EEG-Based Neurodegenerative Disease Classification using LSTM Neural Networks

Michele Alessandrini
Dept. of Information Engineering
Università Politecnica delle Marche
Ancona, Italy
m.alessandrini@univpm.it

Giorgio Biagetti
Dept. of Information Engineering
Università Politecnica delle Marche
Ancona, Italy
g.biagetti@univpm.it

Paolo Crippa
Dept. of Information Engineering
Università Politecnica delle Marche
Ancona, Italy
p.crippa@univpm.it

Laura Falaschetti
Dept. of Information Engineering
Università Politecnica delle Marche
Ancona, Italy
l.falaschetti@univpm.it

Simona Luzzi
Dip. Medicina Sperimentale e Clinica
Università Politecnica delle Marche
Ancona, Italy
s.luzzi@staff.univpm.it

Claudio Turchetti
Dept. of Information Engineering
Università Politecnica delle Marche
Ancona, Italy
c.turchetti@univpm.it

2023 IEEE Statistical Signal Processing Workshop (SSP) | 978-1-6654-5245-8/23/$31.00 ©2023 IEEE | DOI: 10.1109/SSP53291.2023.10208023

***Abstract*—In recent years, the use of electroencephalography (EEG) for the clinical diagnosis of neurodegenerative diseases, such as Alzheimer's disease, frontotemporal dementia and dementia with Lewy bodies, has been extensively studied. The classification of these different neurodegenerative diseases can benefit from machine learning techniques which, compared to manual diagnosis methods, have higher reliability and higher recognition performance, being able to handle large amounts of data. The purpose of this work is to develop an automatic classification method that can recognize a number of different neurodegenerative diseases such the aforementioned ones, having similar corresponding EEGs or being difficult to discern by inspection from a human operator. We show how a recurrent neural network (RNN) based on long short-term memory (LSTM) elements can successfully perform the task of classification, when the data are properly pre-processed.**

***Index Terms*—recurrent neural network (RNN), deep neural network (DNN), electroencephalography (EEG), neurodegenerative disease, multi-class classification, disease classification, spectral domain**

I. Introduction

Alzheimer's disease (AD) and other neurodegenerative conditions, such as frontotemporal dementia (FTD) and dementia with Lewy bodies (LBD), result in severe effects on many aspects of a patient's life, like cognitive operation, memory and basic living functionality. Millions of people are affected by these diseases and their number is expected to increase in the future due to people living longer [1], [2].

Diagnosing such diseases in their early stages (e.g. mild cognitive impairment) it is important for a correct treatment [3]. Recently, the use of electroencephalography (EEG) has grown as a diagnostic tool, being a relatively simple and non-invasive method compared to other procedures like histological sampling or magnetic resonance [4]–[8].

The EEG can provide information, albeit indirectly, about the inner workings of a subject's brain; however its analysis can be a difficult task, both for the similarity of a pathology to other related ones or to normal symptoms of ageing, and because of the large amount of data that must be carefully examined by trained professionals for specific patterns. Research has thus focused on the application of machine learning algorithms to the detection of patterns in EEG signals, in order to help clinicians classify a subject's disease among a predefined set of possible diseases.

Machine learning algorithms, such as k-nearest neighbors (k-NN) and support-vector machines (SVM) [9]–[11], or deep neural networks (DNNs) [12]–[16], have resulted in many effective approaches to the mentioned task.

In our previous work [12] we successfully applied recurrent neural networks (RNNs) to distinguish AD subjects from healthy ones. The main advantage of RNNs is that they are able to model, unlike traditional neural networks, a collection of records (i.e. time collection) so that each pattern can be assumed to be dependent on previous ones [17]–[20]. The main disadvantages are: the risk of gradient explosion and vanishing problems; slow and complex training procedures; limitation in processing very long sequences if using Tanh or ReLU as activation functions.

