# A Semi-automatic Framework for Mining ERP Patterns

Jiawei Rong, Dejing Dou
Computer and Information Science
University of Oregon, USA
{jrong, dou}@cs.uoregon.edu

Gwen Frishkoff
Learning Research and Development Center
University of Pittsburgh, USA
gwenf@pitt.edu

Robert Frank, Allen Malony
NeuroInformatics Center
University of Oregon, USA
rmfrank@mac.com
malony@cs.uoregon.edu

Don Tucker
NeuroInformatics Center
University of Oregon, USA
Electrical Geodesics, Inc.
dtucker@egi.com

## Abstract

*Event-related potentials (ERP) are brain electrophysiological patterns created by averaging electroencephalographic (EEG) data, time-locking to events of interest (e.g., stimulus or response onset). In this paper, we propose a semi-automatic framework for mining ERP data, which includes the following steps: PCA decomposition, extraction of summary metrics, unsupervised learning (clustering) of patterns, and supervised learning, i.e. discovery, of classification rules. Results show good correspondence between rules that emerge from decision tree classifiers and rules that were independently derived by domain experts. In addition, data mining results suggested ways in which expert-defined rules might be refined to improve pattern representation and classification results.*

## 1. Introduction

Research in cognitive and clinical neuroscience has given rise to a wealth of data over the past several decades. It is becoming increasingly clear that management and distribution of these data will require advanced tools for data representation, mining, and integration. In this paper, we propose a semi-automatic framework that is designed to classify ERP patterns related to visual word comprehension. The results of this process will function as inputs to ERP database ontologies, to support future work on mining and classification of higher-order patterns, cross-laboratory collaboration, and integration of study results [10].

Electroencephalography (EEG) is a widespread, noninvasive method for imaging brain activity. EEG data are acquired by placing sensors on the head to measure electrical signals that are generated in cortex and conducted to the scalp surface. Compared with other imaging techniques, such as Positron Emission Tomography (PET) and functional Magnetic Resonance Imaging (fMRI), EEG methods have two advantages: first, they provide a direct measure of neuronal activity (PET and fMRI measure the hemodynamic response, which is closely linked with neuronal activity); and second, they have excellent temporal resolution - on the order of milliseconds, compared with 6 seconds or more for hemodynamic measures. Given that most sensory-motor and cognitive processing takes place within a few hundred milliseconds, fine-grained representation of the time course of brain activity is extremely important. In addition, with the advent of dense-array methodologies, modern EEG methods are now characterized by high spatial, as well as high temporal, dimensionality.

Event-related potentials (ERPs) are derived by averaging across segments of EEG data, time-locking to events of interest (e.g., stimulus onset). Signals that are not event-related tend towards zero as the number of averaged trials increase. In this way, ERP methods increase the signal-to-noise ratio (SNR) and provide measures of brain electrical activity that are tightly linked to stimulus processing (e.g., Figure 1(A)).

While ERP methods have led to many important findings over the last several decades, this research area faces some current challenges that call for advanced computational solutions. One current challenge is that of establishing robust methods for pattern classification. At each time point, many parts of the brain may be simultaneously active, contributing overlapping (or “superposed”) patterns to the measured signal. The goal of ERP research is to separate and classify these patterns (or “components”) and to relate them to specific brain and cognitive functions. Distinct patterns are

21st International Conference on
Advanced Information Networking and Applications Workshops (AINAW'07)
0-7695-2847-3/07 $20.00 © 2007 IEEE

characterized by their time course (e.g., early or late), polarity (positive or negative), and scalp distribution, or topography. For example, as illustrated in Figure 1, the "P100 component," which was extracted from the superposed data (A) using Principal Components Analysis, has a peak latency of approximately 100ms (B) and is positive over occipital areas of the scalp (C).

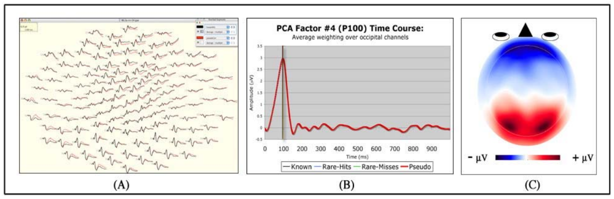

**Figure 1.** (A)128-channel EEG waveplot; positive voltage plotted up. Black, response to words; Red, response to nonwords. (B) Time course of P100 factor for same dataset, extracted using Principal Components Analysis. (C) Topography of P100 factor.

