National Technical University of Athens
School of Electrical and Computer Engineering
Division of Industrial Electric Devices and Decision Systems

Alzheimer's Disease Diagnosis Using a Multimodal Approach with 3D MRI and PET

Diploma Thesis

of

ANTHI-MARIA VOZINAKI

Supervisor: Dimitrios Askounis
Professor NTUA

Athens, February 2025

National Technical University of Athens
School of Electrical and Computer Engineering
Division of Industrial Electric Devices and Decision Systems

Alzheimer's Disease Diagnosis Using a Multimodal Approach with 3D MRI and PET

Diploma Thesis

of

ANTHI-MARIA VOZINAKI

Supervisor: Dimitrios Askounis
Professor NTUA

Approved by the examination committee on 24th February 2025.

(Signature)	*(Signature)*	*(Signature)*
.....	.....	.....
Dimitrios Askounis	John Psarras	Vangelis Marinakis
Professor NTUA	Professor NTUA	Assistant Professor NTUA

Athens, February 2025

National Technical University of Athens
School of Electrical and Computer Engineering
Division of Industrial Electric Devices and Decision Systems

DISCLAIMER ON ACADEMIC ETHICS AND INTELLECTUAL PROPERTY RIGHTS

Being fully aware of the implications of copyright laws, I expressly state that this diploma thesis, as well as the electronic files and source codes developed or modified in the course of this thesis, are solely the product of my personal work and do not infringe any rights of intellectual property, personality and personal data of third parties, do not contain work / contributions of third parties for which the permission of the authors / beneficiaries is required and are not a product of partial or complete plagiarism, while the sources used are limited to the bibliographic references only and meet the rules of scientific citing. The points where I have used ideas, text, files and / or sources of other authors are clearly mentioned in the text with the appropriate citation and the relevant complete reference is included in the bibliographic references section. I fully, individually and personally undertake all legal and administrative consequences that may arise in the event that it is proven, in the course of time, that this thesis or part of it does not belong to me because it is a product of plagiarism.

(Signature)

.....

Anthi-Maria Vozinaki

Certified Electrical and Computer Engineer

24th February 2025

Abstract

Alzheimer's disease is an irreversible brain disease that severely damages human thinking and is the seventh leading cause of death worldwide. Early diagnosis plays an important part especially at the Mild Cognitive Impairment stage, where timely intervention can help slow its progression before it advances to AD. Neuroimaging data, like MRI and PET scans, can help detect brain changes early by providing structural and functional brain changes related to the disease. However, despite the availability of various imaging modalities for the same patient, the development of multi-modal models leveraging these modalities remains underexplored.

This thesis aims to address this gap by proposing and evaluating classification models using 3D MRI and amyloid PET scans in a multimodal framework. We first employ a 3D Convolutional Neural Network, followed by three fusion techniques: feature concatenation, Gated Multimodal Unit, and Gated Self-Attention. To further improve classification performance and computational efficiency, we integrate a Mixture of Experts model, which dynamically selects the most relevant subnetworks for each prediction. Finally, we utilize Grad-CAM to visualize disease-related regions, ensuring model interpretability.

The results show that the GMU-based model achieves 95.47% accuracy and specificity of 96.73% in the NC vs. AD classification task, outperforming state-of-the-art approaches. Additionally, the model successfully locates disease-related regions in both MRI and PET scans, with different activation patterns in each modality, according to Grad-CAM analysis. This result supports the effectiveness of a multimodal strategy in the diagnosis of AD by confirming the complementary nature of MRI and PET.

Keywords

Alzheimer's Disease, Multimodal, Neuroimaging data, Convolutional Neural Networks, Mixture of Experts

Περίληψη

Η νόσος Αλτσχάιμερ είναι μια μη αναστρέψιμη νευροεκφυλιστική ασθένεια που προκαλεί σοβαρές βλάβες στη σκέψη και αποτελεί την έβδομη κύρια αιτία θανάτου παγκοσμίως. Η έγκαιρη διάγνωση παίζει καθοριστικό ρόλο, ιδιαίτερα στο στάδιο της ήπιας γνωστικής εξασθένησης (άννοια), όπου η έγκαιρη παρέμβαση μπορεί να επιβραδύνει την εξέλιξη της νόσου. Νευροαπεικονιστικά δεδομένα, όπως οι απεικονίσεις MRI και PET, μπορούν να βοηθήσουν στην πρώιμη ανίχνευση της νόσου, παρέχοντας πληροφορίες για τις δομικές και λειτουργικές μεταβολές του εγκεφάλου. Ωστόσο, παρά τη διαθεσιμότητα πολλαπλών απεικονιστικών μεθόδων για τον ίδιο ασθενή, η ανάπτυξη πολυτροπικών μοντέλων που αξιοποιούν αυτές τις πληροφορίες παραμένει περιορισμένη.

Η παρούσα διπλωματική εργασία στοχεύει στην αντιμετώπιση αυτού του κενού, προτείνοντας και αξιολογώντας μοντέλα ταξινόμησης που χρησιμοποιούν τρισδιάστατες MRI και PET απεικονίσεις στο πλαίσιο μιας πολυτροπικής προσέγγισης. Αρχικά, εφαρμόζουμε ένα Τρισδιάστατο Συνελικτικό Νευρωνικό Δίκτυο για την εξαγωγή χαρακτηριστικών, ακολουθούμενο από τρεις τεχνικές συνδυασμού πληροφοριών: απλή συνένωση χαρακτηριστικών, Gated Multimodal Unit και μηχανισμό προσοχής με πύλη. Για τη βελτίωση της ακρίβειας ταξινόμησης και της υπολογιστικής πολυπλοκότητας, ενσωματώνουμε ένα Mixture of Experts μοντέλο, το οποίο επιλέγει δυναμικά τα πιο σχετικά υποδίκτυα για κάθε πρόβλεψη. Τέλος, χρησιμοποιούμε την τεχνική Grad-CAM για να οπτικοποιήσουμε τις περιοχές του εγκεφάλου που συμβάλλουν στις αποφάσεις του μοντέλου, διασφαλίζοντας τη διαφάνεια και την ερμηνευσιμότητα του συστήματος.

Τα αποτελέσματα δείχνουν ότι το GMU μοντέλο πέτυχε ακρίβεια 95.47% και ειδικότητα 96.73% στην ταξινόμηση υγιών και μη ασθενών, ξεπερνώντας τις υφιστάμενες μεθόδους αιχμής. Επιπλέον, σύμφωνα με την ανάλυση Grad-CAM, το μοντέλο εντοπίζει περιοχές του εγκεφάλου που σχετίζονται με τη νόσο σε απεικονίσεις MRI και PET, με διαφορετικά μοτίβα ενεργοποίησης για κάθε μέθοδο. Αυτό το αποτέλεσμα αποδεικνύει την αποτελεσματικότητα της πολυτροπικής στρατηγικής στη διάγνωση της νόσου Αλτσχάιμερ, επιβεβαιώνοντας την συμπληρωματική φύση των MRI και PET απεικονίσεων.

Λέξεις Κλειδιά

Νόσος Αλτσχάιμερ, Πολυτροπική προσέγγιση, Συνελικτικά Νευρωνικά Δίκτυα, Νευροαπεικονιστικά δεδομένα

to my parents

Ευχαριστίες

Αρχικά, θα ήθελα να εκφράσω την ευγνωμοσύνη μου στον επιβλέποντα μου, καθηγητή Δημήτριο Ασκούνη, για την εμπιστοσύνη που μου έδειξε και την ανάθεση ενός τόσο ενδιαφέροντος θέματος για τη διπλωματική μου εργασία.
Επίσης, θα ήθελα να ευχαριστήσω θερμά τον διδάκτορα Λουκά Ηλία, του οποίου η πολύτιμη καθοδήγηση και αμέριστη υποστήριξη υπήρξαν καταλυτικές για την ολοκλήρωση αυτής της εργασίας.
Τέλος, ένα μεγάλο ευχαριστώ στους φίλους μου και στην οικογένειά μου, που με τη συνεχή τους στήριξη, την κατανόηση και την ενθάρρυνσή τους στάθηκαν δίπλα μου σε κάθε βήμα αυτής της διαδρομής.

Athens, February 2025

Anthi-Maria Vozinaki

Table of Contents

List of Figures

List of Tables

Chapter 1

Εκτενής ελληνική περίληψη

1.1 Εισαγωγή

Η Νόσος Αλτσχάιμερ είναι μια χρόνια, νευροεκφυλιστική ασθένεια και η κυριότερη αιτία άνοιας, καθώς ευθύνεται για το 60% έως 80% των περιπτώσεων.. Η νόσος χαρακτηρίζεται από προοδευτική έκπτωση της μνήμης, δυσκολία στην επικοινωνία και τον προσανατολισμό, διακυμάνσεις στη διάθεση, απώλεια κινήτρων και σταδιακή μείωση της λειτουργικότητας. Το 2019, περίπου 5,8 εκατομμύρια Αμερικανοί άνω των 65 ετών διαγνώστηκαν με Αλτσχάιμερ, ενώ οι προβλέψεις δείχνουν δραματική αύξηση των περιστατικών τα επόμενα χρόνια. Επιπλέον, το εκτιμώμενο κόστος υγειονομικής περίθαλψης έφτασε τα 305 δισεκατομμύρια δολάρια το 2020, επιβαρύνοντας σημαντικά τα συστήματα υγείας. Οι αιτίες της νόσου αναπτύσσονται δεκαετίες πριν από την εμφάνιση των πρώτων συμπτωμάτων, γεγονός που καθιστά την έγκαιρη διάγνωση ιδιαίτερα δύσκολη.

Σε μοριακό επίπεδο, η νόσος σχετίζεται με τη συσσώρευση β-αμυλοειδών πλακών στο εξωτερικό των νευρικών κυττάρων και πρωτεΐνης tau στο εσωτερικό τους, οδηγώντας σε εκφυλισμό των νευρώνων, μείωση των συνάψεων και εγκεφαλική ατροφία, κυρίως στον ιππόκαμπο και τον φλοιό, περιοχές κρίσιμες για τη μνήμη και τη λήψη αποφάσεων. Η φυσιολογική λειτουργία του εγκεφάλου επηρεάζεται περαιτέρω από τη μειωμένη ικανότητά του να μεταβολίζει τη γλυκόζη, το κύριο καύσιμό του. Παράγοντες κινδύνου περιλαμβάνουν γενετικούς και περιβαλλοντικούς παράγοντες, καθώς και συνήθειες του τρόπου ζωής.

Όσον αφορά τη διάγνωση της νόσου, δεν υπάρχει μοναδική εξέταση που να την επιβεβαιώνει. Έτσι, οι ειδικοί βασίζονται σε συνδυασμό μεθόδων, ο οποίος περιλαμβάνει το ιστορικό του ασθενούς, νευρολογικές εξετάσεις, ακτινογραφίες εγκεφάλου και τεστ μνήμης. Επιπλέον, εξετάσεις αίματος βοηθούν στον αποκλεισμό άλλων αιτιών άνοιας, όπως όγκοι ή εγκεφαλικά επεισόδια. Η τεχνολογία απεικόνισης του εγκεφάλου έχει βελτιώσει σημαντικά τη διάγνωση. Η Τομογραφία Εκπομπής Ποζιτρονίων (PET) ανιχνεύει β-αμυλοειδείς πλάκες και τον μεταβολισμό της γλυκόζης, ενώ η Μαγνητική Τομογραφία (MRI) εντοπίζει ατροφία στον ιππόκαμπο και τον φλοιό. Ο συνδυασμός αυτών των μεθόδων αυξάνει την ακρίβεια της διάγνωσης και επιτρέπει πιο στοχευμένες θεραπείες [1, 2].

Η παρούσα διπλωματική εργασία χρησιμοποιεί εξετάσεις MRI και PET λόγω της συμπληρωματικής πληροφορίας που παρέχουν στη διάγνωση της νόσου. Στη συνέχεια, παρουσιάζονται οι περιορισμοί

προηγούμενων μεθόδων στο συγκεκριμένο πρόβλημα, περιγράφεται το σύνολο δεδομένων που χρησιμοποιήθηκε, η προεπεξεργασία που εφαρμόστηκε, η προτεινόμενη μεθοδολογία, καθώς και τα αποτελέσματα και τα συμπεράσματα που προέκυψαν.

1.2 Περιορισμοί προηγούμενων μεθόδων

Βιβλιογραφική ανασκόπηση των υφιστάμενων μεθόδων αναδεικνύει ορισμένους περιορισμούς και ελλείψεις στο συγκεκριμένο πρόβλημα. Αρχικά, οι πολυτροπικές προσεγγίσεις δεν έχουν μελετηθεί επαρκώς. Ακόμα και στις περιπτώσεις όμως όπου έχουν εφαρμοστεί, η ενσωμάτωση των διαφορετικών τροπικοτήτων περιορίζεται συνήθως σε μια απλή συνένωσή τους, παραβλέποντας τις μεταξύ τους αλληλεπιδράσεις.

Επιπλέον, ένας σημαντικός περιορισμός αφορά την αδυναμία των υφιστάμενων μεθόδων να προσαρμόζονται δυναμικά στα δεδομένα εισόδου. Οι περισσότερες προσεγγίσεις βασίζονται σε πυκνά επίπεδα για την τελική κατηγοριοποίηση των ασθενών, τα οποία, λόγω της στατικής τους φύσης, δυσκολεύονται να προσαρμοστούν σε πιο περίπλοκα και ετερογενή δεδομένα. Αυτό αποτελεί κρίσιμο ζήτημα στην περίπτωση της άνοιας, μιας διαταραχής που από τη φύση της παρουσιάζει μεγάλο διαγνωστικό βαθμό δυσκολίας.

Ένας ακόμη σημαντικός περιορισμός αφορά την ερμηνευσιμότητα των μοντέλων. Τα περισσότερα μοντέλα που έχουν αναπτυχθεί λειτουργούν ως ¨μαύρα κουτιά¨, όπου λαμβάνουν δεδομένα εισόδου και παράγουν αποτελέσματα εξόδου, χωρίς να παρέχουν πληροφορίες σχετικά με τα χαρακτηριστικά που συνέβαλαν στην τελική τους απόφαση.

Αυτό αποτελεί σημαντικό πρόβλημα στην κλινική πράξη, καθώς για να μπορέσει ένας γιατρός να αξιολογήσει τα αποτελέσματα του μοντέλου, είναι απαραίτητο να γνωρίζει ποιες περιοχές του εγκεφάλου θεωρούνται παθολογικές. Η έλλειψη διαφάνειας δυσχεραίνει την εμπιστοσύνη και την εφαρμογή αυτών των μεθόδων στη διάγνωση και την κλινική λήψη αποφάσεων.

1.3 Δεδομένα και Προεπεξεργασία

Στην παρούσα εργασία χρησιμοποιήθηκε το σύνολο δεδομένων ADNI, το οποίο προέρχεται από μια ερευνητική συνεργασία που ξεκίνησε το 2004 με στόχο την ανάπτυξη βιοδεικτών για την έγκαιρη διάγνωση και την παρακολούθηση της εξέλιξης της νόσου Αλτσχάιμερ. Το ADNI συγκεντρώνει δεδομένα από πολλαπλές πηγές, όπως MRI, FDG-PET, γενετικές αναλύσεις και κλινικές αξιολογήσεις, δημιουργώντας ένα ολοκληρωμένο και πολύτιμο σύνολο πληροφοριών για τη νόσο.

Οι συμμετέχοντες περιλαμβάνουν υγιή άτομα (NC), άτομα με ήπια γνωστική εξασθένηση (MCI) και ασθενείς με Αλτσχάιμερ (AD). Ένα από τα κύρια πλεονεκτήματα του ADNI είναι η πολιτική ανοιχτής πρόσβασης, που επιτρέπει σε ερευνητές από όλο τον κόσμο να αξιοποιούν τα δεδομένα μέσω της πλατφόρμας LONI.

Προκειμένου να επιλεχθούν οι πιο ποιοτικές εικόνες του συνόλου δεδομένων ακολουθήθηκε η μεθοδολογία των Song et al [3]. Συγκεκριμένα, επιλέχθηκαν MRI εικόνες που έχουν υποστεί gradwarp για διόρθωση της γεωμετρικής παραμόρφωσης της εικόνας, B1 correction που αντιμετωπίζει τις διακυμάνσεις έντασης και N3 bias field correction για όξυνση κορυφών. Για τις PET εικόνες επιλέχθηκαν εκείνες που έχουν ευθυγραμμιστεί με το πρώτο καρέ (Co-registration of dynamic frames), έχουν υπολογιστεί ως μέσος όρος των καρέ (Averaging) και έχουν τυποποιηθεί σε σταθερό πλέγμα 160 × 160 × 96 voxel. Έτσι παρόλο που τα δεδομένα του συνόλου είναι πολύ περισσότερα καταλήξαμε σε μόνο 379 ασθενείς προκειμένου να έχουμε υψηλής ποιότητας δεδομένα.

Στη συνέχεια, η προεπεξεργασία των εικόνων MRI ξεκινά με την αφαίρεση του κρανίου (skull-stripping) μέσω του εργαλείου FSL (FMRIB Software Library) και συγκεκριμένα του εργαλείου Brain Extraction Tool (BET), απομακρύνοντας μη εγκεφαλικούς ιστούς όπως κρανίο και δέρμα, για πιο στοχευμένη ανάλυση. Ο συντελεστής κατωφλίου ορίζεται στο 0.5 για ισορροπημένη εξαγωγή, ενώ εφαρμόζεται διόρθωση του bias field για βελτίωση της ποιότητας της εικόνας. Στη συνέχεια οι εικόνες MRI ευθυγραμμίζονται με το πρότυπο MNI152 μέσω του εργαλείου FLIRT, που διορθώνει χωρικές διαφορές με γραμμικό μετασχηματισμό (μετατοπίσεις, περιστροφές, κλιμάκωση). Αντίστοιχα, οι εικόνες PET υφίστανται skull-stripping και co-registration με τις αντίστοιχες MNI ευθυγραμμισμένες MRI εικόνες, διατηρώντας ενιαίο προσανατολισμό και ανάλυση, διευκολύνοντας το μοντέλο στην μετέπειτα εξαγωγή χαρακτηριστικών.

Για αποδοτικότερη επεξεργασία, οι εικόνες PET και MRI περικόπτονται σε ανάλυση 160 × 180 × 80, με στόχο τη μείωση της υπολογιστικής πολυπλοκότητας. Στο Σχήμα 1.1 παρουσιάζονται τα στάδια της προεπεξεργασίας των δεδομένων MRI και PET για το ίδιο, τυχαίο άτομο με υγιή εγκέφαλο.

1.4 Προτεινόμενη Μεθοδολογία

Η συνολική μεθοδολογία της παρούσας εργασίας παρουσιάζεται στην Εικόνα 1.2. Αποτελείται από την εξαγωγή χαρακτηριστικών από τις MRI και PET εικόνες, με τη χρήση ενός 3D συνελικτικού νευρωνικού δικτύου. Τα εξαγόμενα χαρακτηριστικά συνενώνονται με τρεις διαφορετικούς τρόπους, προκειμένου να διερευνηθεί η βέλτιστη στρατηγική συνδυασμού τους. Στη συνέχεια, η τελική κατηγοριοποίηση των ασθενών πραγματοποιείται μέσω του μοντέλου Mixture of Experts. Επιπλέον, εφαρμόζεται ανάλυση Grad-CAM με στόχο την ερμηνεία των προβλέψεων του μοντέλου, επιτρέποντας την οπτικοποίηση των περιοχών που συνέβαλαν στην τελική απόφαση.

Όσον αφορά την εξαγωγή χαρακτηριστικών χρησιμοποιήσαμε 3D Συνελικτικά Νευρωνικά Δίκτυα, τα οποία είναι εξειδικευμένα νευρωνικά δίκτυα σχεδιασμένα για την επεξεργασία δεδομένων με χωρική δομή, όπως εικόνες και βίντεο. Διατηρούν τις χωρικές σχέσεις και τα ιεραρχικά χαρακτηριστικά μέσω μιας αρχιτεκτονικής βασισμένης σε στρώματα.

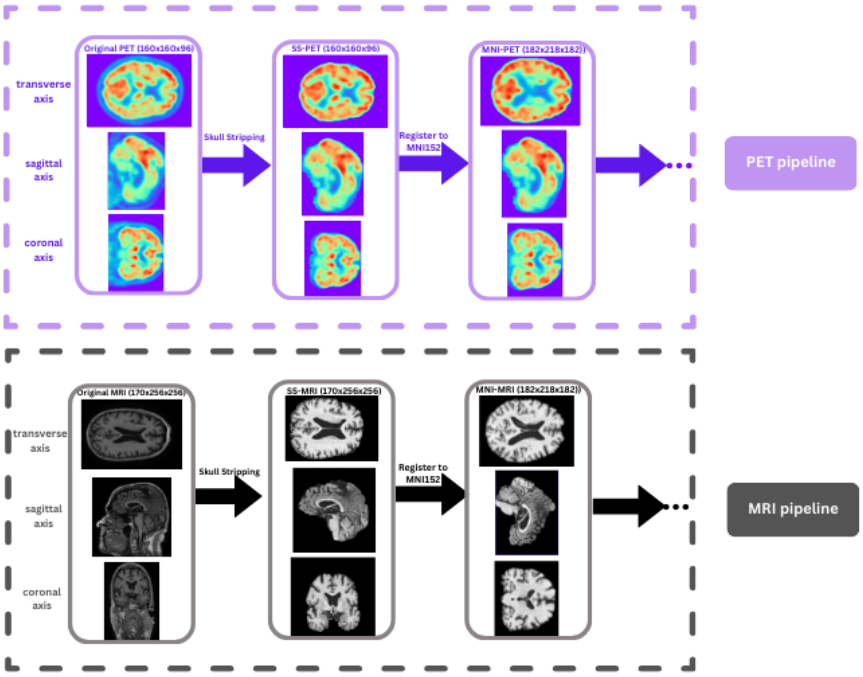


Figure 1.1. *Διαδικασία προεπεξεργασίας δεδομένων MRI και PET*

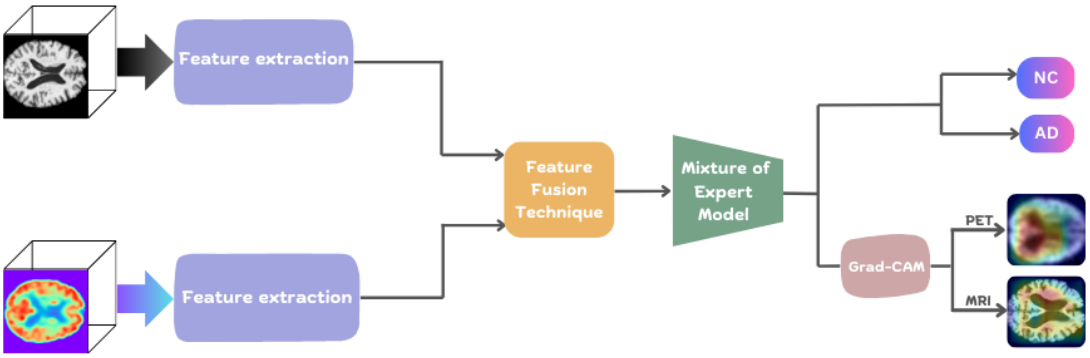


Figure 1.2. *Η συνολική μεθοδολογία αυτής της εργασίας*

Ένα τυπικό Συνελικτικό Νευρωνικό Δίκτυο περιλαμβάνει συνελικτικά, ενεργοποίησης, υποδειγματοληψίας (pooling) και πλήρως συνδεδεμένα στρώματα. Τα συνελικτικά στρώματα ανιχνεύουν μοτίβα, τα στρώματα pooling μειώνουν τις διαστάσεις, και τα πλήρως συνδεδεμένα στρώματα συγκεντρώνουν τα χαρακτηριστικά για την τελική πρόβλεψη.

