ETHZ DeepLearning class AS24 - Project Report: Benchmarking Neural Latent Representations on EEG data for Sleep Stage Classification

Dingxi Zhang [*1] **Raphael Kreft** [*1] **Shaswat Gupta** [*1] **Zefan Holliger** [*1]

Abstract

Sleep Stage Classification (SSC) is critical in understanding sleep physiology and diagnosing disorders. This study addresses a key gap in understanding the role of Self-Supervised-Learning (SSL) in EEG-based SSC by benchmarking three SSL paradigms in combination with three representative encoder architectures. We focus on the learned representation quality, offering insights into optimizing SSL frameworks for robust and generalizable SSC systems. Our Latent-space benchmarks and linear evaluation suggest clear evidence of how to best combine the SSL paradigm and backbone architectures. Source code is available at: https://github.com/RaphaelKreft/DeepLearningProject

1. Introduction

The sleep of humans can be categorized into different sleep stages (Patel et al., 2024). Classifying recorded *Electroencephalography (EEG)* signals into these stages is critical for diagnosing sleep disorders and understanding sleep physiology (Šušmáková, 2004). While this task is traditionally carried out by human experts, automatic Sleep Stage Classification (SSC) using traditional Machine Learning(ML) (Li et al., 2018) and deep learning (DL) models aim to save resources and time and enable real-time monitoring.

While DL models show the most accurate results, traditional supervised approaches have reached a performance plateau. One of the main reasons is that most EEG data remains unlabelled due to sleep experts' costly and labor-intensive annotation process. Inspired by its success in the domain of NLP and Computer Vision (Jing & Tian, 2021), the community most recently started using *Self-supervised learning (SSL)*, as it enables leveraging unlabeled data to learn robust feature representations.

While SSL for SSC shows great potential to improve performance and robustness against inter-patient and recording variability (Lee et al., 2024a;b; Mohammadi Foumani et al., 2024; Chien et al., 2022), there is yet no thorough research on how different parameters in the design space of SSL pipelines affect the quality of the latent space, that SSL pre-trained encoders generate.

In this work, we close this gap by examining the effect of different encoder architectures and pre-training frameworks on the latent space structure and quality, as well as the subsequent effect on the downstream task of classification.

2. Related Works

It has already been demonstrated that SSL techniques are effectively applied in SSC (Jiang et al., 2021b). The most prominent SSL paradigms include *Masked Prediction (MP)* and *Contrastive Learning (CRL)*.

SleePyCo (Lee et al., 2024b) achieves state-of-the-art performance in CRL using a simple CNN feature pyramid as the feature-extraction backbone and appends a transformer-based classifier. MAEEG (Chien et al., 2022) employs a masked convolutional autoencoder to perform MP on raw EEG signals, whereas EEG2REP (Mohammadi Foumani et al., 2024) utilizes a transformer-based autoencoder to apply MP in the latent space. NeuroNet (Lee et al., 2024a) adopts a hybrid approach, combining CRL and MP. NeuroNet first applies a 1D multiscale ResNet to generate latent tokens of overlapping frames in the signal. Subsequently, a Transformer Autoencoder is used to apply MP and CRL to these tokens.

3. Methods

In this section, we describe our pipeline and the evaluation parameter space. We first pre-train three different backbone architectures on the PhysioNet SleepEDF-2018 dataset (Goldberger et al., 2000) using three different SSL paradigms. We then freeze the backbone and train a simple classifier in a supervised setting to perform SSC based on the backbone's latent space, allowing for Linear Evaluation. From this, we compute latent space and linear evaluation metrics on the test set of the dataset.

3.1. Backbone Architectures

We selected the three backbone architectures to capture a range of commonly used designs in SSC, ensuring they are representative of the domain while allowing us to as-

[*]Equal contribution [1]Department of D-INFK, ETH Zürich, Zurich, Switzerland.

sess the impact of architectural variations on benchmark performance. All models are configured to output a 128-dimensional latent to ensure comparability.

Convolutional Neural Network First, we chose a basic CNN backbone architecture. We adapted SleePyCo's backbone (Lee et al., 2024b) since it is very basic while showing state-of-the-art performance.

The architecture comprises five convolutional blocks, each being a sequence of a 1-D convolutional layer, a 1-D batch normalization layer, and a parametric rectified linear unit (PReLU) (He et al., 2015). All convolutions have kernel size 3, stride 1, and a padding of 1, hence keeping the spatial dimension unchanged. To decrease it, max-pooling is applied between convolutional blocks. Additionally, each convolutional block includes a squeeze and excitation module (Hu et al., 2019) before the activation function.

To be able to perform Masked Prediction (MP), we implemented a decoder backbone, which is simply the inversed encoder, with the difference that we replace max-pooling with transposed convolutional layers so we increase spatial dimension between the convolutional blocks.

CNN with Multi-Head Attention Since "Attention is all you need" introduced transformers (Vaswani et al., 2023), attention is used widely in all kinds of models. To study the effect of using attention, we take the basic CNN backbone described previously and add Multi-headed Attention with eight heads to the last two convolutional blocks before the activation function. Attention is computationally heavy, so we use it where it's most effective, which is in later layers where simple signal statistics have already been learned.

Transformer Recognizing the recent success of transformers across a wide range of deep-learning applications, we include a Transformer backbone in our evaluation. Specifically, we adopt the Transformer component of NeuroNet (Lee et al., 2024a), which demonstrates state-of-the-art performance. We explicitly omit the additional CNN backbone NeuroNet uses to keep the models comparable. Starting from the original signal dimension, we project it down to the latent space dimension. The transformer then works on this dimension to output latent space. For MP, we additionally implement a Transformer decoder to complete the framework.

3.2. Pretraining Paradigms

Contrastive Learning The core principle of Contrastive Learning is to ensure that augmented samples derived from the same base signal exhibit high similarity in the latent space while simultaneously maximizing the distance between negative pairs.

