuracy	AuROC
wm	Logistic Regressor	ROI	✓	85.71 ± 8.75	80.00 ± 13.48	**82.86** ± 4.90	**81.95** ± 5.00
	MLP	ROI	✓	75.71 ± 6.39	**83.64** ± 11.85	79.68 ± 2.83	79.87 ± 4.90
	ConvNet 3D	FT	✓	**91.43** ± 5.98	69.09 ± 13.79	80.26 ± 5.37	77.53 ± 8.09
	ViT	ROI	✓	82.86 ± 10.83	72.73 ± 11.13	77.79 ± 3.41	**81.04** ± 3.16
	MLP-Mixer	ROI	✓	67.14 ± 18.63	80.00 ± 14.94	73.57 ± 3.99	74.03 ± 5.70
	gMLP	None	✗	81.43 ± 8.14	72.73 ± 11.13	77.08 ± 2.64	73.25 ± 6.48
mwp	Logistic Regressor	ROI	✓	92.86 ± 8.75	81.82 ± 12.86	87.34 ± 4.46	89.81 ± 5.96
	MLP	ROI	✓	95.71 ± 3.91	83.64 ± 11.85	89.68 ± 4.41	90.00 ± 4.61
	ConvNet 3D	ROI	✗	97.14 ± 3.91	81.82 ± 9.09	89.48 ± 4.05	92.99 ± 1.86
	ViT	ROI	✗	90.00 ± 9.58	87.27 ± 10.37	88.64 ± 1.66	93.64 ± 1.48
	MLP-Mixer	ROI	✗	94.29 ± 9.31	**90.91** ± 6.43	**92.60** ± 2.77	**96.75** ± 0.80
	gMLP	ROI	✗	**98.57** ± 3.19	85.45 ± 4.98	92.01 ± 1.97	94.42 ± 2.08

(d) Trained on CMND Train Split - Tested on whole FTLDNI

Data	Model	Crop	Whiten	Sensitivity	Specificity	Bal. Accuracy	AuROC
wm	Logistic Regressor	ROI	✗	79.20 ± 5.02	80.00 ± 7.79	79.60 ± 3.18	85.49 ± 2.97
	MLP	ROI	✗	75.20 ± 4.60	**87.82** ± 3.67	81.51 ± 1.29	87.64 ± 1.21
	ConvNet 3D	FT	✓	66.00 ± 13.64	80.55 ± 9.01	73.27 ± 5.18	75.38 ± 8.29
	ViT	ROI	✓	**86.80** ± 5.22	84.00 ± 5.36	**85.40** ± 2.17	**90.72** ± 0.78
	MLP-Mixer	ROI	✓	80.40 ± 9.84	83.45 ± 7.88	81.93 ± 2.74	87.65 ± 1.71
	gMLP	ROI	✓	82.80 ± 3.63	86.55 ± 4.61	84.67 ± 1.94	**90.22** ± 1.68
mwp	Logistic Regressor	ROI	✓	82.00 ± 4.24	85.82 ± 6.08	83.91 ± 4.78	88.14 ± 3.58
	MLP	ROI	✗	**90.00** ± 3.74	84.36 ± 5.91	87.18 ± 1.64	91.67 ± 0.98
	ConvNet 3D	FT	✓	88.80 ± 3.03	91.64 ± 4.33	90.22 ± 1.28	94.39 ± 0.97
	ViT	FT	✓	82.80 ± 4.38	90.36 ± 4.10	86.58 ± 0.64	91.49 ± 0.34
	MLP-Mixer	ROI	✗	**89.60** ± 4.77	89.64 ± 4.15	**89.62** ± 0.90	94.68 ± 0.81
	gMLP	ROI	✗	86.80 ± 3.63	**92.55** ± 5.43	**89.67** ± 1.20	**95.01** ± 0.78

Table 5: Classification performance of different input Macro-ROI combinations. AuROC (mean$_{\pm\text{std}}$) obtained by gMLP on modulated and whitened data trained on FTLDNI and tested on CMND. Macro-ROIs aggregate the left and right sides of the frontal (F), subcortical (S), and temporal (T) brain regions.

