International Journal of Data Science and Analytics
https://doi.org/10.1007/s41060-024-00514-z

REGUALR PAPER

Genetic algorithm-based hybrid deep learning model for explainable Alzheimer's disease prediction using temporal multimodal cognitive data

Hager Saleh[1] · Nora ElRashidy[2] · Mohamed Abd Elaziz[3,4,7,8,9] · Ahmad O. Aseeri[5] · Shaker El-Sappagh[4,6]

Received: 9 August 2023 / Accepted: 24 January 2024

Abstract
Alzheimer's disease (AD) is an irreversible neurodegenerative disease characterized by progressive neuronal deterioration. Early detection of AD is critical for mitigating disease progression and improving patient life. While deep learning (DL) techniques have shown promise in analyzing neuroimaging data for AD diagnosis, their interpretability has been a significant impediment toward explainable medical diagnosis. The typical practices of diagnosing AD, which involve clinical biomarkers and neuroimaging tests, have been limited in producing trustworthy and explainable progression detection models at an early stage. Another method based on finding the patient's cognitive score through analyzing time series data has been adopted as an acceptable and cost-effective alternative in providing a deeper insight into patients' conditions. In this study, we propose a hybrid CNN-LSTM-based model for predicting AD progression based on the fusion of four longitudinal cognitive subscores modalities. Our hybrid model employs the Bayesian optimizer as a computational technique to help optimize the selection of the adequate DL model's architecture. A genetic algorithm-based feature selection has been incorporated as an optimization step to determine the best feature set from the extracted deep representations of the CNN-LSTM, and we replaced the traditional SoftMax classifier with a robust and optimized random forest classifier. An extensive set of experiments utilizing the ADNI dataset examined the operational role of each optimization step while demonstrating the effectiveness of the proposed hybrid model. The model achieved the best results compared to other DL and classical machine learning methods. To ensure diagnostic interpretability, we used the SHAP and LIME techniques to provide explainability features for the proposed model's decisions. This work attempts to present the best possible, trustworthy decision-making AD diagnostics, potentially in deployable real-world settings.

Keywords Alzheimer's disease · Deep learning · Time series data analysis · Explainable AI · hybrid modeling

✉ Ahmad O. Aseeri
a.aseeri@psau.edu.sa

Hager Saleh
Hager.Saleh@fcih.svu.edu.eg

[1] Faculty of Computers and Artificial Intelligence, South Valley University, Hurghada, Egypt

[2] Machine Learning and Information Retrieval Department, Faculty of Artificial Intelligence, Kafrelsheiksh University, Kafrelsheiksh 13518, Egypt

[3] Department of Mathematics, Faculty of Science, Zagazig University, Zagazig 44519, Egypt

[4] Faculty of Computer Science and Engineerings, Galala University, Suez 435611, Egypt

[5] Department of Computer Science, College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Al-Kharj 11942, Saudi Arabia

[6] Information Systems Department, Faculty of Computers and Artificial Intelligence, Benha University, Benha 13518, Egypt

[7] Artificial Intelligence Research Center (AIRC), Ajman University, Ajman 346, UAE

[8] Department of Electrical and Computer Engineering, Lebanese American University, Byblos 13-5053, Lebanon

[9] MEU Research Unit, Middle East University, Amman 11831, Jordan

Published online: 08 March 2024

1 Introduction

Alzheimer's disease (AD) is a progressive neurodegenerative disorder that poses a significant global health challenge. It is not fully understood, researchers approve no clear etiology, and there is no effective drug to treat, prevent, reverse, or stop the progression of AD [1]. The early detection and identification of the disease biomarkers are crucial elements for timely intervention and suitable disease management to slow disease progression, hence reducing its incidence rate and mortality [2–4]. While classical diagnostic methods are often expensive, time-consuming, and invasive, machine learning (ML) has emerged as a promising instrument to enhance AD diagnosis by analyzing various data modalities, including neuroimaging, genetics, and cognitive assessments. ML algorithms can identify subtle patterns and biomarkers in these data that are often missed by human experts, potentially leading to earlier and more accurate diagnoses.

Many knowledge- and data-driven-based solutions have been introduced to tackle the problem of AD diagnostic [4–7]. Most of the AD literature was solely based on neuroimaging modalities, including magnetic resonance imaging (MRI), ignoring patients' clinical and biological information, hence making the progression detection of AD a costly, inaccurate, and medically irrelevant task [3, 5, 6, 8, 9]. In addition, the majority of the literature was based on baseline visit data and neglected the crucial role of longitudinal analysis of AD data [10]. On the other hand, time series data analysis has emerged as a powerful mechanism, offering unique insights into the temporal evolution of AD biomarkers and clinical manifestations. By leveraging longitudinal data collected through various modalities, specifically cognitive assessments and biofluid analysis, time series analysis can identify subtle changes that may precede the onset of clinical symptoms. Time series data analysis has been shown to be effective in predicting the progression of chronic diseases such as AD [2]. As research in this field continues to advance, time series analysis holds immense potential for revolutionizing AD diagnosis and paving the way for more personalized and effective treatment strategies.

There are many cost-effective cognitive scoring techniques (CSs) that clinical experts administer to predict AD, including functional activities questionnaire (FAQ), mini-mental state examination (MMSE), clinical dementia rating—a sum of boxes (CDRSB), Alzheimer's disease assessment scale (ADAS), etc. These techniques provide information about the neuropsychological conditions of the patient and are commonly used to measure the severity of the most critical symptoms of AD (e.g., memory, attention, executive functions, language, and visuospatial abilities). Da et al. [11] investigated the critical role of CSs in detecting and diagnosing AD. Some studies fused imaging modalities with cognitive scores to improve performance [2]. Nonetheless, most literature studies used only the summary values of these cognitive scores (CSs summarize large numbers of sub-scores) while neglecting the importance of analyzing the sub-scores. For example, MMSE has 31 sub-scores with significant medical meanings to provide insights into patients' medical conditions. Clinical sub-scores have a significant role in tracking AD behavior [12]. These sub-scores are longitudinal, and using time series data to detect AD is intuitive and crucial because AD is a chronic disease. The hybrid power of neural network-based methods such as long short-term memory (LSTM) and convolutional neural networks (CNNs) have not been well explored for interpreting and learning time series CS sub-scores yet. To our knowledge, no existing study in the literature has built a hybrid deep learning-based model on the deep representation learning of the fused time series cognitive sub-scores exclusively. This type of analysis provides a deeper understanding of a detailed patient's medical cognitive behavior and can facilitate the discovery of new AD biomarkers that could support more accurate disease understanding and management.

The neural network-based methods are insufficient to get physicians' trust [13].To introduce a model design that considers the model's uncertainty about its outputs, our proposed model incorporates a Bayesian-based approximation method to boost its trustworthiness, interpretability, and robustness. In [14], we surveyed the recent literature on trustworthy AI in the AD domain. Model explainability is a crucial feature for any medical application which allows an ML model to answer the question of "why a specific decision has been taken" [15]. Such a model with this capability is known as an explainable AI (XAI) model [16]. In addition, we surveyed in [17] the critical features needed to build a truly accepted ML model in medical practice, especially in the AD domain. SHAP [18, 19] and LIME [20] are popular explainability techniques to provide local and global explainability features. Here, we explore the role of these techniques to provide a deep insight into DL model decisions to know the most critical features that affect model decisions.

This study proposes a Bayesian-optimized hybrid deep learning model with explainability features to detect subtle patterns in fused multimodal time series CS sub-score feature sets for AD detection in its early stages. To the best of our knowledge, this is the first study that integrates the deep analysis of time series CS sub-scores to provide a detailed and medically understandable investigation of AD behavior. In addition, the genetic algorithm-based feature selection optimization technique is explored for potential improvement of the resulting hybrid DL model to select the optimal list of deep temporal features from the fused time series data. The study has added an extended step in the DL and AD progression detection literature. To sum up, the key original contributions of our work can be summarized as follows:

- We propose an end-to-end hybrid (CNN-LSTM) deep learning model to address the limitations of existing deep models for detecting AD progression. This model predicts the AD progression after one year, i.e., at month 48 (M48) based on the data collected regularly for 24 months (i.e., baseline, M06, M12, M18, and M24).
- The model prediction is based on the sub-scores of four popular and medically relevant longitudinal CSs, including MMSE, FAQ, CDR, and ADAS. Patient demographics are fused with the sub-scores time series modalities. The used CSs biomarkers are easily interpretable and generate robust, medically relevant, verifiable, and reliable predictive models.
- Three optimization phases are used to achieve superior results: (1) Automatic optimization of the proposed DL architecture is done by the Bayesian optimizer. (2) Feature selection optimization is done using a genetic algorithm technique. The optimum features from the deep representations extracted using CNN-LSTM are selected in this step to enhance the performance. (3) Because the default SoftMax classifier used regularly in DL models is not accurate enough compared to other classifiers such as SVM and random forest, we replace the SoftMax with an optimized random forest ensemble model to boost the performance of the resulting hybrid CNN-LSTM model.
- The medical problem has been formulated into three different classification tasks with different complexities (i.e., 2-class, 3-class, and 4-class). Extensive experiments using the ADNI real-world dataset are conducted in a variety of settings to optimize each classification task, and the best models are highlighted. The optimized model is compared with other DL and classical ML models based on statistical features.
- The thorough analysis of the results and comparison to other architectures show that our model outperformed all other models using different evaluation metrics and under different conditions.
- To improve the model's trustworthiness, we extend the proposed robust model by adding extra XAI features. We explore the potential of SHAP and LIME techniques to provide local and global XAI features to the model decisions. The resulting model is robust and explainable, which makes it more suitable for application in the real medical environment.

The remaining parts of the paper are organized as follows. Section 3 is the dataset description. Section 4 is the proposed model. Section 5 is the results and discussion, and the paper is concluded in Sect. 7.

2 Related work

In this section, the literature on AD progression detection is explored by focusing on classical ML and DL models. Several ML and DL techniques have been used to tackle this problem [4, 21–23]. Different data types (modalities) have been used to predict AD such as magnetic resonance imaging (MRI) [24], positron emission tomography (PET) [25], cerebrospinal fluid (CSF) biomarkers [26], and fusion of different neuroimages [27]. Saleem et al. [28] and Weiner et al. [29] deeply surveyed AD different modalities that are used in the literature of ML. The majority of the literature depends mainly on the MRI modality which produced limited results [30, 31].

Using MRI, Rabeh et al. [31] integrated the SVM and decision tree and built a classifier to differentiate AD from MCI. The authors extracted the hippocampus, corpus callosum, and cortex region of interest (ROIs) from MRI images; they used three SVM classifiers, one for each ROI independently to classify subjects, and the final decision was made by combining the results of the three SVMs using a decision tree. Ferreira et al. [4] used SVM and compared the diagnostic accuracy of MRI, PET, and Single-photon emission computed tomography (SPECT) images in detecting AD. Recently, DL got significant attention in the last few years and has been used widely in the AD domain because it achieved better results than classical ML, especially with neuroimaging data [22, 23]. For the recent literature of DL architectures in the AD domain with different datatypes as sMRI, fMRI, PET, DTI, EEG, genetics, CSF, and numerical features, readers are guided to these surveys [14, 22]. Farooq et al. [25] proposed a 2D convolutional neural network classifier based on MRI images to build an AD classifier for detecting the disease. The study used transfer learning on GoogleNet, ResNet-18, and ResNet-152 models. Akter et al., [32] proposed AD-CovNet, a hybrid DL model combining long short-term memory (LSTM) and multilayer perceptron (MLP) to predict the severity of AD in patients with COVID-19. Jain et al. [26] utilized VGG-16 pre-trained on ImageNet for feature extraction to detect AD using MRI images. These models did not achieve accurate results.

As noticed, most of these studies are based on neuroimaging modalities only such as magnetic resonance imaging (MRI) and positron emission tomography (PET) for AD prediction, ignoring patients' clinical and biological information, which makes the process costly, inaccurate, and medically irrelevant [3, 5, 6, 8, 9]. In addition, most studies were based on single time step and neglected the crucial role to longitudinal analysis of AD data [10]. For exam-

ple, Zhang et al. [33] proposed a multimodal DL classifier based on MRI scans, cerebrospinal fluid (CSF) tests, and fluorodeoxyglucose PET(FDG-PET) scans to detect AD. Xu et al. [34] utilized the volumetric MRI, FDG-PET, and florbetapir PET modalities to classify MCI vs AD. Huang et al. [35] combined MRI and CSF modalities to classify normal people from MCI patients. Time series data analysis is important to predict chronic diseases such as AD [2]. Existing literature concentrated on the analysis of time series image data [36]. For instance, Chincarini et al. [37] utilized four MRI volumes from ADNI as input data for each patient to predict AD progression. Saleem et al. [28] reviewed the recent advances of DL architectures in the AD domain.

There is no significant role of AI-based clinical decision support systems in real medical environments [14, 38]. Reasons for this conclusion include the usage of expensive modalities like MRI and PET which achieved lower accuracies compared to other modalities such as cognitive scores [39]. Depending only on neuroimages is expensive and medically insufficient because domain experts usually examine the patients clinically and then check the neuroimages [2]. In addition, neuroimage scanners are not always available in clinical centers, especially in poor countries, and it is not recommended and not preferred by both physicians and patients to expose the patient to radiations of these scanners [40]. Moreover, existing AD studies are not trustworthy enough to be used in real medical environments [14]. CSs including MMSE, ADAS, CDRSB, and FAQ provide an excellent alternative to neuroimages to predict AD progression. Da et al. [11] investigated the critical role of CSs in the detection and diagnosis of AD. Donnelly-Kehoe et al. [41] concluded that the maximum accuracy achieved by MRI features does not reach the one using MMSE alone. Because of their importance, some studies have detected AD by predicting some of these scores as regression tasks [2].

The fusion of imaging modalities with cognitive scores has improved the performance of DL models [2, 8]. Some studies proposed deep learning models by combining neuroimages with few cognitive scores such as MMSE [42], and other studies used classical ML models such as SVM to integrate MRI images with ADAS cognitive test scores to detect AD [3]. In addition, all literature studies just used the summary values for these scores. The total scores summarize the large number of sub-scores. For example, MMSE has 31 sub-scores which have significant medical meanings. Clinical sub-scores have a significant role in tracking AD behavior [12]; Weiner et al. [43] deeply studied the associations between cognitive measures and AD biomarkers. The literature has used these tests to improve ML models' performance to detect AD [30]. Classical ML models such as support vector machine (SVM), random forest (RF), and others have been used to predict AD based on statistical features extracted from the time series data [44]. El-Sappagh et al. [37] built an AD detection classifier based on a random forest algorithm and a fused collection of statistical features collected from 2.5-year time series data including comorbidities, cognitive scores, medication history, and demographics. DL models such as LSTM and CNN which are dedicated to understanding time series data can outperform classical ML models because they can extract deep temporal representations from fused time series data [2, 14]. LSTM [2] has been used to detect AD using different modalities other than CSs. In addition, LSTM has been used to detect AD using a combination of CSs total scores and other modalities like comorbidities [?].

The performance of DL models can be boosted by many techniques including ensembling [36], hyperparameter optimization [45], and model hybridization [46]. El-Sappagh et al. [2] designed a two-stage long short-term memory (LSTM)-based DL model for AD progression detection. The study was based on the early fusion of multimodal time series data. Rahim et al. [36] proposed a deep ensemble model based on the fusion of time series 3D MRI images with patients demographics. Bucholc et al. [46] proposed a two-stage hybrid model based one classical machine learning algorithms. However, hybrid deep learning models improve the performance of AD detection [47], and they have been explored in other domains [48]. Robust hybrid deep learning models have already been successfully applied to AD progression detections [11]. Zhang et al. [49] 3D ResAttNet, CNN with a self-attention residual mechanism, for diagnosis of AD using structural 3D MR images. Vu et al. [50] integrated high-level layer concatenation auto-encoder (HiLCAE) and 3D-VGG-16 to build an AD detection model. The model was pre-trained using the HiLCAE, and the learned weights were used by the 3D-VGG-16 classifier. The input was fused MRI and PET images. El-Sappagh et al. [8] proposed a hybrid CNN-LSTM deep learning model. In this architecture, five modalities were learned with five different CNN-LSTM hybrid models. The resulting models are complex and costly. Balaji et al. [51] proposed a CNN-LSTM hybrid deep learning model to predict AD based on the fusion of MRI, PET, and standard neuropsychological test scores. However, a balance between model complexity, model performance, and model cost should be achieved.

Building a DL classifier based on cognitive sub-scores is expected to be more accurate, transparent, cost-effective, applicable in real environments, and medically relevant. To the best of our knowledge, no study in the literature has built an optimized hybrid deep learning model based on the deep representation learning of fused time series cognitive sub-scores. This type of analysis provides a deeper understanding of a detailed patient's cognitive behavior and can facilitate the discovery of new AD biomarkers that could support more accurate disease detection. In other words, the more in-depth feature mining of patients' clinical information and biolog-

ical information can significantly improve the performance of the resulting DL model [9].

Recent DL models are complex, and it is difficult to justify their decisions. This is not acceptable in life-sensitive domains like medicine. Model trustworthy is crucial to be suitable for its applicability in real environment [13]. Generally, model performance is not sufficient to get the trust of medical experts; in [17], we surveyed the critical features to build a truly accepted ML model in medical practice, especially AD domain. Model explainability provides ML model the ability to answer the question of "why a specific decision has been taken" [15]. Systems with this capability are called explainable AI (XAI) systems [16]. According to European General Data Protection (GDPR) regulations, using black-box models is strictly prohibited in various domains, particularly in healthcare systems [16]. AI-based clinical decision support systems must have a certain level of transparency. Using ML models in a secure, ethical, fair, and trustworthy manner requires implementing embedded XAI features. Recently, we comprehensively reviewed the XAI techniques [52]. Some of the popular XAI techniques are explored in this study to investigate the role of different CS sub-scores in predicting AD.

3 Dataset description

The data utilized for this study were accessed on March 18, 2022, from ADNI 1, ADNI GO, and ADNI 2, as documented in [53]. A total of 1107 participants took part in the study and were classified into four groups based on their clinical diagnosis at both the baseline and future time points. (1) The CN category consisted of 358 individuals who were diagnosed as cognitively normal (CN) at the beginning of the study and remained so throughout the duration of manuscript preparation. (2) The sMCI group comprised 378 subjects who were consistently diagnosed with mild cognitive impairment (MCI) at all time points. (3) The pMCI group included 138 participants who were initially diagnosed with MCI during the baseline visit, subsequently progressing to Alzheimer's disease (AD) within 48 months. (4) The AD group consisted of 233 subjects who carried a clinical diagnosis of AD in every visit.