Based on a signal sample, we apply a two-transform, which returns two randomly augmented versions. We sequentially apply six augmentations: *RandomBandStopFilter*, *RandomTimeShift*, *RandomZeroMasking* from SleePyCo (Lee et al., 2024b), and *TimeWarping Permutation*, and *CutoutResize* from (Jiang et al., 2021a) with a probability of 0.5. The reasoning behind choosing these specific augmentations and their probabilities of application are described in detail in the appendix C.

Using shared weights, we process both augmented versions through the backbone and apply the NTXent Loss (Sohn, 2016), which maximizes the similarity between positive pairs while minimizing it on negative ones.

Masked Prediction In Masked Prediction, a part of the input is masked (by setting values to 0). For MP, the model, typically comprising an encoder and decoder, tries to reconstruct the masked parts of the signal. While MP can be applied in the latent space as well as on the original signal, we follow MAEEG (Chien et al., 2022) in doing masking on the signal directly as it allows for better comparability to other paradigms without having to modify the model. We apply fixed-proportion random masking (which sorts noise and takes the lowest 25% and masks the respective parts of the signal), which makes the experiments reproducible. We apply standard L2 loss on the masked areas. After the training, the decoder is discarded, and we keep the trained encoder.

Hybrid We combine MP with CL to evaluate joint performance and research mutual reinforcement possibilities in a hybrid paradigm. For training, we pass both a masked signal and two transformed augmented signals through the model. We calculate contrastive loss (NTXent) on the latent space outputs and reconstruction loss (L2) on the output of the decoder. In early experiments, we mentioned that the average contrastive Loss is smaller than the Reconstruction Loss by a factor of $30 - 40$, which we need to cope with as we want to have a comparable setting where both training paradigms have equal contribution to learning. Hence the combined loss is a weighted sum: $L_{total} = L_{rec} + 35L_{cl}$

3.3. Dataset

We utilize the PhysioNet Sleep-EDF Expanded dataset (Goldberger et al., 2000) in the 2018 version. Aligning with prior research (Phan et al., 2022), we focus on Sleep Cassette (SC) recordings from 79 healthy individuals aged 25–101 years and filter non-sleep epochs and periods marked as MOVING or UNKNOWN. We further preprocess the dataset for training and classification on single-channel EEG data: Selecting the Fpz-Cz EEG channel, segmenting the signal into 30-second epochs, and downsampling signals to 100 Hz. Our classification of sleep

signals in sleep stages, as well as the dataset annotation, follows the R&K rules (Rechtschaffen, 1968), which classifies sleep into five stages. Analysis of the processed dataset revealed 198,289 epochs, with a class distribution of 35% WAKE, 11% N1, 35% N2, 7% N3, and 13% REM, highlighting a significant imbalance.

We employed an 80-10-10 train-validation-test split by partitioning the data at the patient level, preventing data leakage.

3.4. Benchmarking

Latent Space Benchmarks We employ a comprehensive set of quantitative metrics to assess the quality of various architectures and paradigms. These include the Silhouette Score (Rousseeuw, 1987), Davies-Bouldin Index (Davies & Bouldin, 1979), and Adjusted Rand Index (Hubert & Arabie, 1985), among others, providing a robust evaluation framework for latent space representation. Please refer to the Appendix for more details.

Latent Space Visualizations We investigate latent space properties using visualization techniques such as t-SNE (Van der Maaten & Hinton, 2008), UMAP (Sainburg et al., 2021), and PCA (F.R.S., 1901). These visualizations enable an intuitive understanding of the structure and separability within the latent representations.

Linear Evaluation To assess the quality of the learned representations and their applicability to downstream classification tasks, we furthermore perform a linear evaluation by freezing the weights of the trained encoders and training a 3-layer MLP with ReLU activations and dropout regularization. Performance is comprehensively evaluated using metrics such as accuracy, macro F1-score, and Cohen's Kappa (Sokolova & Lapalme, 2009), ensuring a robust analysis of the representation's effectiveness in enabling accurate and balanced classification.

4. Experiments

For training the backbones, we train for a maximum of 500 epochs with early stopping activated with the patience of 10 validations while evaluating the validation set after every epoch. We use a batch size of 128, set a learning rate of 0.001, and apply a weight decay of 0.0001 for all configurations. We freeze the backbone and use the same parameters for training the classifiers except for a maximum of 100 epochs and a dropout of $p = 0.5$. We split the dataset into a train, val, and test set where the split is along the 79 subjects to allow for proper validation and test sets. Data from 65, 7, and 7 distinct subjects were used in the training, validation, and test set.

For more details about this two-stage training process or training parameters, refer to D and E

	CRL	MP	Hybrid
CNN	76.9/66.1/0.67	63.9/50.8/0.486	78.8/68.7/0.700
CNN+Attn	79.8/69.2/0.715	69.0/53.9/0.552	78.9/67.7/0.702
Transformer	49.5/29.4/0.265	62.5/48.4/0.462	56.4/41.6/0.374

Table 1. Metrics Accuracy(↑) / Macro-F1(↑) / Cohens-Kappa(↑) for all combinations of backbone models and SSL paradigms. The best value for each metric is highlighted in green, the second best in blue, and the third best in red.

4.1. Experiment results

Table 1 depicts the linear evaluation *quantitative* results of different combinations of SSL paradigms(columns) and backbone architectures(rows) where we report classification accuracy, Macro-F1, and Cohen's Kappa. Out of the three backbones, CNN+Attn consistently outperforms the vanilla CNN and Transformer, demonstrating a superior capacity for capturing discriminative EEG features. While the performance difference between CNN and Transformer is quite small, it explodes in comparison to Transformer. This suggests that the attention mechanism not only pinpoints critical regions in the data but also fosters improved alignment of latent representations. In contrast, the Transformer backbone underperforms across all paradigms, likely due to its reliance on large-scale training data and its challenges in modeling short EEG segments. Focussing on differences on the paradigm level, CRL achieves the highest performance with both CNN and CNN+Attn backbones, indicating that contrastive objectives are well-suited for learning meaningful latent representations. The Hybrid paradigm offers a balanced alternative, particularly when paired with CNN. It integrates complementary SSL objectives, improving feature compactness and separability. Meanwhile, Masked Prediction (MP) shows limited effectiveness, possibly due to insufficient learning of fine-grained temporal dependencies. These horizontal and vertical comparisons highlight the importance of tailoring both the encoder and SSL paradigms to optimize latent space quality, which directly influences downstream SSC performance.