	F_L	F_R	S_L	S_R	T_L	T_R	AuROC
1 Region	✓						$92_{\pm1}$
		✓					$89_{\pm1}$
			✓				$85_{\pm1}$
				✓			$81_{\pm3}$
					✓		$88_{\pm2}$
						✓	$84_{\pm1}$
2 Regions	✓	✓					$92_{\pm0}$
	✓		✓				$91_{\pm0}$
	✓			✓			$92_{\pm1}$
	✓				✓		$91_{\pm1}$
	✓					✓	$91_{\pm1}$
		✓	✓				$90_{\pm1}$
		✓		✓			$90_{\pm0}$
		✓			✓		$90_{\pm0}$
		✓				✓	$89_{\pm1}$
			✓	✓			$83_{\pm2}$
			✓		✓		$88_{\pm1}$
			✓			✓	$86_{\pm1}$
				✓	✓		$87_{\pm1}$
				✓		✓	$84_{\pm1}$
					✓	✓	$87_{\pm1}$
3 Regions	✓	✓	✓				$92_{\pm0}$
	✓	✓		✓			$92_{\pm1}$
	✓	✓			✓		$92_{\pm1}$
	✓	✓				✓	$91_{\pm1}$
	✓		✓	✓			$91_{\pm0}$
	✓		✓		✓		$91_{\pm1}$
	✓		✓			✓	$91_{\pm1}$
	✓			✓	✓		$91_{\pm1}$
	✓			✓		✓	$90_{\pm1}$
	✓				✓	✓	$91_{\pm0}$
		✓	✓	✓			$90_{\pm1}$
		✓	✓		✓		$91_{\pm1}$
		✓	✓			✓	$89_{\pm1}$
		✓		✓	✓		$91_{\pm1}$
		✓		✓		✓	$90_{\pm1}$
		✓			✓	✓	$90_{\pm1}$
			✓	✓	✓		$87_{\pm2}$
			✓	✓		✓	$86_{\pm1}$
			✓		✓	✓	$88_{\pm1}$
				✓	✓	✓	$88_{\pm0}$
4 Regions	✓	✓	✓	✓			$92_{\pm0}$
	✓	✓	✓		✓		$92_{\pm1}$
	✓	✓	✓			✓	$92_{\pm0}$
	✓	✓		✓	✓		$92_{\pm1}$
	✓	✓		✓		✓	$92_{\pm1}$
	✓	✓			✓	✓	$92_{\pm0}$
	✓		✓	✓	✓		$90_{\pm0}$
	✓		✓	✓		✓	$90_{\pm1}$
	✓		✓		✓	✓	$92_{\pm0}$
	✓			✓	✓	✓	$90_{\pm0}$
		✓	✓	✓	✓		$90_{\pm1}$
		✓	✓	✓		✓	$90_{\pm1}$
		✓	✓		✓	✓	$90_{\pm1}$
		✓		✓	✓	✓	$90_{\pm1}$
			✓	✓	✓	✓	$87_{\pm1}$
5 Regions	✓	✓	✓	✓	✓		$92_{\pm1}$
	✓	✓	✓	✓		✓	$92_{\pm1}$
	✓	✓	✓		✓	✓	$92_{\pm0}$
	✓	✓		✓	✓	✓	$92_{\pm1}$
	✓		✓	✓	✓	✓	$91_{\pm1}$
		✓	✓	✓	✓	✓	$90_{\pm1}$
All	✓	✓	✓	✓	✓	✓	$91_{\pm1}$

