Deep Embedded Clustering regularization for supervised imbalanced cerebral emboli classification using transcranial Doppler ultrasound

1st Yamil Vindas*
yamil.vindas@creatis.insa-lyon.fr

2nd Emmanuel Roux*
emmanuel.roux@creatis.insa-lyon.fr

3rd Blaise Kévin Guépié†
blaise_kevin.guepie@utt.fr

4th Marilys Almar‡
marilys.almar@atysmedical.com

5th Philippe Delachartre*
philippe.delachartre@creatis.insa-lyon.fr

*Univ Lyon, INSA-Lyon, Université Claude Bernard Lyon 1, UJM-Saint Etienne, CNRS, Inserm, CREATIS UMR 5220, U1294, F-69100, LYON, France
†Université de Technologie de Troyes, Laboratoire Informatique et Société Numérique, 10004 Troyes, France
‡Atys Medical, 17 Parc Arbora, 69510 Soucieu-en-Jarrest, France

***Abstract*—High intensity transient signals (HITS) can be detected using transcranial Doppler ultrasound monitoring to help stroke prevention. The various types of HITS are by nature imbalanced: solid emboli are rare events, compared to artifacts or gaseous emboli. Therefore, when training deep models on these data, one have to take into account the class imbalance. In this work, we propose a deep embedded clustering (DEC) based regularization technique, DEC-R to handle imbalanced datasets. Our proposed method decompose a classification model into an encoder and a classifier, with DEC-R applied to the former during supervised training. We validate our method on four synthetic datasets and one HITS datasets. The results show that our approach is robust against imbalanced datasets, allowing to increase the classification performances of different models on both, imbalanced (maximum imbalance ratio of 20) and balanced datasets. We achieve an increase of 4.86% and 1.27% in terms of Matthews correlation coefficient on two imbalanced datasets, a synthetic dataset (of 3D points) and a HITS clinical dataset, respectively. Our code can be found in GitHub[1][2].**

***Index Terms*—Deep Regularization, Imbalanced Data, Embedded Clustering, Emboli Classification, Transcranial Doppler**

I. Introduction

Stroke is a large public health problem as it is one of the leading cause of disability and death in the world [1]. Cerebral emboli (CE) can be associated to the risk of ischemic stroke [2] as they can block a cerebral artery. Therefore, its detection is crucial to help clinicians to prevent stroke.

Moreover, as CE are rare events, a long-duration monitoring is necessary to be able to detect them. To do this, transcranial Doppler ultrasound can be used to monitor the cerebral blood flow over long periods of time. CE are then detected using high intensity transient signals (HITS) which are potential solid or gaseous emboli. What is more, a portable transcranial Doppler can be used to allow longer recordings in ambulatory patients. The two main challenges with this technique are that (1) portable TCD is more prone to artifacts, and (2) the set of obtained HITS is imbalanced, the majority being artifacts and gaseous emboli, whereas the minority are solid emboli.

Furthermore, since the early 2000s, several works have used signal processing and machine learning techniques to detect and classify CE. Signal processing techniques mainly focus on frequency-based approaches using Fourier or wavelet transforms, often combined with machine learning techniques such as support vector machines, random forest, naive Bayes, or fuzzy-rules based classifiers [3]–[7]. Deep learning techniques often use time-frequency representations (TFRs) combined with convolutional neural networks (CNN) [8]–[10]. More recent work have exploited both TFRs and raw signals to classify CE using CNNs and Transformers models [11]. Although both types of methods have reached great performances, to our knowledge, few works use portable TCD data [7], [11], or take into account gaseous emboli for classification [9]–[11]. Moreover, none of these works directly handle the imbalance in the used datasets. On the other hand, deep embedded clustering (DEC) [12] has proven to be effective for unsupervised classification and relatively robust against imbalanced datasets. Therefore, it has been studied in biomedical applications [13]–[15] and adapted to semi-supervised learning contexts [16] for classification and segmentation [17].

