Comput Stat (2013) 28:5–18
DOI 10.1007/s00180-011-0292-y

ORIGINAL PAPER

On the discovery of events in EEG data utilizing information fusion

Martin Schels · Stefan Scherer · Michael Glodek · Hans A. Kestler · Günther Palm · Friedhelm Schwenker

Received: 18 October 2010 / Accepted: 2 November 2011 / Published online: 29 November 2011

Abstract One way to tackle brain computer interfaces is to consider event related potentials in electroencephalography, like the well established P300 phenomenon. In this paper a multiple classifier approach to discover these events in the bioelectrical signal and with them whether or not a subject has recognized a particular pattern, is employed. Dealing with noisy data as well as heavily imbalanced target class distributions are among the difficulties encountered. Our approach utilizes partitions of electrodes to create robust and meaningful individual classifiers, which are then subsequently combined using decision fusion. Furthermore, a classifier selection approach using genetic algorithms is evaluated and used for optimization. The proposed approach utilizing information fusion shows promising results (over 0.8 area under the ROC curve).

Keywords Multiple classifier systems · EEG data · P300 event

M. Schels · S. Scherer · M. Glodek · H. A. Kestler · G. Palm · F. Schwenker (✉)
Institute of Neural Information Processing, University of Ulm, Ulm, Germany
e-mail: friedhelm.schwenker@uni-ulm.de

M. Schels
e-mail: martin.schels@uni-ulm.de

M. Glodek
e-mail: michael.glodek@uni-ulm.de

G. Palm
e-mail: guenther.palm@uni-ulm.de

S. Scherer
Speech Communication Laboratory, Trinity College Dublin, Dublin, Ireland
e-mail: scherers@tcd.ie

H. A. Kestler
Internal Medicine I, University Hospital, Ulm, Germany
e-mail: hans.kestler@uni-ulm.de

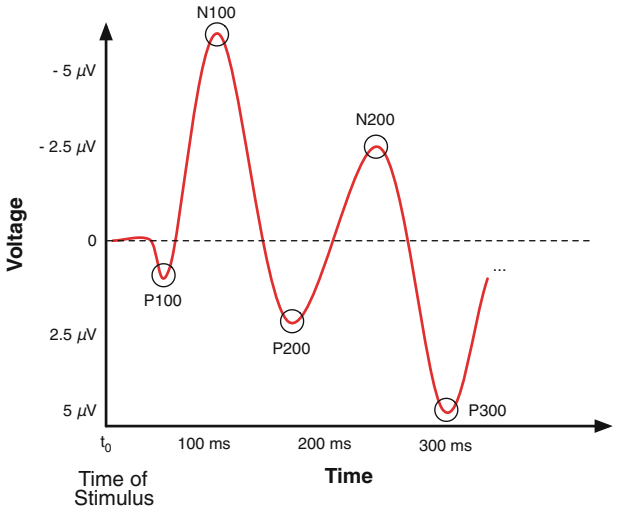

Fig. 1 Schematic presentation of the succession of bioelectrical artifacts after a stimulus. The *letter* denotes the sign of the voltage whereas the following number indicates the approximate delay after the trigger (plot adapted from Birbaumer and Schmidt 2006)

1 Introduction

In the past few years brain computer interfaces became part of the most prominent applications in neuroscience (Chumerin et al. 2009). In the present study the goal is to investigate the possibility to automatically determine whether a human subject has just seen a target on a presented image by solely analyzing the event related potentials (ERP), that are recorded using electroencephalography (EEG). ERP typically reflect cognitive processes in the brain that follow a more or less strict timely pattern that can be visualized by filtering and averaging ERP signal recordings (Gray et al. 2004). A typical EEG progression can be seen in Fig. 1. Well established states that are passed during this information process resembled in the signal are the P100, N200 and the well known P300, all named after their voltage and approximate delay of the stimulus-response (Gray et al. 2004; Dujardin et al. 1993). Furthermore, actual amplitudes and latencies in the typical ERP such as the N200 or the P300 are dependent on factors such as the subjects attention, age, the stimulus modality (e.g., audio or visual), and the frequency of the stimulus (Dujardin et al. 1993). Another source of ERP are motor signals which correspond to a task related physical action of a subject like pressing a button. Such a potential may be even further delayed after the stimulus and the aforementioned patterns. These bioelectrical phenomena are normally overlaid with heavy noise, that is caused inevitably even by subtle movements of a test person (e.g. heartbeats). To make the actual ERP visible a denoising technique called ensemble averaging (Sörnmo and Laguna 2005) is applied in physiology: for all sequences of a category, the subsequent samples after a stimulus are averaged. In the present investigation visual stimuli were presented by following a typical oddball paradigm: the non target (background) type stimuli were presented very frequently,

whereas the targets were displayed very rarely. According to Dujardin et al. (1993) this type of experimental setup should lead to a prominent P300 representation in the EEG.