In the present work an enhanced version of the previous method is proposed, to perform a multi-class classification of several neurodegenerative diseases presenting similar characteristics with regard to EEG recordings, namely AD, FTD and LBD, together with normal subjects (N). To the best of our knowledge, these diseases have been analyzed with several machine learning methods but always as a binary classification, as in [21]–[28].

Furthermore, due to similarity among the different diseases, together with the presence of significant artifacts in the recordings, negatively affecting the performance of the RNN, an improved RNN has been developed, and a further pre-

This research was partially funded by the European Union – Next Generation EU. Project Code: ECS00000041; Project CUP: C43C22000380007; Project Title: Innovation, digitalization and sustainability for the diffused economy in Central Italy – VITALITY.

TABLE I
DATASET CONSISTENCY

Class	Subjects	Duration (s)
Normal (N)	15	17,932
Alzheimer's disease (AD)	20	28,586
Frontotemporal dementia (FTD)	16	21,722
Dementia with Lewy bodies (LBD)	17	22,835
Total	68	91,075

processing step for removal of artifacts has been applied in order to achieve a good classification accuracy,

II. DATASET

The dataset used in this work consists of EEG recordings obtained from a set of hospitalized subjects, affected by different neurodegenerative diseases, and from a different set of healthy subjects used as control group. Data were collected in a hospital environment as part of ordinary diagnostic procedures performed by qualified clinicians. Data collection was performed in compliance with the Declaration of Helsinki for principles regarding human experimentation, with informed consent from every subject, and the data were anonymized before being used in the following elaborations.

Table I shows a summary of the composition of the dataset for the three different diseases we considered: AD, FTD and LBD. Class N represents the healthy subjects. The subjects were further divided into three groups, for training, validation and testing (see Section IV for details). The demographic distribution of the subjects (age, sex, education, illness duration) was chosen to be as similar as possible among the different groups.

The original dataset is composed of European data format (EDF) files, a standard file format frequently used for the storage and exchange of medical data. Each file includes a set of signals, which may differ among the different subjects (see Section III), all sampled at 128 Hz. Recordings of the different subjects have different durations, but all the signals for a given subject are acquired synchronously.

The equipment used for the acquisition was a Galileo BE Plus PRO Portable, Light version, capable of 37 total connections, consisting of 22 unipolar and 8 bipolar AC/DC inputs. The electrodes were applied in the standard 10-20 configuration (distances between adjacent electrodes, in percentage of the total available space on the subject's skull).

III. METHODOLOGY

The proposed method operates on the spectral representation of the input signals, thus converting the original, time-based data to a system of inputs not depending on time, but on a set of static parameters (features) expressed in a suitable basis.

To this end, the well-known discrete Karhunen-Loève transform (DKLT) [12], [29] is used, which has the property of separating the time-dependent components of a signal from its fixed features. The number of these features can also be truncated to the their principal components, with the advantage of both reducing the complexity of the problem and isolating relevant information from noise and other low informative content components. The resulting inputs are then processed by an RNN to perform the actual classification, as described in the following subsections.

A. Data Preparation

Data pre-processing was required because of the heterogeneity of the different input files. Notably, the signals included for each subject are a slightly different subset of standard EEG tracks. The number of existing signals for every subject varies from 21 from 23 and they are not in the same order.

For that reason a common subset of signals have been identified, resulting in the following 16 tracks: Fp1, Fp2, F7, F3, F4, F8, T3, C3, C4, T4, T5, P3, P4, T6, O1, O2. Data have therefore been saved with those signals in the given order, using the *npz* format (used by Python numpy library) for more convenience in the following steps. Data in this format are purely numerical, consisting of matrices of size $n \times 16$, where n is the number of time points and 16 is the number of EEG tracks. Such matrices retain no information on the meaning of every track, but are simply temporal series of 16-dimensional points associated to each subject.