These linear evaluation results are supported by table 2 that depicts t-sne visualizations of latent spaces generated on the test set. We clearly see that CRL and Hybrid from clear boundaries and clusters as compared to the MP latent spaces. Comparing the models, we also observe that there is more overlap of the clusters for Transformer, suggesting a less clearly separated latent space.

Table 3 depicts specific latent space benchmarks. It reveals that CNN+Attention combined with CRL consistently achieves the best overall performance, characterized by high Silhouette Scores, elevated Adjusted Mutual Information (AMI), and favorable Compactness-to-Separation Ratios

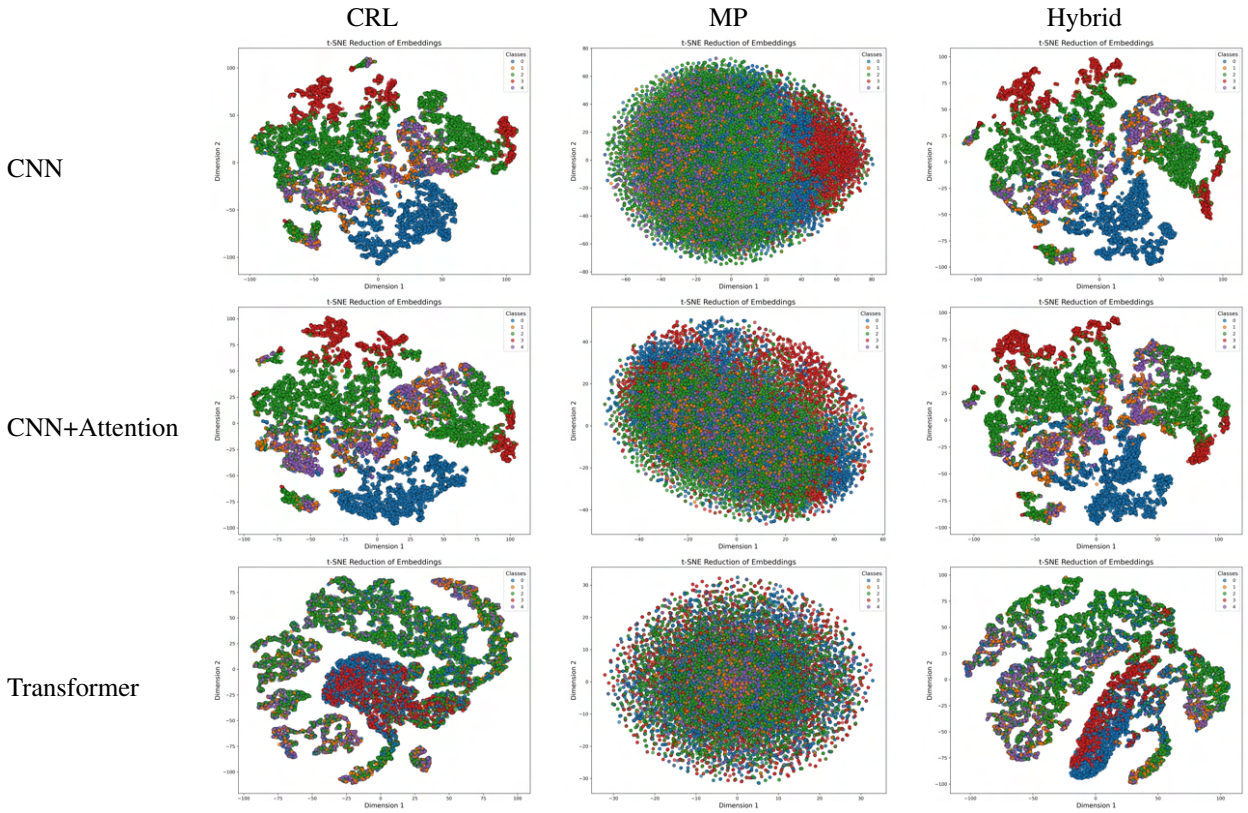

Table 2. Latent Space Visualization using t-SNE method

Model/Paradigm	CRL	MP	Hybrid
CNN	0.189/0.274/1.509/3.898/1.037	0.033/0.162/1.747/7.438/0.987	0.157/0.320/1.413/4.167/1.015
CNN+Attention	0.186/0.336/1.409/4.450/1.024	0.018/0.152/1.773/10.973/1.038	0.158/0.341/1.381/4.376/1.024
Transformer	0.36/0.235/1.608/3.893/1.042	0.003/0.22/1.625/14.932/1.015	0.305/0.224/1.608/3.041/1.011

Table 3. Most important latent space benchmark Metrics for all backbone models and SSL paradigms combinations. Each cell contains: Silhouette Score(↑) / Adjusted Mutual Information(↑) / Average Entropy(↓) / Compactness-to-Separation Ratio(↓) / Alignment(↑). The best value for each metric is highlighted in green, the second best in blue, and the third best in red.

(CSR). These results point to a well-structured latent space with clear class separability and robust alignment. The Hybrid paradigm also performs strongly across most metrics, particularly with a CNN backbone, suggesting an effective balance between feature compactness and class separability. In contrast, the Transformer backbone struggles with CSR and Alignment—especially under MP—indicating that it does not always leverage its latent structure efficiently for class-level discrimination in EEG tasks.

