EEG Hidden Information Mining Using Hierarchical Feature Extraction and Classification

Deng Wang, Duoqian Miao, Chen Xie, Hongyun Zhang
Department of Computer Science and Technology, Tongji University
Key Laboratory of Embedded System & Service Computing, Ministry of Education of China, Tongji University
Shanghai, 201804, China
E-mail: w_deng208@hotmail.com; miaoduoqian@163.com

Abstract—This paper presents a hierarchical feature extraction and classification method for electroencephalogram (EEG) hidden information mining. It consists of supervised learning for fewer features, hierarchical knowledge base (HKB) construction and classification test. First, the discriminative rules and the corresponding background conditions are extracted by using autoregressive method in combination with the nonparametric weighted feature extraction (NWFE) and k-nearest neighbor. Second, through ranking the discriminative rules according to validation test correct rate, a hierarchical knowledge base HKB is constructed. Third, given an EEG sequence, it chooses one or several discriminative rules from the HKB using the up-bottom search strategy and calculates classification accuracy. The experiments are carried out upon real electroencephalogram (EEG) recordings from five subjects and the results show the better performance of our method.

Index Terms—Electroencephalogram, feature extraction, EEG classification, discriminative rule, hierarchical knowledge base.

I. INTRODUCTION

The electroencephalogram (EEG) signal is believed to contain considerable information with regard to mental activity. By recording and analyzing EEG signals from scalp, more brain's mental activities patterns can be mined. One of important research fields bases on security testing such as guilty knowledge tests, deception detection, lie detection or concealed information tests [1-6]. The applications may involve criminal justice and counterterrorism. For example, P300-based EEG signal analysis can determine the presence or absence of specific details such as a crime or group affiliation stored in investigator's brain and this method has been suggested as an alternative approach for conventional polygraph [7].

Usually, application of pattern recognition approach on EEG signal includes following steps: signal acquisition, preprocessing, feature extraction, feature selection and classification. In this study, an important step is extraction of relevant features of EEG signals which containing necessary information of EEG signals. Recently, much attention has been paid to feature extraction methods for EEG. Polat [8] used discrete fourier transform (DFT) and Subasi [9] used discrete wavelet transform (DWT) as the extraction methods. Morphological features, frequency features and wavelet features were defined and evaluated for lie detection in [10].

II. SYSTEM ARCHITECTURE

In pattern classification applications, EEG-based BCI technology has huge applicable potentials since the corresponding recording method for EEG signals is relatively

Proc. 9th IEEE Int. Conf. on Cognitive Informatics (ICCI’10)
F. Sun, Y. Wang, J. Lu, B. Zhang, W. Kinsner & L.A. Zadeh (Eds.)
978-1-4244-8040-1/10/$26.00 ©2010 IEEE
353
convenient, inexpensive, harmless and possesses a high temporal resolution [16]. However, the demerit of recording EEG signal such as low signal-noise rate and low spatial resolution makes the improving of classification accuracy become a challenging aim for researchers. On the other hand, the information transfer rates is another key factor for evaluating the performance of a BCI system which means the processing time should be decreased. Generally, many methods report a good classification accuracy while also need to long time for training in order to get a test optimization. Inspiration from human cognition, fast or real-time decision-making with environment is more useful instead of exhaustive search for getting global optimization. From [17], we know that the local information is important, and from analysis of human cognition, hierarchical features extraction and classification [18-19] also investigated in our study.

A. Supervised learning

Learning set is fed into for extracting discriminative rules. It is well known that feature extraction is a crucial step affecting both performance and computation time of classification systems in pattern recognition applications. Nonparametric weighted feature extraction (NWFE) [14] is one of the useful tools for solving this problem. The main ideas of NWFE are putting different weights on every sample to compute the “weighted means” and compute the distance between samples and their weighted means as their “closeness” to boundary, and then define nonparametric between-class and within-class scatter matrices which put large weights on the samples close to the boundary and de-emphasize those samples far from the boundary. In order to reduce the space of features, autoregressive (AR) coefficients are computed by Burg’s method using points of the original EEG signals.

An autoregressive model with time-varying coefficients \( p \) is defined by

\[
Y_c(t) = \sum_{i=1}^{p} a_{i,c} Y_c(t-i),
\]

where \( a_{i,c} \) is the \( i \)-th coefficient of the AR model for channel \( c \). \( i = 1, \ldots, p \).