Since different signals of a subject, and corresponding signals of different subjects, have significant differences in their magnitude, probably due to high sensitivity to physical variations of the electrode setup, statistical standardization has been applied to each signal of every subject. Standardization consists in scaling a given signal so that the final mean and standard deviation are 0 and 1, respectively, obtained with the formula:

$$y[n] = \frac{x[n] - \mu}{\sigma} \tag{1}$$

with μ and σ being the mean and standard deviation, respectively, of the original signal. This has proven to improve the final classification accuracy, due to the dependency on the variance of input data when extracting the principal signal features and training the RNN.

B. Data Windowing

Input data consist of relatively few subjects with records of varying length. In order to have inputs of fixed size and to increase the number of inputs for the RNN, and thus have a statistically significant dataset, input data have been split along the time axis in windows of fixed size w and o overlapping samples, so that the n-th data window is given by samples in the range

$$[\ (w-o)\,n\ ,\ (w-o)\,n+w-1\ ] \tag{2}$$

of the original samples.

Windowing is regularly performed in neural networks and other machine learning algorithms [12], [18], [30], with w and o being important hyper-parameters in such methods. Section IV describes the values that were used.

Data windows from the different subjects are finally merged, so that the final dataset has size $N \times w \times 16$, with N being the total number of data windows.

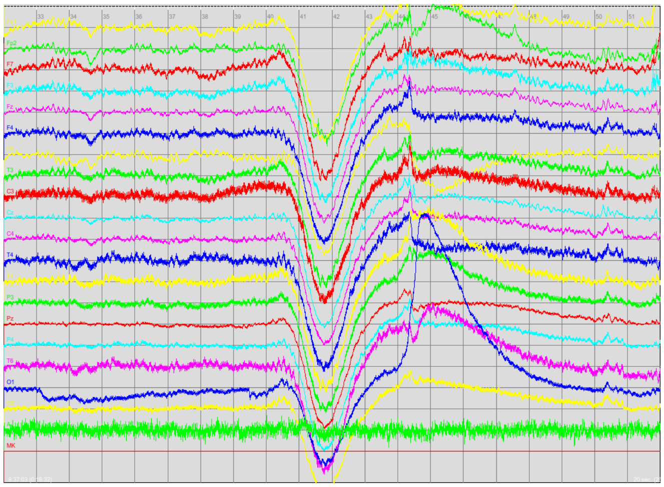

Fig. 1. Sample of EEG tracks corrupted by artifacts.

C. Removal of Artifacts

A significant portion of the input signals is contaminated by artifacts, that is portions of signals that are severely distorted due to temporary non-ideal conditions, like the subject moving or the supervisor adjusting the physical setup.

While a trained professional can easily identify such areas and even discern the valid signal component from the altered envelope, those artifacts have a negative impact on the machine learning algorithm, sensibly altering the signal properties for part of the inputs.

As shown in Fig. 1, artifacts mainly consist of sudden variations in signal magnitude, in a range that is often very large with respect to the normal signal.

Since the input data have previously been split in windows, it is convenient to discard the singular windows whose statistical properties differ significantly from the rest of the signal. Operatively, being the mean value and standard deviation of the full signal already been normalized to 0 and 1, respectively, and letting μ_w be the mean value of a data window, the window is discarded if

$$|\mu_w| > k \tag{3}$$

where k is a parameter that must be tuned in order to discard most of the windows containing artifacts, while maintaining a large portion of the original data.

D. Data Augmentation

Since the inputs used for training the RNN are not equally balanced among the different classes, a bias towards a specific class may arise. A solution to this problem is data augmentation. In this work oversampling has been used: data from classes with a small number of occurrences are duplicated and noise with a Gaussian distribution and a standard deviation of 0.03 is added to the new data, so as not to yield perfectly identical sequences. The resulting dataset is thus more uniformly distributed among the different classes.

