uc3m | Universidad Carlos III de Madrid

Master in Big Data Analytics
2023-2024

Final Master thesis

“ECG based cancelable biometric system”

Design and implementation of a cancelable biometric system based on Deep Learning using ECG data

Arturo Calvera Tonin

Tutor

Pedro Peris Lopez

Lugar y fecha de presentación prevista

ABSTRACT:

Debe incluir palabras clave

DEDICATORIA:

Tabla of contents

Índice de figuras

1. Introduction:

In this first introductory chapter, a set of definitions for keywords across the project is presented alongside with the motivation for the project, its objectives and its structure.

Deep Learning:

Biometric Systems:

Biometric Signals:

Biometric Markers

CPU. GPU, Neural Networ, Dense Network, Convolutional Neural Netrowk

gradient descent

train set and validation set

TODO términos clave,

TODO técnologías?, apparently yes

The intrinsic key concepts to understand correctly the following sections and chapters are therefore defined.

ALGO DE ODS IN MOTIVATION

1.1. Motivation

The motivation behind this project can be split into personal motivation and social/industry needs.

Diving into the personal motivation, I shall point out that the study and implementation of deep learning based schemas has become for me one of the most interesting fields of work along my development across this master. I believe that data science and, more specifically, deep learning are going to become the tool and engine to move society across the following decades. Therefore, due to my interest in learning and producing impact and value, I chose this theme for my final master thesis.

With respect to the industry or the social needs, I shall describe the current irruption of biometric systems in our daily lives. Fingerprint recognition has been widely used in smartphone authentication, computer login, and access control systems for buildings. Face recognition and iris-based user verification are often used in modern electronic devices. Biometrics are now combined with electronic passports for border control systems in many countries. Biometric signals are constantly being collected and used by these systems and the risk and implications associated with this is sometimes disregarded.

For example, just a few months before the publication of this work, a new cryptocurrency, *WorldCoin,* made headlines for the wrong reasons. The project, founded by OpenAI CEO Sam Altman and Alex Blania, who is the Co-Founder and CEO of Tools for Humanity made headlines due to privacy concerns regarding the scanning of the user's Iris. *WorldCoin* scanned the user's Iris using a device called "*The Orb*" in exchange for its own cryptocurrency. A frenzy ensued for a couple of months in any metro station of any major city. "Free money" was seemingly the most common expression around these "*scanning centers*" and no one seemed to stop and think about the implications of signing up for this "service". Irises are a well known biometric marker and the implications of an adversary taking hold of this marker can be fatal. On March 6 2024, Spain's data privacy watchdog, the AEPD, alongside similar institutions from different countries, banned Worldcoin for up to three months, ordering it to cease the collection of personal data and stop using data it had already gathered. The AEPD said its action came after several complaints regarding insufficient information, the collection of data from minors or not allowing for the withdrawal of consent. [1]

Here is where cancelable biometrics come to play. Cancelable biometrics refers to the intentional and systematic distortion of biometric features to protect sensitive user-specific data. Biometric markers are mapped to a template, which is subsequently used, making it difficult to obtain the original biometric image from the distorted one. This approach aims to address privacy and security concerns associated with biometric recognition.

[1] G.Pascual, M. (2024,March 6). La Agencia de Protección de Datos prohíbe seguir recogiendo datos de iris a Worldcoin, que daba criptomonedas a cambio | Tecnología | EL PAÍS

Electrocardiogram (ECG) has been investigated as a promising biometrics for authentication, identification and liveness validation. (Israel et al., 2005), (Irvine et al., 2008), (Komeili et al., 2018). It has presented great possibilities for its strength against counterfeit. However, the ECG feature templates are irreplaceable, and a compromised template implies a permanent loss of identity. I strongly believe that biometric systems based on ECG signals are going to become part of our daily routine in the foreseeable future. Therefore, investigating and solving the problems related to these systems before the problem arrives is of interest to society.

1.2. Objectives

The following list aims to evidence the objectives that the work here described should cover. These objectives should be clearly stated and defined from the beginning in order to have a clear understanding of the path the work will take.

- Study the state of the art and understand the motivation behind biometric systems.
- Evaluate previous implementations of cancelable biometric systems based on ECG signals.
- Explore the cancelable biometric's industry and propose a first approach to deep learning based cancelable biometric systems
- Open the way for future researchers in the field of deep learning based cancelable biometric systems.
- Define the characteristics and objectives that the approach designed in this work should cover.
- Address and state the possible challenges faced by the approach even though some of the proposed solutions to those may not be implemented here.
- Implement a first version of a biometric system using the deep learning approach defined in this work.
- Evaluate the implemented biometric system and evidence that the objectives are covered.
- Work along the whole project making use of the knowledge gained during the whole master's degree.
- Document all the processes along the project.

1.3. Structure

In the following list the structure of the current document is presented. The document is organized in chapters and sections grouping key points and specific points within those respectively.

- **Chapter 1 Introduction:** the current chapter. The reader is introduced to the context, motivation and objectives of the project.
- **Chapter 2 Design:** in this chapter, the state of the art is studied and evaluated, the objectives of the approach are stated and defined and the deep learning approach to cancelable biometrics is designed.
- **Chapter 3 Implementation:** in this chapter, a biometric system using the previous deep learning approach is implemented.
- **Chapter 4 Evaluation and Future Work:** in this chapter, the previously implemented system is evaluated and the future work is stated.
- **Chapter 5 Conclusion:** in this final chapter, the conclusions gathered from the work accomplished are presented.
- **Annex I**: TODO

2. Design

In this design chapter, the current state of the art in cancelable ECG biometric systems is studied and evaluated and a deep learning approach to cancelable biometrics is designed and its objectives defined.

2.1. State of the art

In this section, the relevant articles related to cancelable ECG biometrics are going to be reviewed in chronological order.

- **(Peng-Tzu et al., 2017):** in this scheme, distinct biometric templates for a given beat bundle are constructed via "subspace collapsing." To determine the identity of any unknown beat bundle, the multiple signal classification (MUSIC) algorithm, incorporating a "suppression and poll" strategy, is adopted. Knowledge of the distortion transform is not required for recognition. Using the PTB [2] database, the researchers achieved the best recognition rate of 97.58 % under the test condition *Ntrain* = 10 and *Ntest* = 10.

  In contrast to this previous work, the approach here designed will tackle independent beats instead of beat bundles. Also the acquired accuracy will be used as a possible benchmark.

- **(Kim & Chun, 2019):** in this article, a cancelable ECG biometrics by deriving a near-optimal generalized likelihood ratio test (GLRT) from a composite hypothesis testing in CS domain was proposed. Also a novel revocation process for CS based cancelable biometrics was proposed. The authors assured "that it is robust to record multiplicity attacks". Using the ECG-ID[3] database, the researchers achieved a 93% detection probability at 2% false alarm ratio (FAR) and 3.8% equal error rate (EER), which turned out to be comparable to or even better than the non-cancelable baseline.

  The approach here designed will also try to achieve a robust revocation schema and comparable performance results.

- **(Hammad et al., 2019):** in this article, the authors generated a novel cancelable ECG template for human authentication system based on two methods, improved Bio-Hashing and matrix operation, which, they claim, are much more secure than other cancelable methods. They overcome the main drawbacks of the Bio-Hashing method in the worst case that an impostor steals the Hash key by using the improved Bio-Hashing method. Also they overcome the main drawbacks of the Improved Bio-Hashing method in case an impostor tries to

[2] Bousseljot, R., Kreiseler, D., & Schnabel, A. (1995). Nutzung der EKG-Signaldatenbank CARDIODAT der PTB über das Internet. Biomedizinische Technik / Biomedical Engineering, 40(s1), 317-318.

[3] T. S. Lugovaya, "Biometric human identification based on electrocardiogram," M.S. thesis, Dept. Fac. Comput. Technol. Inf., Saint Petersburg Electrotech. Univ., Saint Petersburg, Russia, Jun. 2005

perform a brute force attack to cause a false accept by combining the Improved Bio-Hashing method with aliveness detection system (ECG system).

Also worth mentioning, this approach used a Feed-Forward Neural Network (FFNN) classifier for authentication to improve the performance. Thus appearing the first relationship between this field of work and deep learning.