To ensure the reliability of the study, individuals who demonstrated improvement in their clinical diagnosis over the course of follow-up were excluded. This involved cases where individuals initially diagnosed with MCI reverted back to being cognitively normal, as well as cases where individuals diagnosed with AD later reverted to MCI or CN. Such exclusions were made due to the potential uncertainty associated with clinical misdiagnosis, considering the irreversible nature of AD. Additionally, cases with a direct conversion from CN to AD were also eliminated from the study. The study included eligible participants aged between 54 to 91 years, with a minimum of 4 years of education. They were enrolled in one of three groups: CN, MCI, or AD. At the beginning of the study, CN individuals were free from memory complaints and had an MMSE score ranging from 29 to 30, along with an average CDR score of 0.029. MCI individuals had MMSE scores between 23 to 30 and an average CDR of 1.407. AD patients met the NINCDS-ADRDA diagnostic criteria for probable AD [54], with MMSE scores ranging from 19 to 27 and an average CDR of 4.37.

The collected dataset encompasses two main types of data: time series data and static baseline data. The time series data comprises five readings or time steps for each patient, collected at the baseline and at regular intervals of 6 months up to a duration of two years (i.e., 24 months). As per the ADNI approach, The demographic includes the primary information about each patient's AGE (Age), PTGENDER (Sex), PTEDUCAT (Education), PTETHCAT (Ethnicity), PTRACCAT NA (Race), PTMARRY (Marital), APOE4 NA (number of APOEe4 alleles), FDG (FDG-PET metaROI), ABETA (CSF ABETA), and TAU (CSF TAU). The time series data are divided into five distinct yet complementary modalities: CDR (6 features), MMSE (31 features), FAQ (10 features), and ADAS (13 features). We contend that the optimal combination of these modalities offers a comprehensive understanding of the patient's condition. Furthermore, static baseline data, collected only once during the baseline visit, encompasses demographic information. For further details regarding data acquisition, please refer to the ADNI website [53]. A comprehensive description of the dataset can be found in Table 1.

Table 2 presents the number of categorical and numeric features in each modality, provides insights into the types of features present in each modality, and helps explain why specific modalities may yield better results in our experiments. By examining this table, it becomes evident that modality 1 has a higher proportion of numeric features, MMSE has a more significant number of categorical features, and the CDR module has a whole numeric feature. Demographic data has half of them categorical, and the other half is numeric. These variations in feature composition across modalities can potentially explain the differential performance observed in our experiments.

4 Propose framework

Fig. 1 shows the proposed framework. In this model, we explore different architectures of ML and DL models. The performance of different models is reported, and the best model is empirically selected. In the following subsections, we discuss the models of the proposed architecture.

Table 1 Descriptive statistics for used dataset

	CN (n=358) - 32%	MCI (n= 516)		AD (n=233) - 21%	Combined (n= 1107)
		sMCI (n=378) 34%	pMCI (n=138) - 12%		
Gender (F/M)	191/167	154/224	52/86	104/129	501/606
Age (years)	75.10±4.80	74.77±7.168	73.881±7.003	73.435±7.791	74.643±06.242
Education (years)	16.60±2.870	15.855±3.017	17.005±2.369	15.187±2.929	16.295±02.905
FAQ	0.114±0.730	2.489±3.636	6.169±5.616	16.385±7.042	03.471±06.271
MMSE	29.18±1.024	28.035±2.217	25.515±2.796	21.607±4.287	27.483±03.301
ADAS	7.974±3.760	14.688±6.408	20.935±6.320	32.792±11.095	14.437±10.004
CDR	0.037±0.188	1.229±0.905	2.373±1.3780	05.395±02.590	01.304±02.017

Table 2 The number of categorical and numeric features in each modality

Module name	Num of categorical features	Number of numeric features
ADAS	13	0
MMSE	31	0
CDR	0	6
FAQ	10	0
GDS	15	0
Demographic	5	5

4.1 Data collection and preprocessing

We adopted two preprocessing steps including the handling of the missing values and normalizing the data. In this section, we discuss these methods.

Handling missing values: All patients with more than or equal to 30% missing are removed from the dataset. The missing values of the resulting dataset are handled using the mean for numerical features and mode for categorical features.

Data normalization: All numerical features are normalized using the z-score technique, i.e., $z_j = \left(x_j - \mu_j\right)/\sigma_j$ is the participant's original value for feature j, z_j is the normalized value, μ_j is the feature's mean, and σ_j is the feature's standard deviation. The values of categorical features have been encoded using one hot encoding technique. To prevent the information leakage problem, the fitted operators on the training set are used to directly transform the test set.

4.2 Data splitting and fusion

The fused time series dataset is split into training and testing sets. The training set is used to train and validate the model, and the testing set is used to measure the model's generalizability. The dataset is divided into 70% training and 30% testing. Concurrently, the demographic static dataset is preprocessed and divided into training and testing. In the data fusion step, all training sets are fused, and all testing sets are fused to generate the final training and testing sets. Note that for each patient, each demographic example is repeated n times according to the number of time steps of the sub-score sets. As a result, the fused datasets can be considered as time series data. The resulting training dataset optimizes classical ML models, including RF, KNN, DT, NB, LR, XGB, and LGBM. These classical models are trained using preprocessed statistical features collected from the time series modalities. These statistical features include min, max, and mean for numerical features and first, last, and mode for categorical features. In addition, the resulting fused time series modality is used to optimize different DL architectures, including LSTM, GRU, 1D CNN, and hybrid models of LSTM-CNN and CNN-LSTM using a Bayesian optimizer. Further, we deeply studied enhanced DL architectures by integrating optimized feature selection and robust classifier phases in the proposed DL pipeline.

4.3 Classical machine learning model optimization

To measure the improved performance of the proposed hybrid DL model, we optimized a collection of popular classical and ensemble ML models including RF, KNN, DT, NB, LR, XGB, and LGBM. The results of these models are used as the baseline performance to compare with. The optimization of these models is done using the grid search technique. Different classifiers are optimized for each of the three classification problems (i.e., 2-class, 3-class, and 4-class). Statistical features are extracted from the resulting fused time series data, and we used these statistical features to train these regular models.

4.4 Single deep learning models

Different DL models are applied: 1D CNN, GRU, and LSTM neural.

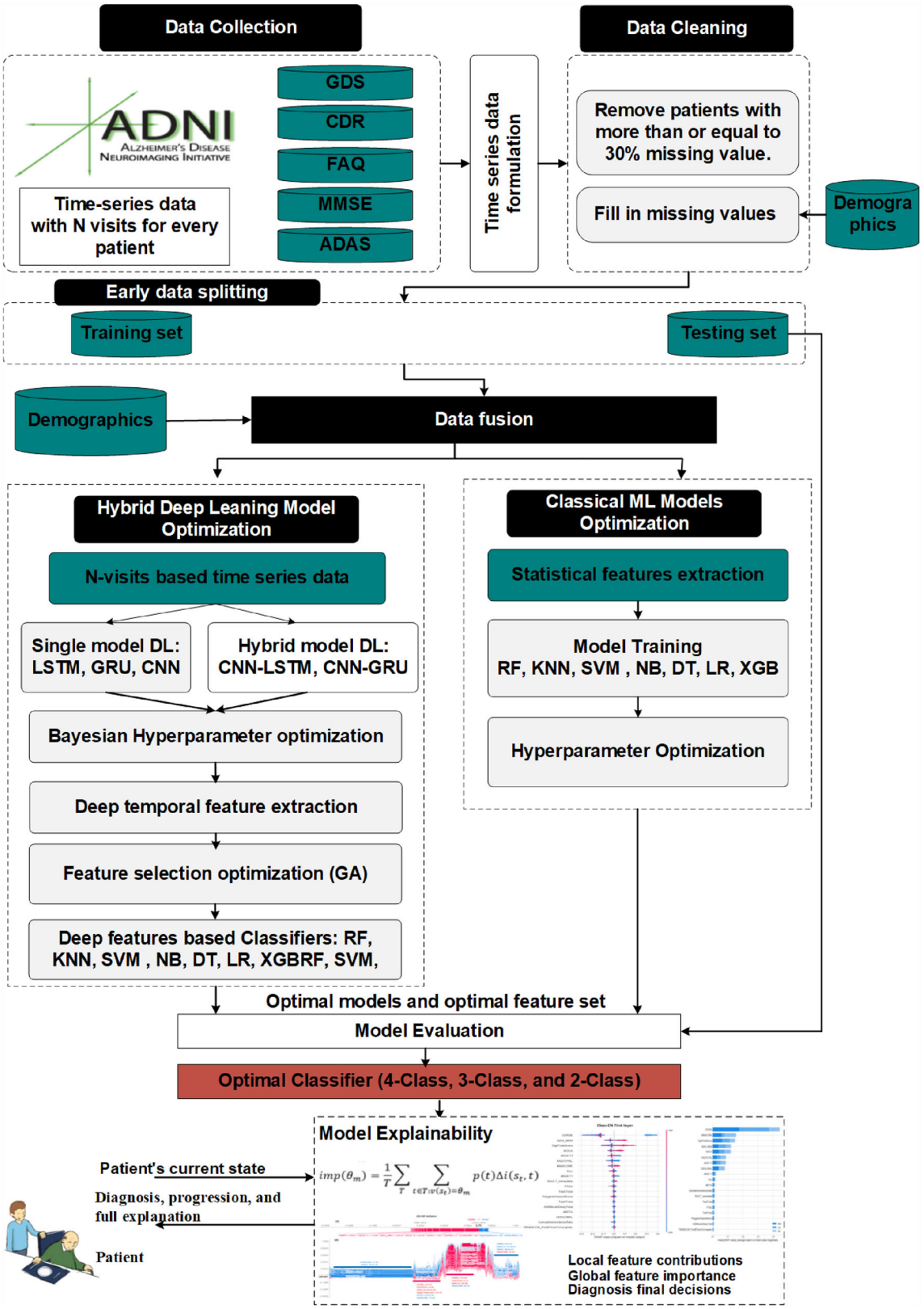

Fig. 1 Proposed hybrid deep learning model

4.4.1 Long short-term memory (LSTM)

LSTM is a variant of recurrent neural network (RNN) for capturing longitudinal and temporal patterns in time series and solves the problem of vanishing and exploding gradients [18]. The operations of an LSTM cell are listed in Eqs. (1)–(7).

$$f_{t_n} = \sigma\left(\theta_f \cdot \left[h_{t_{n-1}}, x_{t_n}\right] + b_f\right) \tag{1}$$
$$i_{t_n} = \sigma\left(\theta_i \cdot \left[h_{t_{n-1}}, x_{t_n}\right] + b_i\right) \tag{2}$$
$$\tilde{C}_{t_n} = \tanh\left(\theta_C \cdot \left[h_{t_{n-1}}, x_{t_n}\right] + b_C\right) \tag{3}$$
$$C_{t_n} = \left(f_{t_n} \otimes C_{t_{n-1}} \oplus i_{t_n} \otimes \tilde{C}_{t_n}\right) \tag{4}$$
$$o_{t_n} = \sigma\left(\theta_o \cdot \left[h_{t_{n-1}}, x_{t_n}\right] + b_o\right) \tag{5}$$
$$h_{t_n} = o_{t_n} \otimes \tanh\left(C_{t_n}\right) \tag{6}$$
$$y_n = \varphi\left(\theta_y h_{t_n} + b_y\right) \tag{7}$$

It has three gates: input gate i_{t_n}, forget gate f_{t_n}, and output gate o_{t_n}. C_{t_n}, $C_{t_{n-1}}$ and $\tilde{C}_{t_n}$ are the current cell status value at any time t_n, last time step cell status value, and the update of the current cell status value, respectively. $h_{t_{n-1}}$ is the output value by each memory cell in the hidden layer at the previous time step. h_{t_n} is the value of the hidden layer at time t_n based on $\tilde{C}_{t_n}$ and $C_{t_{n-1}}$. θs and b s are the set of weight matrices and biases vectors, respectively, updated following the backpropagation through time algorithm. In addition, $\otimes$ is the Hadamard product; σ is the logistic sigmoid function; $\bigoplus$ is the concatenation operator; and φ is the output activation function, e.g., SoftMax. LSTM has been widely used for learning time series data [27, 55, 56], and we used it to diagnose AD and predict its progression [2, 35].

4.4.2 Gated recurrent unit (GRU)

GRU has only two gates (i.e., update gate and reset gate), where it combines the input and forget gates of LSTM into the update gate. This update gate z_t determines the amount of data to retain at the current moment, see Eq. 10. The reset gate r_t controls the amount of forgotten information, see Eqs. 8 and 9.

$$z_t = \sigma\left(W_{zh} \cdot h_{t-1} + W_{zx} \cdot x_t + b_z\right) \tag{8}$$
$$r_t = \sigma\left(W_{rh} \cdot h_{t-1} + W_{rx} \cdot x_t + b_r\right) \tag{9}$$

The input data x_t and the previous time step data $h(t-1)$ are linearly transformed. Different matrices W are right multiplied, respectively. The reset gate r_t and $W_{hh}h_{t-1}$ are multiplied, see Eq. (12). Then, the new data h'_t of the current time is calculated by the tanh activation function.

$$h'_t = \tanh\left(W_{xh} \cdot x_t + r_t \cdot W_{hh} \cdot h_{t-1}\right) \tag{10}$$

Equation (13) shows the product of z_t and $h(t-1)$ which is the final data retained at the previous time step. This product and the data retained from the current memory to the final memory are equal to the content h_t output by the final gated loop unit.

$$h_t = z_t h_{t-1} + (1 - z_t)\, h'_t \tag{11}$$

4.4.3 1D Convolutional neural network

CNN can efficiently extract local spatial features from large quantity of input data. A feed-forward neural network depends on convolution structures to extract local spatial or temporal features from data according to the input dimensions, e.g., n-D image, text, or time series. CNN replaces the dense connections of MLPs with kernel convolutions, as shown in Eq. (12), for kernel K, nonlinear activation function s, and H, W, and C are the height, width, and channel dimensions of the input.

$$f(x)_{m,n} = \sigma\left(\sum_{i}^{H}\sum_{j}^{W}\sum_{c}^{C} x_{i,j,c} K_{m+i-1,n+j-1,c}\right) \tag{12}$$

It is based on weight sharing, local connections, and downsampling dimensionality reduction concepts [34]. If there are m inputs and n outputs, the weight matrix has m×n entities. CNN can reduce the size of the weight matric to k×n by sliding a convolution kernel with size k×k over the input series. Time series CNN introduces 1D convolution (1DConv), which can process univariate time series data. The kernel size is $k \times 1$. Formally, if the input vector x is $l \times 1$ and the kernel r is $k \times 1$, then the 1DConv maps the input shape to $(l-k)/d + 1, 1]$, where d is the

step size. Based on the number of filters, CNN expands every univariate time series to more abstract and informative features called feature maps, which are more suitable for prediction using other DL layers such as LSTM layers. Each value of a feature map is calculated as f_i, see Eq. (13), where g is a nonlinear activation function (e.g., ReLU), b is the bias, and $x^{(i:i+m-1)}$ is m observations from x.

$$\mathrm{f_i} = \mathrm{g}\left(\mathrm{r^T X^{(i:i+m-1)}} + \mathrm{b}\right) \tag{13}$$

If the input size to the CNN is $n \times l \times f$, then by using k filters of $(m \times 1)$ the corresponding output tensor is T, see Eq. (14), where n is the number of samples, l is the number of time steps, and f is the number of features. Pooling layers are used to prevent overfitting and selecting the most important features.

$$\mathrm{T} = \mathrm{n} \times \left(\frac{\mathrm{l} - \mathrm{m}}{\mathrm{d}} + 1\right) \times \mathrm{k} \tag{14}$$

4.5 Bayesian hyperparameter optimization

The optimization of DL hyperparameters always improves the performance of resulting architectures. Hyperparameter optimization (HO) searches for the optimum x^* of f, such that the unknown function $f(x)$ can be evaluated for $x \in X$, and X is a hyperparameter space, i.e., $x^* = \text{argmax}_{x \in X} f(x) \cdot f(x)$ is the ML algorithm objective function, and x is a vector of hyperparameters. Bayesian optimization (BO) is one of the best sequential techniques for efficiently solving this optimization problem [34]. It has a prior assumption for estimating a response surface function $\hat{f}(x)$, utilizes $\hat{f}(x)$ to choose the next setting of x_n to attempt, evaluates the true function $f(x_n)$ using x_n, calculates the posterior belief using the evaluated performance $f(x_n)$, and repeats the process in sequence until a stopping criterion is satisfied, such that the maximum number of iterations is reached (see Algorithm 1). More details on the BO can be found in [34]. In ML, three techniques are widely used for building the response surface of $\hat{f}(x)$ including the tree-structured Parzen estimator, Gaussian process, and RF regressor. These algorithms use acquisition functions that provide a trade-off between exploration and exploitation. Three commonly used acquisition functions are as follows: probability of improvement, expected improvement, and upper confidence bound. In our study, $f(x)$ represents the performance of an ML model with x as the vector of hyperparameters. BO selects a specific set of hyperparameters for the tested DL model and trains this model using the training data. The model then selects another list of hyperparameters based on the exploration and exploitation trade-off. The algorithm continues for a specific number of predefined trails. The best list of hyperparameters is then returned. BO automates the process of searching the hyperparameters space. $f(x)$ is a highly non-convex function with a complex surface. The algorithm of the Bayesian optimizer is shown in 1. For more details about BO, readers are guided to read these references [57, 58].

Algorithm 1 Bayesian optimizer.

Require: Objective function f, BO algorithm $\hat{f}$
parameter space X, N data points, $H \leftarrow \varnothing$
1: **for** $n \leftarrow 1...N$ **do**
Select $x_n \in \text{argmax}_{x \in X} \hat{f}(x, H)$
Evaluate $y_n \leftarrow f(x_n)$
Update $H \leftarrow H \cup (x_n, y_n)$
Check for exit criteria
2: **end for**

4.6 Genetic algorithms

A genetic algorithm is a computational approach to optimizing solutions to intricate problems. Drawing inspiration from the principles of natural selection and evolution, it mimics genetic and evolutionary processes to seek out optimal or near-optimal solutions [59]. The algorithm comprises four fundamental steps, which can be summarized as follows:

- Initialization: The algorithm commences by creating an initial population of potential solutions, represented as individuals or chromosomes. Each individual embodies a potential solution to the problem at hand [60].
- Genetic Operators: By employing genetic operators such as selection, crossover, and mutation, new offspring individuals are generated through the combination and modification of genetic information inherited from existing individuals [60]. Selection determines which individuals are more likely to contribute genetic material to the next generation, while crossover and mutation introduce variation and diversity.
- Fitness Evaluation: As the evolution progresses, individuals in the population are evaluated and assigned a fitness value based on their performance in solving the problem. Higher fitness values indicate better solutions [61]. Individuals with superior fitness are more likely to be selected for reproduction, passing their genetic information to the subsequent generation. Conversely, individuals with lower fitness may be eliminated from the population.
- Iteration and Convergence: Through repeated iterations of selection, crossover, and mutation, the genetic algorithm explores the solution space, gradually improving the quality of solutions across generations [62]. This iterative process allows the algorithm to refine the population and converge toward an optimal or near-optimal solution.