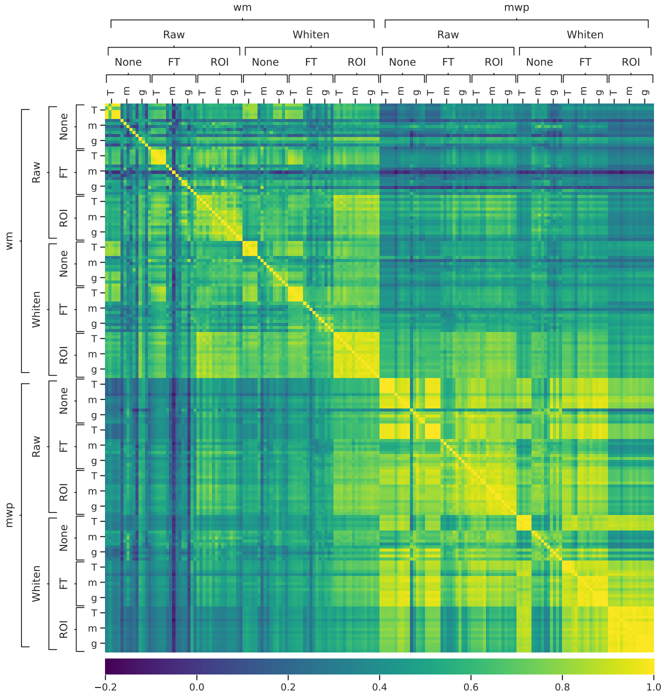


Figure 4: CMND: Pearson Correlation Coefficients among predictions of various transformer-based models. T = ViT, m = MLP-Mixer, g = gMLP. For each configuration, we report 5 runs with randomly initialized weights. Note that when ROI processing is used, predictions tend to highly correlate independently from the model or random weight initialization used.

Table 6: Computational complexity of compared models in terms of floating point operations (FLOPs) and number of parameters.

Data Crop	None		FT		ROI	
	FLOPs	Params	FLOPs	Params	FLOPs	Params
Logistic Reg.	4.2M	2.1M	1.7M	856.7k	2.3M	1.1M
MLP	424.6M	212.3M	171.4M	85.7M	227.8M	113.9M
ConvNet 3D	45.3k	471.0k	16.9k	471.0k	678.0k	3.4M
ViT	973.9M	468.6k	282.0M	443.8k	3.2G	142.0M
MLP-Mixer	2.7G	41.5M	1.1G	8.2M	2.6G	128.0M
gMLP	8.2G	9.4M	2.6G	5.2M	5.5G	268.8M

8. Conclusion

In our work, we investigated several neural network architectures with the purpose of creating a *robust* and *generalizable* model that was able to identify patients affected by behavioral variant frontotemporal dementia (bvFTD) from medical imaging data obtained by different acquisition devices. We considered the Frontotemporal Lobar Degeneration Neuroimaging Initiative (FTLDNI) as primary dataset on which we performed training and testing, and the Center for Neurodegenerative Diseases and the Aging Brain (CMND) dataset as source to validate the generalizability of the model without performing any fine tuning. We considered different architectures, from the simple logistic regressor to threedimensional convolution networks and the latest vision transformers with its similar variants. In general, we found that all architectures perform well in the bvFTD identification task with both dataset, but the transformer-based ones are the most stable in terms of weight initialization conditions, consistently reaching and exceeding the 91.0% for AuROC and balanced accuracy values. These results let us validate that overall data intra-mixing (i.e., as it emerges from the attention mechanism and its variants) is a principal component in imaging classification.

We plan to further dig into the most recent attention-based architectures, trying to define a model able to intra-mix data in linear-time complexity using learned intermediate representations or frequency analysis [39], as well as extending the systems robustness by testing them on future-available imaging datasets.

Acknowledgement

This work was partially funded by: AI4Media - A European Excellence Centre for Media, Society and Democracy (EC, H2020 n. 951911); AI-MAP - Tuscany POR FSE 2014-2020 AI-MAP (CNR4C program, CUP B15J19001040004); Extension - European Space Agency (Grant No. 4000132621/20/NL/AF); Regione Puglia (DGR n. 1284 - 27.05.2015) for the project "Registro Regionale per la SLA, la FTD-P e le patologie affini in Regione Puglia" (SLAPDem register- www.slapdem.it); Italian Ministry of Health (Ricerca Corrente), Regione Puglia and CNR for Tecnopolo per la Medicina di Precisione (D.G.R. n. 2117 of 21.11.2018).

References

[1] G. Logroscino, M. Piccininni, Amyotrophic lateral sclerosis descriptive epidemiology: the origin of geographic difference, Neuroepidemiology 52 (1-2) (2019) 93–103.

[2] E. Ratnavalli, C. Brayne, K. Dawson, J. Hodges, The prevalence of frontotemporal dementia, Neurology 58 (11) (2002) 1615–1621.