Finally, using the PTB database, the team of researchers obtained a EER of 20% and 6% at a FAR of 27% and 2% for both the improved Bio-Hashing and the matrix operation approaches. A bit worse than Kim & Chun.

- **(Sakr et al., 2022):** in this article, the authors propose a novel cancelable ECG method using DNA and Amino Acid method combined with deep learning, which satisfies all cancelable requirements. They are the first to propose a cancelable ECG system that employs deep learning for human authentication, which overcomes the problems that faced most of the previous traditional machine learning cancelable methods. They use the pre-trained VGG-16[4] model.

  In relation to the results, both the ECG-ID and PTB datasets where used and they achieved, respectively, an average accuracy of 98.60% and 98.85% and EERs of 0.44% and 0.4%. This makes this approach the best performing one. In the work carried out here, the conclusions and knowledge gathered by these authors from working with deep learning methods will be taken into account.

2.2. Objectives and Design of the Proposed Approach

In this section the proposed approach will be introduced. First the objectives intended to cover will be defined and then the actual design process will be described. Once again it should be reiterated that the main objective of this work is to explore and propose a first approach to deep learning based cancelable biometric systems. Therefore, this approach aims to open the way for future researchers in the field and not completely solve all the challenges and problems known to cancelable biometric systems.

2.2.1. Objectives of the proposed approach:

- **Ease of use and simplicity:** the approach here defined should be as simple as possible. Simple to use, simple to understand and simple to modify in case of improvements. If need be, for the sake of simplicity, some of the more complex cancelable requirements can be disregarded or given less consideration.

- **Deep Learning:** the approach here defined should use deep learning as its base.

[4] C. Sitaula, M.B. Hossain, Attention-based VGG-16 model for COVID-19 chest X-ray image classification | Applied Intelligence, Appl. Intell. 51 (5) (2021) 2850–2863.

As it was previously mentioned, deep learning based approaches led to the best results in this field of work.

- **Comparable Results:** the approach here defined should be evaluated with the database or databases commonly used for this field of work and the results obtained should be comparable. Ideally using the same performance metrics.

- **Open-Source:** this approach should be designed and implemented with open-source/free-to-use programming languages and libraries.

- **Cancelable Biometrics Requirements:** the cancelable biometrics requirements will be considered across the design and implementation of the schema. These requirements appear here listed in order of importance:
    - **Non-invertibility**: The recovery of original biometric data should be impossible.
    - **Efficiency:** satisfying the requirements imposed by cancelable biometrics should not deteriorate recognition performance.
    - **Diversity**: The same cancelable template shall not be used across various applications; so many protected templates from the same biometric feature need to be generated.
    - **Reusability (Revocability)**: there should be straightforward revocation and reissue procedures in the event of compromise.

- **Recognition, Authentication and Identification:** all these three actions, even though they can be considered very similar, should be supported by the schema.

2.2.2. Design of the proposed approach:

The design of the proposed approach should be split into different subtasks. The list containing such subtasks and their respective resolution/approach taken is here listed:

- **Database, programming language and libraries :** firstly, due to its ease of use and its wide use around the ECG biometric industry, the previously mentioned ECG-ID database will be used in this work. This dataset is developed for authentication tasks and considered the best choice for working with ECG biometric. This database has records from 90 persons (46 women and 44 men) and each record is twenty seconds from lead I. Each signal has 12-bit resolution and is digitized at 500 Hz. The records of each subject were collected from 2 to 20 session data. The database will be collected from the available resources in *PhysioNet.*[5] *PhysioNet* is an online repository and resource for physiological signals and related data for research and education in biomedical engineering

[5] Goldberger, A., Amaral, L., Glass, L., Hausdorff, J., Ivanov, P. C., Mark, R., ... & Stanley, H. E. (2000). PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. Circulation [Online]. 101 (23), pp. e215–e220.

and physiology. Provides free access to a wide range of physiological signals, time series, and related clinical data, along with software tools and resources for analyzing and interpreting these data. It forms part of the National Institute of Biomedical Imaging and Bioengineering (NIBIB) of the National Institutes of Health (NIH).

Likewise, due to its ease of use, flexibility and its extended use in the deep learning industry, the python library *PyTorch*[6] was chosen to support the deep learning efforts across this work. *PyTorch* is an open-source machine learning framework developed by Facebook's AI Research lab (FAIR). It provides a flexible and dynamic computational graph, making it particularly well-suited for deep learning tasks. *PyTorch* allows researchers and developers to build and train neural networks efficiently, with support for both CPU and GPU acceleration. Its key features include automatic differentiation, which simplifies the process of computing gradients for optimizing neural network parameters, and a rich ecosystem of libraries and tools for tasks such as computer vision, natural language processing, and reinforcement learning.

Finally, the library *scikit-learn*[7] often abbreviated as *sklearn* will be used across this project to support a wide array of machine learning tasks. It is a widely-used open-source machine learning library in Python. It offers simple and efficient tools for data analysis and machine learning tasks, including classification, regression, clustering, dimensionality reduction, and model selection. It also includes utilities for data preprocessing, feature extraction, and model evaluation. Its emphasis on usability, scalability, and interoperability has contributed to its widespread adoption in academia and industry for developing machine learning applications.

- **Data handling & preprocessing:** data handling and signal preprocessing is a very important step (containing many different substeps) within ECG related work:
    - **Signal filtering**: The ECG signal represents the electrical activity of the heart over time, typically recorded through electrodes placed on the skin. However, these signals are often corrupted by various types of noise and artifacts, which can obscure important features and make accurate interpretation difficult. In this case, the database provided by *Physionet* offers an already filtered version of the original raw signal "*ECG I Filtered".* This signal is chosen to be used across the work here described.

[6] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., … Chintala, S. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32* (pp. 8024–8035).
[7] Pedregosa, F., Varoquaux, Ga"el, Gramfort, A., Michel, V., Thirion, B., Grisel, O., … others. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*(Oct), 2825–2830.

- **Outlier detection and correction**: Initial outlier detection and correction in the gathered signals was initially considered. After careful consideration, it was disregarded due to the possibility of information leakage. That is, leaking information from the testing set into the training set. Traditional threshold based methods can incur in this and thus, in the end, set up the trained models for unrealistic performance estimation. Another argument in favor of this decision is the fact that none of the previous work using this database refers to outlier correction.

- **ECG feature extraction**: traditionally, ECG based biometrics have followed two approaches: either using whole ECG records as data points or using individual heartbeats as the data points. Both come with advantages and disadvantages and in the previously reviewed state of the art, both techniques are used by the different authors.

  Considering the future applications of the system, individual heartbeats are considered more suitable. Recording a single heartbeat for recognition is faster and more probable in the real world.

  Nevertheless, heartbeat extraction or heartbeat feature extraction is not a straightforward task. ECG signals can be thought of different features or components:
  - The QRS complex is a key feature of the ECG signal, representing the depolarization of the ventricles of the heart. It consists of three distinct waves:
    - Q wave: The initial downward deflection from the baseline.
    - R wave: The sharp upward deflection following the Q wave.
    - S wave: The downward deflection following the R wave.
  - In addition to the QRS complex, other important features and components of the ECG signal include:
    - P Wave: Represents atrial depolarization, which precedes ventricular depolarization. The P wave is typically smaller and precedes the QRS complex.
    - T Wave: Represents ventricular repolarization, occurring after ventricular depolarization (QRS complex). The T wave is typically a smooth, upward deflection from the baseline.
    - ST Segment: The segment of the ECG waveform between the end of the S wave and the beginning of the T wave. It represents the interval between ventricular depolarization and repolarization and is important for assessing myocardial ischemia or infarction

Choosing the correct complex of features to use can be considered a hard problem itself. Also the different complexes and waves occur at different intervals depending on the person therefore complicating the task.

The industry standard to solve this problem is heartbeat extraction using R-peaks detection. Many different algorithms to detect these peaks have been implemented and studied. This technique consists of detecting the R-peaks in the signal and then collecting a window of data around said peak. Usually around 40% of the data points are collected from signal values before the peak and the rest are collected from the signal after the peak.

