Two Different Approaches of Feature Extraction for Classifying the EEG Signals

Pari Jahankhani, Juan A. Lara, Aurora Pérez, Juan P. Valente

Two different approaches of feature extraction for classifying the EEG Signals

Pari Jahankhani[1], Juan A. Lara[2], Aurora Pérez[2], Juan P. Valente[2]

[1]University of East London, School of Computing, Information Technology and Electronic, London E16 2RD, United Kingdom, pari@uel.ac.uk

[2]Technical University of Madrid, School of Computer Science, Campus de Montegancedo,28660, Boadilla del Monte, Madrid, Spain j.lara.torralbo@upm.es,{aurora, jpvalente}@fi.upm.es

Abstract. The electroencephalograph (EEG) signal is one of the most widely used signals in the biomedicine field due to its rich information about human tasks. This research study describes a new approach based on i) build reference models from a set of time series, based on the analysis of the events that they contain, is suitable for domains where the relevant information is concentrated in specific regions of the time series, known as events. In order to deal with events, each event is characterized by a set of attributes. ii) Discrete wavelet transform to the EEG data in order to extract temporal information in the form of changes in the frequency domain over time- that is they are able to extract non-stationary signals embedded in the noisy background of the human brain.

The performance of the model was evaluated in terms of training performance and classification accuracies and the results confirmed that the proposed scheme has potential in classifying the EEG signals.

Keywords: Wavelet, Feature extraction, EEG, Classifying

1   Introduction

The electroencephalograph (EEG) signal is one of the most widely signal used in the biomedicine field due to its rich information about human tasks.. In practical applications of pattern recognition, there are often diverse features extracted from raw data which needs recognising. Time series modelling has many applications like, for example, feature identification across a group of time series, or model comparison measuring the likeness among groups of time series, or the evolution of one and the same group over time. In actual fact, in many domains, like medicine, the mere observation of the model by the expert can turn out to be very useful in the decision-making process.

The relation of EEG signals to the human movements and behaviour has been extensively studied in past decades [1].

A key data mining problem is the construction of feature models from set of time series.

In the field of time series data mining, there are well-established methods for comparing two time series, finding subsequence that are repeated several times throughout the same time series and techniques that try to determine whether a time series contains a particular sequence. Also there are techniques that try to generate a representative reference model from a set of time series [2], [3], [4], [5].

A powerful method was proposed in the late 1980s to perform time-scale analysis of signals: the wavelet transforms (WT). This method provides a unified framework for different techniques that have been developed for various applications.

Nevertheless, in many cases only particular regions of the series contain relevant knowledge and the data mining techniques should focus on these regions (known as events) [6]. This applies to domains like seismography, the stock market or medicine. In seismography, for example, the only moments of interest are when the time series indicates an earthquake, volcanic activity leading up to the quake, or replications. The lengthy periods between these events provide hardly any information. Figure 1 shows an example of an EEG time series, highlighting an event corresponding to the electrical activity generated by the nervous system in response to a stimulus.
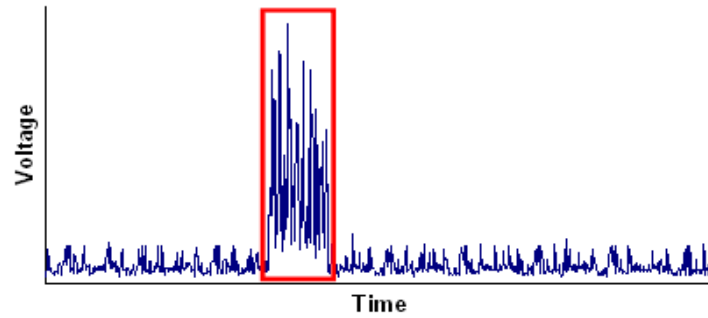
Fig. 1. Example of an EEG time series.

Detecting patterns in EEG produced from the normal mental states has some problems, due to the fact that EEG signals which are recorded by surface electrodes can contain noise as a result of electrical interference and movement of the electrodes on the scalp or EEG can be corrupted by eye blinks and other muscular activity that produce signals of greater magnitude than produced by cortical activity. In this work, two different methods are applied for feature extraction and classification.

