POLITECNICO DI TORINO

Corso di Laurea Magistrale
in Ingegneria Biomedica

Dicembre 2025

Tesi di Laurea Magistrale

Unsupervised deep learning framework for Semantic feature extraction from EEG data.

Relatori	**Candidato**
Prof. Luca Mesin	Vito Di Leo
PhD. Hossein Ahmadi	

Abstract

In this work, we investigate whether unsupervised learning can be useful for extracting well-separated latent semantic representations from EEG data. To train and evaluate the model, we used the University of Bath's publicly available dataset of semantic concepts for imagination and perception tasks, which contains data acquired from 12 subjects under various stimuli (auditory, pictorial, and orthographic) in two different cognitive conditions (imagination and perception) for three different semantic concepts (guitar, flower, penguin). EEG data were preprocessed, segmented into epochs and then convolutional autoencoders, integrated with temporal sequence modelling blocks (LSTMs and Transformers), were used to extract latent representations. Clustering metrics (Adjusted Rand Index, Normalized Mutual Information, Silhouette Score) and visualizations (t-distributed Stochastic Neighbor Embedding and Uniform Manifold Approximation and Projection) reveal that these latent representations are subject-specific and, for several subjects, can be distinguished based on cognitive state. However, semantic concepts remain hidden and don't form separated clusters, a sign of the limitations of the unsupervised approach.

Table of Contents

1 Introduction

This chapter will introduce the fundamentals of electroencephalography, define semantic features, and illustrate the Deep Learning techniques used in this thesis.

1.1 Electroencephalography

1.1.1 Central Nervous System

The Central Nervous System (CNS) is one of the two fundamental components of the nervous system. It consists of the brain and spinal cord and has the function of receiving and processing information from the afferent branches of the peripheral nervous system, then making decisions on the actions to be performed and sending appropriate instructions through the efferent branches.

The brain is composed of the cerebrum, diencephalon, cerebellum, and brainstem; it’s a large and complex structure, consisting of an external structure, the cerebral cortex, which carries out the highest brain functions, and internal subcortical regions. The cerebral cortex is divided into two hemispheres, each of which is made up of four regions, or lobes, specialized for specific functions [1]:

- Frontal lobe
  This is the largest lobe and is located anteriorly. It includes the primary motor cortex, involved in the control of motor activity, and other areas involved in language and personality determination.
- Parietal lobe
  Located immediately behind the frontal lobe, it includes the primary somatosensory cortex, responsible for processing sensory information associated with sensations of touch, temperature, and pain.
- Occipital lobe
  Located posteriorly, it includes the visual cortex, involved in receiving, processing, and interpreting visual information, which is then sent to other areas for further processing.

- Temporal lobe
  It is located inferior to the other lobes. It includes the auditory cortex, which is involved in processing auditory processes. Furthermore, this lobe is used in object recognition and in assigning semantic meaning to words and objects.

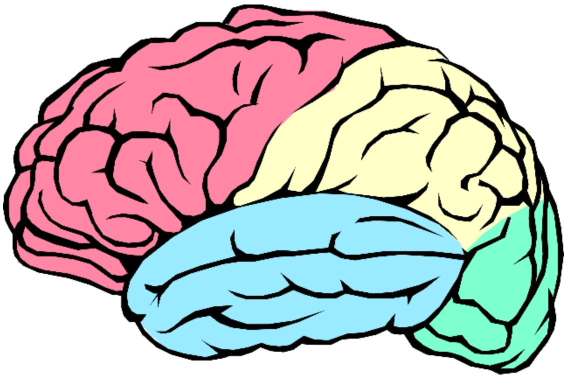


Figure 1: Brain Lobes. Frontal Lobe (red), Parietal Lobe (yellow), Occipital Lobe (Green), Temporal Lobe (Blue).

The brain, like the entire nervous system, is made up of two types of cells: neurons and glial cells.

The former are excitable cells that communicate with each other through the reception (via the dendrites), processing (via the cell body), and transmission (via the axons) of electrical signals, while the latter perform structural and metabolic support functions for neurons.

Communication between neurons occurs through electrical impulses, which are generated by the opening or closing of ion channels in response to certain stimuli. This results in a disturbance of the resting transmembrane potential (approximately -70 mV), and three main phases can be distinguished:

- Hyperpolarization: the transmembrane potential reaches values lower than the resting value
- Depolarization: the transmembrane potential increases compared to the resting value
- Repolarization: the transmembrane potential returns to the resting value

If the electrical stimulus is hyperpolarizing, inhibition occurs; if, on the other hand, it is depolarizing and exceeds a critical threshold of the transmembrane potential, excitation occurs and the action potential is generated, which will propagate along the entire axon of the neuron through saltatory and unidirectional conduction between the nodes of Ranvier. The action potential is then transmitted to the dendrites of neighbouring neurons through synapses, points where the membranes of adjacent neurons are extremely close together.

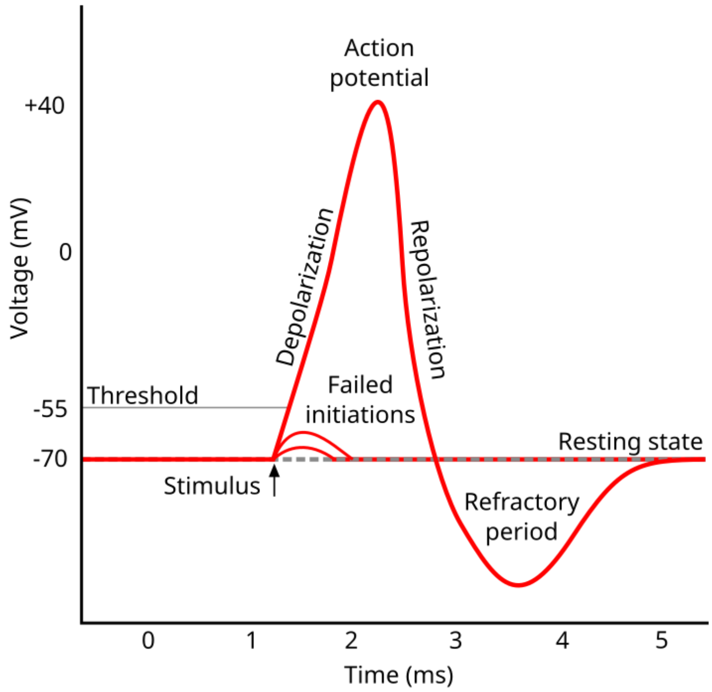


Figure 2: Action Potential

Synapses are essentially divided into electrical and chemical synapses. The latter represent the majority in the human nervous system and involve the release of chemical substances, neurotransmitters, which are released when an action potential propagates and depolarizes the axon membrane. They subsequently diffuse into the synaptic cleft and bind to specific receptors on the membranes of the dendrites of adjacent neurons, causing the generation of excitatory or inhibitory postsynaptic potentials, depending on the neurotransmitter and receptor involved.

1.1.2 Fundamentals of Electroencephalography

The human brain contains approximately 80×10^9 neurons, each of which forms on average 10^4 synapses with other neurons. This results in a vast and extremely complex network of neurons, whose electrical activity can be recorded [2].

Electroencephalography (EEG) is the most widely used non-invasive method for this purpose: it achieves high temporal resolution and allows electrical signals to be recorded at the level of the cerebral cortex through the use of electrodes (typically in AgCl surfaces) placed on the scalp. Since the potential decays in space as $1/r$, the activity recorded by each electrode is not related to the action potential of individual neurons but is the result of two contributions [3]:

- Spatial contribution
  Each electrode captures the activity of a population of neurons beneath it, generating a spatial average that depends both on the electrode surface (typically $10\ cm^2$) and on the conductive properties of the intervening tissues (skull, skin, connective tissue), which distort and attenuate the original signal.
- Temporal contribution
  Only synchronous neuronal activity is recorded; different brain states will acquire signals with different amplitudes. States associated with low frequencies tend to generate greater amplitudes: the lower the frequency, the larger the time window corresponding to the spectrum, therefore it is more likely to find the synchronous activity of more neurons than time windows relating to higher frequencies.

Although higher frequencies can be measured, most of the signal power is concentrated in one band [0.1–40 Hz], which is conventionally divided into five sub-bands [2]:

- Alpha (8–13 Hz)
  This was the first to be studied and analysed and corresponds to a relaxed state of wakefulness, with maximum amplitude with eyes closed. Alpha waves in sensorimotor areas are called Mu waves.
- Beta (13–30 Hz)
  Corresponds to states of arousal and concentration.
- Gamma (> 30 Hz)
  Linked to strong states of arousal and more complex cognitive processes.
- Delta (0.1 – 4 Hz):
  Relates to states of deep sleep or pathological states such as coma or loss of consciousness.
- Theta (4 – 8 Hz)
  Relates to states of light sleep or meditation.

Despite its usefulness and widespread use, EEG is susceptible to numerous artifacts. First of all, it is necessary to consider electromagnetic interference from external sources (such as power line) but also from internal sources (electromyogram, electrocardiogram, electrooculogram). Other artifacts are instead attributed to the instrumentation used (not perfect interface between electrodes and skin, noise of the amplifiers used, quantization noise of the analog-digital converter). The first artifacts are attenuated by the use of digital filters, while for the second ones good quality instrumentation is recommended [2].

1.1.3 Recording Techniques

In order to compare different studies, it is essential to use standardized electrode positions; the 10-20 system is the most widespread. It begins by defining four landmarks: nasion, inion, and the right and left preauricular points. The electrodes are then positioned so that adjacent ones are 10% or 20% of the distance between the landmarks; This way, a total of 21 electrodes can be used. Each electrode's position is identified by a letter and number:

- The letters F, P, T, and O indicate the associated lobe (frontal, parietal, temporal, and occipital, respectively).
- C and Z are used as references for the centre.
- Even numbers refer to the right hemisphere, while odd numbers refer to the left hemisphere.

One of the major limitations of EEG is its low spatial resolution, which is why this system can be extended to others that allow for the placement of multiple electrodes, reducing the inter-electrode distance:

- 10-10 system: allows the placement of up to 74 electrodes.
- 5-10 system: allows the placement of up to 345 electrodes [4].

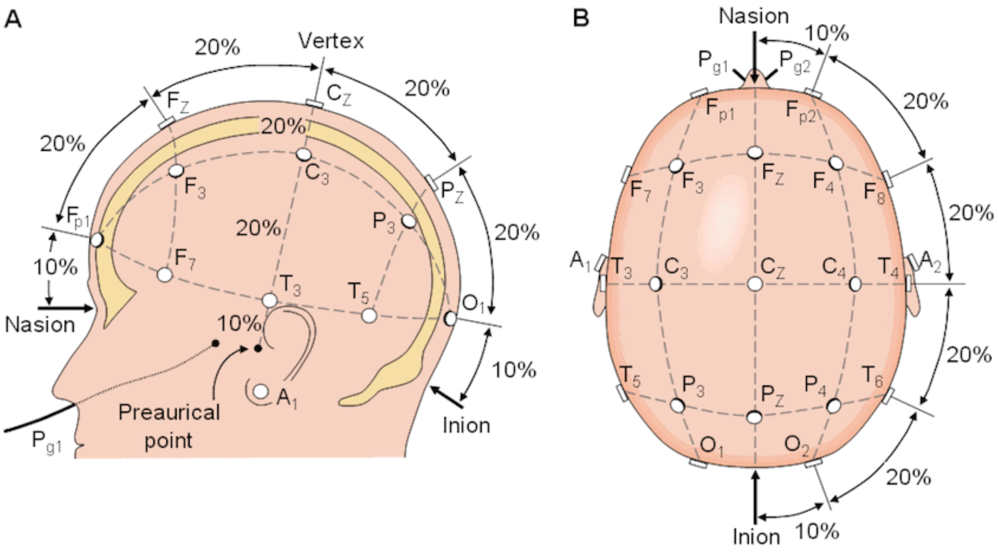


Figure 3: 10-20 System

1.1.4 EEG-based Brain Computer Interfaces

In non-invasive Brain-Computer Interface (BCI), EEG is the most widely used acquisition technique due to its low cost, portability, and high temporal resolution.

Each BCI system essentially consists of acquisition and processing of brain signals, feature extraction, classification and translation into commands to control an external device, to allow people affected by neuromotor pathologies to be able to interact with the external world again [5]. EEG is suitable for this purpose because it allows for the detection of dynamic variations in cortical activity related to motor intentions, sensory perception, or cognitive processes.

EEG-based BCIs can be divided into two main categories based on the type of signal recorded [2]:

- Spontaneous EEG

  Refers to brain activity recorded in the absence of external stimuli. The most widely used paradigm in this context is Motor Imagery (MI). It consists in imagining the execution of movements (typically of the limbs) without actually executing them, inducing a modulation of the oscillatory activity of the alpha and beta rhythms in the sensorimotor cortex [2] [5].
- Event-Related Potentials (ERP)

  These are brain activity that occurs in response to a specific external stimulus, whether sensory, cognitive, or motor. The amplitudes of these potentials are very low compared to those of spontaneous EEG signals; However, since ERPs are time-locked events, it is possible to improve the Signal-to-Noise Ratio by averaging, which is not possible for continuous EEG [2]. Among the main paradigms used for this type of signals are the Steady State Evoked Potentials (SSEP) and the P300.

- SSEP

  The subject is stimulated with periodic sounds or flashing images at a specific frequency, and it is possible to record brain activity that contains harmonic components at that specific frequency [5].

- P300

  It is a positive component that appears with a latency that can vary from 250 to 500 ms after the stimulus [2]. It generally occurs in the Oddball Paradigm [6]: a series of stimuli is presented to the user, the majority of which belong to one class, while the target stimulus belongs to another, rarer class; the target stimulus will trigger the formation of this potential. The amplitude depends both on the intervals between one stimulus and the next and on the rarity of the target stimulus.

1.1.5 Feature Extraction in EEG Data

Following acquisition and preprocessing, feature extraction is a crucial step to reduce the dimensionality and complexity of the data. In this paragraph the main feature extraction techniques for EEG signals are listed [7].

- Time Domain Features

  These are the simplest to extract and include signal features over time (amplitude, zero-crossing rate), features derived from descriptive statistics (mean, variance, standard deviation, skewness, kurtosis) and entropy measures, which quantify the signal's uncertainty and complexity.

- Frequency Domain Features

  These include Fourier Transform and Power Spectral Density (PSD) of the five bands into which the EEG signal can be decomposed.

- Time-Frequency Domain Features

  Provide two-dimensional representations of the signal, highlighting how the spectrum varies over time; they are particularly useful when the signal is non-stationary, as in the case of EEG. Among the most common are the Short-Time Fourier Transform (STFT), which produces a spectrogram, and the Wavelet Transform, which consists of decomposing the signal as the weighted sum of a mother wavelet that is appropriately scaled and translated in time.

