Neurocomputing 412 (2020) 197–215

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Multimodal multitask deep learning model for Alzheimer's disease progression detection based on time series data

Shaker El-Sappagh [a,e,1], Tamer Abuhmed [b,1], S.M. Riazul Islam [c], Kyung Sup Kwak [d,*]

[a] Centro Singular de Investigación en Tecnoloxías Intelixentes (CiTIUS), Universidade de Santiago de Compostela, 15782, Santiago de Compostela, Spain
[b] College of Computing, Sungkyunkwan University, Republic of Korea
[c] Department of Computer Science and Engineering, Sejong University, Republic of Korea
[d] Department of Information and Communication Engineering, Inha University, Republic of Korea
[e] Information Systems Department, Faculty of Computers and Artificial Intelligence, Benha University, Banha 13518, Egypt

ARTICLE INFO

Article history:
Received 13 November 2019
Revised 6 April 2020
Accepted 25 May 2020
Available online 1 June 2020
Communicated by Zhang Zhaoxiang

Keywords:
Alzheimer's disease
Progression detection
Multimodal multitask learning
Deep learning
Machine learning
Time series data analysis

ABSTRACT

Early prediction of Alzheimer's disease (AD) is crucial for delaying its progression. As a chronic disease, ignoring the temporal dimension of AD data affects the performance of a progression detection and medically unacceptable. Besides, AD patients are represented by heterogeneous, yet complementary, multimodalities. Multitask modeling improves progression-detection performance, robustness, and stability. However, multimodal multitask modeling has not been evaluated using time series and deep learning paradigm, especially for AD progression detection. In this paper, we propose a robust ensemble deep learning model based on a stacked convolutional neural network (CNN) and a bidirectional long short-term memory (BiLSTM) network. This multimodal multitask model jointly predicts multiple variables based on the fusion of five types of multimodal time series data plus a set of background (BG) knowledge. Predicted variables include AD multiclass progression task, and four critical cognitive scores regression tasks. The proposed model extracts local and longitudinal features of each modality using a stacked CNN and BiLSTM network. Concurrently, local features are extracted from the BG data using a feed-forward neural network. Resultant features are fused to a deep network to detect common patterns which jointly used to predict the classification and regression tasks. To validate our model, we performed six experiments on five modalities from Alzheimer's Disease Neuroimaging Initiative (ADNI) of 1536 subjects. The results of the proposed approach achieve state-of-the-art performance for both multiclass progression and regression tasks. Moreover, our approach can be generalized in other medial domains to analyze heterogeneous temporal data for predicting patient's future status.

1. Introduction

Alzheimer's disease (AD) accounts for 60% to 70% of dementia in seniors, and 115.4 million people are expected to have AD in 2050 [1]. There is no cure for AD, and current treatments only decelerate its progression [2]. As a result, its early prediction is of fundamental importance for timely treatment and progression delay. Mild cognitive impairment (MCI) is a broad, ill-defined, highly heterogeneous phenotypic spectrum that causes a relatively less noticeable memory deficiency than AD [3]. Around 10% to 20% of MCI patients progress to AD per year [4]. The gradual change from MCI to AD takes years, if not decades [5]. It is a challenging task to identify stable MCI (sMCI) patients who do not progress to AD, and progressive MCI (pMCI) patients who will later have AD [6]. Machine learning (ML) techniques can play a critical role in helping medical experts analyze patient data. AD symptomatology is multimodal and longitudinal [7,8]. The patient data comprise a collection of heterogeneous, yet complementary, data of different types including magnetic resonance imaging (MRI), positron emission tomography (PET), genetics, cerebrospinal fluid (CSF), *etc.* [9]. The combination of multimodalities facilitates detecting and distinguishing all subtle changes in the patient's progression status, and supports reliable diagnoses [5]. Over the past decade, regular ML algorithms (especially the support vector machine [SVM] and random forest [RF]) have been utilized for MCI conversion prediction [9–13]. Most studies utilized single-modality and single-task models, such as sMCI vs. pMCI classification [5,14] or cognitive score regression [15]. This design paradigm is called single-

* Corresponding author.
E-mail addresses: shaker.elsappagh@usc.es (S. El-Sappagh), tamer@skku.edu (T. Abuhmed), riaz@sejong.ac.kr (S.M. Riazul Islam), kskwak@inha.ac.kr (K.S. Kwak).
[1] These authors contributed equally to this work.

https://doi.org/10.1016/j.neucom.2020.05.087
0925-2312/

modality single-task, where the model only optimizes a single objective function based on one type of data [16]. In these models, neither the correlation among tasks nor the complementary information across modalities are explored [17]. It has been shown that multimodal systems usually yield comprehensive insights, more accurate results, more stable behaviors, and are consequently more acceptable in the medical side [13,18–20]. Liu et al. [21] and Duchesne et al. [22] used regular machine learning techniques to study multimodal single-task classification and regression, respectively. Multimodal data could be fused in different ways, and selecting the best modality combination and suitable fusion scheme is a challenging task [13,23]. Besides, the single-task models lack the ability to provide useful knowledge to medical experts regarding the possible cognitive behavior of the patient at the time of progression. Some studies design MCI progression as multimodal single-task regression models, where some cognitive scores, such as the mini-mental state examination (MMSE) and the Alzheimer's diseases assessment scale (ADAS), are indicators for AD progression [22,24]. Contrarily, Zhou et al. [25] studied AD progression as a single-modal multitask regression problem. However, in real medical environments, many modalities are chronically analyzed, and multiple clinical variables have to be predicted. The ML models that are able to do this job are called multimodal multitask models, where every task has features from multiple sources, and multiple tasks are related in a chronological sequence [7,8,17,26]. Zhang and Shen [9] proposed an SVM-based method to jointly predict multiple medical scores (*i.e.,* MMSE, ADAS, and diagnosis features) by fusing multimodal data (*i.e.,* MRI, PET, and CSF). Recently, Ding et al. [8] asserted that most AD studies consider only a limited number of factors, which is potentially insufficient for understanding the complex and multifactorial nature of the disease. Most studies consider MCI progression as a binary sMCI vs. pMCI classification problem based exclusively on baseline data [4]. This is a suboptimal strategy because baseline data are less discriminative for progression detection than considering a patient's longitudinal data, which in turn results in less accurate models. Because AD is a chronic disease, the patient's data are always time series in nature. Patient data are accumulated from different visits and form continuous patient supervision. The disease state at a certain point in time is not independent of the state at a previous point in time. As a result, AD data are not only multimodal but also time series. Consequently, considering AD multimodal data as a time series is the intuitive solution for the AD progression problem. However, the vast majority of research does not consider this temporal/sequential nature of AD data [1]. Some work in the literature adopted traditional time series algorithms for the AD progression detection problem [27,28], and the correlation between the patient's multimodal data and how they evolve has not been analyzed [6]. Recently, Li et al. [4] asserted the urgency of multimodality and longitudinal analysis of AD data. Multimodal multitask modeling of AD progression based on time series data is a challenge that promises great improvement in models' performance, because multitask learning acts as a regularizer for all tasks [29]. Besides, most AD classifiers, such as the SVM, are based on the two independent steps of dimensionality reduction and classification. These two models are mathematically independent and involve different assumptions. Additionally, these techniques require the use of kernels that are chosen from a pre-specified set. Recently, deep learning (DL) techniques have demonstrated promising prediction results in several areas [30,31]. All previous challenges could be effectively managed by using DL [5,32–37]. In the AD context, Choi and Jin [38] utilized a CNN to detect pMCI cases based on a single-source (PET images) single-task model. Spasov et al. [39] proposed a multimodal single-task classification model based on a CNN to detect AD progression based on the late fusion of MRI, demographics, neuropsychological, and apolipoprotein E4 (APOe4) genetic data. Liu et al. [7] proposed a CNN-based model for joint AD classification and clinical score regression. The model is based on the fusion of MRI with three demographic features collected from baseline visits only. Most of the Alzheimer's DL models are based on the CNN and single (baseline) MRI scans [5]. These models are less accurate, less sufficient, and not medically acceptable, because a medical expert usually studies the longitudinal multimodal patient data before making progression decisions [8]. In this paper, we investigate the effect on prediction performance and progression by using time series multimodality data of AD patients. We exploit the CNN and recurrent neural network (RNN) to capture the local and long-term temporal dependencies, respectively [40]. Wang et al. [40] proposed a long short-term memory (LSTM)-based regression model to predict AD progression from time series data with non-uniform visit-time intervals. Unlike the AD progression problem, advanced DL techniques based on the combination of CNN and RNN models have been proposed in different fields of industry [26,30,31,41,42], and have achieved superior performance compared to the non deep learning techniques. Cui et al. [43] proposed a CNN-BiLSTM model for AD diagnosis based on MRI time series data of six time steps. Most DL models in the AD domain are implemented as binary classifications, but multiclass models are still far from reaching satisfactory results for clinical applicability [44]. Because AD data are complex, DL models based on combined CNN-BiLSTM could outperform models based on CNN or LSTM alone. In addition, increasing the number of time steps used in longitudinal data improves system performance [43]. Our hypothesis is that multimodal joint prediction of multiple categorical and continuous variables based on late fusion of time series and static data could perform better than predicting each individual variable separately. The main contributions of this work can be summarized as follows.

- We propose an advanced multimodal multitask DL architecture for detecting AD progression. The framework leverages the patient's time series data to jointly predict multiple variables from multiple sources. The resulting comprehensive system is medically intuitive, more stable, and more accurate than existing state-of-the-art studies. To the best of our knowledge, no prior studies have investigated the way to integrate the multimodal and multitask architecture based on time series data to create a personalized, accurate, and medically trusted AD progression model using deep learning.
- The proposed model jointly learns to simultaneously predict the patient's progression status and the values of four critical cognitive scores at the time of progression. The predicted clinical scores are ADAS, MMSE, the functional assessment questionnaire (FAQ), and the clinical dementia rating sum of boxes (CDRSB) score, which are implemented as four regression tasks. Progression detection is a multiclass classification task (*i.e.,* cognitive normal [CN] vs. sMCI vs. pMCI vs. AD). Medically, these related tasks share common relevant feature subsets.
- Compared with previous studies, our model is the first attempt to extract temporal features from five heterogeneous data sources and a set of static baseline features. Each time series source is separately learned using a pipeline of stacked CNN-BiLSTM blocks. The CNN automatically extracts local features from each time series, and LSTM extracts temporal features from each feature and from temporal relationships among the features. Then, the learned features from all modalities are fused to extract context-aware common features.
- Since little effort has gone into exploring the role of static data as background knowledge (BG) to improve model performance [45], we studied the effect of these data on the performance of our model. To prepare the BG data, several types of baseline data were collected from the patients' first visits (*e.g.,* age, gen-

Table 1
Descriptive statistics from the used dataset.

	CN		MCI (n = 778)				AD (n = 339)	
	(n = 419)		sMCI (n = 473)		pMCI (n = 305)			
	Baseline	M84	Baseline	M84	Baseline	M84	Baseline	M84
Gender (M/F)	191/228	191/228	283/190	283/190	179/126	179/126	187/152	187/152
Age (years)	73.84 ± 05.78	73.84 ± 05.78	72.92 ± 07.76	72.92 ± 07.76	73.95 ± 07.02	73.95 ± 07.02	75.01 ± 07.81	75.01 ± 07.81
Education (y)	16.43 ± 02.70	16.43 ± 02.70	15.80 ± 02.97	15.80 ± 02.97	15.93 ± 02.78	15.93 ± 02.78	15.13 ± 02.98	15.13 ± 02.98
FAQ	00.19 ± 00.73	00.41 ± 01.87	02.10 ± 03.13	03.69 ± 05.14	05.38 ± 04.84	18.83 ± 08.09	13.32 ± 06.85	18.80 ± 07.71
MMSE	28.98 ± 01.14	28.87 ± 01.32	27.63 ± 02.13	27.06 ± 02.72	26.32 ± 02.27	20.48 ± 05.62	21.94 ± 03.64	20.00 ± 05.41
MoCA	25.68 ± 01.97	24.84 ± 02.73	23.14 ± 02.70	22.50 ± 03.15	21.28 ± 02.08	17.19 ± 05.05	17.48 ± 03.54	16.59 ± 05.20
FDG	06.56 ± 00.50	06.53 ± 00.51	06.33 ± 00.59	06.24 ± 00.63	05.97 ± 00.50	05.22 ± 00.50	05.32 ± 00.60	05.194 ± 00.58
APOe4	00.27 ± 00.48	00.27 ± 00.48	00.51 ± 00.66	00.51 ± 00.66	00.84 ± 00.69	00.84 ± 00.69	00.85 ± 00.71	00.85 ± 00.71
p-TAU (pg/mL)	22.77 ± 07.69	22.77 ± 07.69	26.00 ± 12.96	26.00 ± 12.96	32.89 ± 14.09	32.89 ± 14.09	37.07 ± 13.14	37.07 ± 13.14
TAU	240.0 ± 77.19	240.0 ± 77.19	271.64 ± 118	271.64 ± 118	330.08 ± 121	330.08 ± 121	371.2 ± 120.7	371.2 ± 120.7
ADAS 11	05.64 ± 02.83	06.24 ± 03.18	09.13 ± 3.91	10.89 ± 06.41	12.92 ± 04.42	23.87 ± 12.00	19.64 ± 06.74	24.97 ± 11.84
ADAS 13	08.70 ± 04.09	09.53 ± 04.80	14.80 ± 05.84	17.08 ± 08.87	20.86 ± 06.11	34.53 ± 14.26	30.00 ± 07.99	35.86 ± 13.68
RAVLT imm.	45.80 ± 09.72	45.32 ± 10.95	36.41 ± 10.65	33.46 ± 12.06	28.80 ± 07.54	18.90 ± 08.74	22.64 ± 07.47	18.60 ± 08.56
RAVLT learn	06.03 ± 02.19	05.60 ± 02.44	04.57 ± 02.50	03.89 ± 02.65	02.98 ± 02.28	01.65 ± 01.81	01.83 ± 01.77	01.57 ± 01.74
RAVLT forget	03.66 ± 02.73	03.34 ± 02.90	04.42 ± 02.51	04.50 ± 02.40	05.01 ± 02.17	03.95 ± 02.12	04.45 ± 01.83	03.96 ± 02.08
RAVLT % forget	33.47 ± 26.89	32.97 ± 29.90	53.70 ± 31.37	62.80 ± 34.21	75.96 ± 28.29	90.98 ± 33.47	89.44 ± 20.87	93.41 ± 25.24
CDR	00.084 ± 0.30	00.17 ± 00.86	01.37 ± 00.86	01.77 ± 01.49	02.13 ± 00.99	07.23 ± 03.89	05.34 ± 02.21	07.19 ± 03.66
AV45	01.29 ± 00.19	01.31 ± 00.21	01.37 ± 00.23	01.38 ± 00.24	01.47 ± 00.20	01.57 ± 00.17	01.58 ± 00.18	01.58 ± 00.18
HCI	08.92 ± 03.58	09.09 ± 03.62	11.40 ± 04.05	12.02 ± 04.96	14.23 ± 04.95	22.91 ± 05.81	22.01 ± 07.28	23.33 ± 07.14
Hippo. vol.	07.47 ± 00.94	07.21 ± 01.02	06.96 ± 01.10	06.70 ± 01.16	06.19 ± 01.02	05.39 ± 00.10	05.74 ± 01.02	05.45 ± 01.09

*Data are mean ± standard deviation.

der, CSF, symptoms, *etc.*) plus a set of statistical measures extracted from time series data. These features are fed into the model, where they are simultaneously learned by a separate feed-forward neural network. Then, the learned deep features are again fused with the modalities' learned features to jointly a predict patient's progression status and cognitive scores.

- We conducted extensive experiments to evaluate our model in different settings using a dataset of 1536 patient samples from the Alzheimer's disease neuroimaging initiative (ADNI) database. Six experiments were implemented and tested. We proved that: (1) deep-stacked CNN-BiLSTM is more accurate than a concatenated CNN-BiLSTM network structure; (2) late fusion of learned features from time series and BG knowledge achieved better performance, compared to an early fusion structure; (3) a multitask model of classification and regression tasks produced a more stable and more accurate system than single-task models; (4) adding more data to the DL model (*i.e.,* more time steps to the same modality, or more modalities) enhanced its performance even if the data were noisy; (5) based on an extensive modal-selection process, MRI and PET are the most informative modalities; and finally, (6) the statistical features extracted from time series data are more important than baseline data.

