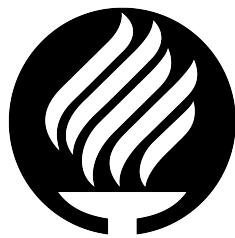


INSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES DE
MONTERREY
CAMPUS QUERÉTARO
COMPUTER AND MECHATRONICS DEPARTMENT



Tecnológico de Monterrey

Individual Classification through Recurrent Neural Networks: Leveraging EEG Signals for Accurate Profiling

by

Bernardo Estrada Fuentes

Integrative Project for the Development of Business Solutions

*Computer and Systems
Engineering*

Avdisor: Jose Antonio Cantoral-Ceballos, Ph.D.,

Santiago de Querétaro, Querétaro, México
20/11/2023

Abstract

This integrative project explores the application of Recurrent Neural Networks (RNNs), specifically the Gated Recurrent Unit (GRU), in the realm of individual classification using electroencephalogram (EEG) signals. Brainwave patterns, indicative of physical and emotional states, have garnered increasing interest across diverse fields, including epilepsy detection and neuro-prosthetic interfaces. However, challenges arise from low signal-to-noise ratios (SNRs) in multi-electrode setups, prompting the need for non-linear techniques.

The study utilizes the "EEG Motor Movement/Imagery Dataset" and outlines a comprehensive methodology for data selection, preprocessing, and GRU model implementation. Initial challenges, such as low model accuracy, are addressed through iterative adjustments, including data transformation, noise reduction through channel selection, and model fine-tuning.

Results indicate significant improvements in model performance, with a cumulative train accuracy of 98% and 95%, and a test accuracy of 78% and 74% on two- and ten-subject datasets, respectively. The research underscores the importance of meticulous fine-tuning efforts and strategic exploration of hyperparameters.

Areas of opportunity for future exploration are identified, including the observation of test accuracy plateauing and the potential impact of EEG channels. Additionally, refining classification into two categories is suggested for enhanced practicality in individual identification.

In conclusion, this integrative project contributes to understanding the application of GRUs in individual classification through EEG signals. The outcomes offer valuable insights for future research, laying the groundwork for continued refinement and the development of robust biometric solutions.

Contents

Abstract	i
Abbreviations	iii
1 Introduction and State of the Art	1
2 Problem and Proposed Solution	3
3 Data and Methodology	4
3.1 Data Selection	4
3.2 Data Transformation and Prepossessing	5
3.3 Initial Model Implementation	5
3.4 Data Tagging and DataLoader	6
3.5 Data Transformation for Improved Training	6
3.6 In-Depth Analysis	6
3.7 Reducing Noise through Channel Selection	7
3.8 Model Fine-Tuning	7
4 Results and Discussion	10
4.1 Results and Interpretation	10
4.2 Areas of Opportunity	10
5 Conclusions	12
A Relevant Code	14
Bibliography	17

Abbreviations

CNN	Convolutional Neural Network
EEG	Electroencephalogram / Electroencephalography
GRU	Gated Recurrent Unit (RNN)
LR	Learning Rate
RNN	Recurrent Neural Network
SNR	Signal to Noise Ratio

Chapter 1

Introduction and State of the Art

In recent years, the exploration of brainwave patterns has gained substantial traction across diverse fields. This surge in interest stems from the growing demand for sophisticated analytics across applications, including epilepsy detection, neuro-prosthetic interfaces, and even neural image reconstruction[1].

Brainwaves, categorized by oscillation frequencies, hold promise as indicators of physical and emotional states. However, extracting prevalent brainwave types using multi-electrode setups is a challenge due to low signal-to-noise ratios (SNRs). Linear methods like FFT struggle with noisy signals[1], prompting the need for non-linear techniques. Conventional neural networks require extracted features for optimal functioning. Current efforts emphasize feature extraction methods like Wavelet Transform and Power Spectral Density, but these may introduce bias[1].

The Gated Recurrent Unit (GRU), a variant of Recurrent Neural Networks (RNNs), is renowned for its accomplishments in various domains, particularly in handling sequential data. GRUs exhibit remarkable capabilities in capturing intricate spatiotemporal patterns while demonstrating resilience to distortions, rendering them particularly well-suited for the analysis of data with various levels of Signal-to-Noise Ratio (SNR).

