2-21-2023 10:00 AM

Attention-based Multi-Source-Free Domain Adaptation for EEG Emotion Recognition

Amir Hesam Salimnia, *The University of Western Ontario*

Supervisor: Boyu Wang, *The University of Western Ontario*

A thesis submitted in partial fulfillment of the requirements for the Master of Science degree in Computer Science

Abstract

Electroencephalography (EEG) based emotion recognition in affective brain-computer interfaces has advanced significantly in recent years. Unsupervised domain adaptation (UDA) methods have been successfully used to mitigate the need for large amounts of training data, which is required due to the inter-subject variability of EEG signals. Typical UDA solutions require access to raw source data to leverage the knowledge learned from the labelled source domains (previous subjects) across the target domain (a new subject), raising privacy concerns. To tackle this issue, we propose Attention-based Multi-Source-Free Domain Adaptation (AMFDA) for EEG emotion recognition. AMFDA attempts to transfer knowledge of source models to the target domain by aggregating adapted source models based on a set of learnable weights without accessing the source data. While the classifiers of source models are frozen, the set of learnable weights and the feature extractors are learned based on information maximization and a novel self-supervised pseudo-labelling method. A channel-wise attention layer is also used in the proposed framework to enhance the performance of source models, which in turn improves the performance of target models. We conducted extensive experiments on SEED and SEED-IV. The experimental results demonstrate that the proposed AMFDA method performs comparably to UDA state-of-the-art methods.

Keywords: Electroencephalogram (EEG), Emotion Recognition, Affective Computing, Brain-Computer Interface, Deep Learning, Self-supervised Learning, Domain Adaptation (DA)

Summary for Lay Audience

It is important to understand emotions as they are a fundamental part of human communication and behaviour. Thus, it is crucial to understand how emotions can be interpreted through physiological signals in human-computer interaction. In general, physiological signals, such as EEG, can be highly affected by the psychological and physical characteristics of individuals, thereby necessitating the collection of large amounts of data. Additionally, EEG signals contain extensive private information that can be used to identify individuals.

To protect the privacy concerns of subjects and mitigate the need for large datasets, we introduce a novel approach to recognizing emotions based on EEG. The proposed method involves transferring knowledge from previous subjects (source domains) to a new one (target domain). The results of our research have demonstrated that our proposed method performs as well as those methods that require data from source domains, while also maintaining the privacy of participants by not utilizing the information of previous participants.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

As technology becomes more advanced and widely used, it must be designed in a way that is intuitive and easy for people to operate. The study of how people interact with computers and other technology, known as Human-Computer Interaction (HCI), ensures that technology is easy to use and can be adopted by a wide range of people, regardless of their abilities, backgrounds, or expertise levels. As a common mental phenomenon that has a vital role in human behaviour, action, decision making and communication, emotion analysis has been the subject of many studies. Even though emotion recognition has other applications beyond the medical field, it can have significant medical benefits. Using emotion recognition technology, doctors and nurses could monitor patients' health more effectively and provide them with more personalized and effective care based on their emotional state. Moreover, it assists in diagnosing mental health conditions early, such as autism spectrum disorders, depression [86, 9], and anxiety, allowing patients to receive more appropriate treatment. As a result of emotion recognition technology, patients with conditions such as autism or dementia could better understand and interpret others' emotions, thereby improving their social skills and overall quality of life.

Emotional states are individuals' subjective feelings in response to internal or external stimuli, and they can range in intensity, duration, and valence, encompassing both positive and negative emotions such as happiness, sadness, anger, fear, excitement, and contentment. These emotional states have the potential to affect various cognitive processes, including attention, memory, and decision-making. Positive emotional states have been found to enhance creativity and problem-solving abilities, while negative emotional states have been shown to impair cognitive function and increase the likelihood of impulsive decision-making, according to research. Emotions can be expressed primarily through internal (biological) and external (non-biological) responses. The typical external responses of humans include facial expressions, gestures or speech, while internal responses can be heart rate, blood pressure, respiration rate, electroencephalogram (EEG), and magnetoencephalogram (MEG). Research has traditionally classified emotions using non-physiological indicators like factual expressions[64, 32]. However, the neuroscience perspective indicates that several major brain cortex regions are closely related to emotions, such as the orbital frontal cortex, the ventral medial prefrontal cortex, and the amygdala [8, 21, 53, 47]. Using an electroencephalogram (EEG) to record brain activity is a popular method as it is non-invasive and relatively inexpensive. Unlike other brain imaging techniques, such as functional MRI, that require extensive and expensive equipment, EEG utilizes a simple and portable device that can easily be worn on the head. Further, EEG allows for

real-time monitoring of brain activity and is highly sensitive to changes. Hence, researchers can develop algorithms to detect and interpret emotions based on the brain activity associated with them.

Several studies have used manual feature extraction in order to solve time-series problems, such as emotion recognition using EEG data. These methods are characterized by their simplicity and low computational costs.

In contrast to traditional methods, machine learning approaches offer several advantages for EEG-based emotion recognition. One of the primary advantages is that machine learning methods automate the process of feature extraction and classification, eliminating the need for manual feature engineering. This is particularly critical for EEG data, which can be complex and difficult to interpret using traditional techniques. Another advantage of machine learning methods is that they are able to handle complex relationships among variables and extract intricate patterns, which is important for accurately capturing the nuances of emotional states. Additionally, machine learning techniques are more scalable than traditional time-series solutions, allowing for easy integration of additional training data without the need to manually adjust thresholds or other parameters. Finally, machine learning algorithms can be very accurate at classification tasks, processing large amounts of data efficiently using multiple GPUs and CPUs, which is important for achieving high levels of accuracy in EEG-based emotion recognition.

Many fields, including computer vision, natural language processing, and biomedical signal processing, have demonstrated that deep learning outperforms traditional machine learning. As a result, many deep learning-based methods for recognizing emotions using EEG have also been widely applied. However, deep learning methods for emotion recognition based on EEG signals face two main challenges. Firstly, deep learning models require a large amount of data for training, as they have numerous variables that can be learned. Without sufficient labelled EEG data, the models could end up being overfitted. Secondly, traditional machine learning algorithms rely on the assumption that training and test data are independent and symmetrical, which cannot be met using EEG signals. Several factors determine EEG signals, including the subject's mental state, electrode impedance, head shape, etc. Additionally, EEG signals acquired from a single participant in different sessions may also be very biased, resulting in substantial challenges in developing a practical EEG-based emotion recognition model.

Researchers have developed practical algorithms based on unsupervised domain adaptation (UDA) to overcome these obstacles. Referring to the existing labelled subjects as source domains, and the unlabelled new data as target domain, UDA methods aim to eliminate the differences in the distribution of EEG data between the source domain and target domain. Minimizing the domain shift between domains allows the model to extract transferable features for emotion recognition.

1.1 Motivation

The existing DA methods have demonstrated impressive performance in classifying emotions based on EEG data. Nevertheless, these approaches overlook concerns about data privacy and commercial copyright. There have been many pre-trained deep learning models released, but their training data may not be accessible for adapting them to a novel domain. Their

reasons for not sharing training data may be privacy or copyright concerns. Specifically, EEG signals contain personal information, including personal preferences, physical characteristics, and emotional states, many of which are protected under data protection laws.

Instead of using source data for adaptation, we propose to use only the pre-trained source model directly, which decreases the risk of sharing and storing private information. In addition, given the discrepancies between the EEG data of different subjects, each subject is treated as a separate source domain. As a result, the proposed method adapts prior knowledge from various source models to the target model. The algorithm can be applied to a wide range of setups since each source model is independent of the data and structure of others.

1.2 Contributions

There are many different ways to approach the task of EEG-based emotion recognition. One promising approach involves using unsupervised domain adaptation methods. These methods are particularly effective because they are able to account for the inherent variability in EEG data between subjects [46]. Despite the impressive performance of state-of-the-art domain adaptation methods for emotion recognition, all techniques require access to source data during the adaptation phase. Therefore, due to the high level of sensitivity of medical data such as EEG, previously proposed state-of-the-art methods are facing privacy concerns. So, these methods are limited in their scope of use since many clinics do not allow the sharing of their subjects' data. Thus, these methods would be restricted to public datasets, which are substantially smaller amounts of data than private data. Focusing on the unsupervised domain adaptation methods, to fully protect subjects' data privacy by prohibiting access to the datapoints, we propose the Attention-based Multi-Source-Free Domain Adaptation for EEG Emotion Recognition (AMFDA) method. It is a pioneering attempt to capture transferable information from multiple subject models to promote target prediction without access to source data. Based on the similarity between source domains and target domain, we introduce a self-supervised pseudo-labelling method for unlabeled target data. To improve the robustness of the target model in emotion classification, a contrastive learning approach was used in the adaptation phase, relying on a random channel-weakening augmentation method. Additionally, we developed an attention module that improves emotion classification by extracting channel-wise features from EEG data.