- Space Domain Features
  The Common Spatial Pattern (CSP), primarily used for binary classifications in Motor Imagery, involves projecting the signal into a space that maximizes the variance between the two classes.
- Deep learning (DL) techniques
  Are an alternative to the classical features extraction methods which allow the automatic extraction features from signals, bypassing manual extraction and the related feature engineering

In the next section, semantic features will be defined, a new and promising category of features that could prove essential for future applications of Brain Computer Interface and Brain to Brain Communication.

1.2 Semantic Features

1.2.1 Definition of semantic features

Semantic features are high level, abstract and coherent neural representation that the brain uses to encode and differentiate a concept, that is, those mechanisms that represent the meaning of something based on our personal experiences and not on our perception (for example, through colour, shape, sound) [8].

A challenge in research is to define what properties a neural representation must possess in order to be considered a semantic feature. Ahmadi and Mesin [13] have proposed four requirements:

1. Task independence
   It must remain applicable across different tasks, stimuli, and conditions without relying on task-specific labels or supervised models.
   In this regard, Simanova et al. [14], in an fMRI study, were able to decode semantic categories regardless of the type of stimulus modality with which a concept was presented: spoken and written names, photographs and natural sounds. A model capable

of classifying two semantic categories (animals and tools) was trained, obtaining significant accuracies both for each separate modality and across different modalities, also allowing to localize the brain regions involved in the decoding of the concepts.

2. Robustness to inter-subject variability

   Semantic features vary from person to person due to [8]:

   - Neuroanatomical differences, including tissue impedances and cranial geometry
   - The neural organization for accessing concepts is individual
   - Non-stationarity and neuroplasticity: neural representations can vary even between different sessions for the same subject

3. Scalability and generalizability

   Efficiently scale to large datasets and generalize to new and unseen data with minimal performance degradation.

   As for EEG, different datasets come from different acquisitions; A semantic feature must not depend on how many EEG channels were used for the acquisition, on the length of the acquisitions, nor on noise and artifacts caused by different sources and laboratory conditions.

   For example, we mention two recent studies [15, 16] that consisted in CSP as extracted features and involved the use of ensemble models, consisting of several stacked classifiers for the classification of Motor Imagery signals. Although these models were trained on multiple datasets, achieving near-perfect classifications, applying them to new datasets related to the same paradigm showed a drastic drop in performance, leading to the hypothesis that the CSP does not allow the extraction of semantic features.

   A further study [16], consisted of the binary classification between imagination and perception in different stimulus modalities for a single dataset; semantic features were extracted from EEG topomaps using a CNN model with attention mechanisms. Excellent performance was achieved, taking into account that the signals came from different perception and imagination tasks.

   Finally, a novel and promising model has been introduced [13]: it consists of a combination of CNN, Autoencoder and Transformer for the universal extraction of semantic features independent of the task and the paradigm type. For training and validation, several EEG datasets from different paradigms (MI, ERP, SSVEP) were used, with a training approach based exclusively on signal reconstruction. Post-training classifications revealed that the extracted features allow discriminating between the various classes.

4. Interpretability
   Since semantic features represent higher level cognitive processes, they should provide information useful for understanding the underlying neural dynamics.
5. Compatibility with downstream analysis: can be analysed with both supervised and unsupervised models. Supervised models allow for higher classification accuracies, crucial for BCI applications, but are also highly dependent on the type of task.

1.2.2 Research overview and possible future developments

The existence of a system dedicated to semantic processing and distinct from the perceptual one was highlighted by a pioneering study in cognitive neuroscience [9]: it was shown that some neurological patients were able to describe the perceptual characteristics of inanimate objects without being able to access their conceptual meaning.
Despite this difference between the two systems, it is crucial to ensure that the features extracted by a neuroimaging system are effectively semantic and not confused with low-level perceptual properties. For example, when studying these representations through the use of visual stimuli, it will be necessary to take into account that visual perception will be used to access semantic information [10]. A research based on the study of semantic coding from EEG-based ERPs [11] following three different types of stimuli, through multivariate analysis, highlighted how the channels that contributed most to decoding were those related to the early visual components; furthermore, a later contribution was found relating to the N400, an ERP component commonly associated with semantic processing.

Semantic features must also distinguish between different cognitive states. Some studies have investigated how specific neural patterns are involved in visual imagination and perception. Neurological patients showing dissociation between these two processes have confirmed that these are two clearly distinct states [19]. Nevertheless, overlaps have been highlighted between the brain areas activated for the two states. For example, in EEG, a sharing of neural dynamics in the alpha band has been discovered for the parieto-occipital area, although with different timing (these common representations emerge later in imagination) [18].

As for possible future applications, it is worth mentioning semantic communication and Brain-to-Brain communication; these features would allow data transmission no longer based on bits but exclusively on their meaning, eliminating redundant and unimportant information, while maintaining transmission accuracy [12].

Since semantic features are high-level latent characteristics, their extraction requires advanced methods beyond traditional ones. Deep neural networks, discussed in the next section, represent promising approaches to automatically learn these representations from raw EEG data.

1.3 Deep Learning and Neural Networks for EEG Analysis

In this section, a brief introduction to Deep Learning and the Neural Networks used in this work will be given.

1.3.1 Deep Learning and learning approaches

Deep Learning (DL) is a subset of Machine Learning (ML) that enables the learning of representations of raw data across different levels of complexity. The flexibility of these algorithms and the fact that they achieve state-of-the-art performance has led to a significant increase in their use for EEG data in recent years. Furthermore, DL algorithms allow for automatic feature extraction, which is not feasible with traditional ML methods. On the other hand, DL algorithms require large amounts of training data to achieve optimal performance. Another limitation, especially for use in the medical field, is related to the difficult interpretability of the extracted features, effectively rendering them black boxes. To address this limitation, explainable AI techniques are being developed recently.

DL algorithms can follow different training approaches:

- Supervised learning
  Labelled data is used; this improves performance but requires domain experts for manual labelling and a significant time investment.
- Unsupervised learning
  Models discover hidden features using unlabeled data.
- Semi-supervised learning
  This is a hybrid of the two above, used when not all data is labelled.
- Self-supervised learning

This is used for models trained on large datasets, when it is not possible to label all the data. It involves using techniques that create pseudo-labels directly from the data itself, without human supervision.

Deep Learning is based on deep Artificial Neural Networks (ANN), whose architecture takes inspiration from the functioning of anatomical neurons connected to adjacent ones through synapses. Each neuron receives a certain number of inputs, each of which is associated with a weight and the weighted sum is performed with the possible addition of a bias. This information is then processed through a nonlinear activation function and transmitted as an output to the other neurons.

Training an ANN involves propagating the input data within the network; the output prediction error is calculated, and backpropagation continues, allowing the various weights and biases to be updated.

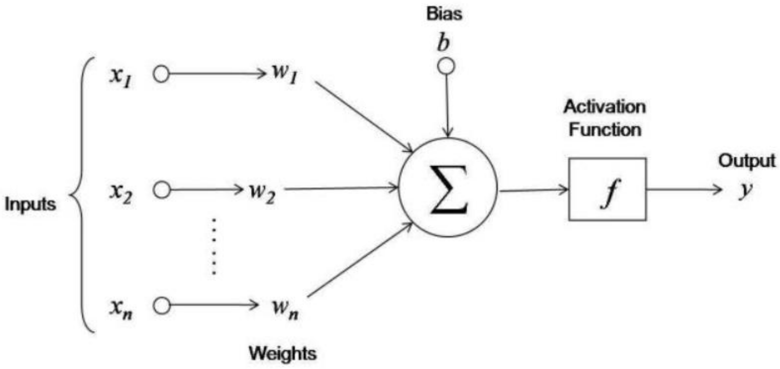


Figure 4: Artificial Neural Network architecture

1.3.2 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are special types of ANNs that use convolution operations to extract features from signals. Convolution consists of a filter, also called a kernel, that runs through the entire signal to extract local and hierarchical features. The results of the convolutions, called feature maps, are nonlinearized with the activation function. Next, the Pooling layer acts as a feature selector and reduces the dimensionality by eliminating unnecessary information. The final building block consists of a Fully Connected layer, where each neuron is connected to all those in the previous state. This layer is used to aggregate the features extracted in the previous layers to obtain a classification.

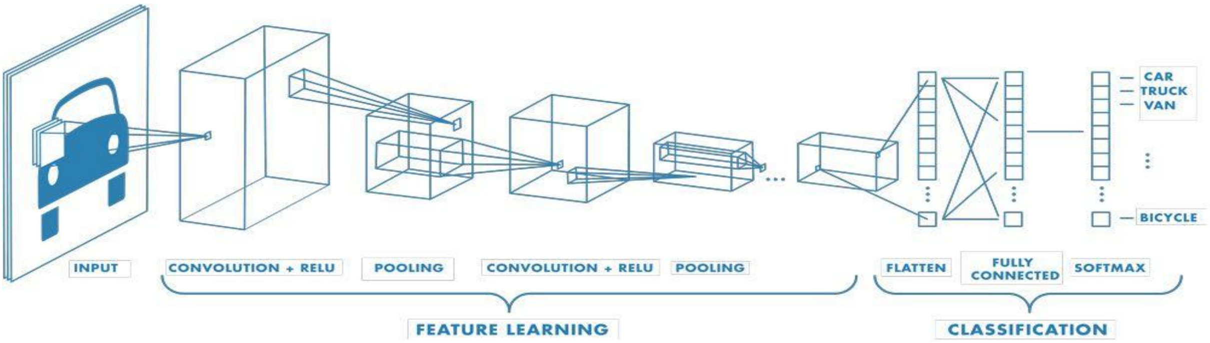


Figure 5: CNN architecture

Although initially born for image processing, CNNs are widely used in EEG analysis for classification due to their versatility and their ability to extract temporal, spatial and spatiotemporal features [20, 21].

1.3.3 Autoencoders

Autoencoders (AEs) are ANNs used to learn latent representations of data in an unsupervised manner. They are characterized by two components, typically symmetrical:

- Encoder: It reduces the dimensionality of the raw input data, progressively compressing it into a latent dimension.
- Decoder: reconstructs the input data from the latent dimension, thus returning to the initial dimensions.

The training goal is therefore to minimize the reconstruction error so that the latent space can capture salient information from the input [22].

For EEG analysis, AEs are widely used for dimensionality reduction, feature learning, and denoising. The latter involves corrupting the input with noise and forcing the decoder to reconstruct the clean data.

1.3.4 Long Short-Term Memories

Long Short-Term Memories (LSTMs) are a variant of Recurrent Neural Networks (RNNs) designed to capture long-term temporal dependencies and mitigate the vanishing/exploding gradient problem. These types of ANNs are used to process sequential data by using an internal memory that takes into account inputs from previous moments in the sequence. For this reason they are effective for EEG signals, given their non-stationary temporal nature.

The architecture of an LSTM consists of a long-term memory cell (cell state) and three gates that allow information to be added or deleted from the cell state, regulating its flow. Forget Gate is responsible for eliminating information from the cell state, Input Gate allows new information to enter the cell state and Output Gate determines which cell state information should be propagated to the output

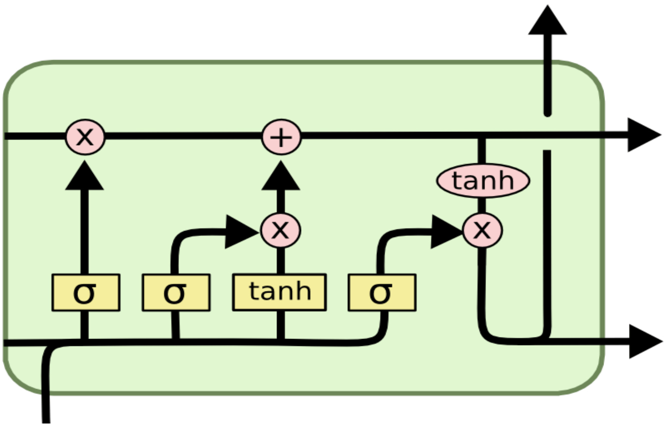


Figure 6: LSTM Architecture

1.3.5 Transformers

Transformers are models based exclusively on the self-attention mechanism, doing away with recurrence and convolutions [23] initially used for Natural Language Processing but also used for time series analysis.

A Transformer essentially consists of:

- Embedding of the input sequence to which a Positional Encoding is added, which allows to inject information relating to the position of the elements in the sequence.
- Self-Attention
  It is a mechanism that allows to process similarity calculations and weights the importance of each element of the sequence with respect to itself and to all the others, managing to capture global dependencies [23].
  Self-attention is performed in parallel multiple times (Multi-Head Attention) with different projections to allow the model to capture different global dependencies.

- Normalization and Feed-Forward Network, which consists of Fully Connected layers with nonlinear activation function.
- Residual connection, which allows connecting layers of the network that are not consecutive.

Since Transformers, unlike networks such as CNNs, are able to capture global dependencies between channels and time, they are being used in EEG analysis with excellent results; however, they present greater complexity in terms of weights and training time than traditional neural networks and require large datasets to be trained effectively.

2 Materials and Methods

This chapter describes the dataset used for this study, the preprocessing pipeline, the model architecture, their training, and the methods used for their validation.

All phases of the study were implemented in a Python environment. MNE-Python library [24] was used for the preprocessing part, while Scikit-learn library [25] was used for the validation part. The model training was performed with the PyTorch 2.9.1 framework [26] with an NVIDIA A40 GPU available in the Legion Cluster of the Politecnico di Torino [27].

2.1 Dataset

This study used the EEG-based BCI Dataset of Semantic Concepts for Imagination and Perception tasks from the University of Bath [28], specifically proposed for studies on BCI and cognitive neuroscience, given the scarcity of similar open-source datasets.

It consists of the acquisition of EEG signals from 12 subjects, three of whom performed two acquisition sessions. Furthermore, a subject had both visual and auditory impairments (sub-16).

In addition to the signal acquisitions, the subjects completed two questionnaires relating to the subjective vividness of auditory and visual imagery: the Vividness of Visual Imagery Questionnaire (VVIQ) [29] and the Buckell Auditory Imagery Scale (BAIS-V) [30].

The experiment took place in a dark, soundproof room and consisted of EEG acquisition during the perception and subsequent imagination of three semantic concepts: guitar, flower, and penguin. Signals were acquired from 124 channels at a sampling rate of 1024 Hz with 24-bit resolution. The electrodes were mounted using the five percent system.

Subjects perceived semantic concepts in three different stimuli:

- Visual pictorial
  It consisted of projecting a colored image of the semantic concept onto a screen over a black background, which could vary in complexity (simple, medium, naturalistic) for a duration of 3 seconds.
- Visual orthographic
  The name of the concept was displayed on the screen, which could vary in font and text color, for a duration of 3 seconds.

- Auditory comprehension

  It consisted of hearing a speech that named the concept in a tone of voice that could be normal, low or high.

Immediately after perception, subjects were asked to imagine the semantic concepts for a duration of 4 seconds, regardless of the stimulus type.

