HOLISTIC SEMI-SUPERVISED APPROACHES FOR EEG REPRESENTATION LEARNING

Guangyi Zhang and Ali Etemad

Department of Electrical and Computer Engineering
Queen's University, Kingston, Canada

ABSTRACT

Recently, supervised methods, which often require substantial amounts of class labels, have achieved promising results for EEG representation learning. However, labeling EEG data is a challenging task. More recently, holistic semi-supervised learning approaches, which only require few output labels, have shown promising results in the field of computer vision. These methods, however, have not yet been adapted for EEG learning. In this paper, we adapt three state-of-the-art holistic semi-supervised approaches, namely MixMatch [1], FixMatch [2], and AdaMatch [3], as well as five classical semi-supervised methods for EEG learning. We perform rigorous experiments with all 8 methods on two public EEG-based emotion recognition datasets, namely SEED and SEED-IV. The experiments with different amounts of limited labeled samples show that the holistic approaches achieve strong results even when only 1 labeled sample is used per class. Further experiments show that in most cases, AdaMatch is the most effective method, followed by MixMatch and FixMatch.

Index Terms— Semi-supervised learning, deep learning, electroencephalography, emotion recognition

1. INTRODUCTION

Human emotions are highly informative non-verbal cues that can be widely used to enhance human-machine interaction. Many solutions have been proposed for affective computing [4] using Electroencephalography (EEG), as EEG is widely used for directly measuring brain activity with high spatio-temporal resolution. Recently, various deep learning models have achieved state-of-the-art performances in emotion recognition tasks due to the capability of learning highly discriminative information from multichannel EEG recordings [5, 6]. However, as the significant majority of existing EEG-based deep emotion recognition solutions are '*supervised*' methods [6, 7, 8, 9, 10], they rely on the annotated labels. Labeling EEG, on the other hand, is difficult, time-consuming, and costly. To obtain reliable labels, many EEG-based studies require various emotion annotation approaches, including pre-stimulation and post-experiment self-assessments, as well as multiple expert evaluations [5, 6, 11]. Moreover, most existing fully supervised learning algorithms suffer from performance drop when only a few training samples are labeled. As a result, it is important to investigate how to develop effective solutions that can deal with limited labeled training samples.

Semi-Supervised Learning (SSL) has been recognized as a powerful paradigm to address the problems of a scarcity of labeled training samples, achieving great success in the field of computer vision [12]. For example, pseudo-labeling was proposed to encourage low-entropy predictions of unlabeled data [13]. To do so, a model was first trained using limited labeled data. The model was then used to predict pseudo-labels for the unlabeled data. Finally, the model was retrained using the entire data with true labels and pseudo labels combined [13]. Consistency regularization, in conjunction with data augmentation, has lately become popular in SSL studies [14, 15, 16]. Common regularization techniques including stochastic augmentation applied to input data and dropout applied throughout the network have been employed in recent SSL frameworks [14]. For instance, the Π-model [14] trained the network (with dropout) on both the original and augmented inputs, and minimized the distance between their corresponding outputs, as an unsupervised loss. Meanwhile, a supervised cross-entropy loss was only computed for the labeled set. To improve the Π-model, 'temporal ensembling' and 'mean teacher' methods further relied on consistency regularization techniques for SSL [14, 15].