In this specific project, the biosignal processing library *Biosppy*[8] was chosen to tackle this task. The library offers many algorithms to deal with ECG data and in particular, offers a method to perform heartbeat extraction using the technique previously described. In particular, 300 data points make up the extracted heartbeat therefore providing a fixed dimensionality for the input data or signals used for training.

It's worth mentioning that the database provides an annotation file with information about 10 annotated beats (unaudited R- and T-wave peaks annotations from an automated detector) yet not all R-peaks in the record are annotated as the following figure evidences.

[8] Carreiras, C. et al., 2015. BioSPPy: Biosignal Processing in Python, Available at: "https://github.com/PIA-Group/BioSPPy/"

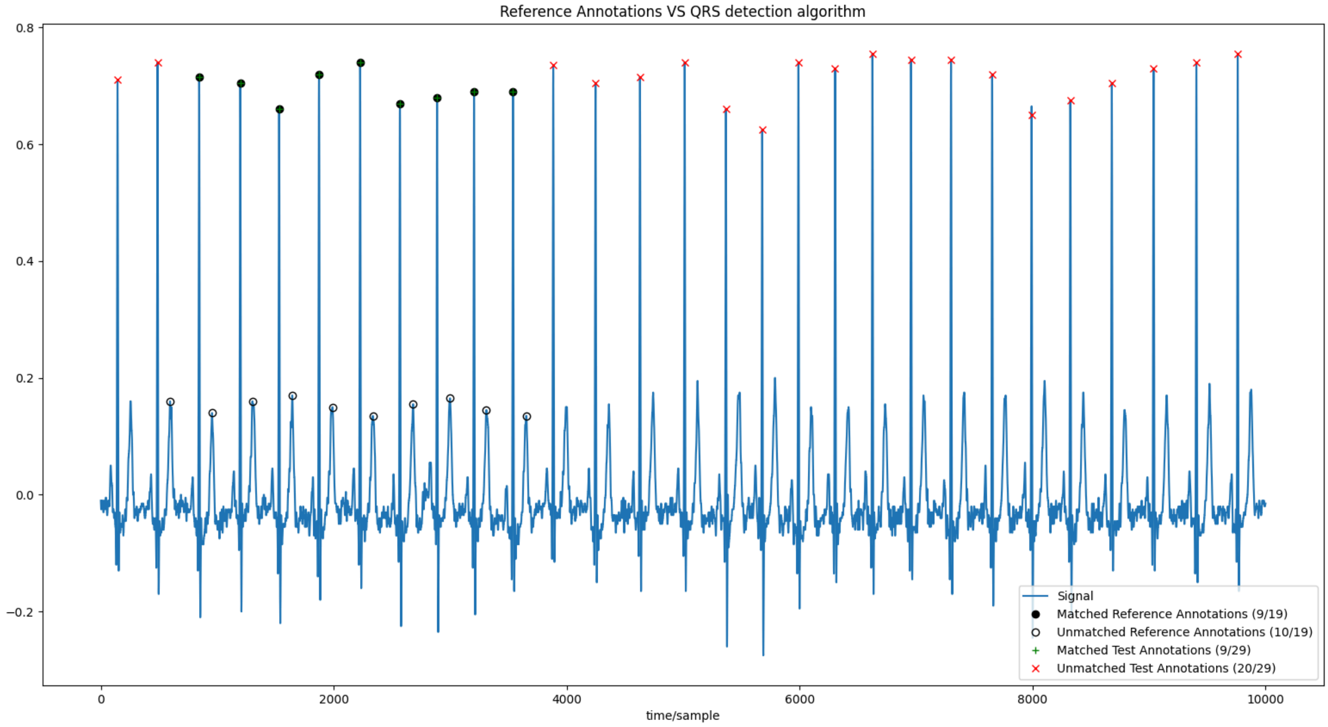


Figure 1: ECG record for a person in ECG-ID database. Annotated beats vs detected beats.

After running the previously mentioned R-peaks detection and heartbeat extraction algorithm, we can superpose the heartbeats and get an idea of the data points related for each person.

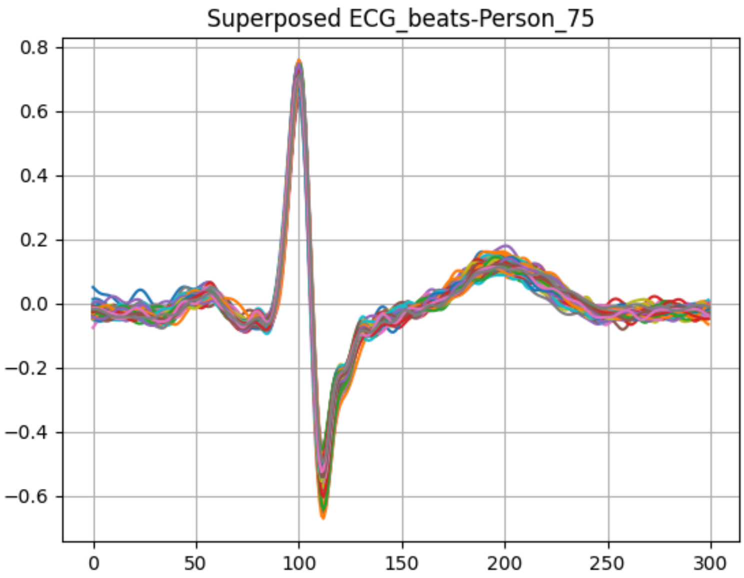


Figure 2: Superposed heartbeats gathered from ECG records for person 75.

Differences between the superposed beats for each person can sometimes be appreciated to the naked eye. It’s also worth mentioning that sometimes “noisier” heartbeats are collected but this is going to be treated as an advantage. More diverse data for each person sets up the models for correct generalization in the testing environment.

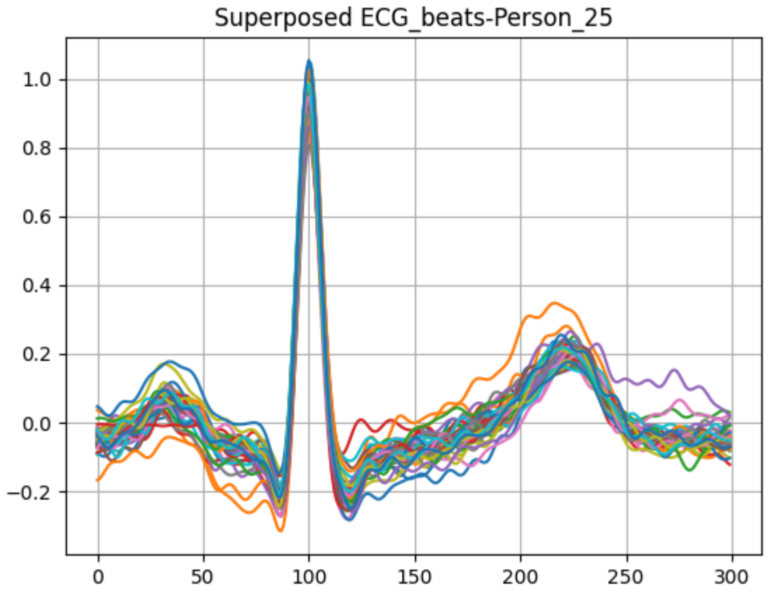


Figure 3: Superposed heartbeats gathered from ECG records for person 25.

- **Data imbalance:** data imbalance is a problem inherent to this database. The problem arises due to the fact that all users have either 2 records available or more. Other authors in the state of the art have mentioned this problem and have solved it with different approaches. In this project, for those users with more than 2 ECG records available, a random sampling is performed and only 2 are chosen therefore reducing the imbalance problem. Also, due to the fact that the “data unit” are individual heartbeats, we’re going to look at the distribution of number of heartbeats collected per user:

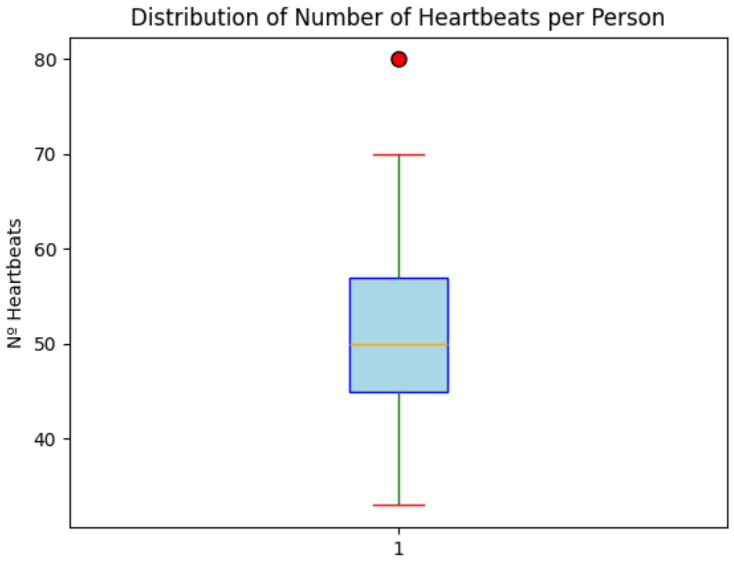


Figure 4: Boxplot of the distribution of number of extracted heartbeats per user.