The three semantic concepts were chosen based on the same number of syllables, 2, and the same semantic distance in the Word2Vec latent space. They were also chosen because they were uncommon objects in everyday life, so as not to introduce external biases.

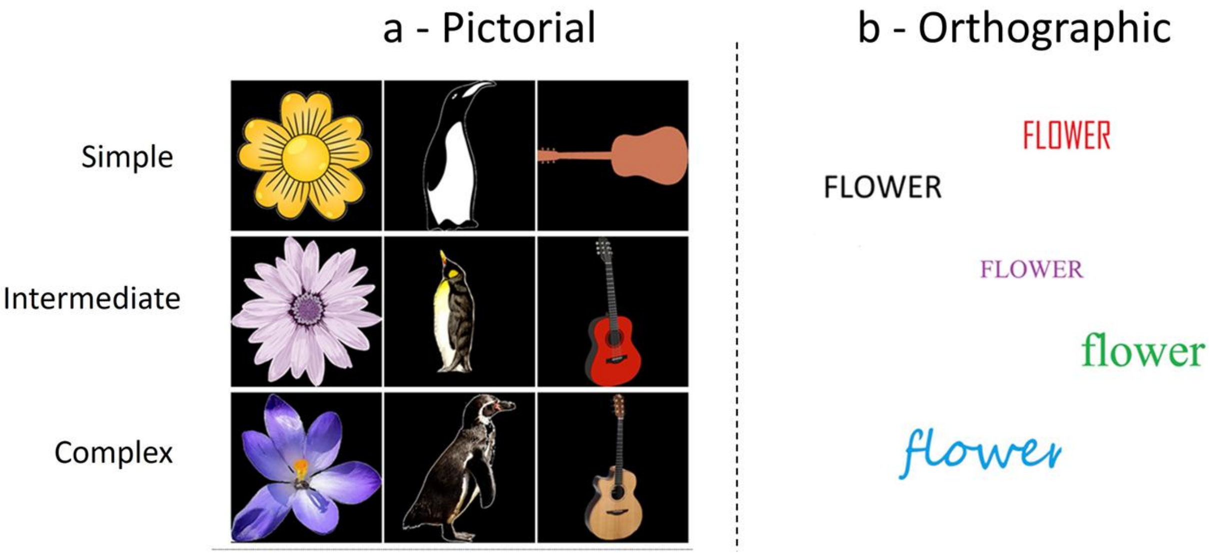


Figure 7: Examples of the visual (a) pictorial and (b) orthographic stimuli used in the experiment. Pictorial stimuli ranged in complexity from simple to intermediate to naturalistic, while orthographic stimuli varied in colour and font. [28]

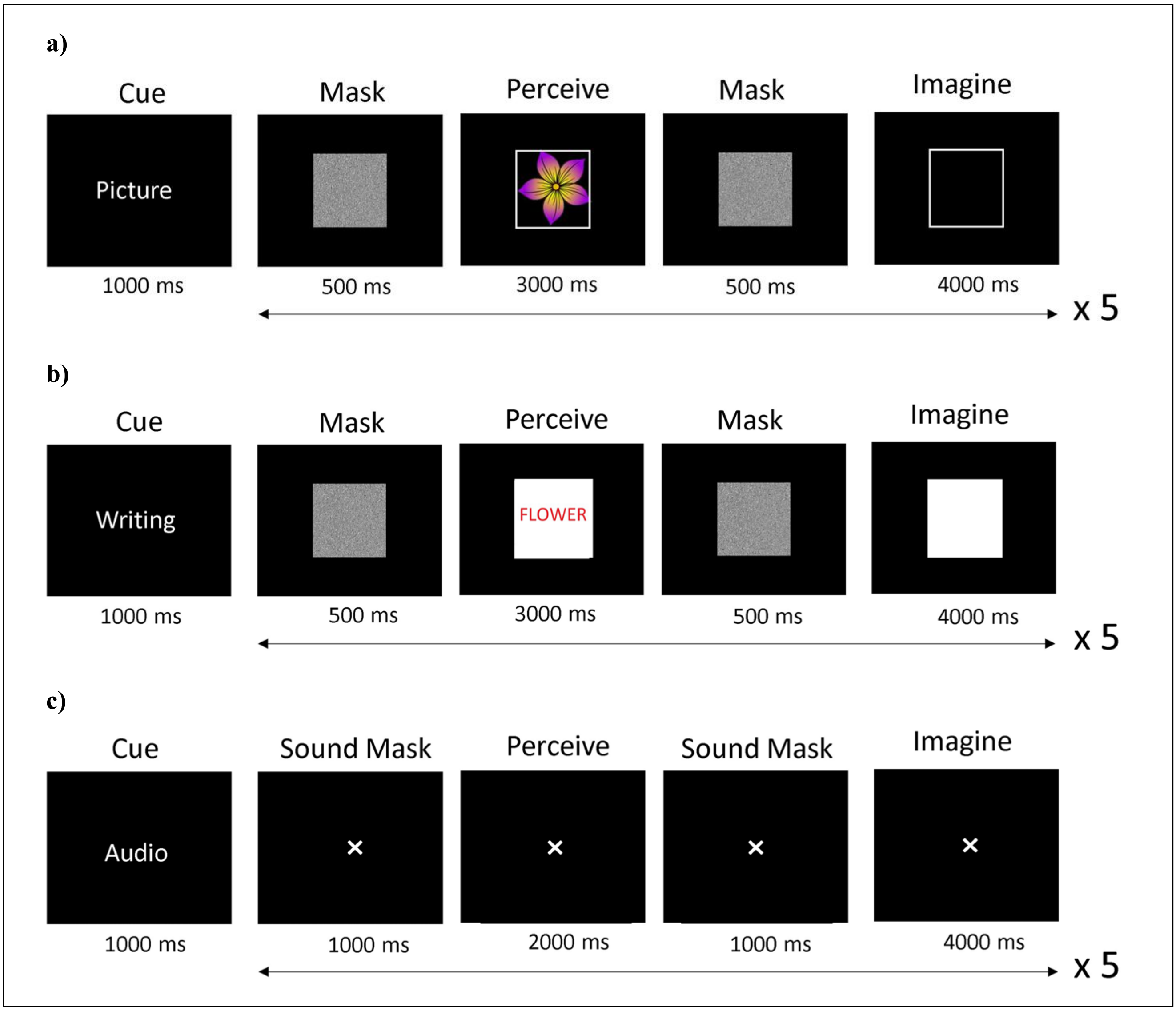


Figure 8: Examples of the structure of the various trials; a) visual pictorial, b) visual orthographic, c) auditory [28]

2.2 Preprocessing

2.2.1 Resampling and Filtering

First, it was chosen to downsample the raw EEG signals from 1024 Hz to 128 Hz to reduce computational costs. An anti-aliasing filter was not necessary because the effective bandwidth of the signals was lower than 64 Hz.

The signals were then filtered with a 50 Hz notch filter to attenuate power line interference, followed by a zero-phase bandpass FIR filter with cutoff frequencies of 0.1 Hz and 40 Hz to remove high-frequency noise and low-frequency drifts.

2.2.2 Interpolation of bad channels

For each recorded session, the presence of bad channels, characterized by low signal-to-noise ratios or large amplitudes related to high impedances, probably due to an incorrect interface between the electrode and the skin, was visually noted. For this reason, the PyPrep pipeline [31] was used: it applies a robust reference and automatically identifies bad channels to subsequently interpolate them.

A channel is identified as bad if it presents extreme amplitudes, poor correlation with other channels, poor predictability from other channels, and unusually high noise [32]; once identified, these channels are interpolated by spherical splines.

2.2.3 Eye artifacts removal

The removal of ocular artifacts (eye blinks and eye movements) was performed by using Independent Component Analysis (ICA), a source separation technique that assumes that a signal is a linear combination of statistically independent component, , which are estimated by maximizing non-Gaussianity.

EOG channels were not present in the dataset, so the independent components related to ocular artifacts to be excluded were those that best correlated with the channels closest to the eyes, Fp1 and Fp2.

2.2.4 Epochs segmentation

Since the task length differs across paradigms, the raw signals were divided into 4-s long epochs for imagery tasks and 2 s and 3 s long epochs for auditory and visual perception tasks, respectively. However, since the neural network architectures used in this study require the use of data of the same length, it was necessary to zero-pad the perception epochs to 4 s.

For each epoch, a metadata was created, containing information on: subject number, semantic concept (guitar, flower, penguin), task type (imagination, perception) and event type (e.g. visual imagination of a penguin); this metadata were not used for models training, which is unsupervised, but only for data augmentation and for the clustering phase.

2.3 Data Preparation

2.3.1 Dataset split and Data Augmentation

The dataset was split into training set (81% of the total epochs), validation set (9% of the total epochs) and test set (10% of the total epochs).

Since acquiring EEG data is time-consuming and Deep neural networks require large amounts of data to stabilize the training of their large number of parameters and avoid overfitting, it was necessary to resort to data augmentation. Classic data augmentation techniques (e.g., adding Gaussian noise, dropping channels, or clipping portions of the signals) may alter the neurophysiological characteristics of a cognitive state, so it has been opted to create synthetic epochs from those present in the training set as done by Olawunmi et al. [33].
For each sample of the training set, 8 epochs relating to the same subject and the same task type were randomly extracted. Each pair (nth epoch, extracted epoch) was then averaged, thus generating 8 new synthetic epochs.
In this way, it was possible to increase the number of samples in the training set by 9 times.

2.3.2 Winsorization and Normalization

Another way to stabilize training and make data samples comparable is to normalize them, limiting them to a defined range of values. In this case, Z-score normalization was used, which consists of transforming the data into a variable with a zero mean and unit standard deviation:

$$Z = \frac{X - \mu}{\sigma}$$

Z-score normalization, however, is susceptible to outliers, so before using it, the data was Winsorized to replace the extreme values with those at the 1st and 99th percentiles.

Winsorization and normalization were performed along the channels and their parameters (percentiles, means, standard deviations) were calculated exclusively from the training set, to avoid data leakage; these were then applied to augmented data, validation set, and test set.

2.4 Network Architectures

In this work, two Convolutional Autoencoder (CAE) models were studied for semantic feature extraction.

Both models were analysed in their purely convolutional form and subsequently in variants consisting of the integration with modules used for modelling time sequences, LSTMs and Transformers. In total, six different architectures have been implemented and studied, which will be explored in more detail in this section.

2.4.1 Model A

The first architecture is based on the CNN Autoencoder used in the work of Ahmadi and Mesin [13]. It is designed as a compact architecture, consisting of 1D convolutional blocks which combine the spatial features related to the EEG channels with the local ones extracted by the convolutional filters, allowing a good compromise between a relatively low number of trainable parameters and an adequate signal reconstruction capacity.

2.4.2 Model B

The second model integrates the first by adding purely temporal and spatial convolutions. The former apply temporal filters that extract features related to the different frequency bands that compose the EEG signal, while the latter apply spatial filters to capture the relationships between different brain areas. Increasing the number of convolutional blocks results in a more complex architecture that allows for the extraction of more abstract and deeper features, but at the expense of higher vulnerability to overfitting with small datasets.

2.4.3 Purely Convolutional Autoencoder (CAE)

The combination of an Autoencoder with convolutional blocks allows for the hierarchical extraction of local spatial and temporal features as the network depth increases.

The encoder applies convolutional filters to the EEG signal, projecting it into a latent space, with the aim of extracting relevant features. The decoder gradually reconstructs the signal from this latent representation, striving to obtain an output as similar as possible to the input.

Generally, the latent representation has a lower dimensionality than the input; however, it was decided not to use Pooling as it was noted that it led to worse performance.

Architectures of purely CAE models are showed in figure 9.

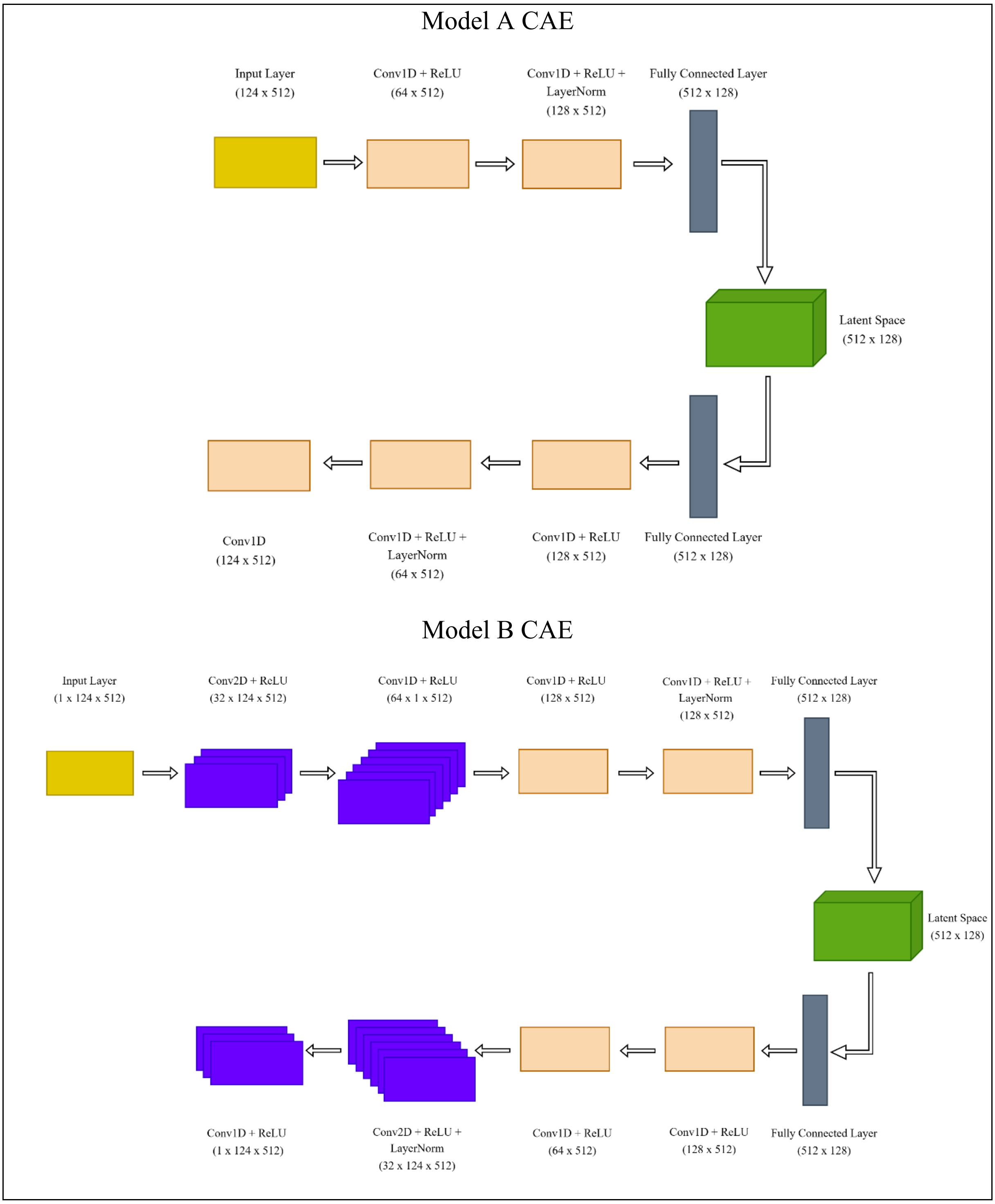

Model A CAE
Input Layer (124 x 512)
Conv1D + ReLU (64 x 512)
Conv1D + ReLU + LayerNorm (128 x 512)
Fully Connected Layer (512 x 128)
Latent Space (512 x 128)
Fully Connected Layer (512 x 128)
Conv1D + ReLU (128 x 512)
Conv1D + ReLU + LayerNorm (64 x 512)
Conv1D (124 x 512)
Model B CAE
Input Layer (1 x 124 x 512)
Conv2D + ReLU (32 x 124 x 512)
Conv1D + ReLU (64 x 1 x 512)
Conv1D + ReLU (128 x 512)
Conv1D + ReLU + LayerNorm (128 x 512)
Fully Connected Layer (512 x 128)
Latent Space (512 x 128)
Fully Connected Layer (512 x 128)
Conv1D + ReLU (128 x 512)
Conv1D + ReLU (64 x 512)
Conv2D + ReLU + LayerNorm (32 x 124 x 512)
Conv1D + ReLU (1 x 124 x 512)

Figure 9: Representation of architectures for purely convolutional variants.

