Ensemble Methods for the NTCIR-13 NAILS Task

Holly Hutson
Queensland University of Technology
Brisbane, QLD, Australia
h.hutson@qut.edu.au

Shlomo Geva
Queensland University of Technology
Brisbane, QLD, Australia
s.geva@qut.edu.au

Philipp Cimiano
Bielefeld University
Bielefeld, Germany
cimiano@cit-ec.uni-bielefeld.de

ABSTRACT

The QUT team participated in the NTCIR-13 Neurally Augmented Image Labeling Strategies (NAILS) task, this report describes our approach to solving the problem of developing machine learning models for classifying EEG data from an RSVP image search task. We explore the use of commonly used successful methodologies from the P300 Speller Paradigm, in particular the use of ensembles of support vector machines, and evaluate whether these methods still apply to the potentially more complex image search task.

Team Name

QUT

Subtasks

NAILS

Keywords

Brain-Computer Interface, Ensemble Learning, Machine Learning, Feature Selection, Signal Processing, Information Retrieval

1. INTRODUCTION

Brain computer interfaces (BCIs) initially developed as a potential assistive technology for users who may no longer be able to use traditional computer interface mechanisms, and in turn has seen a breadth of research into its applications in the medical field since the 1970's, with its primary use still theoretical. However, in recent years advances in technology have made Electroencephalography (EEG) methods for BCIs increasingly accessible to a broader range of researchers and users. This has led to some investigation of possible application to information retrieval tasks, such as labeling large datasets of multimedia images, identifying when a user's attention is piqued, or otherwise annotating media content. The NTCIR-13 NAILS Task [6] aims to evaluate machine learning pipelines for classifying image search tasks using a Rapid Serial Visual Presentation (RSVP) paradigm, providing a range of image-search tasks across a variety of users.

One of the most common BCIs is the P300 Speller paradigm, a system designed by Farwell and Donchin in 1988 [4] that allows users to spell messages via a BCI using a matrix of letters and numbers. This interface and its respective classification problems have seen a breadth of research since it's inception, including open competition datasets such as the BCI Competition III [2] that allow investigation of best practice for machine learning classification of EEG signals. Its use is based on identifying the P300 event related potential (ERP) [11], a measurable response in brain waves to surprising task-relevant events, including those elicited by visual and non-visual oddball tasks. While the RSVP image search paradigm is based on the same P300 response, given the previous lack of open datasets for this paradigm, it is unclear whether these traditional approaches to classifying P300 ERP signals are still as effective, particularly given the potential additional complexities afforded by realistic image processing tasks.

Based on a review of successful machine learning methodologies devised for the P300 Speller paradigm, we have identified ensembles of classifiers as a common, high performing method, and investigate their application to the image-search paradigms provided by the NAILS Task. Ensemble methods in machine learning involve the combination of multiple classifiers via a variety of methods such as bagging (averaging or voting), boosting, and stacking, to increase performance and reduce overfitting. Ensemble approaches of classifiers for P300 data are particularly common in competitions, with the top performing BCI Competition III Dataset II result using an ensemble of SVM's [12], as well as some other top performing classifiers on the competition set during and after the competition [3] [8] [10], typically via a bagging method with either voting or averaging. This process of bagging can help ensure classifiers don't overfit to the noise in the dataset, which is particularly prevalent in EEG datasets. Stacking of two classifiers also shows considerable success in an approach determined by Bigdely-Shamlo et al. [1] which stacks two Fisher linear discriminant analysis (FLDA) classifiers trained on two different subsets of features (time and time-frequency features) via a Naive Bayes classifier, and sees improvements in classification results over the two individual classifiers. Based on this, we investigate a bagging method motivated by the competition winners of the BCI Competition III Dataset II [12], and a stacking methodology inspired by the success of the approach designed for RSVP image search tasks by Bigdely-Shamlo [1].