The analysis of brainwave patterns has gained momentum in various disciplines due to their potential to reveal insights into cognitive and emotional states[2]. However, these patterns are often obscured by noise, posing a significant challenge for accurate interpretation and classification[3]. Also, these patterns are mostly used to identify actions or discrepancies happening in the brain[4], not for profiling or classifying individuals or used as a biometric parameter.

Harnessing brainwave patterns for individual identification offers a novel and secure biometric solution with applications ranging from authentication systems to personalized medical interventions[5]. Traditional linear methods, though effective in controlled environments, struggle with real-world noise, hindering their practicality[4]. RNNs offer a promising avenue to overcome these challenges, but their performance in extracting individual-specific features from noisy brainwave data requires in-depth exploration[3].

Chapter 2

Problem and Proposed Solution

Chapter 3

Data and Methodology

In this section, we will describe the data used for the study, the preprocessing steps, and the methodology followed to implement and fine-tune the Gated Recurrent Unit (GRU) for brainwave classification.

3.1 Data Selection

The dataset used for this study is the "EEG Motor Movement/Imagery Dataset" [6] which comprises over 1500 one- and two-minute EEG recordings from 109 volunteers. These recordings encompass various tasks, including baseline measurements, tasks involving motor movement (real and imagined), and subsequent repetitions of these tasks. Each task has annotations (events) representing different conditions and actions.

The tasks performed are as follows:

- Baseline with eyes open.
- Baseline with eyes closed.
- Task 1: Open and close left or right fist.
- Task 2: Imagine opening and closing left or right fist.
- Task 3: Open and close both fists or both feet.
- Task 4: Imagine opening and closing both fists or both feet.

Annotations for events (T_0 , T_1 , and T_2) indicate the onset of motion for left or right fist, both fists, and both feet, corresponding to real or imagined actions, depending on the task and run number.

3.2 Data Transformation and Prepossessing

To facilitate the classification task, the dataset was transformed as follows:

- Each task was cropped into its annotations, resulting in multiple files, each representing all instances of a specific event within a run.
- Each event file contains a 3-dimensional vector of size $(8, 720, 64)$, representing 8 events (15 for rest events), each with 720 samples (at 160Hz, equivalent to 4.5 seconds), and 64 channels per sample. This can be seen on Fig. 3.1
- Samples were padded using the wrap method to ensure uniform sizes.

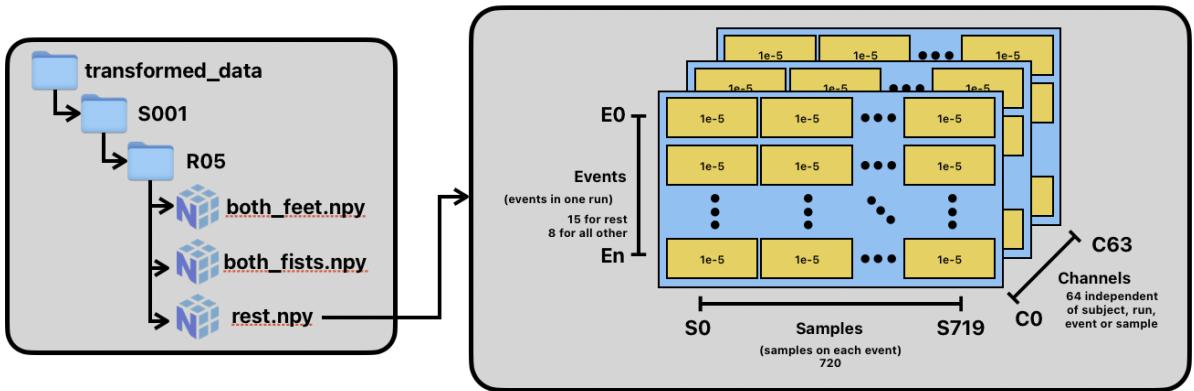


FIGURE 3.1: Data structure after prepossessing

3.3 Initial Model Implementation

An initial GRU model was implemented to begin the training process. Multiple runs were performed with default hyper-parameters to assess the model's performance.

3.4 Data Tagging and DataLoader

Tags were generated to match the sizes of each subject's events, ensuring correct classification. A DataLoader was implemented to shuffle the training data for model training.