Following is a summary of our main contributions:

- Developing an Attention-based Multi-Source-Free Domain Adaptation for EEG Emotion Recognition (AMFDA) model to transfer knowledge from multiple source domains to enhance target prediction without access to source data, only by using pre-trained source models. To the best of our knowledge, this is the first paper to explore multi-source-free domain adaptation in the field of EEG-based emotion recognition
- Evaluating the performance implications of contrastive learning with channel-wise augmented data in source-free adaptation for EEG-based Emotion Recognition
- Enhancing the performance of the emotion classification model by applying a novel channel-wise attention module to extract the frequency representation of EEG signals.

- Performing extensive experiments and comparing our method to several domain adaptation baselines. This study also shows that our approach preserves subjects' privacy while obtaining comparable results to state-of-the-art UDA methods
- Conducting a detailed ablation study to assess the impact of each component and the effectiveness of using learnable weights to aggregate source models.

1.3 Thesis Outline

The remainder of this thesis is organized as follows: Section 2 discusses related works on domain adaptation in general and EEG-based emotion recognition in particular. Section 3 presents the general framework and the mathematical description of our method. In Section 4, the experiments and results are presented and analyzed, including an ablation study that indicates the impact of each component. Finally, the discussion and conclusions of this paper are given in section 5.

Chapter 2

Backgrouand and Related Work

2.1 Background

2.1.1 Learning Algorithms

There are many different learning algorithms used in machine learning to solve a wide range of problems. Based on the type of data and the goal of algorithms, learning algorithms can be classified as Supervised Learning, Unsupervised Learning, Semi-Supervised Learning, Reinforcement Learning, and Transfer Learning.

Supervised Learning

Supervised learning algorithms are a type of machine learning algorithm that learns from labelled training data where the correct output is provided for each example in the training set. These algorithms build a model that maps the input data to the corresponding output labels, and can then use the trained model to predict labels for unseen data. As its name implies, this type of algorithm is supervised in the sense that it leverages supervision by comparing current predictions with the grand truth and correcting itself as needed. It is common to use linear regression, logistic regression, decision trees, support vector machines (SVMs), and neural networks as supervised learning algorithms.

Some of the applications of supervised learning algorithms include image classification, speech recognition, natural language processing, and predictive modelling. These algorithms are commonly used in situations where there is a large amount of labelled training data available and the goal is to make accurate predictions.

Unsupervised Learning

There are many real-world problems in which the grand truth is not available or requires extensive annotations. In these cases, unsupervised learning algorithms can be used to learn based on unlabeled data. Unlike supervised learning algorithms, which learn by being given the correct output for each example in the training set, unsupervised learning algorithms must find their own way to make sense of the data. The goal of unsupervised learning is to find patterns

or structures in the data by grouping similar examples together. There are a variety of applications for unsupervised learning, including clustering (e.g. K-Means), dimension reduction (e.g. Principal Component Analysis), density estimation, and anomaly detection.

Similarly to unsupervised learning methods which do not require a labelled dataset, Self-Supervised Learning (SSL) is a machine learning approach in which the model trains itself by leveraging one part of the data to predict the other part and generate labels accurately. Ultimately, this method turns an unsupervised learning problem into a supervised one. Self-supervised learning methods have demonstrated remarkable abilities for solving complex problems, in different areas. SSL has been used everywhere from app documentation generation to sentence completion and text suggestions in the Natural Language Processing field.

Semi-Supervised Learning

Semi-supervised learning can be considered a middle ground between Supervised and Unsupervised machine learning. This is different from supervised learning, which employs only labelled data, and unsupervised learning, which employs only unlabelled data. Semi-supervised learning can be useful when there is a large amount of unlabeled data available, but only a small amount of labelled data. By using both labelled and unlabeled data, semi-supervised learning can often achieve better performance than either supervised or unsupervised learning alone. The model uses the labelled data to learn the structure of the data and the unlabeled data to improve its accuracy and generalization performance.

Reinforcement Learning

As a type of machine learning technique, reinforcement learning trains an agent to make decisions in complex and uncertain environments. Unlike supervised learning, where the answer key is available, reinforcement learning rewards positive actions and penalties negative actions. The agent interacts with its environment by taking action and observing the resulting rewards and state changes. Over time, through trial and error or by using algorithms that learn from previous experiences, the agent improves its decision-making abilities to determine the optimal policy, or sequence of actions, that maximizes the total rewards it receives.

Reinforcement learning has been applied to a wide range of problems, including robot control, game playing, and natural language processing. It has the potential to enable agents to learn complex behaviours and adapt to changing environments, making it a powerful tool for solving challenging real-world problems.

Transfer Learning

Transfer learning can be a powerful tool in machine learning, especially when dealing with limited data or computational resources. In transfer learning, a model trained on one task is used as the starting point for a model on a related task. This allows the model to transfer knowledge and features learned from the first task to the second task, reducing the amount of data and computational resources needed to train the second model.

2.1.2 Neural Network

A neural network (NN) or Artificial Neural Network is a computational learning system aiming to map inputs to desired outputs. Inspired by the human brain, NNs were originally introduced to mimic biological neural networks. It is composed of many interconnected processing nodes, called neurons, which work together to process information and solve complex computational problems. The structure of a neural network allows it to learn from and make predictions, making it a powerful tool for many different applications.

Perceptron

Developed by Frank Rosenblatt in 1958, a perceptron is a single-layer neural network that has a single neuron that receives inputs from multiple sources. As it is shown in Figure 2.1, after multiplying all input values and their weights and adding them, the weighted sum will be fed into a non-linear activation function to obtain the desired output. The weights of a perceptron are adjusted during the training phase, in which the perceptron is provided with the correct binary output corresponding to each input data.

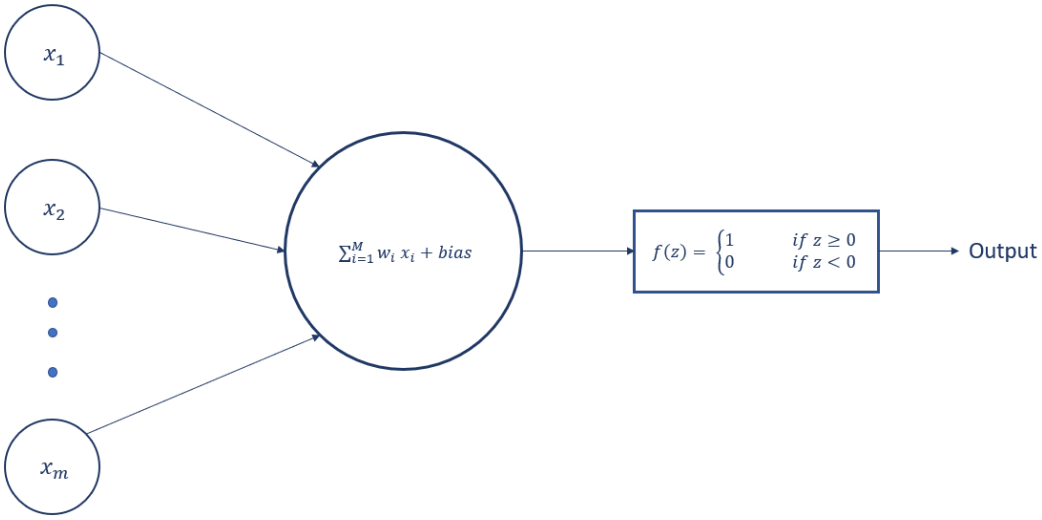


Figure 2.1: Perceptron Architecture

In spite of the fact that perceptions are capable of representing simple mappings such as ”AND” and ”OR”, they are unable to find the mapping of more complex functions such as exclusive OR (XOR).

Multi-layer Perceptron

To mitigate the incapacity of a perceptron to represent non-linear functions, a Multi-layer Perceptron (MLP) was developed. A multi-layer perception is composed of an input layer, one or more hidden layers, and an output layer, as shown in Figure 2.2. MLP is a feedforward algorithm, which means that the output of each layer is propagated to the next layer, and there is no feedback to the previous layer or to the current layer.