4.7 The proposed model

Two hybrid models are explored, including the combination of LSTM and 1D CNN as feature extractors, i.e., LSTM-CNN and GRU-CNN networks. The hyperparameters of these models are automatically optimized using the BO. The results are deep temporal feature representations that capture the hidden temporal patterns in the fused time series data. We extend the proposed DL model by collecting the deep features, and we use a genetic algorithm to select the best specific neurons from the last layer of the hybrid model. This modification aimed to enhance the overall performance of the DL model by leveraging the strengths of both DL and ML techniques. The resulting DL model is more accurate and less complex. Last, we explore the role of the classifier in the last step of any DL classifier. We replaced the weak SoftMax classifier with ML classifiers, including RF, LR, DT, SVM, KNN, NB, and XGB. We discovered the crucial role of the classifier and feature selection to improve the performance of the DL architecture. The output dense layer y_i of a given input x_i and weight matrix W is defined as $y_i = f(Wx_i + b_i)$, and

the used dropout layer y_i randomly turns off a set of units based on probability p as $y_i = (1 - p) f\ (Wx_i + b_i)$

4.7.1 The pipeline steps of the proposed model

This subsection describes the pipeline steps of the proposed models in detail for the best model obtained for two classes. In the CNN-LSTM phase, different DL architectures have been explored, including LSTM, 1D CNN, and GRU. In addition. The best-achieving model is CNN-LSTM, where the CNN model is used to learn the short-term features in time series, and LSTM is used to learn long-term features. We used the Bayesian optimizer to optimize hyperparameters to select the best hybrid architecture. The hybrid CNN-LSTM deep learning model achieved the best results. The CNN block is one Conv1D layer followed by one max pooling layer. The Conv1D layer works by convolving an input tensor with 3×3 kernel size and 64 filter size, adding biases, and finally passing the resulting feature maps through a nonlinearity based on the ReLU activation function. The convolution is done for each feature separately because there are no valuable patterns related to the order of features as in image analysis. This way of convolution keeps the unique patterns extracted from each feature separate from other features. The convolution layer is often followed by a pooling layer, which compacts features and reduces the number of parameters. We add one max pooling layer of kernel size (3×3) and a stride of 64 after the Conv1D layer. Note that we use the same padding on the convolution layer. The LSTM block is one LSTM, ReLU activation function), L2 regularization, and one dropout layer (probability= 0.5). Each modality has five regular time steps (i.e., baseline, M06, M12, M18, and M24), where patient progression can happen within this period. It predicts the probable patient's progression to AD within the next 18 months.

The model is based on integrating five different modalities of temporal cognitive sub-scores. We are based on the early fusion mechanism of multi-modalities data. CNN first learns data, followed by LSTM and one dense layer. The CNN subnetwork is proposed to extract local features in the time series, and the LSTM subnetwork is applied to learn the temporal relationships among features. We used the Bayesian optimization technique to automatically select the best architecture of the DL model to achieve the best performance and prevent overfitting. After the LSTM block, the output is flattened in a 1D vector and enters a dense layer with 270 units and a ReLU activation function. Several strategies are used to defend against overfitting. We apply regularization [55], dropout layers [56], and early stopping [63]. For regularization, the loss function becomes $j(\theta) + \lambda \cdot \Phi(\mathrm{W})$, where θ is the model's weights and biases parameters, $\Phi(\cdot)$ is the regularizations item or model complexity based mainly on model weights W, and λ is the ratio of the cost of the model complexity and total cost. We used $\mathrm{L}_2 = \Phi(\mathrm{W}) = \sum_i w_i^2$ regularization, which penalizes Ws of high absolute values without making any weights zeroes. When some weights are very small, their corresponding square values become small enough to be ignored. The dropout works by randomly dropping units and their connections during training. This technique helps the network to learn only robust features, prevents complex co-adaptations on training data, reduces the amount of computation, and improves the model's generalizability. Early stopping is a technique to terminate the optimization algorithm before overfitting and to determine the suitable number of epochs.

4.8 Performance evaluation

The resulting ML and DL models are compared and evaluated using different evaluation metrics, as shown in Eqs. (15)-(18), and used to optimize three different classification tasks with different complexity levels (i.e., 4-class, 3-class, and 2-class) for AD prediction.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}. \tag{15}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{16}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{17}$$

$$\text{F} - \text{score} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{18}$$

These are the most used in the medical informatics literature to increase the possibilities of results comparison where TP is the true positive, TN is the true negative, FP is the false positive, and FN is the false negative.

4.9 Model explainability

The medical domain is sensitive because it directly affects human lives. Medical experts do not trust the decisions of ML models based on their performance alone. Model explainability is a set of techniques that allow medical experts to know why an ML model takes specific decisions, and so the expert can confirm if the model is working using the right logic or not, see Fig. 2. XAI methods are globally divided into post hoc and intrinsic methods. Post hoc methods explain the decisions of the black-box models, and intrinsic methods are interpretable by nature, like decision trees, rule-based systems, case-based reasoning, and logistic regression. The resulting trustworthy models gain the trust of the user [14]. A comprehensive list of XAI techniques can be found in our recent and other references [52, 64, 65]. Here, we extend the best ML models by adding XAI features based on the SHAP and LIME techniques [20, 52].

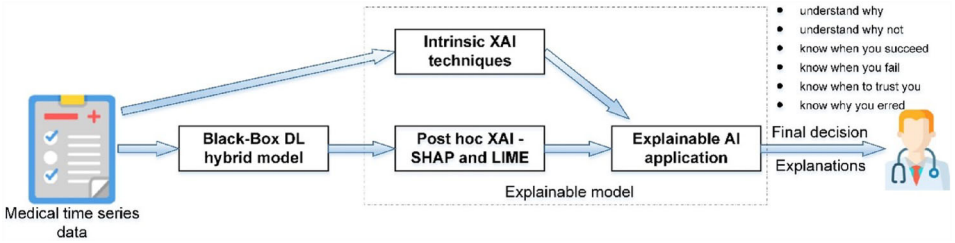

Fig. 2 Role of XAI features to improve the results of DL models

4.9.1 SHAP

The SHAP model-agnostic model is used to provide local and global explainability [18, 19]. It linearly combines binary variables $g\left(x'\right) = \emptyset_0 + \sum_{j=1}^{M} \emptyset_j x'_j$, where x'_j is a mapping to x (i.e., the original input) by using $\mid x = h_x\left(x'\right), x' \in \{0,1\}^M$ is the coalition vector, and 1 indicates that features in new data are the same as those of the original x, and 0 indicates that features of the created data are different from the original x, M is the number of features, and $\emptyset_j \in \mathbb{R}$ is the Shapely value for the average attribution of feature j to x. SHAP tries to ensure that:

$$\emptyset_j = \sum_{S \in \{x_1,\ldots,x_M\} \setminus \{x_j\}} \frac{|S|!(M-|S|-1)!}{M!} \left(f\left(S \cup \{x_j\}\right) - f(S)\right) \tag{19}$$

where S has positive indexes in x' and is a subset of the set of features, x' is the vector of values for the needed to be explained instance, $(|S|!(M-|S|-1)!)/M!$ is a weighting factor, and $f(S) = E\left[f(x) \mid x_S\right]$ is the f value with features in S that are marginalized over features that are not included in S.

4.9.2 LIME

Locally Interpretable Model Agnostic Explanations (LIMEs) are a model-agnostic post hoc XAI method that explains individual models by approximating a black-box model with another local and interpretable model [20]. LIME can be used to explain any classifier, and it provides local explanations by fitting a local model (e.g., interpretable models such as decision trees) using sample data points that are like the example x being explained, see Eq. (19).

$$\emptyset(x) = \underset{g \in G}{\operatorname{argmin}} L\left(f, g, \pi_x\right) + \Omega(g) \tag{20}$$

where G is the group of interpretable models like decision trees, $g \in G$ is the explanation model, $f : \mathbb{R}^d \rightarrow \mathbb{R}$, $\pi_x(z)$ is the proximity measure of an instance z from x, and $\Omega(g)$ is the measure of complexity of the explanation $g \in G$. The objective function is the local minimization of L without any assumptions about f because LIME is model-agnostic.

5 Results

In this section, we test our hypotheses that (1) the hybrid CNN-LSTM could achieve better results than single DL model such as 1D CNN, GRU, or LSTM, (2) neural architecture search based on BO could improve the performance of the resulting DL models, (3) the fusion of multimodal time series data (i.e., ADAS, CDR, MMSE, GDS, and FAQ) could enhance the performance compared to single modalities, (4) integration of better classifiers (e.g., DT, LR, KNN, SVM, KNN, NB, RF, and XGB) with deep feature extraction could achieve better results compared to the default SoftMax, and (4) adding a feature selection phase after the deep representation learning could enhance the performance. Please note that the previous hypotheses are evaluated in the different problem settings with different complexity levels including the four-class (AD, CN, pMCI, and sMCI), the three-class (AD, CN, and sMCI), and the two-class (AD and CN) classification tasks. In the three-class classification task (AD, CN, and sMCI), we combine the instance of pMCI with the AD instance. In the two-class classification task (AD and CN), we combine the instance of pMCI with the AD instance and the instance of sMCI with CN. The main idea is to explore the role of ML and DL in diagnosing Alzheimer's with different modalities and explore the role of data fusion in improving overall performance. Data fusion is done by combining different modalities to achieve the best performance. To enhance the overall performance, we develop a hybrid model combining CNN with LSTM models on every modality and then on the fused data. We also apply the genetic algorithm to select the best number of neurons from the last layer of the CNN-LSTM model before the output layer and then replace the SoftMax activation function in the output layer with the different ML models. DL models are optimized using BO. Also, all models were evaluated with various evaluation metrics, including (accuracy, precision, recall, and F1-score) The experiments of all models were presented as mean ± standard deviation (SD). All models evaluated through various evaluation metrics include accuracy, precision, recall, and F-measure.

Table 3 The GA parameters

Parameters	Values
Population size	120
Mutation rate	0.01
Crossover rate	80%
Selection method	Roulette wheel selection
Termination criteria	Max_generation = 9

5.1 Experiment setup

Throughout this paper, the experiments were conducted on a machine equipped with an 11th Gen Intel(R) Core(TM) i7-1165G7 processor running at a clock speed of 2.80 GHz, with 16.0 GB of RAM. The experiments were performed using Python 3.7.3 distributed in Anaconda 4.7.7 (64-bit). The proposed models were implemented using the Keras library with TensorFlow as the backend. For the classification task, a SoftMax activation function with cross-entropy loss was utilized. The Adam optimizer was employed with a fixed learning rate of 0.0001, while other parameters were kept at their default values, as mentioned in [66]. To ensure consistency, the training batch size and number of epochs were set to 50 and 70, respectively, for all experiments. The model training process was parallelized across the GPUs to enhance training speed. To mitigate overfitting, dropout, L2 regularization, and early stopping mechanisms were employed. The network settings, including the dropout rate and the L2 regularization penalty coefficient, remained unchanged for all experiments to ensure a fair comparison. A total of 1107 cases from the dataset were used for the experiments. The dataset was randomly divided into training, validation, and testing sets, with proportions of 65%, 10%, and 25%, respectively, resulting in 747, 83, and 277 cases for each set.

Our study employs the genetic algorithm as a feature selection technique to select the best number of neurons from the last layer of the CNN-LSTM model before the output layer and replace the SoftMax activation function in the output layer with the ML model. This modification aimed to enhance the overall performance of the proposed DL model. Table 3 shows the default hyperparameters of the used GA. Regarding optimizing hyperparameters in the DL models for each classification task, we employ the Bayesian optimizer to search for the best set of hyperparameters. The best list of hyperparameters is identified through this optimization process and is presented in Table 4.

5.2 Statistical analysis

In this study, we explore the role of cognitive scores in predicting AD progression. Total cognitive scores have a significant relation to the progression of the disease. In this section, we explore the statistical significance of these scores, including FAQ, MMSE, ADAS, and CDR. We used the non-parametric statistical test of the Kruskal–Wallis H test in this statistical analysis. Statistical significance $\alpha = 0.05$. Figure 3 shows these scores let us discriminate between the four classes at each visit. The scores have statistically significant differences among different classes. In Fig. 3, CDR, MMSE, FAQ, and ADAS have statistically significant differences among the four classes (P value< 0.0001). CDR has large values for AD patients, and the values become lower as the patient is more normal. MMSE has inverse behavior, where CN patients have high MMSE, and AD patients have lower values. FAQ and ADAS scores are high in AD patients and small in CN patients. Note that there are significant differences among classes for all scores. It can be seen clearly that these scores can play a critical role in differentiating the AD classes in our classification problem.

More interestingly, the longitudinal analysis of these CSs for every class shows the progression of the score with time. *Regarding the ADAS*, it can be noticed in Fig. 4 that AD class has statistically significant differences among time steps from baseline visit to M24 visit (P value < 0.001). The pMCI cases follow the same pattern, where the patient's conditions start with high ADAS values at baseline and become severe with time. On the other hand, the CN patients have small ADAS values at the baseline visit, and these values have not significantly changed with time. This means that the CN patient has stable conditions. For sMCI patients, they have high ADAS values at the baseline visit compared to CN patients, but these values have not significantly changed with time as in the pMCI cases. As a result, the temporal data of ADAS score can track the conditions of the patient.

Regarding CDR score, it can be noticed in Fig. 5 the significant difference among classes. In addition, the score has clear differences among time steps for both AD and pMCI classes (P value< 0.001). Baseline CDR for AD class is the largest value, pMCI is the second, and sMCI is the third. CN patients have the smallest CDR values and these values have not changed significantly with time. sMCI cases start with lower CDR compared to pMCI, but the values remain stable with time. It can be noticed that DL model can learn these temporal patterns and differentiate different classes.

Regarding FAQ score, CN patients have small FAQ scores, and the score value increases as the patient conditions become more severe. Figure 6 shows the temporal progress of the score for every class in different time steps. AD patients have a high FAQ at baseline and this score increases significantly over time (P value < 0.001). pMCI patients start with lower values compared to AD, but the score increases significantly with time. This medically makes sense because pMCI patients become AD after a period. sMCI patients have lower FAQ scores compared to pMCI, but the score remains

Table 4 The best hyperparameters for the utilized model

Models	Parameters	Two classes	Three classes	Four classes
CNN	Filter_size	64	64	64
	kernel_size	3	4	3
	reg_rate	0.0001	0.0005	0.0004
	number of units	500	600	650
LSTM	number of units	270	300	350
	dropout	0.5	0.5	0.3
RF	Max_depth	3	4	5
	min_samples_split	2	2	2
	bootstrap	True	True	True
	criterion	gini	gini	gini

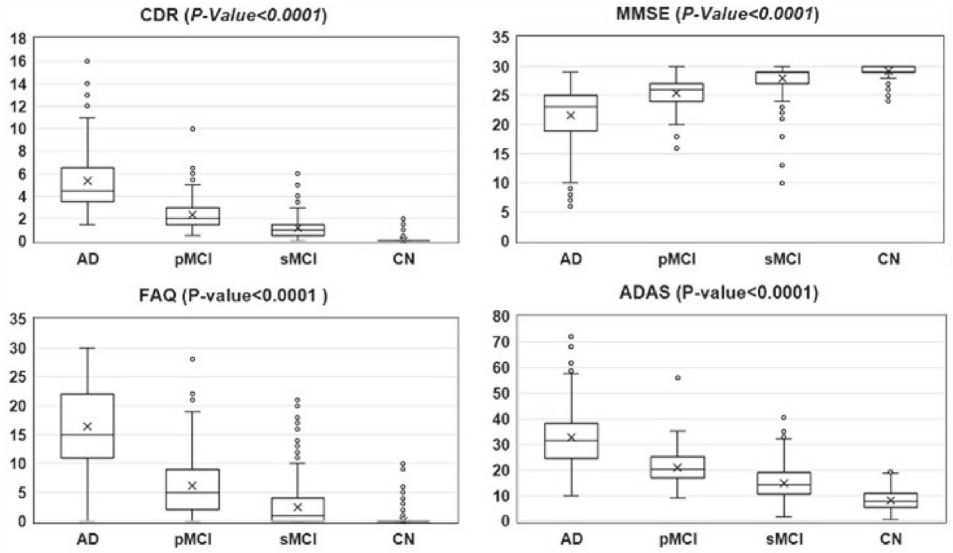

Fig. 3 Statistical significance of selected cognitive scores

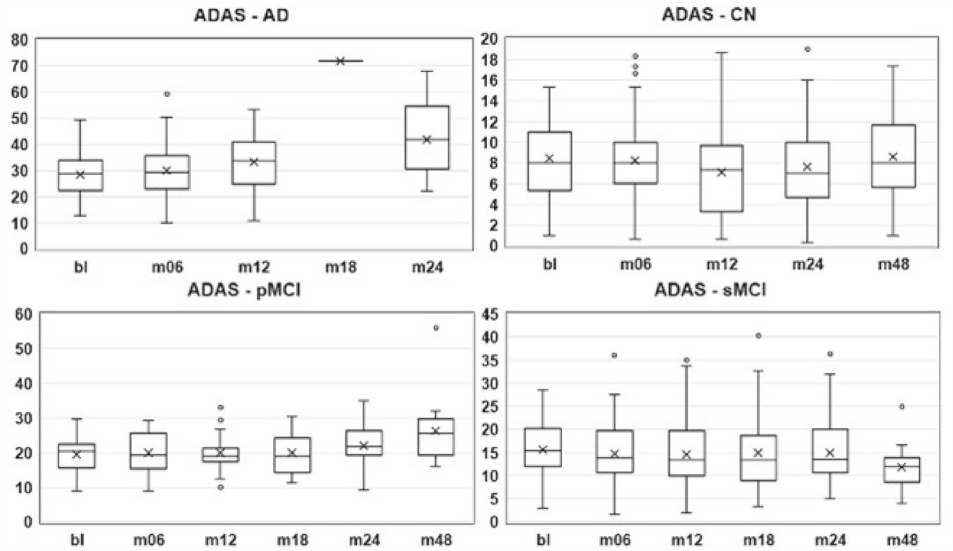

Fig. 4 Statistical significance of longitudinal ADAS score

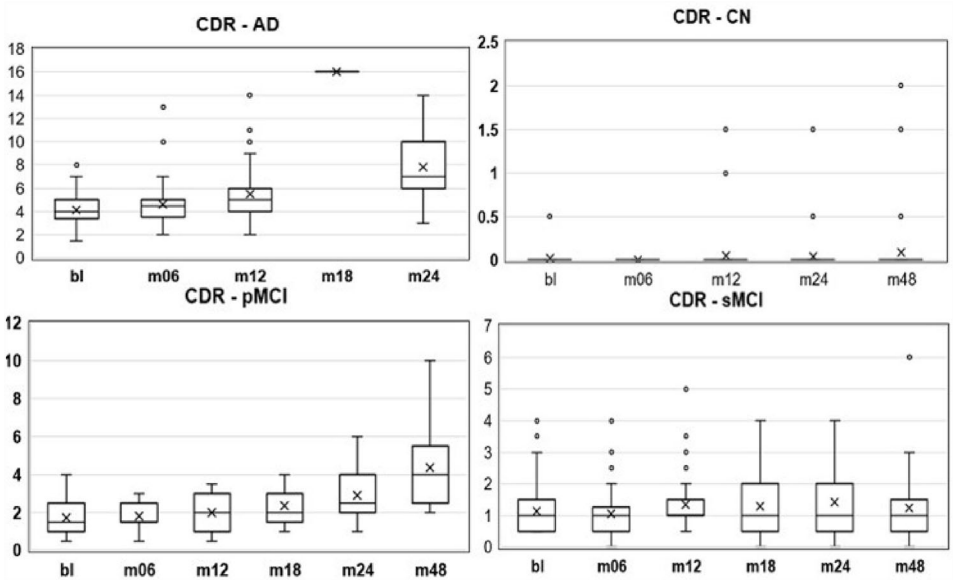

Fig. 5 Statistical significance of longitudinal CDR score

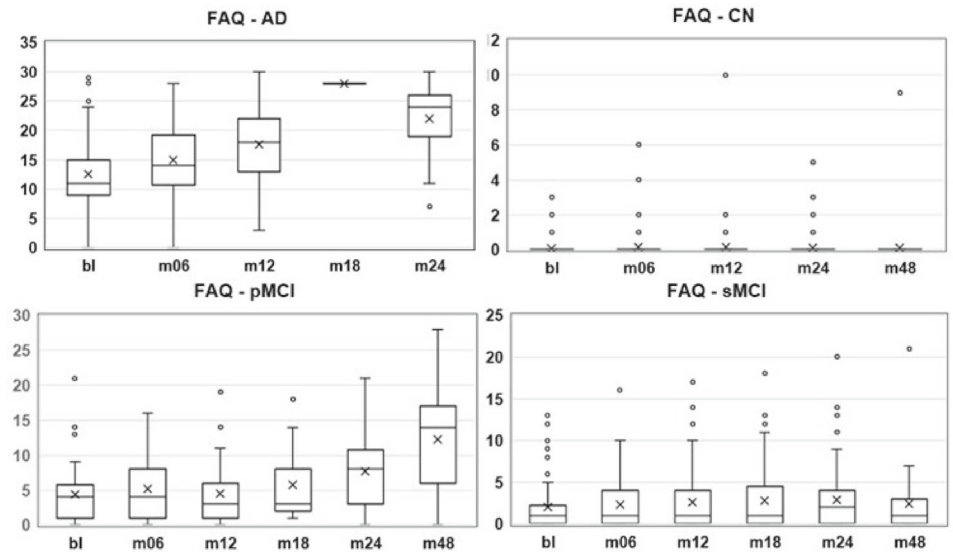

Fig. 6 Statistical significance of longitudinal FAQ score

stable over time because the sMCI patient has stable conditions compared to AD and pMCI. CN patients have the lowest FAQ scores, and these values are stable in all time steps.

