



# Predicting Epileptic Seizures from Intracranial EEG Using LSTM-Based Multi-task Learning

Xuelin Ma<sup>1,3</sup> , Shuang Qiu<sup>1</sup> , Yuxing Zhang<sup>2</sup> , Xiaoqin Lian<sup>2</sup> ,  
and Huiguang He<sup>1,3,4</sup>  

<sup>1</sup> Research Center for Brain-inspired Intelligence  
and National Laboratory of Pattern Recognition, Institute of Automation,  
Chinese Academy of Sciences, Beijing, China

{maxuelin2015, shuang.qiu, huiguang.he}@ia.ac.cn

<sup>2</sup> School of Computer and Information Engineering,  
Beijing Technology and Business University, Beijing, China

zhang.y.x0325@foxmail.com, lianxq@263.net

<sup>3</sup> University of Chinese Academy of Sciences, Beijing, China

<sup>4</sup> Center for Excellence in Brain Science and Intelligence Technology,  
Chinese Academy of Sciences, Beijing, China

**Abstract.** Epilepsy afflicts nearly 1% of the world's population, and is characterized by the occurrence of spontaneous seizures. It's important to make prediction before seizures, so that epileptic can prevent seizures taking place on some specific occasions to avoid suffering from great damage. The previous work in seizure prediction paid less attention to the time-series information and their performances may also be restricted to the small training data. In this study, we proposed a Long Short-Term Memory (LSTM)-based multi-task learning (MTL) framework for seizure prediction. The LSTM unit was used to process the sequential data and the MTL framework was applied to perform prediction and latency regression simultaneously. We evaluated the proposed method in the American Epilepsy Society Seizure Prediction Challenge dataset and obtained an average prediction accuracy of 89.36%, which was 3.41% higher than the reported state-of-the-art. In addition, the input data and output of middle layers were visualized. The visual and experimental results demonstrated the superior performance of our proposed LSTM-MTL method for seizure prediction.

**Keywords:** Seizure prediction · LSTM · Multi-task learning  
Intracranial EEG

## 1 Introduction

Epilepsy is a common brain disorder characterized by intermittent abnormal neuronal firing in the brain which can lead to seizures [15]. Seizure forecasting

systems have the potential to help epileptic to lead a more normal lives [20, 22]. With these systems, epileptic could avoid to do dangerous activities like driving or swimming and medications could be administered before impending seizures. Therefore, predicting epilepsy before seizure it's very important for building the seizure forecasting systems [15].

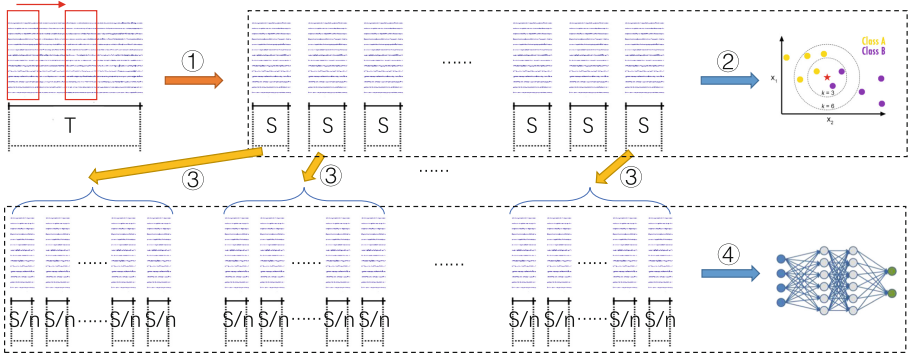
Intracranial EEG (iEEG) is a chronological electrophysiological record of epileptic. Seizure prediction from iEEG has been extensively studied in the previous work. Most work to date relies on spectral information and pays attention to traditional machine learning methods like k-nearest neighbors algorithm (KNN), SVM, Random Forest and XGBoost [4], etc. On the other hand, iEEG also contains a lot of timing information except spectral information. Most of the previous work doesn't utilize the sequential information of iEEG data [16].

Inspired by the success of deep recurrent neural networks (RNNs) for speech feature learning and time series prediction [8, 9], we intend to build an effective seizure prediction model based on deep Long Short-Term Memory (LSTM) network. The applications of LSTM remain a challenge in neuroimaging domain. One of the reasons is the limited number of samples, which makes it difficult for training large-scale networks with millions of parameters [1]. This problem can be alleviated by applying sliding window approach over the raw data, which would increase the amount of training samples hundreds of times [7, 11, 19].