3.5 Data Transformation for Improved Training

As the initial GRU model was not yielding satisfactory results, the following data transformations were made:

- Events were flattened, consolidating them into a single dataset with 1800 events (10 subjects with 180 events each).
- Each event was transformed to have fewer GRU cells, joining samples together to achieve 10 cells with an input size of 4,608. See Fig. 3.2

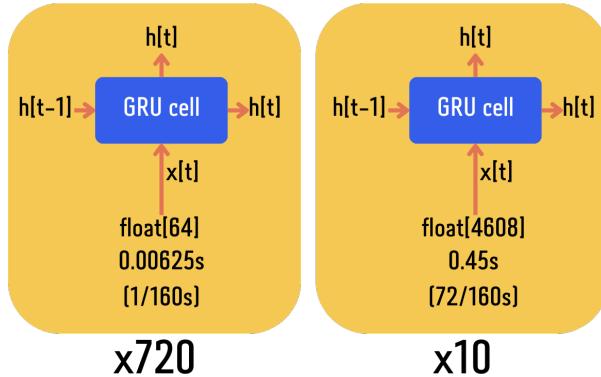


FIGURE 3.2: Increase context for each GRU cell and reduce count of GRU cells. Before (left) and after (right)

3.6 In-Depth Analysis

Despite the above changes, the model's accuracy remained low, and in-depth analysis was conducted:

- Data was reviewed to identify anomalies, but none were found.

- The problem was reduced to comparing events of a single subject, which should theoretically yield better results. However, accuracy remained below random guessing (50%)

3.7 Reducing Noise through Channel Selection

Upon consultation with thesis advisor, Jose Antonio Cantoral-Ceballos, Ph.D., the following change was implemented to reduce noise in the system:

- A selection of the central group of 9 nodes ($FC[1, Z, 2]$, $C[1, Z, 2]$, $CP[1, Z, 2]$) was made.
See Fig. 3.3

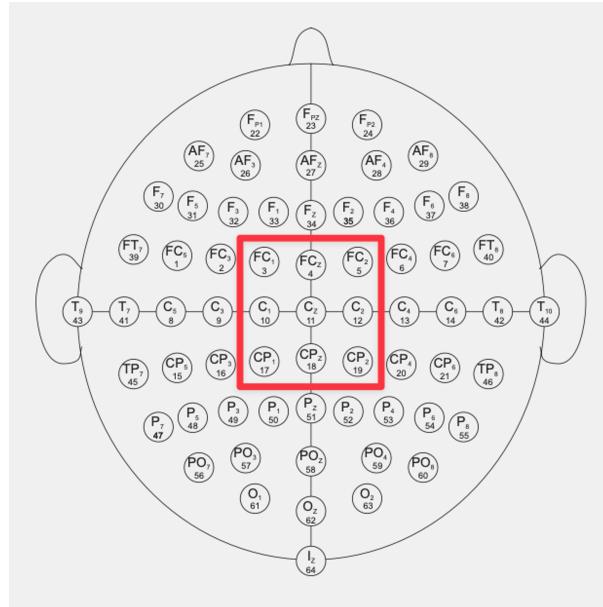


FIGURE 3.3: Graphic shows all available EEG channels. Red square indicates selection of selected central nodes

3.8 Model Fine-Tuning

Additional tests were conducted with the modified dataset and GRU model:

- Different LRs were tested to find optimal option to train with. The best LRs in training showed to be $1e-2$, $8e-3$ and $5e-3$ as shown on Fig. 3.4

Later with a test sample these yielded the following accuracies:

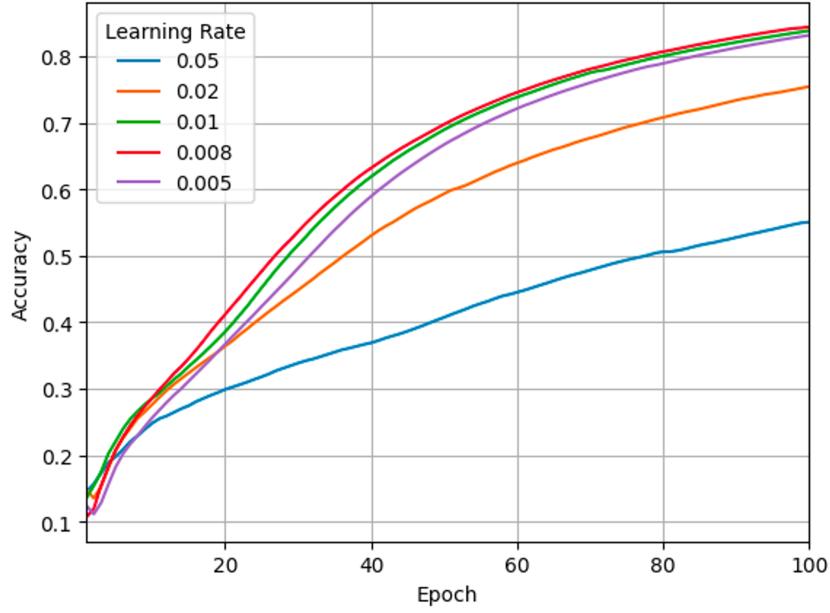


FIGURE 3.4

- $1e-2 \rightarrow 62.3$
- $8e-3 \rightarrow 54.7$
- $5e-3 \rightarrow 54.5$
- Implementing OneCycleLR scheduler [7] with various values was tested and showed small but consistent positive results of +5% in test accuracy (Figs. 3.5a and 3.5b).

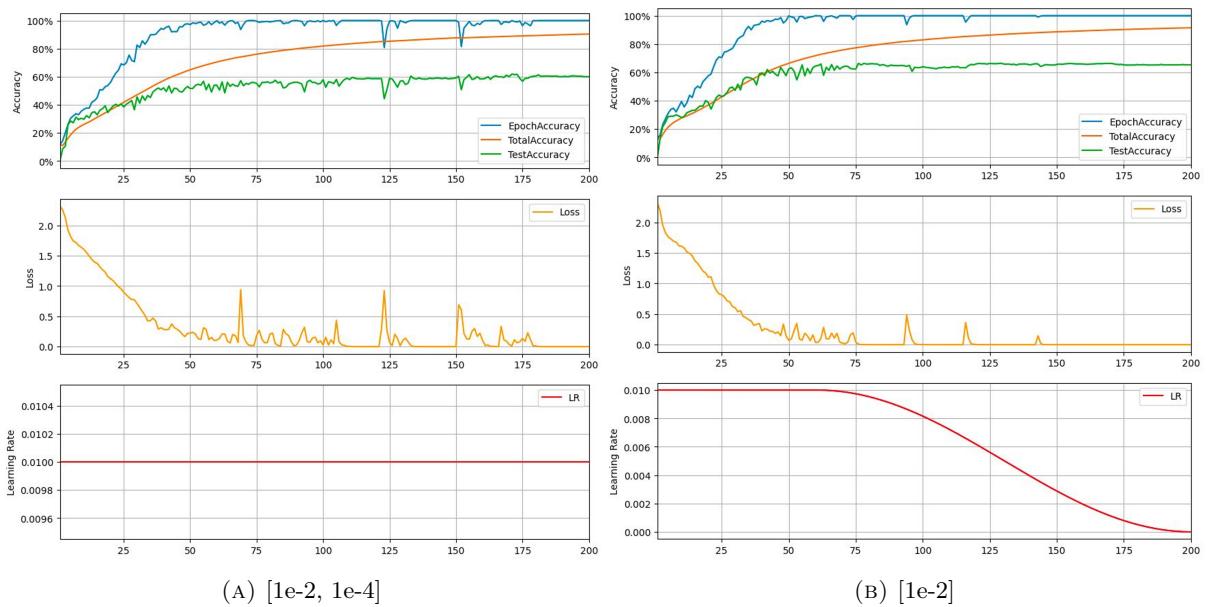


FIGURE 3.5: OneCycleLR comparisons

- Finding an appropriate Label Smoothing in the Cross entropy loss function [8]. This showed a big increase in test accuracy (From 40% to 80%) (Figs. 3.6a and 3.6b)