5. Conclusion

In summary, we trained three representative backbone encoders mixed with three fundamental SSL frameworks to understand their interactive effects by benchmarking and analyzing their latent space embeddings using linear evaluation and specific latent space benchmarks. The contrastive learning (CRL) and Hybrid paradigms generally yield superior, more distinguishable latent representations for EEG-based SSC than masked prediction, translating into stronger downstream performance, particularly with CNN and CNN+Attention backbones. Although Transformers occasionally exhibit promising cluster separations, as suggested by certain silhouette-based metrics, these gains do not consistently manifest in other latent space and classification metrics, indicating that short EEG segments pose unique challenges for Transformers' data-hungry attention mechanisms. Consequently, CNN+Attention emerges as a robust encoder choice—showcasing its ability to pinpoint and amplify critical EEG features—while CRL and Hybrid objectives outperform MP by delivering more compact yet discriminable latent spaces. Future work could focus on using other datsets or multi-channel EEG classifications as well as investigating Transformers more.

References

Chien, H.-Y. S., Goh, H., Sandino, C. M., and Cheng, J. Y. Maeeg: Masked auto-encoder for eeg representation learning. *arXiv preprint arXiv:2211.02625*, 2022.

Davies, D. L. and Bouldin, D. W. A cluster separation measure. *IEEE transactions on pattern analysis and machine intelligence*, (2):224–227, 1979.

F.R.S., K. P. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11): 559–572, 1901. doi: 10.1080/14786440109462720.

Goldberger, A. L., Amaral, L. A. N., Glass, L., Hausdorff, J. M., Ivanov, P. C., Mark, R. G., Mietus, J. E., Moody, G. B., Peng, C.-K., and Stanley, H. E. Physiobank, physiotoolkit, and physionet. *Circulation*, 101 (23):e215–e220, 2000. doi: 10.1161/01.CIR.101.23.e215. URL https://www.ahajournals.org/doi/abs/10.1161/01.CIR.101.23.e215.

He, K., Zhang, X., Ren, S., and Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, 2015. URL https://arxiv.org/abs/1502.01852.

Hu, J., Shen, L., Albanie, S., Sun, G., and Wu, E. Squeeze-and-excitation networks, 2019. URL https://arxiv.org/abs/1709.01507.

Hubert, L. and Arabie, P. Comparing partitions. *Journal of classification*, 2:193–218, 1985.

Jiang, X., Zhao, J., Du, B., and Yuan, Z. Self-supervised contrastive learning for eeg-based sleep staging. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, 2021a. doi: 10.1109/IJCNN52387.2021.9533305.

Jiang, X., Zhao, J., Du, B., and Yuan, Z. Self-supervised contrastive learning for eeg-based sleep staging. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, 2021b. doi: 10.1109/IJCNN52387.2021.9533305.

Jing, L. and Tian, Y. Self-supervised visual feature learning with deep neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11): 4037–4058, 2021. doi: 10.1109/TPAMI.2020.2992393.

Lee, C.-H., Kim, H., jee Han, H., Jung, M.-K., Yoon, B. C., and Kim, D.-J. Neuronet: A novel hybrid self-supervised learning framework for sleep stage classification using single-channel eeg, 2024a. URL https://arxiv.org/abs/2404.17585.

Lee, S., Yu, Y., Back, S., Seo, H., and Lee, K. Sleepyco: Automatic sleep scoring with feature pyramid and contrastive learning. *Expert Systems with Applications*, 240:122551, April 2024b. ISSN 0957-4174. doi: 10.1016/j.eswa.2023.122551. URL http://dx.doi.org/10.1016/j.eswa.2023.122551.

Li, X., Cui, L., Tao, S., Chen, J., Zhang, X., and Zhang, G.-Q. Hyclasss: A hybrid classifier for automatic sleep stage scoring. *IEEE Journal of Biomedical and Health Informatics*, 22(2):375–385, 2018. doi: 10.1109/JBHI.2017.2668993.

Mohammadi Foumani, N., Mackellar, G., Ghane, S., Irtza, S., Nguyen, N., and Salehi, M. Eeg2rep: Enhancing self-supervised eeg representation through informative masked inputs. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, pp. 5544–5555, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400704901. doi: 10.1145/3637528.3671600. URL https://doi.org/10.1145/3637528.3671600.

Patel, A. K., Reddy, V., Shumway, K. R., and Araujo, J. F. Physiology, sleep stages. In *StatPearls [Internet]*. StatPearls Publishing, 2024.

Phan, H., Mikkelsen, K., Chén, O. Y., Koch, P., Mertins, A., and De Vos, M. Sleeptransformer: Automatic sleep staging with interpretability and uncertainty quantification. *IEEE Transactions on Biomedical Engineering*, 69(8): 2456–2467, 2022. doi: 10.1109/TBME.2022.3147187.

Rechtschaffen, A. A manual of standardized terminology, techniques and scoring system for sleep stages of human subjects. pp. 1–55, 1968.

Rousseeuw, P. J. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987. ISSN 0377-0427. doi: https://doi.org/10.1016/0377-0427(87)90125-7. URL https://www.sciencedirect.com/science/article/pii/0377042787901257.

Sainburg, T., McInnes, L., and Gentner, T. Q. Parametric umap embeddings for representation and semisupervised learning. *Neural Computation*, 33(11):2881–2907, 2021.

Sohn, K. Improved deep metric learning with multi-class n-pair loss objective. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper_files/paper/2016/file/

6b180037abbebea991d8b1232f8a8ca9-Paper.pdf.

Sokolova, M. and Lapalme, G. A systematic analysis of performance measures for classification tasks. *Information processing & management*, 45(4):427–437, 2009.

Šušmáková, K. Human sleep and sleep eeg. *Measurement science review*, 4(2):59–74, 2004.