The remainder of this paper is organized as follows. Section 2 presents the methodology of the study and the proposed model. Section 3 presents the results of the study, and Section 4 features a discussion about the study findings findings and limitations, offering some future directions for research. Finally,the conclusion is in Section 5.

2. Materials and method

2.1. Study cohorts

Data used in the current study were accessed on March 18, 2019, from ADNI 1, ADNI GO, and ADNI 2. The study included 1536 subjects (54.7% male) categorized into four groups based on the individual clinical diagnosis at baseline and future time points (see Supplementary File 1). The participants are categorized into four categories. The first category includes 419 subjects diagnosed CN at baseline and who remained CN at the time this study was prepared. The second category includes 473 subjects diagnosed as stable MCI at all time points of the study. The third category includes 305 subjects evaluated to be progressive MCI at the baseline visit and who progressed to AD at some point in time during the study (84 months long). Finally, the fourth category includes 339 subjects with a clinical diagnosis of AD in all visits. Subjects showing improvement in their clinical diagnosis during follow up, *i.e.,* those clinically diagnosed as MCI but who reverted to CN, or those clinically diagnosed as AD but who reverted to MCI or CN, were excluded from the study due to the potential uncertainty of the clinical diagnosis, considering that AD is an irreversible form of dementia. Moreover, cases that showed a direct conversion from CN to AD were also removed. The full details of the used subjects in our study can be found in Supplementary File 2. Demographic and clinical information on the subjects is in Table 1.

2.2. Proposed deep learning model

A high-level overview of the proposed network design is shown in Fig. 1A. The proposed model is designed for a multimodal multitask purpose to learn AD progression and the four cognitive scores based on multivariate time series data. Initially, five time series modalities with 15 regular time steps (*i.e.,* baseline, M06, M12,. . ., M84) were fed separately into the model, along with BG data. The local and temporal feature learning of our model are based on a stacked CNN and BiLSTM subnetworks. As shown in Fig. 1A, the model initially prepares the time series data of five modalities. The neuroimaging features of MRI and PET modalities are extracted using the FreeSurfer. The extracted features from modalities (*i.e.,* cognitive scores, neuropsychological battery, MRI, PET, and assessment modalities) are preprocessed to improve the quality of the data, as discussed in Section 2.3. For features reduction, PCA technique is used to extract the principal components form the high dimensional MRI and PET data. Then, the deep features are separately learned from each time series multimodality by using a stacked CNN-BiLSTM model. Section 2.2.1 and Section 2.2.2 discuss the role of CNN and BiLSTM modules, respectively. The abstract deep features learned from the previous step are fused to extract common features from all modalities using a set of dense layers. Concurrently, the baseline background data are preprocessed then learned using a set of dense layers to extract

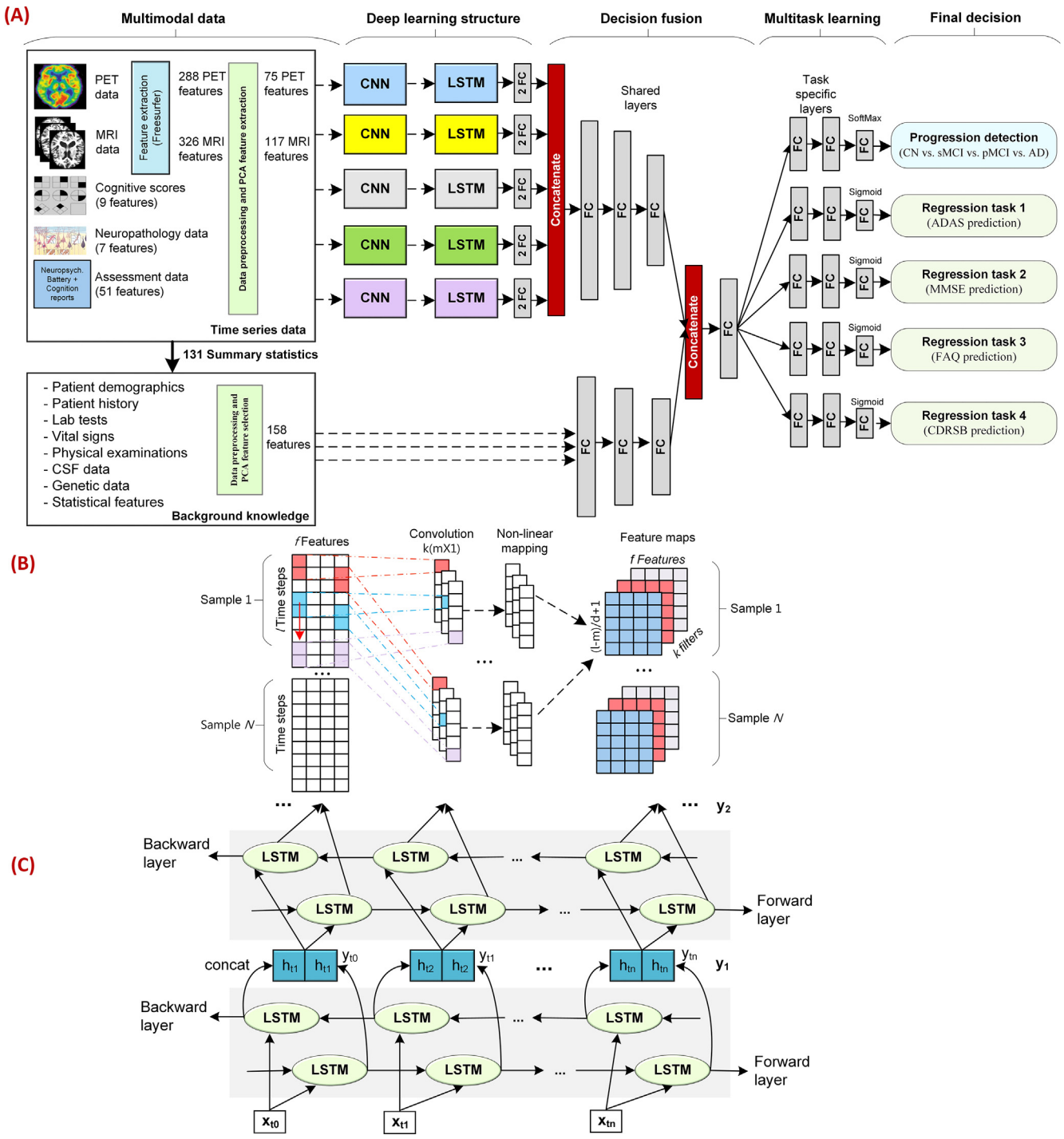

Fig. 1. Schematic of the modeling architecture. (A) The proposed ensemble multimodal multitask DL framework. (B) The CNN network to learn local features using Conv1D layer. (C) The architecture of BiLSTM network.

representative deep features. Getting the common deep features of the time series modalities and the representative deep features of the background data, a second decision fusion by a dense layer is used to get more abstract deep features. The final step is the task specific learning, where a set of dense layers are used to learn task specific features; then, the Softmax or Sigmoid is used for the classification or regression task, respectively. Fig. S2 in Supplementary File 2 provides the full details of the proposed model structure and Section 3.2 provides the hyperparamters of the trained model.

2.2.1. Convolutional neural network

As illustrated in Fig. 1A, a separate CNN subnetwork is used for learning each modality. The CNN for time series introduces 1D convolution (Conv1D), which can learn univariate time series data. Convolution is done separately along the time dimension for every input vector (see Fig. 1B). Formally, if input vector $x \in \mathbb{R}^{l\times 1}$ and kernel r is $m \times 1$, then the Conv1D maps x to a new feature space, $\dot{x} \in \mathbb{R}^{[(l-m)/d+1,1]}$, where d is the step size. Based on the number of filters, the CNN expands every univariate time series to more abstract

and informative features, called feature maps, which are more suitable for LSTM prediction. Each value f_i of feature map f is then fed into an activation function, g, to calculate $f_i = g(r^T \times x^{(i:i+j-1)} + b)$, where g is a non-linear activation function ($ReLU(x) = \max(0, x)$ in our case), b is bias, and $x^{(i:i+j-1)}$ are j observations from$x$. For each modality, we propose to use one CNN layer to separately transform the time series of multiple tensors into a new feature space. For any modality $M \in \mathbb{R}^{n\times l\times f}$, by using k filters of $(m \times 1)$, the corresponding output tensor is $\grave{M} \in \mathbb{R}^{n\times\left(\frac{l-m}{d}+1\right)\times f\times k}$, where n is the number of samples, l is the number of time steps, and f is the number of features. A max pooling layer is used to smooth the input, prevent overfitting, and learn higher-level abstractions.

2.2.2. Long short-term memory

To benefit from the temporal correlation of time series data, we added BiLSTM layers to find temporal patterns from longitudinal data. The input to the BiLSTM block is the learned features from CNN layer. Each LSTM unit in Fig. 1C has the internal structure represented in Fig. S1 in Supplementary File 2. The core structure of the LSTM cell is the use of three gates: the input gate (i_{t_n}), the forget gate (f_{t_n}), and the output gate (o_{t_n}). These gates control the update, maintenance, and deletion of information contained in a cell state. C_{t_n}, $C_{t_{n-1}}$, and $\tilde{C}_{t_n}$,respectively, are the current cell status value at time t_n, the last time step cell status value, and the update of the current cell status value; $h_{t_{n-1}}$ is the value output 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 $C_{t_{n-1}}$, and the θs and the bs are the set of weight matrices and biases vectors, respectively, updated following the backpropagation through time algorithm. In addition, $\otimes$ represents the Hadamard product; σ is the standard logistic sigmoid function; $\oplus$ is the concatenation operator; and $\varphi$is the output activation function, *e.g., SoftMax* or *Tanh*. Eqs. (1)–(7) give the transmission of information in the memory cell at each step.

$$f_{t_n} = \sigma\left(\theta_f \bullet \left[h_{t_{n-1}}, x_{t_n}\right] + b_f\right) \tag{1}$$

$$i_{t_n} = \sigma\left(\theta_i \bullet \left[h_{t_{n-1}}, x_{t_n}\right] + b_i\right) \tag{2}$$

$$\tilde{C}_{t_n} = tanh\left(\theta_C \bullet \left[h_{t_{n-1}}, x_{t_n}\right] + b_C\right) \tag{3}$$

$$C_{t_n} = \left(f_{t_n} \otimes C_{t_{n-1}} \oplus i_{t_n} \otimes \tilde{C}_{t_n}\right) \tag{4}$$

$$o_{t_n} = \sigma\left(\theta_o \bullet \left[h_{t_{n-1}}, x_{t_n}\right] + b_o\right) \tag{5}$$

$$h_{t_n} = o_{t_n} \otimes \tanh\left(C_{t_n}\right) \tag{6}$$

$$y_n = \varphi\left(\theta_y h_{t_n} + b_y\right) \tag{7}$$

Single LSTM captures only the previous context, but does not utilize the future context. BiLSTM [46] combines two separate hidden LSTM layers of opposite directions to the same output. BiLSTM processes an input sequence, $X = (X_{t0}, X_{t1}, \ldots, X_{tn})$, from the opposite direction to a forward hidden sequence, $\overrightarrow{h}_t = \left(\overrightarrow{h}_{t0}, \overrightarrow{h}_{t1}, \ldots, \overrightarrow{h}_{tn}\right)$, and a backward hidden sequence, $\overleftarrow{h}_t = \left(\overleftarrow{h}_{t0}, \overleftarrow{h}_{t1}, \ldots, \overleftarrow{h}_{tn}\right)$. The output vector of hidden layer $y_t = (y_{t0}, y_{t1}, \ldots, y_{tn})$, $t = 1, 2, \ldots, t$ is the concatenation of $\overrightarrow{h}_t$ and $\overleftarrow{h}_t, y_t = \left[\overrightarrow{h}_t, \overleftarrow{h}_t\right]$, as shown in Eqs. (8)–(11).

$$\overrightarrow{h}_{tn} = \sigma\left(\theta_{\overrightarrow{h}_n} \bullet \left[\overrightarrow{h}_{t_{n-1}}, x_{t_n}\right] + b_{\overrightarrow{h}_n}\right) \tag{8}$$

$$\overleftarrow{h}_{tn} = \sigma\left(\theta_{\overrightarrow{h}_n} \bullet \left[\overleftarrow{h}_{t_{n-1}}, x_{t_n}\right] + b_{\overrightarrow{h}_n}\right) \tag{9}$$

$$\left(\overrightarrow{h}_{t0}, \overleftarrow{h}_{t0}\right) \ldots \left(\overrightarrow{h}_{tn}, \overleftarrow{h}_{tn}\right) = BiLSTM(X_{t0}, X_{t1}, \ldots, X_{tn}) \tag{10}$$

$$y_t = \sigma\left(\theta_{y_t \overrightarrow{h}_n} \overrightarrow{h}_{tn} + \theta_{y_t \overleftarrow{h}_n} \overleftarrow{h}_{tn} + b_{y_t}\right) \tag{11}$$

Then, the output y_t is used as input to the next hidden layer and so on. Each BiLSTM block in Fig. 1A is structured with three stacked BiLSTM layers, an L2 regularization layer, and a dropout layer. Output y_t from the lower layer becomes the input to the upper layer, as seen in Fig. 1C. The three BiLSTM layers will not increase the computation load, because our time series are not very long. The CNN (before the BiLSTM) performs a preprocessing step to learn local features, and its results are a shorter series with high-level features. A separate BiLSTM subnetwork is trained for a different modality, i.e. $X \in \{X_{PET}, X_{MRI}, X_{CSD}, X_{NPD}, X_{ASD}\}$.

2.2.3. The model's multitask cost function

The model has five concurrent stacked CNN-BiLSTM pipelines plus one feed-forward neural network. For each CNN-BiLSTM pipeline, the CNN block has one Conv1D layer followed by max pooling. The BiLSTM block has three stacked BiLSTM layers, an L2 regularization layer, and a dropout layer. The model is based on the late fusion of five different sources of temporal data, including neuroimaging, neuropsychological battery, etc. (see Supplementary File 2). The CNN subnetwork is applied to extract local features in each time series feature, as illustrated in Fig. 1B. Following that, the stacked BiLSTM subnetwork is applied to learn the temporal relationships within a single time series and among features of the same modality as illustrated in Fig. 1C. Then, learned features from the BiLSTM block are input to 2 dense layers for deeper feature learning. The output of the five streams is then fused by three dense layers to form more distinctive and deeper features. In addition, baseline data play the role of BG to enhance the accuracy and confidence of the learning process. These baseline data are the patient's static features, such as demographics and some statistical features extracted from his/her longitudinal time series data. Other distinctive and deep features are extracted from the baseline data separately using a feed-forward neural network. The results of these two feature extraction steps are fused by a set of shared dense layers to learn more fine common features for the classification and regression tasks. The proposed model concurrently learns many related tasks including a multiclass classification problem (*i.e.,* AD progression) and four regression problems (*i.e.,* the most medically sensitive cognitive scores related to Alzheimer's disease [7]). *We expect that this type of information is critical for physicians in order to trust the results of the DL model.* Our model exploits parameter sharing of the DL network for multitask learning. It is applied by sharing the hidden layers between all tasks, as illustrated in Fig. 1A. Multitask learning works as a regularizer, and reduces the risk of overfitting. Resulting models are more general and more stable than single-task models [47].

The proposed DL framework jointly learns a set of related tasks, Y, based on a set of modalities, M. Consider having M modalities of data represented as $X = \left\{\mathrm{X}^{(1)}, \ldots, \mathrm{X}^{(\mathrm{M})}\right\}$, and having multitasks to be learned represented as $\mathrm{Y} = \left\{\mathrm{y}^1, \ldots, \mathrm{y}^{(\mathrm{T})}\right\}$. Each *jth* task is $\mathrm{y}^{(\mathrm{j})} = \left\{\mathrm{y}_1^{(\mathrm{j})}, \ldots, \mathrm{y}_N^{(\mathrm{j})}\right\}$, $j \in \{1, \ldots, T\}$. Each modality X^{m} is represented as $\mathrm{X}^{\mathrm{m}} = \left\{\mathrm{x}_1^{(\mathrm{m})}, \ldots \mathrm{x}_{\mathrm{i}}^{(\mathrm{m})}, \ldots, \mathrm{x}_{\mathrm{N}}^{(\mathrm{m})}\right\}$ from N patient examples, and each example $\mathrm{x}_{\mathrm{i}}^{(\mathrm{m})} \in \mathbb{R}^{t\times f}$ is a multivariate time series, $\mathrm{x}_{\mathrm{i}}^{(\mathrm{m})} = \left\{\mathrm{x}_{\mathrm{it}_1}^{(\mathrm{m})}, \mathrm{x}_{\mathrm{it}_2}^{(\mathrm{m})}, \ldots, \mathrm{x}_{\mathrm{it}_f}^{(\mathrm{m})}\right\}$, for $t = 1, \ldots, s$ time-steps and f sets of univariate time series.