1) Build reference models from a set of time series, based on the analysis of the events that they contain, is suitable for domains where the relevant information is concentrated in specific regions of the time series, known as events. The method enables to define regions of interest according to the knowledge extracted from the domain experts, which is a plus compared with other methods addressing the time series as a whole without taking into account that certain regions can be irrelevant in the domain in question.

2) Discrete Wavelet Transform (DWT) has been applied for the time–frequency analysis of EEG signals and an Adaptive Fuzzy Inference Neural Network System (AFINN) [7], [8], [9] scheme for the classification using wavelet coefficients.

2 Method 1: Feature extraction of events

Electroencephalographic devices generate time series that record scalp electrical activity (voltage) generated by brain structures. EEG signals contain a series of waves characterised by their frequency and amplitude. In EEG time series it is possible to find certain types of special waves that are characteristic of some neurological pathologies, like epilepsy. Those waves are known as paroxysmal abnormalities and can be considered as events.

During this research we have taken into account three kinds of events:

*   Spike Wave: It is a wave whose amplitude is relatively higher than the rest of waves in the signal. It has a period of between 20 and 70 millisecond.
*   Sharp Wave: It is a wave whose amplitude is relatively higher than the rest of waves in the signal. It has a period of between 70 and 200 millisecond. Figure 2 shows an example of a sharp wave event.
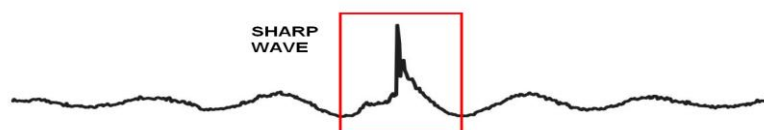*   Spicule: It is a sharp wave with an abrupt change of polarity.

Fig. 2. Sharp wave event.

The features characterising these events are as follows:

- Duration of the wave.
- Amplitude of the wave.

To identify the EEG events and determine their features, the proposed method calculates the point where the polarity of the signal changes as shown in figure 3. The method identifies points where there is a local maximum or minimum whose distance to the polarity change value is higher than a certain threshold (∂). That distance is the amplitude of the event. The duration of the wave is then calculated by analysing the two intersections between the time series and the polarity change value line. Depending on the duration, the event is classified as a spike or a sharp wave, according to the experts' criteria. Finally, those sharp waves that have an abrupt change of polarity are classified as spicules.
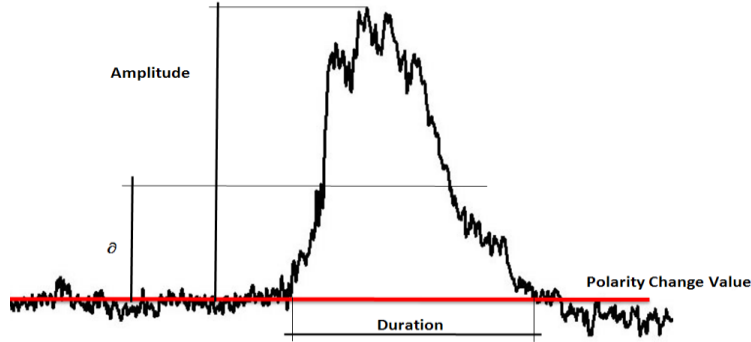

Fig. 3. Event taken from an EEG time series.

2.1 Model Generation Method

The model generation method presented here is suited for domains where important information is only confined to certain regions while the remaining of the time series hardly provides any information.
In order to deal with events, each event is characterized by a set of attributes.

The model generation method receives a set of time series $S = \{S_1, S_2, ..., S_n\}$, each containing a particular number of events, and generates a reference model M that represents this set of time series. The model M is built on the basis of the most characteristic events. The most characteristic events of S are those events that appear in the highest number of timer series of S.