In addition to the classical SSL approaches mentioned above, several new holistic SSL methods [1, 2, 3] have been recently proposed and achieved state-of-the-art results (we discuss these in detail in the next Section). However, their use has been mostly limited to computer vision tasks, and have therefore not been explored in other domains such as brain-computer interfaces (BCI) or biological signal processing. Only in a recent paper [17], few of the classical SSL frameworks were used for BCI with EEG representation learning. In this paper, we ***adapt, implement, and where necessary, modify*** three state-of-the-art holistic SSL methods, MixMatch [1], FixMatch [2], and AdaMatch [3], in addition to the five classical SSL methods, Π-model [14], temporal assembling [14], mean teacher [15], convolutional autoencoders [18], and pseudo labeling [13], for EEG representation learning. We then ***perform a comprehensive study***, comparing these methods against one another when different amounts of
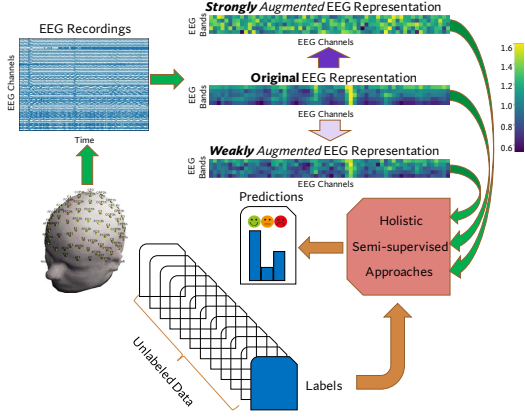
Fig. 1. An overview of holistic SSL approaches for EEG representation learning in the training phase is presented.

labels are used per class. Our study shows that these methods can indeed be effectively used in the field of BCI for EEG learning, with the holistic methods, particularly AdaMatch and MixMatch, showing reasonable performances even when 1 label per class is used for training. An overview of the holistic SSL methods for EEG learning is shown in Figure 1.

2. HOLISTIC SELF-SUPERVISED EEG LEARNING

2.1. Problem Setup

Let's denote $\mathcal{D}_l = \{(x_i^l, y_i^l)\}_{i=1}^M$, $\mathcal{D}_u = \{x_i^u\}_{i=1}^N$, $\mathcal{D}_v = \{(x_i^l, y_i^l)\}_{i=1}^I$, and $\mathcal{D}$ as the labeled, unlabeled, validation, and entire training sets, where x and y represent EEG data and class labels. Here, M, N, and I are the size of the three subsets. We have $\mathcal{D}_l \cup \mathcal{D}_u \cup \mathcal{D}_v = \mathcal{D}$ and $\mathcal{D}_l \cap \mathcal{D}_u \cap \mathcal{D}_v = \emptyset$. We randomly choose $M = m \times k$ samples to form $\mathcal{D}_l$, where k and m are the number of output classes and labeled training samples per class. Given our interest in relying only a few labeled samples, we set $M \ll N$.

Notation: We denote $\{(x_b^l, y_b^l)\}$ and $\{x_b^u\}$ as labeled and unlabeled sets in each training batch, where B is the batch size. Strong and weak augmentations are represented by $\mathcal{A}_s$ and $\mathcal{A}_w$, respectively (discussed in Section 3.4). We also use the terms p_m and θ to refer to the deep learning model and model parameters, respectively.

2.2. Holistic SSL Approaches

MixMatch. To adapt MixMatch [1] for EEG learning, we first perform a data augmentation on *labeled* data to obtain $\mathcal{A}_w(x_b^l)$. We then apply two different data augmentations on the *unlabeled* data and calculate the model predictions:

$$q_b^x = p_m(y \mid \mathcal{A}_w(x_b^l); \theta) \quad \text{and} \tag{1}$$

$$q_b^{u1} = p_m(y \mid \mathcal{A}_w(x_b^u); \theta), q_b^{u2} = p_m(y \mid \mathcal{A}_s(x_b^u); \theta), \tag{2}$$

where q_b^x is the model prediction on augmented labeled inputs. q_b^{u1} and q_b^{u2} are the two model predictions on the same unlabeled input (x_b^u) with different augmentations.

In the second phase, we average the predictions over the augmented unlabeled data. We then minimize the entropy of the label distribution (a process known as sharpening), using:

$$\text{sharpen}(p_b, T) = p_b^{1/T} \Big/ \sum_{1}^{k}(p_b^{1/T}), \tag{3}$$

where p_b is the average prediction as $p_b = (\text{softmax}(q_b^{u1}) + \text{softmax}(q_b^{u2}))/2$. T is the 'temperature' to adjust the entropy of the label distribution. We choose $T = 1$ as suggested in [1].