E. Spectral Representation

As mentioned, an important step of the proposed algorithm is converting the original data to their spectral representation, using the DKLT. To this end, given a matrix $\mathbf{X}$, a general method consists in obtaining its singular value decomposition (SVD), so that

$$\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}^T \tag{4}$$

where $\mathbf{S}$ is a diagonal matrix containing the singular values, and $\mathbf{U}$, $\mathbf{V}$ contain the singular vectors. After that, a new matrix can be obtained, namely

$$\mathbf{X}^* = \mathbf{X}\mathbf{V} \tag{5}$$

which expresses $\mathbf{X}$ as a series of features, in terms of the vector base represented by $\mathbf{V}$.

It is then customary to perform principal component analysis (PCA), keeping only the most significant singular values and the associated columns of the base $\mathbf{V}$. In this way a dimensionality and complexity reduction is achieved, being $\mathbf{X}^*$ smaller than $\mathbf{X}$, and only the most representative components of the signals are retained, usually leading to better overall results.

In the case of DNNs, a problem arises from the input matrix being more than bidimensional; in this case it is tridimensional with size $N \times w \times 16$ (Section III-B). For this reason a series of dimension transposition and flattening has been applied to the input matrix to reduce it to the bidimensional case, then (4) and (5) can be used, and finally the matrix is reverted to its original form. The choice of the intermediate shape has been dictated by the need to obtain a proper matrix size for the SVD, such that the resulting vector base is a good representation with respect to the number of resulting features. In detail, the performed steps are:

$$\begin{aligned} &\mathbb{R}^{N\times w\times 16} \to \mathbb{R}^{N\times 16\times w} \to \mathbb{R}^{(16N)\times w} \\ &\xrightarrow{\text{DKLT}} \mathbb{R}^{(16N)\times p} \to \mathbb{R}^{N\times 16\times p} \to \mathbb{R}^{N\times p\times 16} \end{aligned} \tag{6}$$

where p is the number of components retained in the DKLT ($p < w$).

F. RNN Architecture

Practical RNNs can be effectively implemented using long short-term memory (LSTM) elements, which model both long-term and short-term states, solving many computational and stability problems that may arise in RNNs, especially for long sequences [31], [32].

The RNN used in this work consists of the layers listed in Table II. The specific sizes shown refer to a truncation to 50 components of the signal features, so that each input (data window) has size 50×16. The RNN in this configuration has 67,461 trainable parameters.

The core of the RNN are two cascaded LSTM layers. Before the LSTM layers there is a fully-connected layer (dense) that helps isolate the relevant features of input data; the last layer is another fully-connected one, whose output represents the classification in one of the four classes.

TABLE II
DETAILS OF RNN ARCHITECTURE

Layer	Input Size	Output Size	Parameters
Spatial 1D dropout	(–, 50, 16)	(–, 50, 16)	0
Dense 1	(–, 50, 16)	(–, 50, 64)	1,088
LSTM 1	(–, 50, 64)	(–, 50, 64)	33,024
LSTM 2	(–, 50, 64)	(–, 64)	33,024
Dense 2	(–, 64)	(–, 5)	325

A problem that must be overcome when training a DNN is overfitting, that is adapting the network too closely to the training data and resulting in poor prediction when using different data for testing. Overfitting is critical when the number of subjects is not very high and there are significant differences among different subjects, like in this case. An effective solution, beyond tuning all the network hyperparameters, is using dropout, that is discarding a different part of the data at every training epoch, and thus avoiding fitting the network too closely on the same data. Dropout is performed both in the internal components of the LSTMs and by a dedicated layer, namely the spatial 1D dropout, dropping entire 1D feature maps instead of individual elements.

Globally, the output is optimized through a loss function of sparse categorical crossentropy type, representing the loss function to be minimized in the train process as

$$J(W) = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)\right] \qquad (7)$$

where W is the set of network parameters (like node weights), N is the number of input features, y_i and $\hat{y}_i$ are the true and predicted outputs, respectively.

IV. Experimental Results

The RNN was developed with TensorFlow 2.11.0 and Keras 2.11.0 on a computer with an Intel Core i7-6800K CPU with 32 GiB of RAM.