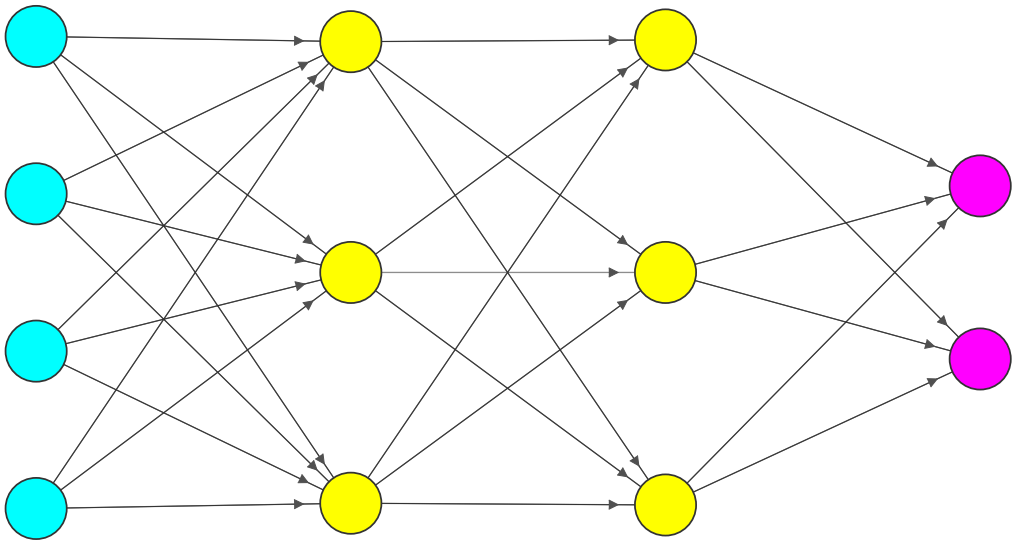


Figure 2.2: Multi-Layer Perceptron Architecture

To exploit the capacity of MLPs, Backpropagation was developed to update the weights of nodes. It allows MLPs to iteratively adjust the weights in the network, aiming to minimize the cost function. The backpropagation method works by propagating the error from the output layer of the network backwards through the hidden layers and calculating the gradient of the loss function with respect to each weight in the network. This is done using the chain rule from calculus, which allows the gradient to be calculated by breaking it down into the product of the partial derivatives of each layer.

2.1.3 Activation Functions

As discussed in Section 2.1.2, an activation function is a crucial component of a neural network. A major reason for this is that the activation function introduces non-linearity into each neuron. In other words, if the neurons do not have any activation function (f(x) = x), we can simplify all the layers into a single layer, since the combination of linear functions is a linear function. Moreover, activation functions normalize the output of each node, which helps to stabilize the learning process and improve the overall performance of the network.

There are many different types of activation functions, and each has its own unique properties and characteristics. However, to be able to use backpropagation techniques to train NN, we need to use differentiable functions. Some of the most common activation functions include the Logistic function, the Hyperbolic Tangent function, the Rectified Linear Unit (ReLU) function, and Leaky Rectified Linear Unit (LeakyReLU). The choice of activation function can have a significant impact on the performance of a neural network, and certain activation functions are better suited to different types of problems.

Logistic Function

The logistic function has an ”S” shaped curve, which is why it is also called Sigmoid function. It takes a real-valued input and squashes it into a range between 0 and 1, so it is often used as the activation function for the output layer of a neural network when the network is being used for binary classification tasks. The mathematical equation for the sigmoid function is:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{2.1}$$

The sigmoid function has a number of useful properties. It is differentiable, which means that the gradient of the function can be calculated, which is necessary for training the network using backpropagation. It also has a smooth transition between the output of 0 and 1, which allows the network to make smooth predictions.

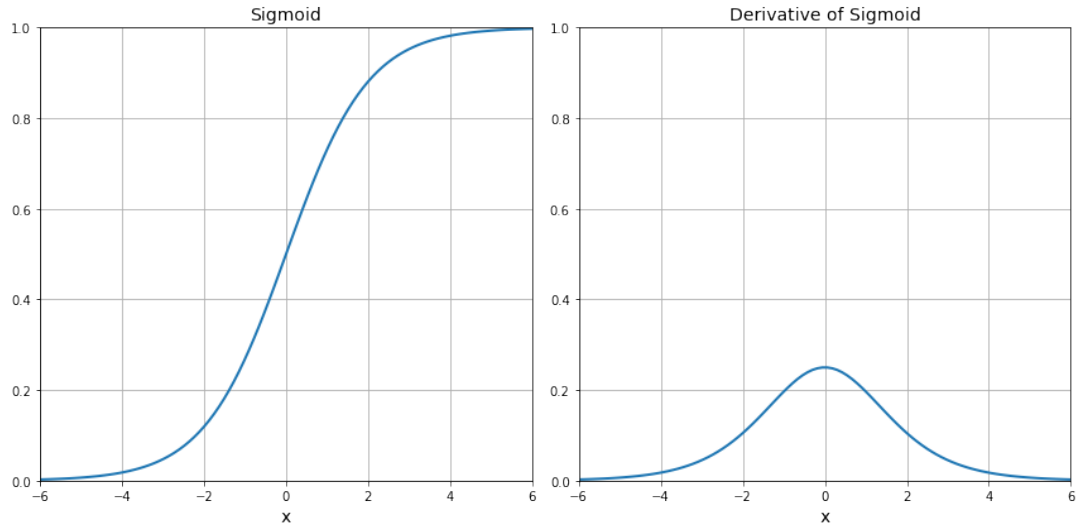


Figure 2.3: The Sigmoid Activation Function and its derivative

As it is shown in Figure 2.3, the logistic activation function suffers from the vanishing gradient problem, which can make training difficult in some cases. As a result, other activation functions such as the ReLU function are often used in place of the sigmoid function.

Hyperbolic Tangent

The hyperbolic tangent activation function, also known as the tanh activation function, is a type of non-linear activation function that is often used in neural networks. The mathematical equation for the tanh function is:

$$\tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{2.2}$$

An important property of the tanh function is its smooth transition from outputs of -1 to 1, which allows the network to predict predictions smoothly.

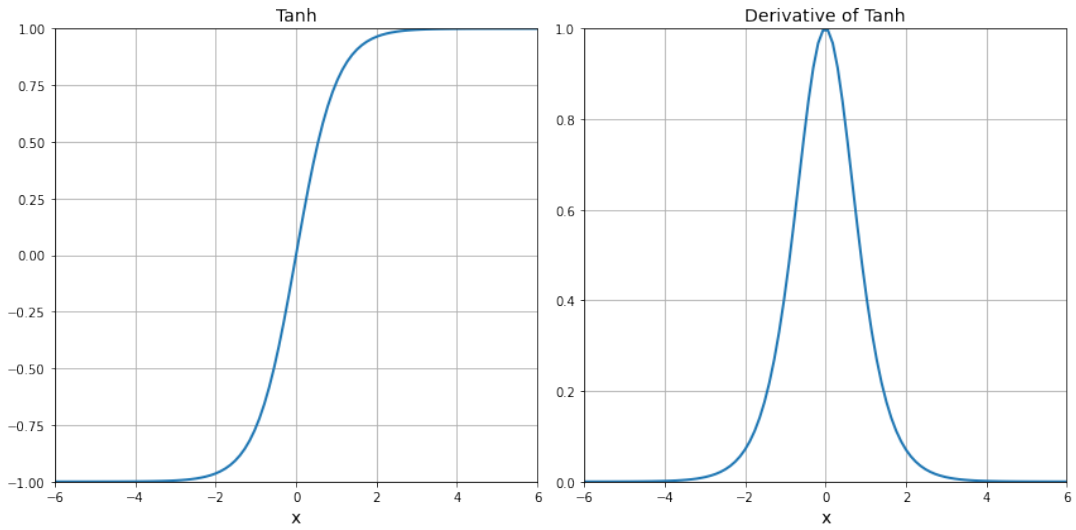


Figure 2.4: The tanh Activation Function and its derivative

According to Figure 2.4, the hyperbolic tangent has a broader output spectrum compared to the logistic function, which can improve the convergence of the backpropagation algorithm. The tanh function is often used as the activation function for the hidden layers of a neural network. However, it can also be used in the output layer for regression and multi-class classification tasks. Like the sigmoid function, it can suffer from the vanishing gradient problem, which can make training difficult in some cases.