Actually, most of the seizure datasets focus on the classification between preictal state (prior to seizure) and interictal state (between seizures, or baseline) [12, 13]. The preictal data are recorded with the latency before seizure, which can be utilized as additional information for seizure prediction. Multi-task learning (MTL) [3] aims to improve generalization performance of multiple tasks by appropriately sharing relevant information across them. Some studies showed that the MTL method performed better than methods based on individual learning [5, 17, 21]. Therefore, the additional latency information can be integrated by multi-task learning.

In this paper, we proposed a novel Long Short-Term Memory based multi-task learning framework (refer to LSTM-MTL) for seizure prediction. The LSTM network can inherently process the sequential data, and the multi-task learning framework performs prediction and latency regression simultaneously to improve the prediction performance. We evaluated our proposal on public seizure dataset and showed that the LSTM-MTL framework outperformed the KNN and XGBoost methods. The LSTM-MTL model showed the prediction AUC up to 89.36%, which is 3.41% higher than the reported state-of-the-art and 2.5% higher than the LSTM model without MTL. The results demonstrated the effectiveness of our proposed LSTM based multi-task learning framework.

In addition, the input data and the output of middle layers of the multi-task LSTM network are visualized for intuitive perception. The visual results demonstrated that the representation learning ability of the network is remarkable. The linearly inseparable original data become linearly separable gradually through layer-by-layer process.



**Fig. 1.** The overall preprocessing flowchart. Step 1: Sliding Window Approach with window length of  $S$  and 50% overlapping. Step 2: Feature extraction (see Sect. 2.1) over each sample for traditional classifier input. Step 3: Sliding Window Approach with window length of  $S/n$  and no overlapping. Step 4: Feature extraction (see Sect. 2.1) over the sequential subsamples for LSTM networks.

The rest of this paper is organized as follows: Sect. 2 introduces the method we adopted and the framework we proposed. Section 3 describes the experiments in detail. Section 4 shows the experiment result and some discussion about it. Section 5 is the conclusion of this work.

## 2 The Proposed Method

In this section, we introduce the feature extraction, sliding window approach, training and testing strategies and the proposed LSTM based multi-task learning architecture.

### 2.1 Feature Extraction

Typically, given an iEEG record segment  $r \in \mathbb{R}^{C \times S}$ , where  $C$  denotes the numbers of channels,  $S$  denotes the time steps, the feature vector  $x \in \mathbb{R}^{1 \times (C^2 + 8 * C)}$  are extracted with following components:

One part of the features are the average spectral power in six frequency bands of each channel and the standard deviation of this six powers, resulting a vector with length of  $C * 7$ . The six frequency bands are delta (0.1–4 Hz), theta (4–8 Hz), alpha (8–12 Hz), beta (12–30 Hz), low gamma (30–70 Hz) and high gamma (70–180 Hz).

Another part of the features are the correlation in time domain and frequency domain (upper triangle values of correlation matrices) with their eigenvalues, resulting a vector with length of  $C * (C + 1)$ . Therefore, the length of feature vector  $x$  is  $C^2 + 8 * C$ .

## 2.2 Sliding Window Approach

We preprocess the raw data to obtain more samples for training deep networks. The overall preprocessing flowchart of our proposed method is shown in Fig. 1.

Typically, the given datasets can be denoted as

$$D^i = \{(E^1, y^1, L^1), \dots, (E^{N_i}, y^{N_i}, L^{N_i})\},$$

where  $N_i$  denotes the total number of recorded segments for patient  $i$ . The input matrix  $E^j \in \mathbb{R}^{C \times T}$  of segment  $j$ , where  $1 \leq j \leq N_i$ , contains the signals of  $C$  recorded electrodes and  $T$  discretized time steps recorded per segment. The corresponding class label and latency of segment  $j$  are denoted by  $y^j$  and  $L^j$ , respectively.  $L^j$  is defined as the beginning timesteps of the sequential segment.

The sliding window approach is applied to divide the segment data into individual samples, which are used for later processing. Each sample has a fixed length  $S$ , with 50% overlapping between continuous neighbors.

For traditional classifier, a sample can be denoted as follow:

$$r_k^j \in \mathbb{R}^{C \times S}, \quad (1)$$

$$y_k^j = y^j \quad (2)$$

where  $1 \leq j \leq N_i$  and  $1 \leq k \leq \lfloor \frac{T}{(S/2)} \rfloor - 1$ .