2.4.3.1 Model A CAE

The Encoder consists of two 1D convolutions, composed of 64 and 128 filters respectively and a ReLU activation function; then Layer Normalization [34] along channel dimension is applied to the feature maps. Although CNNs achieve better performances and faster convergence with Batch Normalization, it was decided not to use it to maintain consistency with other architectures (LSTM and Transformer) that benefit from Layer Normalization when used for temporal series.

The feature maps of the convolutions are projected into the latent space through a 128-unit FC layer, obtaining embeddings of dimension (512, 128) that will be extracted in the evaluation phase for clustering.

The decoder is symmetric to the encoder and consists of a 128-unit FC layer, two 1D transposed convolutions with 128 and 64 filters, respectively, followed by ReLU activations. Next, another Layer Normalization and a final convolution with 124 channels are applied to reconstruct the original signal.

2.4.3.2 Model B CAE

The CNN Encoder is composed of two 2D convolutions and two 1D convolutions. The first is a 2D temporal convolution that consists of 32 temporal filters and a ReLU activation function. The second 2D convolution applies 64 spatial filters to each instant of the sequence and ReLU activation. Subsequently, two 1D convolutions with 128 channels each and ReLU activation are applied. As for model A, the feature maps are layer-normalized and are then mapped into the latent space via a FC layer to get the embeddings of dimensions (512, 128).

The Decoder, mirroring the Encoder, features an additional FC layer, two transposed 1D convolutions with 128 and 64 filters respectively, and a transposed spatial convolution composed of 32 filters. Layer normalization is applied and a final temporal convolution with a single filter reconstructs the signal.

2.4.4 Convolutional Autoencoders + Long Short-Term Memory (CAE + LSTM)

EEG is a temporal signal that, in addition to local features, also exhibits dynamics that evolve throughout the entire sequence, which CAE alone would be unable to capture. For this reason, a LSTM is introduced, which allows modelling long-term temporal features over latent representations through gating mechanisms that regulate the flow of information in the cell state.

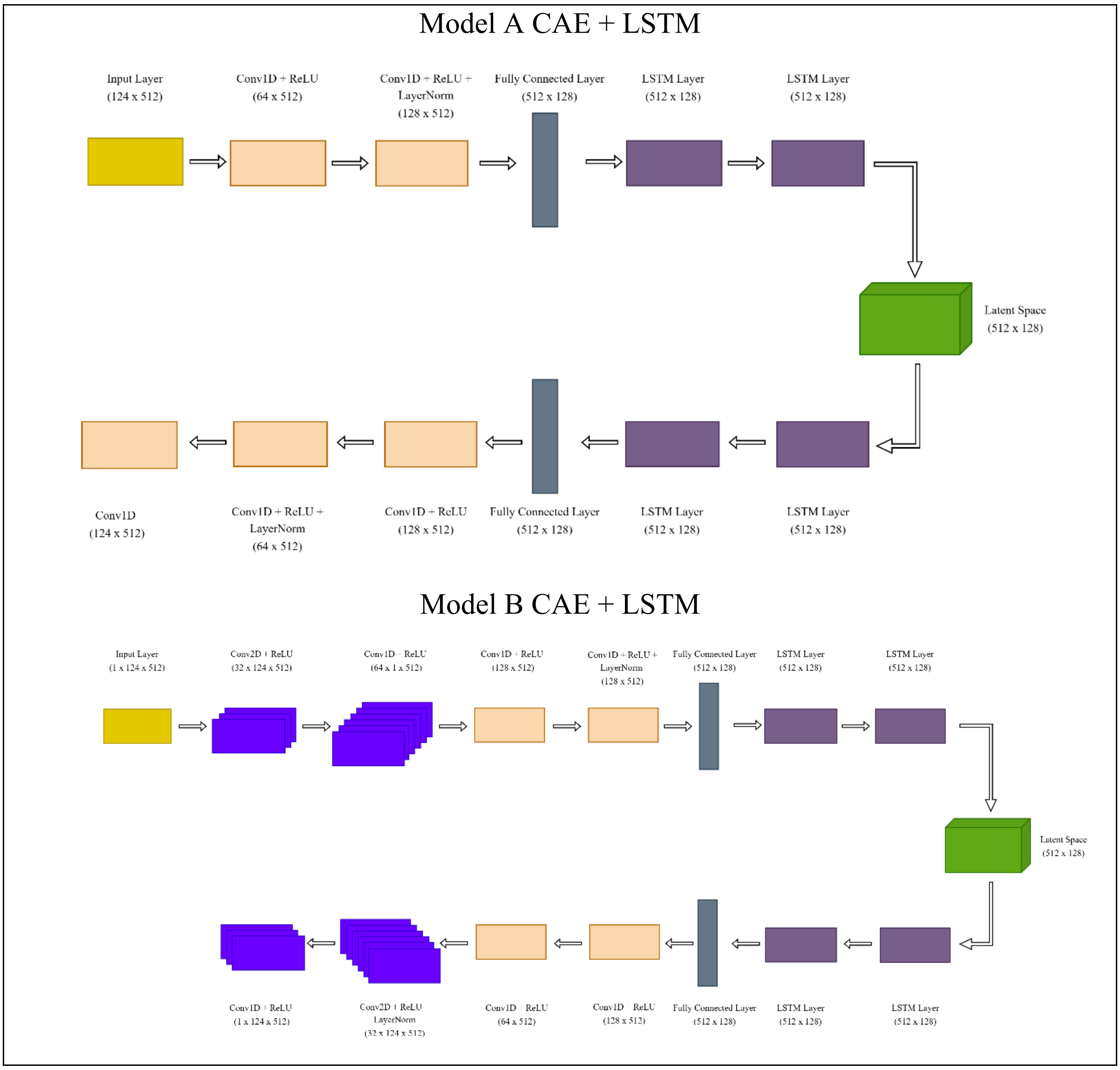

Model A CAE + LSTM
Input Layer (124 x 512)
Conv1D + ReLU (64 x 512)
Conv1D + ReLU + LayerNorm (128 x 512)
Fully Connected Layer (512 x 128)
LSTM Layer (512 x 128)
LSTM Layer (512 x 128)
Latent Space (512 x 128)
Conv1D (124 x 512)
Conv1D + ReLU + LayerNorm (64 x 512)
Conv1D + ReLU (128 x 512)
Fully Connected Layer (512 x 128)
LSTM Layer (512 x 128)
LSTM Layer (512 x 128)
Model B CAE + LSTM
Input Layer (1 x 124 x 512)
Conv2D + ReLU (32 x 124 x 512)
Conv1D + ReLU (64 x 1 x 512)
Conv1D + ReLU (128 x 512)
Conv1D + ReLU + LayerNorm (128 x 512)
Fully Connected Layer (512 x 128)
LSTM Layer (512 x 128)
LSTM Layer (512 x 128)
Latent Space (512 x 128)
Conv1D + ReLU (1 x 124 x 512)
Conv2D + ReLU + LayerNorm (32 x 124 x 512)
Conv1D + ReLU (64 x 512)
Conv1D + ReLU (128 x 512)
Fully Connected Layer (512 x 128)
LSTM Layer (512 x 128)
LSTM Layer (512 x 128)

Figure 10: Representation of architectures for LSTM-based variants.

Figure 10 shows how the LSTM modules are inserted into the architectures. The first block, characterized by two layers with a hidden dimension of 128, models the temporal dynamics of the latent features; subsequently, a block with the same configuration is present in the decoder for signal reconstruction.

2.4.5 Convolutional Autoencoders + Transformer (CAE + Transformer)

Transformer represents an alternative to LSTMs for modeling time series through the self-attention mechanism, allowing for the capture, in addition to the local features extracted from the CAE, of global dependencies on the latent sequences encoded by the convolutions. To simultaneously capture multiple global dependencies, self-attention is performed by multiple heads in parallel. Self-Attention is defined as:

$$Attention\ (Q, K, V) = softmax\ \left(\frac{Q \cdot K^T}{\sqrt{d_k}}\right) \cdot V$$

where:

- Q (Query) represents the current input element.
- K (Key) represents all the elements of the sequence that will be compared to the Query to determine their relevance.
- V (Value) represents the actual information of each element.
- d_k is the size of the features representing each element of the sequence and is used as a scaling factor to stabilize the training.

Since the Transformer does not use recurrences, it is necessary to provide information regarding the positions of the various elements that make up the sequence to be analysed. For this purpose, a Positional Encoding defined as follows is added to the input embedding [23]:

$$PE_{(pos,2i)} = \sin\left(\frac{pos}{10000^{2i/d_k}}\right)$$

$$PE_{(pos,2i+1)} = \cos\left(\frac{pos}{10000^{2i/d_k}}\right)$$

where pos is the position and i is the dimension, so to each dimension corresponds a specific sinusoid.

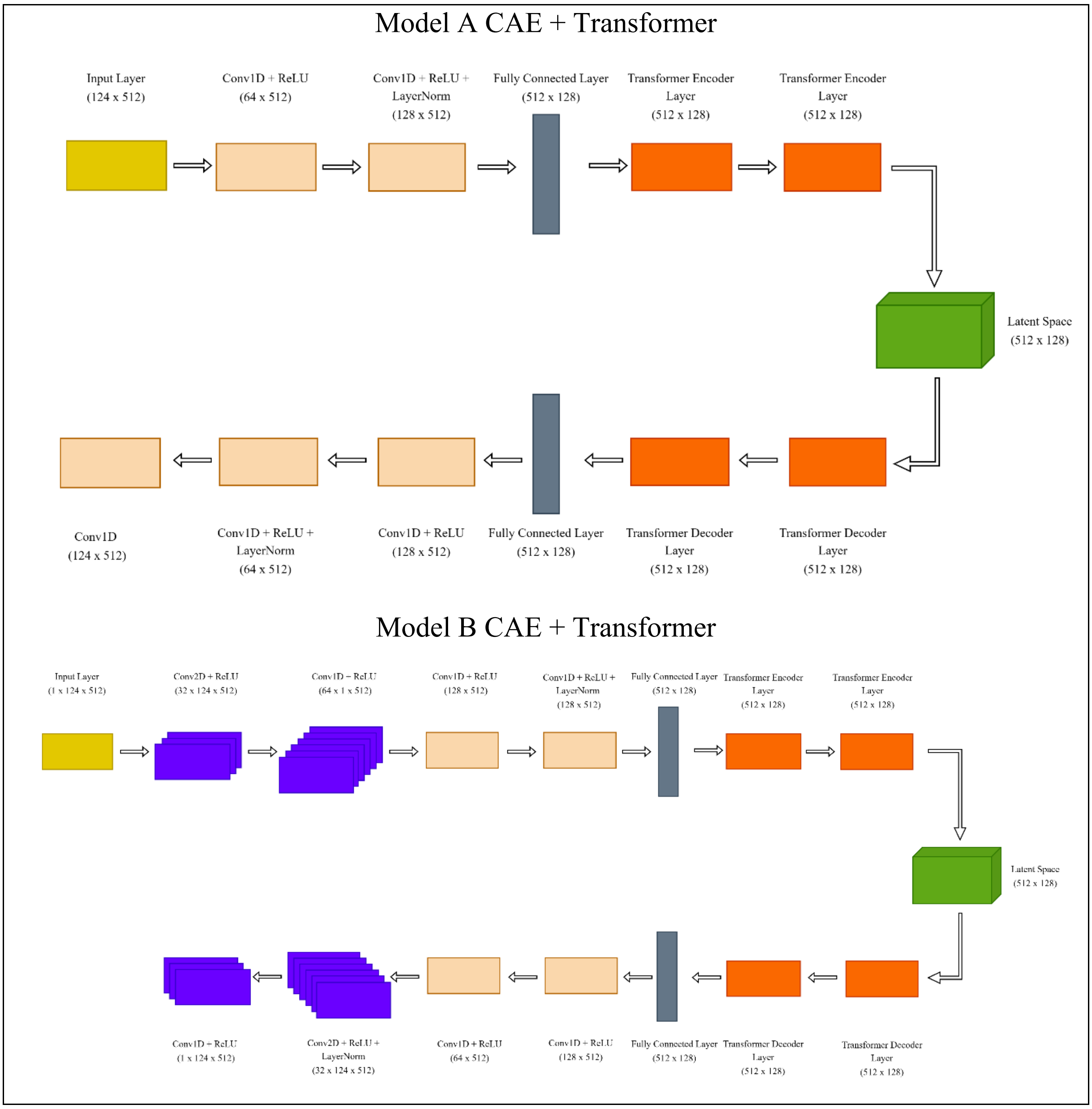

Model A CAE + Transformer
Input Layer (124 x 512)
Conv1D + ReLU (64 x 512)
Conv1D + ReLU + LayerNorm (128 x 512)
Fully Connected Layer (512 x 128)
Transformer Encoder Layer (512 x 128)
Transformer Encoder Layer (512 x 128)
Latent Space (512 x 128)
Transformer Decoder Layer (512 x 128)
Transformer Decoder Layer (512 x 128)
Fully Connected Layer (512 x 128)
Conv1D + ReLU (128 x 512)
Conv1D + ReLU + LayerNorm (64 x 512)
Conv1D (124 x 512)
Model B CAE + Transformer
Input Layer (1 x 124 x 512)
Conv2D + ReLU (32 x 124 x 512)
Conv1D + ReLU (64 x 1 x 512)
Conv1D + ReLU (128 x 512)
Conv1D + ReLU + LayerNorm (128 x 512)
Fully Connected Layer (512 x 128)
Transformer Encoder Layer (512 x 128)
Transformer Encoder Layer (512 x 128)
Latent Space (512 x 128)
Transformer Decoder Layer (512 x 128)
Transformer Decoder Layer (512 x 128)
Fully Connected Layer (512 x 128)
Conv1D + ReLU (128 x 512)
Conv1D + ReLU (64 x 512)
Conv2D + ReLU + LayerNorm (32 x 124 x 512)
Conv1D + ReLU (1 x 124 x 512)

Figure 11: Representation of architectures for Transformer-based variants.