From the previous boxplot of the distribution of number of extracted heartbeats per person, it can be concluded that keeping only those users with more than 40 extractor heartbeats will fairly suppress the data imbalance problem.

- **Deep learning model capable of template issuing:** in this design step, a deep learning schema capable of creating cancelable biometric ECG templates that fulfill the objectives previously designed must be envisioned.

  Taking on from the lessons learnt across the development of the master's degree, convolutional neural networks and, in particular, one dimensional convolutional neural networks seem the correct model to learn the hidden features within heartbeats which make them unique to each person.

  Classification or identification of the persons based on their heartbeats can be achieved by introducing a dense neural network or MLP that feeds from the output of the previously mentioned convolutional neural network. The array output of this network will determine the probability that a certain heartbeat belongs to each of the persons enrolled in the system, that is, each of the persons for which the network was trained.

  The following schema represents the initial network design.

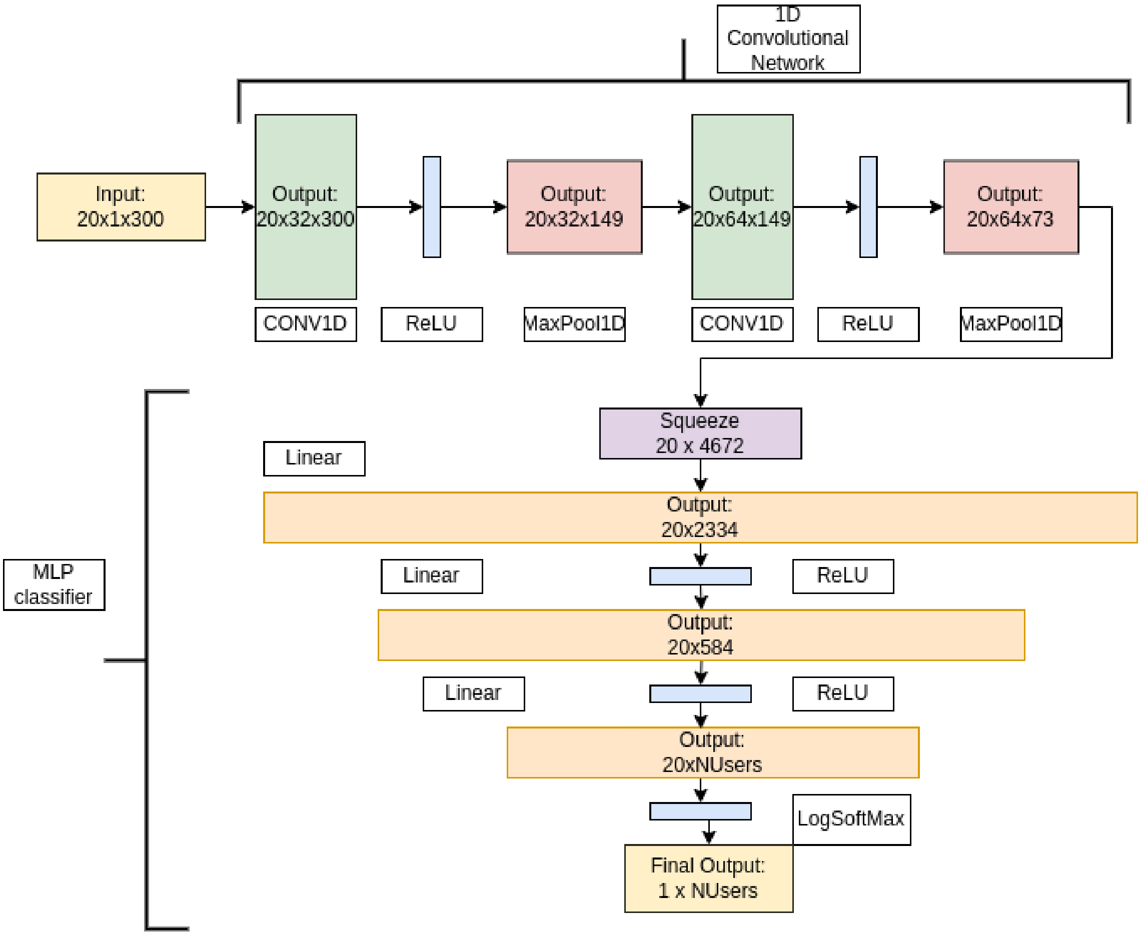


Figure 5: Architecture of the Deep Learning model. (No regularization)

In the above schema, the regularization layers imperative to ensure the model correctly generalizes in the testing set are not represented for the sake of simplicity. As it's commonly done in the deep learning field of work, a complex enough network is defined and once the model is proven able to "memorize the data" the regularization techniques or layers are implemented.

It should be noted that these layers consist of:

- 1 dimensional batch normalization layers placed after each of the convolutional layers. Batch normalization normalizes the activations of each layer by subtracting the batch mean and dividing by the batch standard deviation. This helps mitigate the internal covariate shift problem, where the distribution of activations in each layer changes during training, leading to slower convergence and degraded performance. Also the vanishing gradient problem is addressed. This occurs during the backpropagation algorithm, where gradients in the early layers of the network become extremely small as they are propagated backward through the network, also degrading the performance. Batch normalization acts as a form of regularization by adding noise to the activations during training. This helps prevent overfitting and improves the generalization performance of the model even though this is not a very deep model.
- Drop-out layers placed right before each of the linear layers. Dropout is a form of regularization that helps prevent overfitting by randomly dropping out (setting to zero) a proportion of the neurons during training. This forces the network to learn more robust features that are not dependent on the presence of specific neurons, leading to better generalization to unseen data. The probability of dropout will be chosen later on in the implementation step.

It's also worth mentioning that early stopping will be used in the training process of the model.

Once this model is working with high accuracy, an efficient biometric system that has learnt the hidden aspects of the heartbeat signal for each user has been built. That is, the model is a deep learning machine that has inner knowledge stored in its parameters that allows for user identification based on heartbeat signals.

Here is where the novelty of the designed approach comes. To the author's knowledge, this technique has not been previously used to generate cancelable biometric templates.

Cancelable templates through activation maximization: activation maximization is a technique used to visualize and understand the behavior of neural network models, particularly convolutional neural networks, by generating input stimuli that maximally activate specific neurons or layers within the network. The goal of activation maximization is to find input patterns that produce high activations in desired neurons or layers, providing insights into what features or patterns the network has learned to detect.

The process optimizes the input stimuli, in this case random data, to maximally activate the selected target neurons or layers. This is typically done by performing gradient ascent on the input stimuli with respect to the activations of the target neurons or layers. The optimization process involves iteratively updating the input stimuli to increase their activations until convergence.

Once the optimization process converges, the resulting input stimuli represent patterns that produce high activations in the target neurons or layers. Using this approach and the previously accurate convolutional model, any neuron *z* in the last layer can be used in this process to generate a “synthetic” input signal that represents the inner knowledge of the model when classifying a heartbeat as belonging to the person *z*.

Now is a good time to evaluate the approach to validate that all the cancelable requirements are met:

- **Non-invertibility**: The recovery of original biometric data is impossible given the new generated template.
- **Efficiency:** The efficiency should be evaluated in the results section. Nevertheless, in this section, the design of the results validation schema will be stated.
- **Diversity**: Due to the iterative nature of the process and the fact that many different initial signals can be used, the approach does provide different templates every time.
- **Reusability (Revocability)**: the reissue of the templates is straightforward yet the revocability is not. This is one of the main drawbacks of the approach but possible solutions are stated below.