By feature extraction,  $r_k^j$  is converted into  $x_k^j \in \mathbb{R}^{1 \times (C^2 + 8 * C)}$ , which can be used as the input to traditional classifiers.

For sequential deep learning models, a sample  $r_k^j$  is clipped into  $n$  non-overlapping sequential records and can be denoted as follow:

$$rs_k^j \in \mathbb{R}^{n \times C \times \frac{S}{n}}, \quad (3)$$

$$y_k^j = y^j, \quad (4)$$

$$l_k^j = \begin{cases} L^j + \frac{S/2}{T}, & \text{if } y_k^j = 1 \\ 0, & \text{if } y_k^j = 0 \end{cases} \quad (5)$$

where  $1 \leq j \leq N_i$  and  $1 \leq k \leq \lfloor \frac{T}{(S/2)} \rfloor - 1$ .

In the same way,  $rs_k^j$  is converted into  $xs_k^j \in \mathbb{R}^{n \times 1 \times (C^2 + 8 * C)}$ , which can be used as the input to sequential deep learning models.

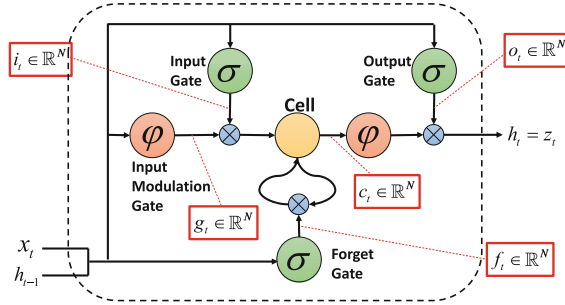
## 2.3 Training and Testing Input

In training, using samples  $\{(r_1^1, y_1^1), \dots, (r_k^1, y_k^1), \dots, (r_1^{N_i}, y_1^{N_i}), \dots, (r_k^{N_i}, y_k^{N_i})\}$  and  $\{(rs_1^1, y_1^1), \dots, (rs_k^1, y_k^1), \dots, (rs_1^{N_i}, y_1^{N_i}), \dots, (rs_k^{N_i}, y_k^{N_i})\}$  as input to traditional classifiers and LSTM-based models, respectively.

In testing, we evaluated the models with sample data  $r_k^j$  or  $rs_k^j$ , used the mean rule to fuse  $k$  predicted sample label probabilities  $pred\_p_k^j$  into predicted segment label probability  $pred\_p^j$  and computed the segment-wise Area Under Curve (AUC).

## 2.4 Long Short-Term Memory Network

RNN is a class of neural network that maintains internal hidden states to model the dynamic temporal behaviour of sequences through directed cyclic connections between its units. LSTM extends RNN by adding three gates to an RNN neuron, which enable LSTM to learn long-term dependency in a sequence, and make it easier to optimize [10]. There is sequential information containing in the iEEG data and LSTM is an excellent model for encoding sequential iEEG data.



**Fig. 2.** A block diagram of the LSTM unit (there are minor differences in comparison to [6], with  $\phi$  symbol after cell element and some annotation about the dimension of each state inside the red box). (Color figure online)

A block diagram of the LSTM unit is shown in Fig. 2, and the recurrence equations is as follows:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i), \quad (6)$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f), \quad (7)$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o), \quad (8)$$

$$g_t = \phi(W_{xc}x_t + W_{hc}h_{t-1} + b_c), \quad (9)$$

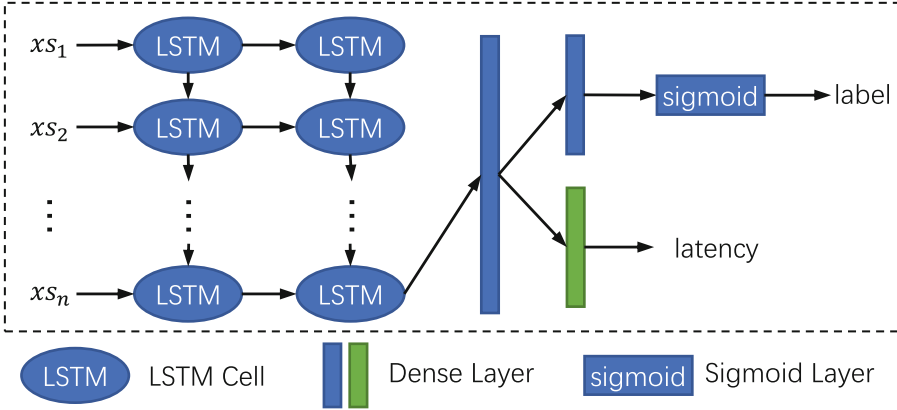
$$c_t = f_t \odot c_{t-1} + i_t \odot g_t, \quad (10)$$