Το συνελικτικό στρώμα αποτελεί το πιο βασικό στοιχείο τους, καθώς εφαρμόζει φίλτρα στα δεδομένα εισόδου μέσω της πράξης της συνέλιξης, επιτρέποντας την ανίχνευση τοπικών χαρακτηρισ-

τικών στις εικόνες. Η πράξη της συνέλιξης περιλαμβάνει τη μετακίνηση κάθε φίλτρου πάνω στα δεδομένα εισόδου και τον υπολογισμό του σταθμισμένου αθροίσματος των τιμών εντός του πεδίου αποδοχής (receptive field). Αυτό παράγει έναν χάρτη χαρακτηριστικών που αναδεικνύει σημαντικά χωρικά μοτίβα και χαρακτηριστικά, όπως ακμές, υφές και σχήματα. Οι τιμές του πεδίου αποδοχής εξαρτώνται από τα δεδομένα εισόδου: για μονοκαναλικά δεδομένα, όπως ασπρόμαυρες εικόνες, οι τιμές αναπαριστούν εντάσεις φωτεινότητας, ενώ για multi-channel δεδομένα, όπως RGB εικόνες, περιέχουν πληροφορίες για τα τρία χρώματα.

Το τρισδιάστατο συνελικτικό στρώμα αποτελεί επέκταση της δισδιάστατης συνέλιξης, προσαρμοσμένη για τρισδιάστατα δεδομένα, όπως ακολουθίες βίντεο ή ογκομετρικές ιατρικές εικόνες (π.χ. αξονικές και μαγνητικές τομογραφίες). Σε αντίθεση με τη δισδιάστατη συνέλιξη, όπου ο πυρήνας εφαρμόζεται μόνο στο ύψος και το πλάτος, στην τρισδιάστατη προστίθεται και η διάσταση του βάθους. Αυτή η προσθήκη επιτρέπει την εξαγωγή ογκομετρικών χαρακτηριστικών, καθιστώντας τη μέθοδο ιδιαίτερα αποτελεσματική στην ανίχνευση σύνθετων ανωμαλιών, όπως όγκοι ή βλάβες που εκτείνονται σε πολλαπλές τομές ιατρικών εικόνων.

Για την εξαγωγή χαρακτηριστικών, ακολουθούμε τη μεθοδολογία που απεικονίζεται στην Εικόνα 1.3, όπου χρησιμοποιούμε δύο πανομοιότυπα αλλά ξεχωριστά μονοπάτια για τις MRI και PET εικόνες. Κάθε μονοπάτι αποτελείται από τέσσερα συνελικτικά επίπεδα, ακολουθούμενα από pooling επίπεδα για τη μείωση των διαστάσεων. Επιπλέον, εφαρμόζεται ένα global average pooling επίπεδο για περαιτέρω συμπίεση της πληροφορίας. Τέλος, χρησιμοποιείται ένα dropout επίπεδο για την αποφυγή της υπερεκπαίδευσης. Από τη διαδικασία αυτή, εξάγονται 128 χαρακτηριστικά για τις MRI εικόνες και 128 χαρακτηριστικά για τις PET εικόνες.

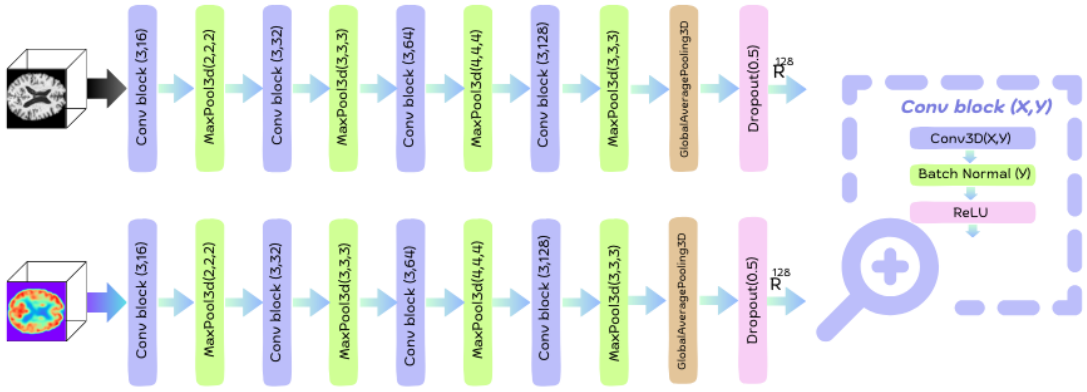


Figure 1.3. *Μεθοδολογία εξαγωγής χαρακτηριστικών*

Τα 128 αυτά χαρακτηριστικά από κάθε τροπικότητα τα συνενώνουμε σε ένα κοινό διάνυσμα με

τρεις διαφορετικές προσεγγίσεις. Η πρώτη αφορά την απλή συνένωση τους με αποτέλεσμα ένα διάνυσμα διάστασης 256, όπως φαίνεται και στην εικόνα 1.4.

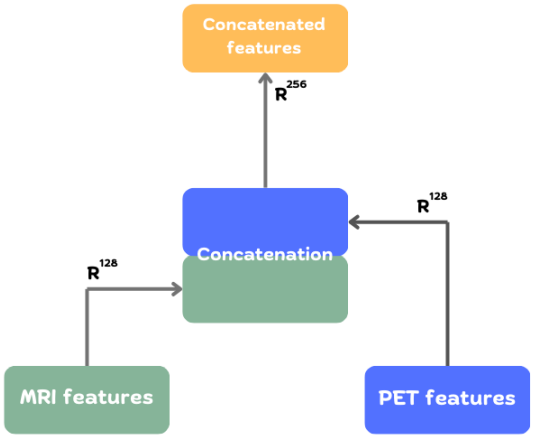


Figure 1.4. *Απλή συνένωση χαρακτηριστικών σε ένα κοινό διάνυσμα*

Η δεύτερη προσέγγιση αφορά έναν έξυπνο μηχανισμό λήψης αποφάσεων, την Gated Multimodal Unit (GMU). Η GMU είναι ένας μηχανισμός σχεδιασμένος για την προσαρμοστική ενοποίηση χαρακτηριστικών από πολλαπλές τροπικότητες. Ρυθμίζοντας δυναμικά τη συμβολή κάθε τροπικότητας, επιτρέπει στο μοντέλο να καταστέλλει τη λιγότερο σημαντική. Για δύο τροπικότητες (f_1 και f_2), η GMU λειτουργεί ως εξής:

$$h_1 = \tanh(W_1 f_1 + b_1), \tag{1.1}$$

$$h_2 = \tanh(W_2 f_2 + b_2), \tag{1.2}$$

$$z = \sigma(W_z [f_1; f_2] + b_z), \tag{1.3}$$

$$h = z \odot h_1 + (1 - z) \odot h_2, \tag{1.4}$$

Οι παράμετροι της GMU αναπαρίστανται ως:

$$\Theta = \{W_1, W_2, W_z\}, \tag{1.5}$$

όπου:

- f_1 και f_2 είναι τα χαρακτηριστικά εισόδου από τις δύο τροπικότητες.
- W_1, W_2, W_z είναι οι εκπαιδεύσιμοι πίνακες βαρών για τις αντίστοιχες τροπικότητες και τον μηχανισμό gating, σχηματίζοντας το σύνολο παραμέτρων Θ.
- b_1, b_2, b_z είναι οι αντίστοιχοι όροι bias.
- z είναι το gating διάνυσμα, το οποίο υπολογίζεται μέσω της συγμοειδής συνάρτησης (σ) και καθορίζει τη σχετική σημασία κάθε τροπικότητας.
- h_1 και h_2 είναι οι μετασχηματισμένες αναπαραστάσεις των χαρακτηριστικών εισόδου, που διέρχονται από μια tanh συνάρτηση ενεργοποίησης.

- h είναι η τελική συγχωνευμένη αναπαράσταση, που υπολογίζεται ως ένας σταθμισμένος συνδυασμός των h_1 και h_2, με το gating διάνυσμα z να ελέγχει τη συμβολή κάθε τροπικότητας.

Το σύνολο παραμέτρων βελτιστοποιείται κατά τη διαδικασία εκπαίδευσης ώστε να διασφαλίζει ότι ο μηχανισμός gating προσαρμόζεται δυναμικά στα δεδομένα εισόδου. Η αρχιτεκτονική της GMU φαίνεται στην εικόνα 1.5 και το διάνυσμα χαρακτηριστικών που προκύπτει έχει επίσης 128 χαρακτηριστικά.

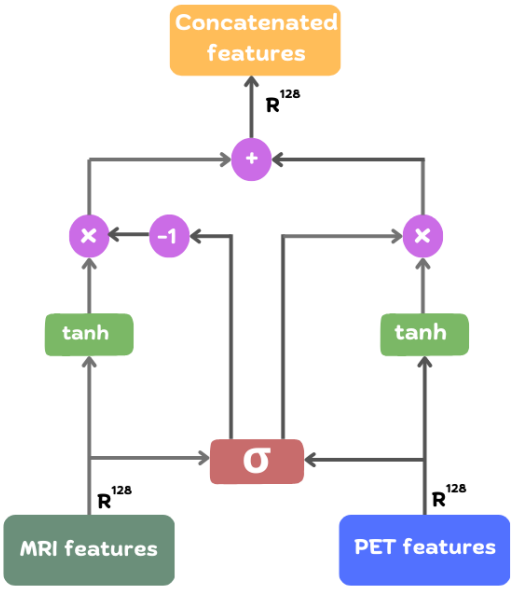


Figure 1.5. *Χρήση GMU για τη συνένωση των χαρακτηριστικών*

Η τρίτη προσέγγιση που χρησιμοποιήσαμε για τη συνένωση των χαρακτηριστικών είναι η χρήση του Gated self-attention. Ο κύριος στόχος μας με αυτή την τεχνική είναι να μοντελοποιήσουμε τις ενδοτροπικές και διατροπικές αλληλεπιδράσεις ταυτόχρονα.

Το self-attention είναι ένας μηχανισμός που επιτρέπει σε κάθε στοιχείο μιας ακολουθίας να δίνει βάρη στη συσχέτισή του με όλα τα υπόλοιπα, βελτιώνοντας έτσι την κατανόηση των εξαρτήσεων στο πλαίσιο μιας εισόδου.

Ο Yu et al. [4] πρότεινε μια παραλλαγή του κλασικού self-attention που με τη χρήση μιας μάσκας M, εντοπίζονται και διατηρούνται μόνο οι πιο σημαντικές συσχετίσεις:

$$M = \sigma\left(FC_g\left(FC_g^q(Q) \odot FC_g^k(K)\right)\right), \tag{1.6}$$

όπου FC_g^q και FC_g^k είναι πλήρως συνδεδεμένα επίπεδα που προβάλλουν τα Q και K σε έναν κοινό χώρο. Η σιγμοειδής συνάρτηση $\sigma(\cdot)$ εξασφαλίζει ότι οι τιμές του M κυμαίνονται στο διάστημα $(0,1)$, φιλτράροντας αποτελεσματικά τα λιγότερο σημαντικά χαρακτηριστικά.

Επιπλέον, η μάσκα M ορίζεται ως:

$$M \in \mathbb{R}^{m \times 2} \tag{1.7}$$

που αντιστοιχεί στις δύο μάσκες $M_q \in \mathbb{R}^m$ και $M_k \in \mathbb{R}^m$, οι οποίες σχετίζονται με τα χαρακτηριστικά Q και V, αντίστοιχα.
Στη συνέχεια, το αποτέλεσμα για το attention map υπολογίζεται ως εξής:

$$A^g = \text{softmax}\left(\frac{(Q \odot \tilde{M}_Q)(K \odot \tilde{M}_K)^T}{\sqrt{d}}\right). \tag{1.8}$$

Τέλος, η τελική αναπαράσταση χαρακτηριστικών προκύπτει ως:

$$F = A^g V. \tag{1.9}$$

Δεδομένων των δύο τροπικοτήτων, το πρώτο βήμα που εκτελέσαμε είναι η συνένωση των αναπαραστάσεών τους:

$$Z = [MRI; PET] \tag{1.10}$$

Εδώ, η συνενωμένη αναπαράσταση Z ορίζεται ως $Z \in \mathbb{R}^{m \times d}$, όπου $m = 4 + 4$ και $d = 128$. Για να διασφαλιστεί ότι διατηρείται η χωρική πληροφορία, εξάγουμε χαρακτηριστικά πριν από την εφαρμογή του global average pooling, αντιμετωπίζοντας τις χωρικές διαστάσεις ως tokens, όπως φαίνεται στο Σχήμα 1.6. Αυτό επιτρέπει στον μηχανισμό self-attention να καταγράψει χωρικές εξαρτήσεις μεταξύ των δεδομένων.

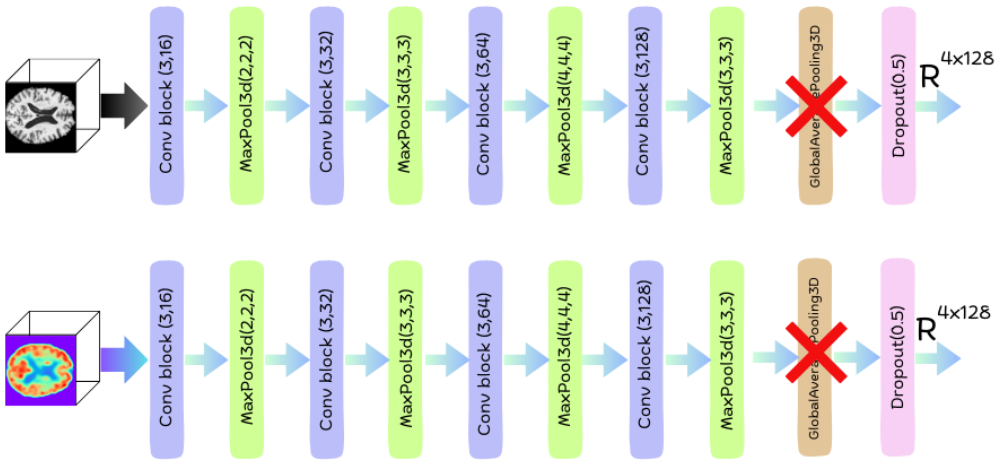


Figure 1.6. *Εξαγωγή χαρακτηριστικών χωρίς Global Average Pooling*

Στη συνέχεια, αυτή η αναπαράσταση χρησιμοποιείται για τον υπολογισμό των πινάκων query (Q),

key (K) και value (V):

$$Q = Z, \quad K = Z, \quad V = Z. \tag{1.11}$$

Έπειτα, εφαρμόζονται οι εξισώσεις 1.6, 1.8 και 1.9, ακολουθώντας τη ροή που απεικονίζεται στο Σχήμα 1.8.

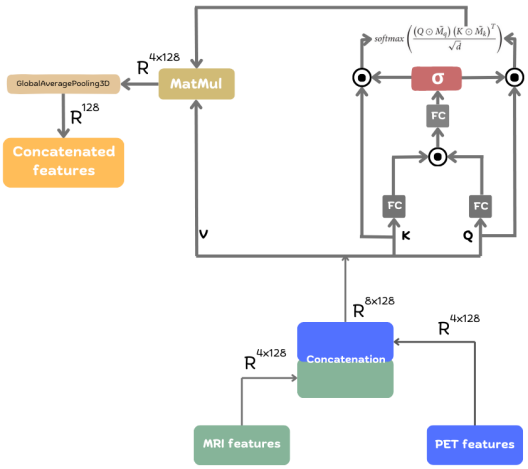


Figure 1.7. *Χρήση Gated self-attention για τη συνένωση των χαρακτηριστικών*

Έπειτα, τα συνενωμένα χαρακτηριστικά δίνονται σε μοντέλο Mixture of Experts (MoE) για την τελική κατηγοριοποίηση των ασθενών. Το MoE αποτελεί μια πρωτοποριακή τεχνική στη μηχανική μάθηση, όπου εξειδικευμένα υπομοντέλα ενεργοποιούνται δυναμικά ανάλογα με την είσοδο, βελτιώνοντας έτσι την αποδοτικότητα και την προσαρμοστικότητα των νευρωνικών δικτύων σε πολύπλοκα προβλήματα. Εισήχθη από τους Jacobs et al. το 1991 [5] και ακολουθεί την αρχή του divide-and-conquer διαιρώντας τον χώρο του προβλήματος σε πολλαπλούς εξειδικευμένους "ειδικούς". Παρόμοια με μια ομάδα ειδικών που συνεργάζονται για την επίλυση ενός πολύπλοκου προβλήματος, κάθε expert στο MoE φέρει μοναδικές δεξιότητες για τη διαχείριση συγκεκριμένων υπο-προβλημάτων, επιτρέποντας στο μοντέλο να επιτύχει υψηλή απόδοση. Η αρθρωτή σχεδίασή του παρέχει επίσης ευελιξία και ερμηνευσιμότητα, επιτρέποντας την ανεξάρτητη ανάλυση και ρύθμιση της συνεισφοράς κάθε expert.

Το gating network, συχνά αναφερόμενο ως router, αποτελεί ένα κρίσιμο στοιχείο της αρχιτεκτονικής. Ο κύριος ρόλος του είναι να αναλύει τα δεδομένα εισόδου και να προσδιορίζει ποιοι experts είναι πιο κατάλληλοι για τη διαχείριση του εκάστοτε προβλήματος. Αυτή η διαδικασία περιγράφεται μαθηματικά ως:

$$y = \sum_{i=1}^{n} G(x)_i E_i(x) \tag{1.12}$$

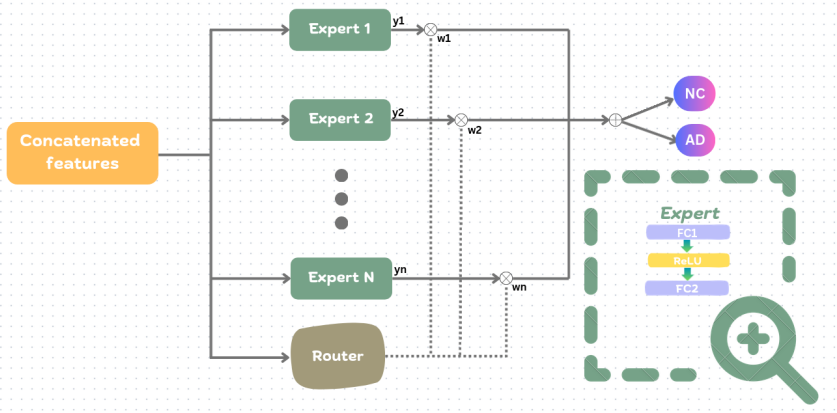


Figure 1.8. *Χρήση Mixture of Experts μοντέλο για την τελική κατηγοριοποίηση*

όπου $G(x)_i$ είναι το gating weight που αποδίδεται στον i-οστό expert βάσει των χαρακτηριστικών της εισόδου x, ενώ $E_i(x)$ είναι η έξοδος του i-οστού expert. Το gating network διασφαλίζει ότι επιλέγονται οι πιο σχετικοί experts, αποδίδοντας δυναμικά τα κατάλληλα βάρη.

Ο μηχανισμός Softmax Gating, που εισήγαγαν οι Jacobs et al. το 1994 [6], υπολογίζει τα gating weights πολλαπλασιάζοντας την είσοδο x με έναν εκπαιδεύσιμο πίνακα βαρών W_g και κανονικοποιώντας τις τιμές μέσω της softmax function:

$$G_\sigma(x) = \text{Softmax}(x \cdot W_g) \tag{1.13}$$

Το αποτέλεσμα $G_\sigma(x)$ αναπαριστά τη σημασία κάθε expert, εξασφαλίζοντας ότι τα βάρη είναι μη αρνητικά και αθροίζουν στο 1. Παρόλο που αυτή η μέθοδος είναι απλή και αποτελεσματική, ενεργοποιεί όλους τους experts για κάθε είσοδο, αυξάνοντας την υπολογιστική πολυπλοκότητα. Οι μηχανισμοί sparse gating αντιμετωπίζουν αυτό το ζήτημα, επιλέγοντας μόνο τους πιο σχετικούς experts για κάθε είσοδο.

Η έννοια της αραιής ενεργοποίησης όπως περιγράφηκε από τους Shazeer et al., παίζει κρίσιμο ρόλο στη βελτίωση της υπολογιστικής πολυπλοκότητας χωρίς να μειώνει τη χωρητικότητα του μοντέλου.

$$G(x) = \text{Softmax}(\text{KeepTopK}(H(x), k)), \tag{1.14}$$

όπου $G(x)$ είναι το gating διάνυσμα, η πράξη KeepTopK διασφαλίζει αραιή ενεργοποίηση διατηρώντας μόνο τις k τιμές στο $H(x)$ και $H(x)$ είναι ένα διάνυσμα με τα αρχικά gating scores:

$$H(x)_i = (x \cdot W_g)_i + \text{StandardNormal}() \cdot \text{Softplus}((x \cdot W_{\text{noise}})_i). \tag{1.15}$$

Η εκπαίδευση των MoE μοντέλων απαιτεί την αντιμετώπιση της πρόκλησης της ισορροπημένης χρήσης των experts. Το μοντέλο ενσωματώνει δύο ρυθμιστικούς παράγοντες, το load-balancing loss, για να διανέμονται οι είσοδοι πιο ομοιόμορφα μεταξύ των experts και το importance loss, που έχει ως στόχο να μην μένει κανένας expert αχρησιμοποίητος. Δίνονται από τις παρακάτω σχέσεις:

$$L_{\text{importance}}(X) = w_{\text{importance}} \cdot \text{CV}(\text{Importance}(X))^2, \tag{1.16}$$

$$L_{\text{load}}(X) = w_{\text{load}} \cdot \text{CV}(\text{Load}(X))^2. \tag{1.17}$$

Αυτοί οι ρυθμιστικοί παράγοντες ενσωματώνονται στην κύρια συνάρτηση κόστους ώστε να σχηματίσουν τον συνολικό στόχο εκπαίδευσης:

$$L_{\text{total}} = L_{\text{loss}} + a * (L_{\text{importance}} + L_{\text{load}}), \tag{1.18}$$

όπου L_{loss} είναι η απώλεια για συνολικό σύστημα (π.χ. cross-entropy loss) και το a είναι μία υπερπαράμετρος που καθορίζει τη σχετική βαρύτητα των όρων regularization.
Παρακάτω φαίνεται ο πίνακας με τις υπερπαραμέτρους που χρησιμοποιήθηκαν για την εξαγωγή των πειραμάτων:

Table 1.1. *Βέλτιστες τιμές υπερπαραμέτρων*

Hyperparameter	**Value**
Learning rate (η)	1×10^{-4}
Weight decay	0.1
Batch size	4
Dropout rate	0.5
Number of experts (n)	5
Selected experts (k)	4
Parameter a in loss	0.6
Optimizer	Adam

1.5 Αποτελέσματα

Ο Πίνακας 1.2 συνοψίζει την απόδοση ταξινόμησης του τελικού μας μοντέλου χρησιμοποιώντας τρεις διαφορετικές μεθόδους συγχώνευσης: Concatenation, GMU και Attention. Το μοντέλο αξιολογείται σε τρία προβλήματα ταξινόμησης: NC vs MCI, MCI vs AD και NC vs AD.