Ensuring revocability of the cancelable templates: as it was just stated, revocability is not straightforward or inherent to this approach yet many “lowcost” solutions can be implemented. For example, the use of watermarks and a helper system that works as the “revocation authority” can be an easy way to fulfill this requirement.

Diving into this approach, we shall first talk about watermarking in neural networks. Watermarking in the context of neural networks has been extensively studied. To the authors knowledge, the most relevant work appears in (Adi et al., 2018), (Uchida et al., 2017) and (Pagnotta et al., 2022). To summarize the concept, watermarking or digital watermarking is the process of robustly concealing information in a signal (e.g., audio, video or image) to be subsequently used to verify either the authenticity or the origin of the signal. In the case of neural networks (NN), this concept can be applied as a way to identify that a certain output was produced by a specific model.

If the activation maximization process can be equipped with a watermarking schema, the templates produced by the model could be unequivocally identified. From this point on, a simple "revocability authority" system could be implemented using a blacklist and a whitelist as such:

- Originally all watermarks (generated and present within each template) for all users belong to the whitelist.
- Once a template has been compromised and revocation is needed, its associated watermark is added to the blacklist. (Will not be removed from the whitelist)
- Finally, to feed a template to the biometric system, the watermark within it is extracted and the following are checked:
    - If the watermark was never in the whitelist, the template is not generated from the original model thus deeming it a fabricated template.
    - If the watermark is in the whitelist and in the blacklist, the template is not suitable for identification. It 's compromised.
    - Else, the template is fed to the system and identification ensues.

This approach ensures that if small changes are applied to the stolen template and the robust watermark is still latent, the blacklisted template will still be revoked. If the watermark cannot be ensured after small changes to the stolen template, the watermarking extraction process will return a never whitelisted watermark thus deeming the template unauthorized for the biometric identification.

Once again, as previously mentioned in the objectives of the project, this proposed solution will not be implemented for the sake of brevity yet it will be considered as one of the first steps to take in the future work.

- **Evaluation schema:** finally, a clear evaluation schema needs to be defined to evaluate all the design steps after their respective implementation.
    a. Evaluate performance of the CNN+MLP model. That is, the model should very accurately identify the users in the database using one heartbeat.
    b. Ensure that different templates can be issued for the same user.
    c. Evaluate the performance of the CNN+MLP model using the generated cancelable templates. That is, the performance of the model using the cancelable templates should be comparable to that obtained using the original heartbeats.

d. Train a new simple machine learning algorithm with generated templates and evaluate that the performance of the new model is comparable to the one obtained in steps a,c.
e. Evaluate the training computational effort.

With these five steps, the previously mentioned requirements for cancelable biometrics will be evaluated (with exception of the revocability constraint).

3. Implementation

In the implementation chapter, the previously defined approach will be implemented. All the implementation steps taken will be addressed chronologically. The considerations of important implementation decisions will be addressed and motivated.

3.1. Implementation of dataset related code:

Firstly, the python file ***dataHandler.py*** was implemented. The objective of this file is to gather and implement all the necessary code to load the ECG-ID dataset and transform its records to allow for deep learning. The next list dives into the contents of this file:

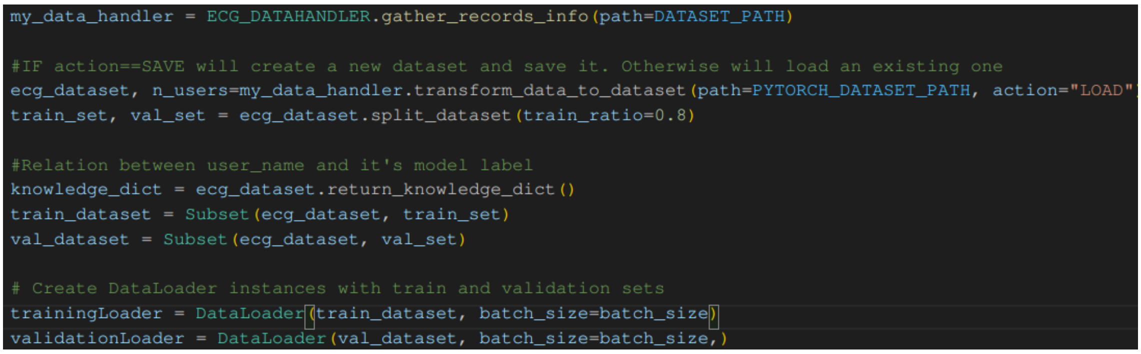
→ ECG_DATAHANDLER: the objective of this class is loading and preparing the data for deep learning. The following list gathers the methods defined within this class:

- **gather_records_info:** this *classmethod* is used to access the locally available dataset. The method iteratively accesses the subdirectories of the dataset and gathers how many users are available and the datafiles available for said users. This information is stored into a directory available to the class. The method returns an instance of the class.
- **prepare_data:** this method reads iteratively the previously mentioned directory. For each user, the available records are read. Right at this point, just as previously mentioned, the data imbalance problem is addressed sampling 2 random ECG records per user. Next, the previously mentioned *BioSPPy* library is used to read the filtered available signal and detect the heartbeats within each sampled record. In particular the *ecg* method is used. It's worth noting that the sampling rate is provided to this method. Finally, the loaded data is stored into a new directory called *beats_dict* containing the read heartbeats per user.
- **transform_data_to_dataset:** this method transforms the previously mentioned heartbeats to *tensors*. This is the datatype used by *PyTorch* models. The data is normalized here and the labels associated with each user are created. Keep in mind that this is a classification problem. The returned object is an instance of the *ECGDataset* class which is explained in the following section. The method provides a mechanism to load a previously stored dataset or to create and save a new dataset. Keep in mind that a new dataset is created because the sampling of records per user is random.

→ ECGDataset: the objective of this class is storing the previously loaded heartbeats in the shape of a *tensor* dataset and making this data available for the deep learning models. The following list gathers the methods defined within this class:

- **split_dataset:** this method is used to split the dataset into training and validation datasets. The partition of the data ensures that each user is equally represented in both subsets.
- **return_knowledge_dict:** this method is used to return the relationship between each user and their associated class label. This information will later be used.

The following figure provides an example to the reader on how these classes are used.



```
my_data_handler = ECG_DATAHANDLER.gather_records_info(path=DATASET_PATH)

#IF action==SAVE will create a new dataset and save it. Otherwise will load an existing one
ecg_dataset, n_users=my_data_handler.transform_data_to_dataset(path=PYTORCH_DATASET_PATH, action="LOAD"
train_set, val_set = ecg_dataset.split_dataset(train_ratio=0.8)

#Relation between user_name and it's model label
knowledge_dict = ecg_dataset.return_knowledge_dict()
train_dataset = Subset(ecg_dataset, train_set)
val_dataset = Subset(ecg_dataset, val_set)

# Create DataLoader instances with train and validation sets
trainingLoader = DataLoader(train_dataset, batch_size=batch_size)
validationLoader = DataLoader(val_dataset, batch_size=batch_size,)
```

Figure 6: Example of use of the ECG_DATAHANDLER and ECGDataset class.

The full code of the implementation of this file here described is available here.

3.2. Implementation of the deep learning related code:

Secondly, the python file ***ECG_CNN_MODEL.py*** was implemented. The objective of this file is to gather and implement all the necessary code to implement the deep learning model that will support the previously defined approach. The next list dives into the contents of this file:

→ ECG_1D_CNN: the objective of this class is implementing the previously defined deep learning model. More specifically this class holds the code to implement a one dimensional convolutional neural network together with a dense network classifier. The layers confirming the model are defined in the constructor of the class. The class extends from the *PyTorch* class *nn.Module*. The following list gathers the methods defined within this class:

- **forward**: this method is always implemented by the classes extending from *nn.Module.* The method controls how the input data traverses through the network. The method has a parameter that controls whether the previously mentioned regularization layers are activated or not.