For N patients, each patient i is represented as $\mathrm{x_i} = \left\{\mathrm{x}_{\mathrm{i}}^{(1)}, \ldots \mathrm{x}_{\mathrm{i}}^{(\mathrm{m})}, \ldots, \mathrm{x}_{\mathrm{i}}^{(M)}, \mathrm{y}_i^1, \quad \mathrm{y}_i^2, \ldots, \mathrm{y}_i^T\right\}$, $\mathrm{i} = 1, \ldots, \mathrm{N}, \mathrm{y}_i^1 \in \{1, 2, 3, 4\}$ is the label of the first task for the *ith* example, and $\mathrm{y}_i^t \in \mathbb{R}$ is the value of the *tth* task for the *ith* example, $\mathrm{i} \in \{2, \ldots, T\}$. The parameters to be optimized are shared (θ^{sh}) and task-specific (θ^t) parameters. The parametric hypothesis per task is $f^t\left(x, \theta^{sh}, \theta^t\right) : X \to \mathrm{y}^{(\mathrm{t})}$, and the task-specific loss functions

are $\mathcal{L}^t(.,.) : y^{(t)} \times y^{(t)} \rightarrow \mathbb{R}^+$. The general optimization problem is a gradient-based multi-objective optimization of task-specific losses, as shown in Eq. (12).

$$\min_{\substack{\theta^{sh},\\ \theta^1,\ldots,\theta^T}} L\left(\theta^{sh},\theta^1,\ldots,\theta^T\right) = \min_{\substack{\theta^{sh},\\ \theta^1,\ldots,\theta^T}} \left(\widehat{\mathcal{L}}^1\left(\theta^{sh},\theta^1\right),\ldots,\widehat{\mathcal{L}}^T\left(\theta^{sh},\theta^T\right)\right) \tag{12}$$

$$\widehat{\mathcal{L}}^t\left(\theta^{sh},\theta^t\right) = -\left[\frac{1}{N}\sum_{i=1}^{N}\frac{1}{K}\sum_{k=1}^{K} W_C\, I(y_k = K)\log\left(\frac{\exp\left(\theta_k^T x_i + b\right)}{\sum_{j=1}^{K}\exp\left(\theta_j^T x_i + b\right)}\right)\right] + \frac{1}{T}\sum_{t=2}^{T}\frac{1}{N}\sum_{i=1}^{N}\left(y_i^{(t)} - \hat{y}_i^{(t)}\right)^2 + \frac{\lambda}{m}\sum_{j=1}^{m}\theta_j^2 \tag{13}$$

$\widehat{\mathcal{L}}^t(.,.)$ is a task-specific loss function defined as $\widehat{\mathcal{L}}^t\left(\theta^{sh},\ \theta^t\right) = \frac{1}{N}\sum_i \mathcal{L}\left(f^t\left(x_i;\theta^{sh},\theta^t\right), y_i^t\right)$.

Our model predicts two types of tasks: classification task and regression tasks. We equally treat the classification and regression tasks with the objective function defined as shown in Eq. (13), in which m is the number of θ^{sh} and θ^t parameters: where the first term is the weighted cross-entropy loss of multiclass classification, the second term is the mean squared loss for four regression tasks, and the last term is the regularization term. T is the number of regression tasks, and $y_i^{(t)}$ and $\hat{y}_i^{(t)}$, respectively, are the actual and predicted values of regression task t for patient i. $I(.)$ is an indicator function, where I(true statement) = 1, and 0 otherwise. W_C is a vector of class weights calculated according to the number of cases in each class. The multiclass classification is based on the *SoftMax* function, and the label y can take on K different values, $y_k \in \{1,2,\ldots,K\}$. For each input x, the model calculates the probability that $P(y_k = K|x;\theta)$ for each $k \in \{1,\ldots,K\}$. The output is a K-dimensional vector of K estimated probabilities where the sum is 1. In this study, the class label and n clinical scores are used in the backpropagation procedure to update network weights in convolution and BiLSTM layers, and to learn the most relevant features in the dense layers.

2.3. Data preprocessing

2.3.1. Missing data handling

Since the features of the dataset are numerical and categorical, the missing values were handled according to the type of data. For the baseline static data, first, we removed any feature where more than 30% of them were missing. Next, we used the k-nearest neighbors (KNN) algorithm to impute missing values, and the missing values were replaced using information from other subjects with the same diagnosis. That is, we found k neighbors, and then, the imputed value was computed by averaging the values of those neighbors. In our study, k was set to 10 empirically via experiment; for the numerical values, the Euclidean distance was also used; for categorical values, a distance of 0 was taken if both values were the same; otherwise a distance of 1 was taken. For time series data, we also removed any feature where more than 30% of them were missing. Any patient cases with missing baseline readings were excluded from the study. Some critical features, such as CSF tau (83%), were missing more than 30% of the time series; however, they were not missing at baseline. We preferred to collect these features at baseline and consider them as BG with the static data. Critical features were determined according to ADNI recommendations and AD diagnosis and progression literature [14]. For handling non-existing time series values, we followed two sequential strategies according to the intuition of ADNI. First, we filled non-applicable values for every category of data, according to ADNI procedures. For example, an ADNI 1 patient who is CN would not do an MRI scan at visit M18. As a result, we should not consider these types of values to be missing, or they are missing not at random. Many lab tests, cognitive tests, and neuroimaging scans are not done for specific diagnoses at specific visits. We followed an accurate procedure to fill these non-applicable values. If the diagnosis had not changed, we used forward filling with previous values. If the diagnosis had changed, we considered the value as missing. This technique is common in the Alzheimer's literature [48]. The second step is to determine the missing value from existing data using statistical or ML techniques. We used a medically intuitive and well-known method to handle this issue. For numerical data, we used the mean value according to the different classes: CN, sMCI, pMCI, and AD. For categorical features, we used the mode value according to the patient class. The resulting time series are regular with six months between any two consecutive visits. As a result, the LSTM and CNN models can be applied directly.

2.3.2. Data standardization

The available participants data for both the baseline and the time series have a different order of magnitude. Using this data directly to train an ML model makes it difficult to converge. To ensure that every feature in the data has the same level of importance, features were standardized using the z-score method, *i.e.,* $z_j = \left(x_j - \mu_j\right)/\sigma_j)$ where x_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 z-score method converts data so they have a 0 mean and unit standard deviation, and helps to remove outliers.

2.3.3. Feature reduction with principal component analysis

We utilized principal component analysis (PCA) to reduce the number of features from MRI and PET data. PCA was implemented with a retained variance of more than 91%. Initially, MRI had 326 features extracted by the UCSF using FreeSurfer software [49], and after applying PCA, the number of principal components was 110. The whole MRI modality had 117 features after adding the seven manually calculated features. PCA reduced PET features from 288 to 75. The total number of features from all modalities was 259, *i.e.,* MRI (117), PET (75), cognitive scores (9), neuropathology (7), and assessment (51). We applied PCA with the same settings on the 131 calculated statistical features, and it generated 30 components. The resulting baseline BG had 158 features (124 ADNI features + 30 PCA components).

3. Experimental results

3.1. Experimental setup

To evaluate the performance and effectiveness of our proposed multimodal multitask DL method, we tested and compared many schemes with different settings, including the combination of different modalities, the type of fusion (*i.e.,* early or late), integration of BG (or not), usage of stacked or concatenated CNN-BiLSTM models, and either multitask usage or a single task. Inspired by Ref. [43], for each experiment, a total of 15 DL models were trained with baseline data (BL), BL + M06, BL + M06 + M12,...,BL+... +M84. The main goal of time series data is to check the increase in system confidence and accuracy as we increase the number of time steps. We implemented and tested a set of DL models using a CNN alone, LSTM alone, concatenated CNN-BiLSTM (CCL), and stacked CNN-BiLSTM (SCL). In the CCL model, the data were concurrently learned by both CNN and BiLSTM, and their resulting dis-

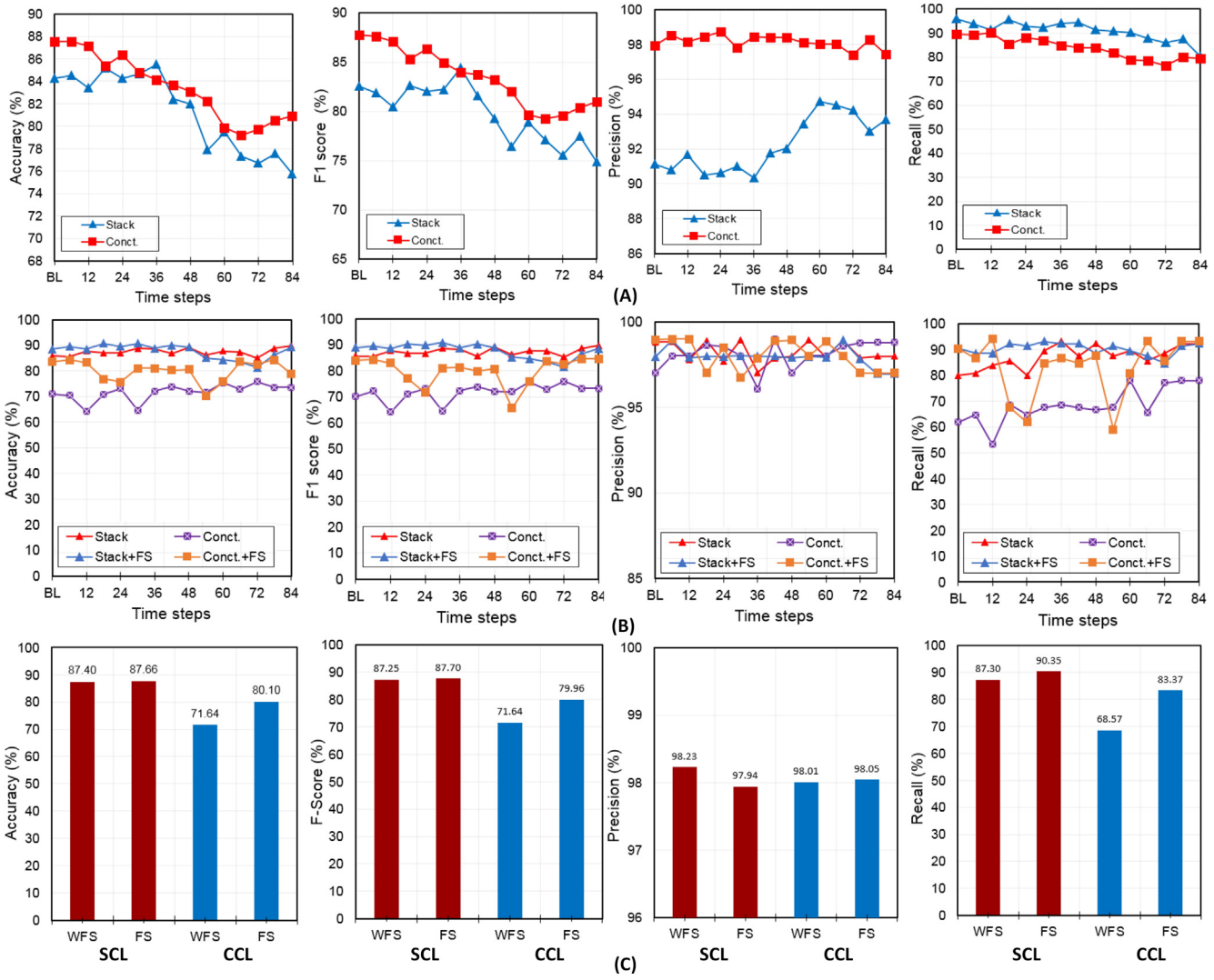

Fig. 2. Results of multimodal multiclass experiments for early fusion of five modalities. (A) Single-modality single-task results using SCL and CCL. (B) Multimodal single task tested with SCL and CCL models. (C) Multimodal single task using a feature selection step. Abbreviations: SCL, stacked CNN-BiLSTM model; CCL, concatenated CNN-BiLSTM model; WFS, deep learning model without feature selection; FS, deep learning model with feature selection; *n* time steps, number of readings considered, *e.g.,* BL + M_{06}+-M_{12}+···+M_n.

tinctive deep features were concatenated for the model's task. In the SCL model, CNN was used to learn local features from the data; then, BiLSTM was used to learn the temporal features. Results of these experiments can be found in Fig. S3 in Supplementary File 2. We find that the stacked model outperformed all other schemes, and accordingly used it for our problem. In all the models we tested, the BG data were included separately and learned by feed-forward dense layers, and then concatenated with the resulting features from the CNN and BiLSTM. Integrating the learned patterns from BG is critical for the performance of the proposed model, as shown later in Section 3.5. For the stacked CNN-BiLSTM settings, we tested the effect of the early fusion and late fusion on the performance of the DL model. Section 3.2 presents the results of DL models using single modalities to predict the AD progression detection task implemented as a multiclass classification problem. Section 3.3 collects the results of applying multimodal single-task DL models. Both early and late fusion were evaluated for the five modalities. The procedures in Section 3.4 improved the single-task models by training multimodal multitask models to optimize the one classification task and the four regression tasks. The multiclass classification task has four classes (*i.e.,* CN vs. sMCI vs. pMCI vs. AD), and the four regression tasks predict the values of the four critical cognitive scores (MMSE, CDRSB, FAQ, and ADAS) at the time of progression. The performance of the classification task was measured using accuracy, precision, recall, and F1-score metrics, and for the regression tasks, we used mean absolute error (MAE).

3.2. Model training

For all the experiments in this paper, we employed an Intel Xeon E5-2620 v3 CPU 2.40 GHz × 24, with Cuda-10.0 platform and two GPU GeForce GTX TITAN X graphics cards, 12 GB of memory; and Python 3.7.3 distributed with Anaconda 4.7.7 (64-bit). The proposed models were implemented using the Keras library based on TensorFlow as a backend. A SoftMax activation function with cross-entropy loss was used for the classification task, while a sigmoid activation function with mean square error loss was used for all the regression tasks. The Adam optimizer was used at a fixed learning rate of 0.0001 with other parameters kept at their default values [50]. The training batch size and number of epochs were 32 and 90, respectively, for all experiments. The model training was

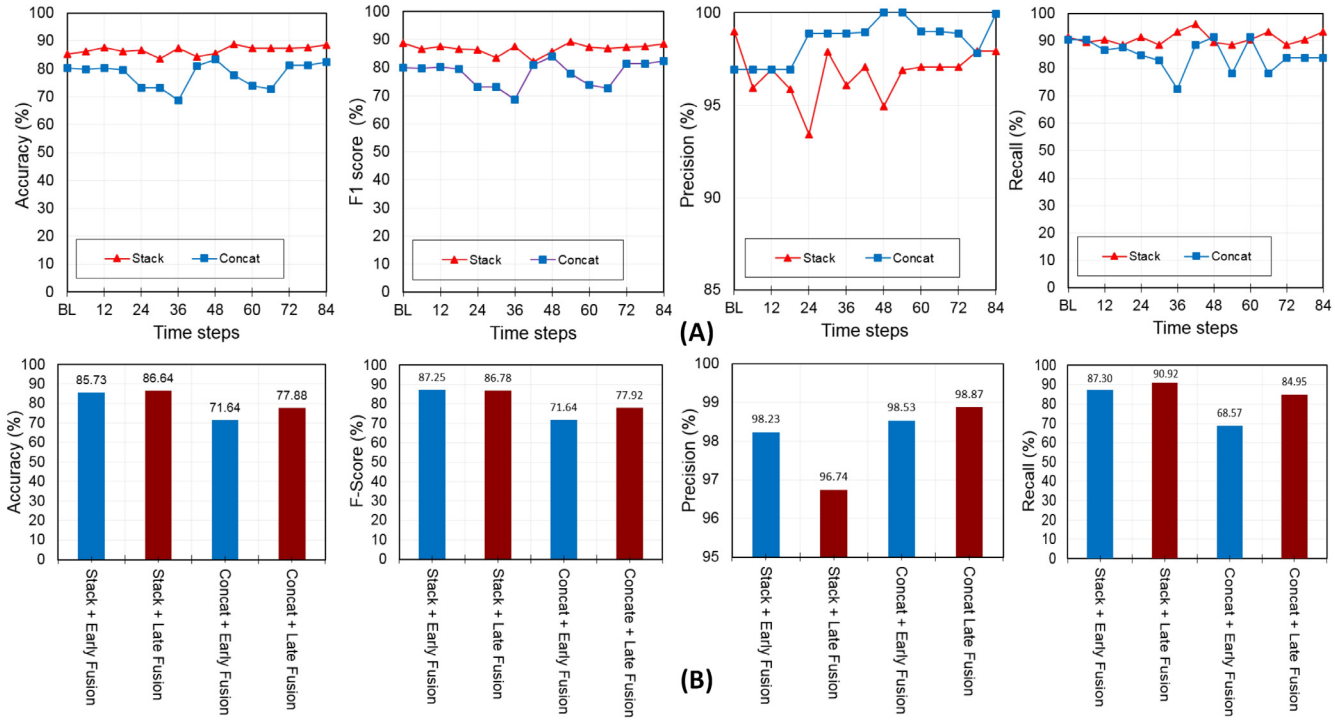

Fig. 3. Results of multimodal multiclass experiments with late fusion of five modalities. (A) Multimodal single-task tested with SCL and CCL models. (B) Comparison between SCL and CCL using early fusion and late fusion. Abbreviations: n Time steps, number of readings considered, *e.g.*, BL + M_{06}+M_{12}+···+M_n.