Originating from these findings, it is compelling to design a machine classifier capable of detecting the subject's recognition of a target stimulus by monitoring the bioelectrical EEG stream. This particular setup imposes several challenges: the oddball recording technique of the data requires a special treatment due to the skewed distribution of classes. Heavily imbalanced datasets require special treatment in order to mitigate the over-representation of a class. Popular techniques are under- and oversampling of the training set with respect to the categories or the usage of error functions that account for skew distributions of classes (Japkowicz 2000; Zhou and Liu 2006). Also, the noisy nature of the employed sensors can impair the recognition performance. Methods designed to improve robustness in low signal to noise ratio conditions include low pass filtering but also information or data fusion (Kuncheva 2002). In this particular domain of information fusion various possibilities to ensure robustness can be applied. In our approach, we gain robustness by combining multiple feature channels as well as by combining the outputs of several independent classifiers.

The remainder of this paper is organized as follows: in Sect. 2, the employed data collection is described in greater detail. In Sect. 3 the main issues about constructing the employed classifiers are explained together with classifier selection and classifier fusion. The conducted experiments are described in Sect. 4, finally, Sect. 5 concludes.

2 Data collection and feature extraction

In this study, the dataset provided by the "Machine Learning for Signal Processing 2010 Competition: Mind Reading[1]" is utilized. The goal of the competition is the classification of stimuli by analyzing EEG recordings. For these recordings, satellite images were presented in a fast sequence to a test person, who was instructed to push a button when a surface to air missile (SAM) site was shown. The images were shown to the subject in a resolution of 500×500 pixels every 100 ms. The data is presented in 75 blocks of 37 images leading to a total number of 2,775 images. Each block is separated by a pause ended by the subject independently. However, only a marginal fraction of the images are actual triggers (i.e. a SAM site is shown) such that the task of identification can be seen as a typical oddball paradigm task (Segalowitz et al. 2001). Out of the 58 triggers only 48 triggers were identified by the subject within a reasonable time window after the presentation of the satellite image containing an actual missile site. For the testing and training only the EEG data recorded after these 48 correctly identified triggers are used in order to ensure that the subject has actually found the target and therefore generating a meaningful EEG phenomenon.

The EEG data consists of 64 channels in total that are recorded with a sampling rate of 256 Hz. The sensors are arranged as it is shown in the center of Fig. 3. Along with the data from these 64 electrodes the onset time of the pressed space bar and the type of displayed image (trigger or no trigger) are provided in the dataset.

In order to prepare the data for classification five different features were extracted locally from every EEG channel. The samples of a time window of 0.5 second

[1] http://www.bme.ogi.edu/~hildk/mlsp2010Competition.html.

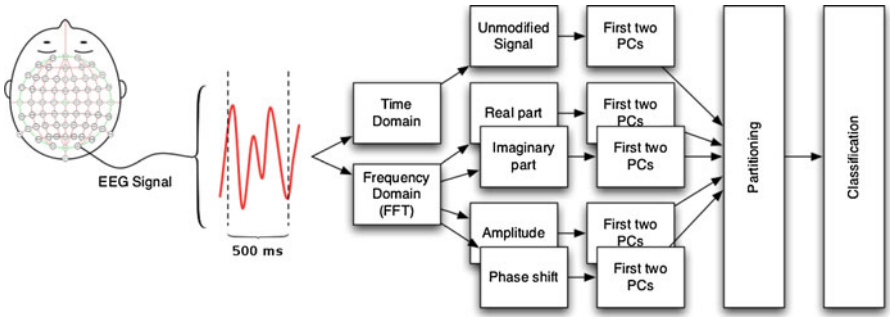

Fig. 2 Feature extraction procedure: the subsequent half of a second is processed in time and frequency domain. Overall, $16 \times 5 \times 2 = 160$ different features per partition (see Fig. 3) are passed to the classification architecture

following each image trigger event were isolated for subsequent analysis. This could also be conducted using unsupervised learning for sequential data like e.g. described in Genolini and Falissard (2010). The frame length of 0.5 second was chosen in order to capture fully the typical ERP (as depicted in Fig. 1). The features for the analysis ERP were computed in both, the frequency and the time domain (Picton et al. 2000). To obtain a first feature, the first two principal components upon this sequence of samples were calculated using a PCA. Four more features were generated by applying the fast Fourier transformation (FFT) on the aforementioned windows. The real and imaginary part of the resulting frequency spectrum were utilized separately to form a second and third feature. The amplitude and the phase shift of the particular frequencies were computed to form a fourth and fifth characteristic features. All these values were separately passed to a further principal component analysis, projecting the data on the two components, having the highest variance. Figure 2 displays the basic steps of this feature extraction procedure.

A comprehensive overview of the competition can be found in Hild et al. (2010). Many different classifier approaches have been evaluated in this context with the support vector machine being the most frequent one. Also, the concept of bagging has been widely used. Thus performances of up to 0.82 area under the ROC curve (AUC) have been reached (for a description of the top-scoring approaches please refer to Leiva and Martens 2010; Iscan 2010; Labbe et al. 2010).

