Convolutional Autoencoder Approach for EEG Compression and Reconstruction in m-Health Systems

Conference Paper · June 2018

DOI: 10.1109/IWCMC.2018.8450511

3 authors:

Abeer Ziad Al-Marridi
Qatar University

Amr Mohamed
Qatar University

Aiman Erbad
Hamad bin Khalifa University

Convolutional Autoencoder Approach for EEG Compression and Reconstruction in m-Health Systems

Abeer Z., Al-Marridi, Amr Mohamed, Aiman Erbad
Department of Computer Science and Engineering
Qatar University
2713, Doha, Qatar
Email: {aa1107191, amrm, aerbad }@qu.edu.qa

***Abstract*— In the last few years, the number of patients with chronic diseases requiring constant monitoring increased rapidly, which motivates researchers to develop scalable remote health applications. Nevertheless, the amount of transmitted real-time data through current dynamic networks with limited and restricted bandwidth, end-to-end delay, and transmission power; limits having an efficient transmission of the data. Motivated by the high energy consumed for transmission, applying data reduction techniques to the vital signs at the transmitter side present an efficient edge-based approach that significantly reduces the transmission energy. However, a new problem arises, which is the ability of receiving the data at the server side with an acceptable distortion rate (i.e. deformation of vital signs because of inefficient data reduction). In this paper, we introduce a Deep Learning (DL) approach based on Convolutional Auto-Encoder (CAE), to compress and reconstruct the vital signs in general and Electroencephalogram Signal (EEG) specifically with minimum distortion. The results show that using CAE provides efficient distortion rate while maximizing compression ratio. However, learning makes CAE application-specific, where each CAE model is designed specifically for a certain application.**

Keywords—Compression; Electroencephalogram Signal; Deep learning; Convolutional Autoencoder; Signal Reconstruction; Data Reduction.

I. INTRODUCTION

Spending on healthcare is rapidly increasing as it is considered a top priority worldwide. The reason behind it is the rising number of patients with different diseases. The United States spends around $2.6 trillion for healthcare, which is considered the highest spending on healthcare, and it is expected to double the expenditure by 2023 [1]. Lately, the number of the patients who need continuous monitoring increased swiftly, and the current model of delivering healthcare services requires physical contact with the caregiver. However, this causes a burden for both the patients and the doctors, as the number of patients is not comparable to the number of physicians. This adds delays in getting the needed services and limits the quality of the one-to-one relationship between the patient and the doctor, posing a challenge for the scalability of the healthcare system. The mobile healthcare system (m-Health) is one key solution to this problem, where mobile devices can be used to support the medical and public health practice. One possible scenario of m-Health systems is to use any communication device (i.e. Patient Data Aggregator, PDA) along with wearable devices in/on or around the patient’s body to aggregate the needed information and send them through the wireless network toward the m-Health center to control and diagnose the situation of the patient as soon as possible. These systems minimize the probability of losing lives of patients if the data was sent at the right time to the caregiver. Delivering such huge data though the network was the main motivation to investigate new reduction and reconstruction approach which maintains the accuracy of the information.

Different data reduction techniques of vital sign data have been well addressed in the literature. A fuzzy data reduction technique was proposed to obtain the most representative electroencephalogram (EEG) samples and neglecting redundant ones while minimizing the loss of knowledge [2]. The authors in [3] modify Block Sparse Bayesian Learning-BSBL-BO method to manage both linear and nonlinear dependency structure of EEG raw signal. However, the technique has high complexity, where the sensing and processing power need to be optimized in order to be implemented into Wireless Body Area Networks (WBAN). A near lossless compression algorithm for EEG signals was introduced in [4] to satisfy low power, latency, and low complexity by combining different tools for data compression and signal processing. The authors in [5] enhance the algorithm proposed by [4] to take into consideration the distortion threshold while looking for the best compression rates and reducing all computational requirements without affecting the compression ratio. In [6] a 1.5-D Multi-Channel EEG Compression Algorithm was proposed to manage the transmission of EEG signals through the network with low complexity and high reliability. The algorithm combines one dimensional DWT to reduce the computational complexity and No List Set Partitioning In Hierarchical Trees (NLSPIHT) algorithm to perform better compression rate. A combination of DWT and Compressive Sensing (CS) in wireless sensors was proposed in [7] as a lossy compression technique to control both compression and energy consumption of encoder and transmitter, having certain energy threshold. However, the authors in [8] believe that CS is not efficient for EEG compression. Moreover, neural networks were used for compression in [8, 9, 10, 11, 12, 13, 14]. A combination of Discrete Cosine Transform (DCT) and Artificial Neural Network (ANN) was used in [9] to develop a near lossless EEG compression approach. The authors first apply DCT and then perform ANN on the DCT coefficients. Arithmetic coding was used to quantize the difference between the original coefficients and the estimated ones to improve the reconstruction performance. In [14], the neural network was used for EEG data compression, where a complete study about the effectiveness of the proposed compression approach with respect to the overall energy consumption was presented. Stacked AutoEncoder (SAE) was used in implementing the compression approach. The authors found that SAE can outperform the state-of-the-art approaches by applying high compression ratios and getting a small distortion rate.