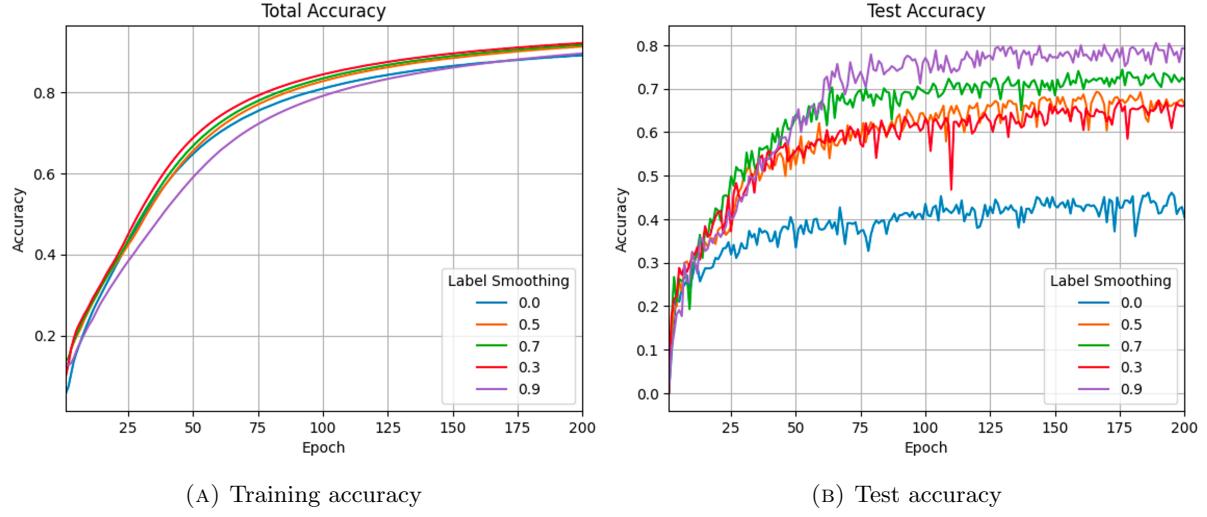
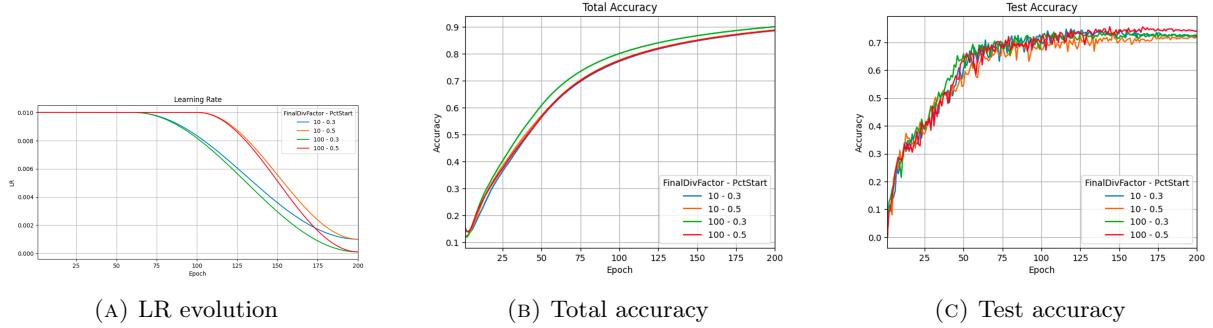


FIGURE 3.6: Runs with various values of label smoothing

- Joining OneCycleLR and Label Smoothing and testing with various parameters to find the best output. (Figs. 3.7a, 3.7b, 3.7c)



Chapter 4

Results and Discussion

4.1 Results and Interpretation

Further tests with the final shown configuration were run. This tests achieved a cumulative train accuracy of 98% and 95% and a test accuracy of 78% and 74% on a two and ten subject dataset respectively.

The results indicate significant improvements in model performance after reducing the number of channels and selecting specific nodes for analysis. The study also demonstrates the importance of fine-tuning the model as this provided an over 20% increase in test accuracy after carefully comparing hyper-parameters.

Preprocessing and finding an efficient data structure for the use case was also a big impact in the process as it helped iterate over different inputs, outcomes and selection of data during the research.

4.2 Areas of Opportunity

Some areas of opportunity on this research are as follows:

- Test accuracy stagnates after around 120 epochs. This can be either due to the amount of training data or the number of EEG channels being too low.