Figure 11 shows the Transformer implementation in both Model A and Model B. The input is projected into the latent space through the FC layer, and the Transformer encoder, featuring two layers with 8 self-attention heads each, extracts global features. Layer Normalization is applied to the self-attention output, and each position is passed through a Feed Forward Network consisting of 512 neurons and a ReLU activation. Finally, a further Layer Normalization and a residual connection are applied, which output the latent embedding from which the semantic features will be extracted.

The Transformer decoder, also composed of 2 layers with 8 self-attention heads each, is responsible for reconstructing the signal and then passing it to the CNN decoder.

Transformers are typically used to generate new sequences by connecting Encoder and Decoder with an additional cross-attention layer. Since the goal in this case is signal reconstruction, the cross-attention layer was not necessary.

2.5 Training Setup

To ensure a fair comparison between all configurations, all models were trained with the same hyperparameters:

- 100 epochs
- Batch size set to 64, for faster trainings than small batches and at the same time more stable
- Learning Rate set to $5 \cdot 10^{-4}$
- Weight Decay set to $1 \cdot 10^{-5}$

The models were trained to minimize the mean squared error (MSE) loss between the original and reconstructed signals:

$$L_{MSE} = \frac{1}{BHW} \sum_{i=1}^{B} \sum_{h=1}^{H} \sum_{w=1}^{W} \left(y_{i,h,w} - \hat{y}_{i,h,w} \right)^2$$

Where B is the batch size, H is the number of EEG channels, and W is the number of time samples.

Weight optimization was performed with Adaptive Moment Estimation (Adam) [35], which adaptively modifies the learning rate for each weight individually, based on the first- and second-order moments of the previous gradients. It enables rapid convergence at the expense of high computational costs.

Validation set was used to monitor the training progress and to prevent overfitting; both the ReduceLROnPlateau and Early Stopping schedulers were implemented. The former reduces the learning rate by a factor of 0.5 if the validation loss has not decreased for 3 consecutive epochs, while the latter terminates training if the validation loss has not decreased for 7 consecutive epochs.

2.6 Semantic Feature Extraction

After training, semantic features are extracted as follows. A signal is encoded in the latent space in an embedding $Z_{i,j} \in \mathbb{R}^{512 \times 128}$, then semantic features for each epoch z_j are obtained as:

$$z_j = \max_{i \in \{1,2,\ldots,512\}} (Z_{i,j}) \in \mathbb{R}^{128}$$

This max pooling approach was used as it gave better results than temporal averaging.

2.7 Evaluation

Evaluation initially consists of assessing the signal reconstruction to verify that the features extracted from the latent space actually represent salient characteristics of the signals.

Further verification of the quality of these latent representations is performed through K-Means clustering, applied in two different phases to verify that they are subject-dependent and can distinguish different cognitive states and/or semantic concepts, regardless of the stimulus type.

2.7.1 K-Means Clustering

K-Means is an unsupervised iterative data clustering algorithm that aims to obtain clusters with minimal intra-cluster variability and maximum inter-cluster variability. First, K centroids are initialized by randomly extracting samples from the dataset, where K is the desired number of clusters. Next, each element of the dataset is assigned to the cluster with the closest centroid, updating the centroids at each iteration by using the mean or median of the elements in the k-th cluster. The process continues iteratively until the centroids move by values less than a certain tolerance or until the maximum number of iterations is reached.

K-Means is initially performed on a cross-subject basis, to verify that the models have learned to extract features containing subject-dependent information. Subsequently, intra-subject clustering is performed to verify that, in addition to subject-dependent characteristics, the extracted features contain high-level information necessary to separate the three semantic concepts (guitar, flower, and penguin) and the cognitive states related to imagination and perception.

The clustering performance, and therefore the semantic feature extraction capacity of the models, were evaluated qualitatively through the visualization of dimensionality reduction techniques (t-SNE and UMAP) and quantitatively through the use of three metrics (ARI, NMI, Silhouette score).

2.7.2 Visualization techniques

They consist of projecting the extracted semantic features from a high-dimensional space into a two-dimensional space through dimensionality reduction, allowing for a qualitative analysis of how separable the clusters are.

- t-Distributed Stochastic Neighbor Embedding (t-SNE) [36]
  It is a nonlinear technique that converts the distances between samples in high-dimensional space into similarity probabilities and then iteratively optimizes analogous t-distributions in the reduced space by minimizing the divergence between the two distributions at each iteration. Using a fat-tailed t-distribution in low-dimensional space allows us to emphasize the proximity of similar data, thus facilitating the visualization of the separation of different clusters.
- Uniform Manifold Approximation and Projection (UMAP) [37]
  It is a dimensionality reduction technique based on Riemannian geometry and algebraic topology. Unlike t-SNE, UMAP is faster to implement and better preserves the global structure of the data.

2.7.3 Quantitative Metrics

To quantitatively evaluate the quality of the clustering, three metrics were calculated that measure different aspects.

- Adjusted Rand Index (ARI)
  It measures the similarity between the predicted clusters and the ground truth classes, taking into account correctly grouped and correctly separated pairs. It is defined as:

$$ARI = \frac{\sum_{ij}\binom{n_{ij}}{2} - \frac{\sum_i\binom{a_j}{2}\sum_j\binom{b_j}{2}}{\binom{n}{2}}}{\frac{1}{2}\left[\sum_i\binom{a_j}{2} + \sum_j\binom{b_j}{2}\right] - \frac{\sum_i\binom{a_j}{2}\sum_j\binom{b_j}{2}}{\binom{n}{2}}}$$

Where:

n_{ij} is the number of samples that simultaneously belong to the ground truth and the predicted cluster j.

$a_j = \sum_j n_{ij}$ is the number of samples belonging to the ground truth i.

$b_j = \sum_i n_{ij}$ is the number of samples belonging to the predicted cluster j.

ARI is a symmetric measure and can take values between -1 and 1, where 1 indicates a perfect agreement between predicted clusters and ground truth while 0 indicates random clustering.

- Normalized Mutual Information (NMI)

  It is a normalized version of Mutual Information, a measure that quantifies the information shared between the predicted clusters and the ground truth classes. It is defined as:

$$NMI = \frac{2\, I(T;C)}{H(T) + H(C)}$$

Where:

$I(T;C) = \sum_{i,j} p(t_i, c_j) log \frac{p(t_i, c_j)}{p(t_i)p(c_j)}$ is the mutual information between predicted clusters C and ground truth classes Y.

$H(T) = -\sum_i p(t_i) \log p(t_i)$ is the Shannon entropy calculated for ground truth classes.

$H(C) = -\sum_i p(c_j) \log p(c_j)$ is the Shannon entropy calculated for predicted clusters.

NMI takes values between 0 and 1, where 1 indicates a perfect match between the predicted clusters and the ground truth classes while a value of 0 indicates that the predicted clusters and the ground truth classes are completely independent.

- Silhouette Score

  It is a measure that indicates the geometric quality of clusters, measuring how close each sample is to samples in the same cluster compared to those in other clusters. It is defined as:

$$S = \frac{1}{N} \sum_{i=1}^{N} \frac{b_i - a_i}{\max(b_{i,} a_i)}$$

Where:

a_i represents the intra-cluster cohesion of an i-th sample and indicates the average distance with all samples of the same cluster.

b_i represents the inter-cluster separation of an i-th sample and indicates the average distance with the samples of the nearest cluster.

N is the total number of samples.

It takes values between -1 and 1: a Silhouette score of 1 indicates perfectly separate clusters while 0 or negative values indicate overlapping clusters.

Unlike ARI and NMI, this measure does not require ground truth labels but is based solely on the geometric characteristics of the clusters, thus depending on how the distances are calculated.

3 Results

This chapter presents the experimental results obtained following training on the six previously analysed architectural configurations and the evaluation on test set.

The results are organized into three main sections: signal reconstruction, qualitative analysis of the extracted semantic feature visualizations, and subsequent quantitative analysis.

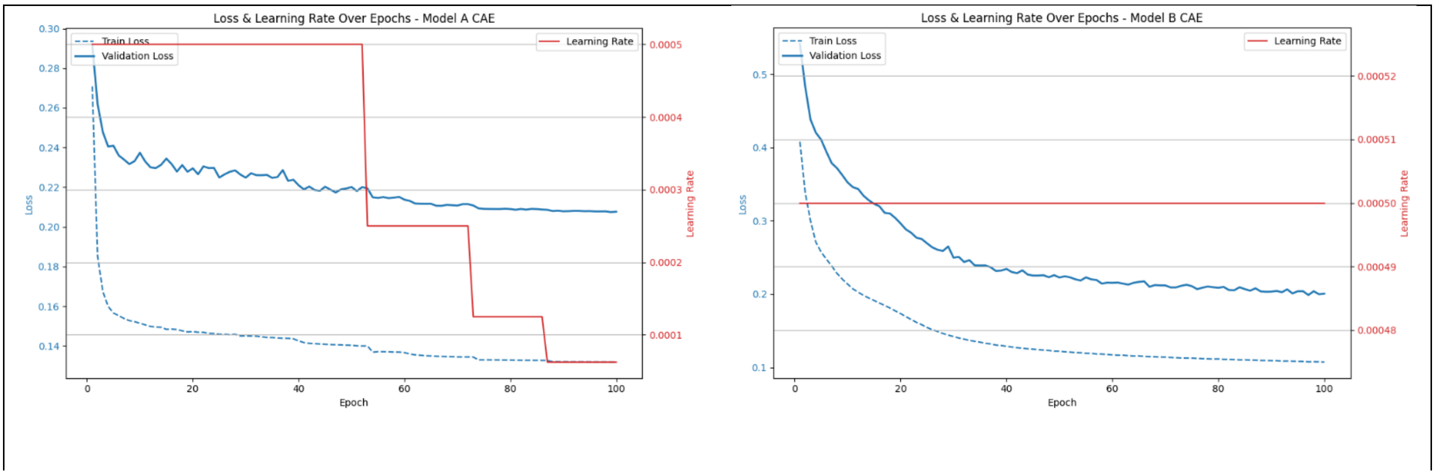
3.1 Signal reconstruction

3.1.1 Convergence during training

Figure 12 shows the MSE loss trend on the training set and validation set for the six configurations during the training epochs; differences in convergence speeds are observed between the configurations.

The variants of Model A converge relatively quickly; the CAE + Transformer variant even converges and stops training at epoch 65.

The three variants of Model B, however, converge significantly more slowly, making little or no use of learning rate rescheduling, probably due to the greater number of trainable parameters compared to the respective variants. In particular, the CAE + Transformer variant does not show convergence, suggesting that it requires longer training.



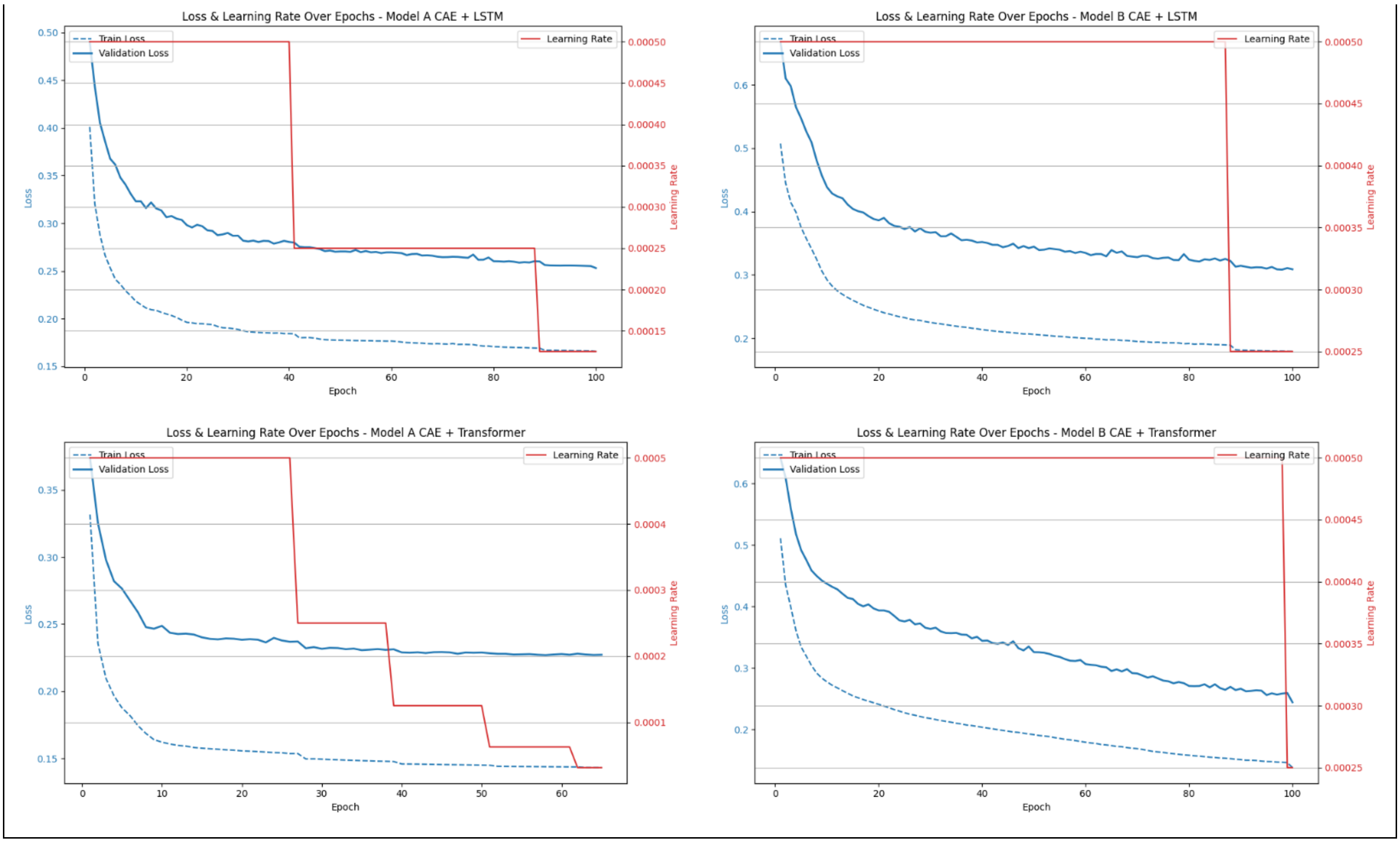


Figure 12: MSE Loss trends and learning rate dynamics during training phase for the six architectures. Model A variants show faster convergence than Model B variants.