parallelized across the GPUs to speed up training. Fig. S2 of the Supplementary File 2 provides a detailed discussion about the proposed model's architecture. The training of the proposed model was an optimization process to find a set of model parameters that allows performing dedicated multitasks. Starting from random weights, the optimization process, guided by the minimization of the loss function, enables adjusting the model's weights in a supervised manner. All CNN and RNN hyperparameters of the final model were evaluated in preliminary experiments using training, validation, and testing before we decided on the final hyperparameters. In our experiments, we fed each modality data into a pipeline of CNN, BiLSTM, and dense blocks. Each pipeline had an equal number of CNN, BiLSTM, and fully connected (FC) layers and parameters, but their weights are independently optimized. In each pipeline, the preprocessed data were fed into one CNN block. This block has the sequence of (1) one Conv1D layer with 128 filters, a 4×1 filter size, and one stride, (2) a rectified linear unit (ReLU) activation function for non-linearity, (3) L2 regularization with parameter 0.01, (4) one max pooling layer of kernel size 2 and stride 2, for down-sampling, and (4) a dropout with probability 0.10. Note that we used the same padding on this CNN layer. The output of the CNN block was then fed into the BiLSTM block. This block had the following sequence: (1) 3 stacked BiLSTM layers with 128 units in each layer and with a Tanh activation function, (2) L2 regularization with parameter 0.01, and (3) a dropout with probability 0.10. The BiLSTM layers integrated the features of multiple time points to learn the longitudinal features. Then, the BiLSTM block is followed by an FC block with 2 FC layers, a ReLU activation function, L2 regularization with parameter 0.01, and a dropout with probability 0.10. All output of the FC blocks for all pipelines was concatenated by a flattening layer and entered into 3 consecutive FC layers to fuse the learned deep features from different modalities, and learn new temporal relationships among these modalities, as illustrated in Fig. S2 in the Supplementary File 2. The background data were added to the model later, after passing three sequential feed-forward dense layers, which reduced the dimensionality of features from 158 to 64. For this subnetwork, we used L2 regularization with parameter 0.01 and a dropout with probability 0.20. The learned features from both time series modalities and background knowledge were again concatenated using a flattening layer, and we used one dense layer to fuse the learned features. Once this feature extraction process finished, all the deep features were fed into the classification task and the four regression tasks. Each task has four separate dense layers, a ReLU activation function, an L2 regularization with parameter 0.001, and a dropout with probability 0.20. The last layer of the classification task uses a Softmax activation function, and regression tasks use a sigmoid function. When the model is trained, the Adam optimizer was used for multi-objective loss function optimization with a learning rate of 0.001, epochs of 90, and batch size of 32. Besides, to prevent overfitting, we used L2 norm regularization with coefficient λ (0.01, 0.001), a dropout for each layer (0.10, 0.20), and an early stop when the error does not decrease within the next 30 epochs. A standard technique to test our model performance is to split our dataset of 1536 cases into stratified datasets at 60% for training, 20% for validation, and 20% for testing. A procedure known as stratification randomizes the instances at each execution, such that all the datasets contain a similar proportion of the different classes. In order to prevent bias, the procedure was repeated 10 times in our experiments. For a fair comparison, network settings for all experiments remain unchanged, including dropout and L2 regularization penalty coefficient. After data preprocessing, we assigned class weights proportional to the class frequencies to preserve data balancing. The network was trained only on the training set, while validation samples were used to determine when to stop the optimization. The results are reported for the unseen test set.

3.3. Single modality single task modeling

Training our multiclass progression detection problem is a challenging ML task, because CN and sMCI have similar features, and pMCI and AD are similar, as well [6,44]. Medically, depending on

a single modality (e.g. MRI) is not acceptable because medical experts always check many sources of data to make a final decision [22,24]. Before combining the five modalities, we tested the performance of each of the single modalities alone and report the average results. Each modality was tested with CCL and SCL. To measure the significance of BG, these models were tested with and without integration of these static data. Results were collected for 15 different combinations of time steps. The results of DL models without the integration of BG are bad (see Figs. S4 and S5 in Supplementary File 2). Fig. 2A shows the average results over the 15 time steps after integrating BG with the learned features from CCL or SCL. Detailed results for each individual modality can be found in Figs. S6 and S7 of Supplementary File 2. The CCL model had an average performance of accuracy (83.46 ± 2.97%), F1-score (83.45 ± 3.02%), precision (98.13 ± 0.38%), and recall (83.85 ± 4.47%). We can see that the performance of SCL was significantly ($P < 0.005$) lower than CCL with accuracy (81.39 ± 3.53%), F1-score (79.82 ± 2.97%), precision (92.23 ± 1.56%), and recall (90.92 ± 4.09%). SCL probably archived a lower performance because single modalities were simple, compared to the level of the deepness with SCL. As can be noticed, the performance of single modalities is not good enough from the medical and ML perspectives. In addition, both CCL and SCL models were not stable because of large standard deviations in their results. Furthermore, we observed from Fig. 2A that the performance of the model degraded as we added more time steps. This indicates that adding more information confuses the model, and is insufficient to improve the performance. As a result, *in the following experiments, we depended mainly on multimodal data fusion to benefit from the strengths of every modality and to generate more stable models with better performance*. We tested both early fusion and late fusion techniques using different collections of time steps.

3.4. Multimodal single task modeling

3.4.1. Experiment 1: Evaluate the stacked and concatenated models

In this experiment, the stacked and concatenated models (*i.e.,* SCL and CCL models) in the multimodal setting were evaluated. We applied the early fusion setting for our dataset modals, where the five modalities of our dataset are fused in a single feature vector. The main purpose of this experiment was to check whether to use the stacked or concatenated setting for our final multimodal model. The learned features from DL models were again concatenated with the BG features to predict the final patient class. As noticed in the results illustrated in Fig. 2B, the CCL model is not deep enough to learn complex patterns from the high-dimensional data. Fig. 2B also shows the results of using different collections of time steps from baseline data only, for up to 15-time steps. The CCL model achieved an average performance of: accuracy (71.64 ± 3.33%), F1-score (71.75 ± 3.33%), precision (98.08 ± 0.82%), and recall (68.57 ± 6.90%). We noticed that the system became more accurate and confident when it had more time steps. Moreover, the SCL model is deeper than CCL; thus, it achieved significantly better results ($P < 0.001$) of accuracy (87.40 ± 1.42%), F1-score (87.25 ± 1.43%), precision (98.23 ± 0.60%), and recall (87.30 ± 4.55%). Besides, the SCL model is more stable than CCL because its results have lower variance than CCL. The combination of multimodalities produced a dataset with very long feature vector (*i.e.,* 259 features), which requires deeper models to recognize complex hidden patterns. However, training very deep models requires much more time, produces slow models, and might overfit the data. To check the effect of a prior feature selection step on a model's performance, the previous experiments were repeated after applying a feature selection step to the early fused data. We apply the Lasso technique [51] with 5-fold cross-validation on the 259 features, which selects the best representative list of 110 features. These features were used to train the previous two DL models. As expected, we found that the feature selection step has less effect on the performance of the SCL model, but it consistently improves the performance of the CCL model. Fig. 2C shows the performance without feature selection (WFS) and with feature selection (FS). For example, the accuracy of SCL was only improved by 0.26%. On the other hand, the accuracy of CCL was significantly improved by 8.46% ($P < 0.003$). Results of this experiment assert the advantages of using the stacked model over the concatenated one due

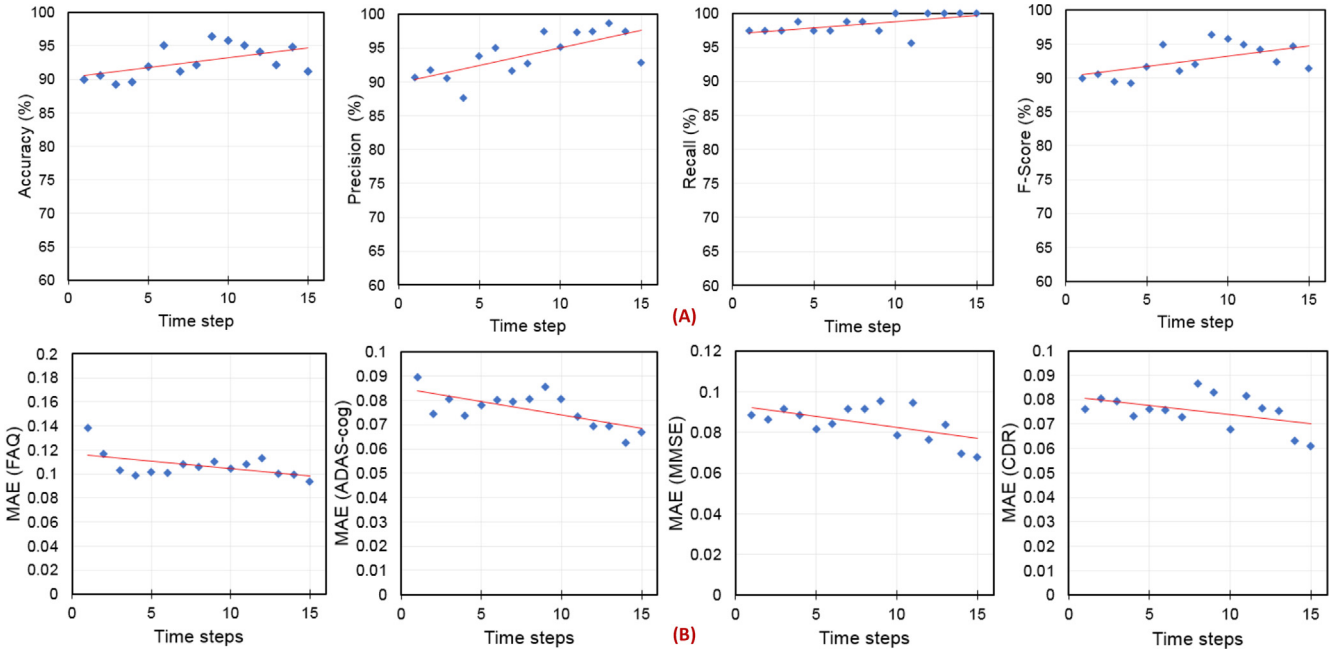

Fig. 4. Results of multimodal multitask experiments for late fusion of five modalities. (A) The multi-class classification task performance; and (B) the four regression tasks for the values of critical cognitive scores at progression time. The solid red lines represent regression lines.

to the insignificant difference between the results of WFS and FS experiments ($P < 0.04$).

3.4.2. Experiment 2: Evaluate early and late fusion

This experiment evaluated the performance of early and late fusion on the five modalities. Its main purpose was to find the optimal DL architecture (*i.e.,* early or late fusion) suitable for one task, and then generalize it to other tasks. In the late fusion design, each modality was learned separately using either stacked or concatenated CNN-BiLSTM models. The learned features from each modality were concatenated and then combined with the learned features from BG data. The resulting deep features vector was entered into dense layers to optimize the multiclass problem. Fig. 3A shows the results of SCL and CCL models for the late fusion scheme. The SCL model had an average accuracy (86.64%), F1-Score (86.78%), precision (96.74%), and recall (90.92%). The CCL model had an average accuracy (77.88%), F1-Score (77.92%), precision (98.53%), and recall (84.95%). As we can notice, the SCL model had significantly better performance ($P < 0.003$) than the CCL model. On the other hand, Fig. 3B compares Experiment 2 with Experiment 1 to select the final architecture of the DL model. The SCL model consistently achieved the most stable performance. SCL with late fusion achieved the highest accuracy (86.64%) and recall (90.92%). On the other hand, SCL with early fusion achieved the highest F1-score (87.25%). The CCL model achieved the lowest F1-score (71.64%) and recall (68.57%). The CCL model with late fusion achieved the best precision (98.87%). For all models, BG data plays a critical role in improving the results and stabilizing the model. According to the results of Experiment 2, we took SCL with late fusion settings as our final architecture and tried to improve its performance, as explained in the following section, to work with the multitask settings.

3.5. Multimodal multitask modeling

Multitask modeling is inherently a multi-objective problem where optimization could improve the performance of the overall DL model [52]. In this section, we not only optimize a multiclass classification task (*i.e.,* CN vs. sMCI vs. pMCI vs. AD), but we also concurrently optimize four other regression tasks. The final model tells medical experts the AD progression status and the expected values of four critical cognitive scores (MMSE, CDR, ADAS, and FAQ) at the time of progression. To the best of our knowledge, this is the first study that predicts future values of the four cognitive scores and the patient's progression diagnosis within 84 months from baseline based on a DL model and longitudinal data. We followed the same experimental settings as the previous experiment.

3.5.1. Experiment 3: Five multimodalities and five tasks

In this experiment, we performed the multimodal multitask experiment of the proposed DL architecture illustrated in Fig. 1A. Fig. 4A shows the performance of the multiclass classification task. On average, this task achieved accuracy (92.62 ± 2.41%), precision (94.02 ± 3.26%), F1-score (92.56 ± 2.38%), and recall (98.42 ± 1.38 %). Although adding more time steps increases the noise in the data, we observe that the performance of our proposed DL architecture improved as we added more time steps. For example, by using baseline data only, the classification task (*i.e.,* the diagnosis

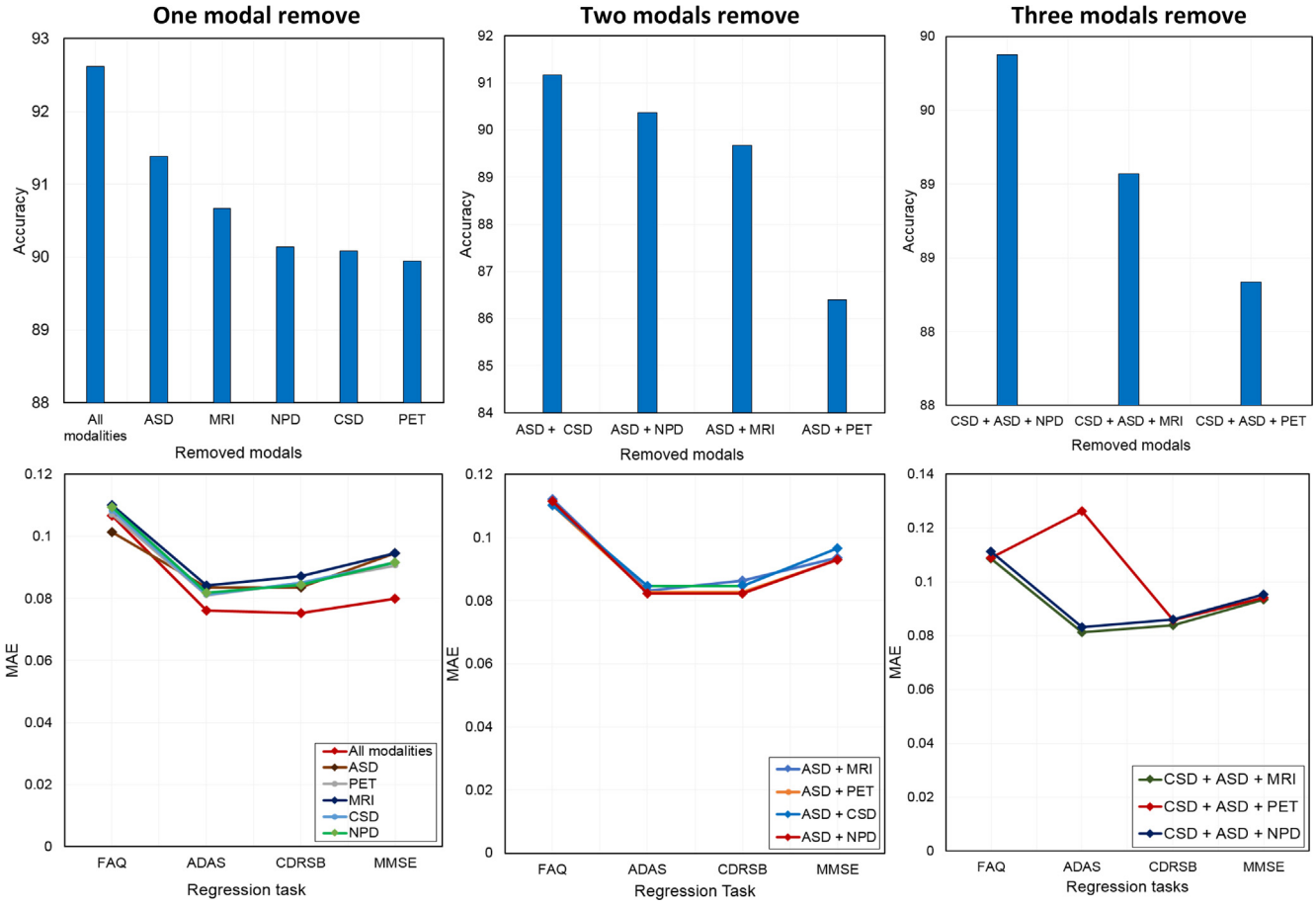

Fig. 5. Effect of removing modalities on the model performance. The first row measures the effect on the classification task, and the second row measures the effect on the four regression tasks.