Rectified Linear Unit

ReLU is the most widely used activation function in neural networks because it is computationally efficient, which makes it a popular choice for deep learning networks. The mathematical definition of the ReLU is as follows:

$$ReLU(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{2.3}$$

Unlike logistic functions and tanh, ReLU does not suffer from vanishing gradients when outputs are positive (Figure 2.5). ReLU's output range is from 0 to infinity, so it cannot normalize nodes' outputs. So, it is mostly used in the hidden layer rather than the output layer. However, it can suffer from the problem of the "dying ReLU" problem, where some neurons are updated to always output 0, which can make training difficult.

Leaky Rectified Linear Unit

The Leaky ReLU is an extension of the rectified linear unit (ReLU) activation function, designed to alleviate the Dying ReLU issue. The Leaky ReLU function is similar to the regular ReLU function, but it allows for a small, non-zero gradient when the input value is negative. As can be seen in Figure 2.6, the Leaky ReLU has a positive slope in the negative area which is determined before training the model.

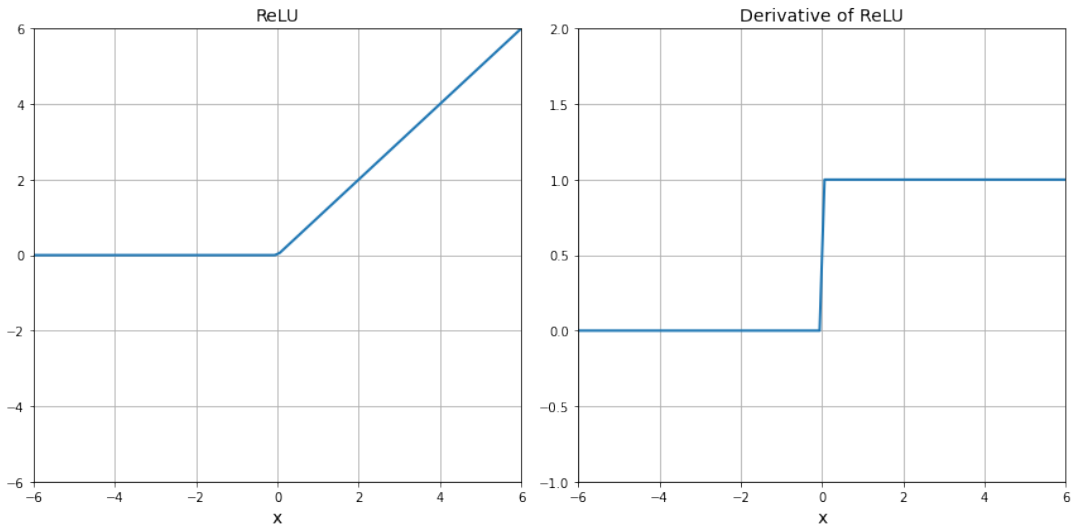


Figure 2.5: The ReLU Activation Function and its derivative

$$LeakyReLU(x) = \begin{cases} x, & x \geq 0 \\ \alpha x, & x < 0 \end{cases} \tag{2.4}$$

where the slope coefficient, alpha is typically set between 0.01 and 0.1. It has been shown to outperform the regular ReLU function in some cases, and it is a popular choice for many different types of neural networks.

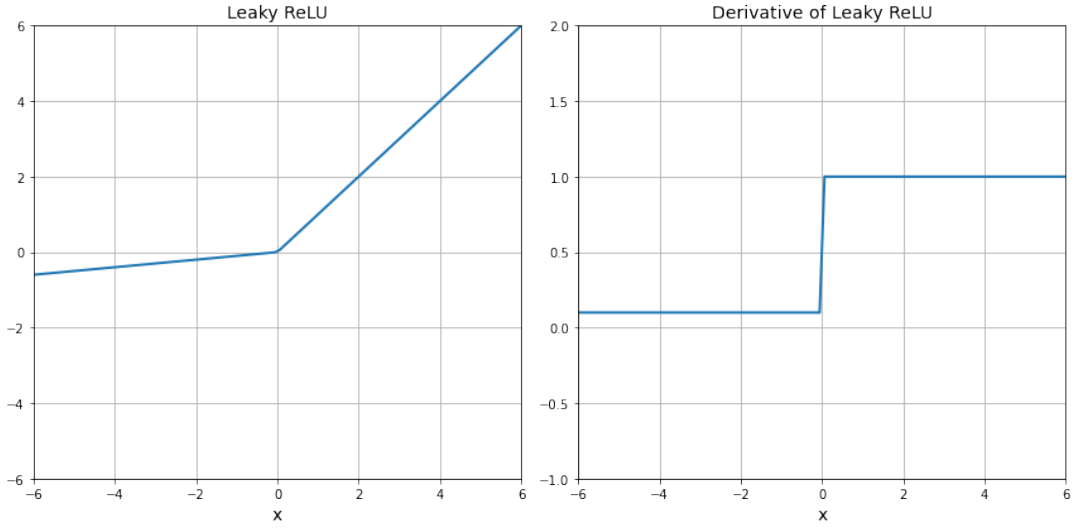


Figure 2.6: The Leaky ReLU Activation Function and its derivative

2.1.4 Loss Functions

A loss function is a criterion showing how well our model works. In other words, it is the function that computes the distance between the current output and the output that we expect to have. This distance will then be used as feedback to update the parameters of the model. Cross-entropy loss and Kullback-Leibler divergence loss will be discussed in the following sections.

Cross-Entropy Loss

Inspired by information theory, the cross-entropy loss has been used to assess the performance of a classification model whose output is a probability value between 0 and 1. It is a measure of the difference between two probability distributions, p and q, for a given random variable or set of events. The value of cross-entropy can be calculated as follows:

$$\mathcal{L}(p, q) = -\sum_{x \in \mathcal{X}} p(x) \ \log q(x) \tag{2.5}$$

Cross-entropy loss increases as the predicted probability diverge from the actual label. It would be heavily penalized if a model predicted a probability of 0.9 for a class that was actually 0, as it is very confident in a wrong prediction.

Kullback-Leibler Divergence Loss

The Kullback-Leibler divergence (KL divergence) is a measure that quantifies how a probability distribution differs from another probability distribution. Considering the true distribution $P(X)$ and the prediction distribution $Q(X)$, the KL measures the distance from Q to P as follows:

$$D_{KL}(P \parallel Q) = \sum_{x \in \mathcal{X}} P(x) \ \log\left(\frac{P(x)}{Q(x)}\right) \tag{2.6}$$

Based on Eq. 2.6, the KL Divergence cannot be used as a distance metric between two distributions since it is not symmetric ($D_{KL}(P, Q) \neq D_{KL}(P, Q)$).

2.1.5 Optimizers

To minimize (or maximize) the loss function (cost function), which is highly dependent on the model's learnable parameters, we can either try to algebraically derive a closed-form solution or approximate it using an iterative method. The number of learnable parameters of the model could be countless, so finding a closed-form solution is not feasible. In this case, optimizer algorithms are introduced to modify the attributes of the neural network such as weights and learning to reduce the loss function iteratively. In the following sections, we will discuss two of the most popular and powerful optimizers.

Gradient Descent

Gradient Descent (GD) is one of the most popular optimization algorithms used in neural network optimization. It is a first-order iterative algorithm for finding the local minimum of

a differentiable function (loss function). The idea is to update the model's parameters in the opposite direction of the gradient of the loss function w.r.t. to the current parameter values. Theoretically, we know the opposite direction of the gradient of the loss is the direction of the steepest descent. As long as the step size (learning rate) is small enough, the loss function value can be reduced iteratively by moving down the direction, leading to a local minimum.

Mini-Batch Stochastic Gradient Descent

Using gradient descent to approximate the gradient w.r.t. the current parameter values is not feasible when the number of data points is too large. To mitigate the problem, gradient descent variants have been introduced. Instead of computing the gradient of the cost function based on the entire training set or a single example, mini-batch gradient descent uses a small, fixed-sized subset of the training data (called a mini-batch) to compute the gradient. While choosing the best batch size might be tricky, overall mini-batch gradient descent is more efficient than all gradient descent variants.

Adaptive Moment Estimation

Adam[37] or Adaptive Moment Estimation is an extension of stochastic gradient descent that uses an adaptive learning rate instead of a constant one. The Adam optimizer computes the exponential moving average of the gradient and the squared gradient to estimate the first and second moments of the gradient, respectively. It then uses these estimates to compute adaptive learning rates for each parameter, which are used to update the model's parameters in the direction that minimizes the cost function. The Adam optimizer uses adaptive learning rates to make more efficient updates to the model's parameters, which often allows it to converge to a minimum of the cost function faster than other optimization algorithms.