Table 1.2. *Αποτελέσματα του συνολικού μας μοντέλου*

Fusion Method	**Task**	**ACC**	**SEN**	**SP**	**AUC**
Concatenation	NC vs MCI	78.25±3.2	75.43±4.1	79.32±2.1	76.56±3.9
	MCI vs AD	80.13±5.3	79.24±5.8	81.21±5.5	76.83±8.1
	NC vs AD	89.52±3.4	87.25±3.2	89.98±4.1	89.64±2.3
GMU	NC vs MCI	**80.46** ± 3.9	**79.71** ± 4	**81.76** ± 3.9	**80.51** ± 3.5
	MCI vs AD	79.13±1.1	77.23±3.3	81.36±4.2	79.94±1.5
	NC vs AD	**95.47** ± 2.1	**94.31** ± 3.2	**96.73** ± 1.8	**95.41** ± 2.6
Attention	NC vs MCI	80.15±2.2	78.35±5.4	83.56±2.6	77.46±1.9
	MCI vs AD	**82.08** ± 2.1	**81.43** ± 1.8	**85.24** ± 2.7	**80.48** ± 3
	NC vs AD	91.53±4.7	92.28±4.4	91.07±4.7	92.29±5.2

Το Σχήμα 7.1 απεικονίζει τις οπτικοποιήσεις Grad-CAM που εφαρμόστηκαν σε MRI και PET σαρώσεις ενός ασθενούς θετικού στη νόσο.

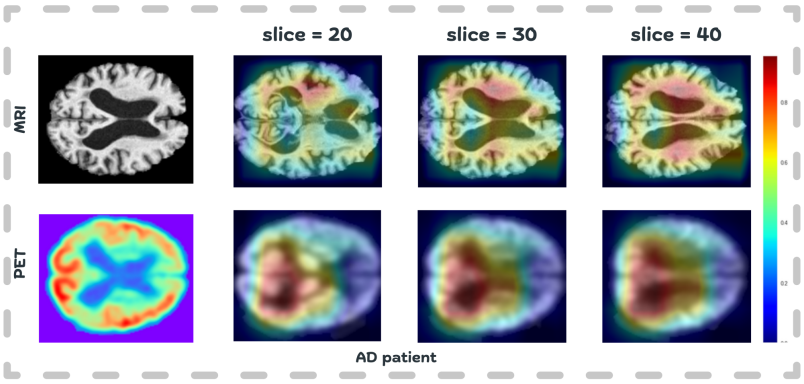


Figure 1.9. *Αποτελέσματα Γραδ-ΚΑΜ για έναν ασθενή με ΑΔ*

Ο Πίνακας 1.3 παρουσιάζει τα αποτελέσματα αξιολόγησης του καλύτερου μας μοντέλου, συγκριτικά με υπάρχουσες τεχνικές:

Table 1.3. *Σύγκριση των αποτελεσμάτων μας με προηγούμενες μεθόδους*

Architecture	Task	Accuracy	Sensitivity	Specificity	ROC-AUC
Unimodal approaches (MRI)					
Support Vector Machine [7]	NC vs AD	–	86	86	–
Random Forest [8]	NC vs AD	–	88.6	92	–
3D CNNs [9]	NC vs AD	95.39	–	–	–
	NC vs MCI	92.11	–	–	–
	MCI vs AD	86.84	–	–	–
Multimodal approaches					
Stacked Auto-Encoders [10]	NC vs AD	87.76	88.57	87.22	–
	NC vs MCI	76.92	74.29	78.13	–
Multiscale DNN [11]	NC vs AD	84.6	80.2	91.8	–
3D CNNs [3]	NC vs AD	94.11	**94.44**	95.04	
	NC vs MCI	**88.48**	**93.44**	**85.60**	–
	MCI vs AD	**84.83**	71.19	**94.69**	–
2D&3D CNNs [12]	NC vs AD	95.00	93.33	96.66	93.00
Our best-performing Model					
GMU	NC vs AD	**95.47**	94.31	**96.73**	**95.41**
GMU	NC vs MCI	80.46	79.71	81.76	**80.51**
Attention	MCI vs AD	82.08	**81.43**	85.24	**80.48**

Για την αξιολόγηση της συμβολής του μοντέλου MoE, το αντικαταστήσαμε με μια απλή αρχιτεκτονική που αποτελείται από τρία πλήρως συνδεδεμένα επίπεδα και αξιολογήσαμε την απόδοσή του χρησιμοποιώντας τρεις διαφορετικές μεθόδους συγχώνευσης για τα χαρακτηριστικά MRI και PET: Concatenation, GMU και Attention-based.

Ο Πίνακας 1.5 παρουσιάζει τις μετρικές απόδοσης των unimodal μοντέλων (MRI και PET) για τα τρία προβλήματα ταξινόμησης.

Table 1.4. *Αποτελέσματα χωρίς το μοντέλο MoE*

Fusion Method	Task	ACC	SEN	SP	AUC
Concatenation	NC vs MCI	67.4±0.8	76.2±6	57.4±4.2	63.1±4.9
	MCI vs AD	68.5±2.8	75.2±5	59.3±3.1	67.4±3.9
	NC vs AD	86.48±5.2	84.32±4.2	87.18±5.6	84.8±4.8
GMU	NC vs MCI	70.5±2.9	74.2±4.1	65.6±4.8	72.3±3.9
	MCI vs AD	69.58±2.7	69.88±3.4	67.13±2.1	67.55±5.6
	NC vs AD	85.65±7.2	84.51±5.6	88.61±5.2	81.25±7.8
Attention	NC vs MCI	68.4±3.8	70.2±5.1	66.8±4.5	67.1±4.1
	MCI vs AD	73.45±2.8	75.5±4	72.3±3.5	74.3±2.9
	NC vs AD	85±8.8	84.91±6.6	84.78±7.2	83.1±7.4

Table 1.5. *Αποτελέσματα των μοντέλων με μία τροπικότητα*

Fusion Method	Task	ACC	SEN	SP	AUC
MRI	NC vs MCI	67.38±0.7	86.17±6	40.35±8.4	63.08±5
	MCI vs AD	64.29±4.6	67.56±5.6	62.19±4.8	61.14±5.5
	NC vs AD	75.32±3.5	80.17±11.1	65.31±10.8	76.13±5.3
PET	NC vs MCI	72.39±3.8	70.15±4.8	76.21±4.2	73.65±4.9
	MCI vs AD	70.81±2.4	68.57±4.2	75.42±4.1	69.6±5.8
	NC vs AD	81.1±1.6	81±1.6	81.88±2.6	84±6

1.6 Συμπεράσματα και Μελλοντικές Επεκτάσεις

Σε αυτή τη διπλωματική εργασία, στόχος μας ήταν η ανάπτυξη ενός αξιόπιστου συστήματος διάγνωσης της Νόσου Αλτσχάιμερ. Δεδομένου ότι η νόσος αποτελεί μία από τις κύριες αιτίες θανάτου, η έγκαιρη ανίχνευση είναι ζωτικής σημασίας. Ιδιαίτερη έμφαση δόθηκε στον εντοπισμό ασθενών με MCI, καθώς η διάγνωση μπορεί να επιβραδύνει την εξέλιξη της νόσου.

Αναλύσαμε MRI και PET σαρώσεις ασθενών ταξινομημένων σε τρεις κατηγορίες: NC, MCI και AD. Τα χαρακτηριστικά εξήχθησαν μέσω ενός 3D CNN, ενώ χρησιμοποιήσαμε τρεις μεθόδους συγχώνευσης: Απλή συνένωση χαρακτηριστικών, GMU, Gated Self-Attention. Χρησιμοποιήσαμε MoE για βελτίωση της απόδοσης, ενώ εφαρμόσαμε Grad-CAM για την ερμηνεία των προβλέψεων του μοντέλου. Η αξιολόγηση του συνολικού μοντέλου έδειξε ότι το GMU μοντέλο ήταν το καλύτερο, φτάνοντας 95.47% ακρίβεια στην ταξινόμηση NC vs AD.

Επίσης πραγματοποιήσαμε μελέτες αφαίρεσης για να αξιολογήσουμε τη συνεισφορά των διαφορετικών στοιχείων του μοντέλου μας. Η αντικατάσταση του MoE με πλήρως συνδεδεμένα επίπεδα οδήγησε σε χειρότερη απόδοση, αναδεικνύοντας τη σημασία του στην επιλογή σχετικών χαρακ-

τηριστικών. Η χρήση μονοτροπικών δεδομένων (MRI ή PET) οδήγησε σε χαμηλότερη ακρίβεια, επιβεβαιώνοντας το πλεονέκτημα της πολυτροπικής προσέγγισης. Το PET ξεπέρασε το MRI σε όλες τις ταξινομήσεις, κάτι που επιβεβαιώθηκε και από την ανάλυση των βαρών στο GMU μοντέλο.

Συγκρίνοντας το προτεινόμενο μοντέλο με σύγχρονες προσεγγίσεις, παρατηρήσαμε υπεροχή για την ταξινόμηση NC vs. AD, καθώς και υψηλότερη ευαισθησία για NC vs. MCI. Τέλος, η ανάλυση Grad-CAM έδειξε ότι το μοντέλο εντόπισε περιοχές που διέφεραν μεταξύ τους στα MRI και PET δεδομένα, επιβεβαιώνοντας πιθανώς τη συμπληρωματικότητά τους.

Οι μελλοντικές επεκτάσεις περιλαμβάνουν:
1) Διαφορετικό μονοπάτι εξαγωγής χαρακτηριστικών για κάθε τροπικότητα
2) Προχωρημένες τεχνικές προεπεξεργασίας MRI
3) Διερεύνηση early fusion μεθόδων
4) Χρήση GANs για αύξηση δεδομένων

Chapter 2

Introduction

2.1 Alzheimer's Disease

Alzheimer's disease (AD) is a condition of progressive neurodegeneration and the most common cause of dementia, estimated to account for between 60 and 80% of all dementia cases. Its main features consist of cognitive decline, memory impairment, and behavioral disturbances. In 2019, an estimated 5.8 million Americans aged 65 and older were affected, with projections indicating substantial growth in prevalence over the coming decades. Pathological changes arising in AD commence years, if not even decades, before the appearance of clinical symptoms, rendering early detection and intervention extremely challenging.

On the molecular level, AD pathology results from the accumulation of the protein fragment beta-amyloid outside neurons and the accumulation of an abnormal form of the protein tau inside neurons (Figure 2.1). These pathological changes threaten synaptic integrity, stopping neuronal communication and leading to widespread neurodegeneration and brain atrophy. With the progressive detoriration and death of the neurons of specific regions, the brain regions shrink, particularly the hippocampus and cortex, which are heavily involved in memory, decision-making, and language processing. Normal brain function is further compromised by the decreased ability of the brain to metabolize glucose, its main fuel. Although the exact mechanisms of AD pathogenesis remain under investigation, Scientists believe that in most cased the disease is caused by a combination of genetic, lifestyle and environmental factors.

As neuronal damage progresses, the effects on cognition become more noticeable, marking the shift from hidden pathological changes to recognizable symptoms. One of the earliest recognizable phases is Mild Cognitive Impairment (MCI), a stage where cognitive decline goes beyond normal aging but hasn't yet disrupted daily life in a major way. Individuals with MCI often experience memory lapses, difficulties with planning, or decision-making challenges, which may also be noticed by those around them. Although many cases of MCI do not develop AD, a significant proportion do, making this stage a crucial window for early diagnosis and intervention [2, 1].

2.2 Diagnosis of Alzheimer's Disease

Diagnosing AD is multifold as no single test has been developed that definitely proves the presence of the disease. Physicians would, in general, depend on several approaches: a complete medical history, physical and neurological examinations and cognitive tests, which serve to assess memory, problem-solving, and other intellectual abilities. Additionally, brain imaging techniques such as Magnetic Resonance Imaging (MRI) and Positron Emission Tomography (PET) scans are commonly used to assess structural and functional brain changes. Input from relatives or caregivers is often be essential for the documentation of the severity of cognitive and behavioral changes. Besides, one may carry out diagnosis tests such as blood tests that rule out other causes for dementia, including tumors, strokes, or vitamin deficiencies.

Advancements in neuroimaging technologies are revolutionizing the detection and diagnosis of AD with remarkable precision. Among these, Positron Emission Tomography (PET) stands out for its ability to detect beta-amyloid plaques, a hallmark of AD, by using specialized tracers that make these abnormal proteins visible. PET can also measure glucose metabolism in the brain, providing insights into regions affected by Alzheimer's. Magnetic Resonance Imaging (MRI) complements PET by identifying structural changes, such as atrophy or shrinkage in critical areas like the hippocampus and cortex, which are often affected in the early stages of the disease. Together, PET and MRI offer both structural and pathological insights. Integrating these technologies into routine clinical workflows enables earlier interventions and more personalized treatment strategies, paving the way for improved patient outcomes [1].

2.3 Deep Learning techniques for AD diagnosis

Deep learning has transformed the way AD is diagnosed, making it possible to automatically analyze complex neuroimaging data and detect subtle patterns that might be overlooked with traditional methods. Early models, like Deep Neural Networks (DNNs) and Recurrent Neural Networks (RNNs), introduced advanced feature extraction and the ability to recognize temporal patterns. However, these models often struggled with efficiency and scalability, limiting their practical use.

A major breakthrough came with Convolutional Neural Networks (CNNs), which excel at identifying spatial features in neuroimaging data. CNN-based studies have successfully pinpointed key biomarkers linked to AD, significantly improving diagnostic accuracy. Adding to this progress, Variational Autoencoders (VAEs) have been instrumental in simplifying complex data while preserving crucial information. Successive development has been also marked by the emergence of various multimodal frameworks, which integrate complementary strengths from different data sources like MRI, PET, and clinical records. Additionaly, the use of attention mechanisms within CNNs have enhanced model focus by allowing them to prioritize critical brain regions linked to AD pathology [13].

Despite these advancements, the lack of explainability in deep learning models remains a significant barrier to their widespread clinical adoption. Black-box models make it difficult for clinicians to interpret how decisions are made. To address this, researchers are exploring Explainable AI (XAI) techniques, such as saliency maps, Grad-CAM, and SHAP values, which provide visual and quantitative insights into the model's decision-making process [14].

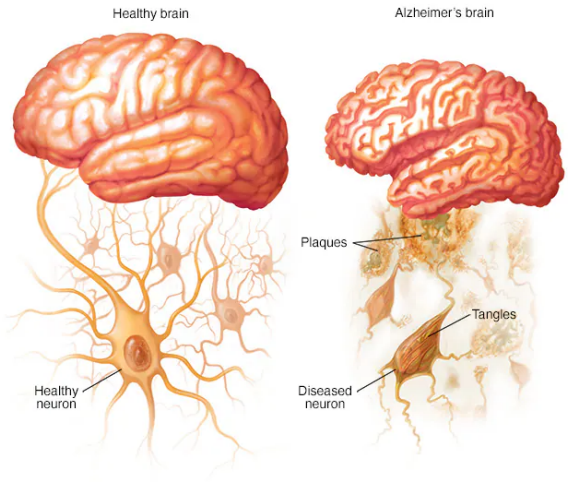


Figure 2.1. *Healthy brain and brain with Alzheimer's disease*
Source: [2].

2.4 Structure of the Thesis

From the above, it becomes clear that developing a reliable and effective diagnostic method for Alzheimer's disease is crucial. This study seeks to achieve this by utilizing a combination of MRI, PET and deep learning techniques. The structure of this thesis is outlined as follows:

- **Chapter 3**: The theoretical background necessary for understanding the various methods and concepts discussed in the study is developed.
- **Chapter 4**: Related Work. This chapter briefly summarizes previous research on Alzheimer's disease diagnosis and identifies areas where improvements are needed, providing the context for this study.
- **Chapter 5**: The dataset used in this study is introduced, along with the preprocessing techniques employed to prepare the data for analysis.
- **Chapter 6**: The proposed methodology used in this study is presented.
- **Chapter 7**: The results obtained are presented.
- **Chapter 8**: The conclusions derived from this research are discussed, along with possible future extensions of the current study.

Chapter 3

Theoretical Background

3.1 Machine Learning

Machine Learning (ML) is a field of Artificial Intelligence that enables systems to learn from data and make predictions based on it without being explicitly programmed for a specific task. The learning process begins with observing data to identify patterns, which helps improve system performance. Machine learning algorithms are widely used in various applications, such as analyzing medical images for disease prediction. ML methods are typically categorized into three main types:
1) **Supervised Learning**: In this category, training data consists of inputs paired with corresponding output values. Supervised learning algorithms are further classified into:

- **Classification**: Here, the output value takes discrete values. During training, the algorithm searches for patterns in the input data that strongly correlate with desired outputs. Once trained, the algorithm predicts the output value for unseen input features. In the context of Alzheimer's disease, examples of classification tasks include:
  - Categorizing MRI or PET images as *Normal*, *MCI*, or *AD*.
  - Determining whether a tumor is benign or malignant.
  - Identifying the presence or absence of beta-amyloid plaques in PET images.
- **Regression**: Here, the output value takes continuous values. Examples include predicting the degree of brain atrophy (e.g., hippocampal volume loss) from MRI images.

2) **Unsupervised Learning**: In this category, the training data consists only of inputs without corresponding output values. The primary goal of unsupervised learning algorithms is to identify groups of features that follow similar patterns.
3) **Reinforcement Learning**: This involves an agent that interacts continuously with the environment, making decisions to maximize rewards in a specific situation. Unlike supervised learning, where the model trains using known output values, reinforcement learning requires the agent to decide what actions to take based on its experience [15].

3.2 Neural Networks

Neural networks, a subfield of Machine Learning and Artificial Intelligence, provide an algorithmic framework for addressing computational problems by drawing inspiration from biological neural networks. At their core, both biological and artificial systems rely on interconnected units that collaborate to transmit and process information. Through their ability to learn from examples and identify complex, non-linear patterns, neural networks have become a foundational component of Artificial Intelligence.

3.2.1 Biological Neural Networks

The human brain is among the most complex and remarkable structures in nature, with its functionality serving as a continuous source of inspiration for advancements in artificial intelligence. Biological neural networks are intricate systems of interconnected neurons, the fundamental units of the nervous system. A typical neuron comprises three primary components: dendrites, the soma (cell body), and the axon. Dendrites receive input signals from other neurons, which are processed within the soma. When the input surpasses a certain threshold, the neuron generates an action potential which is a brief electrical impulse that travels along the axon. At the synapse, this electrical signal is transformed into chemical signals through the release of neurotransmitters, which influence neighboring neurons.

Synaptic weights govern the strength of these interactions, determining whether subsequent neurons will activate. The brain's capacity for parallel processing, with billions of neurons working together simultaneously, serves as a key inspiration for artificial networks. By emulating these mechanisms, artificial intelligence systems aim to replicate aspects of the brain's efficiency and adaptability [16].

3.2.2 Artificial Neural Networks

Artificial Neural Networks (ANNs) are computational models designed to emulate key principles of biological neural networks. ANNs are composed of layers of interconnected nodes structured into input, hidden, and output layers. Each node processes inputs by calculating a weighted sum, applying an activation function, and transmitting the result to the next layer. This layered design enables ANNs to perform hierarchical processing, similar to the way information flows through biological networks.

Figure 3.1 illustrates the structural and functional similarities between biological and artificial neurons by showing how both systems transmit and process information. While biological neurons rely on chemical and electrical signals to communicate, artificial neurons use mathematical operations to simulate this process. Both systems rely on weights to adjust the strength of connections and thresholds to determine activation. During training, ANNs adjust their weights using algorithms like backpropagation and gradient descent, reflecting the adap-

tive processes observed in biological systems. The adaptability makes ANNs a powerful tool for solving complex problems in various domains, including image recognition, natural language processing, and medical diagnostics. Further details on these methods will be discussed in later sections [17].

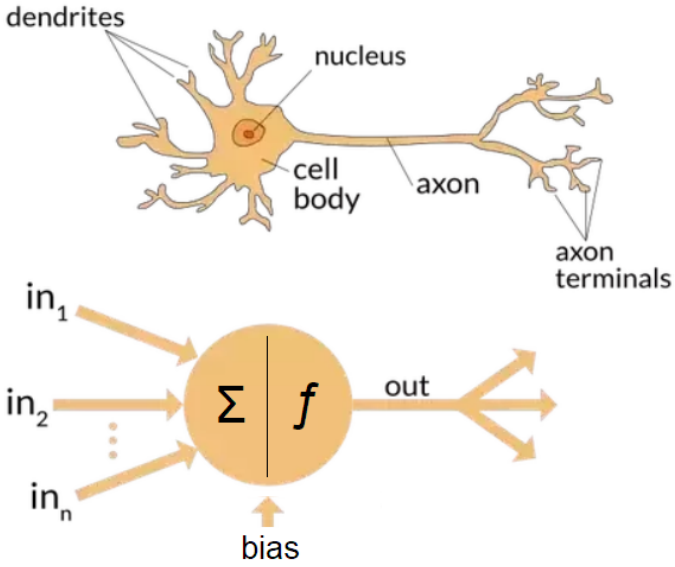


Figure 3.1. *Comparison of a biological neuron and an artificial neuron.*
Source: [17].

The foundational component of Artificial Neural Networks (ANNs) is the Perceptron, as illustrated in the lower part of Figure 3.1. It processes information by receiving a set of inputs (x_i), each associated with a weight (w_i) that represents the importance of the input. These weighted inputs are summed, combined with a bias term (b), and passed through an activation function (f) that determines the output of the perceptron. Depending on the result, the perceptron is either activated or remains inactive, simulating the firing behavior of biological neurons. Mathematically, this is expressed as:

$$y = f\left(\sum_{i=1}^{n} x_i w_i + b\right) \tag{3.1}$$

While the perceptron effectively solves linearly separable problems, its inability to capture non-linear relationships necessitates more advanced architectures, such as Multi-Layer Perceptron.