2. METHODOLOGY

2.1 Pre-Processing

For a full discussion of the dataset, see the NAILS dataset description paper [5] and the overview of the task [6]. EEG data is inherently very high dimensional, and suffers from a low signal-to-noise ratio (SNR). In the case of the NAILS dataset, the dimensionality depends on the domain of the provided data. The time domain features, which is provided with 34 channels and time-samples at a rate of 50Hz between 0.5 seconds pre-stimulus to 1.5 seconds post-stimulus (34 x 100, 3400 features), and the two time-frequency domain features, both of which provide 32 channels, each with 6 frequencies (1 to 12 Hz in 2Hz steps) and either a mean or ratio of 2 second periods for each frequency in each channel between 0.5 seconds pre-stimulus to 1.5 seconds post stimulus (32 x 6 x 20, 3840 features). Therefore, it is vital to decrease the dimensionality by feature selection or reduction before classification, as well as using various pre-processing techniques to minimise noise.

Based on initial investigations into the best combination of filtering technique and down-sampling ratio to maximize the SNR for the time domain features provided, we apply a moving average filter, a low pass Finite Impulse Response filter to smooth data, with a window size of 5, and down-sample the data by a factor of 2 to a sampling rate of 25Hz. Based on research into the varying latency of the P300 signal, time samples are selected from 0ms to 800ms post-stimulus for all feature types.

2.2 Feature Selection

To further reduce the dimensionality of the data, a process of feature selection is undertaken using the so-called Recursive Channel Elimination (RCE) method. This method, initially proposed by Lal and Schröder [9] for a BCI dataset based on the motor imagery paradigm, modifies the traditional Recursive Feature Elimination method to eliminate channel by channel, rather than on an individual feature basis, both decreasing computation time and maintaining more neurophysiological interpretability. It has shown success across BCI paradigms, including those based on the P300 ERP. The original method involves repeatedly training a support vector machine (SVM), using the amount each feature influences the margin to determine feature importance, and removing n of the least important features. In the RCE version, this is modified to use the importance of a channel by computing the average score for each channel. Once a channel is eliminated, it is removed from the set of channels and the process is re-run with the smaller subset until a set number of channels remain, the number of which is treated as a hyper-parameter and selected via a grid search.

For the time-frequency domain features, this process is modified to sum all features across all frequencies for each channel before finding the average.

Despite the computational gains afforded by removing a channel at a time rather than a single feature, repeatedly running RCE as part of cross validation and a grid-search is computationally expensive. Rather than complete a run of the algorithm within each fold, we instead propose computing the cross-validated average ranking for each channel for each subject. This is done via ten-fold cross validation (see section 2.5 for more details on evaluation methodology), where a ranking for the channels is computed in each run by running the recursive channel elimination algorithm until one channel remains, and finally the ranks are summed for each channel and averaged over the ten runs. This allows us to determine a preset ranking of channels for each subject, and these rankings are then later used to keep only the top n channels determined by a grid-search. We determine a separate set of best channels for each domain of features.

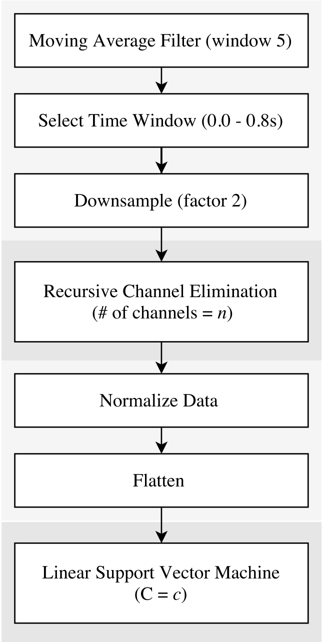
Preprocessing
Moving Average Filter (window 5)
Select Time Window (0.0 - 0.8s)
Downsample (factor 2)
Feature Selection
Recursive Channel Elimination (# of channels = n)
Normalize
Normalize Data
Flatten
Classifier
Linear Support Vector Machine (C = c)

Figure 1: Basic machine learning pipeline. c and n are determined by grid search.