2.1.6 Emotional States

Emotional state refers to the subjective experience of an individual's emotions at a particular moment in time. It includes the intensity, valence, and duration of emotions, as well as the cognitive and physiological responses that accompany them. Emotional states can be positive, such as happiness or excitement, or negative, such as sadness or anger.

Quantifying emotional states involves measuring and analyzing the different components of emotions. One approach is to use self-report questionnaires, where individuals rate their emotional state on a scale, such as the Positive and Negative Affect Schedule (PANAS) [89] or the Profile of Mood States (POMS) [63]. It is also possible to detect emotional states by analyzing changes in facial muscle movements and expressions. An alternative approach is to measure changes in the autonomic nervous system associated with emotional states using physiological measures, such as electroencephalogram (EEG), electrocardiogram (ECG), or skin conductance. Based on patterns of brain and physiological activity, machine learning algorithms can be applied to analyze physiological data and classify emotional states.

2.1.7 EEG

EEG stands for Electroencephalogram, a non-invasive technique used to measure the brain's electrical activity. To detect and record the electrical signals generated by the brain's neurons, electrodes are placed on the scalp (Figure 2.7). An EEG signal can provide valuable information about brain functioning, including patterns of neural activity associated with various cognitive and emotional processes. EEG has many applications in neuroscience, clinical neurology, and psychology. Neuroscience research uses EEG to study brain function during different tasks, including perception, attention, and memory. A variety of neurological disorders, such as epilepsy, and brain function after brain injury is diagnosed and monitored with an EEG in clinical settings. In psychology research, EEG can also be used to study the neural correlates of emotions, personality characteristics, and other psychological processes. EEG data can also be analyzed by machine learning algorithms to identify patterns of brain activity related to cognitive and emotional states.

EEG signals are mixed with various noises from the human body and environment, resulting in challenges to the robustness of the recognition algorithm. Consequently, the collected EEG signals are not directly used to build recognition models and systems. By using Time-frequency analysis techniques, it is possible to obtain a more detailed and accurate representation of the EEG signal and extract features that are relevant for different EEG applications such as sleep analysis, seizure detection, and brain-computer interface. As an example of time-frequency analysis methods, Short-Time Fourier Transform (STFT) [56] calculates the Fourier transform of a signal in short overlapping time windows, and produces a series of spectral estimates that show how the frequency content of the signal changes over time. The STFT is computed by dividing the signal into short segments, typically using a sliding window function, and then applying the Fourier transform to each segment, which can be calculated as follows:

$$X(t, f) = \int_{-\infty}^{\infty} w(m - t)\, x(m) e^{-j2\pi f m} dm \tag{2.7}$$

where $w(m-t)$ represents the short-time analysis window. In essence, STFT extracts several frames of the signal to be analyzed by using a window that moves with time. Moreover, in section 4.1 and 4.2, we will introduce more about our dataset and our preprocessing method based on STFT.

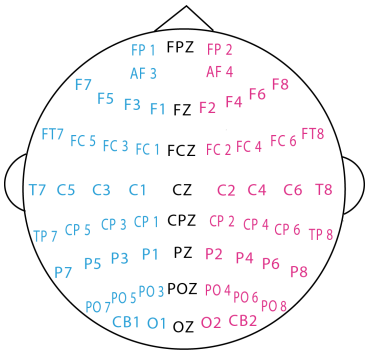


Figure 2.7: Positions of scalp electrodes in the EEG cap.

2.2 Related Work

2.2.1 Domain Adaption

The high performance of most existing machine learning models relies heavily on using massive amounts of annotated training data. Because of the limited number of labels and the time and money involved in manually annotating data, real-world applications cannot meet such a requirement. As a result, it is often necessary to transfer knowledge between domains. Domain shift can occur when knowledge is transferred from one domain to another. This can be caused by a variety of factors, including the evolution of the statistical properties of a domain over time, or the collection of new samples. To overcome this burden, a new research area in machine learning called Domain Adaptation (DA) has been proposed. Domain adaptation methods aim to mitigate the domain shift by minimizing the difference between domain distributions. Specifically, Unsupervised Domain Adaptation (UDA) attempts to reduce the discrepancy between the labelled source data and the unlabeled target data so that domain-invariant representations can be learned in both domains. Unsupervised domain adaptation methods can be divided into two types: traditional methods and deep learning methods[22, 91].

Transfer component adaptation (TCA)[67] is a traditional domain adaptation method that minimizes marginal distributions between source and target domains to learn a domain-invariant feature transformation. Based on the kernel method, Kernel Principal Component Analysis (KPCA) [75] is a nonlinear dimensionality reduction method that maps high-dimensional space to low-dimensional space in a Reproducing Kernel Hilbert Space (RKHS). Transductive Parameter Transfer (TPT) [74] involves learning multiple source subject classifiers and then transferring knowledge about these classifiers to the target individual directly through the use of these classifiers. In order to achieve the transfer process, a regression function is learned which maps the data distribution associated with each source subject to the parameters of the corresponding classifier.

Deep domain adaptation approaches[59, 23], on the other hand, have two main advantages over traditional domain adaptation methods: the capability to extract the generalized feature representation of data, and the ability to satisfy practical end-to-end requirements [22, 91]. Deep domain adaptation can generally be organized into the following categories:

Discrepancy-based

One of the most popular deep methods is discrepancy-based domain adaptation, which minimizes the difference between source and target domains by using statically defined distance functions. The most commonly used distance measures in domain adaptation are maximum mean discrepancy (MMD) [26], Wasserstein metric, correlation alignment (CORAL)[82], Kullback-Leibler (KL) divergence [40], and contrastive domain discrepancy (CDD) [36]. Deep Adaptation Network (DAN) [59] eliminates domain discrepancies across domains as well as preserves task-related features by jointly minimizing Multi-Kernel Maximum Mean Discrepancies (MK-MMDs) and task-related loss. So, it only aligns the marginal distributions and does not consider the conditional distribution disparity across domains. DCORAL[83] proposed incorporating a deep architecture into the CORAL[82] mode which aims to align the second-order statistics.

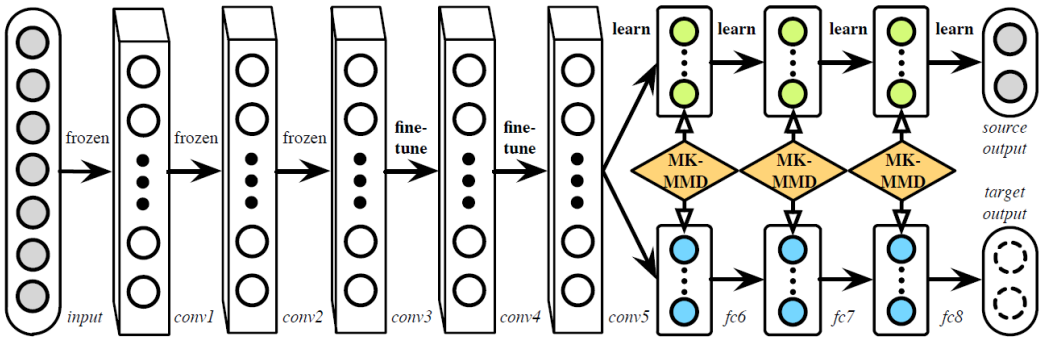


Figure 2.8: The architecture of a Deep Adaptation Network (DAN) model, consisting of frozen convolutional layers to extract general features, convolutional layers needed to be fine-tuned, and fully connected layers (image from [59])

Reconstruction-based

The other category of deep adaptation networks is reconstruction-based domain adaptation, which mostly uses auto-encoders to align discrepancies between domains while preserving domain-specific features. The shared encoder learns a domain invariant and transferable representation across domains, while domain-specific features are preserved by minimizing reconstruction error. [87] introduced Stacked Denoising Autoencoders (SDA), aiming to find common features between both source and target domains via denoising autoencoders. Despite the model's impressive results, it suffers from high computational cost and lack of scalability to high-dimensional features. To mitigate the limitations, Marginal SDA (mSDA) was proposed to denoise the marginal noise with a closed-form solution without using a stochastic gradient descent strategy (SGD) [6]. Deep Reconstruction-Classification Network (DRCN) [24] consist of a convolutional encoder and a deconvolutional decoder network which are trained to be able to both predict source labels as well as reconstruct the target data.