- Low accuracy yielded by the last tests of the model are not suitable for biometric authentication
- Classification into N categories may be replaced with 2 categories (Fig. 4.1). This has the potential to make the model more suitable for identifying a single person.

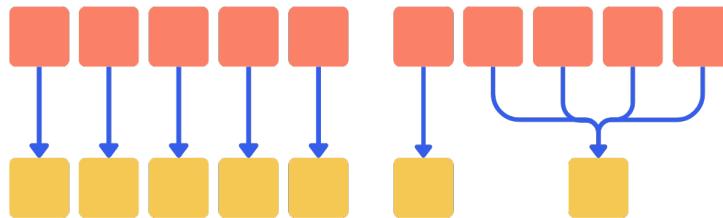


FIGURE 4.1: N to N classification (left) and N to 2 classification (right)

Chapter 5

Conclusions

In conclusion, this research has resulted in some level of progress in the exploration of brainwave patterns for individual classification through the implementation of Recurrent Neural Networks (RNNs) and the Gated Recurrent Unit (GRU). The challenges associated with low Signal-to-Noise Ratios (SNRs) in multi-electrode setups were acknowledged, prompting the adoption of non-linear techniques, specifically GRUs.

The methodology employed for data selection, preprocessing, and the iterative implementation and fine-tuning of the model has been thoroughly detailed. Key adjustments, including noise reduction through channel selection and the introduction of the OneCycleLR scheduler and Label Smoothing, were instrumental in overcoming obstacles and elevating model accuracy.

The results and discussion section revealed commendable advancements in model performance, with a cumulative train accuracy of 98% and 95%, and a test accuracy of 78% and 74% on two- and ten-subject datasets, respectively. These outcomes underscore the significance of meticulous fine-tuning efforts and the strategic exploration of hyperparameters.

Areas for future exploration have been identified, particularly the observation of test accuracy plateauing after a specific number of epochs. Further investigation into training data nuances and the potential impact of EEG channels is warranted. Additionally, the suggestion to refine classification into two categories instead of N categories offers potential avenues for enhanced practicality in individual identification.

In summary, the integrative project has significantly contributed to the understanding of applying GRUs to individual classification through EEG signals. The outcomes provide valuable

insights for future research, laying the groundwork for continued refinement and ensuring accurate profiling and the development of robust biometric solutions in the future.

Appendix A

Relevant Code

color

listings

```
1 def getTaskFromRun(run: int) -> str:
2     if run == 1: return tasks["TOA"]
3     if run == 2: return tasks["TOB"]
4     try: return tasks[f"T{(run-2)%4}"]
5     except: return tasks["na"]
6
7 def transformRunData(n_subject, n_run):
8     # Get data and standarize
9     f_name = mne.datasets.eegbci.load_data(n_subject, n_run, path=
10        MNE_PATH)[0]
11     raw = mne.io.read_raw_edf(f_name, preload=True).load_data()
12     mne.datasets.eegbci.standardize(raw)
13
14     # Rename and get events
15     raw.annotations.rename(getAnnotationMapping(n_run))
16     events_from_annot, event_dict = mne.events_from_annotations(raw
17         )
```

```

18     # Set second (end) value on annotations to first (start) value
19     # of next item -1
20
21     for i in range(len(events_from_annot)):
22         if i+1<len(events_from_annot):
23             events_from_annot[i][1] = events_from_annot[i+1][0]-1
24         else:
25             events_from_annot[i, 1] = len(raw)
26
27
28     # Get data in np arr
29     time_data = raw.get_data().transpose()
30
31
32     def getPaddedRange(a, b):
33         start = a
34         end = b if b-a<720 else a+720
35         ev = time_data[start:end]
36         res = np.pad(ev, ((0, 720-len(ev)), (0,0)), 'wrap')
37         return res
38
39
40     # Save all events
41     for ev in event_dict:
42         events = mne.event.pick_events(events_from_annot, include=[
43             event_dict[ev]])
44
45         data = np.array([getPaddedRange(e[0], e[1])[:] for e in events])
46
47         path = f'data/S{n_subject:03d}/R{n_run:02d}'
48         os.makedirs(path, exist_ok=True)
49         np.save(f'{path}/{ev}', data)

```

LISTING A.1: Data transformation function

```

1 class GRU(nn.Module):
2
2     def __init__(self, params):
3
3         super(GRU, self).__init__()
4
4
5         self.gru_layer = nn.GRU(