To find out whether a particular event in a time series S_i also appears in another time series S_j ($j \neq i$), the event has to be characterized with an attribute vector and compared with the other events of the other series. To speed up this process, all the events present in the time series are clustered, so similar events belong to the same cluster. On the one hand, the clustering process is useful to know the different groups of events. On the other hand, it facilitates the extraction of the most characteristic events. Once we have a set of clusters, the objective is to find those clusters containing events that appear in the highest number of time series, that is, characteristic events. Having located those groups with similar events, an exhaustive cluster analysis is run in order to extract the event representative of each of these groups. This will be described later (steps 5 to 9 of the algorithm). These extracted representative events are the characteristic events of S and will be part of the final model.

Let $S = \{S_1, S_2, ..., S_n\}$ be a set of n time series and m the typical number of events that appear in the time series of S. The algorithm for generating a reference model M representing the set S is as detailed below (with the purpose of making the algorithm more legible key decisions are justified at the end of the algorithm):

1. **Initialize the model.**
   M = Ø.
2. **Identify events.**
   Extract all the events E_v from the series of S and use an attribute vector to characterize each event. This vector covers what the expert considers to be the key features for each type of

domain event. This step is domain dependent, as the event characterization will depend on the time series type. To extract the events, the time series is examined in search of regions that meet the conditions identifying each event type defined according to the knowledge extracted from the expert.

3. **Determine the typical number of events m.**
   m is the typical number of events in each time series of S. At the end of the algorithm it will be discussed how to determine this value.

4. **Cluster events.**
   Cluster all the events extracted in step 2. Bottom-up hierarchical clustering techniques have been used. Taking into account that the proposal described here should be a general-purpose method and there is no a priori information for specifying the optimum number of clusters in each domain, bottom-up hierarchical clustering is a good option, as it is not necessary to specify the number of clusters k beforehand. We have used Hierarchical Clustering

Repeat steps 5 to 9 m times

5. **Get the most significant cluster C_k.**
   Determine which cluster C_k of all the clusters output in step 4 is the most significant. Cluster significance is measured using Equation (1).

$$\text{SIGNF}(C_k) = \frac{\#\text{TS}(C_k)}{n} \tag{1}$$

   That is, cluster significance is given by the number of time series that have events in that cluster over the total number of time series n. Events that have already been examined (step 8 and 9) are not taken into account to calculate the numerator.

6. **Extract the event E_c that best represents the cluster.**
   Extract the event that is most representative of the cluster C_k, that is, the event E_c that minimizes the distance to the other events in the cluster. Let S_j be the time series in which the event E_c was found.

7. **Add the event E_c to the model.**
   $M = M \cup E_c$.

8. **Mark event E_c as examined.**

9. **Mark the most similar events to E_c as examined.**
   From the cluster C_k obtain, for each time series $S_i \neq S_j$, the event E_p from S_i that is the most similar to the representative event (E_c) output in step 6. Each E_p will be represented in the model by the event E_c and therefore these E_p events will also be discarded in order not to be considered in later iterations.

10. **Return M as a model of the set S.**

The most significant clusters, that is, those clusters that contain events present in the highest number of time series were analysed to output the events that are part of the model. To do this, the process of identifying the most significant cluster is repeated m times, outputting a representative and marking as examined both this representative and similar events in each time series. With regard to the algorithm, note that:
   a) The identification of events is domain dependent because the criteria to define events in each domain are required. The rest of the algorithm is domain independent and it can be applied to any domain without any change. Figure 4 shows the overall structure of the proposed method that receives a set of time series S and generates a model M that represents it.
   b) After the representative event of the most significant cluster has been output, it should not be taken into account again for the next iteration, and it is marked as an already examined event.
   c) A cluster may contain not just one but several events from each time series. For this reason, even if a cluster is selected as the most significant, the cluster in question is not omitted in later

