POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

Error-related Potentials Classification in Brain-Computer Interfaces: Validation of a novel LIME framework for signal explainability

Laurea Magistrale in Biomedical Engineering - Ingegneria Biomedica

Author: Silvia Villa

Advisor: Prof. Luca Mainardi

Co-advisor: Andrea Farabbi

Academic year: 2022-2023

1. Introduction

The increasing focus on the study of the EEG (electroencephalogram) signal has led to numerous advances in the field of neuroscience and to the development of new technologies. Notable among these are BCI (Brain-Computer Interfaces), which are used as a support system for people with physical disabilities. In several studies, it has been proved that the performance of these interfaces can be enhanced by introducing a correction system based on the recognition of error-related potentials (ErrPs), namely those potentials that are generated at the level of the Anterior Cingulate Cortex (ACC) and that can be observed in the subject's EEG as a result of discrepancies between the expected and the observed outcome. The focus of the thesis is on the development of a new framework of LIME, a well-known library for the explainability of deep learning models, for the extraction of meaningful features, aiming to build a much simpler and more transparent machine learning model. In doing so, we aim to fill the hole currently left by the library regarding the handling of signals that develop over time.

Throughout this summary, we will initially describe the dataset used and the processing to which these data were subjected. Next, we will focus on training a Convolutional Neural Network, called EEGNet, for the classification of EEG epochs. An explainability process will then be performed on this model for the extraction of meaningful features to add to the typical ErrP features and to additional features in the frequency domain. This explainability process will be carried out through the application of three different LIME frameworks: the original one, our new framework, and an intermediate version between them. Finally, the identified features will be used to train an LDA model, from which it will be possible to draw conclusions about the validity of the new LIME framework we have introduced.

2. Materials and Methods

A dataset containing the EEG signal recording of 6 subjects during the interaction with a BCI interface was used. The subjects monitored on a screen the movement of a cursor toward the target location. At each instant of time, a 20% probability that the cursor moved in the wrong direction was introduced, thus leading to the generation of an ErrP at the recognition of the erroneous movement. A data processing pipeline was defined and applied to the dataset in order to properly extract the epochs of interest needed to train the model. This processing consisted of re-referencing the data by Common Average Reference (CAR), filtering between 1 and 40 Hz to remove low and high-frequency noise, and downsampling the signal from 512 Hz to 64 Hz. At the end of this process, the epochs of interest were defined as the signal in the range [-650, ms +650 ms] with respect to the stimulus.

After this initial pipeline, a process known as subspace regularization was applied to the extracted EEG epochs to isolate the sources of the potentials from the background EEG. Subspace regularization is an approach in which the second-order statistics of the set of measurements are used to constitute the prior information model for evoked potential, proving to be well suited for their single-trial estimation [5].

2.1. EEGNet training

The obtained instances were divided into three subgroups, train, validation, and test sets, containing 70%, 15%, and 15% of the original examples, respectively. This division was performed maintaining the same distribution between the classes present in the original full dataset. To prevent the largest features from dominating the prediction, the obtained epochs were standardized to ensure that each of its features was centered around 0 and had a standard deviation equal to 1. The grand average (i.e., the mean of the epochs containing an ErrP) and the error-minus-correct signal, calculated as the difference between the mean of incorrect responses and the mean of correct responses, obtained as a result of this data preparation process, are shown in Figure 1. As expected, the waveform of ErrPs is clearly visible at the channels indicated in the literature (FCz and Cz).

Since the dataset was highly unbalanced, we had to design a balancing strategy to ensure a correct training of the selected Deep Learning model. This process was carried out by oversampling the minority class through SMOTE and undersampling the majority class, equalizing the number of instances belonging to the two classes without introducing too much synthetic data. The designated model was EEGNet, a compact CNN architecture based on depthwise and separable convolutions.

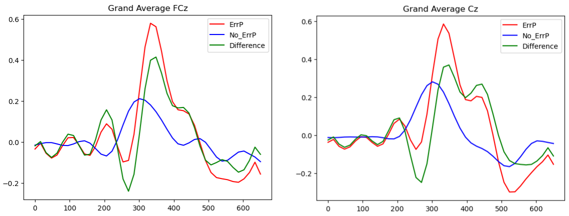

Figure 1: Visualization, at channels FCz and Cz, of the average of the ErrPs epochs (red), the average of the Non-ErrPs epochs (blue), and the difference between the two (green).

2.2. Features Extraction

In response to the need for transparent and interpretable models in the healthcare field, an explainability process was then applied to the EEGNet model. The goal was to extract meaningful features to train a much simpler and interpretable model. For this purpose, three types of features were considered: classical ErrP-related features, computed after subspace regularization; frequency features extracted from RNN; spatiotemporal features extracted from EEGNet.