Although ERP researchers have reached some general agreement on how to define ERP components, in reality, such patterns can be difficult to identify, and definitions vary across research labs. Furthermore, methods for ERP data summary and analysis differ widely across research sites. This variability can make it hard to compare results across experiments and across laboratories, limiting the generalizability of research results in this important domain. To address these issues, we have proposed a new framework, called "Neural ElectroMagnetic Ontologies," or NEMO [10]. The NEMO project proposes to develop database ontologies to support ERP data representation and integration. These databases will be designed to allow researchers to manage large amounts of complex data and to search these data using consistent definitions. Robust ERP pattern definitions are an important part of ERP ontology development.

The rest of the paper is organized as follows. We give a brief overview of ERP research methods. We then describe some applications of our framework and report results from mining of ERP patterns, including data preprocessing with temporal PCA, clustering and cluster-based classification. We conclude by outlining future directions for ERP pattern classification and ontology development efforts.

## 2. Tools for EEG and ERP Preprocessing

ERP data consist of time series, representing temporal fluctuations in the EEG that are time-locked to events of interest (e.g., word or picture stimuli). In dense-array EEG and ERP research, these time series are measured across multiple locations on the scalp surface.

A variety of tools are available for ERP preprocessing and pattern analysis. For example, Net Station [4] is a suite of tools, which includes data cleaning, statistical extraction and visualization techniques. EEGLAB [2] is a Matlab toolbox that provides advanced statistical methods for EEG/MEG and ERP processing, including independent component analysis (ICA) and joint time-frequency analysis (TFA). The Dien PCA Toolbox [1] includes Principal Component Analysis (PCA) tools that are optimized for ERP data decomposition.

## 3. Our framework for mining ERP patterns

Our framework (Figure 2) includes data preprocessing using the tools described in Section 2, clustering and cluster-based classification rule mining. Data preprocessing can be further split in to two parts, i.e., temporal PCA decomposition and extraction of summary metrics. Summary metrics capture spatial, temporal, and functional dimensions of the data. The resulting statistics metrics then are used as input for the clustering process. A decision tree learner is then trained based on the clustering results to derive rules for ERP pattern classification.

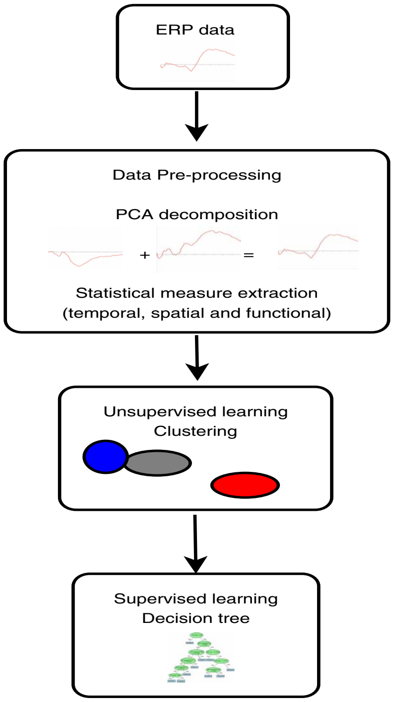

**Figure 2.** Semi-automatic framework for mining ERP patterns.

### 3.1. Data preprocessing with temporal PCA decomposition

In the present paper, we analyzed data collected in two studies of word/nonword processing. Data were acquired using a 128-channel EEG sensor net [3]. Sampling rate was 250hz. The EEG were segmented into 1,500ms epochs, beginning 500ms before stimulus onset (total number of samples = 375).

Together the datasets comprise 89 subjects and 6 experimental conditions (#observations = 534). A description of the experiment paradigm, behavioral measures, scalp ERPs, and cortical (source) waveforms can be found in [13]. For cross-validation of our pattern classification and labeling procedures, subjects were randomly assigned to one of two groups, resulting in 20-24 subjects per subgroup. Subgroups were matched in proportion of males to females and in mean age and handedness.