In this work, we propose a regularization term based on DEC allowing to partially handle imbalanced datasets while improving the generalization capability of the trained models. Indeed, as shown in [12], DEC is fairly robust against cluster imbalance in an unsupervised context. In contrast, we propose to adapt DEC to a supervised learning context. Therefore, we propose to apply DEC to the encoder's latent space of a supervised learning model. This unsupervised clustering task

[1]Username gitanonymoussubmussion and password AYu45X6UQ9QaWVj3

[2]Mail: gitanoymous@gmail.com, Password: Cr8c4w648cqb9FkG

is done in parallel with the supervised classification task. Thus, any supervised learning can be regularized by our approach.

To summarize, our contributions are the following:

- Proposition of DEC-R, a DEC based regularization term adapting it for a supervised learning context.
- Proposition of DEC-R as a new method to handle imbalanced dataset in a supervised learning context, without the use of the label information (label noise robustness).
- Extensive evaluation using different datasets and classification models.

We validate our proposed approach on five datasets, four 3D synthetic datasets and one HITS datasets. The results show that our proposed method allows improving the classification performances of different types of models trained on several imbalanced and balanced datasets (maximum imbalance ratio of 20).

The rest of the paper is structured as follows. In section II we present our regularization DEC-R for supervised learning classification. In sections III and IV we present the used datasets in the different experiments as well as their results. Finally, in section V we conclude and present the guidelines of our future work.

II. Method

The global pipeline of our method can be found in figure 1. We decompose our classification model into an encoder and a classifier. We handle the problem as a multitask learning problem where the encoder latent's space is used for both unsupervised clustering and supervised classification.

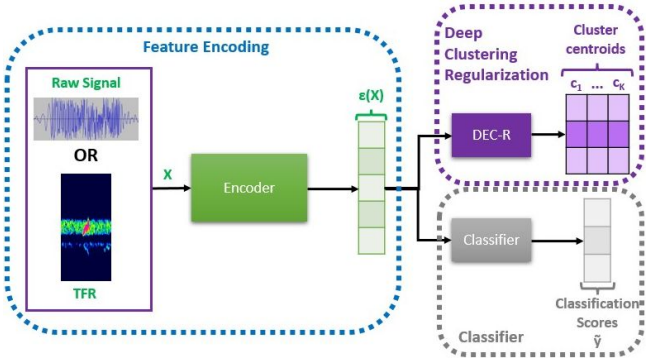

Fig. 1. Pipeline of the proposed method. The DEC-R regularization is applied to the encoder latent's space in parallel to the classification task.

A. Preliminaries

1) Deep embedded clustering: Let us suppose that we have a labeled dataset $(\mathbf{X}_1, y_1)$, ..., $(\mathbf{X}_N, y_N)$, composed of N labeled samples distributed in K classes, where the samples are $\mathbf{X}_1$, ..., $\mathbf{X}_N$ and the labels are y_1, ..., y_N. Let us suppose that we have a classification model $\mathcal{M} = \mathcal{C} \circ \mathcal{E}$ composed of an encoder $\mathcal{E}$ and a classifier $\mathcal{C}$. DEC [12] does clustering on the latent space of $\mathcal{E}$ by using the embedded representations $\mathcal{E}(\mathbf{X}_1)$, ..., $\mathbf{E}_N = \mathcal{E}(\mathbf{X}_N)$ of the input samples.

We denote as $\mathbf{c}_1$, ..., $\mathbf{c}_K$ the centroids of the different clusters, which are initialized using k-means. Note that, as we are in a supervised learning context, we choose the number of centroids to be equal to the number of classes. We can now define, for all $i \in [1, N]$ and $j \in [1, K]$, the soft assignments q_{ij} (interpreted as the probability to assign sample $\mathcal{E}(\mathbf{X}_i)$ to the cluster of centroid c_j) using the Student's t-distribution:

$$\forall i \in [1, N], \forall j \in [1, K], q_{ij} = \frac{\sum_{p=1}^{K} (1 + \frac{||\mathcal{E}(\mathbf{X}_i) - \mathbf{c}_p||^2}{\alpha})^{\frac{\alpha+1}{2}}}{(1 + \frac{||\mathcal{E}(\mathbf{X}_i) - \mathbf{c}_j||^2}{\alpha})^{\frac{\alpha+1}{2}}} \tag{1}$$

where α is the degrees of freedom of the Student's t-distribution, fixed to $\alpha = 1$ for all the experiments presented in this work (as in [12]). We denote as Q the predicted labels' distribution obtained by the q_{ij}.