3.2.3 Multi-Layer Perceptron

A Multi-Layer Perceptron (MLP) classifier extends the capabilities of the Perceptron by learning non-linear functions. It comprises multiple layers of neurons (nodes) connected by weighted edges, including an input layer, one or more hidden layers, and an output layer. The input layer receives raw data, while the hidden layers transform it using activation functions to capture non-linear relationships. The output layer produces the final prediction or classification. An example of an MLP is illustrated in the image 3.2:

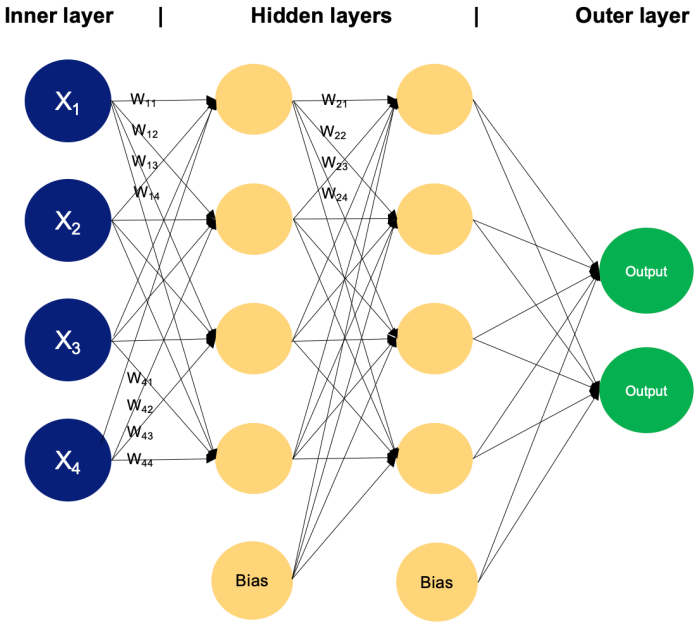


Figure 3.2. *Architecture of Multilayer Perceptron*
Source: [18].

3.2.4 Activation Functions

Activation functions are essential components of neural networks, enabling them to capture complex, non-linear relationships in data. Two of the most commonly used activation functions are the sigmoid and ReLU (Rectified Linear Unit) functions. The sigmoid function maps any real input to a range between 0 and 1, making it particularly useful for problems involving probabilities. It is mathematically expressed as:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{3.2}$$

On the other hand, the ReLU function introduces non-linearity by outputting zero for negative inputs and the input value itself for positive inputs. It is defined as:

$$f(x) = \max(0, x) \tag{3.3}$$

Both activation functions are illustrated in Figure 3.3. While the sigmoid function is effective in specific contexts, such as binary classification, ReLU is often preferred for deep networks due to its simplicity and its ability to mitigate the vanishing gradient problem [19], which is discussed in 3.2.6.
The Softmax function is another widely used activation function, particularly in the output layer of neural networks for multi-class classification problems. It converts raw scores (logits) from the network into probabilities that sum to 1, making it ideal for tasks requiring a probabilistic interpretation of the output. Mathematically, for an input vector $z = [z_1, z_2, \ldots, z_n]$, the softmax function is defined as:

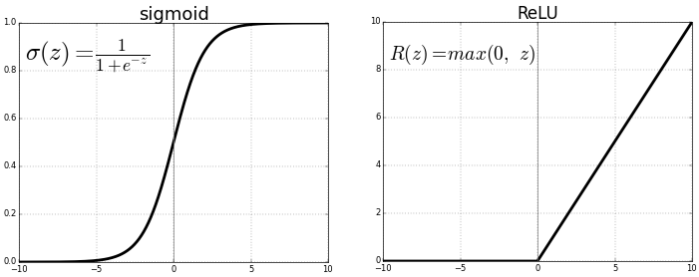


Figure 3.3. *Activation fuctions: sigmoid and ReLU*
Source: [19].

$$\sigma(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}} \tag{3.4}$$

This ensures that each output value is normalized, representing the likelihood of belonging to a particular class [20]. The softmax function is shown in 3.4:

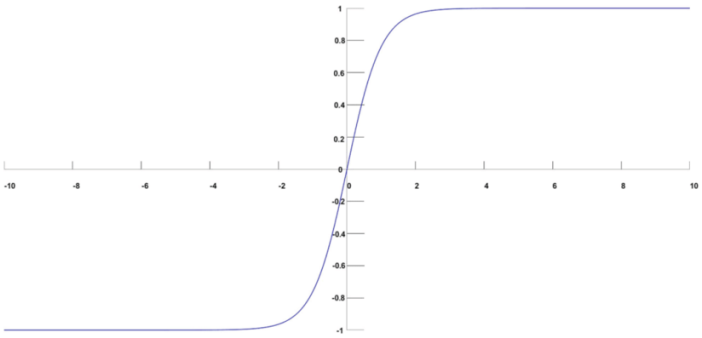


Figure 3.4. *Activation fuctions: Softmax*
Source: [20].

The *tanh* (hyperbolic tangent) activation function is the last of the main activation functions commonly used in neural networks. It transforms input values into a range of -1 to 1, making it zero-centered. This property allows *tanh* to better handle both positive and negative activations, helping to reduce biases in weight updates. The function is computed as:

$$\tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}, \tag{3.5}$$

While *tanh* can still suffer from the vanishing gradient problem, it remains useful in scenarios where normalized outputs are beneficial. Its smooth gradient also allows for efficient backpropagation, making it a popular choice for intermediate layers of neural networks [21].The tanh function is shown in 3.5:

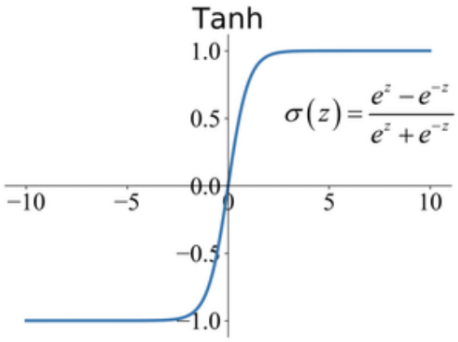


Figure 3.5. *Activation fuctions: Tanh*
Source: [21].

3.2.5 Loss functions

A loss function is a fundamental component of machine learning and deep learning models. It measures the error or difference between the predicted output of the model and the actual target values, serving as a guide for the optimization process. By minimizing the loss function, the model learns to improve its predictions over time. Loss functions can be categorized into various types, with specific ones tailored for regression, classification, and other tasks. In this section, we are going to analyze three of the most commonly used loss functions: Mean Squared Error, Cross-Entropy Loss, and Binary Cross-Entropy Loss, exploring their mathematical formulations and applications.

The Mean Squared Error (MSE) is a widely used loss function in regression tasks. It measures the average squared difference between the true values y_i and the predicted values $\hat{y}_i$. The formula is:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{3.6}$$

The squared term penalizes larger errors more heavily than smaller ones, which makes MSE sensitive to outliers in the data. MSE is smooth and differentiable, allowing for efficient optimization during gradient descent. However, its sensitivity to outliers may lead to suboptimal performance if the data contains significant noise or anomalies.

The Cross-Entropy (CE) Loss is primarily used for classification tasks. The formula for CE is derived from that of Kullback-Leibler divergence (KL divergence), which measures the difference between two probability distributions. It quantifies the difference between the true class distribution and the predicted probability distribution. For multi-class problems, the loss is computed as:

$$\text{Cross-Entropy Loss} = -\frac{1}{n}\sum_{i=1}^{n}\sum_{c=1}^{C} y_{i,c}\log(\hat{y}_{i,c}) \tag{3.7}$$

where C is the total number of classes, and $y_{i,c}$ and $\hat{y}_{i,c}$ represent the actual and predicted probabilities for class c.

For binary classification tasks, the Binary Cross-Entropy (BCE) Loss is commonly used. It is derived as a special case of the CE Loss when the number of classes C=2. Substituting this into the CE formula results in:

$$\text{Binary Cross-Entropy Loss} = -\frac{1}{n}\sum_{i=1}^{n}\left[y_i\log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)\right] \tag{3.8}$$

CE is particularly powerful when paired with softmax or sigmoid activation functions in the final layer of neural networks [22].

3.2.6 Neural Network training

Every Machine Learning model needs a learning algorithm to fit the data that wants to extract patterns from. As we discussed in section 3.2.5, for this purpose in every neural network that inputs a data vector **x** with ground truth **t** and outputs a vector **y**, a loss function is defined to quantify the performance of the model in the training data (equation: 3.9).

$$L(w) = \sum_{i=1}^{N}\text{loss}(y_i, t_i) \tag{3.9}$$

The process begins with computing the loss or error, which quantifies how well the model's predictions align with the ground truth. This computed error serves as the starting point for the backpropagation process, which systematically adjusts the model's parameters to minimize the loss. During backpropagation, the chain rule is applied recursively to calculate the gradients of the loss function with respect to the model's weights, denoted as $\frac{\partial L}{\partial \mathbf{w}}$. These gradients play a crucial role in guiding the optimization process.

However, in very deep neural networks, a challenge known as the vanishing gradient problem can arise. As gradients are propagated backward through many layers, they can diminish exponentially, especially when activation functions like sigmoid or tanh are used. This happens because the derivatives of these functions often result in values less than 1, causing the gradients to shrink layer by layer. As a result, earlier layers in the network receive very small updates, hindering effective learning and slowing down or even halting convergence.

Once the gradients are calculated, the weight vector of the model is updated by moving in the

opposite direction of the gradient vector, scaled by a factor η which determines the step size for each update. This update rule is applied iteratively to minimize the loss, forming the so called gradient descent algorithm, which is aiming to converge in the global minima. The weight update rule can be expressed as:

$$\mathbf{w}^{t+1} = \mathbf{w}^{t} - \eta \frac{\partial \mathcal{L}(\mathbf{w})}{\partial \mathbf{w}} \tag{3.10}$$

Although low efficiency, when optimizing on top of large datasets, and convergence in local minima, based on the initialization of the network's weights, have led to the following variations of the gradient descent algorithm, depending on the application:

- **Batch Gradient Descent**: Computes the gradient using the entire training dataset before updating the weights. It provides stable convergence but can be slow for large datasets.
- **Stochastic Gradient Descent (SGD)**: Updates the weights for each training sample individually. It is faster and can converge quickly, but the updates can be noisy.
- **Mini-Batch Gradient Descent**: Divides the dataset into small batches and updates the weights after processing each batch. It balances speed and stability, making it efficient for large datasets [23].

3.2.7 The problem of overfitting

Overfitting in neural networks occurs when a model learns the training data, including its noise and outliers, too well, resulting in poor generalization to new, unseen data. This leads to high accuracy on training data but poor performance on validation or test sets. Several techniques have been developed to mitigate overfitting in neural networks:

- **Dropout Regularization**: This technique involves randomly deactivating neurons during training. A more detailed analysis of this method is provided in 3.3.5.
- **Early Stopping:** This technique monitors the model's performance on a validation set during training and stops the training process when performance starts to degrade, indicating the onset of overfitting. By halting training at the optimal point, early stopping prevents the model from learning noise in the training data.
- **Weight Regularization (L1 and L2 Regularization):** : These methods add a penalty to the loss function based on the magnitude of the model's weights. L1 regularization encourages sparsity in the model weights, while L2 regularization discourages large weights, both aiming to simplify the model and reduce overfitting.
- **Data Augmentation**: By artificially expanding the size of the training dataset through transformations such as rotations, translations, and scaling, data augmentation exposes the model to a wider variety of scenarios, helping it to generalize better to new data [24].

3.2.8 Types of Neural Networks

Neural networks come in various architectures, each designed to handle specific types of data and tasks effectively. The simplest type is the Feedforward Neural Network (FNN), where data flows in a single direction from the input layer to the output layer, passing through one or more hidden layers. FNNs are widely used for general-purpose tasks, such as regression and basic classification problems. Another key type is the Recurrent Neural Network (RNN), which is designed for sequential data, such as time series or natural language processing. RNNs maintain a hidden state that captures information from previous inputs, allowing them to process temporal patterns. Variants like Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRUs) address the vanishing gradient problem and improve the learning of long-term dependencies. For processing spatial data, such as images, Convolutional Neural Networks (CNNs) are widely used. CNNs use convolutional layers to extract local patterns, such as edges, textures, and shapes, making them highly effective for image recognition, object detection, and similar tasks. Lastly, Generative Adversarial Networks (GANs) are a special class of networks used for generating data, such as images or videos. GANs consist of two components: a generator and a discriminator, which compete against each other to improve the quality of the generated data [25].

In the next section, we will focus on CNNs, analyzing their structure, functionality, and applications in greater detail. CNNs are the main focus of this thesis due to their significance and versatility in solving computer vision problems.

3.3 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are a specialized class of neural networks designed to process data with a grid-like structure, such as images and videos. Unlike traditional neural networks that treat input data as a one-dimensional vector, CNNs preserve spatial relationships and hierarchical features. They achieve this through a layer-based architecture, as illustrated in Figure 3.6, which depicts the network components and their roles in processing input data.

A typical CNN comprises convolutional, activation, pooling, and fully connected layers, each progressively extracting and analyzing features. Convolutional layers detect local patterns, activation functions introduce non-linearity, pooling layers reduce spatial dimensions, and fully connected layers aggregate features for final predictions [26].

In this chapter, we delve deeper into the fundamental components and workings of CNNs, describing how their structure and design have made them indispensable in modern computer vision tasks.

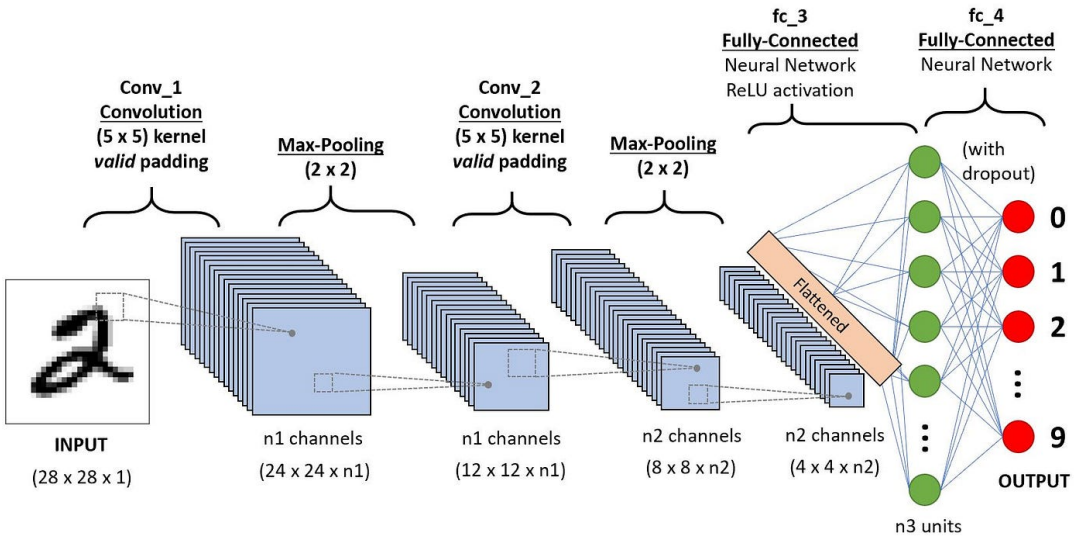


Figure 3.6. *Architecture of a CNN*
Source: [26].

3.3.1 Convolutional Layer

The convolutional layer is a fundamental component of CNNs, specifically designed to process and analyze spatially organized data, such as images. It operates by applying a set of filters or kernels over the input data, performing an operation known as convolution. This mathematical process can be expressed as:

$$Y(i,j) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} X(i+m, j+n) \cdot K(m,n) + b \tag{3.11}$$

where:

- $Y(i,j)$ is the output at position (i,j) in the feature map,
- $X(i+m,j+n)$ is the input value at position $(i+m,j+n)$,
- $K(m,n)$ is the value of the kernel at position (m,n),
- M and N are the dimensions of the kernel,
- b is the bias term.

The convolution operation involves sliding each filter over the input data and calculating the weighted sum of the input values within the receptive field defined by the filter. This process generates an output feature map that emphasizes critical spatial patterns and features, such as edges, textures, and shapes. The nature of receptive field values depends on the input data: for single-channel inputs such as grayscale images, the values are intensity levels, and for multi-channel inputs such as RGB images, they contain color information in red, green, and blue channels. For instance, grayscale images, like those from medical imaging modalities such as CT or MRI scans, consist of single-channel intensity values, whereas natural images commonly include RGB data [27]. An example of a convolution operation on multi-channel data using three distinct kernels is depicted in 3.7:

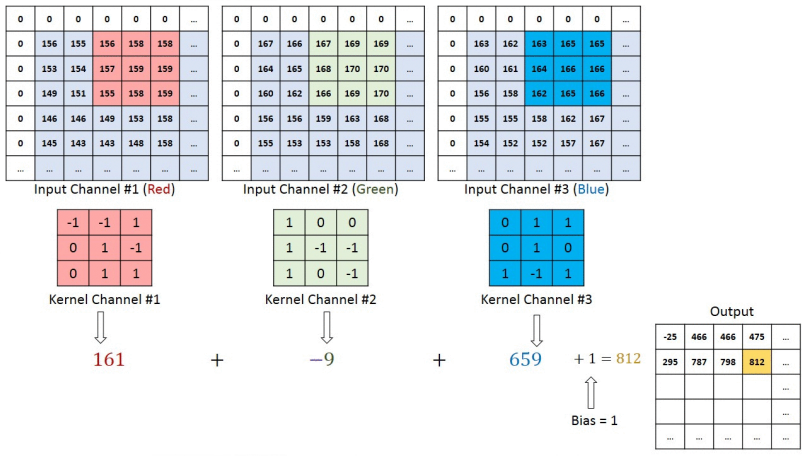


Figure 3.7. *Convolution with multi-chanel data*
Source: [26].

The 3D convolutional layer extends 2D convolutions by operating on three-dimensional data, such as video sequences or volumetric medical images (e.g., MRI scans). While 2D convolutions slide a kernel across height and width, 3D convolutions add a depth dimension, enabling the extraction of volumetric features and spatial-depth relationships. As illustrated in Figure 3.8, this approach is applied to both single-channel and multi-channel data, making 3D convolutions particularly effective for detecting complex anomalies in medical images, such as tumors or lesions spanning multiple slices.

In this study, we focus on employing 3D convolutional layers to analyze 3D MRI and PET scans, demonstrating their superior effectiveness in capturing intricate spatial features for diagnostic and analytical purposes [28].

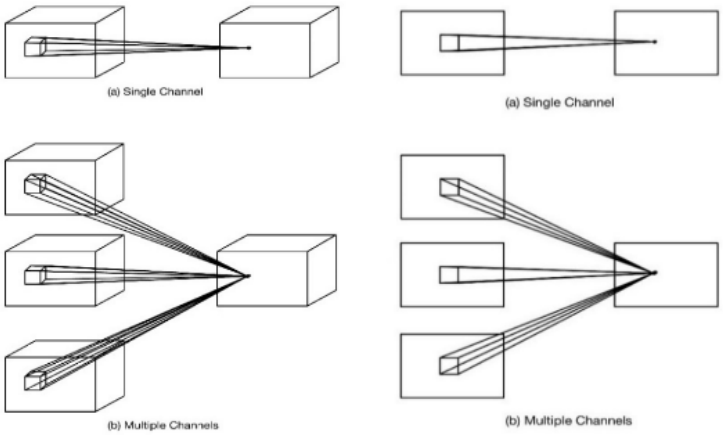


Figure 3.8. *2D and 3D convolution*
Source: [28].

3.3.2 Pooling Layer

The pooling layer plays a crucial role in CNNs by reducing the spatial dimensions of convolved features, which helps lower computational complexity and improve efficiency. Beyond just dimensionality reduction, pooling also preserves dominant features that remain consistent regardless of rotation or position, making it essential for effective model training. Typically, pooling layers follow convolutional layers, progressively condensing feature maps.

There are two main types of pooling: Max Pooling and Average Pooling. Max Pooling selects the highest value within a given region, emphasizing the most prominent features, while Average Pooling calculates the mean of all values in that region, resulting in a smoother representation (Figure 3.9).

In addition to dimensionality reduction, Max Pooling serves as a noise suppressor by filtering out weak activations, effectively enhancing important features while reducing noise. Average Pooling, on the other hand, mainly focuses on reducing spatial dimensions without significantly improving feature sharpness. Because of its ability to highlight crucial details and remove noise, Max Pooling is often preferred for feature extraction in deep learning applications [26].

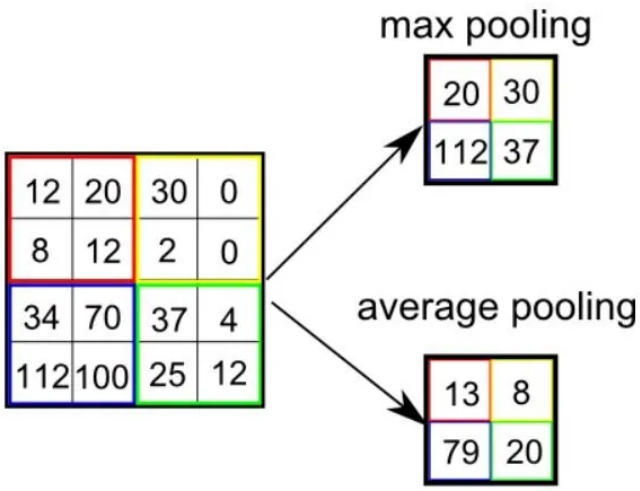


Figure 3.9. *Types of pooling*
Source: [26].

3.3.3 Fully Connected Layer

The fully connected layer is a crucial part of Convolutional Neural Networks (CNNs), responsible for combining the high-level features extracted by earlier layers to make final predictions. Unlike convolutional and pooling layers, which focus on local patterns, the fully connected layer links every neuron from one layer to every neuron in the next. This connectivity allows the network to integrate all extracted features and interpret them as a whole. As shown in Figure 3.6, the fully connected layer flattens the feature maps into a single vector, converting spatial information into a format suitable for classification. The layer then applies weights, biases, and activation functions to capture complex, non-linear relationships. Typically found at the end of a CNN, fully connected layers aggregate learned features and output probabilities or class labels [26].
Mathematically, the operation of a fully connected layer can be expressed as:

$$\mathbf{y} = f(\mathbf{W} \cdot \mathbf{x} + \mathbf{b}) \tag{3.12}$$

In this equation, the input vector $\mathbf{x}$ is multiplied by the weight matrix $\mathbf{W}$, added to the bias vector $\mathbf{b}$, and then passed through the activation function f to produce the output vector $\mathbf{y}$.