### 3.2. Temporal PCA decomposition

ERP data represent a mixture of "signal" (functional brain patterns) and "noise" (extracerebral artifacts and brain activity that is not related to the events of interest). Data decomposition methods can help separate signal from noise and disentangle overlapping patterns. A variety of statistical decomposition methods have been applied to ERP data in the past few decades, such as Independent Component Analysis (ICA), wavelets and Principal Component Analysis (PCA). In this paper, PCA [9] is used to decompose the ERP data. PCA belongs to a family of dimension reduction procedures. It projects the data into a new space of lower dimension. In the present study, we used temporal PCA, as implemented in the Dien PCA toolbox [1]. The dataset used as input to the PCA is organized with the variables corresponding to time points. The number of variables is equal to the number of samples (N=375 in the present case). The waveforms vary across subjects (N=89), channels (N=128) and experimental conditions (N=6). PCA extracts as many factors as there are variables (total N=375). In this experiment, we retained the first 15 PCA factors, accounting for most of the variance ($> 75\%$). The remaining factors are assumed to contain "noise"; this assumption is verified by visual inspection of the time course and topographic projection of each factor.

## 4. Summary metrics extraction

For each PCA factor, we extracted summary metrics representing spatial, temporal and functional dimensions of the ERP patterns of interest (Table 1). After preprocessing, the data consist of vectors containing 25 spatial, temporal and functional attributes derived from the automated measure generation. Thus, the data represent the individual PCA factors of each subject and condition as points in a 25 dimensional attribute space. After clustering, the data in each cluster were compared with labeling datasets that were generated with the rules defined by domain experts to determine the distribution of the pre-defined ERP patterns amongst the clusters. In this report we focus on results for four patterns that were identified by domain experts: the P100 (an occipital positivity, peaking at 100ms), N100 (an occipital negativity, peaking at 180ms), N2 (a left temporal pattern, peaking 250ms), and P300 (a parietal positivity from 300 to 700ms).

| Attribute | Description |
|---|---|
| IN-min | min amplitude |
| IN-max | max amplitude |
| IN-mean | mean amplitude for a specified channel set |
| ROI | region of interest |
| SP-cor | cross-correlation between Factor(FA)<br>topography and topography of target pattern |
| SP-max | channel with max weighting for factor FA |
| SP-max (ROI) | channel grouping(ROI) to which<br>the max channel belongs |
| SP-min | channel with min weighting for factor FA |
| SP-min(ROI) | channel grouping(ROI) to which<br>the min channel belongs |
| TI-max | max latency(time of max amplitude) |
| EVENT | event type (stimon, respon, EKG-R, etc.) |
| STIM | stimulus |
| MOD | modality of stimulus |

**Table 1.** Intensity, spatial, temporal and functional metrics

## 5. Unsupervised learning: Clustering

Traditionally, ERP patterns are identified through visual inspection of grand-averaged ERP data. However, the precise definition of a target pattern, its operationalization, and measurement across individual subjects, can vary considerably across research groups. In our framework, we use unsupervised learning technology, i.e., Expectation Maximization (EM) clustering, to automatically separate ERP pattern, as they are distributed across "latent" (PCA) factors. The factors extracted through PCA are weighted across individual subjects and experiment conditions. Summary metrics extracted from each observation (subject, condition) are then input to EM clustering. Observations that belong to the same pattern are expected to map to the same cluster using this method. The larger aim is to develop an automatic pattern classification method, which can support robust ERP pattern definitions.

21st International Conference on
Advanced Information Networking and Applications Workshops (AINAW'07)
0-7695-2847-3/07 $20.00 © 2007 IEEE

| **Cluster/Pattern** | **0** | **1** | **2** |
|---|---|---|---|
| **P100** | 0 | 0 | **99** |
| **N100** | 46 | 47 | 0 |
| **lateN1/N2** | 47 | **235** | 0 |