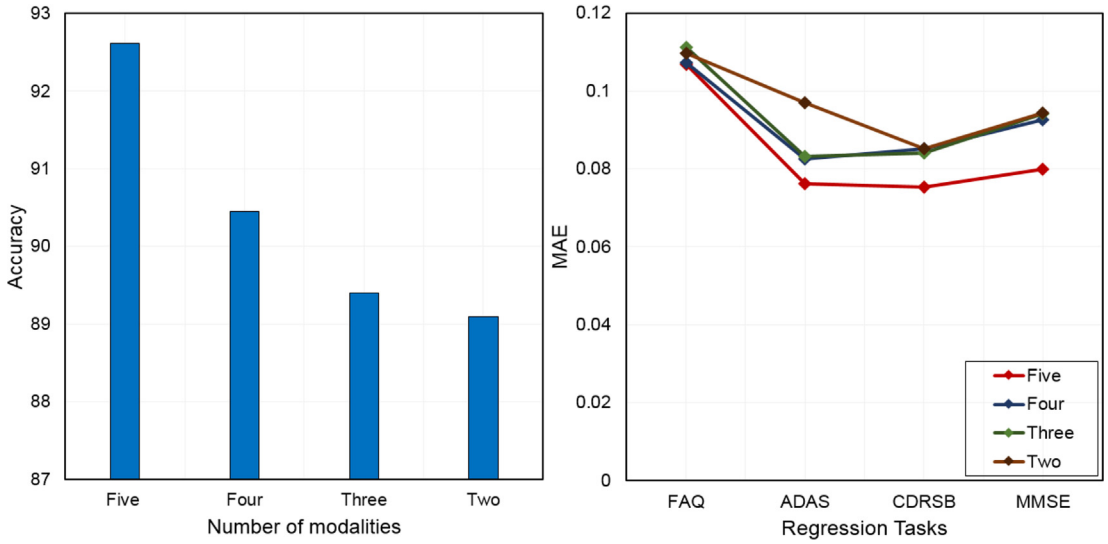

Fig. 6. Comparison between full model and other reduced models. The left part compares the different accuracies, and the right part compares the performance of the five regression tasks with different modality combinations.

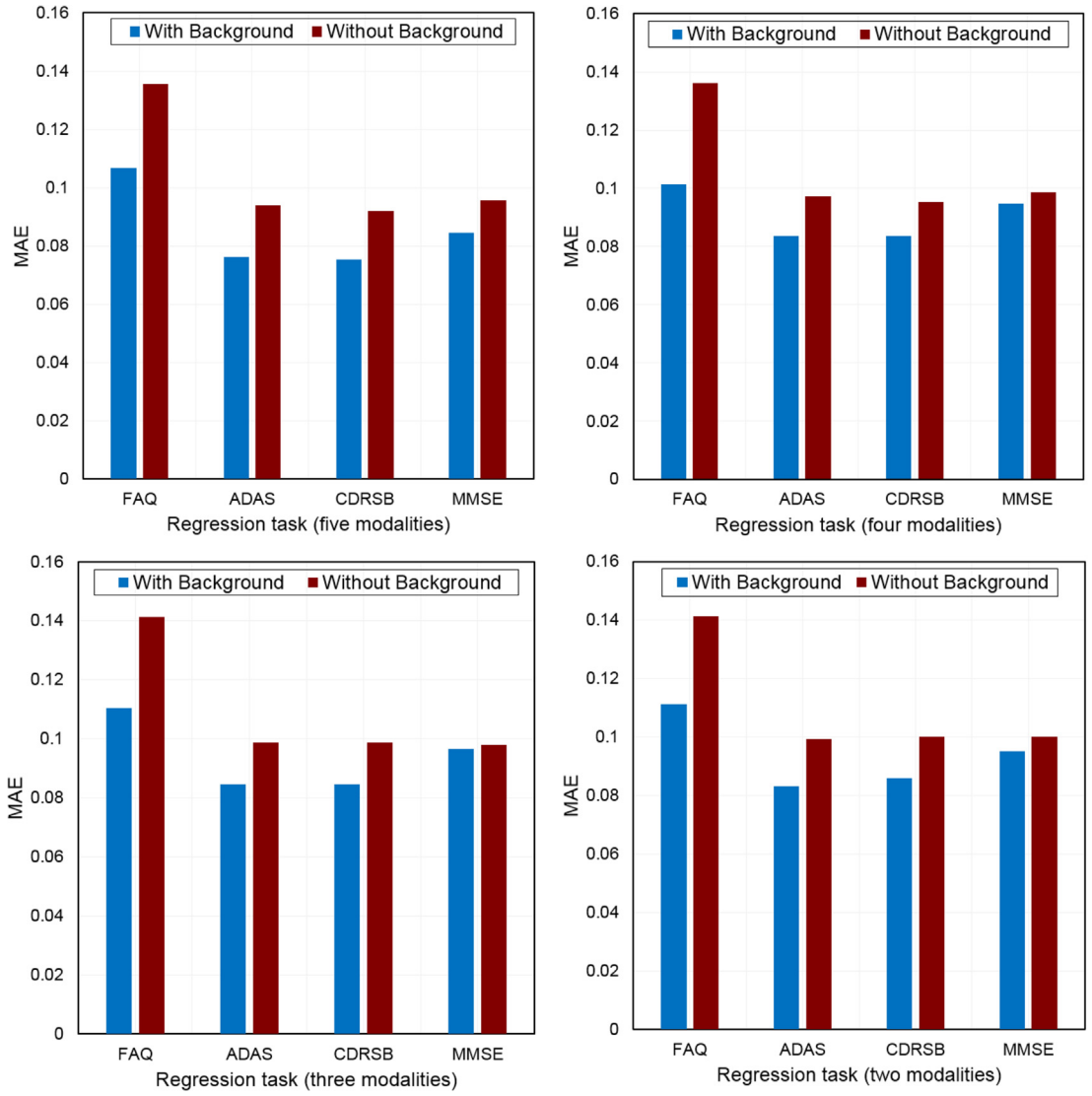

Fig. 7. Role of background knowledge for enhancing the performance of regression tasks for the full model with all modalities included.

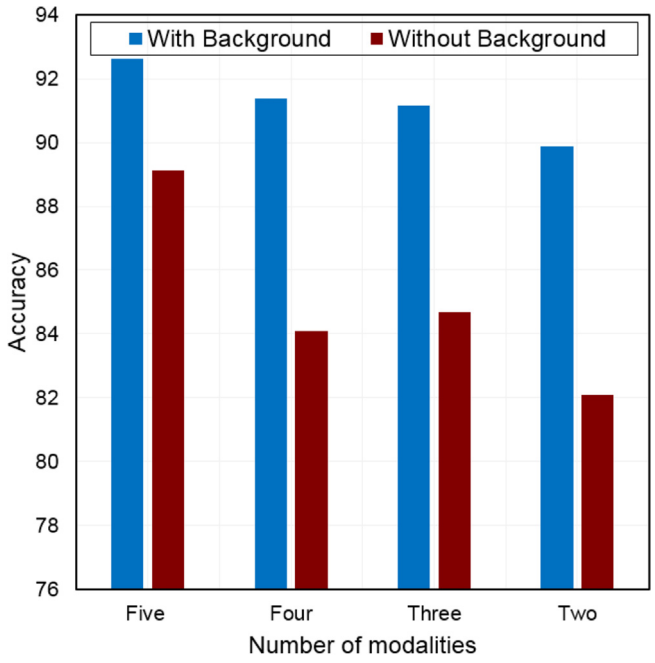

Fig. 8. Role of background knowledge for enhancing the performance of classification models.

task) had an accuracy (89.90%), but using data from all 15 time steps, the task achieved accuracy (91.21%). The same applies to the precision, F1-score, and recall where the adding data from 15 time steps improved them by 2.16%, 1.39%, and 2.56%, respectively. The results indicate that our DL model can utilize the five multimodalities and BG data to extract deep patterns suitable for alleviating noise effects, as well as to improve the five tasks performance. It is worth noting that the complex multitask model achieved better results than the single-task classification model (Experiment 2), where the average accuracy, F1-score, and recall were improved by 5.98%, 5.78%, and 7.5%, respectively. However, precision decreased by 2.72%. Theoretically, multitask learning makes the system more confident and more stable [29,47], and the results of our proposed system confirm this statement. Moreover, the current multitask model outperformed the single-task classification model in Experiment 2, achieving ≈3% accuracy improvement when including data from the 15 time steps. The multitask model with data from 15 time steps achieved accuracy (91.21%), F1-score (91.36%), and recall (99.99%). For evaluation of the four regression tasks (MMSE, ADAS, FAQ, and CDR) at any progression time step of our study, Fig. 4B shows the performance of the model based on MAE. The system achieves an average MAE of (0.107 ± 0.01), (0.076 ± 0.01), (0.075 ± 0.01), and (0.085 ± 0.01), for FAQ, ADAS, CDR, and MMSE, respectively. As expected, training the related multitasks simultaneously improves the performance of the overall system and makes the model more robust. This is due to the optimization of a single multi-objective function, rather than being highly sensitive to the behavior of every single objective task alone. Fig. 4 shows that by adding more time steps, the overall performance of the regression tasks improved. The system approximately achieves the lowest error rate when it uses the 15 time steps (*i.e.,* MAE rates are 0.094, 0.067, 0.061, and 0.068 for FAQ, ADAS, CDR, and MMSE, respectively). On average, the best performing task was the ADAS ($P < 0.007$), and the worst was the FAQ ($P < 0.0001$). To investigate the quality of the predicated regression scores, we calculated the correlation between the original and predicted scores and found that the scores are highly correlated in all 15 time steps. The average correlation coefficient for the FAQ, ADAS, CDR, and MMSE features were 0.875 ± 0.031, 0.832 ± 0.024, 0.854 ± 0.030, and 0.770 ± 0.037, respectively. The average real and predicted scores, respectively, were (9.56 ± 9.54 and 9.69 ± 8.23) for FAQ, (23.57 ± 14.09 and 24.54 ± 12.81) for ADAS, (3.53 ± 3.60 and 3.93 ± 3.30) for CDR, and (24.61 ± 5.11 and 23.95 ± 4.78) for MMSE. We notice that as we added a new time step, the model became more confident, and the correlation coefficient increased for all tasks (see Table S4 in the Supplementary File 2). The best correlation was achieved at time step 14 (*i.e.,* at M78), where it was 0.911, 0.881, 0.914, and 0.864 for FAQ, ADAS, CDR, and MMSE scores, respectively. The predicted and real values for test data at time step 14 can be seen Figs. S8 and S9 in the Supplementary File 2.

3.5.2. Experiment 4: Informative modals selection

Following the experiment outlined in Section 3.5.1, we explored the effect of adding sub-modalities instead of adding all five modalities. We tried to select the smallest feature collection that has the same or better performance than the five modalities, which helps medical experts to track AD patients more accurately with fewer data (*i.e.,* fewer medical examinations, and accordingly, lower cost). We have the five modalities' time series features, and our strategy to select the best subset of these modalities is as follows. First, we selected the best model with four modalities by repeatedly removing one modality and testing the performance. Once we get the best four-modality model with the highest performance, we repeat the previous step to get the best three-modality model. The process of selecting the best model for subsequent investigation continues until we reach a model that includes only two modalities. The balanced accuracy metric is used to track the changes in classification performance, and mean absolute error is used for regression tasks. Fig. 5 illustrates the performance of our experiment, where we removed one modality at a time and reported the performance of the resultant model. For the model with four modalities, the system had an average accuracy of 90.45 ± 0.59%. The accuracy was reduced by 2.17% from the full model, *i.e.,* with five modalities. The model with four modalities achieved the best accuracy of 91.38 ± 2.03% with no ASD and the worst accuracy of 89.94 ± 1.66% by removing the PET modality. The model followed the same behavior with the four regression tasks, see seen in the first row in Fig. 5. This is interesting, because the regression performance degradation is consistent with the classification task performance. Using all modalities, the average MAE of the regression tasks was 0.085 ± 0.015. By removing each single modality, the system performance became worse (*i.e.,* average MAE is 0.092 ± 0.011); however, the system showed no big differences in MAE among the modalities, as follows: 0.091 ± 0.009 without ASD, 0.091 ± 0.011 without PET, 0.094 ± 0.012 without MRI, 0.092 ± 0.012 without CSD, and 0.092 ± 0.013 without NPD. As a result, we decided to continue the modality selection process by removing the ASD modality, because the system achieved the best accuracy and the smallest MAE without it. The results of three modality fusion are illustrated in the second row of Fig. 5. On average, the three modalities models had a balanced accuracy of 89.40 ± 2.09%. The performance decreased by 3.21% from the full model. Moreover, removing two modalities (*i.e.,* one modality plus the ASD) generally had lower accuracy than the previous model of four modalities by 1.044%.