Moreover, in order to train the encoder model and the centroids, a target distribution P is introduced in [12], defined as follows:

$$\forall i \in [1, N, j \in [1, K], p_{ij} = \frac{\frac{q_{ij}^2}{f_j}}{\sum_{p=1}^{K} \frac{q_{ip}^2}{f_p}} \tag{2}$$

where $f_j = \sum_{p=1}^{K} q_{jp}$ is the soft frequency of cluster j. The DEC module is then optimized using the Kullback-Leibler (KL) divergence between the target distribution P and the predicted labels' distribution Q:

$$\mathcal{L}_{DEC} = KL(P||Q) = \sum_{i=1}^{N} \sum_{j=1}^{K} p_{ij} \times \log(\frac{p_{ij}}{q_{ij}}) \tag{3}$$

2) Supervised classification loss: Our main objective is to do classification, so, in parallel to the unsupervised clustering task, we train a classifier in a supervised manner using the Cross-Entropy loss, $\mathcal{L}_{CE}$, using the encodings $\mathcal{E}(\mathbf{X}_1)$, ..., $\mathcal{E}(\mathbf{X}_N)$ and their respective labels are y_1, ..., y_N.

$$\mathcal{L}_{CE}(\mathcal{C}) = \frac{1}{N} \times \sum_{i=1}^{N} \sum_{j=1}^{K} y_{ij} \times \log(\tilde{y}_{ij}) \tag{4}$$

where y_{ij} is the j^{th} component of $\mathbf{y}_i$ corresponding to the true probability that sample $\mathcal{E}(\mathbf{X}_i)$ is of class j, and $\tilde{y}_{ij} = \mathcal{C}(\mathcal{E}(\mathbf{X}_i))_j$ is the j^{th} component of $\tilde{\mathbf{y}}_i = \mathcal{C}(\mathcal{E}(\mathbf{X}_i))$ corresponding to the predicted probability that sample $\mathcal{E}(\mathbf{X}_i)$ is of class j.

B. Proposed approach

1) Deep embedded clustering regularization (DEC-R) loss for a supervised learning context: In the original paper [12], DEC was applied to pre-trained unsupervised models. In our context, DEC-R adapts DEC to supervised models in an end-to-end manner. To do this, we introduce a hyperparameter e_{init} corresponding to the epoch at which DEC is activated. This means that the proposed DEC regularization is only applied for all epochs greater than e_{init}. Hence, the DEC-R unsupervised loss is denoted here as $\mathcal{L}_{DEC-R}(\mathcal{E}, e_{init})$. In short, from epoch e_{init}, the centroids $\mathbf{c}_1$, ..., $\mathbf{c}_K$ of the K clusters are computed and updated over the iterations thanks to gradient descent on $\mathcal{L}_{DEC-R}$. In this way, the computation of the centroids also has an influence on the encoder's parameters.

2) Final loss: The final loss that we used is the combination of $\mathcal{L}_{CE}(\mathcal{C})$ and $\mathcal{L}_{DEC-R}(\mathcal{E})$, with a hyperparameter γ corresponding to the importance of the deep clustering task:

$$\mathcal{L} = \mathcal{L}_{CE}(\mathcal{C}) + \gamma \times \mathcal{L}_{DEC-R}(\mathcal{E}, e_{init}). \quad (5)$$

Both terms of the final loss depend on the parameters of the $\mathcal{E}$ as $\mathcal{L}_{CE}$ uses the encoded representations of the input samples. The main difference between the two terms is that, $\mathcal{L}_{DEC-R}(\mathcal{E}, e_{init})$ takes into account the centroids $\mathbf{c}_1$, ..., $\mathbf{c}_K$, whereas $\mathcal{L}_{CE}(\mathcal{C})$ takes into account the classification predictions $\tilde{\mathbf{y}}_1$, ..., $\tilde{\mathbf{y}}_N$.