iterations. The events already processed are marked as examined and will not be taken into account in future iterations.
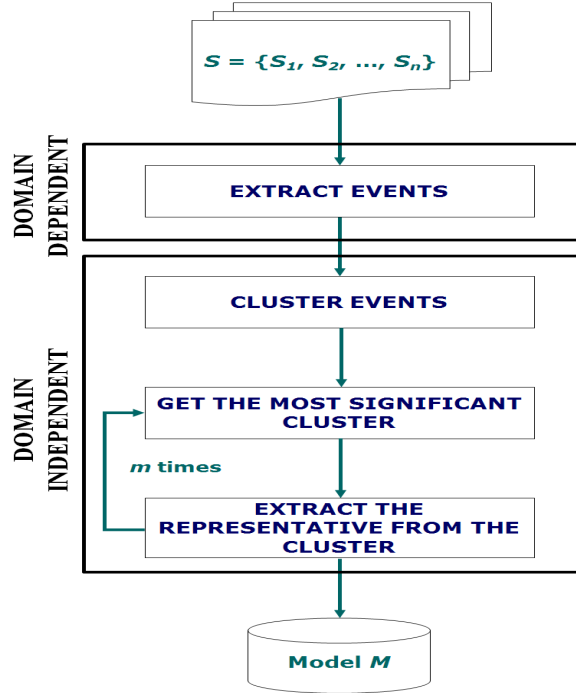

Fig. 4. Overall structure of the proposed method.

Another important issue is the number of events making up the model. In this case, we have chosen the mode (*m*) of the number of events of the time series of *S*. This decision is based on the fact that if the original time series have a typical number of events *m*, it makes sense for the model that represents them to have the same number of events *m*. The typical number of events in the time series of *S* may not be unimodally distributed. This could happen especially if there are not many time series in the set *S*. For non-unimodal distributions, we have opted to take the integer value closest to the mean of the number of events.

A last point to be considered is the distance between events that has been used in the algorithm for clustering, representative event selection and discarding similar events. The *city block* distance is used. Given two vectors, the city block distance calculates the sum of the absolute value of the difference of each of the coordinates of the above vectors:

$$d(x, y) = \sum_{i=1}^{p} |x_i - y_i| \qquad (2)$$

In Equation (2), *x* and *y* are the vectors (that is, the event descriptors) for comparison and *p* is the number of coordinates (dimension). Other distance measures have been studied, but the city block distance was finally chosen. The main reason for this choice is that the clustering algorithm uses the mean distance per attribute as the threshold for determining whether or not two elements are similar enough to belong to the same cluster. This mean distance per attribute is obtained simply by dividing the total city block distance $d(x,y)$ by the number of attributes *p*. The use of the city block distance then saves time as it obviates additional transformations that would make the clustering process more complex to develop and more computationally intensive.

Figure 5 shows an example of the application of the proposed method to a set $S = \{S_1, S_2, S_3, S_4\}$ of 4 time series (*n*=4). In this case, S_1 contains 2 events (E_{11} and E_{12}), S_2 contains 2 events (E_{21} and E_{22}), S_3 contains 3 events (E_{31}, E_{32} and E_{33}) and finally S_4 contains 2 events (E_{41} and E_{42}). Therefore, the typical number of events is 2 (*m*=2). Once the events are extracted, they are clustered into three different clusters (C_1, C_2 and C_3). Then, the most significant cluster is obtained. To do that, it is necessary to calculate the significance of each cluster according to Equation (EQ). In this case, cluster C_1 have events present in 3 out of the 4 time series, cluster C_2 have events that appear in 1 out of the 4 time series and cluster C_3 have events present in 4 out of the 4 time series of *S*. Then, the significance of C_1 is $SIGNF(C_1) = \dfrac{3}{4} = 0.75$, the