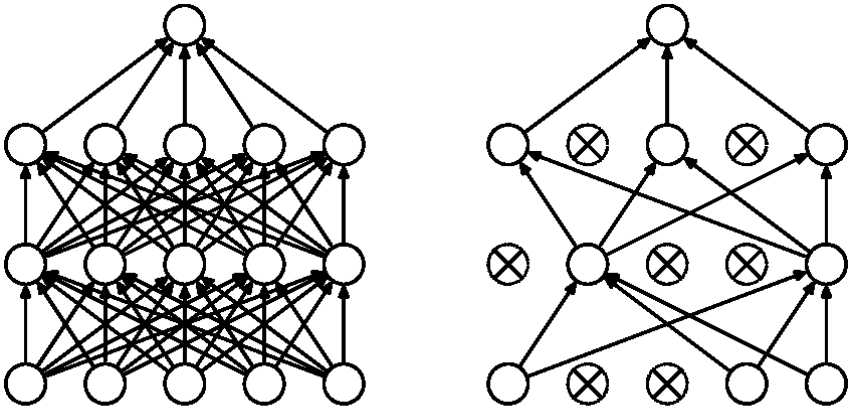
3.3.4 Batch normalization

Batch Normalization is a widely used technique in CNNs that improves training stability and accelerates convergence by normalizing the inputs of each layer. This method standardizes the inputs to a layer by adjusting their mean and variance during training, ensuring that the data remains on a consistent scale. By reducing internal covariate shift—the phenomenon where layer inputs change distribution during training—batch normalization helps the network learn more effectively. Additionally, it reduces the sensitivity of the model to the initial weights and allows for higher learning rates, improving overall performance. Beyond stabilizing train-

ing, it can also act as a regularizer, reducing overfitting in some cases. Batch normalization is often applied between the linear operation (e.g., convolution) and the activation function [29].

3.3.5 Dropout

Dropout is a regularization technique used in CNNs to prevent overfitting and improve the generalization of the model. During training, dropout randomly "drops out" a subset of neurons by setting their outputs to zero, effectively removing them from the network for that iteration. This forces the network to learn more robust and distributed representations, as no single neuron becomes overly reliant on its neighbors. By introducing this stochastic behavior, dropout reduces the risk of overfitting to the training data and enhances the network's ability to generalize to unseen data. An illustration of this process is shown in Figure 3.10. Dropout is typically applied in fully connected layers but can also be used in convolutional layers depending on the architecture. During testing, all neurons are active, and their outputs are scaled based on the dropout rate to maintain consistency with the training phase. This simple yet powerful technique has become a standard practice in modern deep learning models [30].



(a) Standard Neural Net (b) After applying dropout.

Figure 3.10. *Dropout layer*
Source: [30].

3.3.6 Limitations of CNNs

CNNs are exceptional at handling structured data, but they often face challenges with complex and varied tasks. Their fixed architecture and uniform feature extraction make it difficult to adapt to diverse data and specialized tasks. For instance, in medical imaging, the data's clustered structures, anatomical differences, and subtle pathological features highlight these limitations. Additionally, CNNs activate all layers for every input, leading to unnecessary computations and inefficiencies. To address these issues, adaptive architectures like the Mixture of Experts (MoE) framework have been developed.
In the next section, we will explore the Mixture of Experts framework as a solution to these challenges.

3.4 Mixture of Experts

The Mixture-of-Experts (MoE) model architecture is a groundbreaking approach in machine learning, enabling the efficient scaling of neural networks to handle increasingly complex tasks. Introduced by Jacobs et al. in 1991 [5], MoE follows the divide-and-conquer principle by partitioning the problem space among multiple specialized "experts." Similar to a team of specialists working together to solve a complex problem, each expert in MoE brings unique skills to address specific tasks, allowing the model to achieve high efficiency and performance. Its modular design also provides flexibility and interpretability, enabling independent analysis and adjustment of each expert's contribution. These advantages have established MoE as a pivotal solution for a wide range of applications, including natural language processing, computer vision, and speech recognition.

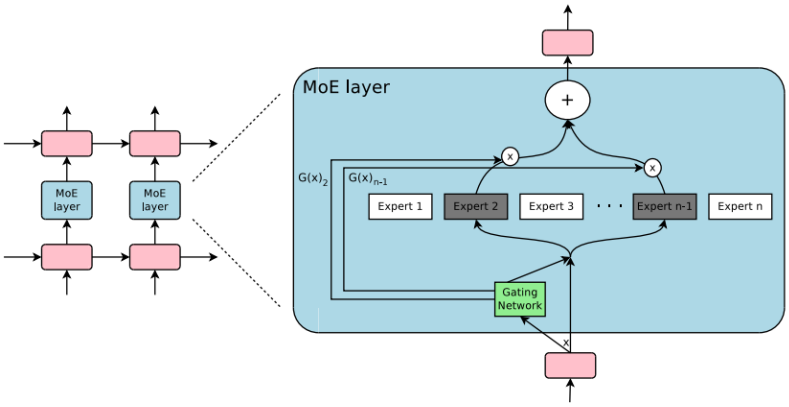


Figure 3.11. *The architecture of an MoE layer*
Source: [31].

Central to the MoE framework is its dynamic approach to leveraging specialized sub-models, or "experts", to process input data. As illustrated in Figure 3.11, the framework includes a mechanism that determines the most relevant experts for a given input [32].
Building on this foundation, the concept of Sparse-MoE was introduced by Shazeer et al. [31]. Sparse-MoE extends the original MoE framework by introducing sparsity in the activation of experts, where only a small subset of experts is selected for each input. This innovation significantly reduces computational costs, enabling the deployment of models with billions of parameters while maintaining efficiency and scalability.
The architecture of the MoE model consists of three key components:

- **Experts**: Specialized sub-models, each responsible for a specific part of the problem space.
- **Gating Network**: As described earlier, it determines the subset of experts activated for a given input and combines their outputs efficiently through weighted summation.
- **Sparse Activation**: Ensures computational efficiency by activating only a subset of experts for each input.

3.4.1 Gating Network

The gating network, often referred to as the router, is a crucial component of the MoE architecture. Its primary role is to analyze the input data and determine which experts are most suited to handle the task at hand. This process is mathematically represented as:

$$y = \sum_{i=1}^{n} G(x)_i E_i(x) \tag{3.13}$$

Here, $G(x)_i$ is the gating weight assigned to the i-th expert based on the characteristics of the input x, and $E_i(x)$ is the output of the i-th expert. The gating network ensures that the most relevant experts are selected by dynamically assigning weights.
The Softmax Gating mechanism, introduced by Jacobs et al. in 1994 [6], computes gating weights by multiplying the input x with a trainable weight matrix W_g and normalizing the scores using the softmax function:

$$G_\sigma(x) = \text{Softmax}(x \cdot W_g) \tag{3.14}$$

The resulting $G_\sigma(x)$ represents the importance of each expert, ensuring non-negative weights that sum to 1. While simple and effective, this dense gating activates all experts for every input, which can be computationally expensive. Sparse gating mechanisms address this by selecting only the most relevant experts for each input.

3.4.2 Sparse Activation

The concept of sparse activation, as outlined by Shazeer et al., plays a critical role in improving computational efficiency without compromising model capacity. Sparse activation ensures that only a subset of the available experts is used for processing each input, with the weights of the remaining experts set to 0. This is mathematically represented as:

$$G(x) = \text{Softmax}(\text{KeepTopK}(H(x), k)), \tag{3.15}$$

where $G(x)$ is the gating vector that determines the weights of selected experts, and $H(x)$ is a vector of raw gating scores defined as:

$$H(x)_i = (x \cdot W_g)_i + \text{StandardNormal}() \cdot \text{Softplus}((x \cdot W_{\text{noise}})_i). \tag{3.16}$$

96The operation KeepTopK ensures sparsity by retaining only the top k scores in $H(x)$, defined as:

$$\text{KeepTopK}(v, k)_i = \begin{cases} v_i & \text{if } v_i \text{ is in the top } k \text{ elements of } v, \\ -\infty & \text{otherwise.} \end{cases} \tag{3.17}$$

In addition to activating only a small subset of experts and enabling the model to scale to billions of parameters while keeping computational costs low, sparse gating offers another key advantage: the introduction of noise through StandardNormal() and load-balancing losses pro-

motes a more even distribution of tasks among experts.

3.4.3 Training of MoE

Training MoE models involves addressing the challenge of ensuring balanced expert utilization. Without appropriate mechanisms, the gating network can disproportionately favor a small subset of experts, leaving others underutilized and reducing overall model performance. To mitigate this, the model incorporates regularization techniques, such as load-balancing losses, to distribute tasks more evenly across experts. One such technique is the **importance loss**, which penalizes the coefficient of variation (CV) of the gating probabilities, encouraging broader expert participation. It is mathematically defined as:

$$L_{\text{importance}}(X) = w_{\text{importance}} \cdot \text{CV}(\text{Importance}(X))^2, \tag{3.18}$$

where Importance(X) represents the sum of gating weights across all inputs X. Complementing this, the **load-balancing loss** minimizes the CV of the load distribution to ensure no single expert is overloaded:

$$L_{\text{load}}(X) = w_{\text{load}} \cdot \text{CV}(\text{Load}(X))^2. \tag{3.19}$$

Here, $\text{Load}(X)_i$ represents the total probability of expert i being assigned tasks across X. Together, these loss terms promote equitable task allocation.
These regularization losses are integrated with the primary loss function (e.g., cross-entropy) to form the overall training objective:

$$L_{\text{total}} = L_{\text{loss}} + a * (L_{\text{importance}} + L_{\text{load}}), \tag{3.20}$$

where L_{loss} is the loss for the primary task (e.g. cross-entropy loss) a and is a hyperparameter denoting the importance we place on these two loss functions [31].

3.5 Feature Fusion Techniques

Multimodal data, such as MRI and PET scans, provides complementary perspectives that, when effectively combined, can significantly improve the accuracy and robustness of machine learning models for complex tasks like disease diagnosis. Effective integration of these modalities is crucial to fully leverage their strengths. While simple feature concatenation is a common approach for combining multimodal data, advanced techniques like attention mechanisms and Gated Multimodal Units provide more powerful alternatives. Attention mechanisms dynamically prioritize the most relevant features from each modality [33], while GMUs use gating mechanisms to balance information sharing and separation across modalities [34]. In this chapter, these techniques will be analyzed in detail.

3.5.1 Gated Multimodal Unit

The Gated Multimodal Unit (GMU) is a mechanism designed to adaptively integrate features from multiple modalities. By dynamically balancing the contribution of each modality, the GMU allows the model to effectively capture both shared and modality-specific information while filtering irrelevant or noisy features. For two modalities (f_1 and f_2), the GMU operates as follows:

$$h_1 = \tanh(W_1 f_1 + b_1), \tag{3.21}$$

$$h_2 = \tanh(W_2 f_2 + b_2), \tag{3.22}$$

$$z = \sigma(W_z[f_1; f_2] + b_z), \tag{3.23}$$

$$h = z \odot h_1 + (1 - z) \odot h_2, \tag{3.24}$$

The parameters of the GMU are collectively represented as:

$$\Theta = \{W_1, W_2, W_z\}, \tag{3.25}$$

where:

- f_1 and f_2 are the input features from the two modalities.
- W_1, W_2, W_z are learnable weight matrices for the respective modalities and gating mechanism, forming the parameter set Θ.
- b_1, b_2, b_z are the corresponding bias terms.
- z is the gating vector, computed via the sigmoid function (σ), which determines the relative importance of each modality.
- h_1 and h_2 are the transformed representations of the input features, passed through a tanh activation function.
- h is the final fused representation, computed as a weighted combination of h_1 and h_2, with the gating vector z controlling the contribution of each modality.

The parameter set Θ is optimized during training to ensure the gating mechanism dynamically adjusts to the given input. The GMU framework is inherently scalable and can be extended to handle more than two modalities. For n modalities, the generalized formulation is:

$$h = \sum_{i=1}^{n} z_i \odot \tanh(W_i f_i + b_i), \tag{3.26}$$

where z_i represents the gating weight for the i-th modality, and the weights satisfy $\sum_{i=1}^{n} z_i = 1$. This ensures that the contributions of all modalities are dynamically balanced, with the most relevant modalities receiving higher weights.

Figure 3.12 illustrates the architecture of the GMU, including its ability to handle two modalities or scale to multiple modalities [34].

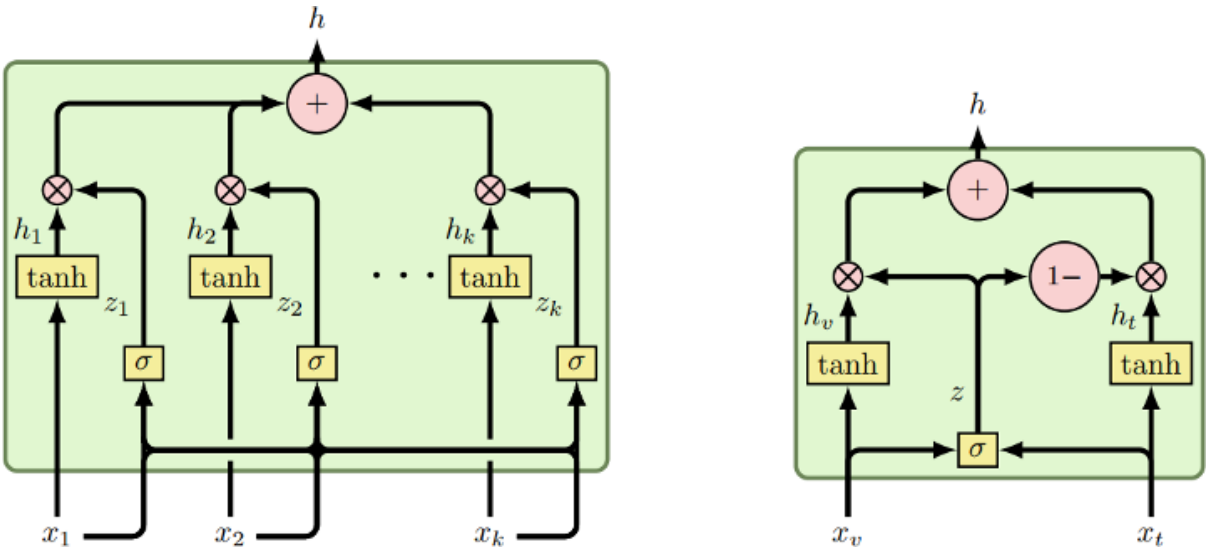


Figure 3.12. *Illustration of the GMU framework for two modalities (right) and multiple modalities (left)*

Source: [34].

3.5.2 Attention mechanism

Whereas the GMU excels at dynamically balancing contributions from multiple modalities, the attention mechanism, introduced by Vaswani et al [33], enables models to focus on the most relevant features within each modality. Unlike traditional approaches that treat all input features equally, attention dynamically assigns importance scores, allowing the model to capture complex dependencies and relationships within the data. Initially proposed as a core component of the Transformer architecture, attention has since become a foundational technique in various domains, including natural language processing, computer vision, and multimodal learning.

Generalized attention provides a versatile mechanism for modeling relationships between elements in an input sequence. At its core, as illustrated in Figure 3.13, the attention mechanism operates using a query-key-value paradigm, where each input element is mapped to three learned representations: queries (Q), keys (K), and values (V). Keys serve as labels for distinguishing features, while queries evaluate all available keys to identify the most relevant ones. The model then uses this evaluation to associate the corresponding values with the input. Specifically, an attention layer processes queries and keys of dimension d_k, and values of dimension d_v. The attention scores, which determine the relevance between queries and keys, are calculated as the scaled dot product, mathematically expressed as:

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V, \tag{3.27}$$

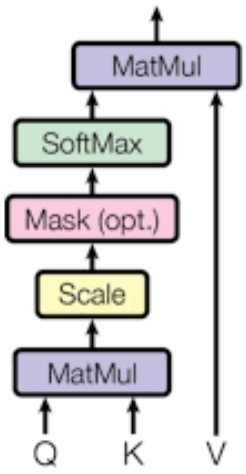


Figure 3.13. *Illustration of the attention mechanism*
Source: [33].

For self-attention mechanisms, the queries, keys, and values are derived from the same input, allowing the model to learn interactions within a single modality and identify which parts of the input are most relevant for making predictions. In the context of multimodal learning with two modalities, self-attention refers to computations performed independently within each

modality. Specifically, the latent feature representations generated by prior layers for each modality act as the queries, keys, and values. For two modalities, the self-attention module performs the following operations:

$$\text{self-attention}(M_1 \rightarrow M_1) \tag{3.28}$$

$$\text{self-attention}(M_2 \rightarrow M_2) \tag{3.29}$$

Here, M_1 and M_2 represent the feature matrices of the two modalities, but it can be generalized to more than two modalities.

In a cross-modal attention mechanism, the queries, keys, and values originate from different modalities. Instead of computing attention within a single modality, as in self-attention, cross-modal attention allows one modality to attend to another. Given two modalities, M_1 and M_2, the cross-modal attention module performs the following operations:

$$\text{cross-modal attention}(M_1 \rightarrow M_2) = \text{softmax}\left(\frac{Q_{M_1} K_{M_2}^T}{\sqrt{d_k}}\right) V_{M_2} \tag{3.30}$$

$$\text{cross-modal attention}(M_2 \rightarrow M_1) = \text{softmax}\left(\frac{Q_{M_2} K_{M_1}^T}{\sqrt{d_k}}\right) V_{M_1} \tag{3.31}$$

where Q, K, and V represent the queries, keys, and values, respectively, with one modality acting as the query source and the other as the key-value source.

Yu et al. [4] introduced a novel technique to improve the quality of the learned attention. A key challenge in self-attention mechanisms is learning an accurate attention map A. Traditional scaled dot-product attention does not explicitly capture the varying importance of individual features, potentially introducing noise i5167 3204 5514 1861nto the attention computation. To address this, Yu et al. proposed a gated attention mechanism inspired by bilinear pooling:

$$M = \sigma\left(FC_g\left(FC_g^q(Q) \odot FC_g^k(K)\right)\right), \tag{3.32}$$

where FC_g^q and FC_g^k are fully connected layers that project Q and K into a shared space, and FC_g further refines the element-wise product ($\odot$) into a feature gating mask. The sigmoid activation function $\sigma(\cdot)$ ensures that the values in M are in the range $(0,1)$, effectively filtering out less relevant features.

Additionally, the feature-wise attention mask M is defined as:

$$M \in \mathbb{R}^{m \times 2} \tag{3.33}$$

which corresponds to the two masks $M_q \in \mathbb{R}^m$ and $M_k \in \mathbb{R}^m$, associated with the features Q and V, respectively.

Next, the masks M and K are tiled to obtain $\tilde{M}_q, \tilde{M}_k \in \mathbb{R}^{m \times d}$ and are subsequently used for computing the attention map as follows:

$$A^g = \text{softmax}\left(\frac{(Q \odot \tilde{M}_Q)(K \odot \tilde{M}_K)^T}{\sqrt{d}}\right). \tag{3.34}$$

Finally, the attended feature representation is obtained as:

$$F = A^g V. \tag{3.35}$$

This approach ensures that the importance of individual features is explicitly considered during attention computation, leading to more discriminative feature representations. Figure 3.14 illustrates the flow of the gated self-attention mechanism.

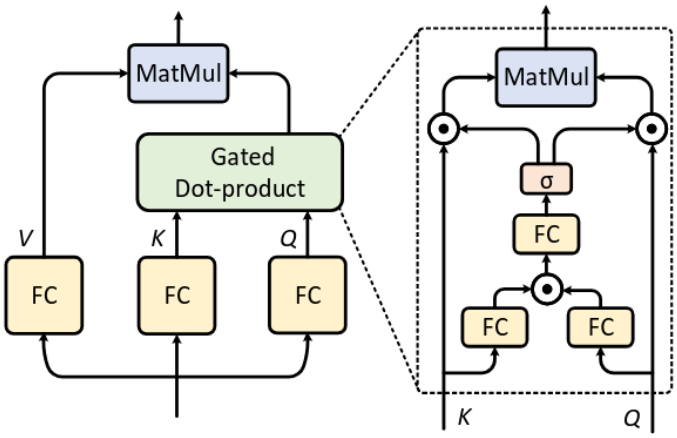


Figure 3.14. *Gated self-attention*
Source: [4]

3.6 Explainability in Deep Learning

In recent years, deep learning models have achieved remarkable success across various domains, including computer vision, natural language processing, and healthcare. However, their increasing complexity and black-box nature have raised concerns about their interpretability and trustworthiness. Explainability refers to the ability to understand and interpret the decisions made by machine learning models, particularly complex neural networks. The aim is to provide human-interpretable insights into how models process input data and generate predictions.

The importance of explainability becomes particularly pronounced in high-stakes applications such as medical imaging, autonomous driving, and finance, where incorrect decisions can have severe consequences. For instance, in medical diagnosis, understanding which parts of a medical image influence a model's prediction can help clinicians verify the validity of the results

and identify potential errors.

Explainability methods are broadly categorized into two types: *global* and *local*. Global explainability provides insights into the overall behavior of a model, such as understanding the importance of features or identifying the relationship between inputs and outputs. Local explainability, on the other hand, focuses on interpreting individual predictions, such as highlighting specific regions of an input image or text that contributed most to a given output [35].

Numerous techniques have been developed to improve explainability in deep learning. These include saliency maps, feature importance methods (e.g., SHAP and LIME), activation visualizations, and class activation mapping (CAM). Among these, gradient-based approaches have gained popularity for their ability to generate visual explanations for CNNs. Grad-CAM, in particular, introduced by Selvaraju et al. [36], generates visual explanations for CNN-based models by leveraging gradients flowing into the final convolutional layers.

3.6.1 Grad-CAM

Grad-CAM highlights the important regions of an input image that are most relevant to the model's prediction, offering insights into the model's decision-making process. Figure 7.1 demonstrates this concept by showcasing Grad-CAM visualizations for two target classes: "Cat" and "Dog". The heatmaps highlight the distinct regions of the image that contribute most to the predictions for each class.

Grad-CAM builds on class activation mapping (CAM) by generalizing it to models without global average pooling layers. It achieves this by combining the gradients of a target class y^c with the feature maps of a convolutional layer. The key steps in Grad-CAM are as follows:
Let A^k denote the activation map of the k-th convolutional filter in the target layer, and let y^c represent the score for the target class c. The importance of each feature map A^k for the target class is computed using the gradient $\frac{\partial y^c}{\partial A^k}$, pooled globally as follows:

$$a_k^c = \frac{1}{Z} \sum_{i,j} \frac{\partial y^c}{\partial A_{ij}^k}, \tag{3.36}$$

where Z is the total number of pixels in the activation map A^k, and A_{ij}^k is the value of A^k at spatial position (i,j). The weights a_k^c represent the contribution of the k-th feature map to the target class c.
The Grad-CAM heatmap $L_{\text{Grad-CAM}}^c$ is then computed as:

$$L_{\text{Grad-CAM}}^c = \text{ReLU}\left(\sum_k a_k^c A^k \right), \tag{3.37}$$

where ReLU ensures that only positive contributions are considered, as negative values are not relevant for the class c. The resulting heatmap $L_{\text{Grad-CAM}}^c$ highlights the regions of the input image that contribute most to the prediction.