```

```
6     params["input_size"] ,  
7     params["hidden_size"] ,  
8     params["num_layers"] ,  
9     bias=params["gru"]["bias"] ,  
10    batch_first=True ,  
11    bidirectional=params["gru"]["bidirectional"] ,  
12    dropout=params["gru"]["dropout"] ,  
13    device=device  
14 )  
15     self.out = nn.Linear(params["hidden_size"]*(2 if params["gru"]  
16 ["bidirectional"] else 1) , params["num_classes"])  
17  
18 def forward(self , x):  
19     r_out , t = self.gru_layer(x , None)  
20     test_output = self.out(r_out[:, -1, :])  
21     return test_output
```

LISTING A.2: GRU definition

The entirety of the code can be found in the GitHub repository:

github.com/BernardoEstrada/eeg-profiling

Bibliography

- [1] S. R. Joshi, D. B. Headley, K. C. Ho, D. Pare, and S. S. Nair, “Classification of brainwaves using convolutional neural network,” in *2019 27TH EUROPEAN SIGNAL PROCESSING CONFERENCE (EUSIPCO)*, European Signal Processing Conference, European Assoc Signal Proc; Univ Coruna; IEEE Signal Proc Soc; Xunta Galicia, Conselleria Cultura Educac, Oredenac Univ; Turismo A Coruna, Oficina Informac Tuursimo A Coruna; Off Navl Res Global; Xunta Galicia, Centro Investigac TIC; MathWorks; Natl Sci Fdn; Fundac Circulo Tecnologias Defensa Seguridad; Catedra UDC Everis Transformac Digital; Colexico Oficial Enxeneiros Telecomunicac Galicia; Inst Tecnologico Galicia, 2019. 27th European Signal Processing Conference (EUSIPCO), A Coruna, SPAIN, SEP 02-06, 2019.
- [2] P. Nguyen, D. Tran, X. Huang, and W. Ma, “Age and gender classification using eeg paralinguistic features,” in *2013 6TH INTERNATIONAL IEEE/EMBS CONFERENCE ON NEURAL ENGINEERING (NER)*, International IEEE EMBS Conference on Neural Engineering, pp. 1295–1298, IEEE EMBS; Battelle; Brain Vis LLC; Cortech Solut; Univ Minnesota, IGERT Program, Syst Neuroengineering; IOP Publishing; Ripple; Univ Minnesota, Inst Engn Med; Plexon, 2013. 6th International IEEE EMBS Conference on Neural Engineering (NER), San Diego, CA, NOV 06-08, 2013.
- [3] J. Teo, C. L. Hou, and J. Mountstephens, “Deep learning for eeg-based preference classification,” in *2ND INTERNATIONAL CONFERENCE ON APPLIED SCIENCE AND TECHNOLOGY 2017 (ICAST'17)* (F. Nifa, C. Lin, and A. Hussain, eds.), vol. 1891 of *AIP Conference Proceedings*, Acad Res Soc Malaysia, 2017. 2nd International Conference on Applied Science and Technology (ICAST), MALAYSIA, APR 03-05, 2017.
- [4] E. A. Oral, I. Y. Ozbek, and M. M. Codur, “Gender clasification based on single channel

- eeg signal,” in *2017 INTERNATIONAL ARTIFICIAL INTELLIGENCE AND DATA PROCESSING SYMPOSIUM (IDAP)*, IEEE Turkey Sect; Anatolian Sci, 2017. 2017 International Artificial Intelligence and Data Processing Symposium (IDAP), Malatya, TURKEY, SEP 16-17, 2017.
- [5] M. DelPozo-Banos, C. M. Travieso, C. T. Weidemann, and J. B. Alonso, “Eeg biometric identification: a thorough exploration of the time-frequency domain,” *JOURNAL OF NEURAL ENGINEERING*, vol. 12, OCT 2015.
 - [6] G. Schalk, D. J. McFarland, T. Hinterberger, N. Birbaumer, and J. R. Wolpaw, “Eeg motor movement/imagery dataset,” 2009.
 - [7] L. N. Smith and N. Topin, “Super-convergence: Very fast training of neural networks using large learning rates,” 2018.
 - [8] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision,” 2015.