Van der Maaten, L. and Hinton, G. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need, 2023. URL https://arxiv.org/abs/1706.03762.

Venna, J. and Kaski, S. Neighborhood preservation in nonlinear projection methods: An experimental study. In *International conference on artificial neural networks*, pp. 485–491. Springer, 2001.

Vinh, N. X., Epps, J., and Bailey, J. Information theoretic measures for clusterings comparison: is a correction for chance necessary? In *Proceedings of the 26th annual international conference on machine learning*, pp. 1073–1080, 2009.

A. Additional Results

To adhere to the report's four-page limit, we include supplementary latent space visualizations via PCA and UMAP, together with calculated metrics, in this section.

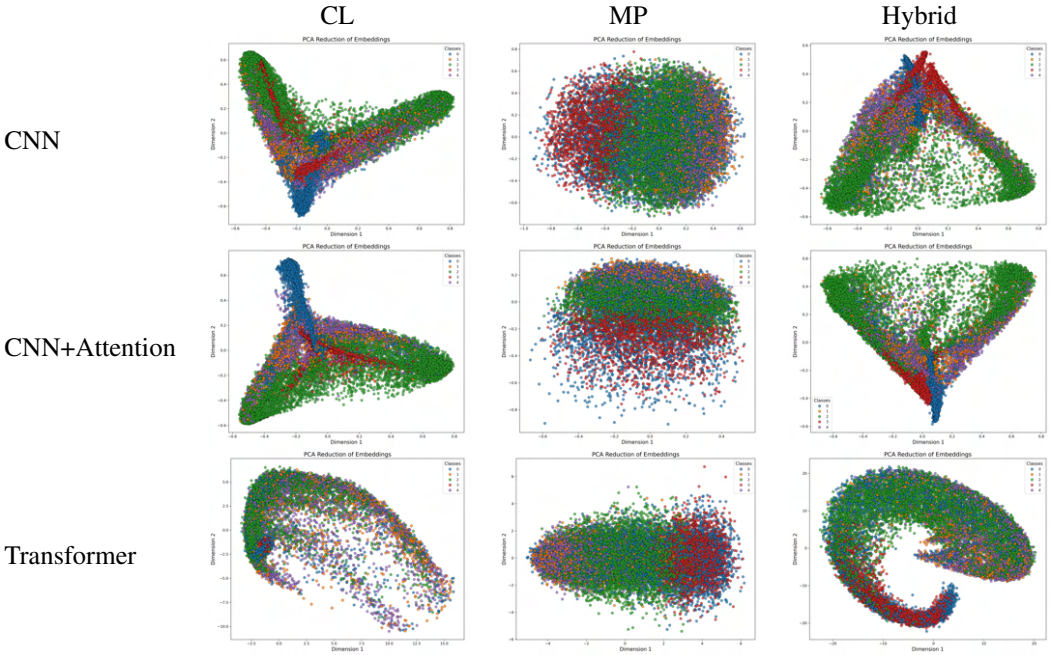

Table 4. Latent Space Visualization using PCA method. This projects latent space to its two dominant principal components.

Table 4 and table 5 show the PCA and UMAP visualizations, respectively. CNN and CNN+Attention backbones exhibit comparable class separation across frameworks, with CNN+Attention showing a slight edge in clarity. Both backbones display distinct clusters in the CRL and Hybrid paradigms, as PCA emphasizes variance-driven separation and UMAP highlights well-defined local structures. However, the Transformer backbone consistently exhibits overlapping clusters, indicating weak latent space organization. In the MP paradigm, CNN and CNN+Attention achieve moderate separation, but the clustering is less distinct than in CRL and Hybrid, and the Transformer backbone again shows poor separation with significant overlap, reflecting its challenges in structuring an effective latent space.

In table 6, we show additional latent space benchmarks that could not fit in the report.

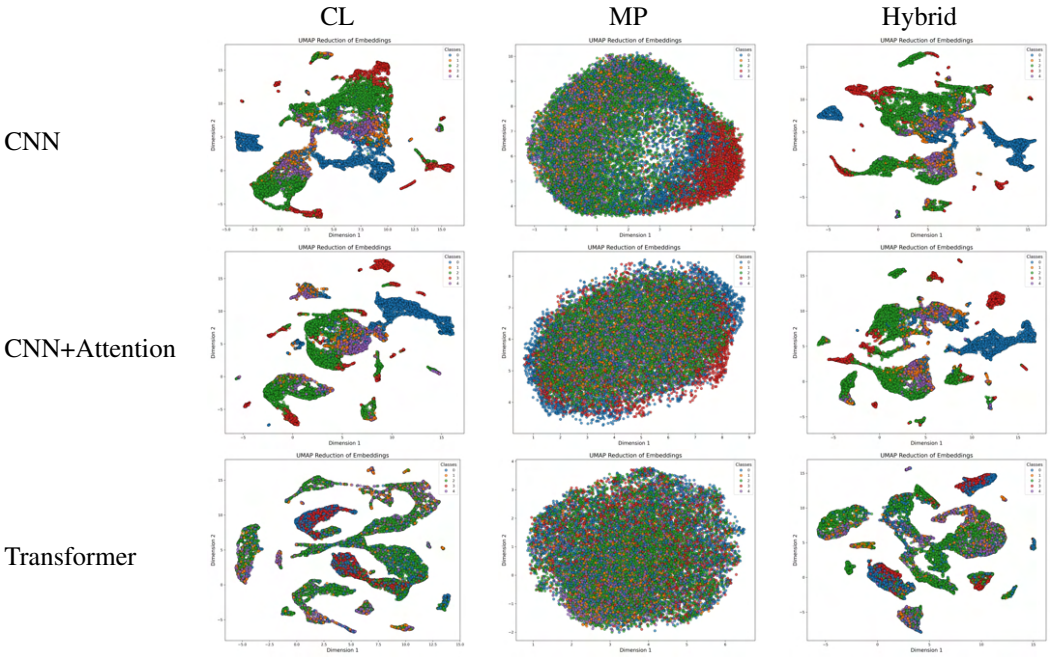

Table 5. Latent Space Visualization using UMAP method.