Our work enhances the work done in [14], where deep learning approach using Convolutional AutoEncoder (CAE) is applied instead of SAE to respect the correlations presented in EEG signals. Since arranging the EEG data into 2D formations will take into consideration the Spatio-Temporal correlation amongst the EEG samples. We conducted a comparative study to demonstrate the effectiveness of our approach in terms of compression efficiency.

The rest of the paper is organized as follows: section II presents the background information related to Deep Learning. Section III presents the proposed approach. Experimental results are discussed in section IV. Section V concludes the paper.

II. Background

Deep neural network is not a new research field. In 1986 back-propagation was introduced for the first time by Rumelhart. However, most neural networks used a single layer; due to the cost of computation and availability of data. In July 2006 Hinton and Salakhutdinov introduce the multi-layer neural networks to reduce the dimensionality of the data in order to facilitate the classification, visualization, communication, and storage of high-dimensional data. In 2010, the first transformational work has been published since 2006 by G. Hinton et. al., where they used deep neural networks in acoustic modeling [15]. There are four major architectures of deep networks: Recurrent Neural Networks (RNN), Autoencoder (AE), Convolutional Neural Networks (CNN) and Recursive Neural Networks. Our compression approach is based on a combination of CNN and AE; which is called Convolutional Autoencoder (CAE) neural network [16].

A. Autoencoder

Autoencoder (AE) is a neural network usually used for unsupervised learning as it aims to recreate the input rather than classify it under certain class [17]. The number of neurons in the input layer equals to the number of neurons in the output layer. Unlike other neural networks, the hidden layers have a smaller number of neurons compared to input/output layers; because autoencoder was proposed to encode the data with lower dimensionality and extract the discriminative features.

B. Convolutional Neural Networks

Convolutional Neural Network (CNN) is an efficient artificial neural network, proposed to manage the data which has local correlations while minimizing the number of training parameters. It was called convolutional because it performs complex operations using convolutional filters on the entire image instead of using neurons. Using filters decrease the number of connections between the layers. CNN was able to outperform the state-of-the-art techniques in advanced computer vision and natural language processing tasks [18].

There are three main types of layers used in a convolutional neural network:

- Fully connected layer
- Convolutional layer
- Pooling layer

C. Convolutional Autoencoder

A combination of Convolutional Neural Networks (CNN) and Autoencoder (AE) produce Convolutional AutoEncoder model (CAE), as shown in Fig. 1. This type of models is with images; because of high spatial correlations imbedded in most of the images.

In the proposed compression solution, CAE will be used as an encoder to compress the EEG signal on the sender side, and the decoder will reconstruct the signal at the receiver side. This technique can be implemented in either in the Patient Data Aggregator (PDA)/ sensors to optimize the transmission energy and end-to-end delay or in the server side to optimize the amount of storage required.

The main advantage of using Convolutional Autoencoder model is applying high compression ratios and better reconstructing of the EEG signal based on leveraging EEG spatial correlations. It worth mentioning that the CAE works on specific application characteristics to maximize the compression ratio as much as possible for low distortion rate.

III. PROPOSED SOLUTION

As mentioned in section II, CAE is used to compress and reconstruct the EEG signal. Fig. 2. summarizes the stages of our proposed solution, starting with the preprocessing of the data before being compressed at the transmitter side and reconstruction of the compressed data at the receiver side. Assuming that the encoder part if the CAE will run on the PDA side and the decoder will run on the m-Health server.