To test the effectiveness of the RNN in classifying new subjects with respect to the ones used for the training, one subject for every category (for a total of four) has been isolated from the rest of the data and used as validation data during the training phase, and two subjects per category (for a total of eight) have been isolated and used as testing data, not affecting the training process.

The input data have been split in windows of 256 samples with 50% overlap, which showed to be optimal in our previous work with data from the same dataset [12]. For the DKLT, 50 principal components have been retained.

Table III shows all the relevant experimental parameters and the results, in particular the global accuracy reached for the validation data (73.3%) and the testing data (75.3%).

Another important result is the confusion matrix associated with the different classes, reported in Fig. 2. From that, a set of interesting metrics can be derived about the performance of the classifier with respect to the single classes: sensitivity, precision and F1-score are reported in Table IV.

TABLE III
EXPERIMENTAL RESULTS

Parameter	Value
Window size w (samples)	256
Window overlap o (samples)	128
Principal components p	50
Artifact threshold k	1.25
Input features N	46,468
Training time (s)	2164
Testing time (s)	4.2
Validation accuracy (%)	73.3
Testing accuracy (%)	75.3

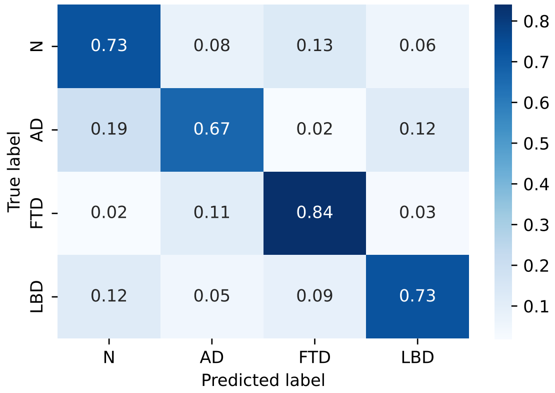

Fig. 2. Confusion matrix for testing data.

TABLE IV
PERFORMANCE OF THE CLASSIFIER EVALUATED ON TESTING DATA

Class	Sensitivity (%)	Precision (%)	F1-score (%)
N	73.2	73.6	73.4
AD	67.3	71.3	69.3
FTD	83.9	83.9	83.9
LBD	73.5	66.9	70.0

It can be seen from the previous results that some classes are recognized with high accuracy, while other ones suffer from a larger error. This can be explained by the relative small dimension of the dataset, together with significant differences in the input data, even for subjects in the same classes.

V. Conclusions

In this work an RNN has been developed for the classification of neurodegenerative diseases from EEG recordings. The RNN operates on the spectral representation of the data and uses LSTM layers as its core, after an artifact removal step in order to improve the network performance. Results show that a global accuracy of 75.3% can be achieved on a subset of the data used as testing, belonging to entirely different subjects with respect to the training material. While some classes are recognized with less accuracy, nonetheless the obtained classification can be useful in spotting possible disease condition in patients and as an aid to the expensive work of professionals who must examine a large amount of data for specific patterns.

References

[1] J. Xie, C. Brayne, and F. E. Matthews, “Survival times in people with dementia: analysis from population based cohort study with 14 year follow-up,” *BMJ*, vol. 336, no. 7638, pp. 258–262, 2008.

[2] J. Jeong, “EEG dynamics in patients with Alzheimer’s disease,” *Clinical Neurophysiology*, vol. 115, no. 7, pp. 1490–1505, 2004.

[3] R. C. Petersen, “Early diagnosis of Alzheimer’s disease: is MCI too late?” *Current Alzheimer Research*, vol. 6, no. 4, pp. 324–330, 2009.

[4] A. Tsolaki, D. Kazis, I. Kompatsiaris, V. Kosmidou, and M. Tsolaki, “Electroencephalogram and Alzheimer’s disease: clinical and research approaches,” *International Journal of Alzheimer’s Disease*, vol. 2014, 2014.