In the third phase, we use MixUp [19] to combine the augmented *labeled* data along with the corresponding labels, with the augmented *unlabeled* data along with the sharpened predictions. This creates new mixed labeled and unlabeled sets, $(\tilde{x}_b^l, \tilde{y}_b^l)$ and $(\tilde{x}_b^u, \tilde{y}_b^u)$ [1]. The process is as follows:

$$\lambda \sim \mathcal{B}(\alpha, \alpha), \tag{4}$$

$$\lambda' = \max(\lambda, 1 - \lambda), \tag{5}$$

$$\mathcal{M} = \{(\langle x_b^l, \mathcal{A}_w(x_b^u), \mathcal{A}_s(x_b^u)\rangle, \langle y_b^l, p_b^u, p_b^u\rangle)\}, \tag{6}$$

$$\mathcal{M}' = \text{shuffle}(\mathcal{M}), \tag{7}$$

$$(\tilde{x}_b^l, \tilde{y}_b^l) = \lambda'\{\mathcal{M}\}_{i=1}^{|x_b^l|} + (1 - \lambda')\{\mathcal{M}'\}_{i=1}^{|x_b^l|}, \tag{8}$$

$$(\tilde{x}_b^u, \tilde{y}_b^u) = \lambda'\{\mathcal{M}\}_{i=|x_b^l|}^{|x_b^l|+2|x_b^u|} + (1 - \lambda')\{\mathcal{M}'\}_{i=|x_b^l|}^{|x_b^l|+2|x_b^u|}, \tag{9}$$

where $\langle , \rangle$ represents the concatenation operation of two or more sets. $\mathcal{B}$ represents Beta distribution with $\alpha = 0.75$. p_b^u is the sharpened prediction based on $\text{sharpen}(p_b, T)$. $\mathcal{M}$ is the concatenation of augmented labeled and unlabeled sets, $\mathcal{M}'$ is the set of $\mathcal{M}$ after a random shuffle operation, and $|.|$ is the size of the set.

In the last step, we train the model on mixed data. Our semi-supervised loss comprises of a supervised term

$$\mathcal{L}_s = \mathcal{H}(\tilde{y}_b^l, p_m(y \mid \tilde{x}_b^l; \theta)), \tag{10}$$

with cross-entropy loss ($\mathcal{H}$), and an unsupervised term

$$\mathcal{L}_u = (p_m(y \mid \tilde{x}_b^u; \theta) - \tilde{y}_b^u)^2. \tag{11}$$

The overall semi-supervised loss is then calculated as $\mathcal{L} = \mathcal{L}_s + w\mathcal{L}_u$, where w is a warm-up function that is used to gradually increase the weight of the unsupervised term as the number of training epochs increases [1].

FixMatch. FixMatch, proposed in [2], provides a simpler method for combining pseudo-labeling with consistency regularization. FixMatch employs cross-entropy between model predictions and pseudo-labels rather than calculating the squared difference as the unsupervised loss. The model prediction is computed using strongly augmented unlabeled data as $p_m(y \mid \mathcal{A}_s(x_b^u))$. The pseudo-label is generated using weakly augmented unlabeled data as $y_b^u = \arg\max(q_b^u)$, where $q_b^u = p_m(y \mid \mathcal{A}_w(x_b^u); \theta)$.

Following that, we utilize a pre-defined confidence threshold τ to guarantee that pseudo-labels are only used to update the loss when the model prediction is confident:

$$\mathcal{L}_u = \mathbb{I}(\max(q_b^u) \geq \tau) \cdot \mathcal{H}(y_b^u, p_m(y \mid \mathcal{A}_s(x_b^u); \theta)). \quad (12)$$

Here, $\mathbb{I}$ is the binary mask and $\max(q_b^u)$ represents the highest class probability in the prediction (q_b^u).

Next, we use the supervised term

$$\mathcal{L}_s = \mathcal{H}(y_b^l, p_m(y \mid x_b^l; \theta)). \quad (13)$$

The total loss is the sum of the supervised and unsupervised terms as $\mathcal{L} = \mathcal{L}_u + \mathcal{L}_s$. No warm-up function is used since the weight of the unsupervised term will automatically rise as the model's confidence grows with each training epoch [2].