A. Preprocessing

Pre-processing steps aim to improve the quality of compressing the EEG signals and minimizing the distortion of the reconstructed signal while applying the maximum possible compression ratio. For this purpose, stages of reshaping and normalizing the data were applied.

1) Reshaping the Data

The Electroencephalogram (EEG) data was represented in a 2D matrix; where each row represents an EEG sample. The ZigZag approach described in [19] was applied to the data as only the even rows are flipped. The authors in [19] believe that ZigZag approach will exploit both spatial and temporal correlations of the EEG and hence enhance the performance of compressing EEG signals.

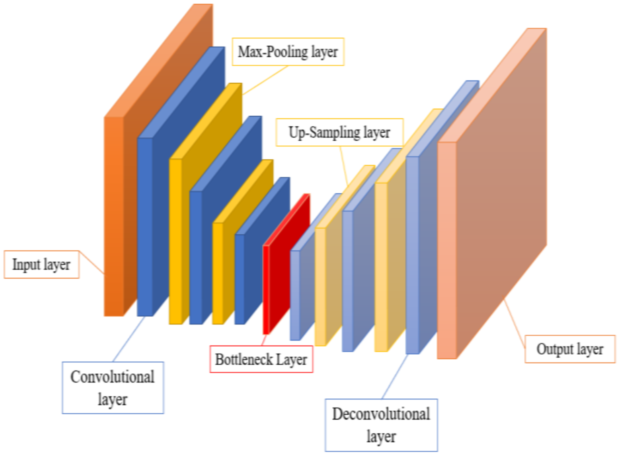

Fig. 1 Representation of Convolutional Autoencoder

2) Normalizing the Data

In machine learning, normalizing the data affect significantly the efficiency during the training and testing. According to [18], normalizing the data to make the values between zero and one, will guarantee a stable convergence of weights and biases. Moreover, in deep neural networks, training is a very important step, and keeping the data without normalizing will make the training step more complicated and slow.

B. Data Compression and Reconstruction

As mentioned earlier we decided to explore the power of deep learning in compressing and reconstructing EEG signals. The main aim is to compress the EEG signal to the maximum while reconstructing the signal with the minimum distortion. The CAE model consists of the input layer, multiple hidden layers, and an output layer.

The output of each layer (feature map) is an input to the next layer, until reaching to the output layer. Convolutional layers were used in the hidden layers as well as in the output layer. Max-pooling layers were used in the encoder of the proposed model to get the maximum value of a certain region of the input, ending up with a matrix of the maximum values of all regions. Up-sampling layers were used in the decoder part of the model as it is considered as the opposite operation of pooling, where it tries to reconstruct and approximate the original input, by repeating the rows and columns of the data. Since convolutional layers will reshape the EEG sample into 2D; the number of filters in each layer change regularly until reaching to the bottleneck layer which gives certain compression ratio.

Equation (1) represents the compression ratio *(CR)*, which is defined as the data reduction in size relative to the uncompressed size of the data. F is the number of filters, $M_i \times N_i$ is the dimensions of the input at the bottleneck layer and S is the size of the original data. Multiplying the number of filters by the input dimensions gives the size of the compressed input.

$$\text{Compression ratio (CR)} = \left|\frac{(\mathrm{F} \times M_i \times N_i) - S}{S}\right| \times 100\% \quad (1)$$

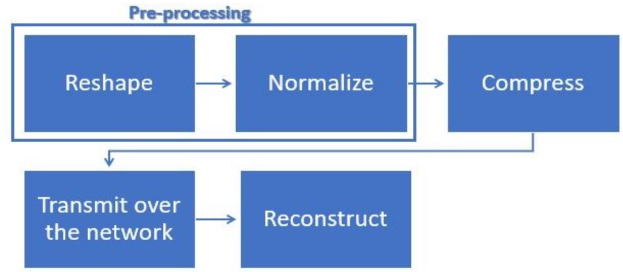

Fig. 2 The workflow of our proposed solution

The number of filters should be an integer, if not then floor function should be applied if the decimal value is less than 5 else ceil function will be applied. The number of the filter can be calculated using (2).

$$Number\ of\ filters\ (F) = \frac{S-(CR \times S)}{M_i \times N_i} \quad (2)$$

It is worth mentioning that the runtime complexity of using CAE approach can be represented by (3), where d is the number of convolutional layers in the neural network, F_i is the number of filters in layer i, K is the kernel size (filter size) and $M_i \times N_i$ is the dimensions of the input for the i^{th} layer. The number of filters in each layer changes based on the applied operation (compression or reconstruction) and the compression ratio.