[3] C. U. Onyike, J. Diehl-Schmid, The epidemiology of frontotemporal dementia, International Review of Psychiatry 25 (2) (2013) 130–137.

[4] K. Rascovsky, J. R. Hodges, D. Knopman, M. F. Mendez, J. H. Kramer, J. Neuhaus, J. C. Van Swieten, H. Seelaar, E. G. Dopper, C. U. Onyike, et al., Sensitivity of revised diagnostic criteria for the behavioural variant of frontotemporal dementia, Brain 134 (9) (2011) 2456–2477.

[5] J. McCarthy, D. L. Collins, S. Ducharme, Morphometric mri as a diagnostic biomarker of frontotemporal dementia: A systematic review to determine clinical applicability, NeuroImage: Clinical 20 (2018) 685–696.

[6] C. Möller, Y. A. Pijnenburg, W. M. van der Flier, A. Versteeg, B. Tijms, J. C. de Munck, A. Hafkemeijer, S. A. Rombouts, J. van der Grond, J. van Swieten, et al., Alzheimer disease and behavioral variant frontotemporal dementia: automatic classification based on cortical atrophy for single-subject diagnosis, Radiology 279 (3) (2016) 838–848.

[7] S. Spasov, L. Passamonti, A. Duggento, P. Liò, N. Toschi, A. D. N. Initiative, et al., A parameter-efficient deep learning approach to predict conversion from mild cognitive impairment to alzheimer's disease, Neuroimage 189 (2019) 276–287.

[8] S. Basaia, F. Agosta, L. Wagner, E. Canu, G. Magnani, R. Santangelo, M. Filippi, A. D. N. Initiative, et al., Automated classification of alzheimer's disease and mild cognitive impairment using a single mri and deep neural networks, NeuroImage: Clinical 21 (2019) 101645.

[9] J. Hu, Z. Qing, R. Liu, X. Zhang, P. Lv, M. Wang, Y. Wang, K. He, Y. Gao, B. Zhang, Deep learning-based classification and voxel-based visualization of frontotemporal dementia and alzheimer's disease, Frontiers in Neuroscience 14 (2021) 1468.

[10] P. R. Raamana, H. Rosen, B. Miller, M. W. Weiner, L. Wang, M. F. Beg, Three-class differential diagnosis among alzheimer disease, frontotemporal dementia, and controls, Frontiers in neurology 5 (2014) 71.

[11] T. W. Chow, M. A. Binns, M. Freedman, D. T. Stuss, J. Ramirez, C. J. Scott, S. Black, Overlap in frontotemporal atrophy between normal aging and patients with frontotemporal dementias, Alzheimer Disease & Associated Disorders 22 (4) (2008) 327–335.

[12] S. Meyer, K. Mueller, K. Stuke, S. Bisenius, J. Diehl-Schmid, F. Jessen, J. Kassubek, J. Kornhuber, A. C. Ludolph, J. Prudlo, et al., Predicting behavioral variant frontotemporal dementia with pattern classification in multi-center structural mri data, NeuroImage: Clinical 14 (2017) 656–662.

[13] M. B. Bachli, L. Sedeño, J. K. Ochab, O. Piguet, F. Kumfor, P. Reyes, T. Torralva, M. Roca, J. F. Cardona, C. G. Campo, et al., Evaluating the reliability of neurocognitive biomarkers of neurodegenerative diseases across countries: A machine learning approach, NeuroImage 208 (2020) 116456.

[14] P. A. Donnelly-Kehoe, G. O. Pascariello, A. M. García, J. R. Hodges, B. Miller, H. Rosen, F. Manes, R. Landin-Romero, D. Matallana, C. Serrano, et al., Robust automated computational approach for classifying frontotemporal neurodegeneration: Multimodal/multicenter neuroimaging, Alzheimer's & Dementia: Diagnosis, Assessment & Disease Monitoring 11 (1) (2019) 588–598.

[15] K. Nemoto, H. Sakaguchi, W. Kasai, M. Hotta, R. Kamei, T. Noguchi, R. Minamimoto, T. Arai, T. Asada, Differentiating dementia with lewy bodies and alzheimer's disease by deep learning to structural mri, Journal of Neuroimaging 31 (3) (2021) 579–587.