III. Datasets

A. HITS dataset

The HITS dataset is composed of 8685 HITS (696 solid emboli, 1 002 gaseous emboli, 6 987 artifacts) extracted from TCD recordings (30 to 180 minutes), performed on 52 subjects (20 men, 25 women, and 6 unknown; median age 68.5, range 21 to 91, computed with the available information) coming from 10 different centers. The recordings were obtained using two Atys Medical devices, the TCD-X Holter and the WAKIe R3, with a 1.5-2.0 MHz robotized probe. Moreover, for each HITS sample, a TFR was extracted, following [11]. Thus, a HITS sample is composed of a raw signal and its TFR. For more details about the acquisition and pre-processing parameters, we invite the reader to refer to [11].

Last, we split the dataset according to subjects into two subsets, one for training, composed of 84% of the samples, and one for testing, composed of 16% of the samples.

B. Synthetic datasets

In an attempt to study the robustness of DEC-R against imbalanced datasets, we propose to use four synthetic datasets, $\mathcal{D}_{SB}$, $\mathcal{D}_{SI}$, $\mathcal{D}_{NB}$, $\mathcal{D}_{NI}$, with controlled structures. Each dataset is composed of three different clusters of 3D points, $\mathcal{C}_1$, $\mathcal{C}_2$, and $\mathcal{C}_3$. We denote as N_k the number of samples in the k^{th} cluster. Moreover, the points of $\mathcal{C}_k$ are defined as follows:

$$\forall \mathbf{x} \in \mathcal{C}_k, \mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

where $\boldsymbol{\mu}_k = \mu_k \times \mathbf{1}$ is the mean vector (center of the cluster) with $\mathbf{1} = [1, 1, 1]$, and $\boldsymbol{\Sigma}_k = \sigma_k \times \mathbf{I}_3$ is the covariance matrix, with $\mathbf{I}_3 \in \mathbf{R}^3$ is the identity matrix. Using this, we create four different datasets with the following characteristics:

- $\mathcal{D}_{SB}$: linearly separable and balanced dataset with $(\mu_1, \sigma_1) = (-2, 0.5)$, $(\mu_2, \sigma_2) = (0, 0.5)$, $(\mu_3, \sigma_3) = (2, 0.5)$, and $N_1 = N_2 = N_3 = 500$.
- $\mathcal{D}_{SI}$: linearly separable and imbalanced dataset. Same as $\mathcal{D}_{SB}$ but with $N_1 = 1000$, $N_2 = 200$, and $N_3 = 4000$.
- $\mathcal{D}_{NB}$: nonlinearly separable and balanced dataset with $(\mu_1, \sigma_1) = (-1, 0.5)$, $(\mu_2, \sigma_2) = (0, 0.45)$, $(\mu_3, \sigma_3) = (2.5, 0.5)$, and $N_1 = N_2 = N_3 = 500$.
- $\mathcal{D}_{NI}$: nonlinearly separable and imbalanced dataset. Same as $\mathcal{D}_{NB}$ but with $N_1 = 1000$, $N_2 = 200$, and $N_3 = 4000$.

Even if these datasets are multi-class, we use the binary classification term *nonlinearly separable* in the sens that at least two classes are nonlinearly separable (in $\mathcal{D}_{NB}$ and $\mathcal{D}_{NI}$, there is an overlap between $\mathcal{C}_1$ and $\mathcal{C}_2$).

What is more, the synthetic dataset $\mathcal{D}_{NI}$ was designed to have two important characteristics of the HITS dataset: (1) imbalanced classes, and (2) two nonlinearly separable classes. Indeed, as mentioned in subsection III-A, the HITS dataset is by nature imbalanced, with the majority class (artifact), being linearly separable with respect to the two other classes. On the contrary, the minority class (solid emboli class), overlap with the gaseous emboli class (nonlinearly separable), which has twice more samples.

Last, for evaluation, we did a $80/20\%$ train/test split.