The nonparametric within-class scatter matrix and between-class scatter matrix for \( L \) classes is defined as same as follows:

\[
S_{W}^{NW} = \sum_{i=1}^{L} P \sum_{i=1}^{N_i} \frac{2}{N_i} (x_i^{(i)} - M_i(x_i^{(i)}))(x_i^{(i)} - M_i(x_i^{(i)}))^T,
\]

\[
S_{B}^{NW} = \sum_{i=1}^{L} P \sum_{i=1}^{N_i} \frac{2}{N_i} (x_i^{(i)} - M_i(x_i^{(i)}))(x_i^{(i)} - M_i(x_i^{(i)}))^T
\]

where \( P \) denotes the prior probability of class \( i \), \( N_i \) is training sample size of class \( i \), and \( x_i^{(i)} \) refers to the \( j \)-th sample from class \( i \). The \( \lambda_{ij} \) denotes the scatter matrix weight which is the function of \( x_i^{(i)} \) and \( M_j(x_i^{(i)}) \).

\[
\lambda_{ij}(x_i^{(i)}, M_j(x_i^{(i)})) = \frac{\text{dist}(x_i^{(i)}, M_j(x_i^{(i)}))}{\sum_{i=1}^{N_j} \text{dist}(x_i^{(i)}, M_j(x_i^{(i)}))} - 1,
\]

where \( \text{dist}(a, b) \) denotes the Euclidean distance from \( a \) to \( b \). If the distance between \( x_i^{(i)} \) and \( M_j(x_i^{(i)}) \) is small, then its weight will be close to 1; otherwise, \( \lambda_{ij} \) will be close to 0.

The sum of \( \lambda_{ij}(x_i^{(i)}, M_j(x_i^{(i)})) \) for class \( i = 1 \) and \( M_j(x_i^{(i)}) \) denotes the weighted mean of \( x_i^{(i)} \) in class \( j \) and defined as

\[
M_j(x_i^{(i)}) = \sum_{k=1}^{N} w_k^{(i,j)} x_k^{(j)},
\]

where

\[
w_k^{(i,j)} = \frac{\text{dist}(x_i^{(i)}, x_k^{(j)})}{\sum_{k=1}^{N} \text{dist}(x_i^{(i)}, x_k^{(j)})} - 1,
\]

is the weight of a sample \( x_k^{(j)} \) in class \( j \) corresponding to \( x_i^{(i)} \) in class \( i \). If the distance between \( x_i^{(i)} \) and \( x_k^{(j)} \) is small, then its weight \( w_k^{(i,j)} \) will be close to 1; otherwise, \( w_k^{(i,j)} \) will be close to 0. The sum of the \( w_k^{(i,j)} \) is 1.

The optimal features \( f \) are the \( f \) eigenvectors with largest \( f \) eigenvalues of the following matrix:

\[
(S_{W}^{NW})^{-1} S_{B}^{NW}.
\]

Two advantages of using the nonparametric scatter matrices mentioned in [14] are: First, they are generally of full rank. Second, the nonparametric nature of scatter matrices reduces the effects of outliers and works well even for non-normal datasets.

B. Hierarchical knowledge base (HKB) construction

HKB = \(<F, K, R>\), which is a 3-tuple where \( F \) is the feature set; \( K \) is the number of neighbors with the same classification. It corresponds background references called K-domain, and parameter \( R \) between 0 and 1 represents the ability of correct classifying under \( F \) and \( K \).

The construction algorithm is as follows.

1) Using equation (1) to extract AR coefficients for all learning samples.
2) Divide learning set \( LS \) into \( M \) subsets.
3) Select one of subsets to validation test set \( VTS \), and the others to validation learning set \( VLS \).
4) Validation feature set \( F = \emptyset \).
5) Select a column from \( VLS \) and add it to \( F \).
6) Divide \( VLS \) to Target class and Non-target class.
7) Calculate $S_w^N$ and $S_b^N$ for each class using equation (2) and equation (3), respectively.

8) Calculate the optimal features $f$ using equation (7).

9) Calculate class probability $R$ using KNN classifier (using the Euclidean distance) and record the corresponding best $K$ value. Thus, a piece of discriminative rule $DRI = <R, F, K>$ is obtained.

10) Repeat 5-9 until all feature columns from VLS been added to $F$.

11) Repeat 3-10 for all subsets.

12) Sort all $DRI$ to $HKB = <R, F, K>$ by $R$ in descending order.

13) Return $HKB$.

Note that the goal of this algorithm is not to learn a functional relation, but a set of discriminative rules, then through ranking the discriminative rules according to class probability $R$, a hierarchical knowledge base $HKB$ was constructed. The different size of feature set $F$ and the different $K$ value with the corresponding class probability $R$ make ways to scale of feature number for predicting the classification results.