Regarding the MMSE score, lower MMSE means severe AD conditions. It can be noticed in Fig. 7 that AD patients have low MMSE values at baseline visits, and these values are significantly reduced over time (P value < 0.001). For pMCI patients, the MMSE score is high at baseline and becomes severe with time as well. On the other hand, for CN and sMCI patients, the MMSE score remains stable over time. These temporal patterns are critical to optimizing the deep learning model. Note that the above discussion is for the global CSs, but in the current study, we further investigate the sub-scores of these features. This detailed analysis has achieved better results because each of these scores has been represented in much more detail using a collection of sub-scores.

5.3 Performance of four-class (CN, sMCI, pMCI, and AD) classification

In this section, we conduct several experiments to choose the best classifier that performs best with four classes (AD, CN, pMC, and sMCI) with both single and multi-modality.

Experiment 1: This experiment evaluates six promising ML classifiers with optimized hyperparameters in diagnosing Alzheimer's with four classes. Table 5 lists the classifiers' average performance. From this table, we can make the following observations: (1) For the ADAS modality, KNN and NB give the least performance in cross-validation (CV) and testing results. The best performance was obtained from RF with a CV accuracy of 71.33% and testing results of 68.76%. (2) Diagnosis based on CDR performs better than other modalities. Utilizing KNN and DT gives the lowest

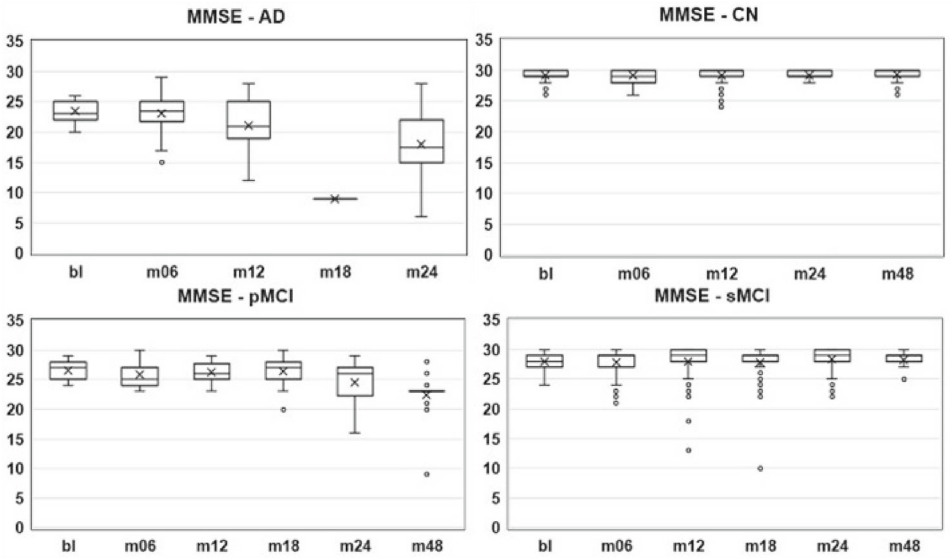

Fig. 7 Statistical significance of longitudinal MMSE score

performance, followed by NB. RF and LR perform similarly in CV, while LR gives a more stable performance in testing results. XGB outscored all other algorithms with a CV accuracy of 82.41% and testing accuracy of 80%. (3) For the FAQ modality, KNN achieved the worst performance in terms of CV results (ACC= 62%). NB and SVM performed similarly, but SVM is considered more stable (SD= 3%). RF (ACC= 68.59%) obtained the best training performance for CV. (4) Diagnosing based on MMSE and GDSC gives similar results to most ML algorithms. The best performance for MMSE was obtained from RF, which achieved a CV accuracy of 68.92% and 65.06% for GDSC. Overall, the experiment shows that ML classifiers can achieve promising results in diagnosing Alzheimer's disease with different modalities.

Experiment 2: In addition to ML algorithms, we explore the performance of various single DL models, including LSTM, GRU, and CNN, as well as hybrid DL models, including CNN-GRU and CNN-LSTM. As presented in Table 6, utilizing DL models improves the overall performance for single and combined modalities. From Table 6, we make the following observations: (1) All DL models achieved enhanced performance that outscored the ML models for the ADAS modality. LSTM achieved an accuracy of 71.81% in CV, while GRU and CNN achieved enhanced performance with an improvement of about 3-7% with all evaluation metrics. Building a hybrid model of LSTM and CNN achieved promising performance that outperformed all single DL models with an accuracy of 78.77% in CV and 73.65% in testing. (2) The CDR modality achieved promising performance with DL, outperforming ADAS by about 8% with increased model stability. The best performance was obtained from CNN-LSTM with an accuracy of 87.62% in CV and 82.31% in testing. (3) The FAQ and MMSE modalities gave adequate performance for diagnosis. CNN-LSTM achieved an accuracy of 79.92% and 77.79% in CV for MMSE and FAQ, respectively. (4) Regarding multi-modality, utilizing full features increases the overall performance with an improvement of about 3-6 for all algorithms. For example, the CNN model achieved an enhanced performance of 88.73% in terms of classification accuracy, while CNN-LSTM was able to predict diagnosis with a promising performance accuracy of 90.48%. (5) Utilizing the genetic algorithm (GA) to choose the optimum features from the DL model and then applying ML models to choose the optimum features enhances the overall performance. The best performance was obtained from the LR model, with an accuracy of 91.49% in CV and 86.60% in testing.

Figure 8 summarizes the results by highlighting the best testing performance across all modalities and for four classes. Regarding the four-class ML models shown in Fig. 8a, the ADAS modality achieved the best performance with an accuracy of 68.95. In contrast, the CDR modality demonstrated higher performance at 80.51. The FAQ modality achieved adequate performance with an accuracy of 65.95, and the MMSE modality achieved moderate performance with an accuracy of 60.95. The GDSCALE modality exhibited lower performance, with an accuracy of 55.12. When combining all features, the XGB model demonstrated the highest performance with an accuracy of 82.39. In the DL four-class scenarios shown in Fig. 8b, the CNN-LSTM model achieved the best performance. The ADAS, CDR, MMSE, and FAQ modalities also demonstrated high performance. The GDSCALE modality exhibited relatively lower performance. Applying the full features with the CNN-LSTM model demonstrated high performance, and optimizing features using GA further improved accuracy. These results highlight the varying performance of the different models

Table 5 The performance of ML models with the four-class task

Models	Features	Cross-validation performance				Testing performance			
		Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
ADAS	RF	71.33±2.79	71.52±2.84	71.33±2.79	70.56±2.82	68.95	67.83	68.95	67.80
	LR	71.81±5.03	72.28±5.24	71.81±5.03	71.29±5.06	68.76	68.60	68.76	68.18
	DT	61.93±6.67	62.54±6.39	61.93±6.67	61.76±6.53	60.29	60.14	60.29	60.14
	SVM	70.36±4.7	71.43±4.13	70.36±4.7	69.58±4.62	67.87	66.84	67.87	66.65
	KNN	62.41±4.34	62.13±5.51	62.41±4.34	60.66±4.43	59.65	58.52	58.65	58.33
	NB	61.08±4.18	64.64±5.33	61.08±4.18	60.14±4.33	59.18	59.47	59.18	59.06
	XGB	69.88±4.41	69.84±4.58	69.88±4.41	69.43±4.4	65.59	65.42	65.59	65.41
CDR	RF	81.81±3.34	80.24±3.92	81.81±3.34	80.11±3.28	78.31	78.20	78.31	78.36
	LR	81.93±3.77	78.82±5.34	81.93±3.77	79.43±4.43	78.23	79.50	78.23	78.81
	DT	76.39±5.17	77.21±3.88	76.39±5.17	76.41±4.57	73.45	73.00	73.45	73.64
	SVM	82.29±3.66	80.52±4.76	82.29±3.66	80.55±4.28	78.39	78.41	78.39	78.97
	KNN	73.37±2.25	70.0±3.08	73.37±2.25	70.81±2.42	70.04	68.45	70.04	68.07
	NB	79.16±4.57	82.75±3.4	79.16±4.57	80.12±4.3	75.51	75.32	75.51	75.10
	XGB	82.41±5.47	82.11±5.93	82.41±5.47	81.93±5.65	80.51	79.95	80.51	80.17
FAQ	RF	67.71±4.72	68.0±3.76	67.71±4.72	67.09±4.35	65.87	65.64	65.87	65.61
	LR	66.14±4.92	66.37±4.19	66.14±4.92	65.13±4.79	65.15	64.32	64.15	64.40
	DT	56.99±6.1	57.6±5.37	56.99±6.1	56.88±5.76	53.43	53.50	53.43	53.44
	SVM	67.83±3.14	67.99±2.97	67.83±3.14	66.55±3.33	65.95	65.12	65.95	65.93
	KNN	62.17±3.78	63.39±6.22	62.17±3.78	59.08±4.14	57.09	57.51	57.09	57.90
	NB	65.9±3.66	66.49±4.44	65.9±3.66	64.01±3.86	63.90	64.18	63.90	62.22
	XGB	64.82±4.02	64.75±3.21	64.82±4.02	64.25±3.71	63.18	62.29	63.18	62.5
MMSE	RF	65.54±5.15	66.28±5.45	65.54±5.15	64.8±5.27	60.95	60.10	60.95	60.63
	LR	65.06±5.36	64.8±6.58	65.06±5.36	64.03±5.89	60.70	60.50	60.70	60.55
	DT	57.47±5.11	57.78±4.86	57.47±5.11	57.28±5.09	54.87	55.98	54.87	55.33
	SVM	64.34±4.15	64.3±4.49	64.34±4.15	63.22±4.34	60.54	60.15	60.54	60.35
	KNN	54.7±4.25	55.33±4.56	54.7±4.25	53.39±4.28	53.79	54.13	53.79	50.90
	NB	48.19±4.6	40.29±13.83	48.19±4.6	38.74±4.17	44.04	53.38	44.04	42.39
	XGB	61.81±2.9	62.11±3.21	61.81±2.9	61.29±3.08	59.29	59.06	59.29	59.47
GDSCALE	RF	58.92±4.05	57.72±4.09	58.92±4.05	57.33±3.89	55.12	55.26	55.12	55.76
	LR	58.67±4.82	57.13±5.34	58.67±4.82	56.75±4.9	53.43	52.89	53.43	52.35
	DT	46.02±7.5	46.3±7.0	46.02±7.5	45.9±7.18	44.04	43.61	44.04	43.70
	SVM	57.11±3.66	55.5±3.45	57.11±3.66	55.11±3.63	46.57	54.29	46.57	43.01
	KNN	49.88±3.97	46.56±5.57	49.88±3.97	46.76±3.94	40.43	36.39	40.43	35.88
	NB	49.28±2.87	49.91±3.17	49.28±2.87	47.48±3.26	38.63	39.03	38.63	39.79
	XGB	56.51±3.07	55.29±2.92	56.51±3.07	55.5±2.92	50.15	50.70	50.15	50.87
Full features	RF	85.9±3.0	85.82±3.42	85.9±3.0	85.25±3.63	81.56	81.12	81.56	81.70
	LR	82.41±3.5	82.46±3.44	82.41±3.5	82.0±3.76	78.70	77.75	78.70	78.01
	DT	79.52±4.54	79.59±5.23	79.52±4.54	79.17±4.87	76.17	76.00	76.17	76.01
	SVM	80.48±3.73	80.55±3.4	80.48±3.73	80.13±3.77	77.98	77.87	77.98	77.81
	KNN	70.0±4.46	70.27±4.71	70.0±4.46	68.2±4.69	66.43	66.48	66.43	64.31
	NB	76.51±3.78	79.04±3.98	76.51±3.78	76.91±3.84	63.90	69.61	63.90	62.85
	XGB	86.87±3.55	86.8±4.28	86.87±3.55	86.52±4.01	82.39	82.61	82.39	82.91

Table 6 The performance of DL for the four-class task

Models	Features	Cross-validation performance				Testing performance			
		Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
ADAS-DEM	LSTM	71.81±2.29	75.11±2.23	66.24±1.39	70.39±1.57	69.31	69.71	69.31	68.77
	GRU	73.73±1.05	74.54±0.66	63.38±1.94	68.48±1.33	72.92	72.68	72.92	72.77
	CNN	77.44±1.55	82.38±1.17	72.08±1.13	76.82±0.9	70.69	69.62	70.69	69.0
	CNN-LSTM	78.77±1.99	79.53±1.94	77.29±2.1	78.38±1.79	73.65	73.32	73.65	73.37
	CNN-GRU	72.99±1.45	76.92±0.95	74.23±1.03	76.92±0.79	72.2	71.35	72.2	71.49
CDR- DEM	LSTM	86.17±1.2	91.24±1.19	83.51±1.15	87.13±0.96	81.83	82.53	81.83	81.84
	GRU	82.45±1.38	85.53±1.37	79.24±1.67	82.26±1.35	81.95	81.2	81.95	81.41
	CNN	86.97±0.97	87.31±0.75	82.65±1.02	84.53±0.78	80.39	78.6	80.39	77.93
	CNN-LSTM	87.62±0.8	91.38±0.84	82.38±1.23	86.59±0.85	82.31	82.35	82.31	82.29
	CNN-GRU	88.88±1.02	93.42±1.18	84.83±1.41	88.84±0.88	82.67	81.72	82.67	81.95
FAQ- DEM	LSTM	76.33±1.79	79.25±1.87	73.04±1.87	75.96±1.55	63.3	61.61	63.3	61.76
	GRU	69.66±1.1	77.89±0.93	64.22±0.89	70.34±0.67	62.57	59.92	62.57	60.31
	CNN	72.71±1.54	75.1±1.3	65.84±1.51	70.16±1.29	63.54	62.56	63.54	62.83
	CNN-LSTM	77.79±1.06	81.51±1.07	70.97±0.85	75.79±0.79	67.69	66.48	67.69	66.47
	CNN-GRU	71.02±2.13	74.78±2.04	76.87±2.23	74.5±1.88	64.02	63.03	64.02	63.0
MMSE-DEM	LSTM	74.03±2.1	76.05±2.3	70.28±2.03	73.04±1.99	65.1	72.76	65.1	66.03
	GRU	74.42±1.67	77.71±1.71	71.32±1.92	74.36±1.62	71.12	71.34	71.12	70.19
	CNN	76.26±1.8	78.61±1.82	73.37±1.9	75.88±1.66	70.16	69.84	70.16	69.41
	CNN-LSTM	79.92±1.93	81.83±1.67	75.74±2.01	78.64±1.6	73.29	73.79	73.29	72.86
	CNN-GRU	77.23±2.05	80.0±2.15	71.07±2.45	75.2±2.05	69.82	69.91	69.82	68.84
GDSCALE	LSTM	68.05±1.19	74.96±3.32	55.18±3.75	63.54±3.45	58.0	56.78	58.0	56.81
	GRU	63.98±1.85	70.96±2.62	51.94±1.58	59.9±1.54	60.05	58.91	60.05	58.63
	CNN	71.9±2.59	76.31±3.88	61.4±2.37	67.92±2.48	60.65	57.72	60.65	57.82
	CNN-LSTM	73.09±2.54	79.86±2.63	64.86±1.91	71.5±1.9	62.82	61.83	62.82	60.91
	CNN-GRU	2.98±1.07	73.62±1.63	45.42±1.41	56.04±1.15	60.51	58.66	60.51	58.38
Full features	LSTM	88.57±3.1	89.02±2.95	88.17±3.03	88.59±2.97	77.14	77.0	77.14	77.74
	GRU	89.84±0.77	90.69±1.09	88.97±0.8	89.81±0.74	83.08	83.69	83.08	83.21
	CNN	88.73±1.92	89.35±1.76	84.46±2.9	86.82±2.17	77.98	79.3	77.98	77.88
	CNN-LSTM	90.48±2.5	90.48±2.66	90.48±2.5	90.08±2.57	84.56	84.6	84.56	84.48
	CNN-GRU	87.87±1.32	88.68±1.57	87.35±1.55	88.0±1.43	81.95	82.78	81.95	82.02
Optimized features GA	RF	90.0±3.45	90.03±3.55	90.0±3.45	89.68±3.46	85.56	85.63	85.56	85.56
	LR	91.49±1.36	91.65±1.25	91.1±1.4	91.37±1.29	86.60	87.56	86.60	86.13
	DT	88.55±3.78	89.16±3.44	88.55±3.78	88.46±3.7	83.39	83.97	83.39	83.61
	SVM	90.6±2.4	90.55±2.55	90.6±2.4	90.18±2.45	84.48	84.44	84.48	84.43
	KNN	89.4±3.18	89.92±3.24	89.4±3.18	89.44±3.1	84.48	85.17	84.48	84.57
	NB	86.14±4.01	87.38±3.65	86.14±4.01	86.42±3.82	84.48	85.70	84.48	84.52
	XGB	89.64±3.62	89.74±3.46	89.64±3.62	89.37±3.53	85.56	85.63	85.56	85.56

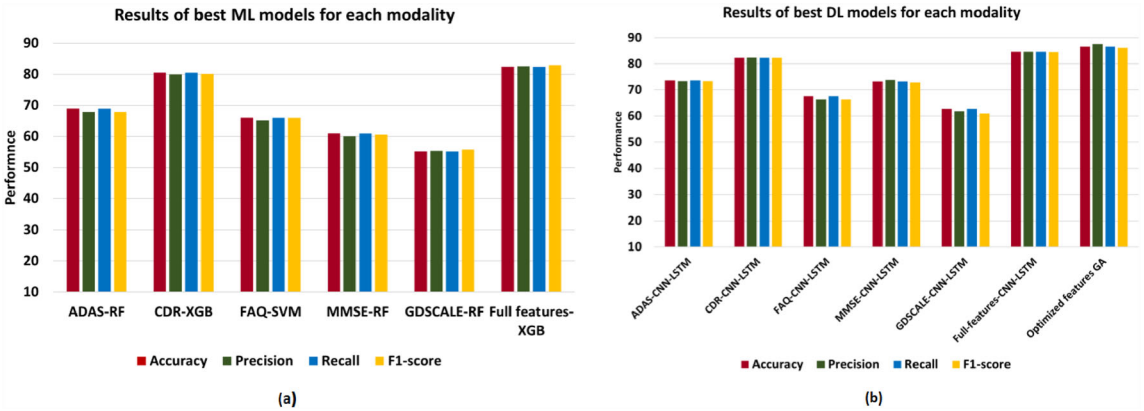

Fig. 8 Performance of the best ML and DL models for each modality for four classes

across the evaluated metrics, with the optimized models showing improved performance.