2.2.1 Classical ErrP features

Typical features, both in the time and frequency domain, that characterize the waveforms of an ErrP, focusing on channels FCz and Cz. Following the literature [2], features over time focused on the values and latencies of the ErrP characteristic peaks. The features in the frequency domain, on the other hand, consisted of the average PSD in the delta and theta bands.

2.2.2 RNN frequency domain features

Additional frequency features are used to investigate the potential discriminative capabilities of other channels. This step was carried out by computing the PSD for each of the epochs obtained as a result of subspace regularization and by classifying them with an RNN architecture.

To gain a deeper insight into the reasons behind the decisions made by this model, an explainability process using LIME was applied. In this way, it was possible to derive the most dominant features in the frequency domain, each constituted by the combination of a channel and a particular frequency among those identified in the construction of the PSD. A validation of the consistency of the identified features was obtained by visualizing the distribution among the four main frequency bands of those features that led to a correct classification of the ErrPs epochs. In fact, as expected from the literature, the classification of ErrPs is mainly driven by frequencies in the delta and theta bands.

2.2.3 EEGNet features

The last group includes the features that most influenced the decisions made by EEGNet. Through this analysis, we can perform a study of the model both in space (the channel of interest) and in time (the identified time instant). These identified points were divided into two categories for feature extraction: points placed at a maximum or minimum of the signal and points placed at an intermediate position of the waveform. In the former case, the value of the positive/negative peak and its latency were extracted, in the latter the signal slope at that time instant.

2.3. LIME Frameworks

The EEGNet features were extracted through the application of the LIME library. This feature extraction process and the comparison of the results were performed with three different versions of LIME: the original framework (henceforth referred to as Original LIME), our modified framework (Kernel LIME), and an intermediate version between the two (Modified LIME).

2.3.1 Original LIME Framework

The basic idea behind LIME is to create an interpretable model that approximates the behavior of the original complex model by creating perturbed instances of the data (signal in our case) and checking how these variations modify the prediction of the model. Within the original LIME framework, these new examples are obtained by drawing, for each feature, a random value among the ones that the feature assumed in the various instances. This random sampling is done by assigning to each possible value a probability related to the frequency with which that value appeared among all the instances.

2.3.2 Revised LIME Frameworks

In Modified LIME a deliberate choice was made to refrain from discretization, given that the considered features do not represent independent entities but collectively shape the temporal and spatial trends within the signal. Additionally, instead of randomly perturbing the features, it was decided to nullify the values of the signal in specific positions. This version serves as an intermediary stage bridging the gap between the original LIME framework and our ultimate solution Kernel LIME. This is essential for assessing whether any subsequent enhancements introduced by our final framework result from the incorporation of the kernel or are primarily attributed to the strategic decision to refrain from discretization and nullify specific data points instead of randomly perturbing features. Finally in Kernel LIME, in addition to the changes already introduced in Modified LIME, it was decided not to modify the points individually but to also take into consideration the influence of surrounding samples. To achieve this, a Gaussian-shaped kernel was employed. The goal was to nullify a given instant while simultaneously smoothing the surrounding ones. This approach aligned with the assumption that each temporal instance is not an independent entity but also necessitates the evaluation of its surroundings to determine its dominance in the prediction. Each kernel encompassed, in addition to the central sample, also the three preceding and three subsequent samples, resulting in a time window of approximately 100 ms. Moreover, to maintain a balance between computational efficiency and a comprehensive assessment of positions, a 50% overlap between successive kernels was imposed. The kernel application involved zeroing the sample associated with the central position of the kernel and reducing the amplitudes of adjacent samples based on the kernel shape.

$$p = (1 - k) \cdot s \tag{1}$$

where s is the value assumed by the original signal at the given time instant, k is the value of the kernel, and p is the value assumed by the signal following the application of the kernel.

3. Results and Discussion

In this section are first reported the performance obtained with the original EEGNet model and with the derived LDA model, focusing on the results obtained with our final LIME framework. Next, these results are compared with those obtained with the other two LIME frameworks, allowing us to obtain a validation of the proposed new method.

3.1. EEGNet Evaluation

By first analyzing the performance obtained with EEGNet, it can be seen that the model tends more to classify correctly rather than incorrectly each of the two classes, with greater accuracy on the Non-ErrP class. *Specificity*, which measures the model's ability to correctly identify negative epochs, is of particular interest in all those applications where it is desired to minimize false positives. Indeed, in this application, it is critical to minimize false positives to avoid correcting the BCI system in the absence of an error. As desirable, this metric settled around 90% for each of the three sets. To take class imbalance into account, two other metrics were considered: *Balanced accuracy* and *F1 score*. Both of these metrics showed a decrease in validation and test sets compared to training. However, it can be seen that the performances are comparable between validation and test sets suggesting a constancy in the performance that the model could have on new data sets (see Table 1).