3 Classification and classifier fusion

3.1 Support vector machines and imbalanced distribution of classes

Many real applications in pattern recognition need to deal with imbalanced training sets. Common techniques to mitigate this issue are over-sampling of the underrepresented classes or under-sampling of the overrepresented classes (Japkowicz 2000; Zhou and Liu 2006). Another approach to imbalanced data sets is utilizing a particular loss function in the chosen classifier design such as individual cost terms for each class. Such a loss function penalizes misclassification of underrepresented classes more severely than others. In the following, such a loss term is incorporated into

the common formulation of the support vector machine (Schölkopf and Smola 2001) as e.g. proposed in Osuna et al. (1997). This concept has proven to be feasible under different circumstances as *fuzzy-input fuzzy-output support vector machine* concept as described by Thiel et al. (2007). A second concept, we exploit in this context is the measurement of class confidence using the distance of a sample from the decision boundary. We gain additional flexibility for the classifier fusion scheme by utilizing these membership degrees instead of crisp labels.

In this section, we describe our modifications to the SVM concerning the proposed loss term for imbalanced classification tasks. The class weights for every data sample are given in two N-dimensional vectors $\mathbf{m}^+$ and $\mathbf{m}^-$ which contain the relative proportions of the two opposing classes in the training data and hence $\mathbf{m}^+ + \mathbf{m}^- = (1, \ldots, 1)$. The goal is then to maximize a soft-margin, stated as

$$\underset{\mathbf{w},b,\boldsymbol{\xi}^+,\boldsymbol{\xi}^-}{\operatorname{argmin}} \ \frac{1}{2}||\mathbf{w}||_2 + C\sum_{n=1}^{N}(\xi_n^+ m_n^+ + \xi_n^- m_n^-) \tag{1}$$

where $n = 1, \ldots, N$ denotes the index of the training samples and $\boldsymbol{\xi}^+$ and $\boldsymbol{\xi}^-$ are linear slack variables penalizing a sample being misclassified. The parameter C controls the penalty of incorrect class assignments. The minimization problem of Eq. (1) is subjected to the constraints

$$\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_n) + b \geq 1 - \xi_n^+ \tag{2}$$
$$\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_n) + b \geq -(1 - \xi_n^-) \tag{3}$$
$$\xi_n^+ \geq 0 \tag{4}$$
$$\xi_n^- \geq 0 \tag{5}$$

where $\mathbf{w}$ and b describe the orientation and the bias of the hyperplane, and $\phi(.)$ denotes a transformation of $x_n \in \mathbb{R}^n$ into a potentially higher dimensional Hilbert space H. The corresponding Lagrangian of this optimization problem defined by (1)–(5) is given by:

$$\begin{aligned} L(\mathbf{w}, b, \xi^+, \xi^-) = \frac{1}{2}||\mathbf{w}||_2 + C\sum_{n=1}^{N}(\xi_n^+ m_n^+ + \xi_n^- m_n^-) \\ - \sum_{n=1}^{N}\alpha_n^+((\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_n) + b) - 1 + \xi_n^+) \\ + \sum_{n=1}^{N}\alpha_n^-((\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_n) + b) + 1 - \xi_n^-) \\ - \sum_{n=1}^{N}\beta_n^+\xi_n^+ \\ - \sum_{n=1}^{N}\beta_n^-\xi_n^-. \end{aligned}$$

Differentiating the Lagrangian with respect to the Lagrangian multipliers and $\mathbf{w}$, b, $\boldsymbol{\xi}^+$, $\boldsymbol{\xi}^-$ and subsequently eliminating the parameters of the hyperplane and the slack variables results in the dual Lagrangian:

$$\widetilde{L}(\alpha^+, \alpha^-) = \sum_{n=1}^{N} \alpha_n^+ + \sum_{n=1}^{N} \alpha_n^- - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} (\alpha_n^+ - \alpha_m^-)(\alpha_m^+ - \alpha_n^-) k(\mathbf{x}_n, \mathbf{x}_m), \quad (6)$$

where the kernel function is defined by $k(\mathbf{x}_n, \mathbf{x}_m) = \phi(\mathbf{x}_n)^{\mathrm{T}} \phi(\mathbf{x}_m)$ and the maximization problem is now constrained to:

$$\begin{aligned} &\sum_{n=1}^{N} (\alpha_n^+ - \alpha_n^-) = 0, \quad \text{with} \\ &0 \leq \alpha_n^+ \leq C m_n^+, \quad \text{and} \\ &0 \leq \alpha_n^- \leq C m_n^-. \end{aligned} \quad (7)$$

In order to satisfy the Karush–Kuhn–Tucker conditions, properties

$$\alpha_n^+ ((\mathbf{w}^{\mathrm{T}} \phi(\mathbf{x}_n) + b) - (1 - \xi_n^+)) = 0, \quad (8)$$
$$\alpha_n^- ((\mathbf{w}^{\mathrm{T}} \phi(\mathbf{x}_n) + b) + (1 - \xi_n^-)) = 0, \quad (9)$$
$$\beta_n^+ \xi_n^+ = (C m_n^+ - \alpha_n^+) \xi_n^+ = 0, \quad (10)$$
$$\beta_n^- \xi_n^- = (C m_n^- - \alpha_n^-) \xi_n^- = 0, \quad (11)$$
$$\forall n = 1, \ldots, N,$$
$$\alpha^+, \alpha^-, \beta^+, \beta^- \geq 0$$

and properties (2)–(5) hold. A numerical solution can be computed using the *sequential minimal optimization* (SMO) approach introduced by Platt (1999a).