IV. Experiments and analysis

We conduct different experiments on the four synthetic datasets and the HITS dataset. We evaluate all the experiments using the Mathew Correlation Coefficients (MCC) and the F1-score. Moreover, all the models were trained with Adam optimizer with a batch size of 32, and all the experiments were repeated 10 times. The reported metrics correspond to the median and mean absolute deviation over the repetitions.

A. Experiment 1: Influence of DEC-R hyperparameters

The objective of this experiment is to study the influence of the two hyperparameters, e_{init} and γ. To do this, we trained a simple classification model using only $\mathcal{D}_{NI}$, and we tried different combinations of the two hyperparameters: $e_{init} \in [0, 1, 5, 10, 20, \frac{n_{epochs}}{2}]$ (with n_{epochs} the total number of training epochs) and $\gamma \in [10, 1, 0.1, 0.01, 0.001]$. The used model is composed of an encoder with one fully connected (FC) linear layer followed by a softplus activation function. Then, a FC layer is applied followed by a softmax function to do classification. The training parameters were the same for all the combinations of γ and e_{init}: 50 epochs and a learning rate of 0.05. The results of this experiment are given in figure 2.

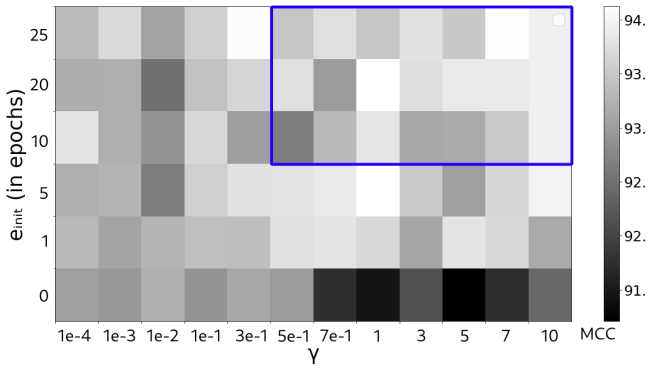

Fig. 2. Experiment 1: influence of the hyperparameters γ and e_{init} on the classification performance (MCC) of the trained model in $\mathcal{D}_{NI}$. The blue zone corresponds to the best choice of hyperparameters used for all the synthetic datasets in experiment 2.

First, for a fixed value of e_{init}, decreasing the value of γ globally reduces the classification performances. Conversely, when γ increases, DEC-R term becomes stronger, giving a more clustered and separable latent space. Secondly, for a fixed

TABLE I
EXPERIMENT 1: RESULTS OF THE EVALUATION OF DEC-R ON SYNTHETIC DATASETS ($\mathcal{D}_{SB}$, $\mathcal{D}_{SI}$, $\mathcal{D}_{NB}$, AND $\mathcal{D}_{NI}$).

Dataset	DEC-R	MCC	F1-Score
$\mathcal{D}_{SB}$	No	100 ± 0	100 ± 0
	Yes	100 ± 0	100 ± 0
$\mathcal{D}_{\mathbf{SI}}$	No	89.65 ± 2.37	95.62 ± 0.91
	Yes	$\mathbf{94.51 \pm 2.09}$	$\mathbf{97.88 \pm 0.77}$
$\mathcal{D}_{NB}$	No	$\mathbf{96.99 \pm 0.51}$	$\mathbf{98.0 \pm 0.33}$
	Yes	96.25 ± 0.74	97.50 ± 0.50
$\mathcal{D}_{\mathbf{NI}}$	No	91.74 ± 1.98	96.59 ± 0.77
	Yes	$\mathbf{93.38 \pm 0.9}$	$\mathbf{97.60 \pm 0.38}$

value of γ, increasing the value of e_{init} globally increases the classification performances (the original DEC was designed to be used with pre-trained models). Thirdly, we can see that the classification performances obtained with $e_{init} = 0$ are globally smaller compared to the other values of e_{init}. This is because DEC-R is initialized with k-means, and for $e_{init} = 0$, the latent space where DEC-R is applied does not have a rich structure, so the initialized centroids are not optimal, leading to poor results. Based on these remarks, we recommend choosing the hyperparameters, γ and e_{init} as follows. First, choose e_{init} starting from low values for a fixed small γ value. This allows measuring the impact of DEC-R without an important degradation of the classification performance. Then, choose γ by gradually increasing it for the previously chosen e_{init}.