→ ECG_1D_CNN_TRAINER: the objective of this class is implementing the code to train the model defined in the previously introduced class. It extends said class *(ECG_1D_CNN).* In the constructor of this class important parameters such as the learning rate, the optimizer or the loss function are defined. More specifically the *PyTorch* implementation of the *Adam optimizer*[9] and of the negative log likelihood loss are respectively used. It's worth mentioning that this loss is specifically useful for classification problems with C classes such as this one. Also, *cuda* [10] capable devices are detected here (more on this later on in the evaluation section). The following list gathers the methods defined within this class:

[9] Diederik P. Kingma and Jimmy Ba (2017). Adam: A Method for Stochastic Optimization. In *arXiv:1412.6980.* https://arxiv.org/abs/1412.6980

[10] NVIDIA, Vingelmann, P., & Fitzek, F. H. P. (2020). CUDA, release: 10.2.89. Retrieved from https://developer.nvidia.com/cuda-toolkit

- **trainloop:** given a validation and a training dataset, this method trains the previously implemented deep learning model and evaluates its performance during the training. The training is done using batches of data for a set of epochs. The method allows for the use of an early stopping callback to regularize the model. The callback evaluates the validation loss to ensure the model keeps learning and generalizing correctly in the validation set. After the training is completed, the extended model is ready to be used.
- **eval_performance:** given a dataset, this method evaluates the performance of the previously trained model against the provided dataset. An instance of the *MetricsHolder* class is returned. This class is later introduced to the reader.
- **activation_maximization:** this method allows for the execution of the previously described activation maximization algorithm. Given a target class, the model iteratively executes a gradient ascent algorithm against a random input to the model until the model indicates that the target class is achieved.

  To evaluate that the target class is achieved two different stopping criteria are implemented. Both need to be covered to stop the process:

  - The output of the model, essentially a probability vector, should indicate that the input belongs to the target class.
  - The difference between the highest probability and the second highest probability should be higher than a *delta,* thus indicating that the model is confident in the prediction.

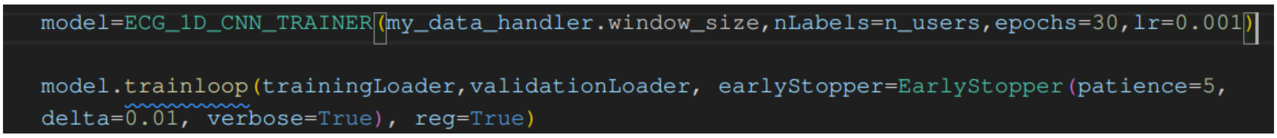
  ADD FORMULA?

- **create_templates:** this method finally creates the cancelable templates associated with each user. As previously mentioned, it uses the activation maximization algorithm implemented within the class. Given a number of templates to generate *T*, the method executes the activation maximization algorithm *T* times for each of the users for which the model was trained to recognize. Said templates are returned and stored in a .csv file and their visual representation is also plotted and stored for later evaluation and use.

→ EarlyStopper: the objective of this class is implementing the code of an early stopping algorithm. This class works as a callback function. It ensures that a model keeps regularizing correctly during training. This is done subsequently evaluating the validation loss and the previously stored best validation loss. If the validation loss does not decrease after each epoch the class signals the model to stop training. A patience counter is also implemented and the parameters of the best model (lowest validation loss) are returned to be used.

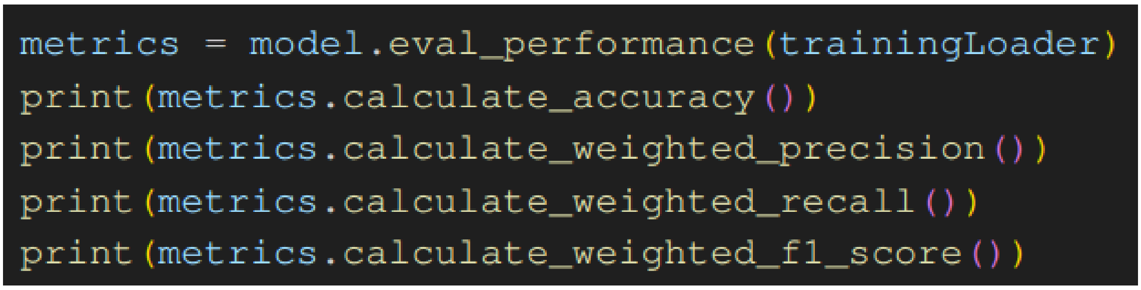
→ MetricsHolder: the objective of this class is implementing an object to easily access performance metrics. The metrics chosen and the implications of using said metrics will be introduced in the evaluation chapter. For the moment the reader should know that a set of methods are available to calculate *accuracy, precision, recall* and *f1-score* given a set of true labels and predicted values.

The following figures provide a series of examples on how the previous classes and methods are used:



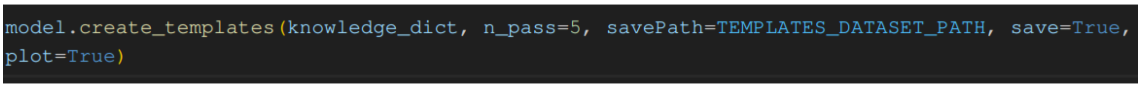
```
model=ECG_1D_CNN_TRAINER(my_data_handler.window_size,nLabels=n_users,epochs=30,lr=0.001)

model.trainloop(trainingLoader,validationLoader, earlyStopper=EarlyStopper(patience=5,
delta=0.01, verbose=True), reg=True)
```

Figure 7: Example of instantiation of the ECG_1D_CNN_TRAINER class and subsequent training with regularization layers and EarlyStopper callback.



```
metrics = model.eval_performance(trainingLoader)
print(metrics.calculate_accuracy())
print(metrics.calculate_weighted_precision())
print(metrics.calculate_weighted_recall())
print(metrics.calculate_weighted_f1_score())
```

Figure 8: Example of performance evaluation on a previously trained model.



```
model.create_templates(knowledge_dict, n_pass=5, savePath=TEMPLATES_DATASET_PATH, save=True,
plot=True)
```

Figure 9: Example of cancelable template creation using a previously trained model.

The full code of the implementation of this file here described is available here.

3.3. Implementation of main code:

Finally, the python file ***main.py*** was implemented. The objective of this file is to gather and implement all the necessary auxiliary code to use all the functionality previously defined. The previous code snapshots come from this file.

The full code of the implementation of this file here described is available here.

4. Evaluation and Future Work

In this chapter, the previously implemented system is going to be trained and evaluated. The obtained results will be discussed and the future work is stated.

4.1. Evaluation Metrics:

To start this chapter, the metrics chosen to evaluate the performance of the biometric system and, in the end, the validity of the cancelable approach should be addressed.

When evaluating the performance of a biometric system, it's crucial to consider metrics that capture both the system's ability to correctly identify legitimate users (genuine positives) and its capability to detect impostors (genuine negatives) while minimizing false positives and false negatives. Weighted accuracy, recall, precision, and F1-score emerge as appropriate performance metrics due to their ability to address these key aspects effectively.

Weighted accuracy offers a comprehensive measure of overall classification accuracy, considering the proportion of correctly classified instances across all classes. In the context of a biometric system, where there might be multiple classes representing different users, weighted accuracy accounts for imbalanced class distributions by weighting each class's accuracy proportionally to its prevalence. This is particularly pertinent since data imbalance was already identified for the particular dataset used in this work.

Weighted recall, also known as sensitivity or true positive rate, evaluates the system's ability to correctly identify genuine users among all actual genuine instances. In a biometric context, recall measures the proportion of legitimate users correctly recognized by the system, crucial for ensuring that authorized individuals are reliably granted access while minimizing false rejections.

Weighted precision, on the other hand, quantifies the system's precision in correctly identifying genuine users among all instances classified as positive. It measures the proportion of correctly identified genuine instances out of all instances classified as genuine by the system. Precision is vital for biometric systems as it reflects the system's ability to avoid falsely accepting impostors, ensuring that unauthorized access attempts are minimized.

The **weighted F1-score**, which is the harmonic mean of precision and recall, provides a balanced measure that considers both false positives and false negatives. It offers a single metric that combines the strengths of precision and recall, making it suitable for evaluating the overall performance of a biometric system. By incorporating both precision and recall, the F1-score ensures that the system maintains a balance between minimizing false acceptances and false rejections, essential for maintaining security and user convenience in biometric authentication scenarios.