3.1.2 Reconstruction performance

Table 1 shows the average losses obtained by the configurations on the validation set and the test set. All configurations maintain losses above 0.2, indicating that the reconstructions preserve the coarse features of the signal but do not perfectly reproduce the fine details of the original EEG activity. A qualitative representation of the reconstruction performances of the six architectures is shown in Figure 13.

The lowest losses on both the validation and test sets are achieved by the CAE basic architectures while the addition of LSTM and Transformer worsens the reconstruction performance. In particular, the LSTM variants showed the highest losses for both models, with test loss increases of 23.24% and 48.93% for CAE models A and B, respectively.

For the Transformer variants, the performance deterioration is less drastic, with test loss increases of 8.67% and 19.92% for CAE models A and B, respectively.

These results suggest that with a limited dataset, more complex models fail to effectively optimize parameters, thus leading to worse reconstruction.

Table 1: MSE Loss on validation set and test set for the six architectures

Architecture	Validation MSE Loss	Test MSE Loss
Model A CAE	0.2074	**0.2263**
Model A CAE+LSTM	0.2531	0.2789
Model A CAE+Transformer	0.2269	0.2459
Model B CAE	**0.1988**	0.2289
Model B CAE+LSTM	0.3083	0.3409
Model B CAE+Transformer	0.2442	0.2745

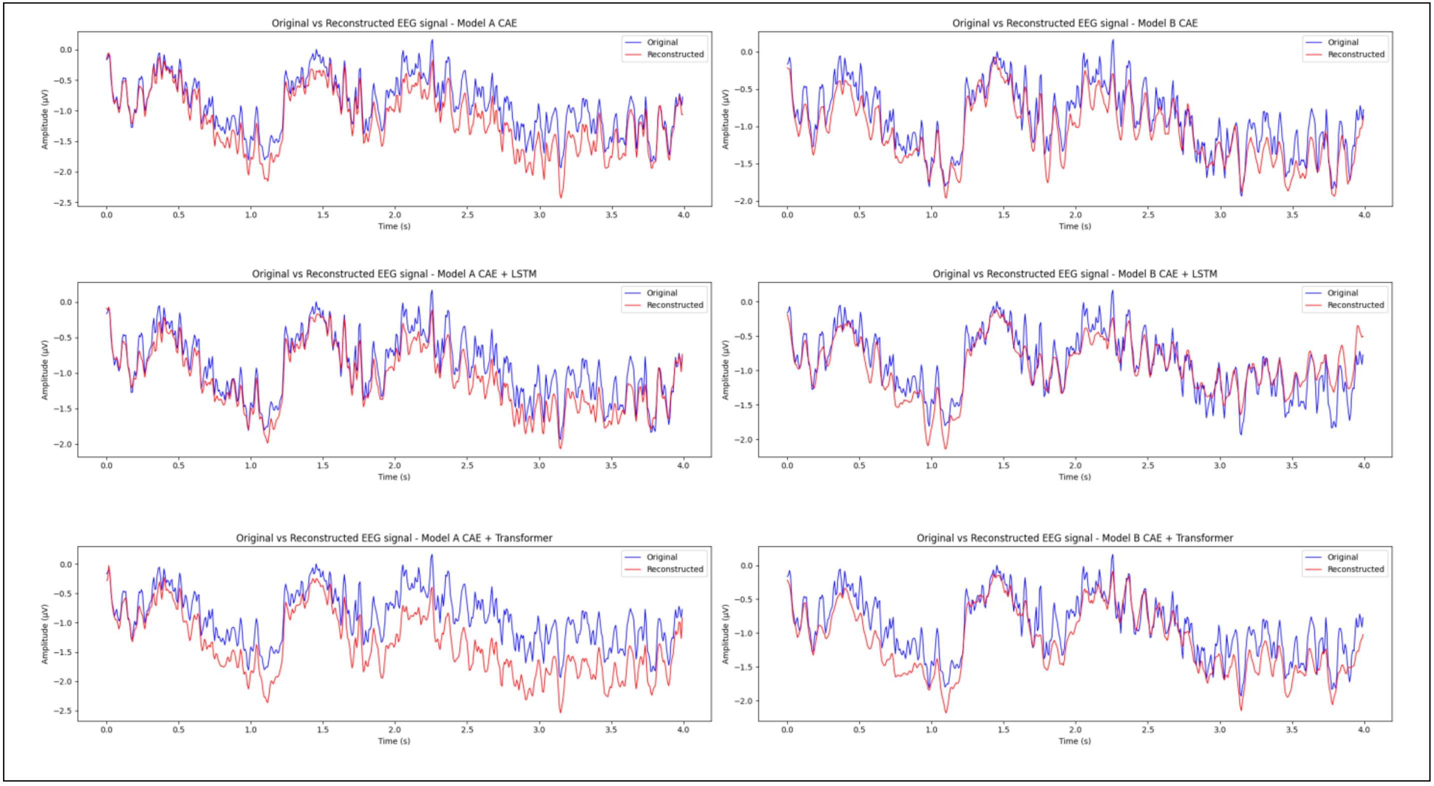


Figure 12: Comparison between original and reconstructed EEG signals for a single channel of a randomly chosen epoch for the six architectures. The imperfect alignment between the original and reconstructed signals highlights how the models were only able to extr

3.2 Analysis of the extracted features

3.2.1 Inter-subject clustering

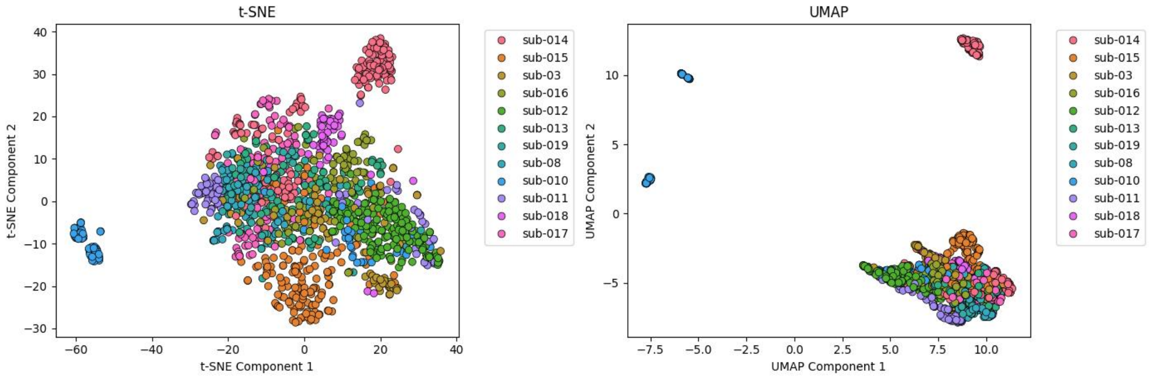
In this first phase, clustering is performed across subjects to see if the features extracted from the latent space contain subject-specific information.

Table 2 shows the values of the three quantitative metrics calculated for inter-subject clustering. For Model A, the CAE+Transformer variant produces better representations, while the CAE-only and CNN+LSTM variants produce slightly lower scores.. However, all three variants yield clusters that are poorly separated and mostly overlapping, as can be seen from the 2D visualizations via t-SNE and UMAP in Figure 14 and as summarized by the Silhouette scores. However, Model B variants achieve significantly better performance, with the CAE+LSTM proving to clearly encode subject-specific patterns: High ARI and NMI indicate that a good portion of the predicted clusters actually correspond to the subjects, while maintaining a high level of information about them. However, some clusters are overlapping and not totally distinct, even if 8 out of 12 subjects manage to be separated.

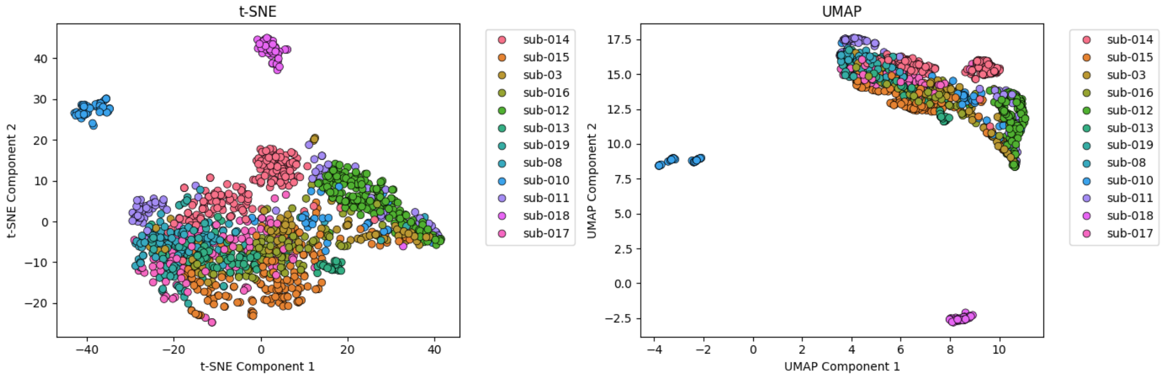
Table 2: Clustering metrics for inter-subject clustering

Architecture	ARI	NMI	Silhouette Score
Model A CAE	0.2079	0.3799	0.0801
Model A CAE+LSTM	0.1987	0.3994	0.1188
Model A CAE + Transformer	0.2441	0.4572	0.0980
Model B CAE	0.4381	0.5811	0.1125
Model B CAE + LSTM	**0.6086**	**0.7232**	**0.2515**
Model B CAE + Transformer	0.3803	0.6055	0.1196

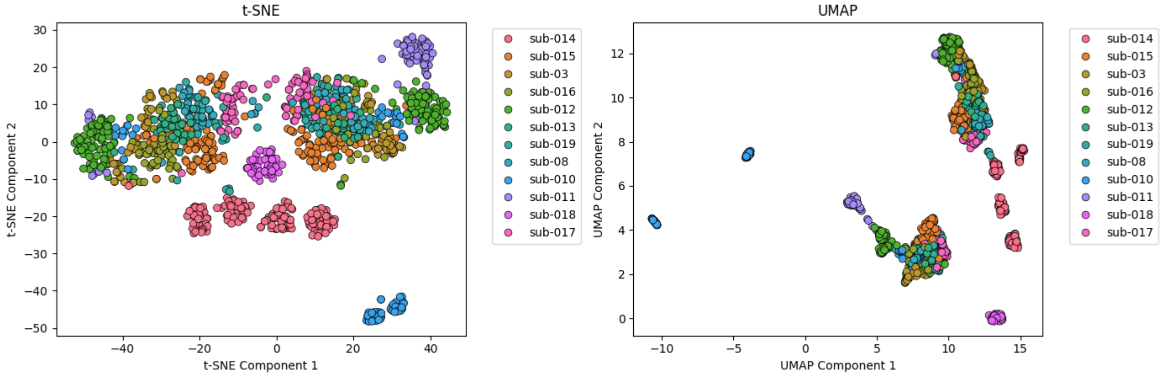
t-SNE and UMAP visualization of extracted features - Model A CAE



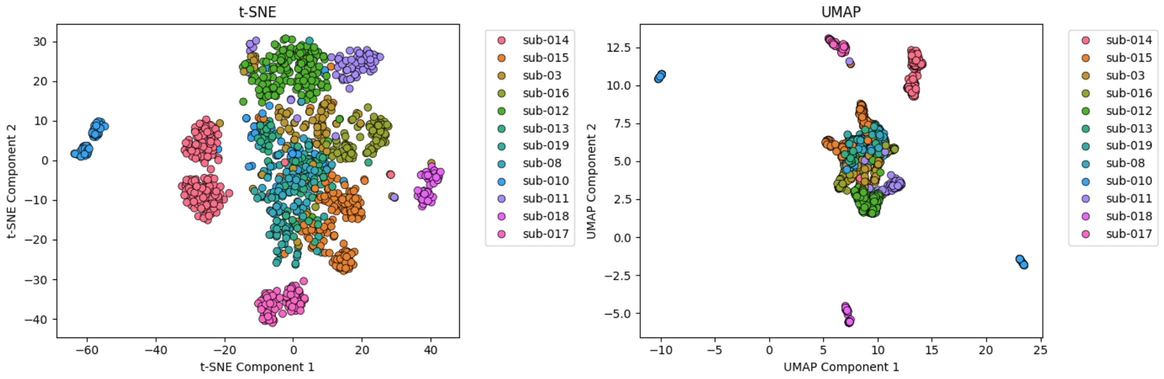
t-SNE and UMAP visualization of extracted features - Model A CAE + LSTM



t-SNE and UMAP visualization of extracted features - Model A CAE + Transformer



t-SNE and UMAP visualization of extracted features - Model B CAE



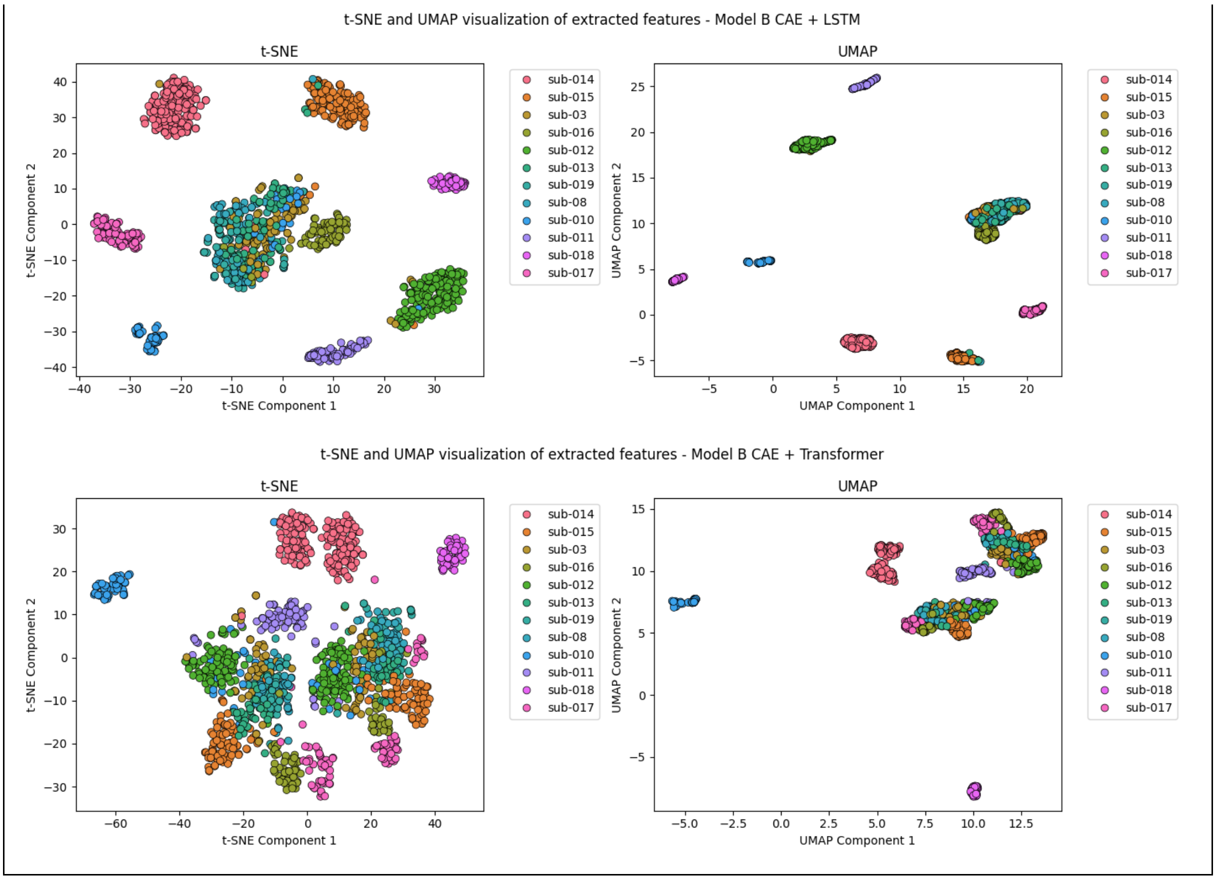


Figure 14: t-SNE and UMAP visualizations of extracted features coloured by subject for the six architectures. Model B variants feature more defined and separate clusters.