C. Classification test

In order to evaluate the learning algorithm, a testing algorithm can be performed according to the following steps.

1) Load the $HKB$.

2) Select a testing sample from test set $TS$.

3) Test feature set $F' = \emptyset$.

4) Select a discriminative rule $DRI$ from $HKB$ using the up-bottom search strategy and add it to $F'$.

5) Calculate class probability $R'$ by using test feature set $F'$ and KNN classifier.

6) Record the best $R'$ and the corresponding class label of the sample.

7) Repeat 4-6 for all $DRI$.

8) Repeat 2-7 for all testing sample.

9) Calculate classification accuracy (Acc)

10) Return Acc.

To sum up, the main idea of this algorithm is to assign a label to a sample using a set of previously assessed discriminative rules on the best feature level.

III. EXPERIMENTS AND RESULTS

A. Materials

The data [20] used in this paper were recorded during a simple oddball test (an autobiographical paradigm). Each subject was asked to say 5 numbers, and one of them was his birth year (such as: 1980, 1981, 1983, 1986 and 1984). The subject didn't say his birth date till the end of experiment. The numbers were displayed to the subject randomly and with 30 repetitions (Totally 5*30 numbers displayed to the subject in each experiment). Each number displayed 1 second and between the numbers, the screen is empty for 2 second. The signals were recorded at frontal (Fz), central (Cz), and parietal (Pz) electrode positions from 10-20 international system.

All sites were referenced to the linked mastoids. Vertical EOG was also recorded for blink artifact detection. EEG signals were digitally sampled with 256 Hz frequency. To provide an idea of the nature of the signals distribution, Figure 1 gives an illustration of the EEG recorded including target and non-target stimulus on channel Pz from one of the subjects.

Figure 1. An illustration of the EEG signals recorded on channel Pz from one of the subjects before feature extraction. The time horizon is one second and thus includes 256 sampling points. The vertical axis is the voltage value in $\mu$V. Blue points represent target stimulus, i.e. subject’s birth year. Blue points are not the target stimulus.

B. Pre-processing

Before constructing a $HKB$ and the cross validation, several pre-processing operations were applied to the data. First, using MATLAB software offline, all data were digitally filtered in the 0.3-30 Hz range. This is the frequency range which is used typically in P300-based concealed information detection studies [21]. Second, normalization methods were applied to EEG signals for reducing large measurement noise or high variability. In our experiments, four different normalization methods were tested, respectively.

1) **Min-max normalization 1 (minmax[0+1])**

   \[ D' = \frac{D - \min(D)}{\max(D) - \min(D)}(U - L) + L \]  

   where $D'$ is the normalized data matrix, $D$ is the natural data matrix and $U$ and $L$ are the upper and lower normalization bound. The equation maps the data matrix into a bound between 0 and 1.

2) **Min-max normalization 2 (minmax[-1+1])**

   Above equation maps the data matrix into a bound between -1 and 1.
3) Nonlinear normalization (log)

\[ D' = \frac{\log(D - \min(D) + 1)}{\log(\max(D) - \min(D) + 1)}. \]  

(9)

4) Zero-mean normalization (z-score)

\[ D' = \frac{(D - \bar{D})}{\sigma}, \]  

(10)

where \( \bar{D} \) is the mean of the natural data matrix \( D \) and \( \sigma \) is the standard deviation of the same data matrix.

Third, the continuous EEG signals were separated into several epochs. Each epoch is a one-second long. Epochs containing eye blinks in EOG channel above 400\( \mu \)V were discarded.

C. Experimental results

All the experiments of this section were done over five subjects. In order to evaluate the performance of our proposed method, k-fold cross-validation method was employed. In the present work, we let \( k = 10 \), that is to say, the data from each subject was divided into ten subsets randomly, one of the subsets was used as the test set and the other nine subsets were put together to form a learning set. Then this procedure was repeated ten times and the average classification accuracy was computed. According to several experimental test runs, the order in autoregressive method was chosen at six.

In Table I, we report the average accuracy and runtimes\(^1\) for all of possible combinations of channels across five subjects. It shows that there is no apparent difference in runtimes. However, the channel Pz performs the best classification accuracy and the accuracy does not improve with the number of channels.