significance of C_2 is $SIGNF(C_2) = \dfrac{1}{4} = 0.25$ and the significance of C_3 is $SIGNF(C_3) = \dfrac{4}{4} = 1$. Therefore, the most significant cluster is C_3. In the next step, the event E_{12} is extracted as the representative event of the cluster C_3 because E_{12} is the event in C_3 that minimizes the distance to the other events in that cluster. Thus, the event E_{12} is a characteristic event of S and will be part of the final model M. This process has to be repeated twice (because $m=2$) to build the final model that consists of the events E_{12} and E_{32}.
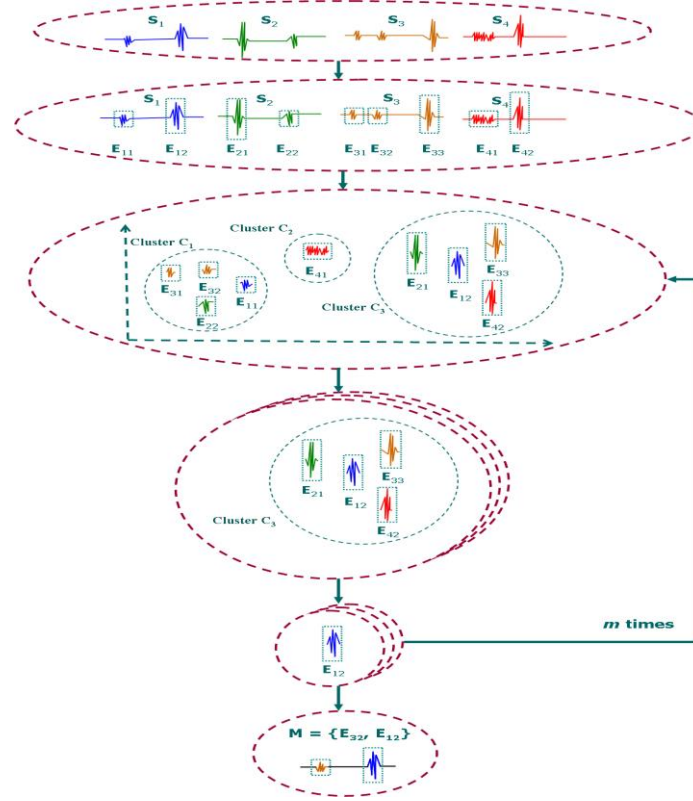
Fig. 5. Example of the application of the proposed method

3 Method 2: Feature Extraction using wavelet Transform

The extracted wavelet coefficients provide a compact representation that shows the energy distribution of the EEG signal in time and frequency.
In order to reduce the dimensionality of the extracted feature vectors, statistics over the set of wavelet coefficients were used. The following statistical features were used to represent the time-frequency distribution of the EEG signals:

1. Maximum of the wavelet coefficients in each sub-band
2. Minimum of the wavelet coefficients in each sub-band.
3. Mean of the wavelet coefficients in each sub-band.
4. Standard deviation of the wavelet coefficients in each sub-band.

Extracted features for the EEG recorded class A and E shown in table 1. For each of these sub-bands, we extracted four measures of dispersion, yielding a total of 20 attributes for sample window.
The complete data set consists of two sets denoted by A and E each containing 100 signal-channel EEG segments. The training and test sets of the AFINN classifier were formed by 3200 vectors (1600 vectors from each class). The extracted wavelet coefficients provided a compact representation that shows the energy distribution of the EEG signal in time and frequency. For each of wavelet sub-bands, we have

extracted four measures of dispersion, yielding a total of 20 attributes per sample window. Those extracted features for two recorded classes A and E are shown in Table 1.

Table 1: The extracted features of two windows from A & E classes

Data	Extracted Feature	Sub-band D1	Sub-band D2	Sub-band D3	Sub-band D4	Approximation
Set A	Max	28.1094	101.757	131.0846	124.377	114.138
	Min	-28.4010	-60.813	-149.072	-158.797	-109.521
	Mean	-0.0022	0.0058	-0.0035	0.0388	3.7950
	Std. Dev.	5.1818	13.6442	23.3685	24.7933	35.1465
Set E	Max	123.3921	278.924	429.6621	375.0564	582.3167
	Min	-90.7055	-23851	-417.120	-468.064	-361.2154
	Mean	0.0131	-0.0281	-0.0359	-0.0071	-5.5526
	Std. Dev.	11.8488	35.9941	73.7659	78.1432	180.4493

4   Evaluation/ Conclusion

A system implementing the described method has been developed. The system has been evaluated by running a battery of experiments using a 10-fold cross validation approach. These experiments were done on time series generated by electroencephalographic devices.

During this research, we have used publicly available datasets described in [10]. The complete data set consists of five sets (denoted *A-E*) each containing 100 single-channel (100 electrodes) EEG recordings of 5 separate patient classes. For this study, we focused on sets labelled *A* (healthy patients) and *E* (epileptic seizure session recordings).