In our proposed CAE models, the size of the input is always divisible by the kernel size. The size of the input changes regularly, as a result of having max-pooling layers.

$$O\left(\sum_{i=1}^{d} F_i \times \frac{M_i \times N_i}{K^2}\right) \quad (3)$$

Training the model on different compression ratios using CPU, needs weeks. However, it will be much faster if we were using GPUs.

IV. EXPERIMENTAL RESULTS

A. Datasets

We conduct our experimental analysis on two datasets

- BCI-IV-2a from the BCI Competition IV [20]: The EEG signals recorded from 9 subjects and 22 electrodes. The EEG signals were sampled at 250 Hz and bandpass-filtered between 0.5 Hz and 100 Hz. The dataset contains a total of 7548 data samples. It was divided into 6416 for training and 1132 for testing.

- BCI-IV-2b from the BCI Competition IV [21]: The EEG signals recorded from three bipolar recordings: C3, Cz, and C4 with a sampling frequency of 250Hz from 9 participants. Like BCI-IV-2a, the signals were bandpass-filtered between 0.5 Hz and 100 Hz. The dataset contains a total of 5892 data samples. It was divided into 5202 for training and 690 for testing.

B. Environment Setup

Python programming language was used to build the CAE compression model where Keras was used with TensorFlow as a backend to hide all the complexity behind the neural network models and facilitate the process of building deep learning [18].

As mentioned in section III, three main layers were used in our model. The details of the layer's parameters listed below.

- Input layer: receive the input and reshape it in 2D, for BCI-IV-2a dataset the dimensions of the input is 32×16, while 32×24 was used for BCI-IV-2b.
- Convolutional 2D layer: 4×4 kernel size for the filters and Relu activation function were used in all convolutional hidden layers. The number of filters changes based on the amount of compression needed. The number of hidden convolutional layers used in BCI-IV-2a equal to three in both, the encoder and decoder. In BCI-IV-2b, two hidden convolutional layers were used.
- Pooling layer: max-pooling layers were used to apply reduction on the input by 2.
- Up-sampling layer: 2 was used as the up-sampling factor for rows and columns.

The model was compiled and configured for training using Adam optimizer, mean square error loss function and mean absolute error, percentage root distortion rate and accuracy as a compilation metrics. *Fig.3.* shows the encoder and decoder models when applying 20% compression on BCI-IV-2b dataset.

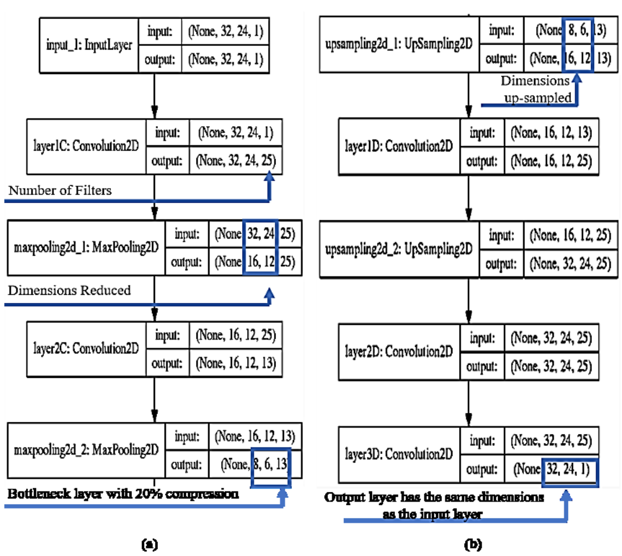

Fig. 3 (a) The encoder part of the model which should be implemented in the edge level of the network (like PDA). (b) The decoder part of the model which should be implemented on the server side of the network (m-Health server).