In summary, weighted accuracy, weighted recall, weighted precision, and weighted F1-score collectively offer a comprehensive framework for assessing the performance of biometric systems. They address the critical requirements of accurately identifying genuine users, minimizing false acceptances and rejections, and accounting for imbalanced class distributions, making them well-suited for evaluating the effectiveness and reliability of biometric authentication mechanisms.

4.2. Training and Evaluation of the Deep Learning Model:

4.2.1. Design of the proposed approach:

Firstly, the hardware environment used to train the model should be stated. This is crucial to correctly understand the computational cost of the solution. In particular, two different hardware environments were available for training in the researchers laptop:

→CPU based training: *Intel® Core™ i7-10510U*. This is a quad-core, eight-thread processor from Intel's 10th generation Comet Lake family, designed for thin and light laptops and ultrabooks. With a base clock speed of 1.8 GHz and a maximum turbo frequency of 4.9 GHz, the i7-10510U offers a balance of power efficiency and performance for a variety of computing tasks. It features Intel's Turbo Boost Technology, which dynamically adjusts the processor's clock speed based on workload demands, allowing it to deliver bursts of performance when needed while conserving power during lighter tasks. This dynamic frequency scaling helps optimize performance which is useful for the task at hand.

→GPU based training: *NVIDIA GeForce MX350*. This is a low-power, entry-level discrete graphics card targeted at thin and light laptops. While it's not in the same performance tier as higher-end GPUs like the *GeForce GTX* or *RTX* series, it still provides a significant boost in computational power over integrated graphics solutions commonly found in laptops. From a technical standpoint, the MX350 features 640 *CUDA* cores and is based on NVIDIA's Pascal architecture. In this specific scenario, it's equipped with 2GB of GDDR5 memory, offering sufficient memory bandwidth for handling moderately sized neural network models used in *PyTorch*. When it comes to training *PyTorch* models, the MX350 can deliver notable performance gains over CPU-only training or relying solely on integrated graphics. Its *CUDA* cores are specifically designed for parallel processing, making it well-suited for accelerating the matrix multiplications and gradient computations inherent in deep learning training workflows.

After this clear definition of the hardware environments, the following table provides information about the computational cost of training the model in each of the different environments:

Data Size:	CPU avg cost per epoch	CPU total training cost	GPU avg cost per epoch	GPU total training cost
- 3335 training instances. - 30 epochs. - batchsize 20.	- 22.932 secs.	- 687.96 secs.	- 5.183 secs.	- 155.53 secs.

Table 1: Performance comparative between CPU and GPU.

As the table emphasizes, the speed up obtained from using the GPU vs the CPU is around 4.42. Therefore the use of the GPU is chosen for the rest of the project. We should also note that the training time per epoch in the GPU is really small thus giving the model the versatility of retraining if needed.

4.2.2. Model regularization:

As previously mentioned, the training of the model can be either regularized or not using a combination of regularization layers and early stopping. To motivate this design decision, a series of performance metrics (against train and validation set) obtained from the training of the model using both settings are shown in the table below.

Setting:	Epochs:	W Accuracy Train	W Accuracy Validation	W F1-score Train	W F1-score Validation
No Reg	75	61.98%	61.47%	61.93%	61,49%
Regularized	30	92.71%	92.89%	92.68%	92.93%

Table 2: Performance comparative between regularization and no regularization.

From the above the following conclusions can be drawn:

- Batch normalization helped mitigate the performance drop possibly caused by internal covariance shift or vanishing gradient.
- Dropout and early stopping ensured the generalization of the model in the validation set.
- Regularization techniques are mandatory to ensure the success of the approach.

To once again reiterate the points here emphasized, the following figure shows the evolution of training and validation loss when training with the regularization techniques activated. We can see that both losses stay close together and the fact that the early stopping callback stopped the training after no improvement on the loss.

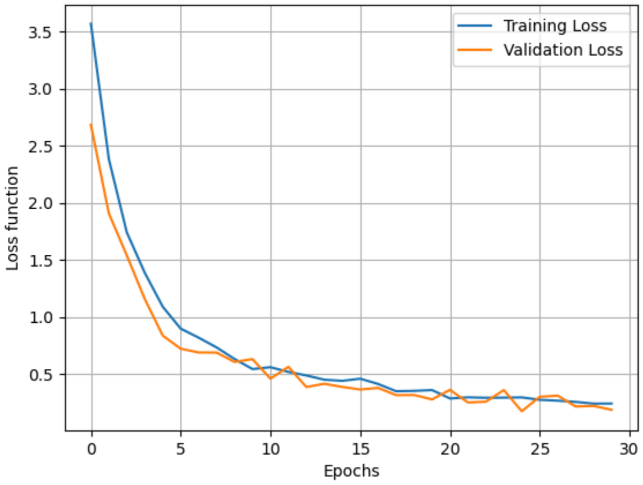


Figure 10: Evolution of the training and validation loss in regularized training.

At this point, we've trained a deep learning model with a high ability to identify users using an ECG heartbeat. Let's now dive into the generation of cancelable biometric templates.

4.3. Generation and Evaluation of the Cancelable Biometric Templates:

4.3.1. Template generation:

As previously mentioned, the generation of templates is done through the activation maximization algorithm. It may be complicated to understand what the output of this iterative process is, therefore, the following figures are going to help visualize the differences and similarities between the original signal and the generated template for a couple of users in the database

Firstly, a case where the superposed beats and the generated template still hold a lot of similarities to the naked eye:

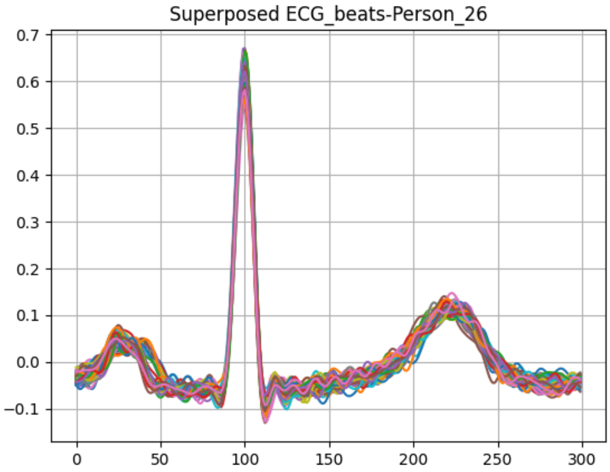


Figure 11: Superposed heartbeats for Person 26.

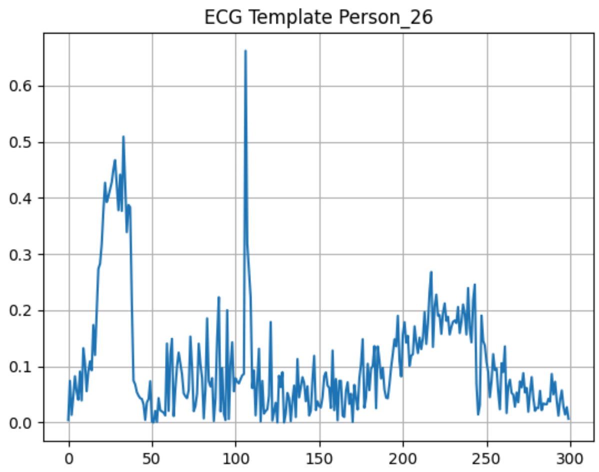


Figure 12: Generated cancelable template for Person 26.