For the results of these rankings, see section 3.1.

2.3 Classifier

The classifier used is a linear support vector machine (SVM). Before the selected features can be used for training the SVM algorithm, they are first normalized to have zero mean and unit variance, and each sample is flattened to the single-dimensional vector required for classification. To deal with the inherent class imbalance in this type of data, rather than using the commonly applied sampling methods, we instead add a penalty for the majority class to the parameter C that allows the minority class (the target images) to have a higher penalty for misclassification.

2.4 Ensemble Methods

In this paper we assess two ensemble methods, one using the bagging method, and one using a stacked ensemble. Bagging, also known as bootstrap aggregating, is an ensemble method where the original training dataset is randomly sampled with replacement to create multiple new training sets, each of which is used to train a classifier. These classifiers are then used to make predictions, and via either voting or averaging of the predictions, a final prediction is made. See figure 2 for a visualization of the model. Stacking is the process of combining several different classifiers, typically trained on the same data, and training a final aggregator model on the predictions from the earlier models, in the hopes of improving accuracy. Stacking of two classifiers using different feature domains (time and time-frequency features) has shown success in an approach determined by Bigdely-Shamlo [1], which stacked two FLDA classifiers and aggregated them via a Naive Bayes classifiers, showing improvements over the two individual models. Here we propose combining the three feature domains provided by the NAILS organisers, and training a final Naive Bayes model on a small left out

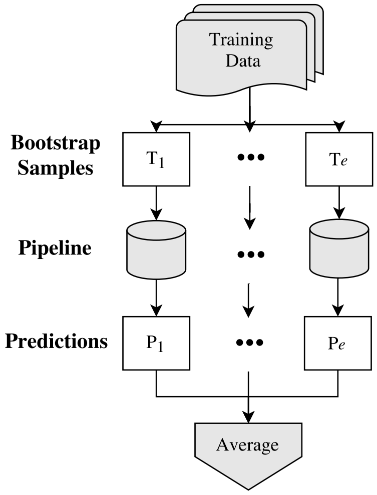

Figure 2: Bagging ensemble. e (number of models) and p (percent of data in each bootstrap sample) are determined by grid search.

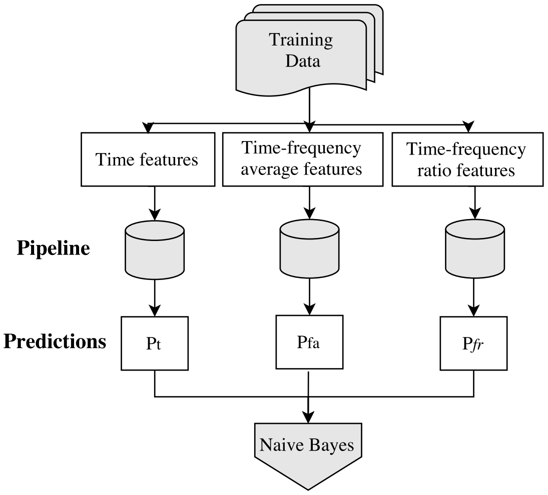

Figure 3: Stacked ensemble

set of data (10% in this case) to aggregate the predictions from these underlying models (see figure 3).

As well as the two ensemble methods, we test two baseline SVM classifiers for comparison with the same classification pipeline, with and without recursive channel elimination. For the version without RCE, six channels are selected based upon the recommendations determined by Krusienski [7], a commonly used combination of traditional P300 channels and posteriorly located channels. The final base pipeline can be seen in figure 1.