C. Results and Discussions

The evaluation metrics are the following:

- Compression ratio (*CR*): Each dataset has maximum compression ratio based on the size of the dataset. For example, BCI-IV-2a dataset can reach up to 98.4% compression which means only 1.6% of the data will be transmitted. However, 93% compression is the maximum possible compression ratio achieved on the BCI-IV-2b dataset. The compression ratio was calculated as mentioned in (1).
- Percent-root mean square distortion (*PRD*): O_S corresponds to the original signal, R_S corresponds to the reconstructed signal and N is the total number of EEG samples.

$$PRD\ (\%) = \sqrt{\frac{\sum_{i=1}^{N}(O_S - R_S)^2}{\sum_{i=1}^{N} {O_S}^2}} \times 100\% \tag{4}$$

- Mean Absolute Error (MAE): it was used during the fitting of the model, where it was implemented and defined manually as a new evaluation metric in our model.

$$MAE(\%) = \left|\frac{\sum_{i=1}^{N} O_S - R_S}{N}\right| \times 100\% \tag{5}$$

Fig.4. represents the relation between the compression ratio and distortion rate (*PRD*) using CAE model on the BCI-IV-2b dataset. The model was able to compress up to 93% of the data and reconstruct with a distortion of 12%. However, the designed CAE model for BCI-IV-2a was able to compress 98% of the data with 1.33% distortion; which outperform the state of the arts approaches. A comparison between our proposed CAE model used in BCI-IV-2a dataset with the other state of the art techniques used on the same dataset was done. The values of the distortion (*PRD*) start to increase at 90% compression ratio using CAE, as it was in a steady state before that. Stacked Autoencoder (SAE) proposed in [14] was able to compress up to 90% with a distortion of 33.5%. DWT was able to outperform SAE, SPIHT-3, and SPIHT-6 at low compression ratios as it reaches to 1.5% distortion at 35% compression. However, CAE was able to achieve 0.3% distortion at 35% compression. This shows the effectiveness of using convolutional layers instead of fully connected layers.

We calculated the sample error rate using (6), where N_s is the total number of samples and N_{cs} is the correct samples after reconstruction of the original signal. The behavior represented by *Fig. 5* gives the same trend as *Fig.4*, where the sample error rate increases while increasing the compression with an error rate less than 40% at the maximum compression. Sample error rate gives a good prediction about the visualization results. A visual representation of the original and reconstructed signal using CAE on BCI-IV-2b dataset on 93% compression is shown in *Fig.6.*

$$Sample\ error\ rate(SER) = \frac{N_S - N_{CS}}{N_S} \times 100\% \qquad (6)$$

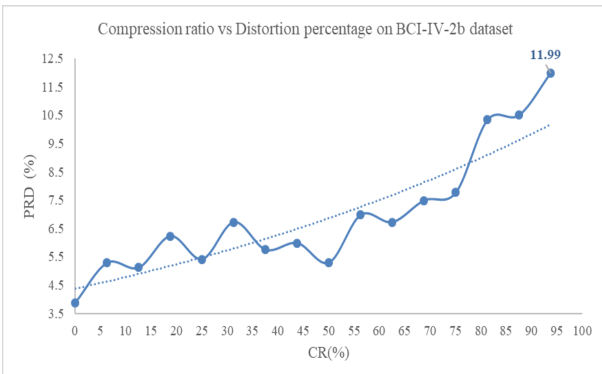

Fig. 4 Relation between Compression ratio and Distortion (BCI-IV-2b dataset)

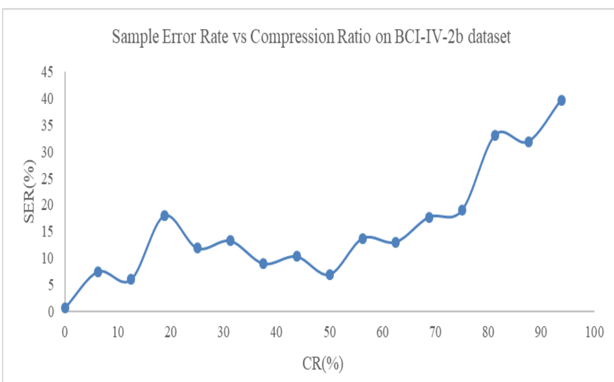

Fig.5 Relation between Compression ratio and Sample Error Rate (BCI-IV-2b dataset)