Once the Lagrangian multipliers $\boldsymbol{\alpha}^+$ and $\boldsymbol{\alpha}^-$ have been found, the parameters $\mathbf{w}$ and b of the hyperplane are determined by:

$$\begin{aligned} \mathbf{w} &= \sum_{n=1}^{N} (\alpha_n^+ - \alpha_n^-) \phi(\mathbf{x}_n), \quad \text{and} \\ b &= \frac{1}{2N_{\mathcal{M}^+}} \sum_{n \in \mathcal{M}^+} \left(1 - \sum_{l \in \mathcal{S}^+} (\alpha_n^+ - \alpha_n^-) k(\mathbf{x}_n, \mathbf{x}_m) \right) \\ &\quad + \frac{1}{2N_{\mathcal{M}^-}} \sum_{n \in \mathcal{M}^-} \left((-1) - \sum_{l \in \mathcal{S}^-} (\alpha_n^+ - \alpha_n^-) k(\mathbf{x}_n, \mathbf{x}_m) \right), \end{aligned} \quad (12)$$

where $\mathcal{S}^+(\mathcal{S}^-)$ is the set of support vectors $\alpha_n^+ > 0 (\alpha_n^- > 0)$ and $\mathcal{M}^+(\mathcal{M}^-)$ is the set of unbounded support vectors with $\alpha_n^+ < C m_n^+ (\alpha_n^- < C m_n^-)$. According to Eqs. (10) and (11) $\xi_n^+ (\xi_n^-) = 0$ if the sample n is in the set $\mathcal{M}^+(\mathcal{M}^-)$. The bias parameter b is

averaged by using Karush–Kuhn–Tucker conditions (8) and (9) to obtain a numerically stable solution.

A class decision can then be obtained by $y = \text{sign}(\mathbf{w}^{\mathrm{T}}\mathbf{x} + b)$. To extend the SVM to a probabilistic output, the distance $d(\mathbf{x})$ of the input $\mathbf{x}$ to the hyperplane is mapped to $\widetilde{y}(\mathbf{x}) \in (0, 1)$ using a sigmoid function (Platt 1999b) with parameter a

$$\widetilde{y}(\mathbf{x}) = (1 + \exp(-a \cdot d(\mathbf{x})))^{-1}$$

can be minimized according to the mean square error on the training data. Since $m_n^+ = 1 - m_n^-$ we only need to consider m_n^+ and optimize

$$E = \frac{1}{N} \sum_{n}^{N} (\widetilde{y}(\mathbf{x}_n) - m_n^+)^2,$$

which can be accomplished by a linear regression technique.

3.2 Classifier fusion and classifier selection

Pattern recognition aims for high recognition rates. One popular way to manage this is to develop and optimize a feature for the particular task and to train and optimize matching classifiers. But typically it is not so clear which feature or classifier approach are the most qualified ones. Therefore, another option is to create several classifiers using different feature approaches and then combine the outputs of the individual classifiers in an appropriate way (Kuncheva et al. 2001; Kittler et al. 1998). A classifier constructed under such paradigm is called multiple classifier system (MCS). In order to benefit from classifier combination individual classifiers, but should be accurate and diverse (Kuncheva and Whitaker 2003).

There are many different approaches to combine classifier outputs to improve classification performances. The methodology of classifier fusion can be divided in approaches implementing a fixed combination (Kittler et al. 1998) such as majority voting or average fusion and approaches, that can be adapted to a specific classifier combination and application (Kuncheva et al. 2001). Examples for trainable classifier fusion schemes are decision templates (Kuncheva et al. 2001) or the mixture of experts approach (Jordan and Jacobs 1994). For any of these MCS approaches, it is essential to construct a classifier team, i.e. pool of classifiers, that incorporates both, individual accuracy and diversity as a team. Suppose one has constructed a variety of individual classifiers, now, it could be essential to select from these underlying classifiers a team of classifiers that is optimal with respect to a particular application. This search process in the space of possible classifier teams is called classifier selection (Kuncheva 2002; Giacinto and Roli 1999) and uses similar techniques as feature selection. There are many local search approaches, adapted from feature selection, to deal with this well known issue like sequential forward selection (Guyon and Elisseeff 2003).

Another search strategy, that is applied in the literature to classifier selection is the meta learning approach of genetic algorithms (Ruta and Gabrys 2005; Kuncheva and Jain 2000). For genetic algorithms a pool of instances of possible solutions of a

problem, often represented as a binary string, is kept. These instances are manipulated in two different ways using mutation and recombination operations. Mutation is realized by an independent bitwise flip with a predefined probability and the recombination step is realized using a crossover operation by selecting a random point on two instances (i.e. parents) and exchanging parts mutually between them. The operations generate new possible solutions that are pooled together with the initial population (Bäck 1996). Thereafter, the pool of instances is evaluated regarding a predefined fitness function and according to this function a number of individuals is selected to form a new population. Such a strategy is called fitness based survivor selection. Using this population the aforementioned steps are iteratively performed until a stopping condition is reached, e.g. maximum number of iterations or convergence. These iterations of crossover, mutation and selection are called epochs in the following. This genetic algorithm uses a combination of explorative and exploitative (recombination) search steps (mutation).