	Train	Validation	Test
Specificity	0.9207	0.8944	0.8815
Bal Acc	0.8919	0.8417	0.8246
F1 Score	0.8886	0.7185	0.6893

Table 1: Performace metrics for EEGNet

3.2. Validation of LIME features

In order to have a general overview of the important features of each of the two classes, the algorithm systematically scans through all correctly classified instances for each class (TP and TN) in the test set and records the distinctive features, identified by our LIME framework, that contribute to the correct predictions. Considering the 10 most significant features identified by LIME for each instance, it was recorded how many times a specific feature had been found to be important for the classification of that class. For each channel, the number of times LIME identified a sample belonging to it was calculated and then normalized for the total number of considered instances. For each of the two classes (TP and TN), the attention was focused on the five predominant channels:

- FCz, FC1, POz, Cz, Pz for the ErrP class (TP);
- Poz, Pz, FC1, FT7, F4 for Non-ErrP class (TN);

Looking at these results, the validity of the explainability carried out was supported by the channels returned. In fact, for the recognition of error potentials, both channels in which the recording of its waveform is expected, namely FCz and Cz, were identified. Notably, the individual points identified in these channels also reflect the values expected from the literature. In fact, the identified samples are concentrated in correspondence with the typical signal waveform, particularly in correspondence with the significant ErrP peaks. These points are then complemented by samples placed at the rising edge of the waveform, suggesting the importance of the signal slope at that instant. In addition, one channel found to be present in the prediction of both classes is POz. This channel is correlated with the subject's level of attention, thus suggesting that the generation of error-related potentials may also be influenced by the subject's level of attention and engagement.

3.3. A Much Simpler and Transparent Model

The guiding principle behind the choice of the Machine Learning model was linked to the prospective need for potentially applying this model in real-time applications. Therefore, it was imperative to restrict the selection to models that are notably simple and efficient. With this premise in mind, three potential models were considered: K-Nearest Neighbors (KNN), Decision Trees, and Linear Discriminant Analysis (LDA). The evaluation of these models involved the assessment of three distinct feature combinations, aimed at pinpointing the most effective model-feature pairing. These three combinations encompass:

1. Only the ErrP features;
2. A combination of ErrP features and LIME features;
3. The incorporation of all three feature types, with the addition of frequency features.

	1	**2**	**3**
KNN	59.5%	70.2%	69.1%
Decision Tree	63.9%	65.6%	61.9%
LDA	60.9%	78.3%	77.4%

Table 2: Balanced accuracy on the test set with various combinations of ML models and features: (1) represents only the ErrP features, (2) the combination of ErrP and LIME features, and (3) the combination of ErrP, LIME and Frequency

Considering the balanced accuracy as the main evaluation metric (Table 2), the LDA was selected as the final model since it combined the best performance with the shortest training and inference time. The results in Table 3 confirm the importance of the features extracted from LIME, as their introduction does not lead to a significant increase only in the balanced accuracy but in all key features.

	1	**2**	**3**
Accuracy	0.6377	0.8012	0.7899
Balanced Accuracy	0.6091	0.7832	0.7741
F1 Score	0.3881	0.6082	0.5932
Specificity	0.6576	0.8138	0.8008
AUC Score	0.61	0.78	0.77

Table 3: Performace metrics for LDA on the test set for the tree combination of features: (1) represents only the ErrP features, (2) the combination of ErrP and LIME features, and (3) the combination of ErrP, LIME and Frequency

3.4. Comparative Analysis of LIME framework

The goal of this final analysis was to validate the effectiveness of the modification we made to the LIME algorithms. As a first step, a qualitative comparison can be made by comparing the five main channels identified by each of the three frameworks for the classification of error potentials. By doing this, we obtained a first validation of the modification introduced: while Modified LIME and Kernel LIME identified, among the dominant channels, those expected from the literature (FCz and Cz), these were not found by Original LIME. In fact, in the latter, the focus is more on parietal channels (POz, P7, P10) rather than, as desirable, on the vertex. This initial conclusion allowed us to confirm how the decision to avoid the discretization of individual time points and to nullify those of interest rather than randomly modifying them could be a valid idea to enhance explainability applied to temporal signals. Moving then to a more quantitative analysis, the first approach was based on comparing the confusion matrices and key evaluation metrics obtained through the three different approaches. Taking advantage of the prior results listed in the previous section, the comparison was set up by training the LDA with the combination of typical ErrP features and features extracted from LIME, thus discarding the frequency features since they led to a slight decrease in performance. The examination of the relevant metrics and the confusion matrices, just like the channel analysis, supported our thesis: in fact, Kernel LIME was the one that exhibited the best performance on all the relevant metrics (Table 4) when compared to the other two versions. Similarly, from the confusion matrices (Table 5), it can be observed that our framework allowed for the best recognition of both classes.