The Grad-CAM process involves the following steps:

- Forward pass: Compute the activations A^k of the target convolutional layer and the model's output y^c for the target class c.
- Backward pass: Compute the gradients $\frac{\partial y^c}{\partial A^k}$ with respect to the activations A^k.
- Compute weights: Use Equation 3.36 to calculate the weights a_k^c by performing global average pooling on the gradients.
- Generate heatmap: Combine the weights with the activation maps and apply ReLU to obtain $L^c_{\text{Grad-CAM}}$ using Equation 3.37.
- Overlay the heatmap: Normalize $L^c_{\text{Grad-CAM}}$ and overlay it on the input image for visualization.

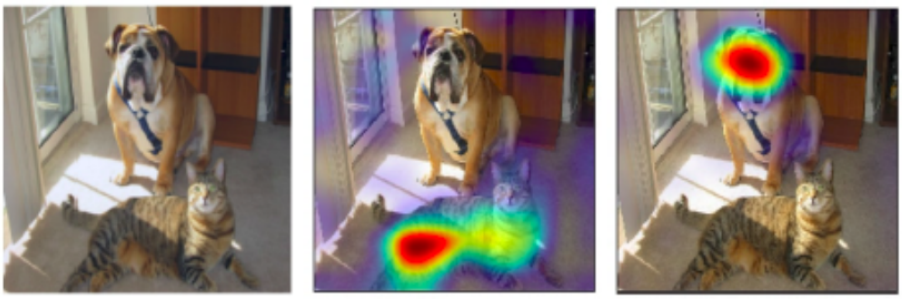


Figure 3.15. *Illustration of Grad-CAM visualizations*
Source: [36]

3.7 Hyperparameter Tuning

In the development of machine learning models, hyperparameter tuning plays a crucial role in optimizing performance. Hyperparameters are the parameters of the learning algorithm that are not updated during training but are set before the training process begins. Examples include learning rate, batch size, the number of hidden layers, and the number of units in each layer. Selecting the best combination of hyperparameters can significantly impact the model's accuracy, efficiency, and ability to generalize to unseen data. However, finding the optimal hyperparameter configuration can be a complex and computationally expensive process, especially for models with high-dimensional parameter spaces.

3.7.1 Methods

There are several established methods for hyperparameter optimization:
Grid search is a brute-force approach that systematically searches through a predefined subset of the hyperparameter space. It evaluates every possible combination of hyperparameters

within this subset to identify the best performing configuration. While simple and effective for low-dimensional problems, grid search can become computationally prohibitive for high-dimensional or continuous hyperparameter spaces.

Random search selects hyperparameter combinations randomly, enabling it to explore a wider area of the hyperparameter space compared to grid search. Studies have shown that random search is more efficient in high-dimensional spaces, as it reduces computational overhead while maintaining the ability to find optimal hyperparameter configurations. Unlike grid search, random search does not evaluate all possible combinations, making it faster and more suitable for problems where only a subset of hyperparameters significantly influences performance.

Bayesian Optimization builds a probabilistic model of the objective function and uses this model to iteratively select hyperparameters that are likely to improve performance. This method balances exploration of new regions of the hyperparameter space and exploitation of regions known to perform well. Bayesian Optimization is particularly effective for problems with expensive evaluation functions, as it focuses the search on promising areas of the hyperparameter space, reducing computational costs [37].

3.7.2 Weights & Biases (W&B)

Modern tools like Weights & Biases (W&B) simplify and accelerate the hyperparameter tuning process. W&B enables users to define hyperparameter sweeps using YAML configurations, track experiments in real-time, and visualize results in an intuitive interface. By supporting various tuning strategies like grid search, random search, and Bayesian Optimization, W&B makes it easier to manage and optimize complex machine learning workflows.

W&B provides an interface to monitor the sweep process, visualize performance metrics across hyperparameter configurations, and identify the best-performing model. Integrating W&B into the hyperparameter tuning process enables practitioners to streamline experimentation, gain deeper insights into model performance, and improve overall productivity [38]

3.8 Evaluation of Machine Learning Algorithms

In the field of machine learning, and specifically in statistical classification problems, the confusion matrix is defined as a specialized table that enables the visualization of the performance of a supervised learning algorithm. In unsupervised learning, this matrix is referred to as a matching matrix. Each row of the matrix represents the instances in a predicted class, while each column represents the instances in an actual class.

True Condition	**Condition Positive**	**Condition Negative**
Predicted Condition Positive	True Positive (TP)	False Positive (FP)
Predicted Condition Negative	False Negative (FN)	True Negative (TN)

Table 3.1. *Confusion Matrix*

The variables of the confusion matrix are defined as follows:

- **True Positive (TP)**: Cases where the model correctly predicts a positive outcome.
- **True Negative (TN)**: Cases where the model correctly predicts a negative outcome.
- **False Positive (FP)**: Cases where the model predicts a positive outcome incorrectly.
- **False Negative (FN)**: Cases where the model predicts a negative outcome incorrectly.

Based on the confusion matrix variables, the following performance metrics are defined:

Accuracy: The proportion of correct predictions over the total number of predictions:

$$\text{accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} = \frac{TP + TN}{TP + TN + FP + FN}. \tag{3.38}$$

However, accuracy alone may not be sufficient to evaluate a model's performance, especially in cases where class distributions are highly imbalanced. For example, in datasets dominated by one class, the classifier may favor the majority class, resulting in high accuracy but poor performance on minority classes. Thus, additional metrics are necessary.

Recall (True Positive Rate/Sensitivity): The proportion of positive samples correctly predicted by the classifier:

$$\text{recall} = \frac{TP}{TP + FN}. \tag{3.39}$$

Precision: The proportion of correct positive predictions among all positive predictions made by the classifier:

$$\text{precision} = \frac{TP}{TP + FP}. \tag{3.40}$$

F1-Score: A metric that combines precision and recall. It is the harmonic mean of precision and recall, mathematically expressed as:

$$F1\text{-score} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = \frac{2 \cdot TP}{2 \cdot TP + FP + FN}. \tag{3.41}$$

False Positive Rate (FPR): The proportion of negative samples incorrectly predicted as positive:

$$FPR = \frac{FP}{TN + FP}. \tag{3.42}$$

Specificity (True Negative Rate): The proportion of negative samples correctly predicted by the classifier:

$$\text{Specificity} = \frac{TN}{TN + FP}. \tag{3.43}$$

ROC Curve and AUC (Area Under Curve): The Receiver Operating Characteristic (ROC) curve is a graphical representation of a classification model's performance across all classification thresholds. It plots the True Positive Rate against the False Positive Rate, and the AUC (Area Under Curve) measures the area under this curve, summarizing the overall performance of the mode [39].

Chapter 4

Related Work

In this chapter, a comprehensive literature review of the methods that have been used for the diagnosis of Alzheimer's will be conducted. Additionally, a comparative study of the evaluation metrics and results achieved by each study will be carried out, which will guide us in the development of our own models aiming to improve these metrics.

One of the earliest studies in predicting AD from neuroimaging data is by Vemuri et al. (2008) [7], which utilized structural MRI analyzed through a support vector machine (SVM) classifier. The study employed three models: Model I used only MRI-based tissue density features, while Models II and III incorporated demographic information (age and gender) and APOE genotype data, respectively. The dataset consisted of 190 AD patients and 190 cognitively normal (NC) controls, matched for age and gender, recruited from the Mayo Clinic Alzheimer's Disease Research Center (ADRC) and the Alzheimer's Disease Patient Registry (ADPR). In Model I, the analysis of tissue densities identified key brain regions affected by AD, such as the hippocampus and medial temporal lobe, achieving a sensitivity and specificity of 86%. This work demonstrated the strong diagnostic potential of structural neuroimaging alone, with Models II and III showing slight improvements when additional information was included.

The study by Lebedev et al. (2014) [8] investigated the use of Random Forest (RF) classifiers for detecting AD and distinguishing it from healthy controls using structural MRI data. The method included preprocessing the MRI data using Freesurfer software to extract morphometric measurements such as cortical thickness and subcortical volumes. Recursive feature elimination was employed to optimize feature selection, and models were trained using Random Forest (RF) with different morphometric modalities. The model with the best performance achieved a sensitivity of 88.6%, specificity of 92%, and an area under the ROC curve of 0.94 for distinguishing AD from CN in the ADNI dataset.

The study by Liu et al. (2014) presents a deep learning-based framework for diagnosing AD and MCI, using multimodal neuroimaging data, specifically MRI and PET, from the ADNI database [10]. The proposed method utilizes stacked sparse auto-encoders for dimensionality reduction and data fusion, combined with a softmax regression layer for multi-class classification. The model achieved an accuracy of 87.76%, sensitivity of 88.57%, and specificity of 87.22% for binary classification of AD vs NC. Additionally, in the 4-class classification task (NC, MCI

converters, MCI non-converters, and AD), the framework attained an accuracy of 47.42%. This deep learning approach overcomes traditional bottlenecks by requiring fewer labeled samples and minimal domain-specific prior knowledge. This is achieved through unsupervised pre-training with stacked auto-encoders, which reduces reliance on labeled data while automatically extracting deep representations from neuroimaging data.

The study by Payan and Montana (2015) was one of the first to apply convolutional neural networks CNNs to the classification of AD using neuroimaging data [9]. They developed a classification framework using 3D CNNs, marking an early and significant contribution to the field. The methodology involved preprocessing MRI scans from the ADNI dataset to normalize voxel intensities and reduce intersubject variability. Sparse autoencoders were first trained to learn feature representations from 3D patches of brain images, which were then used to initialize the 3D-CNN. The CNN architecture consisted of convolutional, pooling, and fully connected layers, allowing the model to capture local 3D patterns and hierarchical features directly from the volumetric MRI data. Their model achieved an accuracy of 95.39% for AD vs. NC, 86.84% for AD vs. MCI, and 92.11% for MCI vs. NC, outperforming the 2D-CNN approach and many traditional machine learning methods.

The study by Sarraf and Tofighi (2016) employed 2D CNNs, specifically the LeNet-5 architecture, to classify AD using functional MRI (fMRI) data [40]. The data, obtained from the ADNI dataset, were preprocessed through standard pipelines, including motion correction, skull stripping, and spatial smoothing, before being converted into 2D JPEG images. These images were then labeled for binary classification (AD vs. NC) and processed through the CNN. The network was trained using 60% of the data, validated on 20%, and tested on the remaining 20%. The model achieved a mean classification accuracy of 96.86% across five runs.

Donghuan Lu et al proposed a novel deep learning framework, the Multimodal and Multiscale Deep Neural Network (MMDNN), for the early diagnosis of AD [11]. Using neuroimaging data from the ADNI, including 1,242 subjects with T1-MRI and FDG-PET scans, the framework combines structural MRI-derived brain volume features and FDG-PET-derived glucose metabolism features at multiple scales. The method involves preprocessing the images to extract multiscale patch-wise features and training independent neural networks for each scale and modality, followed by a feature-fusion network to generate predictions. For the classification of NC vs AD, the MMDNN achieved 84.6% accuracy, with a sensitivity of 80.2% and specificity of 91.8%. Additionally, the framework also showed high performance in predicting conversion from MCI to AD within 1-3 years prior to diagnosis.

The study by Huang et al. (2019) introduced a revolutionary approach to leveraging multimodal imaging data, specifically T1-weighted MRI and FDG-PET, for Alzheimer's Disease diagnosis [41]. The methodology centered around the extraction of 3D patches from the hippocampal region, a key area of atrophy in AD, allowing the model to capture complementary structural and metabolic features. The study utilized the ADNI dataset, comprising 731 NC subjects, 647 AD patients, 441 stable Mild Cognitive Impairment (sMCI) subjects, and 326 progressive MCI

(pMCI) subjects. The model achieved classification accuracies of 90.10% for AD vs. NC, 87.46% for NC vs. pMCI, and 76.90% for sMCI vs. pMCI. This study showed the power of combining multimodal data with deep learning for Alzheimer's diagnosis and has inspired many other researchers to explore similar approaches in the field.

Song et al. (2021) [3] proposed a novel multimodal image fusion method to enhance the diagnosis of AD by creating a new composite imaging modality called GM-PET, which combines structural information from MRI and metabolic information from FDG-PET, with a focus on the gray matter (GM) region critical for AD diagnosis. Using the ADNI dataset, which included 381 subjects categorized as AD, MCI, and NC, the authors pre-processed the data through skull stripping, registration, and segmentation to isolate the GM region. This GM region was then used to fuse complementary information from MRI and PET into the GM-PET modality, which retains brain structure and metabolic data while eliminating noise. Two classification networks, a 3D Simple CNN and a 3D Multi-Scale CNN, were employed to evaluate the fused modality. The GM-PET modality demonstrated significant improvements in diagnostic accuracy, achieving 94.11% accuracy, 93.33% sensitivity, and 94.27% specificity for AD vs. NC classification, and 88.48% accuracy for MCI vs. NC classification.

Golovanevsky et al (2022) proposed the Multimodal Alzheimer's Disease Diagnosis framework (MADDi), a deep learning-based system for diagnosing AD and MCI using imaging (MRI), genetic (SNPs), and clinical data from the ADNI dataset [42]. The dataset included 2,384 participants, with 239 participants having data across all three modalities. The model utilized modality-specific neural network backbones (a CNN for MRI and fully connected networks for clinical and genetic data), followed by self-attention layers to identify critical intra-modality features and cross-modal attention layers to capture interactions between modalities. MADDi achieved a state-of-the-art accuracy of 96.88%, with an F1-score of 91.41%. Unimodal models achieved lower accuracies (clinical: 80.59%, genetic: 77.78%, imaging: 92.28%), highlighting the advantage of multimodal fusion.

Castellano et all (2024) conducted a study on automated AD detection using a multi-modal approach that integrates 3D MRI and amyloid PET imaging, along with transfer learning strategies [12]. Using the OASIS-3 dataset, which includes imaging data from 1098 participants, the researchers developed CNN models to evaluate uni-modal (MRI or PET) and multi-modal configurations. Transfer learning was implemented by pre-training on 3D PET data and fine-tuning on MRI data, and vice versa, but these methods did not surpass uni-modal or multi-modal models, suggesting limited cross-modality feature applicability. The best-performing model, a multi-modal fusion model, achieved an accuracy of 95%, sensitivity of 93.33%, and specificity of 96.66%, outperforming uni-modal methods. MRI scans alone demonstrated stronger diagnostic performance than PET, but the integration of MRI and PET captured complementary features, significantly enhancing diagnostic accuracy. Grad-CAM explainability analysis identified critical brain regions associated with AD, such as the medial temporal lobe and frontal gyrus, emphasizing the clinical relevance of the findings.

Reference Number	Modalities	Method	Dataset	Task	ACC	SEN	SP	AUC
[7]	MRI	SVM	ADRC and ADPR	NC vs AD	N/A	86%	86%	N/A
[8]	MRI	Random Forest	ADNI	NC vs AD	N/A	88.6%	92%	0.94
[10]	MRI and PET	Stacked Auto-Encoders	ADNI	NC vs AD MCI vs NC	87.76% 76.92%	88.57% 74.29	87.22% 78.13%	N/A
[9]	MRI	3D CNNs	ADNI	NC vs MCI MCI vs AD NC vs AD	92.11% 86.84% 95.39%	N/A	N/A	N/A
[40]	MRI	2D CNNs	ADNI	NC vs AD	96.86%	N/A	N/A	N/A
[11]	MRI and PET	MMDNN	ADNI	NC vs AD	84.6%	80.2%	91.8%	N/A
[41]	MRI and PET	3D CNNs	ADNI	AD vs NC NC vs pMCI	90.10% 87.46%	90.85% 90.73%	89.21% 80.61%	90.84% 87.61%
[3]	MRI and PET (GM-PET)	3D Simple CNN and 3D Multi-Scale CNN	ADNI	NC vs MCI MCI vs AD NC vs AD	88.48% 84.83% 94.11%	93.44% 71.19% 94.44%	85.60% 94.69% 95.04%	N/A
[42]	MRI, SNPs and clinical data	CNN & attention	ADNI	NC vs MCI vs AD	96.88%	N/A	N/A	N/A
[12]	MRI and PET	2D&3D CNNs	OASIS-3	NC vs AD	95.00	93.33	96.66	93.00

4.1 Limitations of the state-of-the-art approaches

The above literature review, along with other studies, has identified several limitations in existing methods. Firstly, multimodal approaches have not been adequately studied. Even in cases where they have been applied, the integration of different modalities is usually limited to a simple concatenation, overlooking the interactions between them.

Furthermore, a significant limitation concerns the inability of existing methods to dynamically adapt to input data. Most approaches rely on dense layers for the final classification of patients, which, due to their static nature, struggle to adapt to more complex and heterogeneous data. This is a critical issue in the case of MCI, a stage that is inherently associated with a high diagnostic complexity.

Another key limitation involves the interpretability of models. Most developed models function as "black boxes," where they receive input data and produce output results without providing information about the features that contributed to their final decision.

This poses a significant problem in clinical practice, as for a doctor to evaluate the model's results, it is essential to know which brain regions are considered pathological. The lack of transparency undermines trust and complicates the implementation of these methods in diagnosis and clinical decision-making.

To address these limitations, the following methodology aims to overcome these challenges by introducing a more adaptive, multimodal, and interpretable approach.

Chapter 5

Dataset and Preprocessing

5.1 The Alzheimer's Disease Neuroimaging Initiative

The Alzheimer's Disease Neuroimaging Initiative (ADNI), established in 2004, is a groundbreaking collaborative study designed to develop biomarkers for the early detection and monitoring of Alzheimer's disease progression [43]. ADNI brings together academic institutions, industry leaders, and governmental agencies to create an extensive dataset, serving as a foundation for significant advancements in Alzheimer's diagnostics and therapeutic research.

ADNI focuses on identifying biomarkers that signal the onset of AD before clinical symptoms appear, monitoring disease progression, and enhancing clinical trial designs. Participants include Cognitively Normal (NC) individuals, those with Mild Cognitive Impairment (MCI), and individuals with clinically diagnosed AD, recruited from diverse sites across the United States and Canada to ensure a representative sample.

The dataset collects multimodal data, including magnetic resonance imaging (MRI), fluorodeoxyglucose positron emission tomography (FDG-PET), cerebrospinal fluid (CSF) biomarkers, genetic data, clinical assessments, and psychometric tests, providing a comprehensive view of AD's biological and clinical dimensions. A key feature of ADNI is its open-access policy, enabling researchers worldwide to utilize the data via the Laboratory of Neuro Imaging (LONI). For more information, visit the ADNI website: https://adni.loni.usc.edu/.

5.1.1 Data Overview

In this study, subjects with both T1-weighted MRI (specifically MPRAGE scans for their superior quality) and FDG-PET scans were selected, ensuring the scans were acquired closely in time to improve alignment between the modalities. The final cohort consisted of 370 subjects from the ADNI dataset: 73 with AD, 188 with MCI, and 118 NC individuals. Figure 5.1 illustrates representative MRI slices (sagittal, coronal, and axial views) from subjects in each diagnostic group. Additionally, the distributions of group, age and gender of the individuals are depicted in Figure 5.2.

The MRI and FDG-PET images in ADNI undergo several preprocessing steps.For MRI, the following corrections are applied:

1. **Gradwarp:** Corrects image geometry distortion caused by the gradient model.
2. **B1 non-uniformity correction:** Addresses intensity variations using B1 calibration scans.
3. **N3 histogram peak-sharpening algorithm:** Reduces intensity non-uniformity.

In this study, we utilized fully preprocessed MRI data. To achieve consistency across different systems, the baseline FDG-PET scans are processed through the following steps:

1. **Co-registration of dynamic frames:** Six 5-minute FDG-PET frames acquired 30–60 minutes post-injection are co-registered to the first frame to reduce the effects of patient motion.
2. **Averaging:** The co-registered frames are averaged.
3. **Standardization of image and voxel size:** The averaged image is reoriented into a standard 160 $\times$ 160 $\times$ 96 voxel grid with 1.5 mm cubic voxels, corrected for anterior commissure–posterior commissure alignment, and intensity-normalized using a subject-specific mask so that the average voxel value within the mask equals one.
4. **Uniform resolution:** The normalized image is smoothed using a scanner-specific filter to achieve a uniform isotropic resolution of 8 mm full width at half maximum.

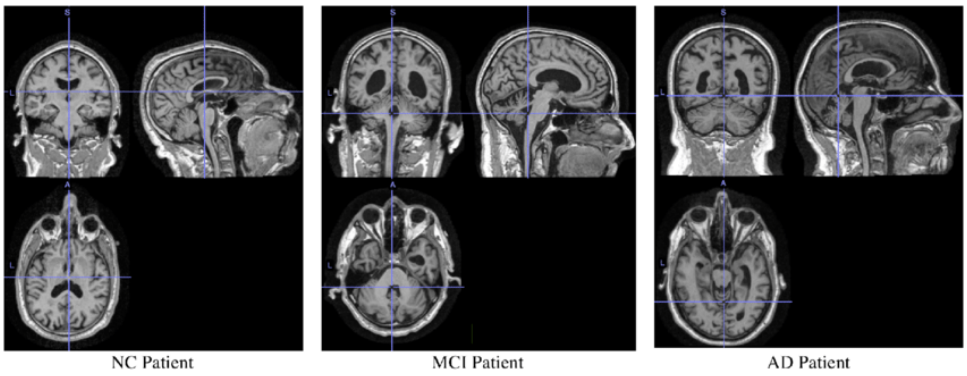


Figure 5.1. *MRI of an NC, MCI, an AD patient. These are images of MRI scans from ADNI patients. The images are oriented in coronal, sagittal, an axial view.*

Source: [44].