5.4 Performance of three-class (CN, sMCI, and AD) classification

This section evaluates ML classifiers, three single DL, and two DL models with optimized hyperparameters for diagnosing Alzheimer's with three classes (sMCI, CN, and AD). The results of ML and DL are listed in Table 7 and Table 8, respectively.

Experiment 1: From Table 7, we can observe the following: (1) All modalities give increasing performance with three classes compared to four classes with all ML algorithms. (2) For ADAS, NB and KNN give less performance, achieving CV accuracy of 66.51% and 68.55%, respectively. This performance was enhanced with linear classifiers such as LR and SVM, which achieved 73.13%, 74.46%, and 73.37%, 70.26% for CV and testing, respectively. The best performance was obtained from RF, achieving 74.70%, outperforming all other algorithms by about 2-4%. (3) Gaussian NB performs worst in classification accuracy of 85.30% for the CDR modality. The LR and SVM model increases the overall performance by about 3%, while the ensemble algorithms such as XGB and RF give the best performance, achieving 86.99%, 87.83%, and 83.28%, 83.09% in terms of CV and testing for XGB and RF, respectively. (4) For FAQ and MMSE, which achieve less performance than CDR and ADAS, the best performance is also obtained from RF, which achieved a CV accuracy of 68.92% and 65.76% for testing. (5) The performance of GDSC does not always give high performance in prediction. However, RF and SVM achieve adequate performance for all evaluation metrics. (6) As mentioned earlier, we combine all modalities to improve prediction performance. Combining all modalities improves the overall performance and stability of the models. All models' performance was enhanced by about 4-11% for all evaluation metrics. The best performance was obtained from XGB, with 89.64% for CV and 85.45 % for testing.

Experiment 2: We explore the performance of single and DL models with three classes. The results are listed in Table 8. From that table, we observe the following: (1) Utilizing DL outperforms all ML models with significant scores ranging from 5-10%. (2) For ADAS, the LSTM model outperforms GRU and CNN with a CV accuracy of 79.68% and a testing accuracy of 74.37%. (3) While the CNN-LSTM model gives the best performance in CV accuracy of 81.07% and testing accuracy of 74.85%. (4) Diagnosing based on CDR gives promising results in all evaluation metrics over other single modalities. The best performance was obtained from the CNN-LSTM model. It gave the best performance for FAQ and MMSE. (5) The overall model performance substantially improved when switching from single modality to multi-modality. With the full features, the proposed model achieved a CV accuracy of 92.46%, which was enhanced to 93.13% when utilizing the GA to choose the optimal features.

Figure 9 summarizes the results by highlighting the best testing performance across all modalities and for three classes. For the three-class ML models shown in Fig. 9a, the full-feature RF model achieved the highest performance across all metrics with an accuracy of 85.89, precision of 85.08, recall of 85.89, and F1-score of 85.93. In contrast, the GDSCALE modality exhibited the lowest performance, with an accuracy of 60.54. The ADAS and CDR modalities achieved the best performance with RF models, while MMSE and CDR achieved the best performance with SVM and LR models, respectively. However, the full-feature RF model outperformed them all. In the DL three-class scenarios shown in Fig. 9b, the CNN-LSTM modality achieved the best performance, with the ADAS, CDR, and FAQ modalities also exhibiting high performance. The MMSE modality achieved lower performance, while the full-feature CNN-LSTM model and the model with optimized features using GA demonstrated the highest performance. These results

Table 7 The performance of ML models with the three-class task

Models	Features	Cross-validation performance				Testing performance			
		Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
ADAS	RF	75.3±4.22	75.55±4.31	75.3±4.22	75.18±4.16	74.70	74.74	74.70	74.72
	LR	74.46±4.5	74.5±4.75	74.46±4.5	74.15±4.71	73.37	73.74	73.37	73.56
	DT	68.31±3.24	68.32±3.16	68.31±3.24	68.15±3.21	62.09	62.00	62.09	62.03
	SVM	73.13±3.81	73.17±3.6	73.13±3.81	72.81±3.73	70.26	70.95	70.26	70.95
	KNN	68.55±4.55	68.18±5.69	68.55±4.55	67.08±5.38	69.40	68.75	68.40	68.86
	NB	66.51±5.22	67.3±5.06	66.51±5.22	65.8±5.27	63.48	63.95	63.48	63.81
	XGB	73.37±4.08	73.39±3.91	73.37±4.08	73.01±3.93	70.65	70.57	70.65	70.57
CDR	RF	87.83±2.31	88.16±2.61	87.83±2.31	87.82±2.39	83.09	83.37	83.09	83.17
	LR	87.47±2.96	87.71±3.12	87.47±2.96	87.45±2.98	84.64	84.63	84.64	84.60
	DT	80.84±3.12	81.08±3.23	80.84±3.12	80.77±3.23	75.67	75.59	75.67	75.63
	SVM	88.67±2.91	89.2±2.92	88.67±2.91	88.75±2.85	84.36	84.38	84.36	84.35
	KNN	81.45±3.06	81.46±3.25	81.45±3.06	81.1±3.22	78.34	78.53	78.34	77.65
	NB	85.3±3.18	86.25±3.13	85.3±3.18	85.25±3.18	80.28	80.96	80.28	80.29
	XGB	86.99±2.46	87.28±2.7	86.99±2.46	86.94±2.54	83.28	83.36	83.28	83.32
FAQ	RF	71.81±4.35	72.13±3.91	71.81±4.35	71.45±4.28	69.73	69.89	69.73	6974
	LR	69.88±4.88	69.69±5.21	69.88±4.88	69.14±5.11	68.84	68.66	68.84	68.42
	DT	62.6±6.22	62.89±6.71	62.6±6.22	62.38±6.28	60.73	60.82	60.73	60.78
	SVM	71.45±4.28	71.79±4.47	71.45±4.28	70.56±4.37	68.01	6803	68.01	68.97
	KNN	67.11±3.96	68.81±5.85	67.11±3.96	64.47±5.01	60.51	60.31	60.51	60.39
	NB	71.2±4.62	72.25±5.69	71.2±4.62	70.03±4.71	68.59	69.03	68.59	67.33
	XGB	69.04±3.19	69.42±3.17	69.04±3.19	68.69±3.09	67.31	67.09	67.31	67.99
MMSE	RF	68.92±4.1	68.7±5.03	68.92±4.1	68.39±4.75	65.76	65.72	65.76	65.74
	LR	67.11±4.07	67.16±4.6	67.11±4.07	66.85±4.38	65.51	65.04	65.51	65.63
	DT	61.93±4.97	62.61±4.79	61.93±4.97	61.76±4.74	58.84	59.91	58.84	59.21
	SVM	68.19±4.35	68.29±4.97	68.19±4.35	67.79±4.65	65.87	65.86	65.87	65.22
	KNN	59.88±4.82	62.65±4.23	59.88±4.82	59.74±4.57	57.12	57.36	57.12	56.76
	NB	52.41±2.81	57.29±6.88	52.41±2.81	57.94±3.24	52.35	54.20	52.35	49.59
	XGB	66.14±3.16	66.41±3.34	66.14±3.16	65.87±3.32	64.06	64.80	64.06	64.89
GDSCALE	RF	65.06±6.0	64.96±5.87	65.06±6.0	64.44±6.0	60.54	60.76	60.54	60.46
	LR	64.94±5.71	64.62±5.74	64.94±5.71	64.27±5.71	60.34	60.71	60.34	60.34
	DT	55.42±6.0	55.64±5.94	55.42±6.0	55.16±6.0	53.43	53.54	53.43	53.45
	SVM	65.3±3.76	64.9±3.8	65.3±3.76	64.35±4.45	61.45	61.53	61.45	61.26
	KNN	56.99±3.92	56.86±4.33	56.99±3.92	55.65±3.83	52.35	51.05	52.35	50.83
	NB	54.94±6.95	56.65±6.95	54.94±6.95	53.65±6.83	51.62	52.09	51.62	50.13
	XGB	62.89±4.75	62.69±5.1	62.89±4.75	62.47±4.95	60.54	60.36	60.54	60.41
Full features	RF	89.52±2.53	90.0±2.49	89.52±2.53	89.56±2.47	85.89	85.08	85.89	85.93
	LR	86.99±2.57	87.14±2.48	86.99±2.57	86.95±2.58	83.03	82.95	83.03	82.94
	DT	86.02±3.1	86.47±3.13	86.02±3.1	86.03±3.07	81.56	81.46	81.56	81.50
	SVM	86.87±2.31	87.03±2.31	86.87±2.31	86.82±2.36	81.59	81.28	81.59	81.29
	KNN	75.18±3.89	76.96±4.16	75.18±3.89	74.27±4.18	69.68	70.70	69.68	68.45
	NB	81.69±3.04	82.47±3.02	81.69±3.04	81.8±2.94	75.42	75.53	75.42	75.25
	XGB	89.64±1.8	89.77±1.78	89.64±1.8	89.63±1.81	85.45	85.40	85.45	85.42

Table 8 Results of DL with three classes

Models	Features	Cross-validation performance				Testing performance			
		Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
ADAS	LSTM	79.68±2.24	79.94±2.18	77.96±2.72	78.91±2.4	74.37	74.25	74.37	74.64
	GRU	77.63±0.98	79.43±0.91	74.47±1.16	76.84±0.82	74.73	74.4	74.73	74.52
	CNN	77.63±0.98	79.43±0.91	74.47±1.16	76.84±0.82	74.73	74.4	74.73	74.52
	CNN-LSTM	81.07±3.63	81.09±3.59	80.31±3.81	80.69±3.68	74.85	75.26	74.85	74.83
	CNN-GRU	80.89±1.36	80.97±1.43	79.7±1.74	80.24±1.46	74.51	74.29	74.51	74.28
CDR	LSTM	90.52±1.59	90.36±1.77	90.29±1.77	90.32±1.76	86.52	86.7	86.52	86.5
	GRU	91.02±0.89	90.91±1.03	90.86±1.12	90.89±1.06	86.88	87.31	86.88	86.88
	CNN	90.19±0.7	90.11±0.44	90.04±0.67	90.57±0.51	87.73	87.81	87.73	87.75
	CNN-LSTM	91.02±1.15	91.99±1.15	91.76±1.31	91.87±1.21	87.85	87.78	87.85	87.8
	CNN-GRU	89.96±0.82	90.16±0.8	90.12±0.79	90.14±0.79	86.4	86.35	86.4	86.36
FAQ	LSTM	74.73±2.22	74.618±2.12	74.73±2.33	74.49±2.25	73.65	73.65	73.65	73.48
	GRU	74.27±2.47	75.8±2.58	73.68±2.42	74.71±2.44	73.77	73.45	73.77	73.5
	CNN	76.87±1.08	80.0±1.0	70.47±1.15	74.86±0.85	74.61	74.03	74.61	74.71
	CNN-LSTM	82.2±2.48	83.88±2.56	81.11±2.58	82.45±2.47	75.93	75.4	75.93	75.29
	CNN-GRU	78.14±0.86	79.87±0.86	78.23±0.9	79.03±0.74	74.73	74.61	74.73	74.49
MMSE	LSTM	72.58±2.76	73.34±2.9	69.36±3.11	71.27±2.9	66.19	66.61	66.19	66.67
	GRU	72.81±1.63	73.61±1.77	70.94±1.61	72.24±1.61	73.17	72.99	73.17	72.81
	CNN	73.96±1.99	74.92±1.89	72.05±1.93	73.43±1.81	71.84	73.83	71.84	72.08
	CNN-LSTM	76.85±2.07	79.08±2.27	74.37±1.99	76.62±1.95	74.49	74.59	74.49	74.39
	CNN-GRU	73.44±2.69	74.73±2.79	70.8±2.53	72.69±2.53	70.4	70.01	70.4	68.81
GDSCALE	LSTM	69.42±1.54	74.48±1.05	73.14±1.25	68.23±0.98	61.85	61.24	61.85	60.86
	GRU	73.96±1.99	74.92±1.89	72.05±1.93	73.43±1.81	63.66	63.15	63.66	63.23
	CNN	73.62±1.17	78.1±1.12	71.5±1.2	68.74±1.04	64.14	63.51	64.14	63.32
	CNN-LSTM	80.36±3.17	81.19±3.32	78.81±3.43	79.96±3.33	70.4	70.01	70.4	68.81
	CNN-GRU	77.22±2.29	80.76±1.97	72.85±2.37	76.56±1.97	65.46	65.17	65.46	64.97
Full features	LSTM	90.91±0.84	91.35±0.91	91.18±1.03	91.26±0.95	87.12	87.39	87.12	87.1
	GRU	91.69±1.5	92.01±1.55	91.94±1.58	91.97±1.56	86.88	86.81	86.88	86.82
	CNN	90.12±1.56	90.84±1.5	89.39±1.85	90.1±1.59	86.0	86.75	86.0	86.05
	CNN-LSTM	92.46±0.74	92.81±0.77	92.81±0.77	92.81±0.77	87.16	87.07	87.16	87.08
	CNN-GRU	90.69±1.02	90.85±0.99	90.85±0.99	95.85±0.99	85.8	85.85	85.8	85.66
Optimized features GA	RF	93.13±1.95	93.22±1.89	93.13±1.95	93.04±1.97	88.86	88.81	88.86	88.83
	LR	93.73±1.6	93.92±1.58	93.73±1.6	93.62±1.63	87.14	87.16	87.14	87.15
	DT	90.84±1.23	91.06±1.28	90.84±1.23	90.76±1.3	84.44	84.47	84.44	84.42
	SVM	93.73±1.85	93.86±1.78	93.73±1.85	93.63±1.84	87.94	87.98	87.94	87.96
	KNN	92.13±1.71	92.35±1.6	92.13±1.71	92.07±1.65	80.10	80.00	80.10	80.03
	NB	91.81±2.34	92.26±2.62	91.81±2.34	91.79±2.42	87.11	87.51	87.11	87.15
	XGB	92.53±2.07	92.77±1.98	92.53±2.07	92.5±2.07	86.60	86.55	86.60	86.54

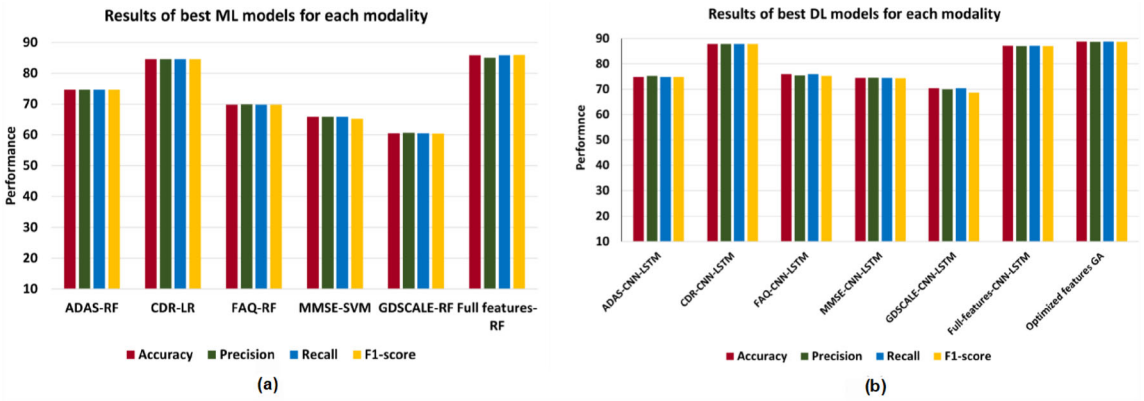

Fig. 9 Performance of the best ML and DL models for each modality for three classes

highlight the varying performance of the different models across the evaluated metrics, with the optimized models showing improved performance.

5.5 Performance of two-class (CN and AD) classification

This section evaluated ML classifiers, three single DL, and two DL models with optimized hyperparameters for diagnosing Alzheimer's with two classes (CN, and AD). The results of ML and DL are listed in Table 9 and Table 10, respectively.

Experiment 1: In this section, we concentrated on using ML, single DL, and hybrid DL to classify instances into AD or CN based on single and multimodalities. All results are listed in Tables 5 and 9. We can observe the following: (1) For ADAS and CDR, RF performs best among all other algorithms. It achieved 89.28%, 89.52%, and 87.81%, 88.97% regarding CV accuracy and testing accuracy, respectively. (2) FAQ and MMSE perform similarly to most algorithms, while the best performance was obtained from RF. (3) Utilizing the full features enhances overall performance by about 2-8% with all algorithms. Tree-based algorithms such as RF and XGB perform similarly for CV accuracy and testing performance. The best performance was obtained by LR and SVM, which gave 92.17%, 91.45%, 88.34%, and 87.25%.

Experiment 2: In Table 10, we present the classification results of single and hybrid models. From Table 6, we make the following observations: (1) LSTM achieves promising performance over GRU and CNN, while CNN provides a more stable model (SD=0.68). Utilizing the CNN-LSTM model gives the best performance for ADAS and CDR modalities. (2) FAQ and MMSE perform similarly in all evaluation metrics; they achieve 90.33%, 90.31%, and 88.81%, 88.9% for CV accuracy and testing performance. (3) We utilized the fused dataset that combines all modalities to increase the overall performance. GRU gives the best performance with the fused features over LSTM and CNN. Utilizing the CNN-LSTM model provides the best performance with the full features. (4) To ensure choosing the optimal feature subset, we utilized GA to select the optimum feature subset from the features that were output from DL. The chosen features were then utilized to build a classification model. The best performance for this step was obtained by utilizing RF, which achieved promising performances of 97.11% and 91.89% in CV accuracy and testing accuracy, outperforming all other models.