The ultimate aim of the evaluation is to measure how good the model generation method is. For the evaluation of the proposed method, two models were created for each class ($M_{healthy}$ and $M_{epileptic}$).
The first model ($M_{healthy}$) was created from a training set composed of 90 of the 100 healthy patients (set *A*). The other 10 patients constituted the test set. The second model ($M_{epileptic}$) was generated from a training set composed of 90 of the 100 epileptic patients (set *E*). The other 10 patients were used as test set. The patients in the test set were chosen at random.

Once the models have been created, they have been evaluated by checking whether the $M_{healthy}$ model properly represents the group of healthy patients and whether the $M_{epileptic}$ model is representative of the group of epileptic patients. To do that, we have classified the 20 individuals in the test group according to their similarity to the two created models (that similarity was determined using the time series comparison method proposed in [11]). This process was repeated ten times changing the training set and the test set.

The training data set was used to train the AFINN model for classification of the two classes of EEG signals. The proposed system was trained and tested with the extracted features using discrete wavelet transform of the EEG signals. The simulation results reveal a perfect performance compared to a classic MLP neural network**.**

The results of the proposed classifier, using 2 different training sets are shown in Table 2.

Table 2: Comparison of the three methods

Class	Reference Model	AFINN	MLP
A	92%	98.12%	94.98%
E	96%	97.96%	95.86%

5 References

1. Zoubir, M., Bosshssh,B: Seizure detection of newborn EEG using a model approach, IEEE Transactions on Biomedical Engineering 45, 673-685 (1998)
2. Philip K. Chan and Matthew V. Mahoney, 'Modeling multiple time series for anomaly detection', in ICDM '05: Proceedings of the Fifth IEEE International Conference on Data Mining, PP. 90–97, Washington, DC, USA, (2005). IEEE Computer Society.
3. Juan P. Caraça-Valente and Ignacio López-Chavarrías, 'Discovering similar patterns in time series', in KDD '00: Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 497–505, New York, NY, USA, (2000). ACM.
4. Zhuo Chen, Bing ru Yang, Fa guo Zhou, Lin na Li, and Yun feng Zhao, 'A new model for multiple time series based on data mining', Knowledge acquisition and Modeling, International Symposium on, 0, 39–43, (2008).
5. Spiros Papadimitriou, Jimeng Sun, and Christos Faloutsos, 'Streaming pattern discovery in multiple time-series', in VLDB '05: Proceedings of the 31st international conference on Very large data bases, pp. 697–708. VLDB Endowment, (2005).
6. Richard J. Povinelli, 'Using genetic algorithms to find temporal patterns indicative of time series events', in in GECCO 2000 Workshop: Data Mining with Evolutionary Algorithms, pp. 80–84, (2000).
7. R. G. Andrzejak, K. Lehnertz, F. Mormann, C. Rieke, P. David, and C. E. Elger, 'Indications of nonlinear deterministic and finitedimensional structures in time series of brain electrical activity: dependence on recording region and brain state.', Physical review. E, Statistical, nonlinear, and soft matter physics, 64, (December 2001).
8. Jahankhani P., Kenneth Revett, Vassils Kodogiannis, "Classification Using Adaptive Fuzzy Inference Neural Network" Proceedings of the Twelfth IASTED International Conference Artificial Intelligence and Soft Computing (ASC 2008), September 1-3, 2008 Palma de Mallorca, Spain, ISBN 978-0-88986-756-7.
9. Jahankhani P., Kenneth Revett, Vassilis Kodogiannis , "Data Mining an EEG Dataset With an Emphasis on Dimensionality Reduction" IEEE Symposium on Computational Intelligence and Data Mining (CIDM) April 1-5, 2007.
10. Jahankhani P., Kenneth Revett, Vassilis Kodogiannis, "EEG Signal ClassificationUsing Wavelet Feature Extraction and Neural Networks" , Sofia, Bulgaria, October 3-6, 2006,pp 120-125 IEEE John Vincent Atanasoff 2006 International Symposium on Modern Computing.
11. J.A. Lara, G. Moreno, A. Perez, J.P. Valente, and A. López-Illescas, 'Comparing posturographic time series through events detection', in Computer-Based Medical Systems, 2008. CBMS '08. 21st IEEE International Symposium on, pp. 293–295, (June 2008).