B. Experiment 2: Robustness of DEC-R against imbalanced datasets

The objective of this experiment is to study the robustness of DEC-R regularization in a controlled environment using synthetic datasets. Therefore, we used $\mathcal{D}_{SB}$, $\mathcal{D}_{SI}$, $\mathcal{D}_{NB}$, and $\mathcal{D}_{NI}$ to train the same classification model of experiment 1 with and without DEC-R. Furthermore, following the recommendations of our hyperparameter study in experiment 1 (IV-A), we obtained the following values: $\gamma = 1$ and $e_{init} = 5$ for $\mathcal{D}_{SB}$ and $\mathcal{D}_{NI}$; $\gamma = 10$ and $e_{init} = 5$ for $\mathcal{D}_{SI}$; $\gamma = 0.5$ and $e_{init} = 1$ for $\mathcal{D}_{NB}$. Moreover, on $\mathcal{D}_{SB}$, $\mathcal{D}_{NI}$, and $\mathcal{D}_{SI}$, we trained the models during 50 epochs with a learning rate of 0.05 for the two first datasets and 0.005 for the last one. Last, for $\mathcal{D}_{NB}$ the model was trained during 100 epochs with a learning rate of 0.01. All the models used a weight decay of 10^{-7}, and the results can be found in table I.

For both imbalanced datasets, $\mathcal{D}_{SI}$ and $\mathcal{D}_{NI}$, the regularized model outperformed the unregularized one by a margin of 4.86% and 1.64% in terms of MCC respectively. Indeed, the DEC-R regularized models tend to produce more compact clusters, allowing a better separability when doing classification.

Additionally, we observe that regularizing with DEC-R does not reduce significantly the classification performances in any dataset, while often reducing the variability, thus giving more stable models.

C. Experiment 3: Application to cerebral emboli classification

The objective of this experiment is twowise: (1) apply our proposed method to a real HITS imbalanced dataset for CE classification, and (2) compare our proposed method to other commonly used methods for imbalanced classification, namely undersampling (to reach the number of samples of the minority class) and class weights [18]. Therefore, we applied our method to two different models of [11]: (1) a 2D CNN model taking as input the TFR, and (2) a 1D CNN-Transformer taking as input the raw signal. Hereafter we list the small changes that we did for the 1D CNN-Transformer model compared to the original work in [11]: last 1D convolutional layer is applied 4 times, 16 attention heads (per multi-head attention), 4 Transformer encoder layers, a Transformer intermediate hidden dimension of 64, and a dimension of 16 for the projected representation used for final classification. Additionally, to test the resistance of DEC-R to noisy-labels, we added symmetric noise to the labels with a noise rate of 20%. Furthermore, the 2D CNN models were trained with a weight decay of 10^{-4}, and a learning rate of 10^{-3}, whereas the 1D CNN-Transformers used a weight decay of 10^{-5} and a learning rate of 0.1 when undersampling was applied and 0.04 when not. The DEC-R parameters can be found in table II. Last, all models were trained during 75 epochs expect for the 1D CNN-Transformer models with undersampling (required 100 epochs). We computed the MCC and F1-score over the last 5 epochs, and results can be found in table III.

First, we can observe that, with respect to undersampling, DEC-R allows increasing the performances of the models, with up to 2.85% in terms of MCC. Indeed, DEC-R allows not only to use all the available samples (compared to undersampling), but it also enforces clustering[3]. Moreover, for the 1D CNN-Transformer model, DEC-R alone outperforms all the other methods. Nevertheless, for the 2D CNN model, class weights (combined to DEC-R) improves the results, but it stays below the 1D CNN-Transformer. However, contrary to class weights, our method does not depend on the labels of the samples, which makes it more robust against noisy-labeled datasets as the results in table IV show it.

Finally, for both models, the best performing methods are the ones using DEC-R (with class weights for the 2D CNN model and without for the 1D CNN-Transformer), with an increase in MCC up to 2.02% with respect to baselines where no imbalance classification method is used.