Figure 10 summarizes the results by highlighting the best testing performance across all modalities and for two classes. Regarding the two-class ML models shown in Fig. 10a, the CDR and ADAS modalities achieved the best performance with RF models. The FAQ and MMSE modality performed well with XGB models, while the GDSCALE modality achieved relatively lower performance. The full features consistently demonstrated high accuracy, precision, recall, and F1-score with XGB models. Regarding the DL two-class scenarios, as shown in Fig. 10b, the ADAS and CDR modalities achieved high performance, while the FAQ and MMSE modalities exhibited good accuracy. The GDSCALE modality achieved lower performance. The full features with CNN-LSTM model demonstrated high accuracy, and the model with optimized features using GA achieved the highest overall performance. These results highlight the varying performance of the different models across the evaluated metrics, with the optimized models showing improved performance.

6 Discussion

In the following subsection, we compare the performance of single and multi-modalities in diagnosing different classes. Firstly, we evaluate the performance of machine learning (ML) and deep learning (DL) models with a single modality, where we compare the proposed model with the

Table 9 The performance of ML models with the two-class task

Models	Features	Cross-validation performance				Testing performance			
		Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
ADAS	RF	89.28±2.61	89.28±3.09	90.48±1.83	90.24±1.78	87.81	87.88	87.81	87.54
	LR	88.43±2.3	89.97±2.8	90.12±2.65	91.46±1.67	86.73	86.67	86.73	86.69
	DT	85.18±3.66	87.98±3.55	90.24±4.38	88.99±2.73	85.56	85.54	85.56	85.12
	SVM	87.23±2.81	89.66±3.15	90.49±3.12	90.5±2.04	86.09	86.01	86.09	86.04
	KNN	82.17±2.79	88.48±3.75	84.43±3.78	86.29±2.09	80.87	81.61	80.87	81.11
	NB	88.55±2.65	89.86±3.08	90.49±3.54	91.56±1.99	86.81	86.84	86.81	86.82
	XGB	87.47±2.3	90.08±2.67	91.31±2.64	90.65±1.72	87.73	87.73	87.73	87.73
CDR	RF	89.52±2.8	89.7±2.66	89.52±2.8	89.47±2.72	88.97	88.95	88.97	88.96
	LR	90.36±2.09	90.45±2.15	90.36±2.09	90.28±2.05	88.61	88.66	88.61	88.44
	DT	83.86±4.04	84.45±3.63	83.86±4.04	83.89±3.8	86.64	86.61	86.64	86.62
	SVM	90.72±2.02	90.87±2.1	90.72±2.02	90.63±1.95	87.25	87.25	87.25	87.09
	KNN	87.95±1.43	87.93±1.48	87.95±1.43	87.81±1.37	87.73	87.87	87.73	87.35
	NB	88.8±2.85	89.04±2.87	88.8±2.85	88.78±2.81	86.17	86.66	86.17	86.29
	XGB	89.64±3.15	89.82±3.05	89.64±3.15	89.59±3.11	87.09	87.06	87.09	87.07
FAQ	RF	89.88±3.1	89.86±3.2	89.88±3.1	89.79±3.16	83.39	84.68	83.39	82.14
	LR	89.16±2.41	89.12±2.45	89.16±2.41	88.97±2.51	82.81	82.71	82.81	82.69
	DT	83.49±3.77	83.69±3.74	83.49±3.77	83.47±3.67	80.28	80.12	80.28	80.12
	SVM	90.0±2.9	90.03±2.94	90.0±2.9	89.86±2.99	82.31	82.27	82.31	82.29
	KNN	89.28±2.5	89.37±2.53	89.28±2.5	89.02±2.61	83.45	83.34	83.45	83.35
	NB	88.55±2.3	89.0±2.58	88.55±2.3	88.62±2.34	82.81	82.80	82.81	82.58
	XGB	88.19±2.89	88.2±3.05	88.19±2.89	88.11±2.96	84.64	84.98	84.64	84.76
MMSE	RF	88.67±2.1	88.81±2.18	88.67±2.1	88.55±2.13	78.64	78.50	78.64	78.42
	LR	87.47±2.1	87.63±2.2	87.47±2.1	87.25±2.15	78.00	78.87	78.00	78.81
	DT	83.37±2.4	83.49±2.27	83.37±2.4	83.31±2.37	79.78	80.02	79.78	79.89
	SVM	88.8±3.1	88.86±3.25	88.8±3.1	88.68±3.1	79.00	79.89	79.00	79.76
	KNN	81.93±2.09	82.87±2.8	81.93±2.09	80.63±2.28	79.06	80.70	79.06	76.78
	NB	80.12±4.32	82.15±3.83	80.12±4.32	80.48±4.24	77.62	78.51	77.62	77.92
	XGB	87.23±2.76	87.48±2.82	87.23±2.76	87.13±2.77	80.20	80.00	80.20	80.95
GDSCALE	RF	82.65±3.33	82.57±3.53	82.65±3.33	82.33±3.53	78.70	78.25	78.70	78.34
	LR	79.64±4.01	79.55±4.14	79.64±4.01	79.07±4.17	76.90	76.22	76.90	76.20
	DT	73.98±4.76	74.35±5.29	73.98±4.76	73.86±4.88	72.56	72.87	72.56	72.70
	SVM	79.88±2.8	79.69±2.97	79.88±2.8	79.36±3.05	73.65	72.96	73.65	71.06
	KNN	73.49±4.24	72.64±4.67	73.49±4.24	72.31±4.52	66.43	64.19	66.43	64.55
	NB	73.61±4.83	73.27±4.82	73.61±4.83	73.09±4.74	66.79	66.97	66.79	66.87
	XGB	81.08±2.75	81.01±2.96	81.08±2.75	80.94±2.83	78.34	78.04	78.34	78.15
Full features	RF	90.84±2.3	90.92±2.29	90.84±2.3	90.8±2.3	88.34	88.75	88.34	88.07
	LR	92.17±2.48	92.26±2.41	92.17±2.48	92.17±2.47	88.34	88.39	88.34	88.36
	DT	87.47±3.06	87.73±2.82	87.47±3.06	87.5±2.98	86.53	86.72	86.53	89.60
	SVM	91.45±2.38	91.46±2.38	91.45±2.38	91.41±2.39	87.25	87.28	87.25	87.27
	KNN	90.48±2.19	90.77±2.14	90.48±2.19	90.23±2.3	87.17	87.11	87.17	87.01
	NB	89.76±3.5	90.22±3.31	89.76±3.5	89.84±3.43	87.34	87.46	87.34	87.38
	XGB	90.72±3.05	90.88±2.95	90.72±3.05	90.7±3.02	88.70	88.72	88.70	88.71

Table 10 The performance of DL for the two-class task

Models	Features	Cross-validation performance				Testing performance			
		Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
ADAS	LSTM	91.87±1.2	91.19±0.99	91.19±0.99	91.19±0.99	89.29	89.45	89.29	89.33
	GRU	88.42±0.43	88.2±0.5	88.2±0.5	88.2±0.5	88.45	88.45	88.45	88.45
	CNN	91.79±0.68	91.67±0.65	91.67±0.65	91.67±0.65	89.24	89.31	89.24	89.27
	CNN-LSTM	92.29±0.74	92.11±0.71	92.11±0.71	92.11±0.71	90.10	90.20	90.10	90.20
	CNN-GRU	89.26±0.85	89.02±0.95	89.02±0.95	89.02±0.95	88.57	88.6	88.57	88.58
CDR	LSTM	90.41±0.24	91.01±0.2	91.01±0.2	91.01±0.2	89.97	89.15	89.97	89.03
	GRU	90.41±0.24	91.01±0.2	91.01±0.2	91.01±0.2	89.89	89.85	89.89	89.86
	CNN	89.56±0.57	89.77±0.59	89.77±0.59	89.77±0.59	89.89	89.85	89.89	89.86
	CNN-LSTM	91.63±0.46	91.99±0.46	91.99±0.46	91.99±0.46	90.25	90.23	90.25	90.24
	CNN-GRU	84.8±1.93	84.2±2.0	84.2±2.0	84.2±2.0	88.21	88.14	88.21	88.15
FAQ	LSTM	89.17±0.78	89.48±0.81	89.48±0.81	89.48±0.81	87.49	87.51	87.49	87.45
	GRU	88.72±0.62	88.19±0.63	88.19±0.63	88.19±0.63	87.73	87.73	87.73	87.67
	CNN	88.73±0.57	88.13±0.6	88.13±0.6	88.13±0.6	88.59	88.55	88.59	88.56
	CNN-LSTM	90.33±0.68	89.99±0.79	89.99±0.79	89.99±0.79	88.81	88.78	88.81	88.79
	CNN-GRU	88.67±0.8	88.33±1.01	88.33±1.01	88.33±1.01	88.45	88.39	88.45	88.42
MMSE	LSTM	82.09±0.85	81.26±0.97	81.26±0.97	81.26±0.97	82.32	82.16	82.32	82.19
	GRU	84.71±1.26	84.1±1.37	84.1±1.37	84.1±1.37	81.35	81.95	81.35	81.51
	CNN	86.38±1.2	86.39±1.2	86.39±1.2	86.39±1.2	83.32	84.14	83.32	82.19
	CNN-LSTM	90.3±0.81	90.02±0.78	90.02±0.78	90.02±0.78	88.09	88.05	88.09	87.84
	CNN-GRU	87.89±0.87	87.39±0.83	87.39±0.83	87.39±0.83	80.92	80.89	80.92	80.9
GDSCALE	LSTM	82.09±0.85	81.26±0.97	81.26±0.97	81.26±0.97	80.19	80.93	80.19	80.98
	GRU	84.71±1.26	84.1±1.37	84.1±1.37	84.1±1.37	81.35	81.95	81.35	81.51
	CNN	81.63±1.26	80.45±1.25	80.45±1.25	80.45±1.25	79.06	80.5	79.06	79.45
	CNN-LSTM	85.04±0.9	84.33±0.85	84.33±0.85	84.33±0.85	82.23	82.05	82.23	82.06
	CNN-GRU	84.4±1.57	84.21±1.79	84.21±1.79	84.21±1.79	79.3	79.57	79.3	79.33
Full features	LSTM	92.53±1.45	92.38±1.43	92.38±1.43	92.38±1.43	88.09	88.05	88.09	87.84
	GRU	92.79±0.68	92.67±0.65	92.67±0.65	92.67±0.65	88.45	88.75	88.45	88.54
	CNN	91.79±0.68	91.67±0.65	91.67±0.65	91.67±0.65	88.08	88.11	88.08	88.08
	CNN-LSTM	94.23±1.51	94.12±1.58	94.12±1.58	94.12±1.58	90.41	90.45	90.41	90.43
	CNN-GRU	93.12±0.93	92.65±0.9	92.65±0.9	92.65±0.9	89.24	89.31	89.24	89.27
Optimized features GA	RF	97.11±1.23	96.69±2.03	97.1±1.21	97.86±0.89	91.89	91.81	91.89	91.80
	LR	97.23±1.08	97.85±1.63	97.1±1.21	97.95±0.79	89.17	89.08	89.17	89.08
	DT	95.9±1.54	96.31±1.88	97.66±2.41	96.94±1.15	87.73	87.63	87.73	87.66
	SVM	97.35±1.05	97.19±1.7	97.92±1.19	97.03±0.76	89.17	89.09	89.17	89.11
	KNN	96.99±1.45	97.17±1.73	98.37±1.26	97.75±1.07	89.17	89.12	89.17	89.14
	NB	97.23±1.08	97.02±1.68	98.92±1.19	97.94±0.79	89.17	89.09	89.17	89.11
	XGB	96.87±1.23	96.52±1.96	98.92±1.64	97.68±0.9	88.45	88.34	88.45	88.31

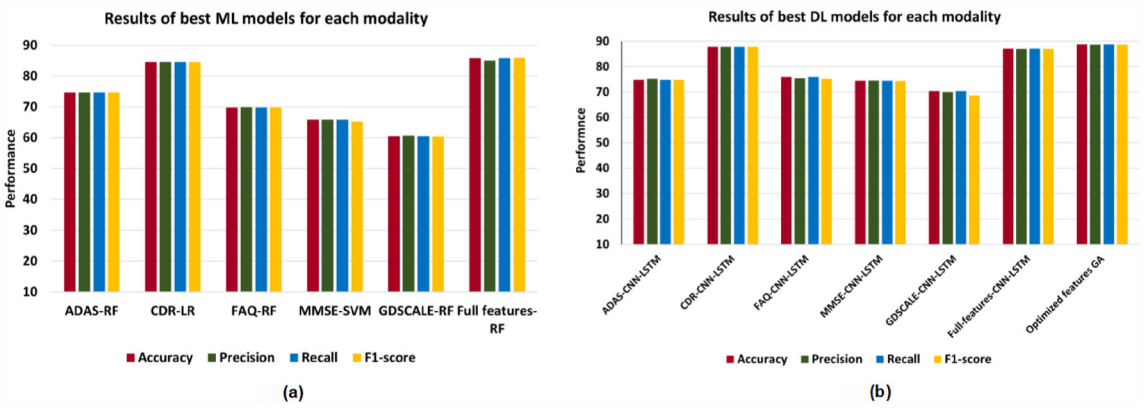

Fig. 10 Testing performance of the best ML and DL models for each modality for two classes

best-performing ML and DL models in each modality. Secondly, we analyze the performance of ML and DL models with multi-modalities and compare them with the proposed model in four-, three-, and two-class contexts.

6.1 Comparison between single and multimodalities

First, for diagnosing with a single modality. Figure 11a compares models in four classes with a single modality. It compares the best ML model, the best DL model, and the proposed mode according to each modality. From that, we can observe the following. (1) CDR gives the best performance in diagnosing based on a single modality followed by CDR, (2) DL outperforms the best ML for all modalities, and (3) the proposed model gives the best overall performance in ML and DL models, which confirmed the efficiency of the proposed work over other algorithms. The same applies to the three classes in Fig. 11b. Figure 11b also shows that diagnosing into three classes improves overall performance than the four classes in Fig. 11c. Second, for multi-modalities, Fig. 12 compares the performance of the best ML model, the best DL model, and the proposed model with multi-modalities in four-, three-, and two-class contexts. We can make the following observations from Fig. 12. (1) Combining all modalities improves the overall performance in diagnosing with all ML and DL algorithms and enhances the model's stability. (2) Results with our proposed model outperform other ML and DL with about 4–6% in four, three, and two classes.

Figure 11c compares the best ML model for the two-class contexts with each modality. As shown in Fig. 11c, the proposed model outscores all ML and single DL in other modalities regarding testing accuracy. Figure 12 compares the performance of the best ML model, the best DL model, and the proposed model with multimodalities in four-, three-, and two-class contexts. As shown, combining all modalities improves the overall performance in terms of all classes' contexts; also, the proposed model outperforms the best ML and DL models.

6.2 Explainability

Explainability in the medical domain. Providing a meaningful explanation for the developed outcome is critical to trust the developed decision. Medical experts need to know why the model made that decision. This leads us to answer several questions: What are the most significant parameters that are highly considered in making the developed decision? How do inputs for the model lead to the output? Are the selected parameters medically affected and sufficient? This subsection will concentrate on implementing XAI for the four, three, and two classes. XAI capabilities provide in various ways, including feature importance according to global and local explanations,

To provide a clear explanation and model interpretation, we utilized both SHAP and LIME, each of them helping in providing a clear explanation about the developed output. First, SHAP, is a model-agnostic method that assigns importance values to each feature, quantifying their contribution to the prediction outcome. In our study, we calculate the SHAP values, representing each feature's marginal contribution to the prediction outcome. By summing up these values, we obtain a measure of feature importance. This information is invaluable for medical professionals, as it helps them identify the features with the most significant impact on the model's predictions. Secondly LIME, it is another interpretability method that explains individual predictions made by a model. It works by approximating the model's behavior around a specific instance to provide local explanations. In our study, we utilized LIME to generate explanations for individual predictions made by our model. These explanations offer insight into how specific patient characteristics or input variables influenced the model's prediction for each instance. This local interpretability helps medical experts understand the

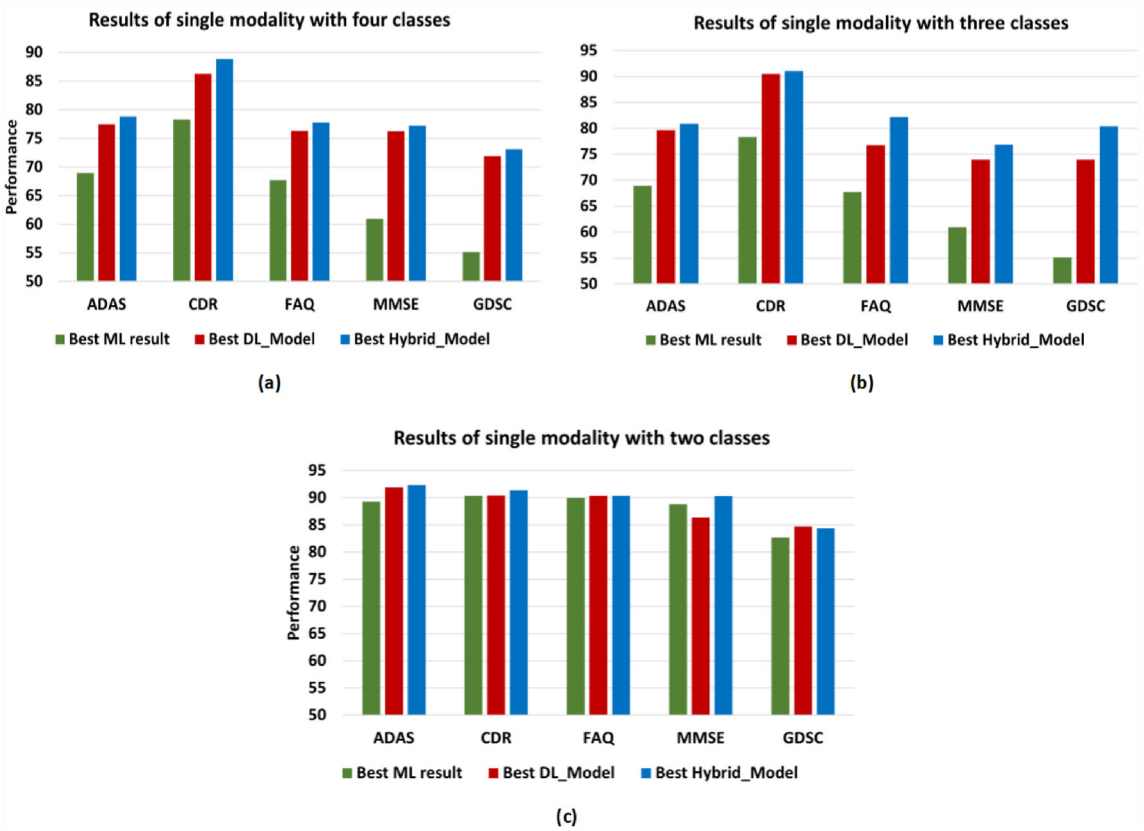

Fig. 11 Performance of the Top ML classifier, Top DL classifier, and the proposed model. **a** comparison between Best ML, DL, and proposed model with single modalities in four-class context, **b** comparison between Best ML, DL, and proposed model with single modalities in three-class context, **c** comparison between Best ML, DL, and proposed model with single modalities in two-class context

reasoning behind the model's decisions and identify potential biases or limitations in specific patients. These insights are crucial for medical professionals in assessing the reliability and applicability of the model's predictions in a medical context. The resulting models in our study are accurate and explainable. As a result, the models can be trusted by domain experts. The practical implications of these models can be summarized in

- Improving the quality of Alzheimer's prediction.
- Reducing the costs of disease management.
- Reduce the workload and the level of quality for physicians.
- Reducing the need and dependence on MRI for disease prediction.