Adversarial-based

Essentially, GANs[25] are deep-learning models that are divided into two submodels: the generator model and the discriminator model. The generator's goal is to produce fake samples that are as similar to the real samples as possible, while the discriminator's goal is to differentiate the real samples from the fake ones. Since they are trained together, the generator must create plausible examples to fool the discriminator, which results in domain-invariant and transferable features. Inspired by GANs, Domain-Adversarial Neural Network (DANN) [23] is proposed in order to integrate domain adaptation into the process of learning representation, so that the final classification decisions can be made based on features that are both discriminative and invariant across domains. As Figure 2.9 depicts, DANN is composed of three sub-networks: a feature extractor, a label predictor that predicts class labels and is used both during training and at test time, and a domain discriminator that discriminates between the source and the target domains during training. While the parameters of the label predictor and the domain discriminator are optimized to minimize their own loss, the parameters of the feature extractor

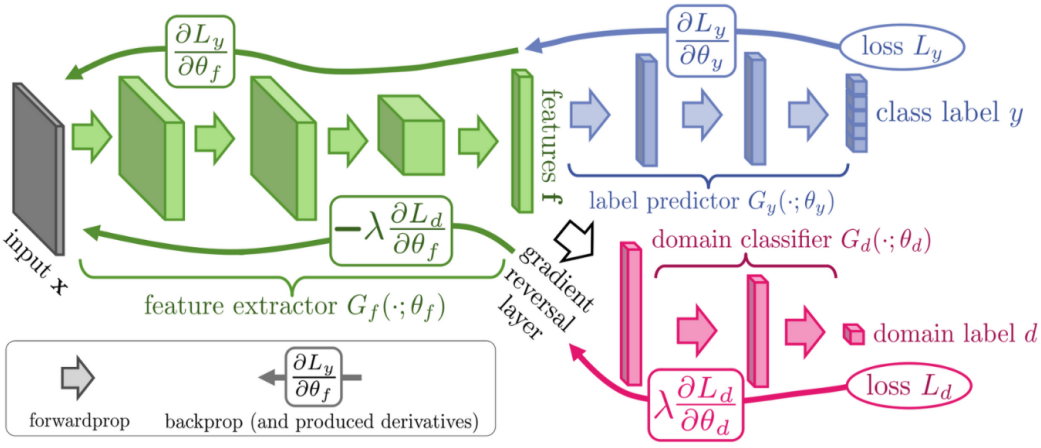


Figure 2.9: The architecture of a domain adversarial neural networks (DANN) model, including a feature extractor (green), a label predictor (blue), and a domain classifier (pink) (image from [23])

are adversarially updated to extract discriminative as well as domain-invariance features.

2.2.2 Multi-source Domain Adaptation

Real-world applications often require the transfer of knowledge from multiple sources to a target domain [98]. The simplest way is to combine all sources into a single source. However, using multi-source data combined as one domain data for transfer learning is ineffective due to the inconsistent data distribution between the target domain and the source domains. As a solution to the mentioned challenges, Multi-source domain adaptation (MSDA) has been developed to extend the standard UDA setting by incorporating knowledge from multiple source models [4, 5]. Based on the assumption that the target distribution can be approximated by a mixture of M source distributions, a weighted combination of source classifiers has been widely employed [62]. There are two common methods for aligning latent spaces: discrepancy-based methods and adversarial methods. For aligning the features of different domains, discrepancy-based methods aim to narrow certain discrepancies across domains, such as maximum mean discrepancy [27, 106], Rényi divergence [30], $\mathcal{L}_2$ distance [70], and moment distance[68]. In addition, adversarial approaches can also align features using a shared domain discriminator. By optimizing H-divergence[96], generative adversarial loss[93], and Wasserstein distance[49, 88, 99], the discriminator helps the model achieve indistinguishable features across multiple domains. Ignoring the fact that each source domain has a different correlation with the target domain would lead to negative transfer, since the target domain might be aligned with dissimilar source domains. Motivated by the distribution weighted combining rule in [62], DCTN [93] suggests that the target predictor can be obtained by integrating all source predictions based on the corresponding source distribution. A major disadvantage of DCTN is that it has the same number of discriminators and category classifiers as the number of source domains, which results in linearly increasing network parameters as the number of source domains increases. ABMSDA [107] proposes a domain recognition model that mea-

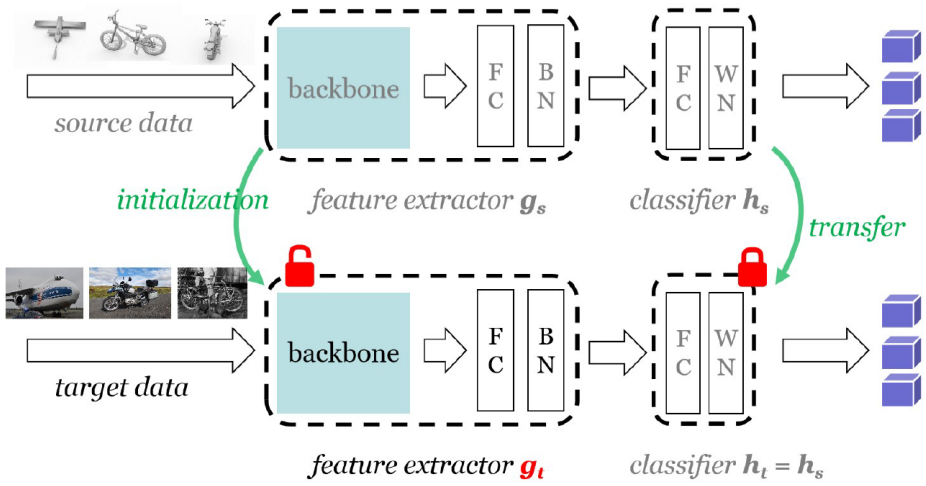


Figure 2.10: SHOT framework consists of a feature extraction module and a classifier module (hypothesis). The hypothesis is frozen, and the target domain is learned using the feature learning module. (image from [51])

sures the similarity between source domains and target domains regardless of the number of source domains, in contrast to DCTN. Consequently, the source and target domains are explicitly aligned with obtained domain correlations using Weighted Moment Distance and Attentive Classification Loss.

2.2.3 Source-free Domain Adaptation

In contrast to normal domain adaptation methods which use source data to transfer source domain knowledge to the target domain, Source Free Domain Adaptation (SFDA) methods only use unlabeled target data to distill valuable knowledge from the pre-trained source model. The SFDA methods can be categorized into the following two categories[58]. In the first group, target samples are used for self-supervised training, while the second group reconstructs virtual source data for knowledge transfer.

Self-supervised Training

Self-supervised target training methods are based on generating reliable pseudo labels for unlabelled target data. Assuming that the pre-trained source model can partially predict some target samples correctly, it may generate noisy pseudo labels. Using clustering algorithms, the noisy pseudo labels could then be further categorized. Inspired by a transfer learning setting known as Hypothesis Transfer Learning (HTL) [41], the Source HypOthesis Transfer (SHOT) [51] framework was proposed. In this setup, the source classifier (hypothesis) is frozen, and the feature extractor of the source model is fine-tuned by maximizing the mutual information between feature representations and outputs of the classifier. However, even though information maximization forces feature representations to match the hypothesis well, it may still align target feature representations with the wrong source hypothesis. To avoid this, they suggest a

pseudo-labelling approach using weighted k-means clustering[10]. Inspired by SHOT many methods have been proposed to investigate source-free domain adaptation. SHOT++[52] has developed a new labelling transfer strategy. Based on the confidence of predictions, it divides the target data into two parts, and then utilizes semi-supervised learning to increase the accuracy of predictions in the target domain that are less confident. Moreover, DECISION [2] proposes a method for adapting multi-source models to the target domain without access to the source data by combining the source models with appropriate weights. Through source Distribution Estimation, SFDA-DE [17] addresses the SFDA task. A spherical K-means clustering algorithm is used to calculate robust pseudo-labels for target data after initializing class vectors with weight vectors (anchors) learned by the pretrained model's classifier. By exploiting target data and anchors, they propose to estimate the class-conditioned feature distribution of source domains. As a final step, they sample surrogate features from the estimated distribution and then minimize a contrastive adaptation loss function to align two domains.

Virtual Source Knowledge Transfer

In virtual source knowledge transfer methods, the source model is used to synthesize some source impressions. Alternatively, the target data is converted into source style to replace the source data. Using synthetic source data, source knowledge can be distilled from pre-trained models, which can then be transferred to target models to prevent source knowledge from being forgotten. As an example, based on conditional GAN, 3C-GAN [44] creates labelled target-style training images. Besides the pre-trained prediction model, 3C-GAN consists of two components: a discriminator that matches target distributions and a generator that produces valid target-style training samples conditioned on randomly sampled labels. Pre-trained prediction models perform better in the target domain when they incorporate generated data, which in turn can promote the generator. To address the semantic segmentation problem, a generative model is used by [57] to model the distribution of target data by generating target-style images.