In other words, the system performance after removing ASD along with other modalities was 91.17 ± 2.18% without CSD, 90.37 ± 1.94% without NPD, 89.67 ± 1.66% without MRI, and 86.40 ± 3.52% without PET. The results indicate that CSD has the least effect on system accuracy. Regarding the four regression tasks, the average MAE for these four modality systems was 0.093 ± 0.013. Again, we notice that removing two modalities has similar effects among the different models. Based on the classification task results, we

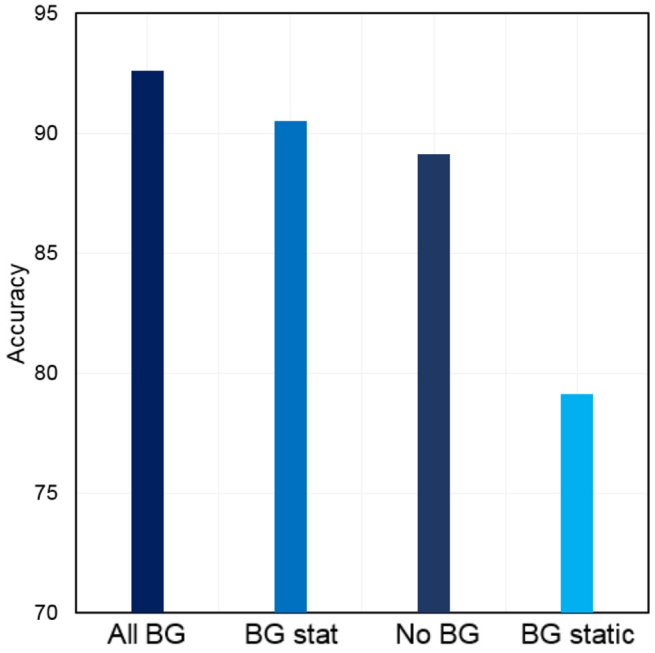

Fig. 9. The effects of different BG data on the accuracy of the full model with all modalities included.

decided to move forward by removing CSD. After removing ASD and CSD, the remaining modalities were MRI, PET, and NPD. As shown in the last row of Fig. 5, we tested the model after separately removing each of these modalities and keeping the other two. The system performance continued to decrease as we limited the number of modalities. The average accuracy of the two modality systems was 89.09 ± 0.77%. The accuracy decreased by 3.52% from the main five modality model. In the absence of ASD and CSD, the average accuracy without MRI was 89.07 ± 3.28%; without PET it was 88.34 ± 0.86%; and without NPD 89.87 ± 1.46%. We notice that there were few differences between the modality combinations, and NPD had the least impact on the accuracy of the classification task. The average MAE for the regression tasks of the model with two modalities was 0.096 ± 0.014. Although removal of different modalities had a similar average effect on the four regression tasks (0.092 ± 0.012 for MRI, 0.104 ± 0.018 for PET, and 0.094 ± 0.013 for NPD), we noticed that removing the PET modality produced the worst MAE. The final selected modalities were PET and MRI, which is medically intuitive because MRI images track the changes in the brain structure, and PET (including FDG and AV45) tracks the changes in brain cell metabolism and the levels of abnormal proteins such as amyloid-beta. Most AD studies are based on these two modalities [4,5,13,22,53]. To conclude, the results of the experiment show that each modality has an important role in improving the model performance for both patient progression task and critical score regression tasks. As shown in Fig. 6, the best accuracy was achieved by the full model, and the same model again achieved the lowest MAE. Although each modality may introduce a level of noise into the data, our DL model can handle the noise, and it benefits from the variances introduced by different modalities. Moreover, the results are medically aligned with the fact that each modality has its limitations, and the need is for multiple modalities in order to have a complete picture of the patient medical diagnosis [13,18–20,54]. The careful design of our DL model, by using stacked CNN-BiLSTM, supports the integration of huge and heterogeneous time series data. However, the model also fuses the learned features from the huge amount of baseline data (*i.e.,* BG) trained by feed-forward dense layers. In the following section, we evaluate the effect of these BG features on the whole model performance.

3.5.3. Experiment 5: The effect of background knowledge

The previous experiment tested the role of time series modalities on the performance of classification and regression tasks; however, the tested models utilized a large set of baseline data (*i.e.,* 240 features), as see in Supplementary File 2. In this experiment, we evaluated the role of these BG features on the classification and regression tasks for each model selected in Experiment 4. Fig. 7 showed the regression performance for each model of Experiment 4, with and without BG data. The full model had the lowest error rate with an average MAE of 0.086 ± 0.015 for the four regression

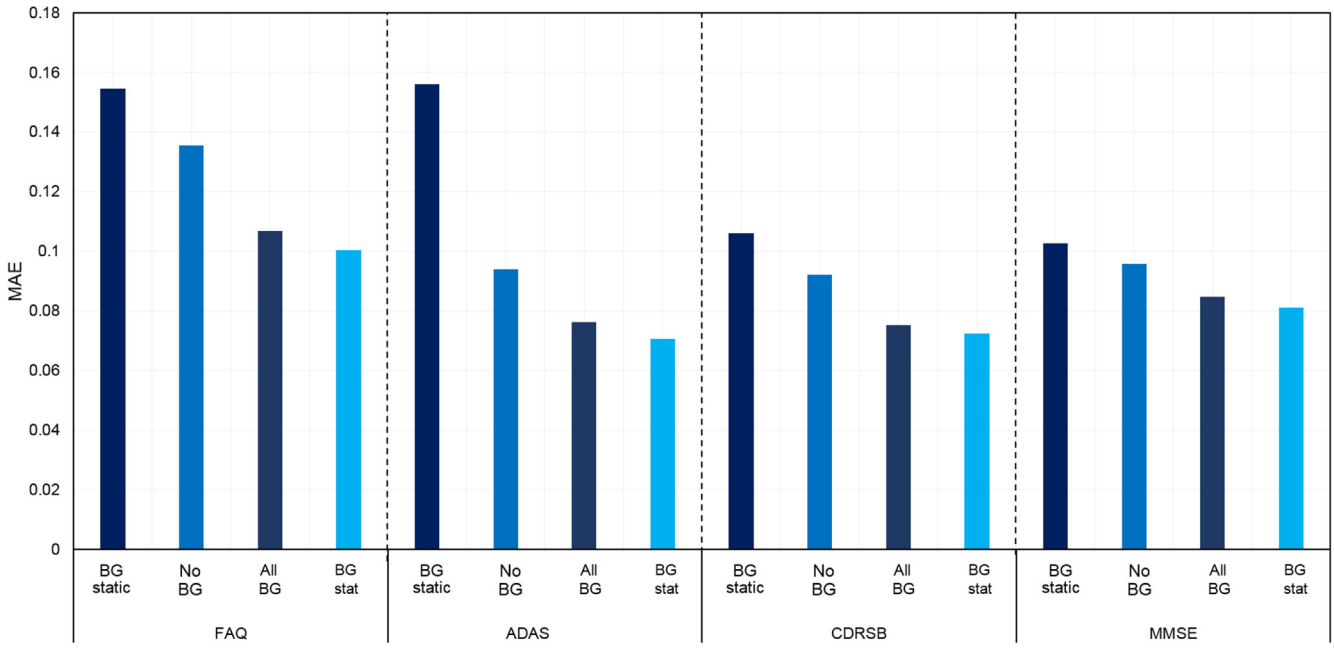

Fig. 10. The effects of different BG data on the regression tasks of the full model with all modalities included.

tasks with BG and 0.104 ± 0.021 without BG. The four modality model had an average MAE of 0.091 ± 0.009 for all tasks with BG and 0.107 ± 0.020 without BG. The three modality model achieved an average MAE of 0.094 ± 0.012 with BG and 0.109 ± 0.021 without using BG. Finally, the two modalities model had an average MAE of 0.094 ± 0.013 with BG and 0.110 ± 0.021 without BG. The MAE of regression tasks utilizing BG was significantly lower than the MAE of tasks not utilizing BG ($P < 0.005$).

Fig. 8 illustrated the accuracy of each model of Experiment 4, with and without BG data. First, the full model achieved an average accuracy of 92.62 ± 2.41%; but without BG, the model had an accuracy of 89.12 ± 6.18%. In addition to the accuracy degradation without BG, BG data have an important role in improving system confidence and stability. This fact was confirmed by the high variance of the model performance without the BG data. Second, the previously selected best model with four modalities had an accuracy of 91.39 ± 2.03% with BG and 84.08 ± 8.35% without BG. Third, the best three modality model had an accuracy of 91.17 ± 2.18% with BG and 84.67 ± 3.48% without BG; and finally, the best two modality model had an accuracy of 89.87 ± 1.46% with BG and 82.09 ± 6.42% without BG. We noticed an accuracy drop in all evaluated models, and the variances in the accuracy were higher when they did not utilize BG data. Moreover, the results of the classification task for models with BG are significantly higher than the results of models without BG ($P < 0.008$). To sum up, the results of Fig. 7 and Fig. 8 asserted the role of background data in improving the performance of each task. We also showed that the full model beats the other models in both the classification task and the regression tasks. As a result, our hypothesis that multiple modalities can help in optimizing complex objective functions of multitask DL models had been proven. As recommended by medical experts (to use as much data about patients to predict AD progression), the DL model based on all the patient's data achieved the best performance. However, in the current experiment, we tested the role of all the BG on the performance of different models. In the next experiment, we explored the effect of the different categories of BG data on model performance. Based on the performance results so far, the best model for AD progression is the five modalities model with the BG data. In this experiment, we investigated the effect of the two main categories of BG data on the model performance. These BG data have two main types: static data (118 features) and extracted statistics from time series data (130 features), as see Supplementary File 2. Fig. 9 shows the performance of the classification tasks. We tested our time series DL model by combining its learned features with all BG, with BG static data only, with BG statistical data only, and without using any BG data. The experiment results show that the model tested using all BG data is the most stable and confident model, and achieved the highest balanced accuracy of 92.62 ± 2.41%. Moreover, the inclusion of all the BG data significantly improved the accuracy of the model ($P < 0.01$). Testing the model with only statistical BG data achieved an accuracy of 90.49 ± 4.02%. On the other hand, testing the system without BG data achieved accuracy of 89.12 ± 6.18%. The performance decreased by about 3.5%; besides, the system became not stable because of the high variance of its results. Finally, by using the static data only, the system had the worst accuracy of 79.13 ± 15.40%; moreover, the system with this setting was unstable and fluctuating, which indicates that the static data were noisy.

The system consistently followed the same behavior for the four regression tasks. As can be seen from Fig. 10, combining the DL model with static BG data resulted in the highest MAE of 0.155 ± 0.016 for FAQ task, 0.156 ± 0.020 for ADAS task, 0.106 ± 0.013 for CDRSB task, and 0.103 ± 0.007 for MMSE task. The absence of BG data had negative effects on all tasks, where MAE errors were 0.1 36 ± 0.013 for FAQ task, 0.094 ± 0.009 for ADAS task, 0.092 ± 0.0 15 for CDRSB task, and 0.096 ± 0.012 for MMSE task. As shown in Fig. 9, the classification performance of the system with the whole BG data and with only statistical features had the best and comparable results. The same behavior could be seen for the regression tasks in Fig. 10. By utilizing the whole BG data, the regression tasks had MAE errors of 0.107 ± 0.011 for the FAQ task, 0.076 ± 0.007 for the ADAS task, 0.075 ± 0.007 for the CDRSB task, and 0.085 ± 0.008

Table 2
A comparison with previous studies on AD progression detection using the ADNI dataset.

Study	Subjects	Modality	Fusion	Time series	BG data	Results	Method
Lee et al. [6], 2019	1618 (ADNI)	Dem, MRI, CSD, CSF	NO	4-time steps	NO	ACC: 81%.	GRU
Zhang & Shen [53], 2012	186 (ADNI)	MRI, FDG-PET, CSF	NO	NO	NO	ACC.: 93.3% (CN/AD), 83.2% (CN/MCI). CC: 0.697 (MMSE), 0.739 (ADAS)	SVM
Ritter et al. [56], 2015	237 (ADNI)	10 modalities	NO	NO	NO	ACC: 73%	SVM
Cui et al. [43], 2019	830 (ADNI)	MRI	NO	6-time steps	NO	ACC: 91.33% (AD/NC), 71.71% (pMCI/sMCI)	Stacked CNN-BGRU
Spasov et al. [39], 2019	785 (ADNI)	MRI	NO	NO	Dem, NPD, APOe4	AUC: 0.925, ACC: 86%, SEN: 87.5%, SPE: 85% (sMCI/pMCI).	CNN.
Liu et al. [7], 2019	1984 (ADNI, AIBL, MIRIAD)	MRI	Late (CNN nets + BG)	NO	3 Dem features	ACC.: 51.8% (CN/ sMCI/ pMCI/ AD). RMSE: 1.666, 6.2, 8.537, 2.373 (CDRSB, ADAS11, ADAS13, MMSE)	CNN
Zhou et al. [25], 2013	648 (ADNI)	MRI	NO	NO	NO	CC: 0.824 for MMSE, 0.854 for ADAS	Lasso
Proposed	1536 (ADNI)	MRI, PET, CSD, ASD, NPD	Late (CNN-BiLSTM nets + BG)	15-time steps	118 static + 130 TS statistics	ACC: 92.62%, PRE: 94.02%, F1: 92.56%, REC: 98.42. MAE: 0.107, 0.076, 0.075, and 0.085, (FAQ, ADAS, CDR, MMSE)	Stacked CNN-BiLSTM

Abbreviations: ACC, Accuracy; BGRU, Bidirectional gated recurrent unit; RMSE, root mean square error; CC, correlation coefficient; Dem, demographics; F1, F1-score; AUC, area under the ROC curve, SEN, sensitivity; SPE, specificity; TS, time-series.

Table 3
A comparison with the state-of-the-art ML classification techniques for AD progression.

ML + Data	Accuracy	Precision	Recall	F1-Score
LR + BG	82.69 ± 1.12	82.97 ± 1.01	83.12 ± 0.96	83.00 ± 1.08
RF + BG	79.23 ± 0.93	80.65 ± 1.00	80.63 ± 0.89	80.64 ± 0.97
SVM + BG	**85.55 ± 1.07**	**86.16 ± 0.81**	**85.70 ± 1.30**	**85.92 ± 0.84**
DT + BG	69.28 ± 1.82	69.15 ± 1.50	69.23 ± 2.00	69.18 ± 1.71
XGBoost + BG	83.15 ± 0.85	83.39 ± 0.60	83.69 ± 0.71	83.52 ± 0.66
NB + BG	40.56 ± 1.72	40.32 ± 1.49	43.83 ± 1.80	33.70 ± 1.91
KNN + BG	74.41 ± 2.01	77.28 ± 1.88	76.24 ± 1.34	74.89 ± 1.87
MLP + BG	81.18 ± 2.21	81.88 ± 2.02	81.78 ± 1.98	81.78 ± 1.94
LR + BL	75.27 ± 3.20	75.76 ± 2.99	75.89 ± 2.81	75.81 ± 3.01
RF + BL	80.21 ± 2.77	81.23 ± 3.00	80.50 ± 3.32	80.65 ± 3.26
SVM + BL	71.97 ± 2.58	71.61 ± 2.53	72.36 ± 3.11	71.94 ± 2.61
DT + BL	72.58 ± 3.40	71.61 ± 3.08	71.77 ± 2.59	71.68 ± 2.77
XGBoost + BL	**80.22 ± 1.83**	**81.35 ± 1.61**	**80.97 ± 2.01**	**81.09 ± 1.91**
NB + BL	70.12 ± 4.01	70.86 ± 3.18	71.08 ± 4.09	69.90 ± 3.85
KNN + BL	62.91 ± 2.22	59.70 ± 1.97	60.87 ± 2.84	59.60 ± 2.77
MLP + BL	76.62 ± 1.89	66.16 ± 2.04	67.21 ± 2.45	66.58 ± 1.97
Proposed	**92.62 ± 2.41**	**94.02 ± 3.26**	**98.42 ± 1.38**	**92.56 ± 2.38**

for the MMSE task. By using the statistic BG data only, the four regression tasks achieved the best performance, where MAE errors were 0.100 ± 0.005 for the FAQ task, 0.071 ± 0.005 for the ADAS task, 0.072 ± 0.006 for the CDRSB task, and 0.081 ± 0.006 for the MMSE task. We noticed that removing static data had no significant effect on the FAQ, CDRSB, and MMSE regression tasks ($P > 0.12$), but for the ADAS task the static data made a significant difference ($P < 0.005$).

4. Discussion

We proposed a medically intuitive and integrative DL model for AD progression detection. The model is based on five time-series modalities and stacked CNN-BiLSTM design. The results proved that advanced CNN-BiLSTM design can lead to a significant improvement to AD patient monitoring. It jointly optimizes two types of tasks, *i.e.,* multiclass classification and four cognitive scores regression, by simultaneously learning and fusing discriminative features from time-series and BG data. The resulting model provides a promising performance. Our experiments suggest that no single modality may be sufficient to assess AD progression on its own. Furthermore, they pointed to the importance of fusing the learned features from these modalities, see Figs. 8 and 9. In this section, a comparative evaluation of the proposed model is conducted with the state-of-the-art ML models and ML/DL-based models proposed in the literature.