2.5 Model Evaluation

Methods are evaluated using ten-fold cross validation. K-fold cross validation involves splitting the dataset into ten disjoint parts, typically using a stratified method that ensures classes are distributed evenly between each fold. In this case, while the main class we are concerned with is the target or non-target class, we also stratify on the image search task being completed, as initial investigations suggested ERP responses may differ between the tasks. The model is then trained 10 times, with each fold being treated as the test set once, with the other nine folds used for training. The choice of hyper-parameters is evaluated on a randomly selected stratified validation set of 30% within each fold via a grid-search of the hyper parameters. Initial investigations suggested that using n-fold cross validation within the outer cross-validation loop for hyper-parameter selection had minor impacts on bias and variance, while resulting in an exponential increase in computation time, so a single run of grid-search is used here. Results are evaluated via balanced accuracy (BA), which is the arithmetic mean of the sensitivity and specificity.

3. RESULTS

3.1 Channel Selection Results

See figure 4 for the results of RCE to create channel rankings for each subject. The variation between subjects is clear here, although some posteriorly located channels that correspond to the commonly used Krusienski [7] set are fairly common across subjects. Interestingly, when we compare these time feature channel selections to the channels selected for the time-frequency domain, we can see some difference in top ranked channels (see figure 5).

3.2 Main

See table 1 for the full results, compared using 10-fold cross validation with the balanced accuracy reported as well as the standard deviation. The best performing model uses the bagging ensemble, which also results in a statistically significant decrease in variance in the folds, suggesting that it may be a more stable model. Interestingly, this model achieved a balanced accuracy score of 85.2% during the evaluation run on the test set left out by the NTCIR-13 NAILS organisers, a higher accuracy than estimated by our cross-validation procedure. This may suggest that the ensemble model is robust to overfitting the training set, compared to the stacked model, which achieved a result of 82.8% on the final test set, suggesting that it may have overfit to the training set. While we see some improvement using ensemble models for this task, the results do not appear to be as strong as similar methods achieved on similar competition datasets for the P300 Speller paradigm, suggesting that this may be a more complex classification problem than the traditional Speller system. However, further investigation is needed to determine this, and comparison to other team's results on the system should make the complexities of the problem clearer.

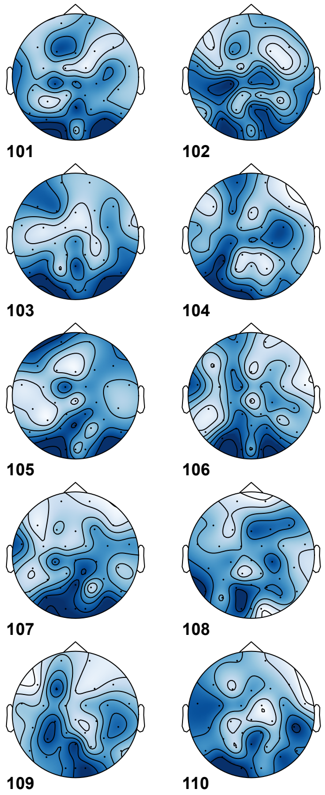

Figure 4: Channel rankings for each subject for time features, where darker colours correspond to a higher channel ranking (eliminated less often during RCE).

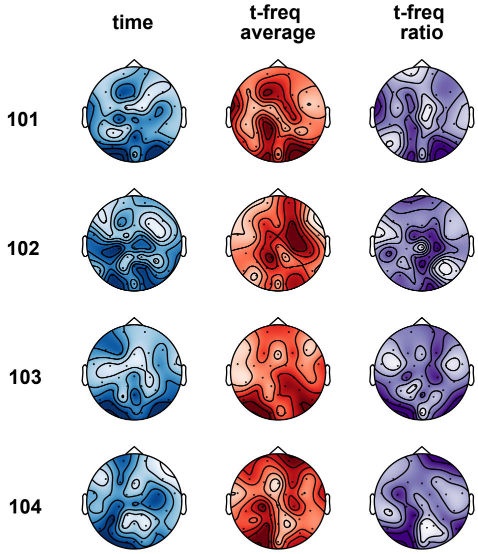

Figure 5: Channel rankings for the first four subjects for each feature type, where darker colours correspond to a higher channel ranking (eliminated less often during RCE).