To bring the concept of classifier selection to genetic algorithms, the representation of a classifier team and an appropriate fitness function have to be defined. There is a natural representation of a population of classifier teams as a bit string indicating whether a classifier is a member of the particular team ("1") or not ("0"). More challenging is the question for a fitness function: there are attempts to rate classifier teams concerning its diversity utilizing the κ diversity measure (Kuncheva and Whitaker 2003) and the related κ-error diagrams, that may be utilized to construct classifiers using AdaBoost (Margineantu and Dietterich 1997). Various other types of diversity criteria can be used, e.g. measures derived from cluster analysis (Kraus et al. 2011) or the generalization error of a classifier team on a validation set as for example implemented in Ruta and Gabrys (2005).

The general concept of multiple classifier systems has been previously applied to the classification of EEG. In Zhang et al. (2007), the ensemble technique boosting is used together with RBF networks and independent component analysis for feature extraction. An ensemble of SVM is employed in Rakotomamonjy and Guigue (2005) for a P300 speller. The SVM are combined using the average over the individual decisions, but also an integration over time is conducted. In Xu et al. (2008), an ensemble of SVM was used in a localized fashion, were the individual classifiers were constructed based on different partitions of the data that are determined using clustering. For testing they choose the respective classifier by computing a k nearest neighbor using th training data.

4 Experiments and results

Before passing the computed features to our machine learning approach, the EEG channels were partitioned into nine overlapping areas containing up to 18 channels at a time. Eight partitions are chosen as coherent slices and the ninth one is defined as the horizontal and vertical cutoff of the EEG device's layout (see Fig. 3 for an overview). These partitions were defined from a machine learning point of view rather than from physiology: thus, one can provide more information for individual classifiers that using only one electrode, but it is still possible to conduct decision fusion. For every partition, the features extracted from the respective electrodes were concatenated to

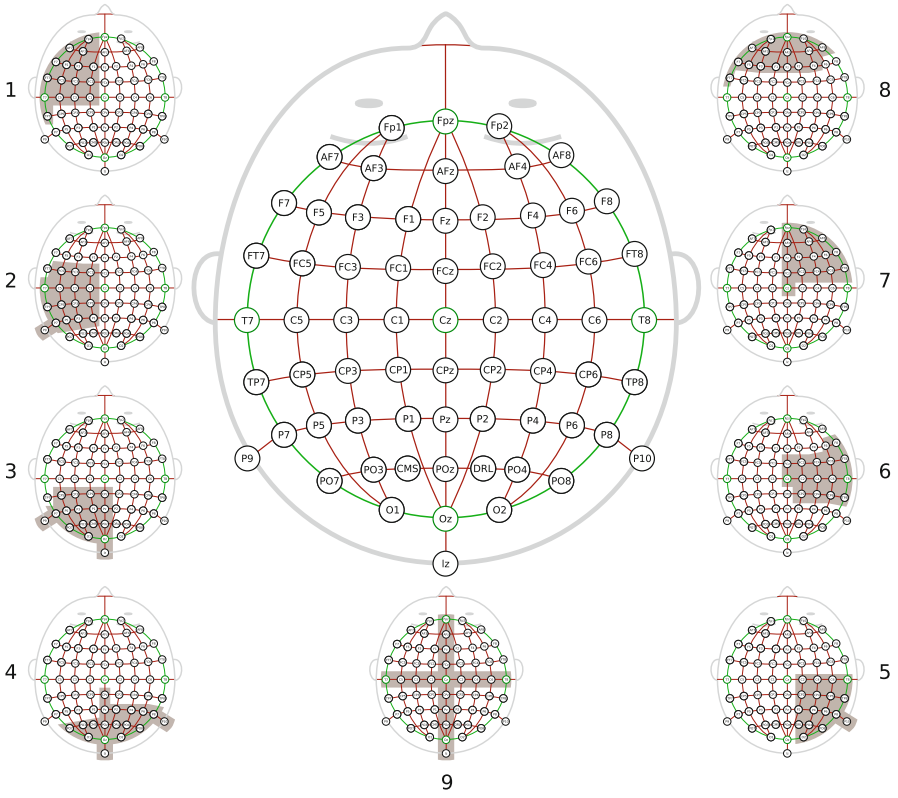

Fig. 3 Positions of the 64 electrodes on the scalp of the subject (Image adapted from http://www.biosemi.com). The small images (1–9) surrounding the main layout illustrate the defined partitions of the 64 EEG channels (*gray* regions), that will be the inputs to the classifiers of the multiple classifier system

form a new feature. Thus, a feature level fusion approach is realized and by doing so the individual classifiers that are constructed based on this representation are supposed to get informative input.