3.2.2 Intra-subject clustering

3.2.2.1 Semantic Concepts separation

Tables 3-8 show the results of intra-subject clustering for the separation of semantic concepts. This was the main goal of the study, but unfortunately it was not achieved: no configuration was able to separate these concepts, which remained hidden, yielding ARI and NMI close to zero, indicating that the latent clusters do not correspond to the three semantic concepts. Silhouette scores are moderately positive, suggesting some structure in the clusters, but unrelated to the ground-truth labels.

A visualization of t-SNE and UMAP showing the non-separation of semantic concepts is shown in Figure 15.

Table 3: Clustering metrics for concept separation, model A CAE

Model A CAE			
Subject	**ARI**	**NMI**	**Silhouette Score**
sub 3	0.0085	0.0171	0.1278
sub 8	-0.0111	0.0112	0.0956
sub 10	-0.0113	0.0208	0.7132
sub 11	0.0150	0.0312	0.1947
sub 12	0.0032	0.0123	0.1401
sub 13	0.0132	0.0258	0.0750
sub 14	0.0011	0.0183	0.1077
sub 15	0.0153	0.0321	0.0572
sub 16	0.0259	0.0465	0.0706
sub 17	0.0016	0.0287	0.0807
sub 18	-0.0332	0.0049	0.0525
sub 19	-0.0152	0.0147	0.0906
Average	0.0034 ± 0.01574	0.022 ± 0.011	0.1505 ± 0.1739

Table 4: Clustering metrics for concept separation, model A CAE + LSTM

Model A CAE + LSTM			
Subject	**ARI**	**NMI**	**Silhouette Score**
sub 3	-0.0067	0.0190	0.2443
sub 8	-0.0017	0.0206	0.0906
sub 10	-0.0133	0.0208	0.7742
sub 11	0.0188	0.0470	0.4830
sub 12	-0.0042	0.0082	0.3604
sub 13	0.0136	0.0414	0.1082
sub 14	-0.0011	0.0130	0.1151
sub 15	0.0004	0.0105	0.1028
sub 16	0.0164	0.0318	0.0713
sub 17	-0.0083	0.0133	0.0665
sub 18	-0.0223	0.0076	0.0852
sub 19	-0.0206	0.0406	0.0800
Average	0.0024 ± 0.0128	0.0228 ± 0.0133	0.2151 ± 0.2109

Table 5: Clustering metrics for concept separation, model A CAE + Transformer

Model A CAE + Transformer			
Subject	**ARI**	**NMI**	**Silhouette Score**
sub 3	-0.0039	0.0192	0.1751
sub 8	-0.0094	0.0134	0.0525
sub 10	-0.0145	0.0168	0.7185
sub 11	0.0015	0.0196	0.1307
sub 12	0.0046	0.0127	0.1596
sub 13	0.0342	0.0713	0.0979
sub 14	-0.0054	0.0049	0.0889
sub 15	0.0164	0.0296	0.0701
sub 16	0.0244	0.0473	0.1369
sub 17	0.0050	0.0179	0.1231
sub 18	0.0142	0.0324	0.1048
sub 19	-0.0194	0.0098	0.0658
Average	0.0040 ± 0.0153	0.0246 ± 0.0178	0.1603 ± 0.1721

Table 6: Clustering metrics for concept separation, model B CAE

Model B CAE			
Subject	**ARI**	**NMI**	**Silhouette Score**
sub 3	0.0026	0.0339	0.2628
sub 8	0.0054	0.0160	0.0545
sub 10	-0.0141	0.0167	0.6782
sub 11	-0.0128	0.0173	0.1238
sub 12	-0.0010	0.0063	0.1002
sub 13	0.0317	0.0432	0.0373
sub 14	0.0079	0.0165	0.0592
sub 15	0.0131	0.0222	0.1191
sub 16	-0.0065	0.0140	0.0664
sub 17	-0.0039	0.0149	0.0358
sub 18	0.0136	0.0335	0.1860
sub 19	0.0062	0.0362	0.0717
Average	0.0022 ± 0.0125	0.0226 ± 0.0108	0.1496 ± 0.1716

Table 7: Clustering metrics for concept separation, model B CAE + LSTM

Model B CAE + LSTM			
Subject	**ARI**	**NMI**	**Silhouette Score**
sub 3	-0.0010	0.0278	0.1133
sub 8	-0.0048	0.0130	0.2327
sub 10	-0.0123	0.0188	0.7419
sub 11	0.0048	0.0274	0.2282
sub 12	0.0014	0.0097	0.1576
sub 13	0.0006	0.0156	0.1827
sub 14	0.0017	0.0114	0.1071
sub 15	0.0066	0.0068	0.1417
sub 16	0.0306	0.0616	0.1185
sub 17	0.0061	0.0332	0.1916
sub 18	-0.0063	0.0696	0.2470
sub 19	-0.0220	0.0060	0.0832
Average	0.0009 ± 0.012	0.025 ± 0.0199	0.2121 ± 0.1678

Table 8: Clustering metrics for concept separation, model B CAE + Transformer

Model B CAE + Transformer			
Subject	**ARI**	**NMI**	**Silhouette Score**
sub 3	0.0037	0.0523	0.1443
sub 8	-0.0047	0.0186	0.0541
sub 10	-0.0122	0.0312	0.6912
sub 11	-0.0125	0.0129	0.1169
sub 12	0.0008	0.0094	0.0629
sub 13	0.0338	0.0458	0.1114
sub 14	-0.0036	0.0045	0.1227
sub 15	0.0001	0.0070	0.0991
sub 16	0.0057	0.0331	0.0648
sub 17	0.0105	0.0260	0.1303
sub 18	-0.0006	0.0335	0.1093
sub 19	0.0100	0.0334	0.0766
Average	0.0026 ± 0.0117	0.0256 ± 0.0147	0.1486 ± 0.1659

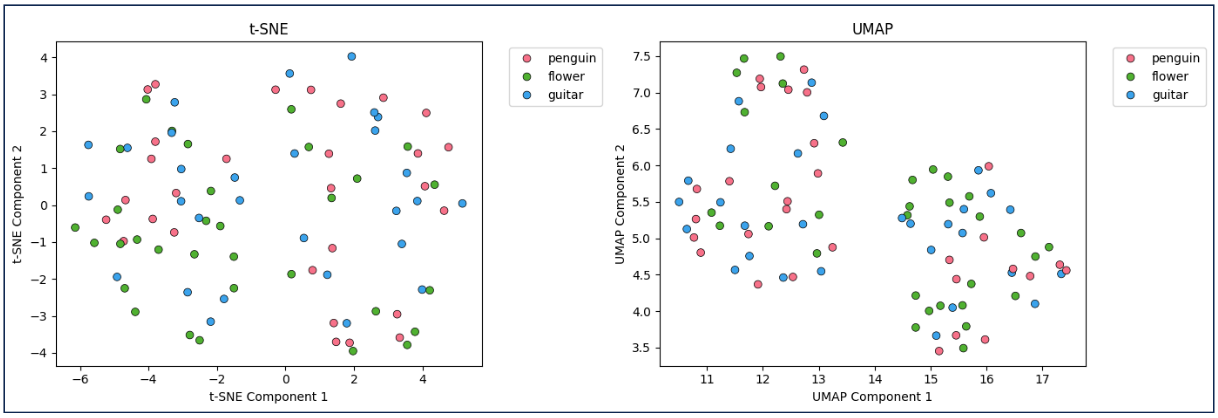


Figure 15: t-SNE and UMAP visualizations of concept separation on extracted features for subject 12 for Model A CAE + Transformer. The presence of separate clusters is observed but they do not correspond to the three semantic concepts.

3.2.2.2 Cognitive States Separation

Tables 9-14 show the results of the intra-subject clustering for the separation of imagination and perception states for the various architectures.

The architecture that best separates the two cognitive states is model A CAE+Transformer (observable in figure 16), achieving average ARI and NMI around 0.78. For some subjects, perfect clustering is obtained, while for subject 10, clustering is almost random, indicating strong subject-specific variability. This subject's extreme behavior is nevertheless present in all networks and visualizations, suggesting that it is an outlier. The CAE+Transformer variant is the only one that performs well for model A, with CAE and CAE+LSTM failing to predict meaningful clusters.

Among the variants of the B model, the basic CAE one achieves the highest performance.

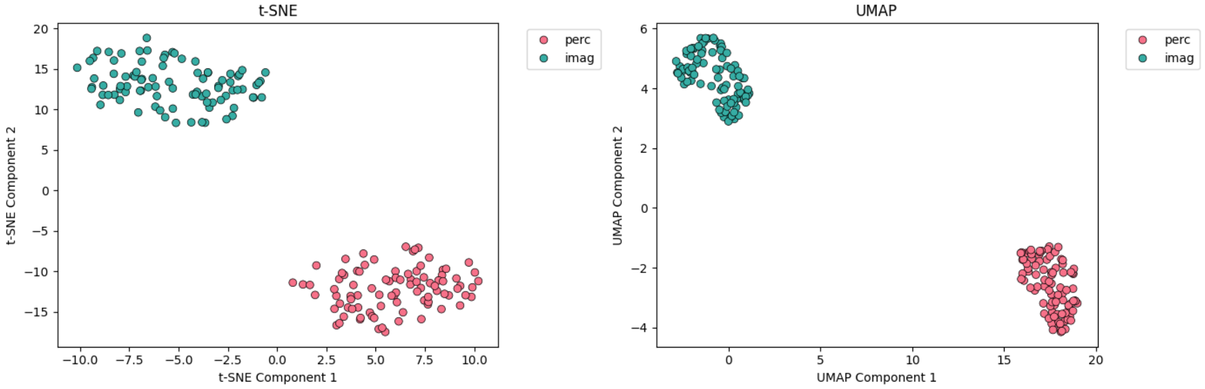


Figure 16: t-SNE and UMAP visualizations of imagination and perception separation on extracted features for subject 12 for Model A CAE + Transformer. We can see how the two states are perfectly separated into two distinct clusters.

Table 9: Clustering metrics for cognitive state separation, model A CAE

Model A CAE			
Subject	**ARI**	**NMI**	**Silhouette Score**
sub 3	0.0216	0.0226	0.2597
sub 8	0.0184	0.0127	0.1363
sub 10	-0.0099	0.0001	0.7857
sub 11	0.0125	0.0118	0.0486
sub 12	-0.0035	0.0020	0.2833
sub 13	0.2091	0.1732	0.0918
sub 14	-0.0036	0.0013	0.1503
sub 15	0.0005	0.0051	0.0688
sub 16	0.0734	0.0627	0.0782
sub 17	-0.0111	0.0000	0.081
sub 18	0.0176	0.0012	0.0725
sub 19	0.0272	0.0326	0.0848
Average	0.0264 ± 0.0598	0.027 ± 0.0475	0.2084 ± 0.2021

Table 10: Clustering metrics for cognitive state separation, model A CAE + LSTM

Model A CAE + LSTM			
Subject	**ARI**	**NMI**	**Silhouette Score**
sub 3	0.0083	0.0123	0.4329
sub 8	-0.0083	0.0000	0.1843
sub 10	-0.0099	0.0001	0.8056
sub 11	0.0046	0.0070	0.5520
sub 12	-0.0024	0.0015	0.4786
sub 13	0.1184	0.1395	0.3280
sub 14	0.0047	0.0073	0.1491
sub 15	-0.0044	0.0015	0.1267
sub 16	-0.0210	0.0183	0.2083
sub 17	-0.0077	0.0018	0.0771
sub 18	0.0990	0.1235	0.3562
sub 19	0.0006	0.0008	0.6477
Average	-0.0151 ± 0.0427	0.0261 ± 0.0475	0.3622 ± 0.2182

Table 11: Clustering metrics for cognitive state separation, model A CAE + Transformer

Model A CAE + Transformer			
Subject	**ARI**	**NMI**	**Silhouette Score**
sub 3	0.4386	0.4882	0.2080
sub 8	1.0000	1.0000	0.1171
sub 10	-0.0099	0.0001	0.7817
sub 11	0.0889	0.1805	0.3225
sub 12	1.0000	1.0000	0.2439
sub 13	0.9556	0.9229	0.1697
sub 14	1.0000	1.0000	0.1487
sub 15	1.0000	1.0000	0.1707
sub 16	1.0000	1.0000	0.1971
sub 17	0.9556	0.9229	0.1260
sub 18	1.0000	1.0000	0.1135
sub 19	0.9365	0.8971	0.1160
Average	0.7804 ± 0.3645	0.7838 ± 0.3417	0.2262 ± 0.1776

Table 12: Clustering metrics for cognitive state separation, model B CAE

Model B basic CAE			
Subject	**ARI**	**NMI**	**Silhouette Score**
sub 3	0.0261	0.1509	0.3084
sub 8	0.7877	0.7000	0.1040
sub 10	0.0099	0.0010	0.8098
sub 11	0.0726	0.0892	0.2792
sub 12	0.4124	0.4587	0.1450
sub 13	0.9121	0.8681	0.1315
sub 14	0.8690	0.8181	0.1617
sub 15	0.6070	0.5956	0.1545
sub 16	0.5002	0.5257	0.1330
sub 17	1.0000	1.0000	0.1400
sub 18	1.0000	1.0000	0.1903
sub 19	0.9365	0.8971	0.0920
Average	0.5928 ± 0.3722	0.5920 ± 0.3722	0.2208 ± 0.1882

Table 13: Clustering metrics for cognitive state separation, model B CAE + LSTM

Model B CAE + LSTM			
Subject	**ARI**	**NMI**	**Silhouette Score**
sub 3	0.0091	0.0270	0.3065
sub 8	0.7099	0.6034	0.2674
sub 10	-0.0099	0.0010	0.8199
sub 11	0.1055	0.0821	0.3735
sub 12	0.0024	0.0077	0.3592
sub 13	0.4089	0.3617	0.2593
sub 14	0.4068	0.3175	0.1772
sub 15	0.1816	0.1414	0.1672
sub 16	0.4087	0.3224	0.1518
sub 17	0.8282	0.7456	0.2017
sub 18	0.1248	0.1454	0.3319
sub 19	0.2297	0.1737	0.0951
Average	0.2838 ± 0.2639	0.244 ± 0.2273	0.2926 ± 0.18

Table 14: Clustering metrics for cognitive state separation, model B CAE + Transformer

Model B CAE + Transformer			
Subject	**ARI**	**NMI**	**Silhouette Score**
sub 3	0.0396	0.0368	0.1424
sub 8	1.0000	1.0000	0.1265
sub 10	-0.0099	0.0001	0.8152
sub 11	0.0722	0.0646	0.2288
sub 12	0.0215	0.0463	0.1667
sub 13	0.1909	0.3089	0.1717
sub 14	-0.0045	0.0007	0.1605
sub 15	1.0000	1.0000	0.1156
sub 16	0.9120	0.8423	0.1025
sub 17	1.0000	1.0000	0.1500
sub 18	1.0000	1.0000	0.1118
sub 19	0.8157	0.7682	0.1081
Average	0.5031 ± 0.4567	0.5057 ± 0.4411	0.2 ± 0.1886

4 Discussion

This section seeks to interpret and contextualize the results presented above, also suggesting possible future developments.