2.2.4 EEG-based Emotion Recognition

Emotion recognition from EEG has been an area of research since 1997 [65], and in recent years, it has received increasing attention. Numerous machine learning and signal processing techniques have been proposed over the past few decades to address the challenge of recognizing emotions from EEG [1].

Typically, EEG emotion recognition methods consist of two main components [79]: the extraction of discriminative EEG features and the classification of emotions. The feature extractors can broadly be categorized based on the type of their data into two categories: frequency-domain features and time-domain features. Time-domain features such as the Hjorth feature[29], fractal dimension feature[55], and higher order crossing feature [69] capture the temporal information in EEG signals. On the other hand, frequency-domain features are obtained by decomposing the EEG signal into several frequency bands and extracting specific features including differential entropy (DE) and power spectral density (PSD) from each band [19], [104].

Additionally, there are two categories of classification methods: linear and nonlinear. Linear tools are machine learning algorithms that assume a linear relationship between input fea-

tures and output classes. Some examples of linear tools include Support Vector Machines (SVM), Linear Discriminant Analysis (LDA), and Artificial Neural Networks (ANN). On the other hand, non-linear methods can handle much more complex relationships between input features and output classes that cannot be modeled effectively by linear methods. Therefore, non-linear methods like fractal dimension [80], sample entropy [35], and nonstationary index [39] have shown better performance than linear methods [81], however, at the expense of requiring high computation.

In [85], one of the non-linear analysis tool, Symbolic Time Analysis (STSA), is employed to overcome the issue by reducing the computational cost. Symbolic Analysis of a signal is an approach in which continuous signals are converted to symbol sequences using partitioning of the continuous signal domain. Using cosine similarity, each test EEG emotional symbolic sequence is compared with symbolic indexes (one for each emotion), and then an emotion class is assigned to the test data.

With the emergence of deep learning methods, a significant amount of attention has been devoted to using these methods, specifically for emotion recognition based on EEG signals[46]. Unlike the first group of emotion recognition methods, machine learning-based methods are capable of extracting much more complex patterns due to their large number of parameters. Various deep models have been used to recognize emotions because of their high capacity to learn representations of signals, such as recurrent neural networks and convolutional layers[73, 94, 45, 84]. For instance, multi-channel EEG data is split and stacked along the third dimension in [73]. A 3D CNN is then fed by the reshaped signal to detect emotions. STRNN [94] proposes integrating feature learning with both spatial and temporal information multichannel EEG signals. Combining CNNs and RNNs, C-RNN[45] extracts inter-channel task-related features from preprocessed multi-channel EEG signals. In a similar manner to C-RNN and STRNN, ACRNN [84] proposes an attention-based convolutional recurrent neural network that considers spatial information, temporal information and attentive information in EEG signals.

Graph representation approaches are also widely used in EEG emotion classification[79, 105]. As an example, in [79], the intrinsic inter-channel relation of electroencephalogram (EEG) signals is learned as an adjacency matrix, resulting in more discriminative feature extraction.

Additionally, EEG emotion recognition tasks can also be carried out using GAN-based methods[95, 61, 60]. GANSER [95] utilizes adversarial training to generate EEG data that is as similar to real EEG data as possible, and self-supervised learning to generalize the classifier to the augmented sample space. In order to capture the spatiotemporal features of EEG signals, they also use UNet [72] with the Channel Masking operation. In [61], GAN-like components are incorporated into the proposed framework, along with a two-step training procedure that includes pre-training and adversarial training. In pre-training, the source domain and target domain are mapped to a common feature space, and in adversarial training, the gap between these mappings is narrowed.

2.2.5 Domain Adaptation for EEG-based Emotion Recognition

Conventional machine learning algorithms fail to recognize emotions based on EEG data, due to domain shifts. In contrast to traditional machine learning methods, where training and testing data are assumed to be independent and identical, EEG signals exhibit inherent variability

because of different physical and mental conditions among multiple subjects and sessions[66]. As a result, the traditional classifier's performance often declines sharply when a non-seen subject or dataset is introduced.

Referring to the EEG of different subjects as individual domains, domain adaptation (DA)takes one or several subjects/sessions as the source domain (or source), and a new subject/session as the target domain (or target). Typically, DA methods aim to eliminate the distribution differences between the source and target domains and further learn the extracted transferable features for recognizing emotions. Existing adaptation methods can be divided into two categories: shallow domain adaptation methods and deep domain adaptation methods[90].

There have been many traditional shallow domain adaptation methods applied to emotion recognition. In [103], the performance of four different traditional domain adaptations, transductive component analysis (TCA) [67], kernel principal component analysis (KPCA) [75], transductive support vector machine (TSVM) [15], and transductive parameter transfer (TPT) [74], is compared. To match the marginal distributions of the two domains' subspaces, [12] proposed an adaptive subspace feature matching (ASFM) method.

While shallow models classify EEG features directly [3], deep neural networks (DNNs) have shown an advantage over shallow models due to their representational learning capacities. [48] proposes a domain adversarial bi-hemisphere neural network (BiDANN) method to reduce the possible domain differences in each hemisphere between the source and target domains. Considering the functional differences of network layers, [43] adapt marginal distributions at shallower layers that produce task-invariant features, and conditional distributions at deeper layers that produce task-specific features. To deal with the variability of subjects or sessions, [13] developed the multi-source marginal distribution adaptation (MS-MDA) technique which takes both domain-invariant and domain-specific features into consideration. Besides, a plug-and-play method of domain adaptation was proposed by [97] to solve the problem of a prohibitively long calibration time in emotion recognition. AD-TCN [28] combines temporal information and an adversarial discriminative domain adaptation method to adapt the target domain feature encoder to the source domain. Using the dynamic domain adaptation algorithm (DDA), [50] minimized local subdomain discrepancy as well as global domain discrepancy. In order to reduce the impact of "negative transfer", [54] proposed a subject clustering-based domain adaptation algorithm. Subspace alignment was then used to determine the emotional state of the target data based on optimal source subjects whose EEG patterns were similar.

Chapter 3

Methodology

3.1 Problem Setting

Let $\mathcal{X}$ and $\mathcal{Y} = [K] := \{1, ..., K\}$ denote the feature space and label space with K categories. Consider each subject as a domain with a joint distribution P^i_{XY} on $\mathcal{X} \times \mathcal{Y}$. Each source/subject data set $\mathcal{D}_{S_i} = \{x^j_{S_i}, y^j_{S_i}\}^{N_i}_{j=1}$ consists of N_i data points, where $x^j_{S_i} \in \mathcal{X}$ and $y^j_{S_i} \in \mathcal{Y}$ denote the j^{th} EEG data and the corresponding label respectively. Every data point x is a $(n_{ch} \cdot n_b)$ dimensional vector, where n_{ch} denotes the number of channels and n_b denotes the frequency bands.

Given a target unlabeled data set $\mathcal{D}_T = \{x^j_T\}^{N_T}_{j=1}$, our goal is to train a classifier model $\theta_T : \mathcal{X} \to \mathcal{Y}$, based on source models $\{\theta_{S_i}\}^{N_S}_{i=1}$ where the i^{th} source model $\theta_{S_i} : \mathcal{X} \to \mathcal{Y}$ is a classification model pre-trained on the i^{th} source/subject data set $\mathcal{D}_{S_i}$. Accordingly, a list of symbols and their definitions is provided in Table 3.1 that will be used in the following sections.

3.2 Proposed Solution

In this section, we propose the Attention-based Multi-Source-Free Domain Adaptation for EEG Emotion Recognition (AMFDA) method, which consists of two main phases (Figure 3.1): Source Training and Target Adaptation. During the source training phase, we construct source models based on the source data. The knowledge of pre-trained source models is then transferred to the target domain without access to either the source data or annotations. Following is a description of each phase.

3.2.1 Source Training Phase

Each source model θ_{S_i} is trained only on the corresponding data set $\mathcal{D}_{S_i}$. This phase is the only phase in which the labelled source data are used, and more importantly, in the training of each source model θ_{S_i}, the other data sets $\{\mathcal{D}_{S_j}\}^{N_S}_{j=1}$, where $i \neq j$, are not used. This ensures privacy preservation since source models are trained independently.