Subsequently, the resulting 45 sets of channels—due to the combination of the five kinds of features with the nine partitions—were trained and classified separately. Preliminary cross validation experiments for k-nearest neighbors, multi layer perceptrons and various types of SVM showed, that weighted SVM—with Gaussian kernel—outperforms the others by AUC. We chose to train the SVM for classification using a Gaussian kernel. As described in Sect. 3.1, a loss term was integrated in this SVM approach tackling the issue of the imbalanced training and test sets (48 positive samples avs. 2,700 negative samples). The performance of the classifiers is determined by the area under the ROC curve (AUC). The results of this first classification step are ranging from a classification performance of 0.478 AUC, which is close to random, to 0.836 AUC. An overview of the performances of all constructed classifiers is depicted in Table 1, showing the average of 15 sixfold cross-validation rounds.

It can be observed from Table 1 that the individual performance depends on the chosen feature extraction approach. Especially the feature extracted from the real part

Table 1 Performances of the constructed individual classifiers in terms of area under the ROC curve

Partition of electrodes	Real part of FFT	Imaginary part of FFT	Amplitude	Phase shift	PCA over time
1	0.622 (0.089)	0.733 (0.091)	0.669 (0.083)	0.480 (0.087)	0.714 (0.101)
2	0.744 (0.087)	0.774 (0.081)	0.593 (0.092)	0.586 (0.088)	**0.836** (0.065)
3	0.758 (0.093)	0.723 (0.090)	0.573 (0.082)	0.497 (0.042)	0.794 (0.082)
4	0.795 (0.078)	0.711 (0.085)	0.610 (0.076)	0.481 (0.082)	0.811 (0.078)
5	0.689 (0.095)	0.698 (0.102)	0.602 (0.086)	0.502 (0.089)	0.659 (0.111)
6	0.677 (0.102)	0.764 (0.086)	0.577 (0.084)	0.478 (0.085)	0.750 (0.090)
7	0.654 (0.096)	0.722 (0.097)	0.493 (0.089)	0.481 (0.071)	0.784 (0.070)
8	0.630 (0.075)	0.755 (0.082)	0.682 (0.086)	0.468 (0.092)	0.796 (0.084)
9	0.661 (0.101)	0.714 (0.098)	0.650 (0.083)	0.522 (0.084)	0.737 (0.100)

The numbers refer to the partitions defined in Fig. 3. The classifiers group themselves regarding the performance by feature types: on the one hand features from the real part of the FFT and PCA in time domain perform well, on the other hand features from phase shift coefficients are only slightly better than random. The SD of the conducted runs is given in in parentheses
The bold value indicates the most accurate individual classifier

of FFT and from time domain results in strong performances, while features from phase shift reveal rather weak performances. On the other hand the actual partition seem to be less important considering this measure: the variability in the columns is relatively small compared to the previously mentioned findings.

In order to further enhance the performance of the proposed classifier, a decision fusion step was implemented to combine the obtained outputs. We decided to use an averaging classifier fusion approach because of stability reasons (Kittler et al. 1998). In order to find a suitable combination of classifiers, a basic genetic search algorithm approach was implemented as described in Sect. 3.2. The classifier selection was optimized locally in every cross validation run: a validation set—i.e. the data of onefold of the training data—is left beside in the training of the individual classifiers and the fitness of the classifier ensembles is determined and optimized on this set. For the subsequent experiments, the number of maintained individual solutions and the maximum number of epochs were set to 10.

Results for this averaging classifier fusion approach are reported in Table 2 with and without classifier selection procedure. Using classifier selection, combinations of classifiers, which further increased the area under the performance were found: The performance of the classifier selection process on the test set is 0.860 AUC while the actual performance when skipping the selection process is marginally smaller (0.853 AUC). Both approaches do actually outperform the optimal individual classifier showing the highest performance (see Table 1). These results reveal that the classifier selection step does not yield benefits in this particular application.

The number of selections of the 45 individual classifiers in 4,050 classifier selection experiments is depicted in Fig. 4. Even though the discriminant message of this figure may be subtle one could argue that the features computed from the amplitudes computed by FFT (green) and maybe the analogical phase shift (cyan) or the real part (blue) are selected more often than the others. Especially in case of Phase shift features this is surprising, because these features show relatively poor performances in Table 1.

Table 2 Results of the classifier fusion an classifier selection approaches in terms of area under ROC curve

Fusion procedure	AuROC
Classifier fusion/selection with GA	0.860 (0.074)
Classifier fusion solely	0.853 (0.081)
Best individual classifier	0.836 (0.065)
Random forest with class weighting	0.789 (0.047)

The results including the classifier selection step are based on 76 search attempts. The results without classifier selection are computed with 15 separate sixfold cross validations. The SD of the conducted runs is given in in parentheses