Model/Paradigm	CRL	MP	Hybrid
CNN	2.118 / 0.217 / 0.489 / 1.0 / 1.569 / 0.403 / 0.271 / -2.948	4.243 / 0.079 / 0.325 / 1.0 / 1.560 / 0.210 / 0.042 / -3.152	2.399 / **0.244** / **0.534** / 1.0 / 1.626 / 0.390 / **0.273** / -3.124
CNN+Attention	2.033 / 0.175 / 0.478 / 1.0 / 1.640 / 0.365 / 0.101 / -3.000	4.142 / 0.087 / 0.319 / 1.0 / **0.888** / 0.081 / 0.057 / -1.708	2.227 / 0.235 / 0.491 / 1.0 / 1.639 / 0.374 / 0.193 / -3.134
Transformer	**1.058** / 0.202 / 0.437 / 1.0 / 80.849 / 20.767 / 0.0 / -5.742	8.212 / 0.120 / 0.381 / 1.0 / 204.492 / 13.694 / 0.044 / -18.412	1.262 / 0.145 / 0.388 / 1.0 / 898.187 / **295.358** / 0.091 / **-17.741**

Table 6. Other latent space benchmark Metrics for all combinations of backbone models and SSL paradigms. Each cell contains: Davies Bould Index(↓) / Adjusted Rand Index(↑) / Purity Score(↑) / Trustworthiness(↑) / Intra-Class Compactness(↓) / Inter-Class Compactness(↑) / Mutual Information(↑) / Uniformity(↓). The best value for each metric is highlighted in bold font.

B. About Latent Space Benchmarks

Evaluating the quality of latent space representations is critical for understanding and improving machine learning models. We employ a suite of quantitative metrics that assess various properties of latent spaces, including clustering quality, structure preservation, and label alignment. In the following, we provide detailed descriptions, explanations, and equations for each metric used in our benchmarks.

Silhouette Score The Silhouette Score (Rousseeuw, 1987) measures how well samples are clustered by evaluating the distance between clusters. For a sample i, the Silhouette Score is computed as:

$$S(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}, \tag{1}$$

where $a(i)$ is the average intra-cluster distance for sample i, and $b(i)$ is the minimum average distance to points in other clusters. The overall score is the mean $S(i)$ across all samples. Higher values indicate better clustering.

Davies-Bouldin Index The Davies-Bouldin Index (Davies & Bouldin, 1979) evaluates cluster compactness and separation. It is defined as:

$$DB = \frac{1}{k} \sum_{i=1}^{k} \max_{j \neq i} \frac{\sigma_i + \sigma_j}{d(c_i, c_j)}, \tag{2}$$

where k is the number of clusters, σ_i is the average intra-cluster distance for cluster i, and $d(c_i, c_j)$ is the distance between cluster centroids c_i and c_j. Lower values indicate better clustering quality.

Adjusted Rand Index (ARI) The ARI (Hubert & Arabie, 1985) measures the similarity between predicted and true cluster assignments, adjusting for chance. It is given by:

$$ARI = \frac{RI - \text{Expected}(RI)}{\max(RI) - \text{Expected}(RI)}, \tag{3}$$

where RI is the Rand Index. The ARI ranges from -1 to 1, with higher values indicating better alignment.

Adjusted Mutual Information (AMI) AMI (Vinh et al., 2009) quantifies the mutual information between predicted clusters and true labels, adjusted for chance. It is expressed as:

$$AMI = \frac{MI - \text{Expected}(MI)}{\max(H(U), H(V)) - \text{Expected}(MI)}, \tag{4}$$

where MI is the mutual information, $H(U)$ and $H(V)$ are the entropies of the true and predicted label distributions. Higher values indicate better agreement.

Purity Score The Purity Score assigns each cluster the most frequent true label and computes the proportion of correctly assigned points. For k clusters, the Purity Score is:

$$\text{Purity} = \frac{1}{N} \sum_{i=1}^{k} \max_{j} |C_i \cap L_j|, \tag{5}$$

where N is the total number of samples, C_i is the set of points in cluster i, and L_j is the set of points with label j.

Average Entropy Average Entropy measures the label distribution's purity within clusters. For k clusters, it is defined as:

$$H = \frac{1}{k} \sum_{i=1}^{k} \sum_{j} -p_{ij} \log(p_{ij}), \tag{6}$$

where p_{ij} is the proportion of points with label j in cluster i. Lower entropy indicates purer clusters.

Trustworthiness Trustworthiness (Venna & Kaski, 2001) evaluates how well the local structure of the data is preserved. It is computed as:

$$T = 1 - \frac{2}{nk(2n - 3k - 1)} \sum_{i=1}^{n} \sum_{j \in U_k(i)} (r_{ij} - k), \tag{7}$$

where $U_k(i)$ is the set of points outside the k-nearest neighbors in the embedding but within the k-nearest neighbors in the original space, and r_{ij} is the rank.

Intra-Class Compactness Intra-Class Compactness evaluates how close points of the same class are to each other. It is defined as the mean distance between points within the same class:

$$C_{intra} = \frac{1}{N} \sum_{i=1}^{N} \sum_{j \in S_i} d(x_i, x_j), \tag{8}$$

where S_i is the set of points in the same class as x_i.

Inter-Class Separation Inter-Class Separation measures how far points of different classes are from each other. It is defined as:

$$C_{inter} = \frac{1}{|P|} \sum_{(i,j) \in P} d(x_i, x_j), \tag{9}$$

where P is the set of point pairs belonging to different classes.

Compactness-to-Separation Ratio This ratio compares intra-class compactness to inter-class separation:

$$R = \frac{C_{intra}}{C_{inter}}. \tag{10}$$

Lower values indicate better separation relative to compactness.