$$h_t = o_t \odot \phi(c_t). \quad (11)$$

A LSTM unit contains an input gate  $i_t$ , a forget gate  $f_t$ , a cell  $c_t$ , an output gate  $o_t$  and an output response  $h_t$ . The input gate and the forget gate govern the information flow. The output gate controls how much information from the cell is passed to the output  $h_t$ . The memory cell has a self-connected recurrent edge of weight 1, ensuring that the gradient is able to pass across many time steps without vanishing or exploding. Units are connected recurrently to each other, replacing the usual hidden units of ordinary recurrent networks.

## 2.5 LSTM Based Multi-task Learning

While class label  $y_k^j$  only provides hard and limited information, the latency  $l_k^j$  can show much softer and more plentiful details about the seizure. To improve



**Fig. 3.** The architecture of Multi-task LSTM network. One task for seizure prediction and another for latency regression (green block). (Color figure online)

the accuracy and robustness of seizure prediction, we propose a LSTM based multi-task learning (LSTM-MTL) framework as shown in Fig. 3 and describe in detail as follow.

The LSTM-MTL model takes sequential data  $xs_k^j$  as input. Two LSTM layers are cascaded to encoding the sequential information of input. The last timestep output of the second LSTM layer is followed by a dense layer for learning representation further. For prediction task, the dense layer is followed by another dense layer with two nodes, which used a sigmoid activate function to generate the final prediction. Simultaneously, for latency regression task, a dense layer with one node is utilized to regress urgency degree from the output of first dense layer.

The loss function of prediction task  $L_{pred}$  is sigmoid cross-entropy and the loss function of latency regression  $L_{reg}$  is mean square error. For multi-task learning, we define a loss function to combine above loss functions as follow:

$$L = \alpha L_{pred} + (1 - \alpha) L_{reg}, \quad (12)$$

where  $\alpha$  is a hyper-parameter between 0 and 1.

### 3 Experiments

This section first describes the dataset used for evaluation, and then describes the experimental setup of the proposed method. Then, we briefly describe the details and parameter settings of the comparison models.

#### 3.1 Dataset

We validated the effectiveness of our method on a public seizure prediction competition dataset [12]. Seizure forecasting focuses on identifying a *preictal* (*prior*

to seizure) state that can be differentiated from the interictal (between seizures, or baseline), ictal (seizure), and postictal (after seizures) states, especially the interictal state. The goal of the dataset is to demonstrate the existence and accurate classification of the *preictal* brain state in humans with epilepsy. It's a binary classification. The dataset contains 2 patients (263 and 210 samples, respectively). Every 10 min of the data is intercepted as a sample. For detailed information, please refer to the website of Kaggle [12].

### 3.2 Implementation Details

The whole neural networks were implemented with the Keras framework and trained on a Nvidia 1080Ti GPU from scratch in a fully-supervised manner. The Adam algorithm was used to optimize the loss function with a learning rate of  $0.5 * 10^{-4}$ . The dropout probability was 0.5. The hidden states number of the LSTM cell was 32. There were 128 nodes in the first dense layer. The hyperparameter  $\alpha$  in loss function was tuned to balance the magnitude of two types of loss.

### 3.3 Comparison Models

Then, we compared our approach with two baseline methods k-nearest neighbors algorithm (KNN) and the eXtreme Gradient Boosting (XGBoost) algorithm which were widely used, as well as the LSTM network without multi-task learning for component evaluation. Here we briefly describe some of the details and parameter settings used in these methods.

**KNN.** The k-nearest neighbors algorithm (KNN) is one of the simplest and most common classification methods based on supervised learning which is classified as a simple and lazy classifier due to its lack of complexity. In this algorithm,  $k$  is the number of neighbors, which may largely affect the classification performance.  $k$  was selected by cross-validation on training set ( $k = 1, 2, 3, \dots, 18, 19, 20, 25, 30, 35, 40$ ).

**XGBoost.** The eXtreme Gradient Boosting (XGBoost) algorithm is a well-designed Gradient Boosted Decision Tree (GBDT) algorithm, which demonstrates its state-of-the-art advantages in the scientific research of machine learning and data mining problem.