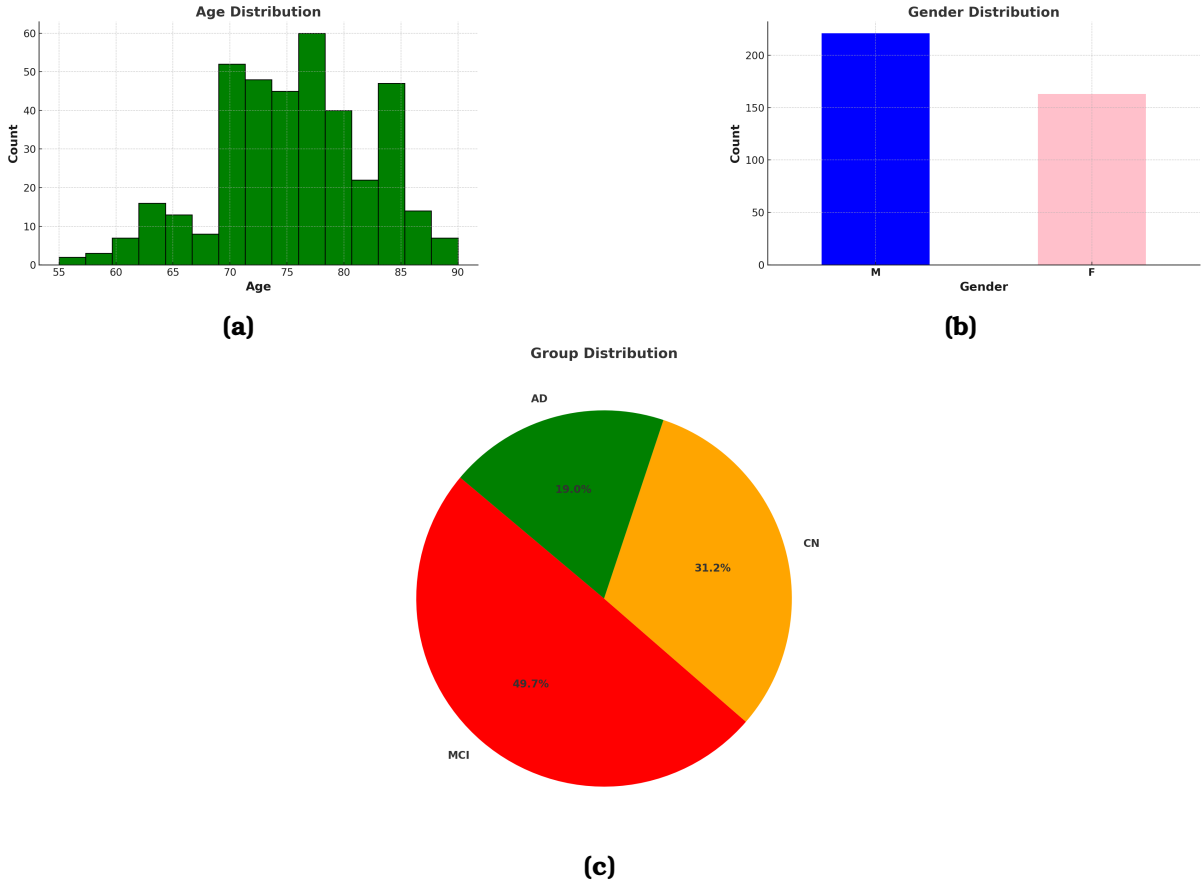


Figure 5.2. *Visualizations of participant distributions: (a) Age distribution, (b) Gender distribution, and (c) Diagnostic group distribution.*

5.1.2 Preprocessing Steps

The preprocessing of MRI scans begins with skull-stripping, implemented using the FSL (FMRIB Software Library) package, specifically the Brain Extraction Tool (BET). Skull-stripping removes non-brain tissues such as the skull, scalp, and dura mater, isolating the brain structure for more focused analysis. In the processing pipeline, a threshold parameter controls the aggressiveness of tissue removal and is set to 0.5 for balanced extraction. Bias field correction

is also applied to improve the clarity of the extracted brain region, enhancing the overall quality of the output.

The skull-stripped MRI (SS-MRI) images are then affine transformed to the MNI152 space, a widely adopted universal brain atlas template. This transformation employs the FLIRT (FMRIB's Linear Image Registration Tool) module within the FSL package, which applies a linear affine transformation to correct for spatial discrepancies between subjects. The registration process aligns MRI scans by correcting translations, rotations, and scaling, standardizing the images to the consistent orientation and voxel dimensions of MNI152 space.

For FDG-PET scans, preprocessing begins withPET skull-stripping to isolate the brain structure, followed by co-registration to their corresponding MNI-aligned MRI images. Both steps are analogous to those performed for MRI scans, ensuring that the PET images adopt the same spatial orientation and voxel resolution (e.g., $1.0 \times 1.0 \times 1.0$ mm) as their MRI counterparts. This alignment guarantees consistency across modalities, facilitating multimodal integration and enabling the model to effectively learn spatial relationships inherent in the data [3].

To improve computational efficiency, the co-registered PET and MRI images are resized to a lower resolution of $160 \times 180 \times 80$. This resizing maintains essential structural details while reducing the computational complexity for subsequent analysis tasks such as classification. Figure 5.3 illustrates the preprocessing pipeline for both MRI and PET data of the same subject, showcasing the steps of skull-stripping and registration to the MNI152 template, alongside the spatial alignment results in transverse, sagittal, and coronal planes for each modality. The displayed subject belongs to the NC group.

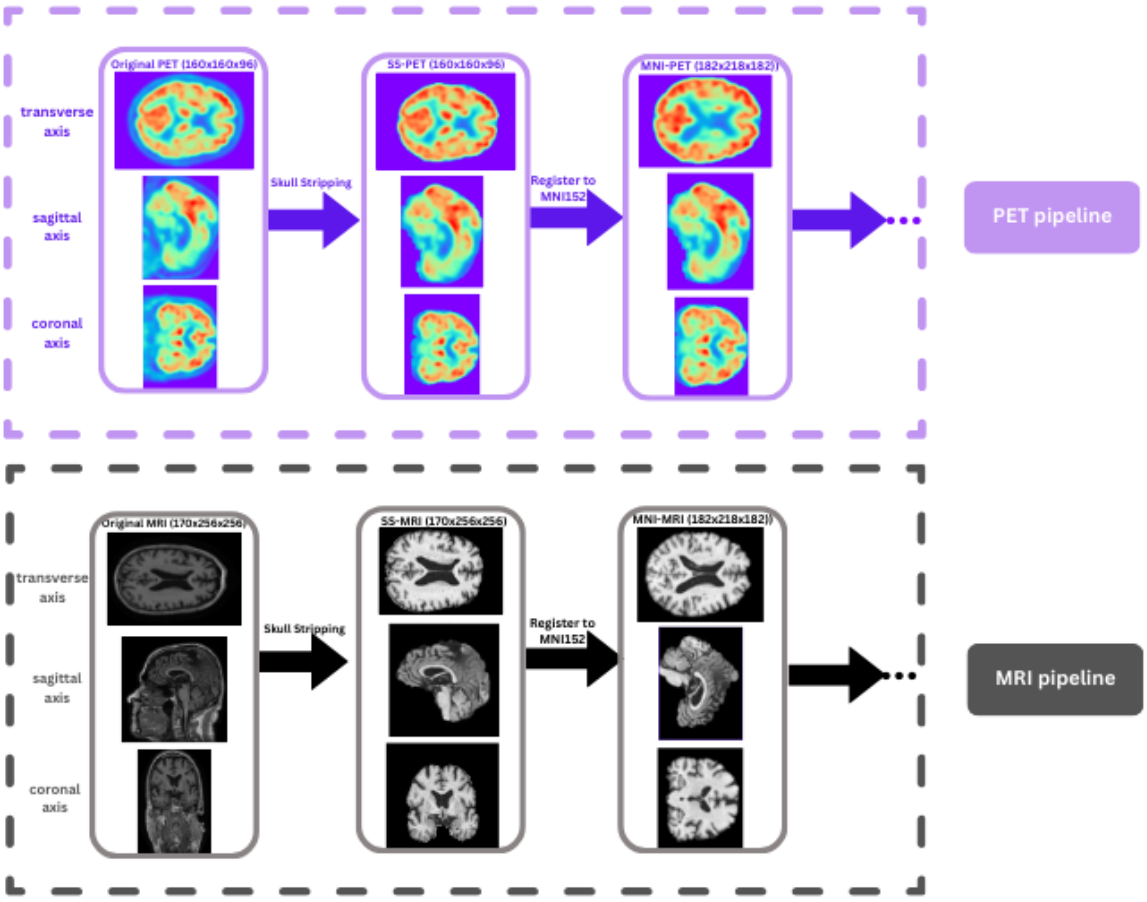


Figure 5.3. *Preprocessing pipeline for MRI and PET scans.*

Chapter 6

Methodology

The methodology of this study follows a pipeline that integrates MRI and PET imaging data, processed to classify subjects into MCI, AD, and NC categories. As illustrated in Figure 6.1, the pipeline begins with extracting modality-specific features from MRI and PET scans through dedicated pathways. These features are then fused using different techniques to create a unified representation. The resulting combined features are fed into a Mixture of Experts model, which leverages specialized sub-models to refine and enhance classification accuracy. In this chapter, we will explore each component of the methodology in detail, including feature extraction using 3D CNNs, fusion strategies, and the Mixture of Experts model.

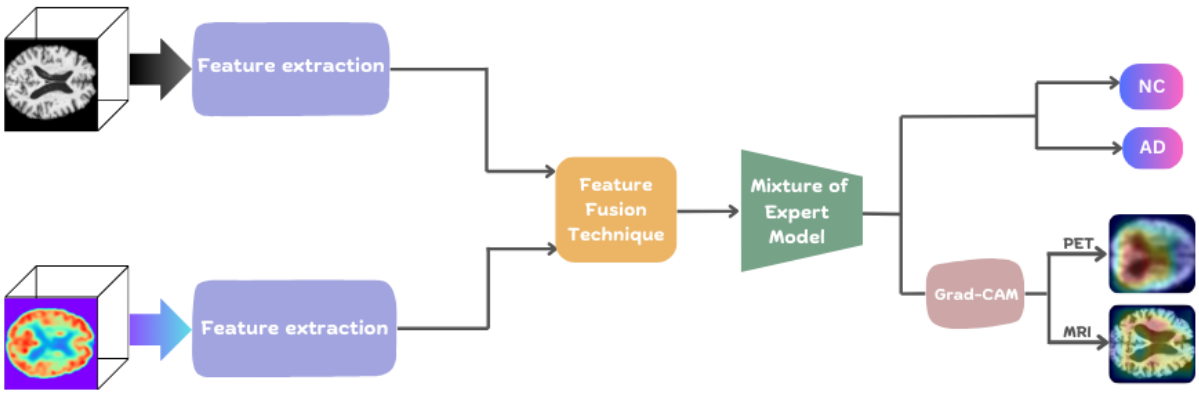


Figure 6.1. *Methodology pipeline*

6.1 CNN Architecture

As discussed in Section 3.3, CNNs have gained significant attention for their effectiveness in medical image classification. Traditional 2D CNN approaches process 3D medical images slice by slice, disregarding anatomical context in z axis. In contrast, 3D CNNs consider volumetric data as a whole, capturing richer spatial information but at the cost of increased computational complexity and memory usage due to the higher number of parameters.

As shown in Figure 6.2, the CNN architecture used in this study consists of four 3D convolutional layers, each with a kernel size of 3×3×3. These layers are followed by batch normalization to stabilize training and ReLU activation to introduce non-linearity. The first convolutional layer generates eight feature maps, with subsequent layers producing 16, 32, 64 and 128 feature maps, respectively. Max-pooling layers are applied after each convolutional block to progressively reduce spatial dimensions while preserving key features. The pooling operations use kernel sizes of 2×2×2, 3×3×3, and 4×4×4. In the final stage, a global average pooling layer further compresses the spatial dimensions, preparing the feature maps for the next processing steps. Dropout regularization is applied after pooling layers to reduce the risk of overfitting.

The architecture features two separate but identical pathways for MRI and PET images, each designed to extract distinct structural and functional information from the respective modalities. Both pathways culminate in 128 feature maps, which are then concatenated along the feature dimension. In the next section, we will discuss three different fusion methods used to integrate these extracted features.

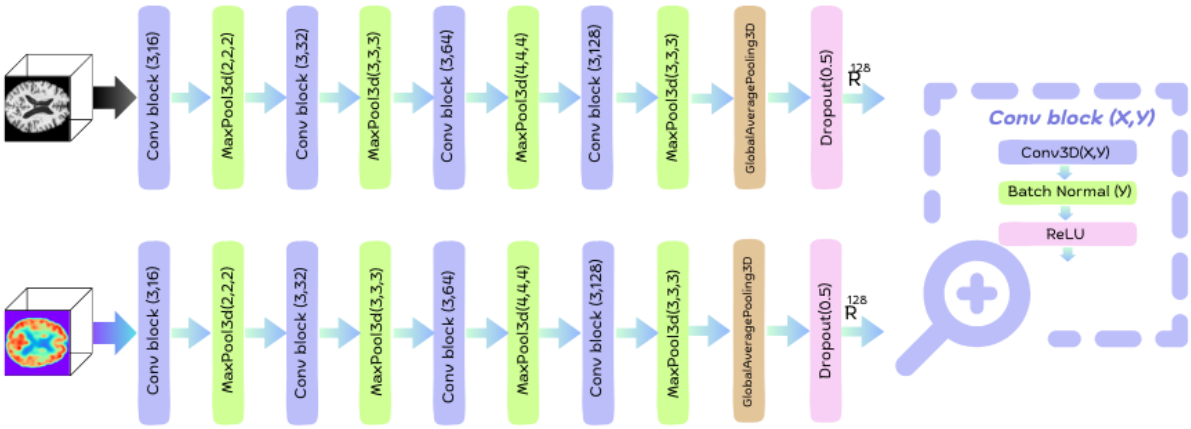


Figure 6.2. *The CNN architecture used in this thesis*

6.2 Feature Fusion Techniques

Three feature integration techniques are employed to process multimodal data. The first technique, as illustrated in 6.3, is feature concatenation, which directly combines MRI and PET features into a single feature vector. This straightforward yet effective approach preserves all the information from both modalities, serving as a baseline for comparison with more advanced methods. The resulting concatenated vector has a dimensionality of 256, as each modality contributes 128 features.

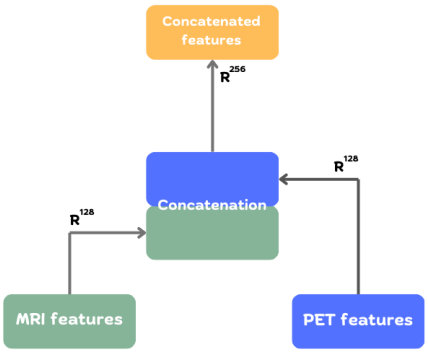


Figure 6.3. *Simple concatenation of the MRI and PET features*

The second technique, shown in 6.4, employs GMU, which incorporates gating mechanisms to balance information sharing and separation between modalities. The GMU takes 128 features from MRI and 128 features from PET as input and produce a unified output of 128 features. The weights $W^p \in \mathbb{R}^{128}$, $W^m \in \mathbb{R}^{128}$, and $W^z \in \mathbb{R}^{128}$ correspond to the PET, MRI, and gating components, respectively. As described in 3.21, 3.22, and 3.23, these trainable weights are used to control the contribution of each modality towards the final classification. A more detailed explanation of the GMU process is provided in 3.5.1.

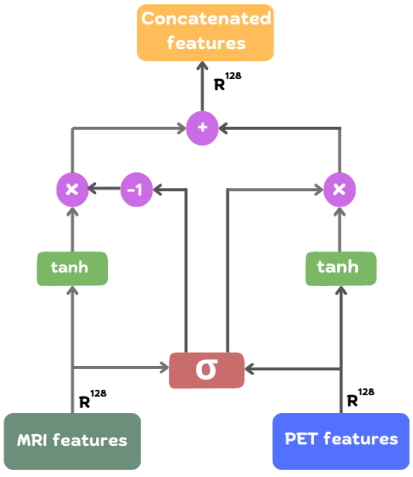


Figure 6.4. *GMU for the concatenation of MRI and PET features*

The third feature fusion technique is Gated Self-Attention. This approach allows the model to capture both intra-modal and inter-modal interactions simultaneously while retaining only the most relevant connections. Given the two modalities, the first step is to concatenate their representations:

$$Z = [MRI; PET] \tag{6.1}$$

Here, the concatenated representation Z is defined as $Z \in \mathbb{R}^{m \times d}$, where $m = 4 + 4$ and $d = 128$. To ensure that spatial information is retained, we extract features before applying global average pooling, treating spatial locations as tokens as shown in 6.5. This enables the self-attention mechanism to capture spatial dependencies both within and across modalities.

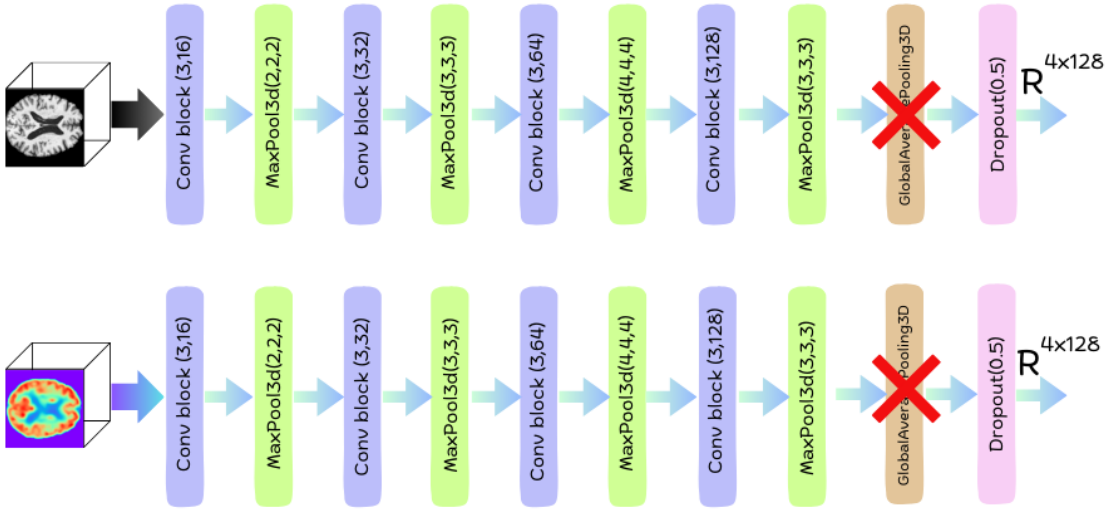


Figure 6.5. *Feature extraction without Global Average Pooling*

This representation is then used for computing the query (Q), key (K), and value (V) matrices:

$$Q = Z, \quad K = Z, \quad V = Z. \tag{6.2}$$

Then, equations 3.32, 3.34 and 3.35 are applied following the pipeline illustrated in Figure 6.6.

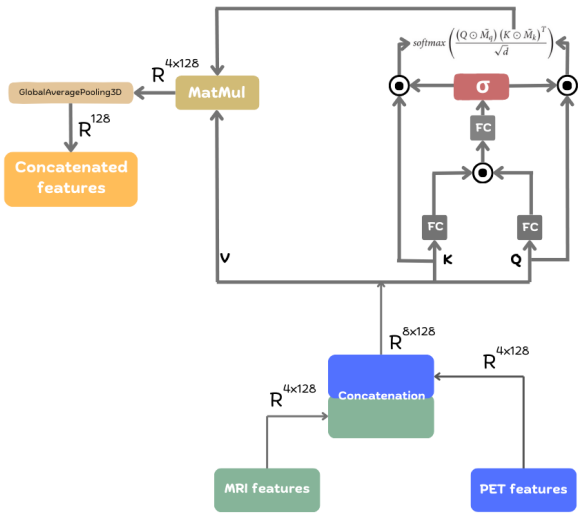


Figure 6.6. *Self-Attention mechanism after the concatenation of the MRI and PET features*

6.3 MoE architecture

As described in 3.4, we employ the MoE architecture, utilizing n experts to process the extracted features from the three previously discussed fusion methods. As shown in 6.7, each expert consists of a fully connected layer followed by a ReLU activation function, and a second fully connected layer for final classification. The model's output is computed according to 3.13, while the router determines expert selection based on the equations 3.15, 3.16, and 3.17. To ensure the correct functioning of the entire architecture, we optimize the total loss 3.20, which is derived from the importance loss 3.18 and load loss 3.19.

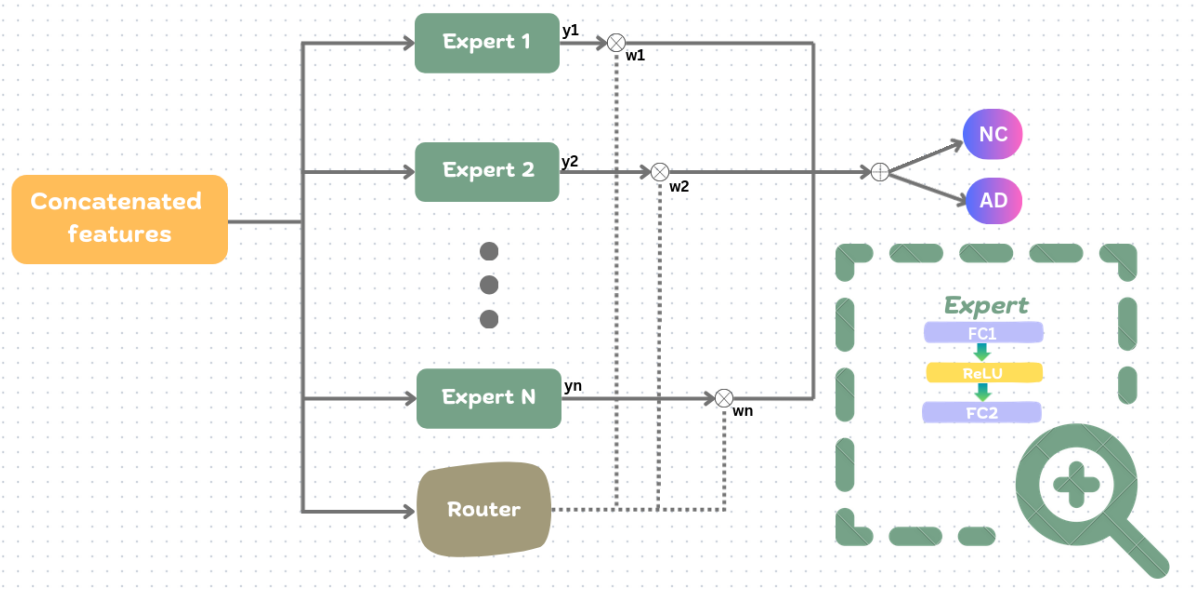


Figure 6.7. *MoE architecture of this thesis*

6.4 Experimental Setup

In this study, the networks are implemented using the PyTorch deep learning framework [45]. We perform three classification tasks: AD vs NC, AD vs MCI, and MCI vs NC. Comparative experiments are conducted on both unimodal and multimodal data. To identify the optimal hyperparameters, we utilize the wandb framework with random search, and the set of optimal hyperparameters identified during the search is summarized in Table 6.1. The Adam optimizer with an initial learning rate of 1×10^{-4} and a weight decay of 0.1 is used to update the network weights during training. Also a batch size of 4 is selected, while for regularization, we applied a dropout rate of 0.5. For the number of experts, we set $n = 5$ and $k = 4$, while for the parameter a in Eq. 3.20, we chose $a = 0.6$. Binary cross-entropy is employed as the loss function.

A 10-fold cross-validation strategy is adopted to ensure fair performance evaluation. The dataset is randomly divided into 10 subsets: two subsets are used as the test set, another two as the validation set, and the remaining six subsets are used for training. Each experiment is trained for 500 epochs, and two learning rate adjustment strategies are applied:

1. If the validation loss does not decrease within 5 epochs and the current learning rate is above 5×10^{-6}, the learning rate is reduced to half of its current value.

2. If the validation accuracy does not improve within 10 epochs and the current learning rate is above 5×10^{-6}, the learning rate is also halved.