<table>
<thead>
<tr>
<th>Channel</th>
<th>Accuracy</th>
<th>Runtimes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fz</td>
<td>0.830</td>
<td>21.298</td>
</tr>
<tr>
<td>Pz</td>
<td>0.838</td>
<td>21.832</td>
</tr>
<tr>
<td>Cz</td>
<td>0.836</td>
<td>21.281</td>
</tr>
<tr>
<td>Fz, Pz</td>
<td>0.825</td>
<td>21.354</td>
</tr>
<tr>
<td>Fz, Cz</td>
<td>0.817</td>
<td>21.221</td>
</tr>
<tr>
<td>Pz, Cz</td>
<td>0.820</td>
<td>21.141</td>
</tr>
<tr>
<td>Fz, Pz, Cz</td>
<td>0.818</td>
<td>21.153</td>
</tr>
</tbody>
</table>

In Table II, the average classification accuracy obtained by using min-max normalization (minmax\([0+1]\), minmax\([-1+1]\)), nonlinear normalization (log) and zero-mean normalization (z-score), respectively. Based on the results, we can see that the minmax\([-1+1]\) and log methods are competitive in accuracy than other normalization methods in the present study.

<table>
<thead>
<tr>
<th>Subject</th>
<th>Minmax ([0+1])</th>
<th>Minmax ([-1+1])</th>
<th>log</th>
<th>z-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>s1</td>
<td>0.797</td>
<td>0.801</td>
<td>0.806</td>
<td>0.803</td>
</tr>
<tr>
<td>s2</td>
<td>0.810</td>
<td>0.833</td>
<td>0.811</td>
<td>0.815</td>
</tr>
<tr>
<td>s3</td>
<td>0.835</td>
<td>0.851</td>
<td>0.855</td>
<td>0.852</td>
</tr>
<tr>
<td>s4</td>
<td>0.869</td>
<td>0.858</td>
<td>0.864</td>
<td>0.835</td>
</tr>
<tr>
<td>s5</td>
<td>0.787</td>
<td>0.834</td>
<td>0.826</td>
<td>0.784</td>
</tr>
<tr>
<td>Average</td>
<td>0.819</td>
<td>0.835</td>
<td>0.832</td>
<td>0.818</td>
</tr>
</tbody>
</table>

TABLE III. FINAL CLASSIFICATION RESULTS

<table>
<thead>
<tr>
<th>Subject</th>
<th>No. of Samples</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>s1</td>
<td>144</td>
<td>0.936</td>
</tr>
<tr>
<td>s2</td>
<td>278</td>
<td>0.822</td>
</tr>
<tr>
<td>s3</td>
<td>245</td>
<td>0.850</td>
</tr>
<tr>
<td>s4</td>
<td>115</td>
<td>0.918</td>
</tr>
<tr>
<td>s5</td>
<td>145</td>
<td>0.929</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>0.891</td>
</tr>
</tbody>
</table>

IV. Conclusion

Finally, we select channel Pz as the target channel, minmax\([-1+1]\] as the normalization method for classification. We use 10-fold cross-validation and the results shows in table III. Subject 1 gives the best classification accuracy 93.6% and the whole average classification accuracy can achieve up to 89.1% which can be applied in BCI system in future. In addition, the total runtime is also acceptable in real-time applications.

<table>
<thead>
<tr>
<th>Subject</th>
<th>Normalization</th>
</tr>
</thead>
<tbody>
<tr>
<td>s1</td>
<td>Minmax ([0+1]), Minmax ([-1+1]), log, z-score</td>
</tr>
<tr>
<td>s2</td>
<td>Minmax ([0+1]), Minmax ([-1+1]), log, z-score</td>
</tr>
<tr>
<td>s3</td>
<td>Minmax ([0+1]), Minmax ([-1+1]), log, z-score</td>
</tr>
<tr>
<td>s4</td>
<td>Minmax ([0+1]), Minmax ([-1+1]), log, z-score</td>
</tr>
<tr>
<td>s5</td>
<td>Minmax ([0+1]), Minmax ([-1+1]), log, z-score</td>
</tr>
</tbody>
</table>

ACKNOWLEDGMENT

The authors gratefully acknowledge Dr. Vahid Abootalebi and the research center of intelligent signal processing (RCISP), Iran, for providing the data. The authors are supported from the National Natural Science Foundation of China (No. 60970061, No. 60775036, No. 60475019) and the Research Fund for the Doctoral Program of Higher Education (No. 20060247039).

\(1\) These runtimes were obtained using an Intel Pentium Dual CPU E2140 @ 1.60GHz, 1GB RAM.
REFERENCES


[19] Chulhee Lee. Feature extraction and classification algorithms for high-dimensional data. Purdue University West Lafayette, IN, USA. 1993.