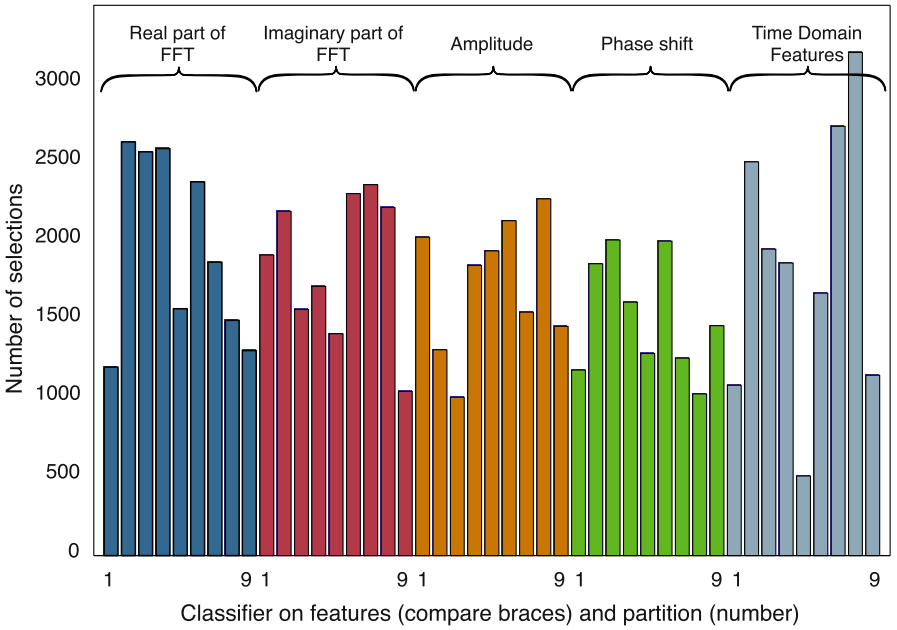

Fig. 4 Number of appearances of a particular classifier being selected: the *x* axis depicts the different partitions of the electrodes using digits 1–9 and for the 5 feature types (compare the braces on the upper part of the figure). The *y* axis shows the number of selections

In order to compare the proposed approach to state of the art classification techniques the experiment has also been conducted using Breiman's random forest (Breiman 2001) extended as proposed by Chen et al. (2004), which takes into account the imbalanced class distribution by weighting with respect to the frequency of the classes. These results are also listed in Table 2. It can be observed that this state of the art approach is outperformed by the proposed SVM in this application.

5 Discussion and conclusion

An information fusion approach to discover ERP in EEG data recorded using an oddball paradigm is described in this paper. Firstly 45 individual classifiers were

trained using various feature extraction strategies and a low level information fusion technique defining partitions of EEG electrodes. Utilizing this first fusion step, the individual classifiers could be constructed to reveal a good performance even though the underlying data is noisy and the distribution of the isolated features of a channel are heavily overlapping.

Further, we implemented a decision fusion procedure and compared a genetic algorithm based classifier selection to a standard averaging fusion approach. Generally, the combination of classifiers succeeded in improving the over-all performance. This can be be interpreted as a indication for the beneficial diversity of these classifiers in combination with others.

This evaluation showed that in this application the relatively computationally expensive search procedure did not bring significant improvement. Nevertheless, there may be an advantage in the selection process regarding the complexity of the classifier. Discarding some of the available classifiers reduces the efforts that have to be spent not only on the classifier fusion step, but also these classifiers obviously do not have to be evaluated. This reduction comes with the drawback of a costly search procedure which is, however, conducted off-line prior to testing. Furthermore, the fact that there are no tremendous differences concerning the length of the bins in the histogram might as well be an argument to explain, that skipping the classifier procedure does not decrease the performance.

In the future, we hope to be able to incorporate the developed architecture into more scenarios such as the recognition or detection of valanced events in cognitive technical companion systems. Hereby the EEG can be integrated into a broader framework to classify user states in human computer interaction together with other physiological signals (Walter et al. 2011).

Acknowledgments This paper is based on work done within the Transregional Collaborative Research Centre SFB/TRR 62 “Companion-Technology for Cognitive Technical Systems”, funded by the German Research Foundation (DFG). The work of Martin Schels is supported by a scholarship of the Carl-Zeiss Foundation. This work was further supported as part of the FASTNET project—Focus on Action in Social Talk: Network Enabling Technology funded by Science Foundation Ireland (SFI) (Grant 09/IN.1/I2631).

References

Bäck T (1996) Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms. Oxford University Press, Oxford

Birbaumer N, Schmidt RF (2006) Biologische psychologie. German edition. Springer, Berlin

Breiman L (2001) Random forests. Mach Learn 45:5–32

Chen C, Liaw A, Breiman L (2004) Using random forest to learn imbalanced data. Technical report, Department of Statistics, University of Berkeley

Chumerin N, Manyakov NV, Combaz A, Suykens JAK, Yazicioglu RF, Torfs T, Merken P, Neves HP, Van Hoof C, Van Hulle MM (2009) P300 detection based on feature extraction in on-line brain-computer interface. In: KI’09: Proceedings of the 32nd annual German conference on Advances in artificial intelligence, Springer, pp 339–346

Dujardin K, Derambure P, Bourriez JL, Jacquesson JM, Guieu JD (1993) P300 component of the event-related potentials (ERP) during an attention task: effects of age, stimulus modality and event probability. Int J Psychophysiol 14(3):255–267

Genolini C, Falissard B (2010) Kml: k-means for longitudinal data. Comput Stat 25:317–328