Additionally, an early stopping strategy is employed, where training is terminated if the validation loss does not decrease within 30 epochs. The classification accuracy (ACC), sensitivity (SEN), and specificity (SPE) are used as evaluation metrics. The results are reported as the mean $\pm$ standard deviation (SD) across the 10 folds. For further analysis, we use the best-performing model from the NC vs. AD classification task, as it is the easiest task, to apply

Grad-CAM for identifying the key areas contributing to the predictions.

This analysis begins with experiments using the full architecture to gain a comprehensive understanding of the model's overall effectiveness. Additionally, we will conduct experiments to assess the impact of each component. Specifically, we will evaluate simplified architectures by systematically removing individual components (e.g., the MoE framework) and analyzing the corresponding changes in performance.

Table 6.1. *Optimal hyperparameters used for the experiments.*

Hyperparameter	**Value**
Learning rate (η)	1×10^{-4}
Weight decay	0.1
Batch size	4
Dropout rate	0.5
Number of experts (n)	5
Selected experts (k)	4
Parameter a in loss	0.6
Optimizer	Adam

Chapter 7

Results

7.1 Results of the whole architecture

Table 7.4 summarizes the classification performance of our final model using three different fusion methods: Concatenation, GMU, and Attention. The model is evaluated on three binary classification tasks: NC vs MCI, MCI vs AD and NC vs AD. We observe that GMU achieves the best results in two out of three tasks, while attention performs best in the remaining one. This suggests that more advanced fusion techniques tend to yield superior performance.

Table 7.1. *Performance of the final architecture*

Fusion Method	Task	ACC	SEN	SP	AUC
Concatenation	NC vs MCI	78.25±3.2	75.43±4.1	79.32±2.1	76.56±3.9
	MCI vs AD	80.13±5.3	79.24±5.8	81.21±5.5	76.83±8.1
	NC vs AD	89.52±3.4	87.25±3.2	89.98±4.1	89.64±2.3
GMU	NC vs MCI	**80.46** ± 3.9	**79.71** ± 4	**81.76** ± 3.9	**80.51** ± 3.5
	MCI vs AD	79.13±1.1	77.23±3.3	81.36±4.2	79.94±1.5
	NC vs AD	**95.47** ± 2.1	**94.31** ± 3.2	**96.73** ± 1.8	**95.41** ± 2.6
Attention	NC vs MCI	80.15±2.2	78.35±5.4	83.56±2.6	77.46±1.9
	MCI vs AD	**82.08** ± 2.1	**81.43** ± 1.8	**85.24** ± 2.7	**80.48** ± 3
	NC vs AD	91.53±4.7	92.28±4.4	91.07±4.7	92.29±5.2

7.2 Grad-CAM results

Figure 7.1 illustrates Grad-CAM visualizations applied to MRI and PET scans of an AD patient. The first column presents the original MRI (top) and PET (bottom) scans, while the subsequent columns display Grad-CAM heatmaps overlaid on different axial slices (*slice = 20, 30, 40*). This visualization aids in interpreting the model's decision-making process by identifying key regions contributing to AD classification. The red regions represent areas that the model considers relevant to AD, with darker shades of red signifying higher importance in the classification decision. We observe that the highlighted regions differ between MRI and PET scans, possibly indicating their complementary nature.

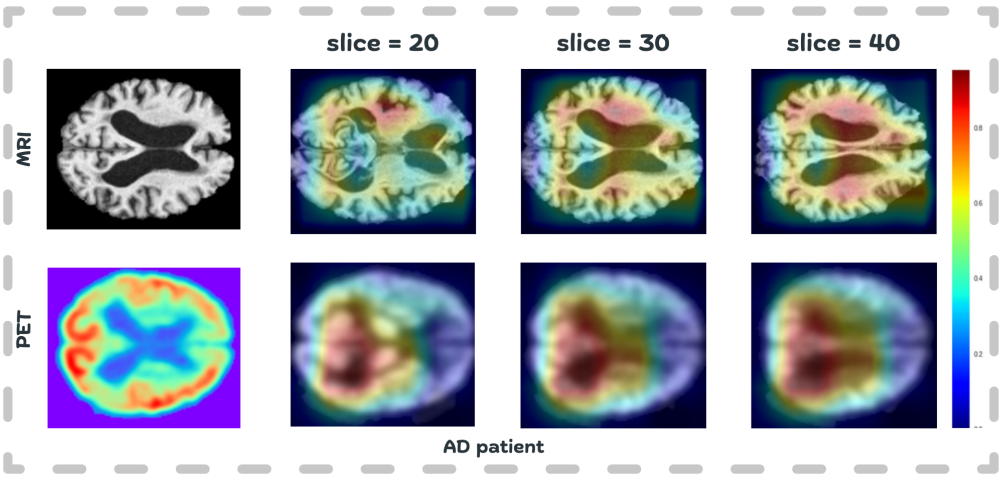


Figure 7.1. *Grad-CAM results for an AD patient*

7.3 Comparison of our best results with preliminary work

Table 7.2 presents a comparative analysis of our best-performing models against state-of-the-art techniques in AD classification. Our GMU model achieved the highest accuracy **(95.47%)** for NC vs AD classification, outperforming prior unimodal and multimodal approaches, including [12] (95.00%) and [3] (94.11%). Moreover, GMU attained the highest specificity **(96.73%)**, demonstrating superior ability in correctly identifying healthy individuals, and an ROC-AUC of **95.41%**. For MCI vs AD classification, our Attention-based model exhibited the highest sensitivity **(81.43%)**, while [3] achieved the best specificity (94.69%).

Table 7.2. *Comparison of our best results with preliminary work*

Architecture	Task	Accuracy	Sensitivity	Specificity	ROC-AUC
Unimodal approaches (MRI)					
Support Vector Machine [7]	NC vs AD	–	86	86	–
Random Forest [8]	NC vs AD	–	88.6	92	–
3D CNNs [9]	NC vs AD	95.39	–	–	–
	NC vs MCI	92.11	–	–	–
	MCI vs AD	86.84	–	–	–
Multimodal approaches					
Stacked Auto-Encoders [10]	NC vs AD	87.76	88.57	87.22	–
	NC vs MCI	76.92	74.29	78.13	–
Multiscale DNN [11]	NC vs AD	84.6	80.2	91.8	–
3D CNNs [3]	NC vs AD	94.11	**94.44**	95.04	
	NC vs MCI	**88.48**	**93.44**	**85.60**	–
	MCI vs AD	**84.83**	71.19	**94.69**	–
2D&3D CNNs [12]	NC vs AD	95.00	93.33	96.66	93.00
Our best-performing Model					
GMU	NC vs AD	**95.47**	94.31	**96.73**	**95.41**
GMU	NC vs MCI	80.46	79.71	81.76	**80.51**
Attention	MCI vs AD	82.08	**81.43**	85.24	**80.48**

7.4 Ablation Study Results

7.4.1 Results without the MoE framework

To assess the contribution of the MoE model, we will replace it with a simple architecture consisting of three fully connected layers and evaluate its performance using three different fusion methods for the MRI and PET features: concatenation, GMU and attention-based fusion. We observe that the results without the MoE framework are worse compared to those of the complete architecture.

Table 7.3. *Performance without the MoE framework*

Fusion Method	Task	ACC	SEN	SP	AUC
Concatenation	NC vs MCI	67.4±0.8	76.2±6	57.4±4.2	63.1±4.9
	MCI vs AD	68.5±2.8	75.2±5	59.3±3.1	67.4±3.9
	NC vs AD	86.48±5.2	84.32±4.2	87.18±5.6	84.8±4.8
GMU	NC vs MCI	70.5±2.9	74.2±4.1	65.6±4.8	72.3±3.9
	MCI vs AD	69.58±2.7	69.88±3.4	67.13±2.1	67.55±5.6
	NC vs AD	85.65±7.2	84.51±5.6	88.61±5.2	81.25±7.8
Attention	NC vs MCI	68.4±3.8	70.2±5.1	66.8±4.5	67.1±4.1
	MCI vs AD	73.45±2.8	75.5±4	72.3±3.5	74.3±2.9
	NC vs AD	85±8.8	84.91±6.6	84.78±7.2	83.1±7.4

7.4.2 Results of the unimodal models

Table 7.4 presents the performance metrics of unimodal models (MRI and PET) for the three classification tasks: NC vs MCI, MCI vs AD and NC vs AD.

Table 7.4. *Performance of the unimodal models*

Fusion Method	Task	ACC	SEN	SP	AUC
MRI	NC vs MCI	67.38±0.7	86.17±6	40.35±8.4	63.08±5
	MCI vs AD	64.29±4.6	67.56±5.6	62.19±4.8	61.14±5.5
	NC vs AD	75.32±3.5	80.17±11.1	65.31±10.8	76.13±5.3
PET	NC vs MCI	72.39±3.8	70.15±4.8	76.21±4.2	73.65±4.9
	MCI vs AD	70.81±2.4	68.57±4.2	75.42±4.1	69.6±5.8
	NC vs AD	81.1±1.6	81±1.6	81.88±2.6	84±6

We observe that unimodal models yield worse results compared to multimodal ones and that PET outperforms MRI in all three classification tasks.

Chapter 8

Conclusion

In this thesis, our goal was to develop a robust and reliable system for diagnosing AD. Today, Alzheimer's is a leading cause of death, with its prevalence expected to rise in the coming years. Symptoms often go undetected, making recovery impossible. This is why identifying MCI is crucial, as early detection can help slow the rapid progression of the disease.

To achieve this, we focused on detecting abnormalities in MRI and PET scans of patients categorized into one of three groups: NC, MCI, or AD. We used the ADNI dataset, which provides high-quality images of patients with both MRI and PET scans. Prior to analysis, we preprocessed the data by extracting the brain region and aligning the modalities to avoid spatial discrepancies. We began by extracting features from both imaging modalities using a 3D CNN through two distinct yet similar pathways. Next, we applied three different fusion techniques to capture both inter- and intra-modality connections. The first method was simple feature concatenation, producing 256 features. The second method was a GMU, which controls the flow of information between different modalities by dynamically weighting their contributions, resulting in 128 features. Lastly, we employed gated self-attention, which enhances feature selection by dynamically weighting the most important information across modalities and also results in 128 features. To improve classification performance and computational efficiency, we employed a MoE model. MoE consists of multiple specialized subnetworks, each trained to process a specific subset of inputs. A gating mechanism determines which experts contribute to each prediction, ensuring that only a few experts are active at any given time. This also reduces computational efficiency, as not all parameters are engaged simultaneously. Additionally, we used Grad-CAM to visualize which regions of the brain influenced the model's final decision. This is a crucial step, as it allows us to ensure that our model is not functioning as a mere black box, but rather making interpretable and biologically meaningful predictions.

For evaluation, we used accuracy, sensitivity, specificity, and ROC metrics to assess model performance. The results showed that our best-performing model was the GMU model for both the NC vs MCI and NC vs AD classification tasks reaching an accuracy of 95.47% in the NC vs AD task. However, for the MCI vs AD task, the attention-based model outperformed the others. Next, we conducted ablation studies to analyze the contribution of different components of our architecture to the final performance. We observed that replacing the MoE model with simple fully connected layers resulted in a performance drop across all classification tasks, highlighting the effectiveness of MoE in selecting relevant features.

Additionally, we evaluated the model's performance when using only a single imaging modality

(MRI or PET) instead of both. The results showed that performance was worse compared to the full multimodal architecture, demonstrating the advantage of fusing both modalities. Interestingly, we found that PET consistently outperformed MRI across all three classification tasks. This observation was further reinforced when analyzing modality contributions within the GMU model. By examining the weight distribution at the end of training of the multimodal architecture, we observed that PET received higher weight assignments compared to MRI.
Finally, we compared our best-performing model with state-of-the-art approaches in both unimodal and multimodal studies within the domain. The results demonstrated that our model outperformed previous work in terms of accuracy, specificity, and ROC metrics for the NC vs. AD classification task. Additionally, our model achieved higher sensitivity in the NC vs. MCI task. Regarding the Grad-CAM analysis, our model successfully identified disease-related regions in both MRI and PET scans, demonstrating its ability to focus on relevant areas. We also observed that the highlighted regions differed between the two modalities, confirming their complementary nature.

8.1 Future Work

Several directions can be explored to further improve our work. One potential avenue is experimenting with different feature extraction pathways for MRI and PET scans. By designing modality-specific extraction strategies, we may be able to capture the most informative features from each modality independently, potentially enhancing overall classification performance.
Another important aspect to investigate is alternative preprocessing techniques for MRI scans. Given that the MRI model underperformed compared to the PET model, evaluating different preprocessing pipelines could help optimize feature extraction and improve MRI's contribution to the classification task. Exploring advanced neuroimaging preprocessing packages may lead to better-aligned and higher-quality inputs.
Additionally, early fusion techniques could be explored to improve multimodal learning. Instead of processing MRI and PET features separately before fusion, early fusion combines the raw input data or low-level extracted features at an earlier stage in the network. This approach may help the model learn shared representations across modalities more effectively.
Finally, Generative Adversarial Networks (GANs) could be utilized to address the limited dataset size. GANs can be employed for data augmentation, generating synthetic but realistic MRI and PET scans to increase the training set diversity. This could improve the model's ability to generalize, particularly in distinguishing between MCI and AD or NC and MCI, where differences can be subtle.

Bibliography

[1] Alzheimer's Association. *2019 Alzheimer's disease facts and figures*. *Alzheimer's & Dementia*, 15(3):321–387, 2019.

[2] Mayo Clinic Staff. *Alzheimer's disease - Symptoms and causes*, 2025. Accessed: 16-Feb-2025.

[3] Juan Song, Jian Zheng, Ping Li, Xiaoyuan Lu, Guangming Zhu και Peiyi Shen. *An Effective Multimodal Image Fusion Method Using MRI and PET for Alzheimer's Disease Diagnosis*. *Frontiers in Digital Health*, 3:637386, 2021.

[4] Zhou Yu, Jun Yu, Yuhao Cui, Dacheng Tao και Qi Tian. *Gated Self-Attention Network for Image-Text Representation Learning*. *arXiv preprint arXiv:1908.04107*, 2019.

[5] Robert A Jacobs, Michael I Jordan, Steven J Nowlan και Geoffrey E Hinton. *Adaptive mixtures of local experts*. *Neural Computation*, 3(1):79–87, 1991.

[6] Michael I. Jordan και Robert A. Jacobs. *Hierarchical mixtures of experts and the EM algorithm*. *Neural Computation*, 6(2):181–214, 1994.

[7] Prashanthi Vemuri, Jeffrey L. Gunter, Matthew L. Senjem, Jennifer L. Whitwell, Kejal Kantarci, David S. Knopman, Bradley F. Boeve, Ronald C. Petersen και Clifford R. Jack. *Alzheimer's disease diagnosis in individual subjects using structural MR images: Validation studies*. *NeuroImage*, 39(3):1186–1197, 2008.

[8] Alexander V. Lebedev, Eric Westman, Gerard J. P. Van Westen, Maja G. Kramberger, Arvid Lundervold, Dag Aarsland και et al. *Random Forest ensembles for detection and prediction of Alzheimer's disease with good between-cohort robustness*. *NeuroImage: Clinical*, 6:115–125, 2014.

[9] Adrien Payan και Giovanni Montana. *Predicting Alzheimer's disease: a neuroimaging study with 3D convolutional neural networks*. *arXiv preprint arXiv:1502.02506*, 2015.

[10] S. Liu, S. Liu, W. Cai, S. Pujol, R. Kikinis και D. Feng. *Early diagnosis of Alzheimer's disease with deep learning*. *2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI)*, σελίδες 1015–1018, Beijing, China, 2014. IEEE.

[11] Donghuan Lu, Karteek Popuri, Gavin Weiguang Ding, Rakesh Balachandar και Mirza Faisal Beg. *Multimodal and Multiscale Deep Neural Networks for the Early Diagnosis of Alzheimer's Disease Using Structural MR and FDG-PET Images*. *Scientific Reports*, 8(1):5697, 2018.

[12] Giovanna Castellano, Andrea Esposito, Eufemia Lella, Graziano Montanaro και Gennaro Vessio. *Automated detection of Alzheimer's disease: A multi-modal approach with 3D MRI and amyloid PET*. *Scientific Reports*, 14(5210), 2024.

[13] Sina Fathi, Maryam Ahmadi και Afsaneh Dehnad. *Early diagnosis of Alzheimer's disease based on deep learning: A systematic review*. *Computers in Biology and Medicine*, 146:105634, 2022.

[14] Amitojdeep Singh, Sourya Sengupta και Vasudevan Lakshminarayanan. *Explainable Deep Learning Models in Medical Image Analysis*. *Journal of Imaging*, 6(6):52, 2020.

[15] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[16] Peter Dayan και Laurence F. Abbott. *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*. MIT Press, 2001.

[17] Vaishnavi Patel. *The Differences Between Artificial and Biological Neural Networks*, 2020. Accessed: 21-12-2024.

[18] Nikita Lunge. *A Deep Architecture: Multi-Layer Perceptron*, 2023. Accessed: 21-12-2024.

[19] Shivansh Srivastava. *Understanding the Difference Between ReLU and Sigmoid Activation Functions in Deep Learning*, 2023. Accessed: 21-12-2024.

[20] BotPenguin. *Softmax Function*. https://botpenguin.com/glossary/softmax-function. Accessed: 2024-12-21.

[21] DataCamp. *Introduction to Activation Functions in Neural Networks*. https://www.datacamp.com/tutorial/introduction-to-activation-functions-in-neural-networks. Accessed: 2024-12-21.

[22] IBM. *What is Loss Function?*, 2024. Accessed: 2024-12-21.

[23] Sebastian Ruder. *An overview of gradient descent optimization algorithms*. *arXiv preprint arXiv:1609.04747*, 2016.

[24] Wikipedia contributors. *Convolutional neural network*, n.d. Accessed: 2025-01-03.

[25] Ian Goodfellow, Yoshua Bengio και Aaron Courville. *Deep Learning*. MIT Press, 2016.

[26] Sumit Saha. *A Comprehensive Guide to Convolutional Neural Networks - the ELI5 way*, 2018. Accessed: 2025-01-02.

[27] Wikipedia contributors. *Convolutional Layer*, n.d. Accessed: 2025-01-02.

[28] Chuqi Wang. *A Review on 3D Convolutional Neural Network*. *2023 IEEE 3rd International Conference on Power, Electronics and Computer Applications (ICPECA)*, σελίδες 1204–1208, Shenyang, China, 2023. IEEE. Accessed: 2025-01-02.

[29] Sergey Ioffe και Christian Szegedy. *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*. *arXiv preprint arXiv:1502.03167*, 2015.

[30] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever και Ruslan Salakhutdinov. *Dropout: A simple way to prevent neural networks from overfitting*. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[31] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton και Jeff Dean. *Outrageously large neural networks: The sparsely-gated mixture-of-experts layer*. *arXiv preprint arXiv:1701.06538*, 2017.

[32] Zilliz. *What is Mixture of Experts?* https://zilliz.com/learn/what-is-mixture-of-experts, 2025. Accessed: 2025-01-04.

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser και Illia Polosukhin. *Attention is all you need*. *Advances in neural information processing systems*, σελίδες 5998–6008, 2017.

[34] Julieta Arevalo, Thamar Solorio, Manuel Montes και Fabio A González. *Gated multimodal units for information fusion*. *International Conference on Learning Representations*, 2017.

[35] Wojciech Samek, Thomas Wiegand και Klaus Robert Müller. *Explainable Artificial Intelligence: Understanding, Visualizing and Interpreting Deep Learning Models*. *arXiv preprint arXiv:1708.08296*, 2017.

[36] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh και Dhruv Batra. *Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization*. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, σελίδες 618–626, 2017.

[37] Bernd Bischl, Martin Binder, Michel Lang, Tobias Pielok, Jakob Richter, Stefan Coors, Janek Thomas, Theresa Ullmann, Marc Becker, Anne Laure Boulesteix και others. *Hyperparameter Optimization: Foundations, Algorithms, Best Practices and Open Challenges*. *arXiv preprint arXiv:2107.05847*, 2021.

[38] Weights & Biases. *Weights & Biases Documentation: Guides*. https://docs.wandb.ai/guides/, 2025. Accessed: 2025-01-06.

[39] David M.W. Powers. *Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation*. *Journal of Machine Learning Technologies*, 2(1):37–63, 2011.

[40] Saman Sarraf και Ghassem Tofighi. *Classification of Alzheimer's disease using fMRI data and deep learning convolutional neural networks*. *arXiv preprint arXiv:1603.08631*, 2016.

[41] Yechong Huang, Jian Xu, Yang Zhou, Tong Tong, Xiaoyong Zhuang και Alzheimer's Disease Neuroimaging Initiative. *Diagnosis of Alzheimer's disease via multi-modality 3D convolutional neural network*. *arXiv preprint arXiv:1902.09904*, 2019.

[42] Michal Golovanevsky, Carsten Eickhoff και Ritambhara Singh. *Multimodal attention-based deep learning for Alzheimer's disease diagnosis*. *arXiv preprint arXiv:2206.08826v2*, 2022.

[43] Alzheimer's Disease Neuroimaging Initiative (ADNI). *Alzheimer's Disease Neuroimaging Initiative (ADNI)*. https://adni.loni.usc.edu/, 2025. Accessed: 2025-01-06.

[44] David S. Cohen, Kristy A. Carpenter, Juliet T. Jarrell και Xudong Huang. *Deep learning-based classification of multi-categorical Alzheimer's disease data*. *Current Neurobiology*, 10(3):141–147, 2019.

[45] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai και Soumith Chintala. *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. *Advances in Neural Information Processing Systems*, τόμος 32. Curran Associates, Inc., 2019.

List of Abbreviations

AD	Alzheimer's Disease
MCI	Mild Cognitive Impairment
NC	Normal Control
MRI	Magnetic Resonance Imaging
DNN	Deep Neural Network
VAE	Variational Autoencoders
XAI	Explainable AI
ML	Machine Learning
ANN	Artificial Neural Network
MLP	Multi-Layer Perceptron
ReLU	Rectified Linear Unit
tanh	Hyperbolic Tangen
MSE	Mean Squared Error
CE	Cross-Entropy
BCE	Binary Cross-Entropy
FNN	Feedforward Neural Network
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory
GRU	Gated Recurrent Units
CNN	Convolutional Neural Network
GAN	Generative Adversarial Network
MoE	Mixture of Experts
CAM	Class Activation Mapping
W&B	Weights and Biases
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative
AUC	Area Under Curve
SVM	Support Vector Machine
FN	False Negative
ADRC	Alzheimer's Disease Research Center
RF	Random Forest
ADRC	Alzheimer's Disease Research Center
MDNN	Multiscale Deep Neural Network

GM	Gray Matter
ADNI	Alzheimer's Disease Neuroimaging Initiative
CF	Cerebrospinal Fluid
LONI	Laboratory of Neuro Imaging
BET	Brain Extraction Tool
SS	Skull-Stripped
ACC	Accuracy
SEN	Sensitivity
SPE	Specificity
SD	Standard Deviation