Two hyperparameters of XGBoost for preventing overfitting was adjusted through cross-validation on training set ( $max\_depth = 3, 4, 5, \dots, 8, 9, 10$  and  $subsample = 0.5, 0.6, 0.7, 0.8$ ).

**LSTM Network with Single-Task Learning.** To evaluate the performance of LSTM-MTL framework strictly, LSTM network with single-task learning

(LSTM-STL) is used in the experiment. The LSTM-STL performs seizure prediction without latency regression task, as shown in Fig. 3 with blue blocks. That is to say, the hyperparameter  $\alpha$  of LSTM-MTL framework is equal to 1. For comparison purpose, we kept all the hyper-parameters of LSTM-STL the same with LSTM-MTL.

## 4 Results and Discussion

In this section, the prediction AUC of different models were showed in Table 1. In addition, we give some visualization figures to explore into the model and have some discussions about the results.

**Table 1.** Prediction AUC of different models

	Patient_1	Patient_2	Average
KNN	41.14	52.78	46.96
XGB	85.93	66.49	76.21
LSTM-STL	89.66	84.06	86.86
LSTM-MTL	<b>92.37</b>	<b>86.34</b>	<b>89.36</b>

### 4.1 Compared with Baseline Performance

Overall, our proposed LSTM-MTL method significantly outperformed the KNN and XGBoost algorithms.

KNN algorithm performed badly, showing an average AUC score even below 50%. This algorithm is very sensitive to local distribution of features and may not work well. In general, no determinate comment can be made about the performance of the KNN classifier in EEG-related problems [18].

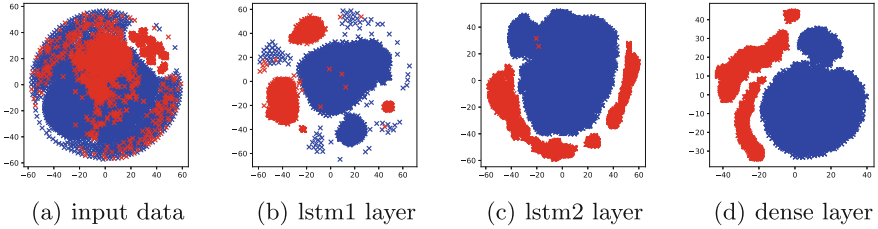
XGBoost performed relatively better with an AUC of 76.21%, but there was a gap between its performance and the state-of-the-art 85.95% [2]. Actually, the reported state-of-the-art was achieved by an ensemble of different features and different classical classifiers. Using only XGBoost algorithm is hard to get a comparable performance.

LSTM networks utilized the same types of features with KNN and XGBoost algorithms but achieved comparable or better performance against state-of-the-art, which illustrated that the LSTM networks can learn useful information from sequential iEEG features.

### 4.2 Compared with LSTM-STL

LSTM-STL network achieved an average AUC of 86.86%, which was comparable with the state-of-the-art. LSTM-MTL outperformed LSTM-STL with AUC





**Fig. 4.** The T-SNE feature visualizations of input data and output of middle layers. (a) The visualization of input data. (b) The visualization of the output feature of the first lstm layer. (c) The visualization of the output feature of the first lstm layer. (d) The visualization of the dense layer output. The figure is best viewed under the electronic edition.

improvement of 2.5%. This results demonstrated the necessity of adding latency regression as an additional task.

The latency of segments can provide urgency degree information about the seizure. Through combining the latency information, the LSTM network can take full advantage of limited data and performed better in prediction task. LSTM-MTL can not only improve the prediction accuracy but also report an urgency degree about seizure, which is important for patients to take nichetargeting action.

### 4.3 Visualization

Finally, we visualized the input data as well as the output of LSTM layers and the dense layer of LSTM-MTL by t-distributed Stochastic Neighbor Embedding (T-SNE) [14]. T-SNE is a tool to visualize high-dimensional data, converting similarities between data points to joint probabilities and tries to minimize the Kullback-Leibler divergence between the joint probabilities of the low-dimensional embedding and the high-dimensional data. The visual results showed that the data became increasingly linearly detachable along with the layer-by-layer process. An visualization example of Patient 1 is shown in Fig. 4. Two classes of the input data are aliasing. Through the first LSTM layer, the data clustered and one cluster contained nearly only one class of data. Through the second LSTM layer, two classes of data could be separated by a simple quadratic function in two-dimension space. Through the dense layer, the data became more linearly detachable.