V. CONCLUSION

In this work, we proposed DEC-R, a regularization for supervised learning models based on deep embedded clustering. Our method decomposes a classification model into an encoder and a classifier, with an unsupervised DEC regularization applied to the encoder while doing supervised training of the entire model. Our experiments show that, on both synthetic

[3]We refer the reader to the GitHub associated to our work for more experimental details assessing this point.

TABLE II
EXPERIMENT 3: DEC-R PARAMETERS.

Model	Undersampling	Class Weights	γ	$\mathbf{e_{init}}$
2D CNN	Yes	-	0.1	10
	No	No	0.1	1
		Yes	0.01	1
1D CNN-Trans.	Yes	-	1	70
	No	No	0.0001	1
		Yes	1	1

TABLE III
EXPERIMENT 3: RESULTS OF THE EVALUATION OF DEC-R ON THE HITS DATASET IN %. *No* MEANS THAT NO METHOD IS USED. US AND CW STAND FOR UNDERSAMPLING AND CLASS WEIGHTS, RESPECTIVELY

Model	Imbalance Method	MCC	F1-Score
2D CNN	No	81.13 ± 1.72	85.04 ± 1.10
	US	78.36 ± 2.17	83.67 ± 1.50
	CW	*83.03 ± 1.76*	*86.34 ± 1.26*
	DEC-R	81.21 ± 1.81	84.69 ± 1.43
	US+DEC-R	78.95 ± 1.55	83.91 ± 0.96
	CW+DEC-R	$\mathbf{83.15 \pm 1.20}$	$\mathbf{86.41 \pm 0.88}$
1D CNN-Trans.	No	86.23 ± 0.94	89.31 ± 0.58
	US	84.95 ± 1.51	88.47 ± 0.89
	CW	86.52 ± 1.11	89.47 ± 0.73
	DEC-R	$\mathbf{87.50 \pm 1.19}$	$\mathbf{89.87 \pm 0.56}$
	US+DEC-R	85.07 ± 1.04	88.39 ± 0.67
	CW+DEC-R	*86.98 ± 1.05*	*89.76 ± 0.67*

data and real TCD data, a DEC-R regularization on the encoder allows facing class imbalance with a more clustered latent space: it improves the classification performances of different classification models (DNN, CNN, and Transformers) in imbalanced and balanced datasets, with inputs of different natures, specially when noise is present in the labels. In future work, we plan to apply DEC-R to multi-feature models.

ACKNOWLEDGMENT

This work was carried out in the context of the CAREMB project funded by the Auvergne-Rhône-Alpes region, within the *Pack Ambition Recherche* program. This work was performed within the framework of the LABEX CELYA (ANR-10-LABX-0060) and PRIMES (ANR-11-LABX-0063) of Université de Lyon, within the program "Investissements d'Avenir" (ANR-11-IDEX-0007) operated by the French National Research Agency (ANR).

TABLE IV
EXPERIMENT 3: RESULTS OF THE EVALUATION OF DEC-R ON THE NOISY-LABELS HITS DATASET IN %. WE FIXED THE LABEL NOISE TO 20%. THE TWO COMPARED METHODS ARE DEC-R AND CLASS WEIGHTS, CW.

Model	Imbalance Method	MCC	F1-Score
2D CNN	CW	45.20 ± 4.18	63.45 ± 2.99
	DEC-R	$\mathbf{52.32 \pm 5.30}$	$\mathbf{65.81 \pm 3.78}$
1D CNN-Trans.	CW	84.49 ± 1.46	87.82 ± 1.01
	DEC-R	$\mathbf{85.27 \pm 1.93}$	$\mathbf{88.60 \pm 1.21}$

REFERENCES

[1] M. George, "CDC Grand Rounds: Public Health Strategies to Prevent and Treat Strokes," *MMWR. Morbidity and Mortality Weekly Report*, vol. 66, 2017.