It could be argued that the P wave, the R peak and the T wave can be recognized in the template. This is probably because we've looked at the previous image before. Paying closer attention to the differences, it can be seen that the range of the generated templates is different to that of the original and also the generated signal is less stable than the original. Nevertheless, it's quite clear that even in the case of some visual similarities, the original signal cannot be recovered from the generated one

Now, an example where the superposed beats and the generated template seem to be totally unconnected:

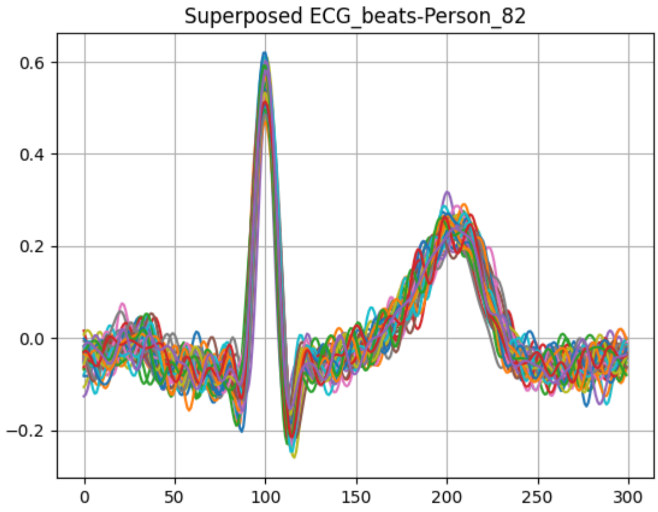


Figure 12: Superposed heartbeats for Person 82.

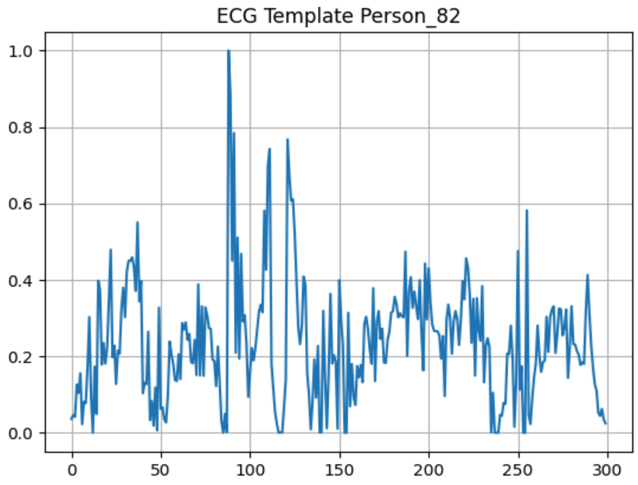


Figure 13: Superposed heartbeats for Person 82.

The point stated previously is quite clear. No one would relate the two signals and the original signal cannot be retrieved from the generated one.

With these two simple examples, the reader has a clear understanding of what the templates look like. Now, the evaluation of the cancelable quality of the templates is due.

4.3.2. Evaluation of the cancelable requirements:

→Non-invertibility: given a complex model like in this case, the original data will never be generated through the activation maximization algorithm. Therefore, no generated template will be the original. Furthermore, given the case that the model's parameters are stolen, recovering the original input solely from the weights of a model is generally not possible, especially in deep learning models with multiple layers and complex transformations. The weights of a neural network represent the learned relationships between the input data and the output predictions, but they do not directly encode the original input data in a reversible manner. Once the input data is processed by the layers of a neural network and transformed by the activation functions and weight matrices, much of the original information is typically lost or highly distorted. This is especially true in deep neural networks where the input undergoes multiple nonlinear transformations. This requirement is therefore fully met.

→Diversity: just like previously explained, due to the iterative nature of the process and the fact that many different initial signals can be used, the approach does provide different templates every time. To further emphasize this point, three generated templates for the same user are presented superposed in the following figure.

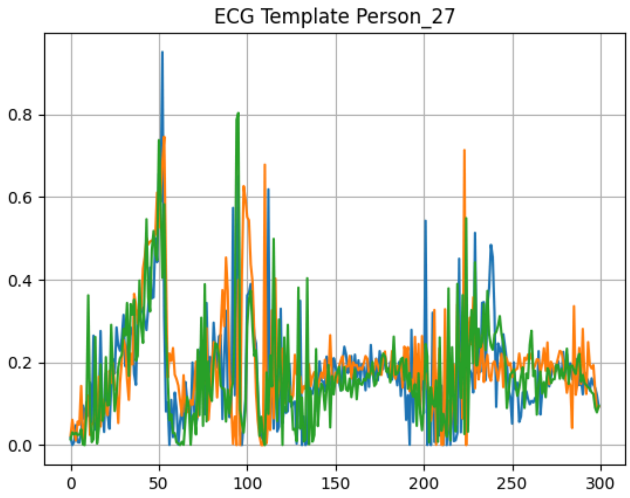


Figure 13: Superposed generate templates for Person 27.

Some correlation is obviously present but still to the naked eye, the signals can be deemed different. This requirement is therefore fully met.

→**Efficiency**: the efficiency requirement is going to be evaluated in two different ways. Firstly, the performance of the original model is going to be evaluated with the generated templates. Secondly, a random forest classifier is going to be trained using as input the generated templates. The original model will try to classify five templates for each user. The random forest classifier will be trained using hyperparameter tuning with gridsearch cross validation. In particular, said cross validation will use 4 templates as training and the final performance evaluation will be done with the never seen missing template.

Model	Weighted Accuracy	Weighted F1-Score
Original Model	95.309%	95.296%
Random Forest Classifier	91.358%	89.094%

Table 3: Efficiency evaluation of the identification using cancelable templates.

Great performance results were obtained in both cases. The original model performed better than previously. This is expected since the activation maximization process tried to ensure that the generated template was associated with the correct user. The new model, arguably simpler, proved a very high performance even with few training instances. Using more training instances and deeper hyper parameter tuning, the simpler classifier is expected to match the performance of the original model. The efficiency requirement of the templates can be deemed fully met.

→**Revocability**: lastly, as previously mentioned, the revocability method proposed in the design chapter will not be implemented or evaluated in this first approach. It will be part of the future development.

Compare the results obtained here with those from the state of the art.

State future work! (Recoke and a way to detect impostors)

Talk about EER and FDR?

X. Bibliography:

Adi, Y., Baum, C., Cisse, M., Pinkas, B., & Keshet, J. (2018, Agosto). "Turning your weakness into a strength: Watermarking deep neural networks by backdooring". *in 27th USENIX Security Symposium (USENIX Security 18) USENIX Association,*, pp 1615-1631.

Hammad, M., Luo, G., & Wang, K. (2019). *"Cancelable biometric authentication system based on ECG"* [Multimedia Tools and Applications 78 (2) (2019) 1857–1887]. https://link.springer.com/article/10.1007/s11042-018-6300-2

Irvine, J.M., Israel, S.A., Scruggs, W.T., & Worek, W.J. (2008). "Eigen-Pulse: Robust human identification from cardiovascular function". *Pattern Recognit*, *41*, 3327-3435.

Israel, S.A., Irvine, J.M., Cheng, A., Wiederhold, M. D., & Wiederhold, B.K. (2005). "ECG to identify individuals". *Pattern Recognit*, *vol*(38), 133-142.

Kim, H., & Chun, S.Y. (2019). "Cancelable ECG Biometrics Using Compressive Sensing-Generalized Likelihood Ratio Test". *IEEE Access*, *vol*(7), 9232-9242. https://ieeexplore.ieee.org/document/8606085

Komeili, M., Armanfard, N., & Hatzinakos, D. (2018). "Liveness detection and automatic template updating using fusion of ECG and fingerprint". *IEEE Trans. Inf. Forensics Security*, *vol*(13), pp 1810-1822.

Pagnotta, G., Hitaj, D., Hitaj, B., Pérez, Cruz, F., & Mancini, L.V. (2022). "“Tattooed: A robust deep neural network watermarking scheme based on spread-spectrum channel coding". *ArXiv*, *vol. abs/2202.06091.*

Peng-Tzu, C., Shun-Chi, W., & Jui-Hsuan, H. (2017). "A Cancelable biometric scheme based on multi-lead ECGs". *Annual International Conference of the IEEE*

Engineering in Medicine and Biology Society. https://doi.org/10.1109/EMBC.2017.8037610

Sakr, A. S., Pławiak, P., Tadeusiewicz, R., & Hammad, M. (2022). "Cancelable ECG biometric based on combination of deep transfer learning with DNA and amino acid approaches for human authentication". *Information Sciences*, *Volume*(585), 127-143. https://doi.org/10.1016/j.ins.2021.11.066.

Uchida, Y., Nagai, Y., Sakazawa, S., & Satoh, S. (2017). "Embedding watermarks into deep neural networks." *ACM on International Conference on Multimedia Retrieva*, 269-277.

Creación de una cuenta de servicio.