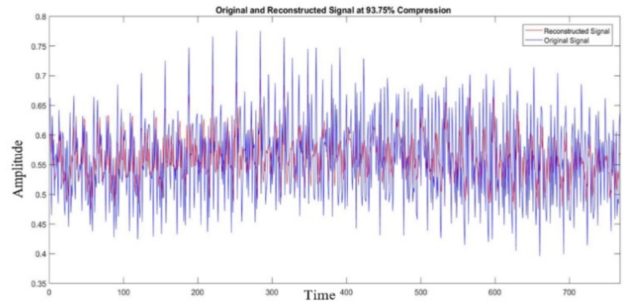

Fig. 6 Original and Reconstructed Signal at high compression (BCI-IV-2b)

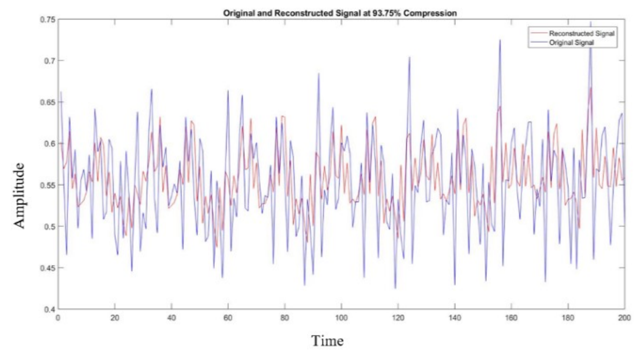

Fig. 7 Original and Reconstructed Signal at high compression (Zoom-in BCI-IV-2b)

V. Conclusion

An adaptive compression technique using convolutional autoencoder model was developed to manage the huge amount of data to be compressed and transmitted through the constrained networks. Reconstructing the compressed data should be within an acceptable distortion range. Deep learning is a very effective approach that can be used in medical signal processing. Our CAE models can be used efficiently to compress EEG signals, with an excellent performance which outperforms the proposed method in [14] (DLDC) and though all other techniques such as DWT, 2D-SPIHT-3-ICs, and 2D-SPIHT-6-ICs. Using BCI-IV-2a dataset, CAE could send 1.6% of the data with a distortion less than 1.5%. Unlike other signal processing approaches, CAE compresses the data based on learning, leading to the fact that each model will be designed to fit the characteristic of a certain application.

Acknowledgment

This work was made possible by NPRP grant # 7 - 684 - 1 - 127 from the Qatar National Research Fund (a member of Qatar Foundation). The findings achieved herein are solely the responsibility of the authors.

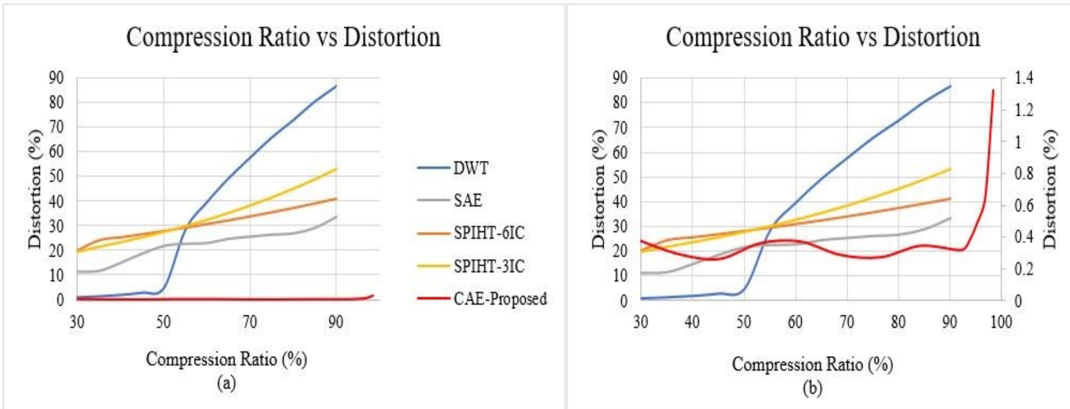

Fig. 8 (a) Relation between Compression ratio and Distortion (PRD) applied to BCI-IV-2a dataset. In (b)another distortion scale was used.

REFERENCES

[1] R. DeVol , A. Bedroussian, A. Charuworn, A. Chatterjee, I. Kyu Kim, S. Kim , and K. Klowden, "An Unhealthy America: The Economic Burden of Chronic Disease -- Charting a New Course to Save Lives and Increase Productivity and Economic Growth," Milken Institute. [Online]. Available: http://www.milkeninstitute.org/publications/view/321.