**Table 2.** EM clustering results for LP1group1 pattern factors

### 5.1. Expectation-Maximization clustering

The EM algorithm [8] is used to approximate distributions using mixture models. It is an iterative procedure that circles around the expectation and maximization steps. In the E-step for clustering, the algorithm calculates the posterior probability, $h_{ij}$, that a sample $j$ belongs to a cluster $C_i$:

$$h_{ij} = P(C_i|D_j) = \frac{p(D_j|\theta_i)\pi_i}{\sum_{m=1}^{C} p(D_j|\theta_m)\pi_m} \qquad (1)$$

where $\pi_i$ is the weight for the $ith$ mixture component, $D$ is the attribute, and $\theta_i$ is the set of parameters for each density functions. In the M-step, the EM algorithm searches for optimal parameters that maximize the sum of weighted log-likelihood probabilities. EM automatically selects the number of clusters by maximizing the logarithm of the likelihood of future data. A detailed implementation of EM clustering can be found at [14]. In the present study, we used the EM clustering algorithm implemented in WEKA [7].

### 5.2. Clustering results

For each of the four experimental datasets, we applied EM clustering to the summary metrics described previously (Table 1). The number of clusters was set equal to the number of patterns that were identified by domain experts. Observations were then assigned to clusters using this semi-automatic approach. Table 3 shows the clustering results for one of the four datasets. The resulting assignment of observations to clusters corresponded closely with the classification results based on expert judgments. On the other hand, there was not a strict one-to-one mapping between clusters and target patterns. Rather, the results showed some pattern "splitting," where observations belonging to a target pattern were assigned to more than one cluster. The proper diagnosis and interpretation of such results will require careful system evaluation to determine the source of this "misallocation of variance" [12]. Figure 3 visualizes the instance distribution through 3 clusters on one attribute - IN-mean (ROI).

| **Cluster/Pattern** | **0** | **1** | **2** | **3** |
|---|---|---|---|---|
| **P100** | 0 | **76** | 0 | 2 |
| **N100** | **117** | 1 | 0 | 54 |
| **lateN1/N2** | 13 | 14 | 0 | **104** |
| **P300** | 0 | 61 | **110** | 42 |

**Table 3.** EM clustering results for LP1group2 pattern factors

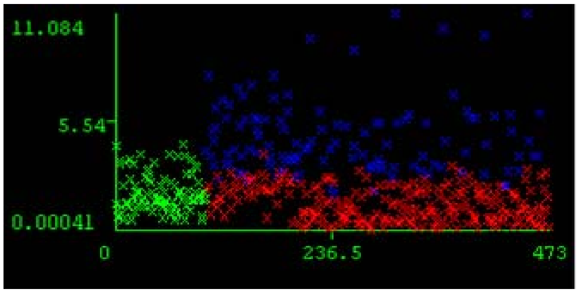

**Figure 3.** Visualization of LP1 group1 clustering result: x-axis is the instance number; y-axis is the value of IN-mean. Instances in cluster 0, 1, 2 are colored as green, red and blue respectively.

### 5.3. Cluster-based classification

EM clustering automatically partitions observations into clusters, as described in the previous section. A related goal is to develop rules that accurately assign observations to clusters. Therefore, after EM clustering, we use classification methods to build decision tree learners. Observations in each cluster can be labeled with cluster names without considering the experts' labels. Once the clustering process becomes more robust, this will obviate the need for manual labeling of patterns, providing a considerable savings in time and a sizable gain in information processing for ERP analysts.

### 5.4. Decision tree classifier

We use a traditional classification technique - the decision tree learner, in the present analysis. Decision trees [14] are flowchart-like trees with each internal node representing an attribute and each leaf node representing a class label. We used J48 in WEKA, which is an implementation of C4.5 algorithm [15], to classify the data. The input to the decision tree learner consists of the observation metrics described in Table 1; These metrics are represented as a vector of dimension 25. Cluster labels are used as classification labels. Figure 4 shows the decision tree learner, which was trained on the Experiment 1, Sample 2 data. It achieves a precision of 97.44% on the training data.