Mutual Information Mutual Information quantifies the dependency between discretized embeddings and true labels. It is given by:

$$MI(U, V) = \sum_{u \in U} \sum_{v \in V} p(u, v) \log \frac{p(u, v)}{p(u)p(v)}, \tag{11}$$

where $p(u, v)$ is the joint probability distribution of true and predicted labels.

Uniformity The Uniformity Metric quantifies the uniformity of the distribution of embeddings by evaluating pairwise squared distances between them. Lower values of the metric indicate a more uniform distribution of the embeddings.

$$\text{Uniformity} = \log\left(\exp\left(-2 \cdot \text{sq\_pdist}\right)\right) \cdot \text{mean} + 10^{-8}. \tag{12}$$

Alignment The Alignment metric quantifies how well the pairwise distances between points in the original space are preserved in the reduced embedding space. It is defined as:

$$\text{Alignment} = \frac{\sum \mathbf{D}_{\mathbf{X}}^2 \cdot \mathbf{D}_{\mathbf{Y}}^2}{\|\mathbf{D}_{\mathbf{X}}^2\| \cdot \|\mathbf{D}_{\mathbf{Y}}^2\|}, \tag{13}$$

where $\mathbf{D_X}$ and $\mathbf{D_Y}$ represent the pairwise Euclidean distances in the original and reduced spaces, respectively, and $\|\cdot\|$ denotes the Euclidean norm. This metric evaluates the preservation of geometric relationships during dimensionality reduction.

C. Data Augmentations

In this section, we provide details about which data augmentations we used in Contrastive Learning and why. We implemented six frequency and time domain augmentations in our Contrastive Learning pipeline and discarded amplitude and noise-based augmentations, as the literature indicates that EEG signal variations are effectively captured within these domains. Each augmentation is applied with a probability of 0.5, resulting in an average of three transformations per signal. This approach ensures sufficient signal distortion to provide strong contrast for the model while preserving essential information.

RandomBandStopFilter Removes a random frequency band to mimic selective frequency attenuation by applying a band-stop filter with center frequency f_c and bandwidth b. The filter response $H(f)$ is defined to maintain Hermitian symmetry and include smooth transition bands to prevent time-domain artifacts:

$$\text{FilteredSignal}[X](t) = \mathcal{F}^{-1}\left[\mathcal{F}[X](f) \cdot H(f)\right]$$

where $\mathcal{F}$ denotes the Fourier transform, and $H(f)$ is given by:

$$H(f) = \begin{cases} 0 & |f - f_c| \leq \frac{b}{2} \\ \frac{1}{2}\left[1 - \cos\left(\pi \frac{|f-f_c| - \frac{b}{2}}{\Delta f}\right)\right] & \frac{b}{2} < |f - f_c| \leq \frac{b}{2} + \Delta f \\ 1 & |f - f_c| > \frac{b}{2} + \Delta f \end{cases}$$

To ensure Hermitian symmetry:

$$H(-f) = H(f)$$

Where Δf defines the width of the transition band.

RandomTimeShift Temporally shifts the signal by a random offset t_s:

$$\text{ShiftedSignal}[X](t) = \begin{cases} X(t + t_s) & \text{if } t + t_s \in \text{domain of } X \\ 0 & \text{otherwise} \end{cases}$$

where t_s is randomly selected. Appropriate padding (zero-padding) is applied for $t + t_s$ outside the original signal domain.

TimeWarping Non-uniformly stretches or compresses segments of the signal along the time axis. For a segment $X_s(t)$ and a scale factor ω:

$$\text{WarpedSegment}[X_s](t) = X_s(\omega t)$$

After warping, the segments are concatenated to match the original signal length:

$$\text{WarpedSignal}[X](t) = \text{Resample}\left(\text{Concat}\left(\text{WarpedSegment}_1, \text{WarpedSegment}_2, \ldots, \text{WarpedSegment}_n\right), T\right)$$

where T is the original time duration of $X(t)$.

Permutation Divides the signal $X(t)$ into n equal-length segments $\{X_1, X_2, \ldots, X_n\}$ and randomly reorders them to disrupt temporal consistency:

$$\text{PermutedSignal}[X] = \text{Concat}\left(X_{\pi(1)}, X_{\pi(2)}, \ldots, X_{\pi(n)}\right)$$

where π is a bijective permutation function such that $\pi : \{1, 2, \ldots, n\} \to \{1, 2, \ldots, n\}$.

RandomZeroMasking Masks a random segment of the signal with zeros. For a mask of length M starting at time t_m:

$$\text{MaskedSignal}[X](t) = \begin{cases} 0 & \text{if } t \in [t_m, t_m + M] \\ X(t) & \text{otherwise} \end{cases}$$

Alternatively, for discrete-time indices:

$$\text{MaskedSignal}[X][i] = \begin{cases} 0 & \text{if } i \in [i_m, i_m + M) \\ X[i] & \text{otherwise} \end{cases}$$

where i is the discrete time index corresponding to t, and i_m is the starting index of the mask.

CutoutResize Randomly removes a segment $X_r(t)$ from the signal and resizes the remaining parts to the original signal length. The process involves:

$$\text{CutoutSignal}[X](t) = \text{Resize}\left(\text{Concat}\left(X_1, \ldots, X_{r-1}, X_{r+1}, \ldots, X_n\right)\right)$$

Where:

- $X = \{X_1, X_2, \ldots, X_n\}$ are equal-length segments of the original signal.
- $X_r(t)$ is the randomly selected segment to remove.
- Resize employs an interpolation method (e.g., linear interpolation) to adjust the concatenated signal back to the original length T.

$$\text{CutoutSignal}[X](t) \in \mathbb{R}^T$$

D. Linear Evaluation Protocol

The training framework employs a **Two-Stage Training Strategy**, integrating Self-Supervised Learning (SSL) followed by supervised linear evaluation.