[16] S. Qiu, P. S. Joshi, M. I. Miller, C. Xue, X. Zhou, C. Karjadi, G. H. Chang, A. S. Joshi, B. Dwyer, S. Zhu, et al., Development and validation of an interpretable deep learning framework for alzheimer's disease classification, Brain 143 (6) (2020) 1920–1933.

[17] A. B. Tufail, Y.-K. Ma, Q.-N. Zhang, Binary classification of alzheimer's disease using smri imaging modality and deep learning, Journal of digital imaging 33 (5) (2020) 1073–1090.

[18] D. Ma, D. Lu, K. Popuri, L. Wang, M. F. Beg, A. D. N. Initiative, et al., Differential diagnosis of frontotemporal dementia, alzheimer's disease, and normal aging using a multi-scale multi-type feature generative adversarial deep neural network on structural magnetic resonance images, Frontiers in Neuroscience 14 (2020) 853.

[19] C. Lian, M. Liu, J. Zhang, D. Shen, Hierarchical fully convolutional network for joint atrophy localization and alzheimer's disease diagnosis using structural mri, IEEE transactions on pattern analysis and machine intelligence 42 (4) (2018) 880–893.

[20] W. Gong, C. F. Beckmann, A. Vedaldi, S. M. Smith, H. Peng, Optimising a simple fully convolutional network for accurate brain age prediction in the pac 2019 challenge, Frontiers in Psychiatry 12 (2021).

[21] V. Vapnik, The nature of statistical learning theory, Springer science & business media, 2013.

[22] C. Cortes, V. Vapnik, Support-vector networks, Machine learning 20 (3) (1995) 273–297.

[23] L. Breiman, Random forests, Machine learning 45 (1) (2001) 5–32.

[24] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, Advances in

neural information processing systems 25 (2012) 1097–1105.

[25] F. Rosenblatt, Principles of neurodynamics: Perceptrons and the theory of brain mechanisms (1961).

[26] F. Rosenblatt, The perceptron: a probabilistic model for information storage and organization in the brain., Psychological review 65 (6) (1958) 386.

[27] J. K. Udupa, G. T. Herman, 3D imaging in medicine, CRC press, 1999.

[28] J. L. Elman, Finding structure in time, Cognitive science 14 (2) (1990) 179–211.

[29] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural computation 9 (8) (1997) 1735–1780.

[30] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in neural information processing systems, 2017, pp. 5998–6008.

[31] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An image is worth 16x16 words: Transformers for image recognition at scale (2021). arXiv:2010.11929.

[32] B. Wu, C. Xu, X. Dai, A. Wan, P. Zhang, Z. Yan, M. Tomizuka, J. Gonzalez, K. Keutzer, P. Vajda, Visual transformers: Token-based image representation and processing for computer vision (2020). arXiv:2006.03677.

[33] I. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit, M. Lucic, A. Dosovitskiy, Mlp-mixer: An all-mlp architecture for vision (2021). arXiv:2105.01601.

[34] H. Liu, Z. Dai, D. R. So, Q. V. Le, Pay attention to mlps, arXiv preprint arXiv:2105.08050 (2021).

[35] Memory and Aging Center — University of California, San Francisco (2021).
URL https://memory.ucsf.edu/research-trials

[36] W.-L. Luo, T. E. Nichols, Diagnosis and exploration of massively univariate neuroimaging models, NeuroImage 19 (3) (2003) 1014–1032.

[37] Neuromorphometrics, Inc. — Building a Model of the Living Human Brain (2021).
URL http://www.neuromorphometrics.com

[38] J. Schrouff, M. J. Rosa, J. M. Rondina, A. F. Marquand, C. Chu, J. Ashburner, C. Phillips, J. Richiardi, J. Mourao-Miranda, Pronto: pattern recognition for neuroimaging toolbox, Neuroinformatics 11 (3) (2013) 319–337.

[39] J. Lee-Thorp, J. Ainslie, I. Eckstein, S. Ontanon, Fnet: Mixing tokens with fourier transforms (2021). arXiv:2105.03824.