[2] M. Rosenkranz, J. Fiehler, W. Niesen, C. Waiblinger, B. Eckert, O. Wittkugel, T. Kucinski, J. Röther, H. Zeumer, C. Weiller, and U. Sliwka, "The amount of solid cerebral microemboli during carotid stenting does not relate to the frequency of silent ischemic lesions," *American Journal of Neuroradiology*, vol. 27, no. 1, pp. 157–161, 2006.

[3] A. Karahoca and M. Tunga, "A polynomial based algorithm for detection of embolism," *Soft Computing*, vol. 19, no. 1, pp. 167–177, 2015.

[4] G. Serbes and N. Aydin, "Denoising performance of modified dual-tree complex wavelet transform for processing quadrature embolic doppler signals," *Medical & Biological Engineering & Computing*, vol. 52, no. 1, pp. 29–43, 2014.

[5] P. Sombune, P. Phienphanich, S. Muengtaweepongsa, A. Ruamthanthong, and C. Tantibundhit, "Automated embolic signal detection using adaptive gain control and classification using ANFIS," in *EMBC*, pp. 3825–3828, IEEE, 2016.

[6] B. Guépié, B. Sciolla, F. Millioz, M. Almar, and P. Delachartre, "Discrimination between emboli and artifacts for outpatient transcranial doppler ultrasound data," *Medical & Biological Engineering & Computing*, vol. 55, no. 10, pp. 1787–1797, 2017.

[7] B. Guepie, M. Martin, V. Lacrosaz, M. Almar, B. Guibert, and P. Delachartre, "Sequential emboli detection from ultrasound outpatient data," *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 1, pp. 334–341, 2019.

[8] P. Sombune, P. Phienphanich, S. Phuechpanpaisal, S. Muengtaweepongsa, A. Ruamthanthong, and C. Tantibundhit, "Automated embolic signal detection using deep convolutional neural network," in *EMBC*, pp. 3365–3368, IEEE, 2017.

[9] A. Tafsast, K. Ferroudji, M. Hadjili, A. Bouakaz, and N. Benoudjit, "Automatic microemboli characterization using convolutional neural networks and radio frequency signals," in *ICCEE*, pp. 1–4, IEEE, 2018.

[10] Y. Vindas, B. Guépié, M. Almar, E. Roux, and P. Delachartre, "Semi-automatic data annotation based on feature-space projection and local quality metrics: an application to cerebral emboli characterization," *Medical Image Analysis*, p. 102437, 2022.

[11] Y. Vindas, B. Guépié, M. Almar, E. Roux, and P. Delachartre, "An hybrid cnn-transformer model based on multi-feature extraction and attention fusion mechanism for cerebral emboli classification," in *MLHC*, PMLR, 05–06 Aug 2022.

[12] J. Xie, R. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *ICML*, ICML'16, p. 478–487, JMLR.org, 2016.

[13] X. Li, S. Zhang, and K.-C. Wong, "Deep embedded clustering with multiple objectives on scRNA-seq data," *Briefings in Bioinformatics*, vol. 22, 04 2021. bbab090.

[14] J. Forte, G. Yeshmagambetova, M. Grinten, B. Hiemstra, T. Kaufmann, R. Eck, F. Keus, A. Epema, M. Wiering, and I. Horst, "Identifying and characterizing high-risk clusters in a heterogeneous icu population with deep embedded clustering," *Scientific Reports*, vol. 11, p. 12109, 06 2021.

[15] P. Ajay, B. Nagaraj, R. Kumar, R. Huang, and P. Ananthi, "Unsupervised hyperspectral microscopic image segmentation using deep embedded clustering algorithm," *Scanning*, vol. 2022, pp. 1–9, 06 2022.

[16] Y. Ren, K. Hu, X. Dai, L. Pan, S. C. Hoi, and Z. Xu, "Semi-supervised deep embedded clustering," *Neurocomputing*, vol. 325, pp. 121–130, 2019.

[17] J. Enguehard, P. O'Halloran, and A. Gholipour, "Semi-supervised learning with deep embedded clustering for image classification and segmentation," *IEEE Access*, vol. 7, pp. 11093–11104, 2019.

[18] G. King and L. Zeng, "Logistic regression in rare events data," *Political Analysis*, vol. 9, p. 137–163, Spring 2001.