Neurocognitive Computing Semester Report: Reproducing State of the Art Experiments

Tiago Comassetto Fróes
froes@rhrk.uni-kl.de
Rheinland-Pfälzische Technische Universität Kaiserslautern-Landau
Kaiserslautern, Rheinland-Pfalz, Deutschland

Ana Luisa Teixeira Costa
tecosta@rhrk.uni-kl.de
Rheinland-Pfälzische Technische Universität Kaiserslautern-Landau
Kaiserslautern, Rheinland-Pfalz, Deutschland

ABSTRACT

This brief report has the purpose of being a walkthrough of how one should tackle the analysis of a real dataset. The chosen dataset is *EOG-based reading activity detection*, it records the head movement of the participants while they read three different text formats and also a not reading state. Data gathering, data pre-processing, experimental conditions, training, evaluation, and results will be described.

KEYWORDS

Machine Learning, Neurocognitive Computing, Random Forest, Gradient Boosting, Data Pre-processing

ACM Reference Format:
Tiago Comassetto Fróes and Ana Luisa Teixeira Costa. 2023. Neurocognitive Computing Semester Report: Reproducing State of the Art Experiments. In *Proceedings of 2023 (INF-78-51-M-6 RPTU Neurocognitive Computing).* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

For a computer science student who wants to venture into machine learning, taking the challenge of stepping outside of the toy dataset included in basic libraries and frameworks can be challenging. The introduction to academic papers and research experiments and the idea of not having a clear answer too can be confusing. Within these circumstances reproducing and adding some small changes to well-documented papers and experiments can be a good first step. Here we choose one classification, supervised learning, experiment documented on [1]. Using the dataset *EOG-based reading activity detection*, which was captured while the people were reading, we'll train a model to classify if the data represents one of four states: (0) not reading, (1) reading English, (2) reading horizontal Japanese text, and (3) reading vertical Japanese text.

INF-78-51-M-6 RPTU Neurocognitive Computing, Kaiserslautern, Deutschland,

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

2 METHODOLOGY

2.1 Dataset

The dataset consists of data collected by JINS MEME EOG Glasses () it represents eye-movement, head and body movements. As reported on [1], data was captured from ten Japanese students in the span of two days. In total 219 hours of raw data were collected. The data is structured in twenty csv files, one for each day of each participant.

2.2 Data pre-processing

When doing data analysis, the first step should be to get a clear vision of what the dataset provides since these are our instances and features, being able to answer questions such as: What is your dataset describing? How was the data collected? What information can we expect to extract from this data? What do the labels indicate? Are essential to get an understanding of what the data is picturing.

Looking at the raw data of our dataset with these questions in mind we get that our dataset consists of two time-related variables, the label, and eight features depicting head and eye movement. These eight features and label are going to feed our model. As reported on [1] we apply feature extraction to our features, the mean of each feature is calculated in a 30-second window, and after that, feature-wise normalization is applied.No new features deriving from the original ones were created. Pre-processing was done using *pandas* and *numpy* frameworks.

2.3 Experimental Conditions

Due to computational and time restraints, we focus on only one experiment reported on [1] that is, *leave-one-day-out cross-validation* and EN vs. JH vs. JV classification task. Since the not reading state is the most abundant in the dataset, consisting of 8799 minutes, 66.2% of all our data we drop every *not reading* instance on our data and split it into train and test sets.

2.4 Training

Within the spirit to try different approaches than the one in the original paper, we choose to explore tree-based models to see how the results would differ from the support vector machine. For this supervised learning task we choose two different models to train, *Random Forest Classifier* and *Gradient Boosting Classifier* were the chosen ones. The training was done using the *Scikit Learn* framework within the *Google Colab* platform. For *Random Forest Classifier* all parameters were left as the default. For *Gradient Boosting Classifier* two separate models were trained, one using the default parameters and a second one with the following changes in parameters: *learning_rate = 0.2, n_estimator = 50, max_features = 'sqrt', warm_start = True, subsample = 0.8.* No *cross-validation* was done to choose these changes, the purpose is solely to see how different models behave.

2.5 Evaluation

For evaluation, we choose three different metrics, *accuracy*, *F1 score*, and confusion matrix. *Accuracy* is an intuitive way to evaluate the model also, it serves as a way to compare the results to those on the original paper. The *F1 score* is the next step in the evaluation, it provides the harmonic mean of *precision* and *recall.* Next, we provide the *c=Confusion Matrix*, all the previous metrics can be derived from here and it provides an intuitive view of how well the model performs in each class.

3 RESULTS AND DISCUSSION

The classification results for our experiments, *leave-one-day-out corss validation* using only data when participants are reading one of the three text formats, are shown in Table 1.

Table 1: Accuracy

Method	Accuracy
Random Forrest	0.43
Gradient Boosting (Default)	0.40
Gradient Boosting (Tweaked)	0.43

As we can see our attempts to get better results with different models were unuseful since as related in the original paper accuracy for this particular experiment was 0.46. One surprise is that the "simplest" learning algorithm, *Random Forest* with default parameters, yielded the best result. But as seen with the results between the other two models, tweaking parameters can improve the results, in this case, an increase of 2%, each is considerable.

Table 2 presents the *F1 scores* for all different models.

Table 2: F1 Score

Method	EN	JH	JV
Random Forrest	0.44	0.40	0.46
Gradient Boosting (Default)	0.37	0.44	0.46
Gradient Boosting (Tweaked)	0.35	0.45	0.47

Table 3, Table 4, and Table 5 show the *Confusion Matrix* for each model. These matrices are normalized.

Table 3: Confusion Matrix for Random Forrest

	EN	JH	JV
EN	0.45	0.29	0.24
JH	0.33	0.41	0.25
JV	0.28	0.27	0.44

Table 4: Confusion Matrix for Gradient Boosting (Default)

	EN	JH	JV
EN	0.3507	0.3550	0.2943
JH	0.2490	0.4763	0.2747
JV	0.2683	0.2650	0.4668

Table 5: Confusion Matrix for Gradient Boosting (Tweaked)

	EN	JH	JV
EN	0.3069	0.3826	0.3104
JH	0.1996	0.5142	0.2862
JV	0.2414	0.2698	0.4888

Even though we did not get a better result than the one reported in the original paper now we have an understanding of the task and can advance in some directions like finding a better way to extract features from the dataset, maybe using deep learning methods as convolutions and that using different architectures and fine-tuning it can yield better results.

4 CONCLUSION

Reproducing experiments reported on published papers can be a really challenging task and a great learning process, thanks to open-source frameworks and publicly available datasets this activity is reachable to most students but there's a line where computational power makes some of it inaccessible. In this report, we walk through one example of it. Reaching your own results through your own code, analyses, and pipeline gives you a better intuition on how research is done, this is one of the pillars of contemporary machine learning research.

5 REFERENCES

[1] Shoya Ishimaru, Takanori Maruichi, Manuel Landsmann, Koichi Kise, and Andreas Dengel. 2019. Electrooculography dataset for reading detection in the wild. In Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers (UbiComp/ISWC '19 Adjunct). Association for Computing Machinery, New York, NY, USA, 85–88. https://doi.org/10.1145/3341162.3343812