4.1. Comparison with previous studies

As shown in Table 2, we compare the performance and architecture of our model with state-of-the-art approaches that can perform tasks of both classification and cognitive score regression. Note that, due to the differences in dataset characteristics and multitask settings, it is not fair to directly compare performance among the methods. However, since performance was obtained based on the ADNI dataset, it is still appropriate to compare their results [55]. The majority of state-of-the-art studies are based on baseline MRI data only [7], as seen in Table 2. Even though they achieved good results, these types of studies are not medically acceptable. They did not mimic the real procedures of AD patient diagnosis, where experts are usually studying different types of chronic data. Compared with DM2L [7], this study is a CNN-based model applied to a large dataset of 1984 patients from ADNI, AIBL, and MIRIAD. DM2L jointly learned a four-class classification task besides four regression tasks (MMSE, ADAS 11, ASAS 13, and CDRSB). However, it depended on the baseline MRI modality only, and did not consider the complementary information that can be added by other modalities. In addition, it did not study the longitudinal changes in these features. In other words, DM2L is a single-modal multitask model. It had multiclass classification accuracy of 51.8%, and a root mean square error of 1.666, 6.2, 8.537, and 2.373 for CDRSB, ADAS 11, ADAS 13, and MMSE, respectively. Furthermore, ADAS 11 is medically considered a subset of ADAS 13, so their error rates are correlated. DM2L tried to apply the idea of BG to help the CNN with three other features (age, education, and gender). However, these three raw features were lately fused with the learned features from deep CNN subnetwork. In our model, the BG data is much larger, and they were first learned by feed-forward dense layers to extract more relevant features before late fusion with the resulting CNN-BiLSTM features. The study did some binary classification experiments, which are not reported here.

Zhang and Shen proposed M3T [9] as a multimodal multitask model for two-year AD prediction. M3T has two main steps: (1) multitask feature selection using Lasso to determine the common subset of relevant features for multiple tasks from each modality, and (2) using an SVM model for separate classification and regression. In M3T, the feature extraction process was independent of the subsequent classification and regression process. In contrast, our model automatically learns local and temporal features from each modality using stacked CNN-BiLSTM subnetworks, and then, it fuses these features to jointly learn discriminative features for the classifier and regressors. M3T was based on a small cohort of 186 patients from ADNI. The data from three modalities (MRI, FDG-PET, and CSF) were collected at baseline only, *i.e.,* no time series analysis. M3T separately learned two regression tasks (MMSE and ADAS) and one binary classification task (CN vs. MCI, CN vs. AD, or sMCI vs. pMCI). The model achieved a binary classification accuracy of 93.3% and 83.2% for CN vs. AD and CN vs. MCI, respectively, in the first experiment, and 73.9% for sMCI vs. pMCI in the second experiment. For regression, it achieved a correlation coefficient of 0.697 and 0.739 for MMSE and ADAS in the first experiment, and 0.511 and 0.531 for MMSE and ADAS, respectively, in the second experiment. One of the strengths of our proposed model is its ability to learn many time series modalities, and jointly optimize the four-class classification task and four regression tasks. Spasov et al. proposed a parameter-efficient CNN model for predicting AD progression within three-years from the baseline visit. The model combined one modality, *i.e.,* MRI (parietal, temporal, and frontal lobes) with nine BG features, namely demographic (four features), neuropsychological (four features), and APOe4 genetic. Data of 785 subjects from ADNI were collected from baseline only. The model optimized two classification tasks: MCI-to-

Table 4
A comparison with ML regression techniques.

ML + Data	FAQ	ADAS	CDR	MMSE
RF + BG	0.122 ± 0.029	**0.113 ± 0.017**	0.113 ± 0.012	**0.120 ± 0.019**
SVM + BG	0.186 ± 0.020	0.158 ± 0.048	0.136 ± 0.015	0.157 ± 0.018
BR + BG	**0.113 ± 0.020**	0.125 ± 0.024	**0.104 ± 0.015**	0.132 ± 0.016
Lasso + BG	0.172 ± 0.015	0.157 ± 0.018	0.143 ± 0.017	0.145 ± 0.018
DT + BG	0.144 ± 0.023	0.139 ± 0.019	0.151 ± 0.012	0.140 ± 0.020
GBR + BG	0.135 ± 0.017	0.129 ± 0.016	0.122 ± 0.018	0.160 ± 0.017
RF + BL	**0.131 ± 0.018**	**0.131 ± 0.020**	0.134 ± 0.067	0.137 ± 0.063
SVM + BL	0.186 ± 0.010	0.178 ± 0.018	0.166 ± 0.016	0.167 ± 0.080
BR + BL	0.140 ± 0.015	0.142 ± 0.013	0.132 ± 0.038	**0.130 ± 0.056**
Lasso + BL	0.172 ± 0.015	0.167 ± 0.018	**0.123 ± 0.017**	0.155 ± 0.083
DT + BL	0.144 ± 0.022	0.135 ± 0.027	0.139 ± 0.082	0.145 ± 0.090
GBR + BL	0.143 ± 0.017	0.143 ± 0.015	0.147 ± 0.043	0.137 ± 0.055
Proposed	**0.107 ± 0.01**	**0.076 ± 0.01**	**0.075 ± 0.01**	**0.085 ± 0.01**

pMCI conversion and CN/AD classification. There were no cognitive scores for regression tasks. This DL model extracted features from MRI using a deep CNN, and then, it fused these features with BG raw features. Fusing CNN extracted features with raw data is expected to decrease the model accuracy. The authors asserted that DL models achieved the best performance based on all data. On average, the model achieved an area under the curve of 0.925, accuracy of 86%, sensitivity of 87.5%, and specificity of 85% for sMCI vs. pMCI. For CN vs. AD, it provided high performance (AUC of 1, and 100% accuracy, sensitivity, and specificity). The model depends only on a single modality and a subset of BG features. Medically, these models could be not applicable. Besides, it optimized two very related, if not identical, binary classification tasks. Although our model optimizes a more complex cost function based on multimodalities, it achieved better performance in predicting MCI progression. Ritter et al. [56] extracted 288 features from 10 different modalities to predict MCI conversion as a single binary classification task. Data from 237 subjects were collected from baseline visits only. The study compared the manual and automatic feature select on the performance of the SVM algorithm. By using the subset of features suggested by a domain expert, the SVM achieved a better accuracy of 73.44%. Although this study confirmed the role of multimodality, it achieved low accuracy and neglected many of our comparison metrics.

Cui et al. [43] proposed a single-modal single-task DL model for AD progression detection. The model was based on a binary classification task, was mainly based on the analysis of MRI features by using a stacked CNN-RNN pipeline. MRI data were from six-time steps. Given an MRI image, the CNN first learned the spatial features, and then, the RNN was built on the output of the CNN from many time steps in order to learn the longitudinal features for AD detection. The model jointly learns the spatial and longitudinal features. As done in our study, this study benefited from a stacked CNN-RNN to extract local and temporal features. It achieved an accuracy of 91.33% for AD vs. NC, and 71.71% for pMCI vs. sMCI. However, depending only on MRI is insufficient in the medical domain. In addition, the study neglected the role of BG and multitasked learning. Zhou et al. [25] proposed a multitask regression model to predict AD progression within the subsequent four years. They used baseline values of MRI modality, age, APOe4, MMSE, ADAS, and education with the Lasso technique to predict M06-M48 values of MMSE and ADAS as two separate regression tasks. The model achieved average correlation coefficients of 0.824 for MMSE and 0.854 for ADAS. Lee et al. [6] proposed a multimodal single-task RNN-based model for MCI-to-AD detection. The model is based on four modalities: demographics (four features), MRI (three features), cognitive scores (two features), and CSF (five features). The model uses transfer learning to train a single task classifier for differentiating CN vs. AD, and then the resulting classifier is retrained to work with sMCI vs. pMCI patients. The authors asserted the role of time series data to improve RNN predictive power. By using time series multimodal data, the model achieved 81% accuracy. By using multimodal baseline data only, it achieved 76% accuracy, and by using single modalities, it achieved 74% accuracy. However, the model optimized only single binary classification tasks and was based on a very small number of features. A separate RNN learns each modality, and there is no fusion of learned features. The model did not check the effect of adding BG data, and its achieved accuracy is not compared to ours. As confirmed by the Table 2 comparisons, our study is the most comprehensive and medically intuitive approach to AD progression detection.

4.2. Comparison with regular machine learning techniques

In this experiment, we compared the performance of our proposed model with eight state-of-the-art classification algorithms, *i.e.,* SVM, logistic regression (LR), random forest (RF), k-nearest neighbor (KNN), decision tree (DT), extreme gradient boosting (XGBoost), naive Bayes (NB), and multilayer perceptron (MLP), as benchmark methods. Note that these techniques were used for the multiclass classification task. We used our aggregated BG data to train and test these models. The best parameters for these techniques have been selected using grid search. The used data were preprocessed as described in Section 2.3. The stratified 10-fold cross validation was used to train and validate these models. Each experiment was repeated 10 times, and for each time, results were the average over ten rotations of the test folds. Table 3 shows the comparison among the competing techniques measured by the averages of accuracy, precision, recall, and F1-score.

In comparison with benchmark approaches, our proposed model achieved much better performance than all conventional ML techniques. The NB model achieved the worst results (*i.e.,* accuracy = 40.56 ± 1.72, precision = 40.32 ± 1.49, recall = 43.83 ± 1 .80, F1-score = 33.70 ± 1.91). On the other hand, SVM achieved the highest performance compare to other regular ML techniques (*i.e.,* accuracy = 85.55 ± 1.07, precision = 8616 ± 0.81, recall = 8570 ± 1. 30, and F1-score = 8592 ± 0.84). The proposed model had an average of 7.07%, 7.86%, 12.72%, and 6.64% performance gain in comparison with SVM for accuracy, precision, recall, and F1-score, respectively. As a result, our model predicts patient progression status more accurately than these benchmark approaches. We noticed that competing models had less standard deviations compared to our model. This is probably because the 15-time step data added more nose, which affected the model confidence. It is noteworthy that our model concurrently learned multiple tasks based

on BG plus time series data. However, all tested conventional ML techniques are single-tasked and achieved their results based on BG data only. This means that our prepared BG data had enough informative features to differentiate between the four classes. To check this hypothesis, we fed the same ML models with the baseline visit data only and reported the results. As shown in Table 3, the highest results achieved by the XGBoost technique (*i.e.,* accuracy = 80.22 ± 1.83, precision = 81.35 ± 1.61, recall = 80.97 ± 2 .01, and F1-score = 81.09 ± 1.91). Compared with the ML models fed with BG data, we noticed that the models built with baseline data have performance degradation, and higher standard deviation. These results highlighted the role of BG data to enhance models' performance. Regarding regression tasks, we compared our model with RF, SVM, Bayesian ridge (BR), lasso, DT, and gradient boosting regression (GBR). These techniques were evaluated separately using BG data and baseline visit data. For each regression task, we implemented a separate regression model, where 24 models were implemented for each dataset. Table 4 shows the resulting MAE. Based on the BG data, BR had the best results for FAQ regression task (MAE of 0.113 ± 0.020) and CDR task (MAE of 0.104 ± 0. 015), and RF had the best result for ADAS (MAE of 0.113 ± 0.017) and MMSE (MAE of 0.120 ± 0.019). Our model had lower MAE for all regression tasks by 0.006 for FAQ, 0.037 for ADAS, 0.029 for CDR, and 0.035 for MMSE. On average, the MAE for RF, SVM, BR, lasso, DT, and GBR were 0.117 ± 0.019, 0.159 ± 0.025, 0.119 ± 0.0 19, 0.154 ± 0.017, 0.144 ± 0.019, and 0.137 ± 0.017, respectively. On average, RF achieved the best results, and our model lower MAE by 0.015, 0.037, 0.038, and 0.035 for FAQ, ADAS, CDR, and MMSE, respectively. The average MAE for FAQ, ADAS, CDR, and MMSE were 0.145 ± 0.021, 0.137 ± 0.024, 0.128 ± 0.015, and 0.14 2 ± 0.018, respectively, where the proposed model decreased the error rate by 0.038, 0.061, 0.053, and 0.057, respectively.

All regression models trained using the baseline data achieved worse results than BG-based models. On average, the FAQ, ADAS, CDR, and MMSE tasks had MAE scores of 0.153 ± 0.016, 0.149 ± 0. 019, 0.140 ± 0.044, and 0.145 ± 0.071, respectively. Again, RF achieved the best results for FAQ (MAE of 0.131 ± 0.018) and ADAS (MAE of 0.131 ± 0.020) tasks. BR achieved the best result for MMSE (MAE of 0.130 ± 0.056), and lasso had the lowest score for CDR task (MAE of 0.123 ± 0.017). The RF, SVM, BR, lasso, DT, and GBR regression models had average MAE rates of 0.133 ± 0.042, 0.174 ± 0.031, 0.136 ± 0.031, 0.154 ± 0.033, 0.141 ± 0.055, and 0.143 ± 0.033, respectively. The performance of regression tasks was consistent with that of classification task. Both classification and regression results of the ML models confirmed the positive role of BG data on the performance, and the superiority of the proposed model.

4.3. Study limitations and future directions

Although our proposed model added an advanced achievement in AD progression detection, we still have several limitations that need further consideration. First, in this study, we concentrated on the optimization of the model's performance. However, medical experts consider the ability to explain model decisions as important as their accuracies. Explainable ML model is trustworthy, accountable, and more acceptable in the real medical domain. DL models are considered as black boxes. However, recently, studies tried to open the box and explain why the model took specific decisions. Second, despite we used the most comprehensive number of modalities, we have not studied the relationship between AD progression and the patient's comorbidities (cardiovascular disease, depression, renal genitourinary, endocrine metabolism, *etc.*), adverse events (*e.g.,* headache, fever, dizziness, insomnia, *etc.*), and previously taken drugs (*e.g.,* tacrine, memantine, *etc.*). The ADNI database collected some data about these modalities, but still, they are not enough to train DL models. Third, extracting only the medically informative features from neuroimaging is of fundamental importance to improve model speed or to concentrate on specific brain regions. For example, for MRI, it is important to study the role of longitudinal change in Volumatic, surface area, etc. of a different region of interest. For PET, there are many types of PET imaging including FDG PET, AV45 PET, and AV1451 PET. Studying the role of each category alone and in combination with other modalities is important. Fourth, we handled missing values based on a medically intuitive procedure. The last step of this procedure is to replace missing values by feature means according to the class label. This method is popular in literature, but it is interesting to check other methods such as LSTM masking. Fifth, we have not added many critical modalities in our study because of the lack of data. Specific modalities such as CSF, lab tests, and APOe4 genotyping are medically critical, but ADNI has collected these data at baseline visits only. Please note that we included all these features in our BG data. Finally, we gave equal weights for the classification and regression tasks in our current work [7], but these tasks might contribute differently. In a future extension of our model, we will study how to learn custom weights for each task automatically.

5. Conclusion

In this paper, we have proposed an ensemble multimodal multitask deep learning model based on the combination of CNN and BiLSTM for jointly learning of AD multiclass classification and four cognitive scores regression. The CNN subnetworks were proposed to extract the local features from individual time series in each modality, and the BiLSTM subnetworks were used to model the time series temporal variations and extract the longitudinal features. The model is based on the fusion of five heterogeneous time series modalities. The proposed DL model is based on the late fusion of five pipelined stacked CNN-BiLSTM subnetworks. The proposed model exploited set of BG static features collected from baseline visits and time series statistics, which enhanced the model performance. Experimental results on the ADNI dataset demonstrated the effectiveness of the proposed model. To the best of our knowledge, this is the first DL model in AD domain that investigates the joint prediction of multiple regression and multiclass classification variables from multiple longitudinal and BG data.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Shaker El-Sappagh: Conceptualization, Methodology, Software, Formal analysis, Writing - review & editing. **Tamer Abuhmed:** Conceptualization, Methodology, Software, Formal analysis, Writing - review & editing. **S.M. Riazul Islam:** Investigation, Data curation, Writing - original draft, Visualization. **Kyung Sup Kwak:** Supervision, Writing - review & editing.

Acknowledgment

This work was supported by National Research Foundation of Korea-Grant funded by the Korean Government (Ministry of Science and ICT)-NRF-2020R1A2B5B02002478 and NRF-2016R1D1A1A03934816. In addition, Dr. Jose M. Alonso is Ramon y Cajal Researcher (RYC-2016-19802), and its research is supported by the Spanish Ministry of Science, Innovation and Universities

(grants RTI2018-099646-B-I00, TIN2017-84796-C2-1-R, TIN2017-90773-REDT, and RED2018-102641-T) and the Galician Ministry of Education, University and Professional Training (grants ED431F 2018/02, ED431C 2018/29, ED431G/08, and ED431G2019/04), with all grants co-funded by the European Regional Development Fund (ERDF/FEDER program). Data used in preparation of this article were obtained from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database (adni.loni.usc.edu). As such, the investigators within the ADNI contributed to the design and implementation of ADNI and/or provided data but did not participate in analysis or writing of this report.

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at https://doi.org/10.1016/j.neucom.2020.05.087.