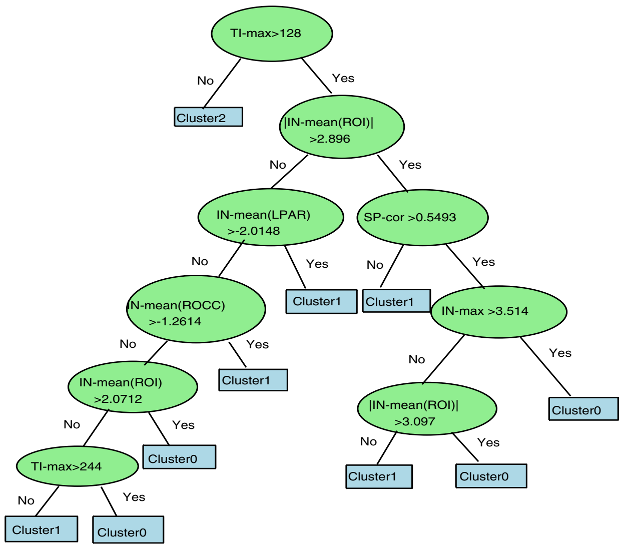

**Figure 4.** Decision tree classifier.

## 5.5. Information gain

From the decision tree shown in Figure 4, we can see that although 25 attributes are input to the learning process, only 6 attributes are used in the final decision tree classifier. This is because there is an attribute selection measure that is used in building the decision tree, which selects the attribute that is most efficient in differentiating classes of data at each level of the tree. The metrics that is used to evaluate this differentiability is called *information gain*. Table 4 shows the information gain of each attribute. In the rules provided by domain experts, the temporal criterion TI-max and the spatial criterion IN-mean (ROI) are used. Information gains for the complete list of attributes suggest that additional spatial metrics, such as SPr (correlation of factor topography with an a priori defined spatial template for a particular pattern) and mean amplitude over left parietal and right occipital sites (IN-mean (LPAR) and IN-mean(ROCC), provide added gains in classification accuracy.

## 5.6. Rule comparison

One advantage of using decision trees is that we can generate rules automatically and use these results to extend and refine the rules generated by domain experts. For example, Table 5 compares the auto-generated (decision tree) rules and expert-generated rules for the N100 and N2 patterns. Two differences that are observed between the two rule types are consequences of the two analysis strategies that have minimal consequences for high-level rule definition. First, the "modality" criterion that is included in the expert-generated rules for the N100 and N2 patterns has a constant value (=visual). Therefore, it is not used in the clustering process. Second, the temporal metrics are only marginally informative in the clustering, given that the temporal PCA reduced the dimensionality from >1,000 to 15 time points.

| Attribute | Average-Merit | Average Ranking |
|---|---|---|
| TI-max | 0.836 | 1 |
| IN-mean (ROI) | 0.238 | 2.2 |
| IN-mean (ROCC) | 0.224 | 3.3 |
| SP-cor | 0.215 | 3.6 |
| ... | ... | ... |

**Table 4.** Information gains of summary metrics

| Expert-defined rule | Decision tree rule |
|---|---|
| $\forall n,\ FAn = N100\, iff$<br>$150ms < TI - max <= 220ms$<br>$\wedge IN - mean(ROI) < -0.4$<br>$\wedge EVENT\ =\ stimon$<br>$\wedge MODALITY\ =\ visual$ | $\forall n,\ FAn \in cluster0\, iff$<br>$TI - max > 128ms$<br>$\wedge \lvert IN - mean(ROI) \rvert > 2.896$<br>$\wedge SP - cor > 0.549$<br>$\wedge IN - max > 3.514$ |
| $\forall n,\ FAn = lateN1/N2\, iff$<br>$220ms < TI - max <= 300ms$<br>$\wedge IN - mean(ROI) < -0.4$<br>$\wedge EVENT\ =\ stimon$<br>$\wedge MODALITY\ =\ visual$ | $\forall n,\ FAn \in cluster1\, iff$<br>$TI - max > 128ms$<br>$\wedge \lvert IN - mean(ROI) \rvert > 2.896$<br>$\wedge SP - cor <= 0.549$<br>... |

**Table 5.** Expert-defined rules vs. Decision tree generated rules

On the other hand, the inclusion of additional spatial metrics in the auto-generated rules – beyond just IN-mean(ROI) – is extremely interesting. In particular, EM clustering results suggested that the SPr spatial metric was important in defining both the N100 (cluster 0) pattern and lateN1/N2 (cluster 1) patterns. Recall from Table 1 that metric is defined as the correlation between an a priori defined spatial "template" (topography) for a target pattern and the spatial projector (topography) for a particular factor. The auto-generated rule for the N100 (cluster 1) requires that the correlation be greater than 0.55. This suggests that the use of a spatial template can be useful for ERP pattern detection, consistent with some prior results [12].