The CAE-based models achieved lower reconstruction errors than the variants integrating LSTM or Transformer. It is plausible that more complex models, characterized by a greater number of learnable parameters, require much larger datasets to avoid overfitting and learn stable latent representations. The presence of strong EEG noise and high inter-subject variability further amplifies this phenomenon: under limited data conditions, high-capacity models tend to focus on subject-specific structures, penalizing reconstruction.
The slower convergence of Model B compared to Model A confirms this: increasing convolutional depth does not lead to an automatic improvement, especially when the dataset is small. The reconstruction error, not being significantly small, does not guarantee that the latent space preserves high-level and semantically discriminative information.
Furthermore, the dataset used includes only 12 subjects, insufficient to model the large intra- and inter-subject variability typical of EEG signals related to semantic processes. The consequence is that features relevant to one subject may not generalize to others.

The primary goal of this study was to evaluate whether it was possible to extract features capable of decoding semantic concepts. However, the six Autoencoder configurations used in this work did not produce latent representations capable of separating the three concepts in a completely unsupervised manner, as evidenced by the ARI and NMI close to zero. The moderate Silhouette Score suggests that the clusters produced by K-Means still have a coherent internal structure, but this structure does not reflect the concepts: it is more likely to capture differences related to the subject, signal quality, noise levels, or individual physiological characteristics.

Unlike semantic concepts, the Transformer configuration of model A and the variants of model B showed a more or less effective separation between imagination and perception. This suggests that these two cognitive states exhibit more marked differences in terms of the amplitude, spatial distribution, and temporal dynamics of signals, making their distinction more accessible to models. However, the high variability in the results indicates that even this separation does not fully generalize across subjects. In this case as well, the poor generalizability can be attributed to the small size of the dataset and the presence of subjects who exhibited anomalous behaviour across all analysis.

The weaker behaviour of the LSTM variants might indicate that long-range temporal dependencies are not the main discriminating factor between perception and imagination; instead, local features in combination with global dependencies captured by convolutions and self-attention mechanisms are better suited to representing these states. On the contrary, it appears that the long-range temporal modelling of LSTMs is better suited to capturing highly subject-specific temporal dynamics and rhythmic patterns.

Future studies could address the limitations identified in this work through several complementary approaches: using larger datasets and architecture-specific hyperparameter optimizations to enable more robust training, and semi-supervised approaches or contrastive learning techniques that could provide the semantic guidance that reconstruction loss alone does not offer.

5 Conclusion

This work aimed to evaluate whether CNN-, LSTM-, and Transformer-based autoencoder models could learn, in an unsupervised way, latent representations capable of capturing semantic concepts evoked through perception and imagination from EEG signals.

Under the circumstances faced in this study – dataset with only 12 subjects, subjects presenting EEG signals with low signal-to-noise ratio, sub-optimal training convergences – we were unable to obtain latent representations that would allow the separation of the three semantic concepts in a completely unsupervised manner.

The analysis of different architectures shows that increasing the complexity of the model, as in the variants with LSTM and Transformer, does not automatically lead to an improvement in the quality of the representations. Larger models require much larger amounts of data and introduce more unstable optimization dynamics, as evidenced by convergence difficulties. Conversely, more compact architectures appear to be more robust in reconstruction, even though they fail to capture semantic information.

A secondary, but still promising, result concerns the separation between perception and imagination states. Although this distinction is at a lower level than the semantic one, it suggests that the models can extract patterns related to cognitive state, likely thanks to the combination of local (CNN) and global (Transformer) features. However, these patterns fail to discriminate concepts, nor do they generalize well across subjects, as shown by the high variance in performance.

Bibliography

[1] K. H. Jawabri e S. Sharma, «Physiology, Cerebral Cortex Functions», in *StatPearls*, Treasure Island (FL): StatPearls Publishing, 2025. Consultato: 29 novembre 2025. [Online]. Disponibile su: http://www.ncbi.nlm.nih.gov/books/NBK538496/

[2] «Electroencephalography», in *Handbook of Clinical Neurology*, vol. 168, Elsevier, 2020, pp. 249–262. doi: 10.1016/B978-0-444-63934-9.00018-4.

[3] G. Buzsáki, C. A. Anastassiou, e C. Koch, «The origin of extracellular fields and currents — EEG, ECoG, LFP and spikes», *Nat Rev Neurosci*, vol. 13, fasc. 6, pp. 407–420, giu. 2012, doi: 10.1038/nrn3241.

[4] R. Oostenveld e P. Praamstra, «The five percent electrode system for high-resolution EEG and ERP measurements», *Clinical Neurophysiology*, vol. 112, fasc. 4, pp. 713–719, apr. 2001, doi: 10.1016/S1388-2457(00)00527-7.

[5] M. Rashid *et al.*, «Current Status, Challenges, and Possible Solutions of EEG-Based Brain-Computer Interface: A Comprehensive Review», *Front. Neurorobot.*, vol. 14, giu. 2020, doi: 10.3389/fnbot.2020.00025.

[6] W. P. Fifer, P. G. Grieve, J. Grose-Fifer, J. R. Isler, e D. Byrd, «High-Density Electroencephalogram Monitoring in the Neonate», *Clinics in Perinatology*, vol. 33, fasc. 3, pp. 679–691, set. 2006, doi: 10.1016/j.clp.2006.06.011.

[7] A. K. Singh e S. Krishnan, «Trends in EEG signal feature extraction applications», *Front. Artif. Intell.*, vol. 5, gen. 2023, doi: 10.3389/frai.2022.1072801.

[8] M. Rybář e I. Daly, «Neural decoding of semantic concepts: a systematic literature review», *J. Neural Eng.*, vol. 19, fasc. 2, p. 021002, apr. 2022, doi: 10.1088/1741-2552/ac619a.

[9] E. K. Warrington e T. Shallice, «CATEGORY SPECIFIC SEMANTIC IMPAIRMENTS», *Brain*, vol. 107, fasc. 3, pp. 829–853, 1984, doi: 10.1093/brain/107.3.829.

[10] R. Bruffaerts, S. De Deyne, K. Meersmans, A. G. Liuzzi, G. Storms, e R. Vandenberghe, «Redefining the resolution of semantic knowledge in the brain: Advances made by the

introduction of models of semantics in neuroimaging», *Neuroscience & Biobehavioral Reviews*, vol. 103, pp. 3–13, ago. 2019, doi: 10.1016/j.neubiorev.2019.05.015.

[11] I. Simanova, M. van Gerven, R. Oostenveld, e P. Hagoort, «Identifying Object Categories from Event-Related EEG: Toward Decoding of Conceptual Representations», *PLOS ONE*, vol. 5, fasc. 12, p. e14465, dic 2010, doi: 10.1371/journal.pone.0014465.

[12] J. Ouyang, M. Wu, X. Li, H. Deng, Z. Jin, e D. Wu, «NeuroBCI: Multi-Brain to Multi-Robot Interaction Through EEG-Adaptive Neural Networks and Semantic Communications», *IEEE Transactions on Mobile Computing*, vol. 23, fasc. 12, pp. 14622–14637, dic. 2024, doi: 10.1109/TMC.2024.3446829.

[13] H. Ahmadi e L. Mesin, «Universal semantic feature extraction from EEG signals: a task-independent framework», *J. Neural Eng.*, vol. 22, fasc. 3, p. 036003, giu. 2025, doi: 10.1088/1741-2552/add08f.

[14] I. Simanova, P. Hagoort, R. Oostenveld, e M. A. J. Van Gerven, «Modality-Independent Decoding of Semantic Information from the Human Brain», *Cerebral Cortex*, vol. 24, fasc. 2, pp. 426–434, feb. 2014, doi: 10.1093/cercor/bhs324.

[15] H. Ahmadi e L. Mesin, «Enhancing MI EEG Signal Classification With a Novel Weighted and Stacked Adaptive Integrated Ensemble Model: A Multi-Dataset Approach», *IEEE Access*, vol. 12, pp. 103626–103646, 2024, doi: 10.1109/ACCESS.2024.3434654.

[16] H. Ahmadi e L. Mesin, «Enhancing Motor Imagery Electroencephalography Classification with a Correlation-Optimized Weighted Stacking Ensemble Model», *Electronics*, vol. 13, fasc. 6, p. 1033, mar. 2024, doi: 10.3390/electronics13061033.

[17] H. Ahmadi e L. Mesin, «Decoding Visual Imagination and Perception from EEG via Topomap Sequences», 8 maggio 2025. doi: 10.36227/techrxiv.174672922.22051031/v1.

[18] S. Xie, D. Kaiser, e R. M. Cichy, «Visual Imagery and Perception Share Neural Representations in the Alpha Frequency Band», *Current Biology*, vol. 30, fasc. 13, pp. 2621-2627.e5, lug. 2020, doi: 10.1016/j.cub.2020.04.074.

[19] S.-H. Lee, D. J. Kravitz, e C. I. Baker, «Disentangling visual imagery and perception of real-world objects», *NeuroImage*, vol. 59, fasc. 4, pp. 4064–4073, feb. 2012, doi: 10.1016/j.neuroimage.2011.10.055.

[20] R. T. Schirrmeister *et al.*, «Deep learning with convolutional neural networks for EEG decoding and visualization», 8 giugno 2018, *arXiv*: arXiv:1703.05051. doi: 10.48550/arXiv.1703.05051.

[21] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, e B. J. Lance, «EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces», *J. Neural Eng.*, vol. 15, fasc. 5, p. 056013, ott. 2018, doi: 10.1088/1741-2552/aace8c.

[22] J. Fan, C. Ma, e Y. Zhong, «A Selective Overview of Deep Learning», 15 aprile 2019, *arXiv*: arXiv:1904.05526. doi: 10.48550/arXiv.1904.05526.

[23] A. Vaswani *et al.*, «Attention Is All You Need», 2 agosto 2023, *arXiv*: arXiv:1706.03762. doi: 10.48550/arXiv.1706.03762.

[24] A. Gramfort *et al.*, «MEG and EEG data analysis with MNE-Python», *Front. Neurosci.*, vol. 7, dic. 2013, doi: 10.3389/fnins.2013.00267.

[25] F. Pedregosa *et al.*, «Scikit-learn: Machine Learning in Python», *Journal of Machine Learning Research*, vol. 12, fasc. 85, pp. 2825–2830, 2011, Consultato: 29 novembre 2025. [Online]. Disponibile su: http://jmlr.org/papers/v12/pedregosa11a.html

[26] A. Paszke *et al.*, «Automatic differentiation in PyTorch», ott. 2017, Consultato: 29 novembre 2025. [Online]. Disponibile su: https://openreview.net/forum?id=BJJsrmfCZ

[27] «HPC@POLITO | Home». Consultato: 29 novembre 2025. [Online]. Disponibile su: https://hpc.polito.it/index.shtml

[28] H. Wilson, M. Golbabaee, M. J. Proulx, S. Charles, e E. O’Neill, «EEG-based BCI Dataset of Semantic Concepts for Imagination and Perception Tasks», *Sci Data*, vol. 10, fasc. 1, p. 386, giu. 2023, doi: 10.1038/s41597-023-02287-9.

[29] D. F. Marks, «VISUAL IMAGERY DIFFERENCES IN THE RECALL OF PICTURES», *British J of Psychology*, vol. 64, fasc. 1, pp. 17–24, feb. 1973, doi: 10.1111/j.2044-8295.1973.tb01322.x.

[30] A. R. Halpern, «Differences in auditory imagery self-report predict neural and behavioral outcomes.», *Psychomusicology: Music, Mind, and Brain*, vol. 25, fasc. 1, pp. 37–47, mar. 2015, doi: 10.1037/pmu0000081.

[31] S. Appelhoff *et al.*, *PyPREP: A Python implementation of the preprocessing pipeline (PREP) for EEG data.* (17 luglio 2025). Zenodo. doi: 10.5281/zenodo.16039994.

[32] N. Bigdely-Shamlo, T. Mullen, C. Kothe, K.-M. Su, e K. A. Robbins, «The PREP pipeline: standardized preprocessing for large-scale EEG analysis», *Front. Neuroinform.*, vol. 9, giu. 2015, doi: 10.3389/fninf.2015.00016.

[33] O. George, R. Smith, P. Madiraju, N. Yahyasoltani, e S. I. Ahamed, «Data augmentation strategies for EEG-based motor imagery decoding», *Heliyon*, vol. 8, fasc. 8, p. e10240, ago. 2022, doi: 10.1016/j.heliyon.2022.e10240.

[34] J. L. Ba, J. R. Kiros, e G. E. Hinton, «Layer Normalization», 21 luglio 2016, *arXiv*: arXiv:1607.06450. doi: 10.48550/arXiv.1607.06450.

[35] D. P. Kingma e J. Ba, «Adam: A Method for Stochastic Optimization», 30 gennaio 2017, *arXiv*: arXiv:1412.6980. doi: 10.48550/arXiv.1412.6980.

[36] L. van der Maaten e G. Hinton, «Visualizing Data using t-SNE», *Journal of Machine Learning Research*, vol. 9, fasc. 86, pp. 2579–2605, 2008, Consultato: 29 novembre 2025. [Online]. Disponibile su: http://jmlr.org/papers/v9/vandermaaten08a.html

[37] L. McInnes, J. Healy, e J. Melville, «UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction», 18 settembre 2020, *arXiv*: arXiv:1802.03426. doi: 10.48550/arXiv.1802.03426.