References

[1] A. Alberdi, A. Aztiria, A. Basarab, On the early diagnosis of alzheimer's disease from multimodal signals: a survey, Artificial Intelligence in Medicine 71 (2016) 1–29.
[2] C.L. Masters, K. Beyreuther, Alzheimer's centennial legacy: prospects for rational therapeutic intervention targeting the aβ amyloid pathway, Brain 129 (11) (2006) 2823–2839.
[3] R.A. Sperling, P.S. Aisen, L.A. Beckett, D.A. Bennett, S. Craft, A.M. Fagan, T. Iwatsubo, C.R. Jack Jr, J. Kaye, T.J. Montine, et al., Toward defining the preclinical stages of alzheimer's disease: recommendations from the national institute on aging-alzheimer's association workgroups on diagnostic guidelines for alzheimer's disease, Alzheimer's & Dementia 7 (3) (2011) 280–292.
[4] H. Li, M. Habes, D.A. Wolk, Y. Fan, A deep learning model for early prediction of alzheimer's disease dementia based on hippocampal mri, 2019, ArXiv abs/1904.07282.
[5] S. Qiu, G.H. Chang, M. Panagia, D.M. Gopal, R. Au, V.B. Kolachalama, Fusion of deep learning models of mri scans, mini–mental state examination, and logical memory test enhances diagnosis of mild cognitive impairment, Alzheimer's & Dementia: Diagnosis, Assessment & Disease Monitoring 10 (2018) 737–749.
[6] G. Lee, K. Nho, B. Kang, K.-A. Sohn, D. Kim, Predicting alzheimer's disease progression using multi-modal deep learning approach, Scientific Reports 9 (1) (2019) 1952.
[7] M. Liu, J. Zhang, E. Adeli, D. Shen, Joint classification and regression via deep multi-task multi-channel learning for alzheimer's disease diagnosis, IEEE Transactions on Biomedical Engineering 66 (5) (2018) 1195–1206.
[8] X. Ding, M. Bucholc, H. Wang, D.H. Glass, H. Wang, D.H. Clarke, A.J. Bjourson, C. D. Le Roy, M. O'Kane, G. Prasad, et al., A hybrid computational approach for efficient alzheimer's disease classification based on heterogeneous data, Scientific Reports 8 (1) (2018) 9774.
[9] D. Zhang, D. Shen, A.D.N. Initiative, et al., Multi-modal multi-task learning for joint prediction of multiple regression and classification variables in alzheimer's disease, NeuroImage 59 (2) (2012) 895–907.
[10] B. Cheng, M. Liu, D. Zhang, B.C. Munsell, D. Shen, Domain transfer learning for mci conversion prediction, IEEE Transactions on Biomedical Engineering 62 (7) (2015) 1805–1817.
[11] C.-Y. Wee, P.-T. Yap, D. Shen, A.D.N. Initiative, Prediction of alzheimer's disease and mild cognitive impairment using cortical morphological patterns, Human Brain Mapping 34 (12) (2013) 3411–3425.
[12] P. Moore, T. Lyons, J. Gallacher, A.D.N. Initiative, et al., Random forest prediction of alzheimer's disease using pairwise selection from time series data, PloS One 14 (2) (2019), e0211558.
[13] E. Moradi, A. Pepe, C. Gaser, H. Huttunen, J. Tohka, A.D.N. Initiative, et al., Machine learning framework for early MRI-based alzheimer's conversion prediction in mci subjects, Neuroimage 104 (2015) 398–412.
[14] M.W. Weiner, D.P. Veitch, P.S. Aisen, L.A. Beckett, N.J. Cairns, R.C. Green, D. Harvey, C.R. Jack Jr, W. Jagust, J.C. Morris, et al., Recent publications from the alzheimer's disease neuroimaging initiative: Reviewing progress toward improved ad clinical trials, Alzheimer's & Dementia 13 (4) (2017) e1–e85.
[15] K. Ito, B. Corrigan, Q. Zhao, J. French, R. Miller, H. Soares, E. Katz, T. Nicholas, B. Billing, R. Anziano, et al., Disease progression model for cognitive deterioration from alzheimer's disease neuroimaging initiative database, Alzheimer's & Dementia 7 (2) (2011) 151–160.
[16] T. Tong, Q. Gao, R. Guerrero, C. Ledig, L. Chen, D. Rueckert, A.D.N. Initiative, et al., A novel grading biomarker for the prediction of conversion from mild cognitive impairment to alzheimer's disease, IEEE Transactions on Biomedical Engineering 64 (1) (2016) 155–165.
[17] L. Nie, L. Zhang, L. Meng, X. Song, X. Chang, X. Li, Modeling disease progression via multisource multitask learners: a case study with alzheimer's disease, IEEE Transactions on Neural Networks and Learning Systems 28 (7) (2016) 1508–1519.
[18] P.S. Pillai, T.-Y. Leong, Fusing heterogeneous data for alzheimer's disease classification, Stud. Health Technol. Inform.
[19] M. Ewers, C. Walsh, J.Q. Trojanowski, L.M. Shaw, R.C. Petersen, C.R. Jack Jr, H.H. Feldman, A.L. Bokde, G.E. Alexander, P. Scheltens, et al., Prediction of conversion from mild cognitive impairment to alzheimer's disease dementia based upon biomarkers and neuropsychological test performance, Neurobiology of Aging 33 (7) (2012) 1203–1214.
[20] K. Li, R. O'Brien, M. Lutz, S. Luo, A.D.N. Initiative, et al., A prognostic model of alzheimer's disease relying on multiple longitudinal measures and time-to-event data, Alzheimer's & Dementia 14 (5) (2018) 644–651.
[21] F. Liu, L. Zhou, C. Shen, J. Yin, Multiple kernel learning in the primal for multimodal alzheimer's disease classification, IEEE Journal of Biomedical and Health Informatics 18 (3) (2013) 984–990.
[22] S. Duchesne, A. Caroli, C. Geroldi, D.L. Collins, G.B. Frisoni, Relating one-year cognitive change in mild cognitive impairment to baseline MRI features, Neuroimage 47 (4) (2009) 1363–1370.
[23] D. Ramachandram, G.W. Taylor, Deep multimodal learning: a survey on recent advances and trends, IEEE Signal Processing Magazine 34 (6) (2017) 96–108.
[24] Y. Wang, Y. Fan, P. Bhatt, C. Davatzikos, High-dimensional pattern regression using machine learning: from medical images to continuous clinical variables, Neuroimage 50 (4) (2010) 1519–1535.
[25] J. Zhou, J. Liu, V.A. Narayan, J. Ye, A.D.N. Initiative, et al., Modeling disease progression via multi-task learning, NeuroImage 78 (2013) 233–248.
[26] Q. Liao, Y. Ding, Z.L. Jiang, X. Wang, C. Zhang, Q. Zhang, Multi-task deep convolutional neural network for cancer diagnosis, Neurocomputing 348 (2019) 66–73.
[27] W. Liu, B. Zhang, Z. Zhang, X.-H. Zhou, Joint modeling of transitional patterns of alzheimer's disease, PloS One 8 (9) (2013), e75487.
[28] L. Huang, Y. Jin, Y. Gao, K.-H. Thung, D. Shen, A.D.N. Initiative, et al., Longitudinal clinical score prediction in alzheimer's disease with soft-split sparse regression based random forest, Neurobiology of Aging 46 (2016) 180–191.
[29] H. Harutyunyan, H. Khachatrian, D. C. Kale, G. Ver Steeg, A. Galstyan, Multitask learning and benchmarking with clinical time series data, Scientific data 6 (1) (2019) 96.
[30] H. Suresh, N. Hunt, A. Johnson, L. A. Celi, P. Szolovits, M. Ghassemi, Clinical intervention prediction and understanding using deep networks, arXiv preprint arXiv:1705.08498.
[31] C. Tian, J. Ma, C. Zhang, P. Zhan, A deep neural network model for short-term load forecast based on long short-term memory network and convolutional neural network, Energies 11 (12) (2018) 3493.
[32] D. Lu, K. Popuri, G.W. Ding, R. Balachandar, M.F. Beg, Multimodal and multiscale deep neural networks for the early diagnosis of alzheimer's disease using structural mr and fdg-pet images, Scientific Reports 8 (1) (2018) 5697.
[33] M. Liu, D. Cheng, K. Wang, Y. Wang, A.D.N. Initiative, et al., Multi-modality cascaded convolutional neural networks for alzheimer's disease diagnosis, Neuroinformatics 16 (3–4) (2018) 295–308.
[34] E. Goceri, Formulas behind deep learning success, in: Int. Conf. Appl. Anal. Math. Model, 2018.
[35] E. Goceri, Diagnosis of alzheimer's disease with Sobolev gradient-based optimization and 3d convolutional neural network, International journal for numerical methods in biomedical engineering 35 (7) (2019), e3225.
[36] E. Goceri, Challenges and recent solutions for image segmentation in the era of deep learning, 2019, pp. 1–6.
[37] E. Goceri, Capsnet topology to classify tumours from brain images and comparative evaluation, IET Image Processing 14 (2020) 882–889(7).
[38] H. Choi, K.H. Jin, A.D.N. Initiative, et al., Predicting cognitive decline with deep learning of brain metabolism and amyloid imaging, Behavioural Brain Research 344 (2018) 103–109.
[39] S. Spasov, L. Passamonti, A. Duggento, P. Liò, N. Toschi, A.D.N. Initiative, et al., A parameter-efficient deep learning approach to predict conversion from mild cognitive impairment to alzheimer's disease, Neuroimage 189 (2019) 276–287.
[40] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Computation 9 (8) (1997) 1735–1780.
[41] A. Yousif, Z. Niu, J. Chambua, Z.Y. Khan, Multi-task learning model based on recurrent convolutional neural networks for citation sentiment and purpose classification, Neurocomputing 335 (2019) 195–205.
[42] Z. Hu, Z. Zhang, H. Yang, Q. Chen, R. Zhu, D. Zuo, Predicting the quality of online health expert question-answering services with temporal features in a deep learning framework, Neurocomputing 275 (2018) 2769–2782.
[43] R. Cui, M. Liu, A.D.N. Initiative, et al., Rnn-based longitudinal analysis for diagnosis of alzheimer's disease, Computerized Medical Imaging and Graphics 73 (2019) 1–10.
[44] N. Amoroso, D. Diacono, A. Fanizzi, M. La Rocca, A. Monaco, A. Lombardi, C. Guaragnella, R. Bellotti, S. Tangaro, A.D.N. Initiative, et al., Deep learning reveals alzheimer's disease onset in mci subjects: results from an international challenge, Journal of Neuroscience Methods 302 (2018) 3–9.
[45] N.P. Oxtoby, D.C. Alexander, Imaging plus x: multimodal models of neurodegenerative disease, Current Opinion in Neurology 30 (4) (2017) 371.
[46] M. Schuster, K.K. Paliwal, Bidirectional recurrent neural networks, IEEE Transactions on Signal Processing 45 (11) (1997) 2673–2681.
[47] S. Ruder, An overview of multi-task learning in deep neural networks, arXiv preprint arXiv:1706.05098.

[48] H. Runtti, J. Mattila, M. van Gils, J. Koikkalainen, H. Soininen, J. Lötjönen, A.D.N. Initiative, et al., Quantitative evaluation of disease progression in a longitudinal mild cognitive impairment cohort, Journal of Alzheimer's Disease 39 (1) (2014) 49–61.
[49] M. Reuter, N.J. Schmansky, H.D. Rosas, B. Fischl, Within-subject template estimation for unbiased longitudinal image analysis, Neuroimage 61 (4) (2012) 1402–1418.
[50] D.P. Kingma, J. Ba, Adam: a method for stochastic optimization, arXiv preprint arXiv:1412.6980.
[51] R. Tibshirani, Regression shrinkage and selection via the lasso, Journal of the Royal Statistical Society: Series B (Methodological) 58 (1) (1996) 267–288.
[52] O. Sener, V. Koltun, Multi-task learning as multi-objective optimization, in: Advances in Neural Information Processing Systems, 2018, pp. 527–538.
[53] S. Lahmiri, A. Shmuel, Performance of machine learning methods applied to structural mri and adas cognitive scores in diagnosing alzheimer's disease, Biomedical Signal Processing and Control 52 (2019) 414–419.
[54] A. Esteva, A. Robicquet, B. Ramsundar, V. Kuleshov, M. DePristo, K. Chou, C. Cui, G. Corrado, S. Thrun, J. Dean, A guide to deep learning in healthcare, Nature Medicine 25 (1) (2019) 24.
[55] H.-I. Suk, S.-W. Lee, D. Shen, A.D.N. Initiative, et al., Deep ensemble learning of sparse regression models for brain disease diagnosis, Medical Image Analysis 37 (2017) 101–113.
[56] K. Ritter, J. Schumacher, M. Weygandt, R. Buchert, C. Allefeld, J.-D. Haynes, A.D. N. Initiative, et al., Multimodal prediction of conversion to alzheimer's disease based on incomplete biomarkers, Alzheimer's & Dementia: Diagnosis, Assessment & Disease Monitoring 1 (2) (2015) 206–215.

Shaker El-Sappagh received the bachelor's degree in computer science from Information Systems Department, Faculty of Computers and Information, Cairo University, Egypt, in 1997, and the master's degree from the same university in 2007. He received the Ph.D. degrees in computer science from Information Systems Department, Faculty of Computers and Information, Mansura University, Mansura, Egypt in 2015. In 2003, he joined the Department of Information Systems, Faculty of Computers and Information, Minia University, Egypt as a teaching assistant. Since June 2016, he has been with the Department of Information Systems, Faculty of computers and Information, Benha University as an assistant professor. Currently, he is a researcher at the Centro Singular de Investigación en Tecnoloxías Intelixentes, Universidade de Santiago de Compostela, Santiago, Spain. He has publications in clinical decision support systems and semantic intelligence. His current research interests include machine learning, medical informatics, (fuzzy) ontology engineering, distributed and hybrid clinical decision support systems, semantic data modelling, fuzzy expert systems, and cloud computing. He is a reviewer in many journals, and he is very interested in the diseases' diagnoses and treatment researches.

Tamer Abuhmed received his Ph.D. degree in Information and Telecommunication Engineering from Inha University in 2012. He is currently an Assistant Professor with the College of Computing, Sungkyunkwan University, South Korea. His research interests include applied cryptography and information security, network security, Internet security, and machine learning and its application to security and privacy problems.

S.M. Riazul Islam (M'10) received the B.S. and M.S. degrees in Applied Physics and Electronics from University of Dhaka, Bangladesh in 2003, and 2005, respectively and the Ph.D. degree in Information and Communication Engineering from Inha University, South Korea in 2012. He has been working at Sejong University, south Korea as an Assistant Professor at the Department of Computer Science and Engineering since March 2017. From 2014 to 2017, he worked at Inha University, South Korea as a Postdoctoral Fellow at the Wireless Communications Research Center. Dr. Islam was with the University of Dhaka, Bangladesh as an Assistant Professor and Lecturer at the Dept. of Electrical and Electronic Engineering for the period September 2005 to March 2014. In 2014, he worked at the Samsung R&D Institute Bangladesh (SRBD) as a Chief Engineer at the Dept. of Solution Lab for six months. His research interests include wireless communications, 5G & IoT, wireless health, bioinformatics, and machine learning.

Kyung-Sup Kwak (M'81) received the Ph.D. degree from the University of California at San Diego in 1988. From 1988 to 1989, he was with Hughes Network Systems, San Diego, CA, USA. From 1989 to 1990, he was with the IBM Network Analysis Center, Research Triangle Park, NC, USA. Since then, he has been with the School of Information and Communication Engineering, Inha University, South Korea, as a Professor, where he had been the Dean of the Graduate School of Information Technology and Telecommunications from 2001 to 2002. He has been the Director of the UWB Wireless Communications Research Center (formerly Key National IT Research Center), South Korea, since 2003. In 2006, he served as the President of Korean Institute of Communication Sciences, and in 2009, the President of Korea Institute of Intelligent Transport Systems. In 2008, he had been selected for Inha Fellow Professor and now for Inha Hanlim Fellow Professor. Dr. Kwak published more than 200 peer-reviewed journal papers and served as TPC/Track chairs/organizing chairs for several IEEE related conferences. His research interests include wireless communications, UWB systems, sensor networks, WBAN, and nano communications. He was a recipient of the number of awards, including the Engineering College Achievement Award from Inha University, the LG Paper Award, the Motorola Paper Award, the Haedong Prize of research, and various government awards from the Ministry of ICT, the President, and the Prime Minister of Korea, for his excellent research performances.