Stage 1: Self-Supervised Pretraining In the first stage, the encoder model f_θ is pre-trained using Contrastive Representation Learning (CRL), Masked Prediction (MP), and a Hybrid training paradigm to learn invariant feature representations from unlabeled EEG data. Each EEG epoch $\mathbf{x} \in \mathbb{R}^d$ is mapped to a latent representation $\mathbf{z} = f_\theta(\mathbf{x}) \in \mathbb{R}^k$.

Contrastive Loss (NT-Xent) The contrastive loss employed is the **Normalized Temperature-scaled Cross Entropy Loss (NT-Xent)**:

$$\mathcal{L}_{\text{NT-Xent}}(\mathbf{z}_i, \mathbf{z}_j) = -\log \frac{\exp\left(\frac{\text{sim}(\mathbf{z}_i, \mathbf{z}_j)}{\tau}\right)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp\left(\frac{\text{sim}(\mathbf{z}_i, \mathbf{z}_k)}{\tau}\right)}$$

where $\text{sim}(\cdot, \cdot)$ denotes cosine similarity, τ is the temperature parameter, and N is the number of positive pairs per mini-batch.

Masked Prediction (MP) The **L2 Loss** is utilized for the masked prediction task to evaluate the reconstruction quality of masked signal segments:

$$\mathcal{L}_{\text{L2}} = \frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} \left(X_i - \hat{X}_i\right)^2$$

Where:

- $\mathcal{M}$ is the set of masked indices.
- X_i and $\hat{X}_i$ are the ground truth and reconstructed values at index i, respectively.
- $|\mathcal{M}|$ is the total number of masked indices.

We apply **fixed-proportion random masking** as per MAEEG (Chien et al., 2022), selecting the lowest 25% of signal values (sorted by magnitude) to ensure reproducibility.

Hybrid Training Paradigm To leverage both MP and CL, we introduce a **hybrid training paradigm** that processes a masked signal alongside two augmented signals. The total loss combines contrastive and reconstruction losses:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{rec}} + \lambda \mathcal{L}_{\text{cl}}$$

Where:

- $\mathcal{L}_{\text{rec}}$ is the reconstruction loss.

- $\mathcal{L}_{\text{cl}}$ is the contrastive loss.
- $\lambda = 35$ scales the contrastive loss to balance its contribution relative to the reconstruction loss, addressing the observed discrepancy where contrastive loss is smaller by a factor of 30–40.

This weighted combination ensures equal contribution from both training paradigms, enhancing the encoder's ability to learn robust representations.

Optimization The Adam optimizer optimizes the encoder with a learning rate of 1×10^{-3} for a fixed number of epochs. Early stopping based on validation loss prevents overfitting. The best-performing encoder, identified by the lowest validation loss, is retained for downstream evaluation.

Stage 2: Supervised Linear Evaluation : In the second stage, the pre-trained encoder f_θ is **frozen** and acts as a feature extractor. A **Multi-Layer Perceptron (MLP)** classifier g_ϕ is attached to perform supervised classification of sleep stages. The classifier maps latent representations $\mathbf{z}$ to class probabilities $\hat{\mathbf{y}} = g_\phi(\mathbf{z}) \in \mathbb{R}^C$, where $C = 5$ corresponds to the five sleep stages.

The classifier is trained using the **Cross-Entropy Loss**:

$$\mathcal{L}_{\text{CE}}(\mathbf{y}, \hat{\mathbf{y}}) = -\sum_{c=1}^{C} y_c \log p_\phi(y = c \mid \mathbf{z})$$

Where $\mathbf{y}$ is the one-hot encoded true label vector.

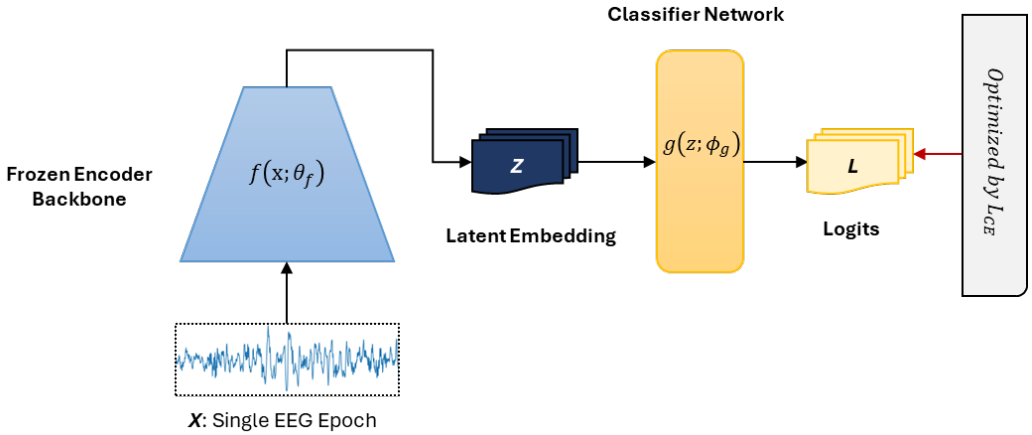

Figure 1. Lnear evaluation protocol

Training uses the Adam optimizer (lr $= 1 \times 10^{-3}$) with early stopping based on validation loss to improve generalization. The best classifier model, identified by the lowest validation loss, is preserved for final evaluation.

This linear evaluation protocol distinctly separates representation learning from classification, ensuring an unbiased assessment of the encoder's ability to extract meaningful EEG features.

E. Training Parameters

The key training parameters for each pre-training framework are as follows.

- Maximum epochs for backbone pre-training: 500
- Batch size: 128
- Learning rate: 0.001

- Weight decay: 0.0001
- Temperature for NT-Xent Loss: 0.1
- Loss weight ratio (α_{CRL}): 35
- Validation interval: 1298 steps
- Early stopping: Patience of 10 epochs