AdaMatch. To adapt AdaMatch [3], which is an improved SSL method based on FixMatch, we first obtain the predictions for weakly augmented *labeled* and *unlabeled* data as $q_b^l = p_m(y \mid \mathcal{A}_w(x_b^l); \theta)$ and q_b^u, respectively, where $q_b^l, q_b^u \in \mathbb{R}^{B \times k}$. Let's represent the class distribution of *labeled* data using the expected values of predictions for the weakly augmented *labeled* data, as $\mathbb{E}(q_b^l) \in \mathbb{R}^k$, and the estimated class distribution of *unlabeled* data using the expected values of predictions for the weakly augmented *unlabeled* data, as $\mathbb{E}(q_b^u) \in \mathbb{R}^k$, where $\mathbb{E}(.)$ is expectation operator. Accordingly, the estimated labels for *unlabeled* data are rectified by multiplying them with the ratio of class distributions of *labeled* data to the estimated class distribution of *unlabeled* data. This operation can be denoted as

$$\tilde{q}_b^u = ||q_b^u(\mathbb{E}(q_b^l)/\mathbb{E}(q_b^u))||, \quad (14)$$

where $\tilde{q}_b^u \in \mathbb{R}^{B \times k}$, and the L1 normalization operator $||.||$ ensures that the class distribution sums up to 1.

Following, the predefined threshold is multiplied by the mean value of the most confident prediction for the weakly augmented labeled data as $\tau \cdot \mu_{\max(q_b^l)} \in \mathbb{R}$, which is referred to as the relative confidence threshold [3], where $\max(q_b^l) \in \mathbb{R}^B$. Consequently, the unsupervised loss is calculated as

$$\mathcal{L}_u = \mathbb{I}(\max(\tilde{q}_b^u) \geq \tau \cdot \mu_{\max(q_b^l)}) \cdot \mathcal{H}(\tilde{y}_b^u, p_m(y \mid \mathcal{A}_s(x_b^u); \theta)), \quad (15)$$

where $\tilde{y}_b^u = \arg\max(\tilde{q}_b^u)$, $\mathbb{I} \in \{0, 1\}^B$, $\max(\tilde{q}_b^u) \in \mathbb{R}^B$, and $\tilde{y}_b^u \in \{0, 1\}^{B \times k}$.

The total loss is the weighted sum of the supervised term ($\mathcal{L}_s$ in Eq. 10) and the unsupervised term ($w\mathcal{L}_u$), where w is a warm-up function used in [3], as $\mathcal{L} = \mathcal{L}_s + w\mathcal{L}_u$.

3. EXPERIMENTAL SETUP AND RESULTS

3.1. Datasets

SEED. The SEED dataset [5] contains EEG recorded using 62 electrodes at the sampling rate of 1000 *Hz*. In the experiments, 15 film clips with three emotions (neutral, positive, and negative) were selected as stimuli. The studies were

completed by a total of 15 individuals, 8 females and 7 males. Each subject takes part in the experiment twice, with each experiment consisting of 15 recording sessions.

SEED-IV. The SEED-IV dataset [6] contains 62-channel EEG recordings obtained with a sample frequency of 1000 *Hz*. 72 video snippets with four emotions (neutral, fear, sad, and happy) were used as stimuli. The experiments were carried out by 15 participants (8 females and 7 males). Each subject repeated the experiment three times, with different stimuli used each time. Each experiment has 24 recording sessions (6 sessions for each emotion).

3.2. Pre-processing and Feature Extraction

We follow the exact pre-processing steps described in [5, 6] as follows. We first down-sample the EEG to 200 *Hz*. Then, to reduce artefacts, band-pass filters with frequencies ranging from $0.3 - 50$ *Hz* and $1 - 75$ *Hz* were applied to recordings from the SEED and SEED-IV. Following pre-processing, EEG were split into continuous segments of the same length (1 second for SEED and 4 seconds for SEED-IV) with no overlap. Differential Entropy (DE) features were extracted from five bands (delta, theta, alpha, beta, and gamma) of each EEG segment. We assume EEG signals obey a Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$, and thus DE can be calculated as $DE = \frac{1}{2} \log 2\pi e \sigma^2$.