	Original	**Mod**	**Kernel**
Accuracy	0.7795	0.7867	0.8012
Bal Acc	0.7545	0.7647	0.7832
F1 Score	0.5697	0.5830	0.6082
Specificity	0.7969	0.8021	0.8138
AUC Score	0.75	0.76	0.78

Table 4: Performace metrics for LDA on the test set for the three LIME frameworks.

Original LIME		
	Non ErrP	*ErrP*
Non ErrP	612	156
ErrP	57	141

Modified LIME		
	Non ErrP	*ErrP*
Non ErrP	616	152
ErrP	54	144

Kernel LIME		
	Non ErrP	*ErrP*
Non ErrP	625	143
ErrP	49	149

Table 5: Confusion matrices for Original LIME (top left), Modified LIME (top right), and Kernel LIME (bottom).

The definitive confirmation was provided by the analysis of the utility gain, a quantitative measure of the improvement introduced by an error correction system. If the gain is higher than 1, the correction system introduces a benefit; conversely, if the gain is lower than 1, the BCI system performs better without the correction system. As suggested in [4], the visualization of gain versus probability of the classifier may constitute the main method to quantitatively evaluate the improvement in BCI performance following the introduction of LIME's new framework. In Figure 2, each curve represents the utility gain introduced by the correction method based on each of the three LIME frameworks, while the dashed red line (utility gain = 1) indicates the limit for which the correction introduces a benefit, identifying when the gain becomes less than its useful value. It is evident that the model trained with features extracted from Kernel LIME was the one introducing the best gain if compared with the others. In fact, it allowed for the improvement of the model performance to values slightly exceeding 80%, while the other two frameworks were unable to reach this threshold. This suggested that the observed improvement is not solely due to the absence of discretization and the decision to nullify the points to analyze, but also to the introduction of the kernel to consider the surrounding signal.

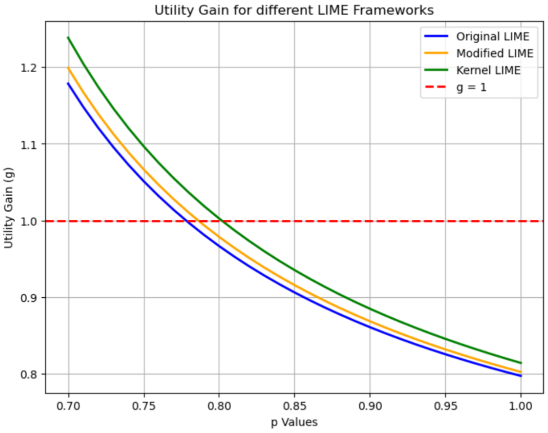

Figure 2: Utility gain introduced by the application of the Original LIME (in blue), Modified LIME (in orange) and Kernel LIME (in green) frameworks.

4. Conclusions

In conclusion, this work has proposed and developed a novel LIME framework specifically crafted for the analysis of signals developing in the time domain, a field where the original LIME version, tailored for images and tabular data, fell short. Our approach opens to new possibilities regarding the interpretability of deep learning models applied to signals in the time domain. The obtained results provide robust pieces of evidence of the validity of our variation, with improved performances and a higher utility gain with respect to the default LIME framework.

To understand how our work fits within the state of the art concerning the classification of error potentials, the obtained results were compared with two studies based on the same dataset, so that the results could be comparable. The approach used in [1] was based on the development of a simple deep Convolutional Neural Network, while in [3] deep neural networks were employed for constructing a Generative Adversarial Network (GAN) architecture. In both cases, the final result was reported in terms of accuracy.

	CNN	*GAN*	*LDA*
Acc.	87.94%	79.91%±2.43%.	80.12%

Table 6: Comparison of our results (last column) with literature values.

Analyzing these results, it was confirmed the consistency of our work with the state of the art in this research field. This is particularly noteworthy if we consider that our model is much easier and more interpretable when compared with the other models. Thus, the efficacy of our work not only translates into improvements in the classification performances but also, hopefully, in the opening of new perspectives for the use of more transparent and easily interpretable models in the field of BCI interfaces.

References

[1] Sunny Arokia Swamy Bellary and James M. Conrad. Classification of error related potentials using convolutional neural networks. 2019.

[2] Falkenstein et al. ERP components on reaction errors and their functional significance: A tutorial. *Biological Psychology*, 51, 2000.

[3] Gao et al. Improving Error Related Potential Classification by using Generative Adversarial Networks and Deep Convolutional Neural Networks. 2020.

[4] Andrea Farabbi, Vanessa Aloia, and Luca Mainardi. ARX-based EEG data balancing for error potential BCI. *Journal of Neural Engineering*, 19(3), 2022.

[5] Jari P et al Kaipio. Subspace regularization method for the single trial estimation of evoked potentials. 1999.