## 5 Conclusion and Future Work

In this paper we presented a novel LSTM based multi-task learning framework for seizure prediction. The proposed multi-task framework performed prediction and latency regression simultaneously and the prediction performance was

improved through this way. Overall, the average AUC score of LSTM-MTL was 89.36%, which was 3.41% higher than the state-of-the-art.

The visualization of middle layers output illustrated the sequential representation ability of the proposed LSTM-MTL network. In the future, we will visualize the weight map of the LSTM units to explore the significations of each channel and each feature, which can be helpful for channel reduction or feature selection.

**Acknowledgments.** This work was supported by National Natural Science Foundation of China (91520202, 81701785), Youth Innovation Promotion Association CAS, the CAS Scientific Research Equipment Development Project (YJKYYQ20170050) and the Beijing Municipal Science&Technology Commission (Z181100008918010).

## References

1. Bashivan, P., Rish, I., Yeasin, M., Codella, N.: Learning representations from EEG with deep recurrent-convolutional neural networks. arXiv preprint [arXiv:1511.06448](https://arxiv.org/abs/1511.06448) (2015)
2. Brinkmann, B.H., et al.: Crowdsourcing reproducible seizure forecasting in human and canine epilepsy. *Brain* **139**(6), 1713–1722 (2016)
3. Caruana, R.: Multitask learning. *Mach. Learn.* **28**(1), 41–75 (1997)
4. Chen, T., Guestrin, C.: XGBoost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794. ACM (2016)
5. Doersch, C., Zisserman, A.: Multi-task self-supervised visual learning. In: The IEEE International Conference on Computer Vision (ICCV) (2017)
6. Donahue, J., et al.: Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2625–2634 (2015)
7. Golmohammadi, M., et al.: Deep architectures for automated seizure detection in scalp EEGs. arXiv preprint [arXiv:1712.09776](https://arxiv.org/abs/1712.09776) (2017)
8. Graves, A., Liwicki, M., Bunke, H., Schmidhuber, J., Fernández, S.: Unconstrained on-line handwriting recognition with recurrent neural networks. In: Advances in neural information processing systems, pp. 577–584 (2008)
9. Graves, A., Mohamed, A.-R., Hinton, G.: Speech recognition with deep recurrent neural networks. In: 2013 IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP), pp. 6645–6649. IEEE (2013)
10. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997)
11. Jaffe, A.S.: Long short-term memory recurrent neural networks for classification of acute hypotensive episodes. Ph.D. thesis, Massachusetts Institute of Technology (2017)
12. kaggle: American epilepsy society seizure prediction challenge. <https://www.kaggle.com/c/seizure-prediction/data>
13. kaggle: Melbourne university aes/mathworks/nih seizure prediction. <https://www.kaggle.com/c/melbourne-university-seizure-prediction/data>
14. van der Maaten, L., Hinton, G.: Visualizing data using t-SNE. *J. Mach. Learn. Res.* **9**(Nov), 2579–2605 (2008)

15. Mormann, F., Andrzejak, R.G., Elger, C.E., Lehnertz, K.: Seizure prediction: the long and winding road. *Brain* **130**(2), 314–333 (2006)
16. O'Regan, S., Faul, S., Marnane, W.: Automatic detection of EEG artefacts arising from head movements using EEG and gyroscope signals. *Med. Eng. Phys.* **35**(7), 867–874 (2013)
17. Ranjan, R., Patel, V.M., Chellappa, R.: Hyperface: a deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* (2017)
18. Tahernezhad-Javazm, F., Azimirad, V., Shoaran, M.: A review and experimental study on the application of classifiers and evolutionary algorithms in EEG-based brain-machine interface systems. *J. Neural Eng.* **15**(2), 021007 (2018)
19. Thodoroff, P., Pineau, J., Lim, A.: Learning robust features using deep learning for automatic seizure detection. In: *Machine Learning for Healthcare Conference*, pp. 178–190 (2016)
20. Tzallas, A.T., Tsipouras, M.G., Fotiadis, D.I.: Epileptic seizure detection in EEGs using time-frequency analysis. *IEEE Trans. Inf. Technol. Biomed.* **13**(5), 703–710 (2009)
21. Van Esbroeck, A., Smith, L., Syed, Z., Singh, S., Karam, Z.: Multi-task seizure detection: addressing intra-patient variation in seizure morphologies. *Mach. Learn.* **102**(3), 309–321 (2016)
22. Wang, Y., et al.: A cauchy-based state-space model for seizure detection in EEG monitoring systems. *IEEE Intell. Syst.* **30**(1), 6–12 (2015)