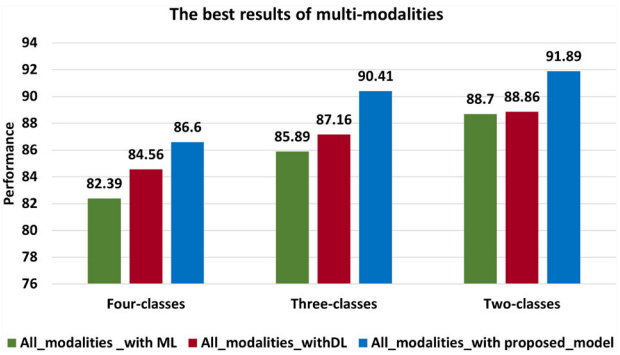

Fig. 12 Comparison between the performance of the best ML model, the best DL model, and the proposed model with multi-modalities in four-, three-, and two-class contexts

6.2.1 XAI for four classes

We utilize SHAP and LIME explainers to show the effect of each of the most significant features in the output. Figure 13 shows the global importance of the features using SHAP; it concentrates on the importance of utilized features in the whole dataset. SHAP shows a global explanation for the prediction. As shown in Fig. 13a, CD_Memory, CD_judge, and CD home significantly impact determining whether the patient is CN or AD. After specifying the global feature importance, we need to ensure the importance of each feature in the individual instances (local explanation). LIME

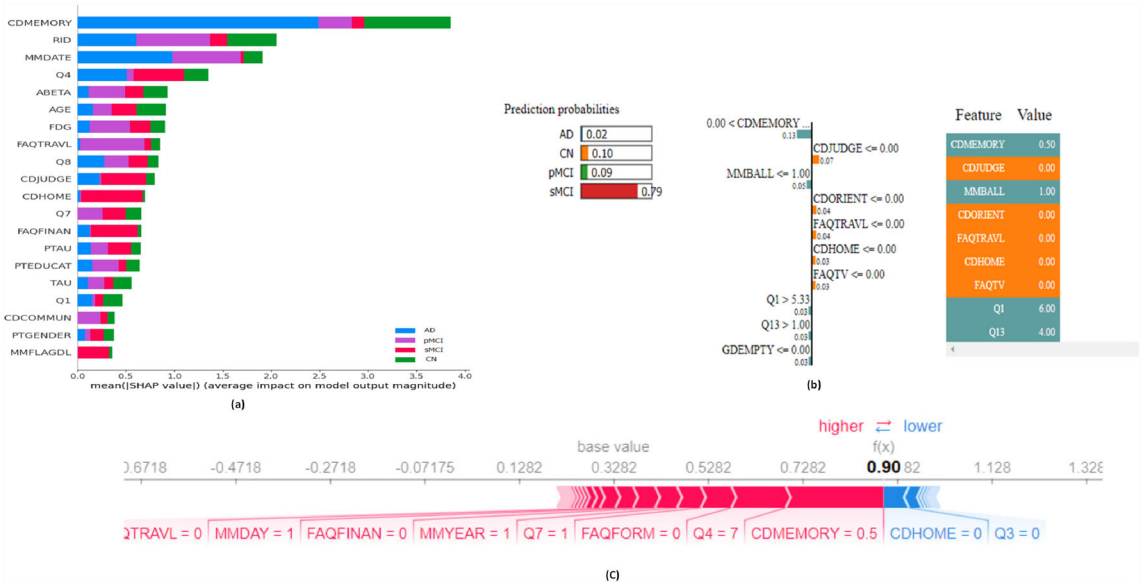

Fig. 13 Explanation of four classes with multi-modality **a** summary plot according to SHAP explainers, **b** The LIME output for Observation, **c** SHAP Force plot for Observation

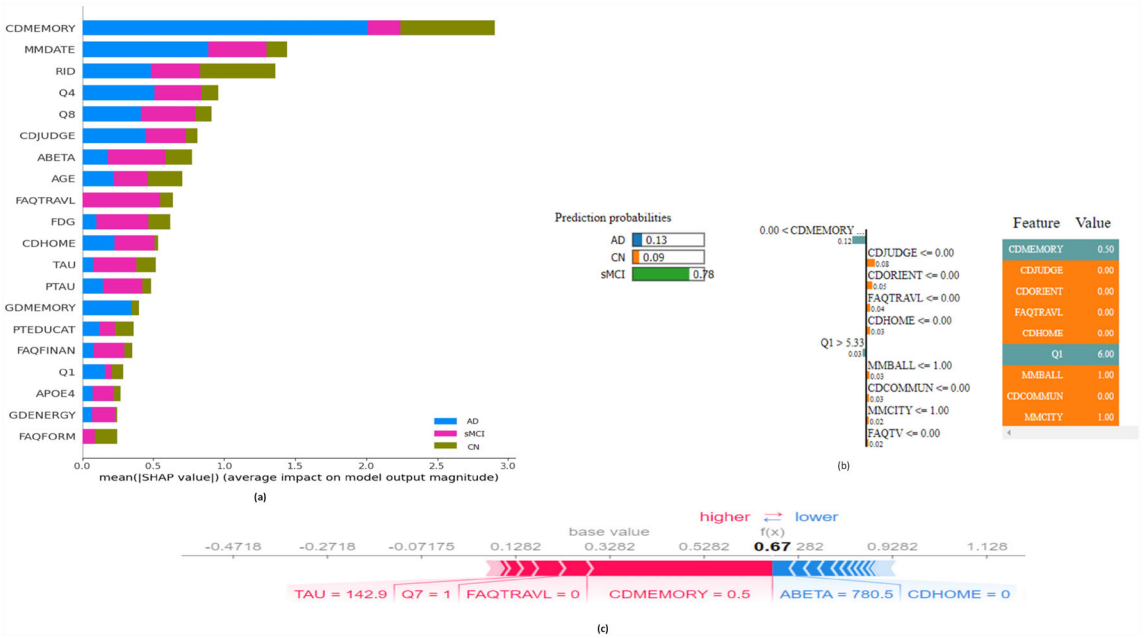

Fig. 14 Explanation of three classes with single and multi-modality **a** summary plot according to SHAP explainers, **b** The LIME output for Observation, **c** SHAP Force plot for Observation

and SHAP show the most important features according to specific instances. For example, for instance, five explained in Fig. 13b, the class was sMCI with probability prediction of =0.79, CD_Memory, CD_Judge, and MMBALL were the most significant features. Orange and green in feature names indicate whether the features negatively or positively affect the decision. These results were also confirmed using the SHAP explainer as in Fig. 13c. MMO, FDG, CD_Memory, CD_home,CD_judge, MMO, Mmoneflr. The most significant features, according to SHAP, are also confirmed in LIME.

6.2.2 XAI for three classes

First, a summary plot is utilized to show the effect of each of the most significant features in the output. Figure 14 shows the global importance of the features using SHAP and LIME. SHAP shows a global explanation for the prediction. Feature

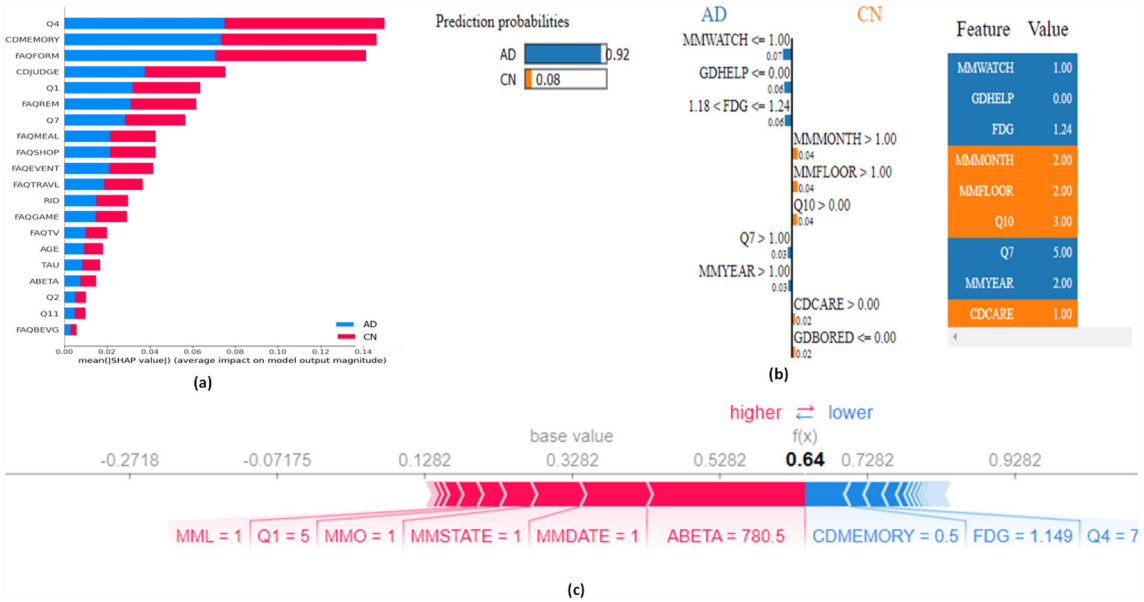

Fig. 15 Explanation of two classes with single and multi-modality **a** Summary plot according to SHAP explainers, **b** The LIME output for Observation, **c** SHAP Force plot for Observation

names show on the X-axis, while the axis shows bars that represent the full feature's importance. Blue, green, and purple represent the effect according to AD, sMCI, and CN classes. As shown in Fig. 14a, CD_Memory, MMDate, significantly impacts AD class prediction but is less important for sMCI and CN classes. FAQTRAVl has a high impact on specifying the sMCI class and has less importance for CN and no importance for the AD class. To check the significance of each feature in the individual instance, we utilized SHAP and LIME to explore local explanations. As shown in Fig. 14b, LIME shows that the most essential features according to a specific instance, for example, instance, five that explained in Fig. 14b, the class was sMCI with probability prediction of =0.79, and CD_Memory, CD_Judge, CDORIENT were the most significant features. Features positively correlated with the target are displayed in green; otherwise, they are shown in red. These results were also confirmed using the SHAP explainer as in Fig. 14c. The base value was 0.32, and the predicted probability was 0.67. FDG, CD_Memory, CD_home, and CD_judge were the most significant features according to SHAP, confirmed in LIME as shown in Fig. 14c.

6.2.3 XAI for two classes

As shown in Fig. 15, we utilized SHAP and LIME to clearly understand the developed decision, at both the dataset level and instance level. First, in terms of global explainability (dataset level), we utilized a summary plot in Fig. 15a which shows each feature's effect on the overall output. Feature names show on the X-axis, while the axis shows bars that represent the full feature's importance. Blue and red represent the effect according to AD and CN classes. For example, Q4 has the same significance in AD and CN classes. Q1 and CD_JUDGE have a higher impact on the AD class than the CN class. Second, we need to ensure the features' impact at the instance level (instance level). As shown in Fig. 15b, MMWATCH, CDHELP, and FDG significantly impact prediction according to their values. Features positively correlated with the target are displayed in green; otherwise, they are shown in red. We utilized the SHAP explainer as in Fig. 15c to ensure that result. The base value was 0.32, and the predicted probability was 0.64. FDG, CD_Memory, CD_home, and CD_judge were the most significant features according to SHAP, confirmed in LIME as shown in Fig. 15c.

Testing results (accuracy/precision/recall/f-score/AUC); CS: Cognitive scores; LOOCV: Leave-one-out cross-validation; CHC: Chettinad Health City, Chennai; CV: Cross-validation

6.3 Comparison with state-of-the-art studies

In this study, we proposed a hybrid deep learning architecture to predict AD based on CSs sub-scores. The study provided a cost-effective and optimal method to predict the disease without costly MRI images. Table 11 provides a survey of the existing studies for AD. As can be seen, most studies are based on the ADNI dataset [67, 67–75, 75]. All studies are based on neuroimaging data [67]. As a result, these studies are mainly based on different CNN architectures [42, 67, 67–70, 72, 73, 75, 75]. Some studies are based on classical ML models like SVM [68, 71]. Hybrid deep learning models are well-known techniques to optimize the performance

Table 11 Comparison with the literature

Ref. (Year)	Dataset	Subjects	Modality	Task (purpose)	DL model	CV	Hyper. Opt	Target	Results
Proposed	ADNI	1107	CS sub-scores	AD progression	Hybrid CNN-LSTM-RF	10-CV	BO	MCI vs. AD	CV:97.11/96.69/97.1/97.86 Testing: 91.89/91.81/91.89/91.80
Proposed	ADNI	1107	CS sub-scores	AD progression	Hybrid CNN-LSTM-RF	10-CV	BO	CN, sMCI, AD	CV:93.13/93.22/93.13/93.04 Testing:88.86/88.81/88.86/88.83
Proposed	ADNI	1107	CS sub-scores	AD progression	Hybrid CNN-LSTM-RF	10-CV	BO	CN vs sMCI vs pMCI vs AD	CV:91.49/91.65/91.10/91.37 Testing:86.60/87.56/86.60/86.13
(2021) [76]	ADNI	300	3D MRI	AD detection	3D CNN	10-CV	Adam optimizer	MCI vs. AD	CV: 89.4/–/95.2/–/–
(2021) [68]	ADNI	1076	sMRI	AD detection	SVM	10-CV	Gridsearch	CN vs. MCI	CV: 89.4/–/95.2/–/–
(2021) [69]	ADNI	1005	3D FDG-PET	AD progression	CNN (MiSePyNet)	train/ validation/ test	SGD	sMCI vs. pMCI	Testing: 83.1/–/72.1/–/86.8
								CN vs. AD	Testing: 93.1/–/90.3/–/97.1
(2021) [70]	ADNI	1248	MRI	AD detection	ResNet	5-CV	Adam optimizer	pMCI vs. sMCI	CV: 77.6/–/71.6/–/–
(2021) [71]	ADNI	762	MRI	AD detection	SVM	10-CV	Grid search	CN vs. AD	CV: 82.6/–/–/78/90
(2021) [72]	ADNI	240	sMRI	AD detection	CNN (ResNet-101)	balanced train/test	–	Cn vs. MCI vs. AD	CV: 96.3/–/96.7/–/–
(2021) [73]	ADNI	807	3D sMRI	AD progression	2D CNN	5-CV	Adadelta optimizer	sMCI vs. pMCI	CV: 71.0/–/59.8/–/71.9
(2020) [42]	CUNDRC	331	rs-fMRI	AD detection	3D CNN (modified VGG-Net)	Stratified 10-CV	Adam optimizer	CN vs. AD	Testing: 85.3/–/-/–/–
(2020) [74]	ADNI OASIS	2179	MRI, sex, age	AD progression	Hybrid model (ResNet-SVM)	LOOCV	Random search	CN vs. sMCI vs. pMCI vs. AD	CV: 86.8/32.1/ 37.8/33.5/–
									CV: 78.6/68.9/ 58.3/60.3/-
(2020) [75]	ADNI	828	MRI	AD progression	CNN (ResNet)	5-CV	Adam Optimizer	sMCI vs. AD	Testing: 87.5/–/-/–/–
								sMCI vs. pMCI	Testing: 75.1/–/-/–/–
(2020) [77]	CHC	200	MRI	AD detection	CNN (AlexNet)	–	–	CN vs. AD	CV: 95/–/95/–/–
(2020) [67]	ADNI	125	MRI	AD detection	CNN	Train/test splitting	Adam Optimizer	CN vs. MCI	Testing: 98/–/-/–/–

of the resulting models. In [74], authors proposed a hybrid model based on ResNet-SVM to interpret MRI images and integrate them with sex and age. However, Alzheimer's disease is a longitudinal disease that has progressed over time. As a result, the patient's collected data are always time series in nature. No study in the table performed an analysis of time series data. Compared to the literature, our study is the first study that performs deep multimodal time series analysis of CSs sub-scores. This provides a good opportunity for discovering hidden AD biomarkers. Our study proposed a suitable DL architecture consisting of multi-layer optimization of the hybrid CNN-LSTM-RF model. CNN modality was used to extract local spatial features from 1D longitudinal features, and the LSTM model was used to learn the deep longitudinal features from the multivariate time series features. The model's hyperparameters have been optimized using Bayesian optimization. RF model' hyperparameters have been optimized using grid search. The extracted deep features from CNN-LSTM deep learning architecture have been optimized using a genetic algorithm. In addition, the resulting model has been extended by adding an extra XAI step using SHAP and LIME techniques. In summary, the study provided an extended step in AD progression detection, DL modeling, and optimization.

7 Conclusion

In this study, we proposed a hybrid deep learning model for predicting AD progression. The model has been used to interpret multimodal time series data of cognitive scores, including FAQ, MMSE, ADAS, GDSCALE, and CDRSB. The problem has been formulated as multiclass classification tasks with different settings, including 4-class, 3-class, and 2-class classification. The proposed model integrates CNN and LSTM models to extract deep longitudinal features from multimodal time series data. An optimization step using GA-based feature selection has been added to the pipeline to select the most relevant deep features learned by the CNN-LSTM blocks. Furthermore, several ML algorithms, including RF, DT, LR, KNN, and XGB, were tested as potential replacements for the SoftMax classifier. The best performance was achieved when RF was used instead of SoftMax. The utilized ML classifier has been used to make the final decision in place of SoftMax. A comprehensive analysis of the results based on ADNI time series data shows the superiority of the proposed model compared to other DL and classical machine learning models. For two classes, the best testing results are (accuracy=91.89, precision =91.81,recall=91.89, and F1-score=91.80). For three classes, the best testing results are (accuracy=88.86, precision=88.81, recall=88.86, and F1-score=88.83). For four classes, the best testing results are (accuracy=86.60, precision=87.56, recall=86.60, and F1-score=86.13). The resulting model has been extended to provide XAI capabilities using two of the most popular XAI techniques, LIME and SHAP. Detailed analysis of the XAI features has been discussed to investigate the role of different sub-scores in detecting AD progression. In the future, we will extend the proposed model by making external validation based on other datasets like NACC. We will explore the capabilities of the recent DL architectures like transformers to learn complex time series data. Finally, we will explore other XAI techniques like case-based reasoning and visualization.