Method	Subject										
	101	102	103	104	105	106	107	108	109	110	Avg.
Basic	76.5 ±3.2	85.1 ±7.2	84.3 ±7.5	84.6 ±5.0	87.4±4.0	79.2 ±4.5	89.2 ± 3.2	82.2 ±3.6	80.7 ±5.6	79.3 ±3.0	82.9 ±6.1
RCE	77.9 ±4.2	85.5 ±6.1	78.7 ±4.2	86.0 ±6.1	89.3 ±3.2	82.5 ±4.2	90.1 ±3.1	84.0 ±5.7	82.8 ±4.4	81.7 ±3.3	83.8 ±6.3
Bag	79.0 ±3.1	86.7 ±7.9	83.5 ±9.0	87.1 ±4.5	89.3 ±3.7	83.6 ±3.6	90.0 ±2.4	83.2 ±6.1	80.9 ±3.6	82.7 ±2.6	**84.6 ±5.9**
Stack	77.0 ±3.2	84.5 ±7.2	83.0 ± 8.3	85.1 ±4.1	89.0 ± 4.3	81.1 ±3.1	87.9 ±5.2	83.3 ±5.1	78.9 ±3.2	81.6 ±7.3	83.2 ± 6.3

Table 1: Balanced accuracy results for each subject for each model, with the standard deviation also provided.

4. REFERENCES

[1] N. Bigdely-Shamlo, A. Vankov, R. R. Ramirez, and S. Makeig. Brain activity-based image classification from rapid serial visual presentation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 16(5):432–441, 2008.

[2] B. Blankertz, K.-R. Muller, D. J. Krusienski, G. Schalk, J. R. Wolpaw, A. Schlogl, G. Pfurtscheller, J. R. Millan, M. Schroder, and N. Birbaumer. The bci competition iii: Validating alternative approaches to actual bci problems. *IEEE transactions on neural systems and rehabilitation engineering*, 14(2):153–159, 2006.

[3] H. Cecotti and A. Graser. Convolutional neural networks for p300 detection with application to brain-computer interfaces. *IEEE transactions on pattern analysis and machine intelligence*, 33(3):433–445, 2011.

[4] L. A. Farwell and E. Donchin. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and clinical Neurophysiology*, 70(6):510–523, 1988.

[5] G. Healy, Z. Wang, C. Gurrin, T. Ward, and A. F. Smeaton. An eeg image-search dataset: A first-of-its-kind in ir/iir. NeuroIIR 2017, March 2017.

[6] G. Healy, T. Ward, C. Gurrin, and A. F. Smeaton. Overview of ntcir-13 nails task. NTCIR-13, December 2017.

[7] D. J. Krusienski, E. W. Sellers, D. J. McFarland, T. M. Vaughan, and J. R. Wolpaw. Toward enhanced p300 speller performance. *Journal of neuroscience methods*, 167(1):15–21, 2008.

[8] B. Labbé, X. Tian, and A. Rakotomamonjy. Mlsp competition, 2010: Description of third place method. In *Machine Learning for Signal Processing (MLSP), 2010 IEEE International Workshop on*, pages 116–117. IEEE, 2010.

[9] T. N. Lal, M. Schroder, T. Hinterberger, J. Weston, M. Bogdan, N. Birbaumer, and B. Scholkopf. Support vector channel selection in bci. *IEEE transactions on biomedical engineering*, 51(6):1003–1010, 2004.

[10] Y. Li, H. Liu, and S. Wang. Exploiting eeg channel correlations in p300 speller paradigm for brain-computer interface. *IEICE TRANSACTIONS on Information and Systems*, 99(6):1653–1662, 2016.

[11] J. Polich. Updating p300: an integrative theory of p3a and p3b. *Clinical neurophysiology*, 118(10):2128–2148, 2007.

[12] A. Rakotomamonjy and V. Guigue. Bci competition iii: dataset ii-ensemble of svms for bci p300 speller. *IEEE transactions on biomedical engineering*, 55(3):1147–1154, 2008.