Giacinto G, Roli F (1999) Methods for dynamic classifier selection. In: ICIAP ’99 proceedings of the 10th international conference on image analysis and processing, IEEE Computer Society, pp 659–664

Gray HM, Ambady N, Lowenthal WT, Deldin P (2004) P300 as an index of attention to self-relevant stimuli. J Exp Soc Psychol 40(2):216–224
Guyon I, Elisseeff A (2003) An introduction to variable and feature selection. J Mach Learn Res 3:1157–1182
Hild KE, Kurimo M, Calhoun VD (2010) The 6th annual mlsp competition. In: IEEE international workshop on machine learning for signal processing (MLSP), pp 107–111
Iscan Z (2010) Mlsp competition, 2010: description of second place method. In: IEEE international workshop on machine learning for signal processing (MLSP), pp 114–115
Japkowicz N (2000) Learning from imbalanced data sets: a comparison of various strategies. In: Proceeding of AAAI workshop learning from imbalanced data sets, pp 10–15
Jordan MI, Jacobs RA (1994) Hierarchical mixtures of experts and the em algorithm. Neural Comput 6:181–214
Kittler J, Hatef M, Duin RPW, Matas J (1998) On combining classifiers. IEEE Trans Pattern Anal Mach Intell 20(3):226–239
Kraus J, Müssel C, Palm G, Kestler HA (2011) Multi-objective selection for collecting cluster alternatives. Comput Stat 26:341–353
Kuncheva L (2002) Switching between selection and fusion in combining classifiers: an experiment. IEEE Trans Syst Man Cybern Part B Cybern 32(2):146–156
Kuncheva L, Bezdek JC, Duin RPW (2001) Decision templates for multiple classifier fusion: an experimental comparison. Pattern Recogn 34(2):299–314
Kuncheva L, Jain L (2000) Designing classifier fusion systems by genetic algorithms. IEEE Trans Evol Comput 4(4):327–336
Kuncheva LI, Whitaker CJ (2003) Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. Mach Learn 51(2):181–207
Labbe B, Tian X, Rakotomamonjy A (2010) Mlsp competition, 2010: description of third place method. In: IEEE international workshop on machine learning for signal processing (MLSP), pp 116–117
Leiva J, Martens S (2010) Mlsp competition, 2010: description of first place method. In: IEEE international workshop on machine learning for signal processing (MLSP), pp 112–113
Margineantu DD, Dietterich TG (1997) Pruning adaptive boosting. In: ICML '97 proceedings of the 14th international conference on machine learning. Morgan Kaufmann Publishers Inc., pp 211–218
Osuna E, Freund R, Girosi F (1997) Support vector machines: training and applications. Technical report, Massachusetts Institute of Technology, Cambridge
Picton T, Bentin S, Berg P, Donchin E, Hillyard S, Johnson R, Miller G, Ritter W, Ruchkin D, Rugg M, Taylor M (2000) Guidelines for using human event-related potentials to study cognition: Recording standards and publication criteria. Psychophysiology 37(02):127–152
Platt J (1999a) Fast training of support vector machines using sequential minimal optimization. In: Advances in kernel methods, MIT press, pp 185–208
Platt J (1999b) Probabilistic outputs for support vector machines and comparison to regularize likelihood methods. In: Advances in large margin classifiers. MIT press, pp 61–74
Rakotomamonjy A, Guigue V (2005) Bci competition iii: dataset ii—ensemble of svms for bci p300 speller. IEEE Trans Biomed Eng 55(3):1147–1154
Ruta D, Gabrys B (2005) Classifier selection for majority voting. Inf Fusion 6(1):63–81
Schölkopf B, Smola AJ (2001) Learning with kernels: support vector machines, regularization, optimization, and beyond (adaptive computation and machine learning). The MIT Press, Cambridge
Segalowitz SJ, Bernstein DM, Lawson S (2001) P300 event-related potential decrements in well-functioning university students with mild head injury. Brain Cogn 45(3):342–356
Sörnmo L, Laguna P (2005) Bioelectrical signal processing in cardiac and neurological applications. 1. Elsevier, Amsterdam
Thiel C, Scherer S, Schwenker F (2007) Fuzzy-input fuzzy-output one-against-all support vector machines. In: Knowledge-based intelligent information and engineering systems 2007, Springer, pp 156–165
Walter S, Scherer S, Schels M, Glodek M, Hrabal D, Schmidt M, Böck R, Limbrecht K, Traue HC, Schwenker F (2011) Multimodal emotion classification in naturalistic user behavior. In: Proceedings of 14th international conference on human–computer interaction (HCII'11), Springer
Xu Y, Yin K, Zhang J, Yao L (2008) A spatiotemporal approach to n170 detection with application to brain-computer interfaces. In: IEEE international conference on systems, man and cybernetics, pp 886–891

Zhang JC, Xu YQ, Yao L (2007) P300 detection using boosting neural networks with application to bci. In: IEEE/ICME international conference on complex medical engineering, pp 1526–1530
Zhou ZH, Liu XY (2006) Training cost-sensitive neural networks with methods addressing the class imbalance problem. IEEE Trans Knowl Data Eng 18(1):63–77