Acknowledgements This study is supported via funding from Prince Sattam bin Abdulaziz University project number (PSAU/2024/R/1445).

Author Contributions All authors contributed to the study's conception and design. Data collection, preparation, and analysis were performed by HS and SES. The first draft of the manuscript was written by HS, NER, AAM, SES, AOA contributed to methodology and writing—review and editing.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Tu, Y., Lin, S., Qiao, J., Zhuang, Y., Zhang, P.: Alzheimer's disease diagnosis via multimodal feature fusion. Comput. Biol. Med. **148**, 105901 (2022)
2. Abuhmed, T., El-Sappagh, S., Alonso, J.M.: Robust hybrid deep learning models for alzheimer's progression detection. Knowl.-Based Syst. **213**, 106688 (2021)
3. Lahmiri, S., Shmuel, A.: Performance of machine learning methods applied to structural MRI and ADAS cognitive scores in diagnosing alzheimer's disease. Biomed. Signal Process. Control **52**, 414–419 (2019)
4. Shukla, A., Tiwari, R., Tiwari, S.: Review on alzheimer disease detection methods: automatic pipelines and machine learning techniques. Sci **5**(1), 13 (2023)
5. Shoaip, N., Rezk, A., El-Sappagh, S., Alarabi, L., Barakat, S., Elmogy, M.M.: A comprehensive fuzzy ontology-based decision support system for alzheimer's disease diagnosis. IEEE Access **9**, 31350–31372 (2020)
6. Tanveer, M., Richhariya, B., Khan, R.U., Rashid, A.H., Khanna, P., Prasad, M., Lin, C.: Machine learning techniques for the diagnosis of alzheimer's disease: a review. ACM Trans Multimedia Comput Commun Appl (TOMM) **16**(1s), 1–35 (2020)
7. Arya, A.D., Verma, S.S., Chakarabarti, P., Chakrabarti, T., Elngar, A.A., Nami, M., Kamali, A.-M.: A systematic review on machine learning and deep learning techniques in the effective diagnosis of alzheimer's disease (2022)
8. El-Sappagh, S., Abuhmed, T., Islam, S.R., Kwak, K.S.: Multimodal multitask deep learning model for alzheimer's disease progression detection based on time series data. Neurocomputing **412**, 197–215 (2020)
9. Suk, H.-I., Lee, S.-W., Shen, D., Initiative, A.D.N., et al.: Hierarchical feature representation and multimodal fusion with deep learning for ad/mci diagnosis. Neuroimage **101**, 569–582 (2014)

10. Rahim, N., El-Sappagh, S., Ali, S., Muhammad, K., Del Ser, J., Abuhmed, T.: Prediction of alzheimer's progression based on multimodal deep-learning-based fusion and visual explainability of time-series data. Inf Fusion **92**, 363–388 (2023)
11. Da, X., Toledo, J.B., Zee, J., Wolk, D.A., Xie, S.X., Ou, Y., Shacklett, A., Parmpi, P., Shaw, L., Trojanowski, J.Q., et al.: Integration and relative value of biomarkers for prediction of mci to ad progression: spatial patterns of brain atrophy, cognitive scores, apoe genotype and csf biomarkers. NeuroImage: Clin **4**, 164–173 (2014)
12. Ihara, R., Iwata, A., Suzuki, K., Ikeuchi, T., Kuwano, R., Iwatsubo, T., Initiative, J.A.D.N., et al.: Clinical and cognitive characteristics of preclinical alzheimer's disease in the Japanese alzheimer's disease neuroimaging initiative cohort. Alzheimer's Dementia: Transl. Res. Clin. Interv. **4**, 645–651 (2018)
13. Huang, X., Kroening, D., Ruan, W., Sharp, J., Sun, Y., Thamo, E., Wu, M., Yi, X.: A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability. Comput. Sci. Rev. **37**, 100270 (2020)
14. El-Sappagh, S., Alonso-Moral, J.M., Abuhmed, T., Ali, F., Bugarín-Diz, A.: Trustworthy artificial intelligence in alzheimer's disease: state of the art, opportunities, and challenges. Artif. Intell. Rev. 1–148 (2023)
15. Islam, E.-S.S.A.J., KS, S.S.A.K.: A multilayer multimodal detection and prediction model based on explainable artificial intelligence for alzheimer's disease. Rep **11**(1), 1 (2021)
16. Del, A.A.D.-R.N.: Ser j bennetot a tabik s barbado a garcía s gil-lópez s molina d benjamins r et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible AI Information Fusion **58**(82), 115 (2020)
17. El-Sappagh, S., Alonso, J.M., Islam, S., Sultan, A.M., Kwak, K.S.: A multilayer multimodal detection and prediction model based on explainable artificial intelligence for alzheimer's disease. Sci. Rep. **11**(1), 1–26 (2021)
18. Lundberg, S.M., Lee, S.-I.: A unified approach to interpreting model predictions. Adv. Neural Inf. Process. Syst. **30** (2017)
19. Molnar, C., König, G., Herbinger, J., Freiesleben, T., Dandl, S., Scholbeck, C.A., Casalicchio, G., Grosse-Wentrup, M., Bischl, B.: General pitfalls of model-agnostic interpretation methods for machine learning models. In: International Workshop on Extending Explainable AI Beyond Deep Models and Classifiers, pp. 39–68 (2020). Springer
20. Ribeiro, M.T., Singh, S., Guestrin, C.: " why should i trust you?" explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1135–1144 (2016)
21. Otin, S., Ávila, F.J., Mallen, V., Garcia-Martin, E.: Detecting structural changes in the retina of neurodegenerative patients through oct supervised segmentation (2023)
22. Khojaste-Sarakhsi, M., Haghighi, S.S., Ghomi, S.F., Marchiori, E.: Deep learning for alzheimer's disease diagnosis: a survey. Artif. Intell. Med. **130**, 102332 (2022)
23. Niyas, K.M., Thiyagarajan, P.: A systematic review on early prediction of mild cognitive impairment to alzheimers using machine learning algorithms. Int. J. Intell. Netw. https://doi.org/10.1016/j.ijin.2023.03.004 (2023)
24. Moradi, E., Pepe, A., Gaser, C., Huttunen, H., Tohka, J., Initiative, A.D.N., et al.: Machine learning framework for early MRI-based alzheimer's conversion prediction in mci subjects. Neuroimage **104**, 398–412 (2015)
25. Grimmer, T., Wutz, C., Alexopoulos, P., Drzezga, A., Förster, S., Förstl, H., Goldhardt, O., Ortner, M., Sorg, C., Kurz, A.: Visual versus fully automated analyses of 18f-FDG and amyloid pet for prediction of dementia due to alzheimer disease in mild cognitive impairment. J. Nucl. Med. **57**(2), 204–207 (2016)
26. Forlenza, O.V., Radanovic, M., Talib, L.L., Aprahamian, I., Diniz, B.S., Zetterberg, H., Gattaz, W.F.: Cerebrospinal fluid biomarkers in alzheimer's disease: diagnostic accuracy and prediction of dementia. Alzheimer's Dementia: Diagnosis Assess. Dis. Monit. **1**(4), 455–463 (2015)
27. Youssofzadeh, V., McGuinness, B., Maguire, L.P., Wong-Lin, K.: Multi-kernel learning with dartel improves combined MRI-PET classification of alzheimer's disease in AIBL data: group and individual analyses. Front. Hum. Neurosci. **11**, 380 (2017)
28. Saleem, T.J., Zahra, S.R., Wu, F., Alwakeel, A., Alwakeel, M., Jeribi, F., Hijji, M.: Deep learning-based diagnosis of alzheimer's disease. J. Pers. Med. **12**(5), 815 (2022)
29. Pescosolido, B.A., Perry, B.L., Krendl, A.C.: Empowering the next generation to end stigma by starting the conversation: bring change to mind and the college toolbox project. J. Am. Acad. Child Adolesc. Psychiatry **59**(4), 519–530 (2020)
30. Cuingnet, R., Gerardin, E., Tessieras, J., Auzias, G., Lehéricy, S., Habert, M.-O., Chupin, M., Benali, H., Colliot, O., Initiative, A.D.N., et al.: Automatic classification of patients with alzheimer's disease from structural MRI: a comparison of ten methods using the ADNI database. Neuroimage **56**(2), 766–781 (2011)
31. Wang, T., Qiu, R.G., Yu, M.: Predictive modeling of the progression of alzheimer's disease with recurrent neural networks. Sci. Rep. **8**(1), 9161 (2018)
32. Akter, S., Das, D., Haque, R.U., Tonmoy, M.I.Q., Hasan, M.R., Mahjabeen, S., Ahmed, M.: AD-CovNet: an exploratory analysis using a hybrid deep learning model to handle data imbalance, predict fatality, and risk factors in alzheimer's patients with covid-19. Comput. Biol. Med. **146**, 105657 (2022)
33. Zhang, Y., Wang, S., Sui, Y., Yang, M., Liu, B., Cheng, H., Sun, J., Jia, W., Phillips, P., Gorriz, J.M.: Multivariate approach for alzheimer's disease detection using stationary wavelet entropy and predator-prey particle swarm optimization. J. Alzheimers Dis. **65**(3), 855–869 (2018)
34. Xu, L., Wu, X., Chen, K., Yao, L.: Multi-modality sparse representation-based classification for alzheimer's disease and mild cognitive impairment. Comput. Methods Programs Biomed. **122**(2), 182–190 (2015)
35. Huang, S., Li, J., Ye, J., Wu, T., Chen, K., Fleisher, A., Reiman, E.: Identifying alzheimer's disease-related brain regions from multi-modality neuroimaging data using sparse composite linear discrimination analysis. Adv. Neural Inf. Process. Syst. **24** (2011)
36. Moore, P., Lyons, T., Gallacher, J., Initiative, A.D.N.: Random forest prediction of alzheimer's disease using pairwise selection from time series data. PLoS ONE **14**(2), 0211558 (2019)
37. Chincarini, A., Sensi, F., Rei, L., Gemme, G., Squarcia, S., Longo, R., Brun, F., Tangaro, S., Bellotti, R., Amoroso, N., et al.: Integrating longitudinal information in hippocampal volume measurements for the early detection of alzheimer's disease. Neuroimage **125**, 834–847 (2016)
38. Zhao, Y., Guo, Q., Zhang, Y., Zheng, J., Yang, Y., Du, X., Feng, H., Zhang, S.: Application of deep learning for prediction of alzheimer's disease in PET/MR imaging. Bioengineering **10**(10), 1120 (2023)
39. Bron, E.E., Smits, M., Van Der Flier, W.M., Vrenken, H., Barkhof, F., Scheltens, P., Papma, J.M., Steketee, R.M., Orellana, C.M., Meijboom, R., et al.: Standardized evaluation of algorithms for computer-aided diagnosis of dementia based on structural MRI: the caddementia challenge. Neuroimage **111**, 562–579 (2015)
40. Reynolds, E.L., Burke, J.F., Evans, L., Syed, F.I., Liao, E., Lobo, R., Cooper, W., Charleston, L., IV., Callaghan, B.C.: Headache neuroimaging: a survey of current practice, barriers, and facilitators to optimal use. Headache: J. Head Face Pain **62**(1), 36–56 (2022)
41. Donnelly-Kehoe, P.A., Pascariello, G.O., Gómez, J.C., Initiative, A.D.N., et al.: Looking for alzheimer's disease morphometric signatures using machine learning techniques. J. Neurosci. Methods **302**, 24–34 (2018)

42. Duc, N.T., Ryu, S., Qureshi, M.N.I., Choi, M., Lee, K.H., Lee, B.: 3D-deep learning based automatic diagnosis of alzheimer's disease with joint MMSE prediction using resting-state FMRI. Neuroinformatics **18**, 71–86 (2020)
43. Jack, C.R., Jr., Bennett, D.A., Blennow, K., Carrillo, M.C., Dunn, B., Haeberlein, S.B., Holtzman, D.M., Jagust, W., Jessen, F., Karlawish, J., et al.: Nia-aa research framework: toward a biological definition of alzheimer's disease. Alzheimer's Dementia **14**(4), 535–562 (2018)
44. El-Sappagh, S., Saleh, H., Ali, F., Amer, E., Abuhmed, T.: Two-stage deep learning model for alzheimer's disease detection and prediction of the mild cognitive impairment time. Neural Comput. Appl. **34**(17), 14487–14509 (2022)
45. Elshawi, R., Maher, M., Sakr, S.: Automated machine learning: State-of-the-art and open challenges. arXiv preprint arXiv:1906.02287 (2019)
46. Bucholc, M., Titarenko, S., Ding, X., Canavan, C., Chen, T.: A hybrid machine learning approach for prediction of conversion from mild cognitive impairment to dementia. Expert Syst. Appl. **217**, 119541 (2023)
47. Kumar, M.S., Azath, H., Velmurugan, A., Padmanaban, K., Subbiah, M.: Prediction of alzheimer's disease using hybrid machine learning technique. In: AIP Conference Proceedings, vol **2523** (2023). AIP Publishing
48. Bitirgen, K., Filik, Ü.B.: A hybrid deep learning model for discrimination of physical disturbance and cyber-attack detection in smart grid. Int. J. Crit. Infrastruct. Prot. **40**, 100582 (2023)
49. Zhang, X., Han, L., Zhu, W., Sun, L., Zhang, D.: An explainable 3d residual self-attention deep neural network for joint atrophy localization and alzheimer's disease diagnosis using structural mri. IEEE J. Biomed. Health Inform. **26**(11), 5289–5297 (2021)
50. Vu, T.-D., Ho, N.-H., Yang, H.-J., Kim, J., Song, H.-C.: Non-white matter tissue extraction and deep convolutional neural network for alzheimer's disease detection. Soft. Comput. **22**, 6825–6833 (2018)
51. Balaji, P., Chaurasia, M.A., Bilfaqih, S.M., Muniasamy, A., Alsid, L.E.G.: Hybridized deep learning approach for detecting alzheimer's disease. Biomedicines **11**(1), 149 (2023)
52. Alia, S., Abuhmedb, T., El-Sappaghb, S., Muhammadc, K., Alonso-Morald, J.M., Confalonierie, R., Guidottif, R., Del Serg, J., Díaz-Rodríguezi, N., Herrerai, F.: Explainable artificial intelligence (XAI): What we know and what is left to attain trustworthy artificial intelligence
53. ADNI. http://adni.loni.usc.edu/methods/documents/. Online; accessed (2023)
54. Raket, L.L.: Statistical disease progression modeling in alzheimer disease. Front. Big Data **3**, 24 (2020)
55. Kukačka, J., Golkov, V., Cremers, D.: Regularization for deep learning: a taxonomy. arXiv preprint arXiv:1710.10686 (2017)
56. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**(1), 1929–1958 (2014)
57. Wu, J., Chen, X.-Y., Zhang, H., Xiong, L.-D., Lei, H., Deng, S.-H.: Hyperparameter optimization for machine learning models based on Bayesian optimization. J. Electron. Sci. Technol. **17**(1), 26–40 (2019)
58. Pelikan, M., Pelikan, M.: Bayesian optimization algorithm. In: Hierarchical Bayesian optimization algorithm: toward a new generation of evolutionary algorithms, 31–48 (2005)
59. Ding, S., Su, C., Yu, J.: An optimizing BP neural network algorithm based on genetic algorithm. Artif. Intell. Rev. **36**, 153–162 (2011)
60. Shapiro, J.: Genetic algorithms in machine learning. In: Advanced course on artificial intelligence, pp. 146–168. Springer, Berlin, Heidelberg (1999)
61. Alam, T., Qamar, S., Dixit, A., Benaida, M.: Genetic algorithm: Reviews, implementations, and applications. arXiv preprint arXiv:2007.12673 (2020)
62. Kallem, S.R.: Artificial intelligence algorithms. IOSR J. Comput. Eng. (IOSRJCE) **6**(3), 1–8 (2012)
63. Prechelt, L.: Early stopping-but when? lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics) 7700 lecture no (2012)
64. Liang, Y., Li, S., Yan, C., Li, M., Jiang, C.: Explaining the black-box model: a survey of local interpretation methods for deep neural networks. Neurocomputing **419**, 168–182 (2021)
65. Tjoa, E., Guan, C.: A survey on explainable artificial intelligence (XAI): toward medical XAI. IEEE Trans. Neural Netw. Learn. Syst. **32**(11), 4793–4813 (2020)
66. Kingma, D., Ba, J.L.: Adam: a method for stochastic optimization 3rd int. conf. learn. Representations (Preprint 1412.6980 v9) (2015)
67. Basheera, S., Ram, M.S.S.: A novel CNN based alzheimer's disease classification using hybrid enhanced ICA segmented gray matter of MRI. Comput. Med. Imaging Graph. **81**, 101713 (2020)
68. Divya, R., Shantha Selva Kumari, R., Initiative, A.D.N.: Genetic algorithm with logistic regression feature selection for alzheimer's disease classification. Neural Comput. Appl. **33**(14), 8435–8444 (2021)
69. Pan, X., Phan, T.-L., Adel, M., Fossati, C., Gaidon, T., Wojak, J., Guedj, E.: Multi-view separable pyramid network for ad prediction at mci stage by 18^f-FDG brain pet imaging. IEEE Trans. Med. Imaging **40**(1), 81–92 (2020)
70. Chen, Y., Xia, Y.: Iterative sparse and deep learning for accurate diagnosis of alzheimer's disease. Pattern Recogn. **116**, 107944 (2021)
71. Poloni, K.M., Oliveira, I.A.D., Tam, R., Ferrari, R.J., Initiative, A.D.N., et al.: Brain MR image classification for alzheimer's disease diagnosis using structural hippocampal asymmetrical attributes from directional 3-D log-Gabor filter responses. Neurocomputing **419**, 126–135 (2021)
72. Buvaneswari, P., Gayathri, R.: Deep learning-based segmentation in classification of alzheimer's disease. Arab. J. Sci. Eng. **46**, 5373–5383 (2021)
73. Li, A., Li, F., Elahifasaee, F., Liu, M., Zhang, L., Initiative, A.D.N.: Hippocampal shape and asymmetry analysis by cascaded convolutional neural networks for alzheimer's disease diagnosis. Brain Imag. Behav. 1–10 (2021)
74. Puente-Castro, A., Fernandez-Blanco, E., Pazos, A., Munteanu, C.R.: Automatic assessment of alzheimer's disease diagnosis based on deep learning techniques. Comput. Biol. Med. **120**, 103764 (2020)
75. Abrol, A., Bhattarai, M., Fedorov, A., Du, Y., Plis, S., Calhoun, V., Initiative, A.D.N., et al.: Deep residual learning for neuroimaging: an application to predict progression to alzheimer's disease. J. Neurosci. Methods **339**, 108701 (2020)
76. Hedayati, R., Khedmati, M., Taghipour-Gorjikolaie, M.: Deep feature extraction method based on ensemble of convolutional auto encoders: application to alzheimer's disease diagnosis. Biomed. Signal Process. Control **66**, 102397 (2021)
77. Chitradevi, D., Prabha, S.: Analysis of brain sub regions using optimization techniques and deep learning method in alzheimer disease. Appl. Soft Comput. **86**, 105857 (2020)