3.3. Evaluation Protocols

We apply the same evaluation protocols that were used in the original dataset publications [5, 6]. For SEED, we use the first 9 sessions for training and the remaining 6 sessions for testing. For SEED-IV, we use the first 16 sessions for training, and the rest 8 sessions for testing. We use the classification accuracy to measure the performance of the semi-supervised techniques for recognition of three emotions (neutral, positive, and negative) in SEED and four emotions (neutral, fear, sad, and happy) in SEED-IV.

3.4. Implementation details

Data Augmentation. We use additive Gaussian noise to augment the EEG signals. We denote the augmented data as $\mathcal{A}(x) = x + \mathcal{N}(\mu, \sigma)$, where $x \sim [0, 1]$, $\mathcal{N}$ is a Gaussian distribution with $\mu = 0.5$. The intensity level of data augmentation can be tuned by changing σ. We choose σ of 0.8 and 0.2 in additive Gaussian Noise to represent strong ($\mathcal{A}_s$) and weak ($\mathcal{A}_w$) augmentations, respectively.

Backbone Network. As our backbone network, we use an efficient lightweight CNN architecture, consisting of two convolutional blocks and one classifier. Each block consists of a 1-D convolutional layer followed by a 1-D batch normalization layer and a LeakyReLU activation. The classifier contains two fully connected layers with a dropout rate of 0.5.

Hyper-parameters. In all the experiments, we run 30 training epochs with a batch size of 8. We employ Adam optimizer [20] with the default learning rate of 0.001. Tuning

Table 1. The performance of holistic approaches in comparison to other semi-supervised methods on SEED and SEED-IV.

Method / m	SEED						SEED-IV					
	1 label	3 labels	5 labels	7 labels	10 labels	25 labels	1 label	3 labels	5 labels	7 labels	10 labels	25 labels
Π-model	60.25 (9.57)	67.87(10.14)	72.43(11.32)	74.94(10.84)	76.35(10.93)	77.87(10.88)	49.93(12.30)	54.65(14.66)	57.65(14.36)	58.79(14.77)	60.14(15.14)	61.92(15.30)
Temporal Ens.	59.22 (9.02)	69.95 (9.07)	73.80 (9.78)	77.15 (9.57)	79.80 (9.53)	83.83 (8.73)	52.76(13.15)	59.76(14.48)	63.00(14.35)	65.26(14.07)	65.92(13.71)	67.25(13.63)
Mean Teacher	53.97 (8.24)	62.75 (9.98)	66.42 (9.46)	69.90(11.32)	71.48 (8.98)	77.09 (9.66)	47.03(11.84)	51.56(12.35)	55.05(13.27)	56.60(12.91)	56.66(11.78)	57.97(13.00)
Conv. AutoEnc.	71.39(12.20)	80.03(11.69)	82.86(10.89)	84.74 (9.70)	85.46 (9.77)	87.34 (8.96)	53.19(18.58)	59.52(18.13)	63.01(16.74)	64.83(15.75)	66.40(17.26)	65.96(16.62)
Pseudo-Label	68.02(13.20)	78.11(12.02)	79.57(10.78)	82.21(11.03)	84.11 (9.79)	85.32 (9.38)	52.31(17.93)	58.08(16.76)	60.36(17.92)	60.84(17.59)	62.13(19.03)	62.71(18.36)
MixMatch	68.97(13.93)	80.89(12.80)	**83.94(10.30)**	85.46 (9.64)	85.84 (9.24)	86.88 (8.78)	56.08(15.92)	65.03(15.79)	**69.42(16.31)**	**70.92(16.02)**	**72.31(16.27)**	**73.20(15.19)**
FixMatch	66.36(13.84)	76.26(11.56)	79.04(10.68)	81.79(10.56)	83.14 (9.98)	84.44 (9.09)	53.37(17.33)	63.57(15.57)	63.43(16.26)	64.62(15.57)	66.50(15.91)	68.54(15.58)
AdaMatch	**74.03(11.78)**	**82.59(10.26)**	83.62(10.84)	**85.84 (9.69)**	**86.71 (9.09)**	**88.02 (8.80)**	**58.30(15.95)**	**66.52(16.58)**	69.12(16.45)	68.11(15.80)	69.31(16.87)	71.43(16.04)
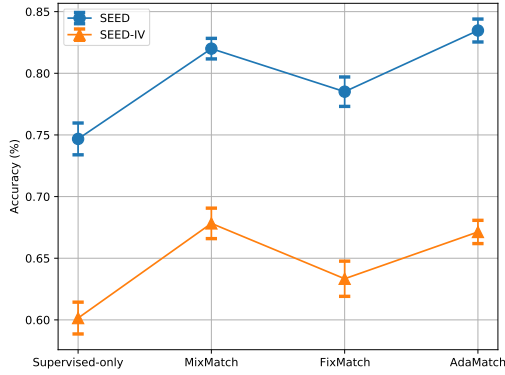
Fig. 2. Average performance of holistic SSL methods in comparison to a supervised method.