## 6. Future work

We have outlined a new framework for semi-automated classification of ERP patterns and rule generation. As described here, this approach can be highly informative when applied to PCA-based metrics generated from high-density ERP data. Ongoing work is focused on system evaluation, that is, identification of errors in pattern classification and potential weaknesses in various system components -e.g., PCA decomposition methods, clustering, or methods for generating decision-tree rules. For example, in the present set of experiments, some patterns "split" across (were assigned to) more than one cluster. Inspection of tPCA results suggested that refinements to the data decomposition process, as well as additional metrics that capture temporal and spatial attributes more accurately, may reduce this "misallocation" of pattern variance. To achieve true accuracy in system evaluation, we will compare system results with a "gold standard," which will be established by expert labeling of early visual-evoked ERP patterns (the P100v, N100v, and N2v).

With further refinements to our pattern classification framework, we will be able to apply this framework to automatically label and store existing ERP patterns. A long-term goal for this project is to store high-level pattern descriptions in a formal ERP ontology database. To this end, we will use a Semantic Web ontology language (e.g., OWL [5]) and a rule language (e.g., SWRL [6]) to define ERP ontologies and their mappings. The ontologies can be used for integrating the data from different resources with an information integration framework OntoGrate [11]. Eventually, this methodology can be extended for integrating other types of neuroscience data (e.g., ERF and fMRI data) and can support other biomedical ontology-based data sharing efforts (e.g., the Gene Ontology.)

## 7. Conclusion

In this paper, we have introduced a semi-automatic framework for mining ERP patterns. This work aims to develop robust methods for classifying and labeling ERP patterns for individual subjects, and for identifying important metrics in classification, which can lead to refinement of high-level concepts and rules. An important feature of our approach is the synergistic nature of bottom-up and top-down methods for ERP pattern classification. The resulting patterns and their definitions can be used in the development of ERP ontologies, as we have described elsewhere [10]. Further, we expect that the methods used to develop our ERP pattern classification framework, and related ontologies, can be extended to other types of neuroscience data and can support other biomedical ontology-based data sharing efforts.

## References

[1] Dien, J. PCA Toolbox (version 1.7). `http://people.ku.edu/ jdien/`.

[2] EEG lab. `http://www.sccn.ucsd.edu/eeglab/`.

[3] Electrical Geodesics, Inc. `http://www.egi.com`.

[4] NetStation Technical Manual. `http://www.egi.com`.

[5] OWL Web Ontology Language. `http://www.w3.org/TR/owl-ref/`.

[6] SWRL: A Semantic Web Rule Language Combining OWL and RuleML. `http://www.w3.org/Submission/SWRL/`.

[7] Weka 3: Data Mining Software in Java. `http://www.cs.waikato.ac.nz/ml/weka/`.

[8] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the em algorithm. *J. Royal Statistical Society*, 39:1–38, 1977.

[9] J. Dien. Addressing misallocation of variance in principal components analysis of event-related potentials. *Brain Topography*, 11(1):43–55, January 1998.

[10] D. Dou, G. Frishkoff, A. Malony, and D. Tucker. Neural electromagnetic ontologies: Erp/meg knowledge representation and integration. *NIH proposal*, October 2006.

[11] D. Dou, P. LePendu, S. Kim, and P. Qi. Integrating Databases into the Semantic Web through an Ontology-based Framework. In *Proceedings of the third International Workshop on Semantic Web and Databases (SWDB'06)*, page 54, 2006. co-located with ICDE 2006.

[12] R. M. Frank and G. A. Frishkoff. Automated protocol for evaluation of electromagnetic component separation (apecs): Application of a framework for evaluating statistical methods of blink extraction from multichannel eeg. *Clin Neurophysiol*, 118(1):80–97, 2007.

[13] G. Frishkoff. Hemispheric differences in strong versus weak semantic priming: Evidence from event-related brain potentials. *Brain Lang.*, 2006.

[14] J. Han and M. Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers, 2001.

[15] J. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, 1993.