[5] N. N. Kulkarni and V. K. Bairagi, “Electroencephalogram based diagnosis of Alzheimer disease,” in *2015 IEEE 9th International Conference on Intelligent Systems and Control (ISCO)*, 2015, pp. 1–5.

[6] T. H. Falk, F. J. Fraga, L. Trambaiolli, and R. Anghinah, “EEG amplitude modulation analysis for semi-automated diagnosis of Alzheimer’s disease,” *EURASIP Journal on Advances in Signal Processing*, vol. 2012, no. 1, pp. 1–9, 2012.

[7] G. Fiscon, E. Weitschek, G. Felici, P. Bertolazzi, S. De Salvo, P. Bramanti, and M. C. De Cola, “Alzheimer’s disease patients classification through EEG signals processing,” in *2014 IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*. IEEE, 2014, pp. 105–112.

[8] N. Houmani, F. Vialatte, E. Gallego-Jutglà, G. Dreyfus, V.-H. Nguyen-Michel, J. Mariani, and K. Kinugawa, “Diagnosis of Alzheimer’s disease with electroencephalography in a differential framework,” *PloS ONE*, vol. 13, no. 3, p. e0193607, 2018.

[9] H. Yu, L. Zhu, L. Cai, J. Wang, J. Liu, R. Wang, and Z. Zhang, “Identification of Alzheimer’s EEG with a WVG network-based fuzzy learning approach,” *Frontiers in Neuroscience*, vol. 14, p. 641, 2020.

[10] M. Tanveer, B. Richhariya, R. Khan, A. Rashid, P. Khanna, M. Prasad, and C. Lin, “Machine learning techniques for the diagnosis of Alzheimer’s disease: A review,” *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 16, no. 1s, pp. 1–35, 2020.

[11] G. Biagetti, P. Crippa, L. Falaschetti, S. Luzzi, and C. Turchetti, “Classification of Alzheimer’s disease from EEG signal using robust-PCA feature extraction,” *Procedia Computer Science*, vol. 192, pp. 3114–3122, 2021.

[12] M. Alessandrini, G. Biagetti, P. Crippa, L. Falaschetti, S. Luzzi, and C. Turchetti, “EEG-based Alzheimer’s disease recognition using robust-PCA and LSTM recurrent neural network,” *Sensors*, vol. 22, no. 10, p. 3696, 2022.

[13] M. Nguyen, N. Sun, D. C. Alexander, J. Feng, and B. T. Yeo, “Modeling Alzheimer’s disease progression using deep recurrent neural networks,” in *2018 International Workshop on Pattern Recognition in Neuroimaging (PRNI)*, 2018, pp. 1–4.

[14] M. Nguyen, T. He, L. An, D. C. Alexander, J. Feng, B. T. Yeo, A. D. N. Initiative *et al.*, “Predicting Alzheimer’s disease progression using deep recurrent neural networks,” *NeuroImage*, vol. 222, p. 117203, 2020.

[15] T. Wang, R. G. Qiu, and M. Yu, “Predictive modeling of the progression of Alzheimer’s disease with recurrent neural networks,” *Scientific Reports*, vol. 8, no. 1, pp. 1–12, 2018.

[16] S. Yang, H.-C. Chen, C.-H. Wu, M.-N. Wu, and C.-H. Yang, “Forecasting of the prevalence of dementia using the LSTM neural network in Taiwan,” *Mathematics*, vol. 9, no. 5, p. 488, 2021.

[17] A.-N. Sharkawy, “Principle of neural network and its main types: Review,” *Journal of Advances in Applied and Computational Mathematics*, vol. 7, p. 8–19, Aug. 2020.

[18] M. Musci, D. De Martini, N. Blago, T. Facchinetti, and M. Piastra, “Online fall detection using recurrent neural networks on smart wearable devices,” *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 3, pp. 1276–1289, 2021.

[19] G. Chevalier, “LSTMs for human activity recognition,” https://github.com/guillaume-chevalier/LSTM-Human-Activity-Recognition, 2016, accessed: 2021-04-16.