of the hyper-parameters has been done on the validation set D_v (with no leakage with the test set). To set the pre-defined threshold (τ) in FixMatch and AdaMatch, we perform a grid search in $[0.0 - 1.0]$ with a step size of 0.1. In FixMatch, we set $\tau = 0.9$ for both datasets, while in AdaMatch, we set $\tau = 0.6$ and $\tau = 0.5$ for SEED and SEED-IV, respectively. Our experiments were carried out on two NVIDIA GeForce RTX 2080 Ti and six Tesla P100 GPUs using PyTorch [21].

3.5. Classical SSL Settings

For all the classical SSL baselines, we use the same convolutional module used as the backbone for the holistic methods (see Section 3.4). For the decoder component of the convolutional autoencoder, we use two transposed convolutional 1-D blocks. In each block, a 1-D transposed convolutional layer is followed by a 1-D batch normalization layer and a ReLU activation. We implement the Π-model, temporal ensembling, mean teacher, pseudo-labeling, and convolutional autoencoder with the same algorithm settings (e.g., loss function, unsupervised weight, etc.) as used in [17].

3.6. Performance

We conduct experiments using all the SSL methods where a few training samples per class are labeled ($m \in \{1, 3, 5, 7, 10, 25\}$), as done in [1, 2, 3]. We repeat each experiment **five** times, each time using a different random seed for the selection of the subsets (D_l, D_u, and D_v). Table 1 displays

the average and standard deviations of accuracies for SEED and SEED-IV datasets. Our results show that in the majority scenarios where very few labeled samples are used, the holistic SSL approaches not only achieve the best results (shown in bold), but also obtain the second-best performances (shown with underline). In SEED, AdaMatch achieves the best performance, while MixMatch achieves the second-best results. AdaMatch performs the best in SEED-IV when very few labeled samples are available ($m \in \{1, 3\}$), while MixMatch performs the best when more samples are labeled ($m \in \{5, 7, 10, 25\}$). AdaMatch and MixMatch consistently rank first and second in most scenarios for both datasets. Among the classical SSL methods, the Convolutional Autoencoder consistently achieves the best results for both datasets.

We compare the performance of the holistic approaches to the supervised-only method, where the results are averaged across all six scenarios where few labeled samples are present. The supervised-only method employs the same backbone network (Section 3.4) but are *only* trained on the labeled samples. As shown in Figure 2, AdaMatch achieves the best performance with $83.47 \pm 10.07\%$ for SEED, while MixMatch achieves the best result with $67.83 \pm 15.92\%$ for SEED-IV. AdaMatch and MixMatch outperform FixMatch in both datasets. In comparison, the supervised-only model achieves accuracies of $74.68 \pm 11.32\%$ and $60.15 \pm 16.28\%$ for SEED and SEED-IV, respectively.

4. CONCLUSION

In this research, we adapt and implement three state-of-the-art holistic semi-supervised learning approaches originally proposed for computer vision [1, 2, 3], for EEG representation learning. We conduct extensive experiments with the holistic SSL approaches and five additional well-known classical SSL methods, where only $1, 3, 5, 7, 10,$ and 25 samples per class are labeled. We evaluate all the SSL methods on two large publicly available datasets, namely SEED and SEED-IV. The holistic approaches achieve the best results and the second-best results in the majority of scenarios for both datasets. The results also show that the holistic approaches remarkably outperform the supervised-only method, addressing the challenge of scarcity of labeled EEG data.