[2] A. Awad, A. Saad, A. Jaoua, A. Mohamed, and C.-F. Chiasserini, "In-Network Data Reduction Approach Based on Smart Sensing," 2016 IEEE Global Communications Conference (GLOBECOM), 2016.

[3] H. Mahrous and R. Ward, "Block Sparse Compressed Sensing of Electroencephalogram (EEG) Signals by Exploiting Linear and Non-Linear Dependencies," Sensors, vol. 16, no. 2, p. 201, May 2016.

[4] Capurro, I., Lecumberry, F., Martín, Á., Ramírez, I., Rovira, E., & Seroussi, G. (2014, September). Low-complexity, multi-channel, lossless and near-lossless EEG compression. In Signal Processing Conference (EUSIPCO), 2014 Proceedings of the 22nd European (pp. 2040-2044). IEEE.

[5] G. Dufort, F. Favaro, F. Lecumberry, A. Martin, J. P. Oliver, J. Oreggioni, I. Ramirez, G. Seroussi, and L. Steinfeld, "Wearable EEG via lossless compression," 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 2016.

[6] G. Xu, J. Han, Y. Zou, and X. Zeng, "A 1.5-D Multi-Channel EEG Compression Algorithm Based on NLSPIHT," IEEE Signal Processing Letters, vol. 22, no. 8, pp. 1118–1122, 2015.

[7] R. Hussein, A. Mohamed, and M. Alghoniemy, "Scalable real-time energy-efficient EEG compression scheme for wireless body area sensor network," Biomedical Signal Processing and Control, vol. 19, pp. 122–129, 2015.

[8] A. B. Said, A. Mohamed, T. Elfouly, K. Harras, and Z. J. Wang, "Multimodal Deep Learning Approach for Joint EEG-EMG Data Compression and Classification," 2017 IEEE Wireless Communications and Networking Conference (WCNC), 2017.

[9] B. Hejrati, A. Fathi, and F. Abdali-Mohammadi, "A new near-lossless EEG compression method using ANN-based reconstruction technique," Computers in Biology and Medicine, vol. 87, pp. 87–94, 2017.

[10] N. Sriraam, "Context-based near-lossless compression of EEG signals using neural network predictors," AEU - International Journal of Electronics and Communications, vol. 63, no. 4, pp. 311–320, 2009.

[11] N. Sriraam, "Correlation dimension based lossless compression of EEG signals," Biomedical Signal Processing and Control, vol. 7, no. 4, pp. 379–388, 2012.

[12] N. Sriraam, "A High-Performance Lossless Compression Scheme for EEG Signals Using Wavelet Transform and Neural Network Predictors," International Journal of Telemedicine and Applications, vol. 2012, pp. 1–8, 2012.

[13] N. Sriraam and C. Eswaran, "Performance Evaluation of Neural Network and Linear Predictors for Near-Lossless Compression of EEG Signals," IEEE Transactions on Information Technology in Biomedicine, vol. 12, no. 1, pp. 87–93, 2008.

[14] A. B. Said, A. Mohamed, and T. Elfouly, "Deep learning approach for EEG compression in mHealth system," 2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC), 2017.

[15] "How Deep Learning Will Change Our World," Circuit Insight. [Online]. Available: http://www.circuitinsight.com/programs/54708.html.

[16] J. Patterson and A. Gibson, Deep learning: a practitioners approach.

[17] Baldi, Pierre. "Autoencoders, unsupervised learning, and deep architectures." Proceedings of ICML Workshop on Unsupervised and Transfer Learning. 2012.

[18] J. Brownlee, Deep learning with Python: develop deep learning models on Theano and TensorFlow using Keras. Place of publication not identified: Machine Learning Mastery, 2016

[19] K. Srinivasan, J. Dauwels, and M. R. Reddy, "A two-dimensional approach for lossless EEG compression," Biomedical Signal Processing and Control, vol. 6, no. 4, pp. 387–394, 2011.

[20] Brunner, C., Leeb, R., Müller-Putz, G., Schlögl, A., & Pfurtscheller, G. (2008). BCI Competition 2008–Graz data set A. Institute for Knowledge Discovery (Laboratory of Brain-Computer Interfaces), Graz University of Technology, 136-142.

[21] Leeb, R., Brunner, C., Müller-Putz, G., Schlögl, A., & Pfurtscheller, G. (2008). BCI Competition 2008–Graz data set B. Graz University of Technology, Austria