[20] M. Alessandrini, G. Biagetti, P. Crippa, L. Falaschetti, and C. Turchetti, “Recurrent neural network for human activity recognition in embedded systems using PPG and accelerometer data,” *Electronics*, vol. 10, no. 14, p. 1715, 2021.

[21] A. Miltiadous, K. D. Tzimourta, N. Giannakeas, M. G. Tsipouras, T. Afrantou, P. Ioannidis, and A. T. Tzallas, “Alzheimer’s disease and frontotemporal dementia: A robust classification method of EEG signals and a comparison of validation methods,” *Diagnostics*, vol. 11, no. 8, 2021.

[22] M. Lindau, V. Jelic, S.-E. Johansson, C. Andersen, L.-O. Wahlund, and O. Almkvist, “Quantitative EEG Abnormalities and Cognitive Dysfunctions in Frontotemporal Dementia and Alzheimer’s Disease,” *Dementia and Geriatric Cognitive Disorders*, vol. 15, no. 2, pp. 106–114, 01 2003.

[23] K. Nishida, M. Yoshimura, T. Isotani, T. Yoshida, Y. Kitaura, A. Saito, H. Mii, M. Kato, Y. Takekita, A. Suwa, S. Morita, and T. Kinoshita, “Differences in quantitative EEG between frontotemporal dementia and Alzheimer’s disease as revealed by LORETA,” *Clinical Neurophysiology*, vol. 122, no. 9, pp. 1718–1725, 2011.

[24] F. Caso, M. Cursi, G. Magnani, G. Fanelli, M. Falautano, G. Comi, L. Leocani, and F. Minicucci, “Quantitative EEG and LORETA: valuable tools in discerning FTD from AD?” *Neurobiology of Aging*, vol. 33, no. 10, pp. 2343–2356, 2012.

[25] M. Dottori, L. Sedeño, M. Martorell Caro, F. Alifano, E. Hesse, E. Mikulan, A. M. García, A. Ruiz-Tagle, P. Lillo, A. Slachevsky *et al.*, “Towards affordable biomarkers of frontotemporal dementia: a classification study via network’s information sharing,” *Scientific reports*, vol. 7, no. 1, pp. 1–12, 2017.

[26] G. Fiscon, E. Weitschek, A. Cialini, G. Felici, P. Bertolazzi, S. De Salvo, A. Bramanti, P. Bramanti, and M. C. De Cola, “Combining EEG signal processing with supervised methods for Alzheimer’s patients classification,” *BMC medical informatics and decision making*, vol. 18, no. 1, pp. 1–10, 2018.

[27] M. S. Safi and S. M. M. Safi, “Early detection of Alzheimer’s disease from EEG signals using Hjorth parameters,” *Biomedical Signal Processing and Control*, vol. 65, p. 102338, 2021.

[28] R. Nardone, L. Sebastianelli, V. Versace, L. Saltuari, P. Lochner, V. Frey, S. Golaszewski, F. Brigo, E. Trinka, and Y. Höller, “Usefulness of EEG techniques in distinguishing frontotemporal dementia from Alzheimer’s disease and other dementias,” *Disease markers*, vol. 2018, 2018.

[29] G. Biagetti, P. Crippa, L. Falaschetti, and C. Turchetti, “A machine learning approach to the identification of dynamical nonlinear systems,” in *2019 27th European Signal Processing Conference (EUSIPCO)*. IEEE, 2019, pp. 1–5.

[30] G. Biagetti, P. Crippa, L. Falaschetti, E. Focante, N. Martínez Madrid, and R. Seepold, “Machine learning and data fusion techniques applied to physical activity classification using photoplethysmographic and accelerometric signals,” *Procedia Computer Science*, vol. 176, pp. 3103–3111, 10 2020.

[31] S. Hochreiter, “The vanishing gradient problem during learning recurrent neural nets and problem solutions,” *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 6, no. 02, pp. 107–116, 1998.

[32] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.