5. REFERENCES

[1] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel, "Mixmatch: A holistic approach to semi-supervised learning," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32, 2019.

[2] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li, "Fixmatch: Simplifying semi-supervised learning with consistency and confidence," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020.

[3] David Berthelot, Rebecca Roelofs, Kihyuk Sohn, Nicholas Carlini, and Alex Kurakin, "Adamatch: A unified approach to semi-supervised learning and domain adaptation," *arXiv preprint arXiv:2106.04732*, 2021.

[4] Rosalind W Picard, *Affective Computing*, MIT press, 2000.

[5] Wei-Long Zheng and Bao-Liang Lu, "Investigating critical frequency bands and channels for eeg-based emotion recognition with deep neural networks," *IEEE Transactions on Autonomous Mental Development*, vol. 7, no. 3, pp. 162–175, 2015.

[6] Wei-Long Zheng, Wei Liu, Yifei Lu, Bao-Liang Lu, and Andrzej Cichocki, "Emotionmeter: A multimodal framework for recognizing human emotions," *IEEE Transactions on Cybernetics*, vol. 49, no. 3, pp. 1110–1122, 2018.

[7] Guangyi Zhang and Ali Etemad, "Capsule attention for multimodal eeg-eog representation learning with application to driver vigilance estimation," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2021.

[8] Guangyi Zhang and Ali Etemad, "Rfnet: Riemannian fusion network for eeg-based brain-computer interfaces," *arXiv preprint arXiv:2008.08633*, 2020.

[9] Wei-Long Zheng, Jia-Yi Zhu, and Bao-Liang Lu, "Identifying stable patterns over time for emotion recognition from eeg," *IEEE Transactions on Affective Computing*, 2017.

[10] Tong Zhang, Wenming Zheng, Zhen Cui, Yuan Zong, and Yang Li, "Spatial-temporal recurrent neural network for emotion recognition," *IEEE Transactions on Cybernetics*, , no. 99, pp. 1–9, 2018.

[11] Juan Abdon Miranda Correa, Mojtaba Khomami Abadi, Niculae Sebe, and Ioannis Patras, "Amigos: A dataset for affect, personality and mood research on individuals and groups," *IEEE Transactions on Affective Computing*, 2018.

[12] Jesper E Van Engelen and Holger H Hoos, "A survey on semi-supervised learning," *Machine Learning*, vol. 109, no. 2, pp. 373–440, 2020.

[13] Dong-Hyun Lee et al., "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks," in *Workshop on challenges in representation learning, ICML*, 2013, vol. 3, p. 896.

[14] Laine Samuli and Aila Timo, "Temporal ensembling for semi-supervised learning," in *International Conference on Learning Representations (ICLR)*, 2017, vol. 4, p. 6.

[15] Antti Tarvainen and Harri Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.

[16] Avital Oliver, Augustus Odena, Colin A Raffel, Ekin Dogus Cubuk, and Ian Goodfellow, "Realistic evaluation of deep semi-supervised learning algorithms," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 31, pp. 3235–3246, 2018.

[17] Guangyi Zhang and Ali Etemad, "Deep recurrent semi-supervised eeg representation learning for emotion recognition," in *2021 9th International Conference on Affective Computing and Intelligent Interaction (ACII)*. IEEE, 2021, pp. 1–8.

[18] Li Tong, Hang Wu, and May D Wang, "Caesnet: Convolutional autoencoder based semi-supervised network for improving multiclass classification of endomicroscopic images," *Journal of the American Medical Informatics Association*, vol. 26, no. 11, pp. 1286–1296, 2019.

[19] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz, "mixup: Beyond empirical risk minimization," in *International Conference on Learning Representations (ICML)*, 2018.

[20] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[21] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al., "Pytorch: An imperative style, high-performance deep learning library," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32, pp. 8026–8037, 2019.