A source model consists of two modules: a feature extractor $f_{S_i} : \mathcal{X} \to \mathbb{R}^{d_i}$ and a classifier $g_{S_i} : \mathbb{R}^{d_i} \to \mathcal{Y}$, where d_i is the feature dimension of the i^{th} model. According to [2], more accurate and generalized source models can lead to better performance in target model adaptation.

Table 3.1: Notation Table

Symbol	Definition
$\mathcal{X}$	Feature space
$\mathcal{Y}$	Label space
S	Source domain
T	Target domain
K	Number of categories
D_{S_i}	Data set of the i^{th} source domain
D_T	Data set of the target domain
N_i	Number of data points in the i^{th} source domain
N_T	Number of data points in the target domain
d_i	Feature dimension of the i^{th} source model
d_T	Feature dimension of the target model
N_S	Number of Source models
θ_{S_i}	i^{th} source model
θ_T	Target model
f_{S_i}	Feature extractor of the i^{th} source model
f_T	Feature extractor of the target model
g_{S_i}	Classifier of the i^{th} source model
g_T	Feature extractor of the target model
n_b	Number of frequency bands
n_{ch}	Number of channels

To increase the capabilities of source models in learning generalized features, we introduce a novel channel-wise attention module.

Channel-wise Attention Layer Previous studies have shown that information from many of the channels would be redundant in a wide range of applications. [71, 33] show that only a few channels are adequate for recognizing emotions. Compared to traditional channel selection methods, automated methods like [84, 16] have demonstrated the capability to identify more relevant channels using an attention mechanism.

A channel-wise attention layer assigns weights to each channel, which indicate how critical each channel is. A channel-wise attention layer consists of a linear layer with parameters $W_1 \in \mathbb{R}^{(n_{ch} \cdot n_b) \times n_{ch}}$ and $b_1 \in \mathbb{R}^{n_{ch}}$.

$$e = \sigma(W_1.x + b_1) \tag{3.1}$$

where σ is a softmax function and $e = [e_1, ..., e_{n_{ch}}], \in \mathbb{R}^{n_{ch}}$ is the importance weight vector. In this proposed method, we use e_i for all frequency bands of channel i. Duplicating the weights for n_b times, $e' = [\underbrace{e_1, ..., e_1}_{n_b}, \underbrace{e_2, ..., e_2}_{n_b}, ..., \underbrace{e_{n_{ch}}, ..., e_{n_{ch}}}_{n_b}]$ is used to update each data point. Finally, the attentive channel feature extracted, v, via channel-wise attention will be calculated as follows:

$$v = x \,.\, e' \tag{3.2}$$

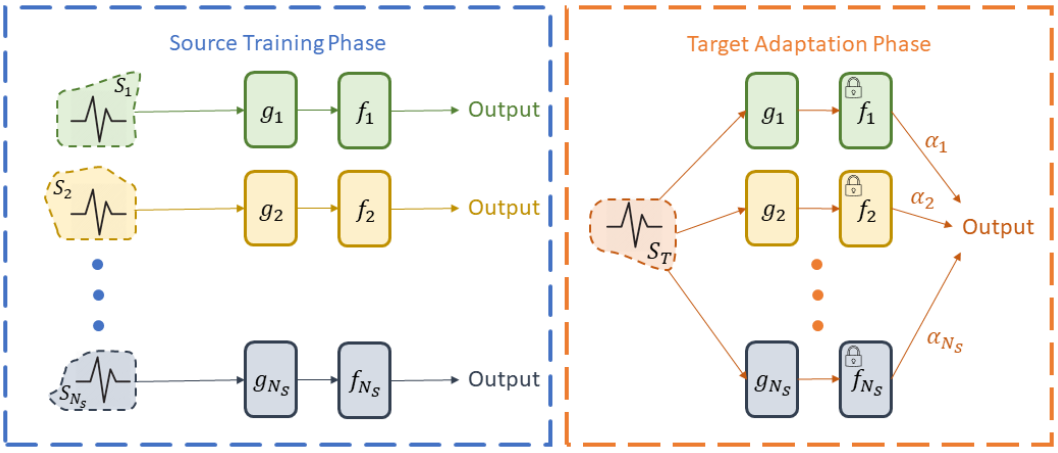


Figure 3.1: The pipeline of our AMFDA framework: During source training, source models are trained on their respective datasets. The adaptive phase involves freezing the classifiers of the source models and incorporating the source models into the target model by jointly optimizing the feature extractors and their weights.

Unlike [84], which used the attention layer on the temporal slice of EEG data, the proposed layer extracts attentive features from different frequency bands. Moreover, as shown in Table 4.3, assigning the same weight to all frequency bands of a channel outperforms the general attention layer in which frequency bands are weighted differently.

3.2.2 Target Adaptation Phase

We propose an algorithm based on DECISION [2] to aggregate the power of all source models in recognizing emotion. For the adaptation phase, a set of learnable weights $\{\alpha_i\}_{i=1}^{N_S}$, while $\alpha_i \geq 0$ and $\sum_{i=1}^{N_s} \alpha_i = 1$, corresponding to each source model are employed. The weights indicate how much the target model should depend on each source model. In other words, the higher the weight, the higher the transferability from that particular domain.

Following a similar strategy as [2, 51, 18], we fix the source classifiers, assuming that the source classifiers contain class distribution information. In this way, the ultimate goal is to find a proper objective function to be optimized over $\{\alpha_i, f_{S_i}\}_{i=1}^{N}$. The following sections describe the core loss elements for jointly adapting each source model and the learning weights.

Weighted Information Maximization Due to the lack of access to the source data in the adaptation phase, we cannot use typical distribution adaptation methods. Similar to [51, 2, 18], we encourage the networks to assign one-hot encoding as the output in order to adapt the source feature maps $\{f_{S_i}\}_{i=1}^{N}$ to the target, based on the information maximization principle [2, 51, 7, 38, 78, 31]. The IM loss consists of two terms, conditional entropy loss

$$\mathcal{L}_{ent} = -\mathbb{E}_{x_T \in D_T} \left[\sum_{i=1}^{K} \delta_i \left(\theta_T(x_T)\right) \log \delta_i \left(\theta_T(x_T)\right) \right] \tag{3.3}$$

where $\theta_T(x_T) = \sum_{j=1}^{N} \alpha_j \theta_{S_j}(x_T)$, and $\delta(.)$ denotes a softmax operation with $\delta_i(x) = \frac{exp(x_i)}{\sum_{j=1}^{K} exp(x_j)}$ for $v \in \mathbb{R}^K$, and diversity loss

$$\mathcal{L}_{div} = \sum_{i=1}^{K} \hat{p}_i \log \hat{p}_i = D_{KL}(\hat{p}, \frac{1}{K}\mathbf{1}_K) - \log K$$

where $\hat{p} = \mathbb{E}_{x_T \in D_T} [\delta(\theta_T(x_T))]$ is the mean output of the whole target domain, and $\mathbf{1}_K$ is a K-dimensional vector with all ones. Combining 3.3 and 3.4 the IM loss will be

$$\mathcal{L}_{IM} = \mathcal{L}_{ent} + \mathcal{L}_{div} \tag{3.4}$$

By incorporating a diversity term into the model, IM loss is more likely to produce a balanced class diversity output than conditional entropy loss. As a result, the model does not always predict the same class to minimize conditional entropy.

Self-Supervised Pseudo-labeling based on Weighted Majority Voting: Using weighted information maximization alone would result in incorrect classification due to domain shift. Taking inspiration from [2], to alleviate this problem, we suggest using a self-supervised clustering method, such as pseudo-labelling [51, 2, 11]. Based on the j^{th} source model, the k^{th} centroid of target data is

$$c_{k,j}^{(0)} = \frac{\sum_{x_T \in D_T} \delta_k(\theta_{S_j}(x_T))\, f_{S_j}(x_T)}{\sum_{x_T \in D_T} \delta_k(\theta_{S_j}(x_T))} \tag{3.5}$$

Instead of aggregating all the centroids of clusters and assigning the labels based on the aggregated clusters like DECISION[2], we propose to label each data point using all source models separately, and then determine the final labels based on the learnable weights $\{\alpha_i\}_{i=1}^{N_s}$ previously introduced. So, based on the calculated centroids of source domains, we assign the label of the nearest centroid to each data point:

$$\hat{y}_j^{(i)} = \arg\min_k \|f_{S_j}(x_T) - c_{k,j}^{(i)}\|_2^2 \tag{3.6}$$

Then, the labels can be aggregated as follows:

$$\hat{y}_T^{(i)} = \sum_{j=1}^{N_s} \alpha_j\, \hat{y}_j^{(i)} \tag{3.7}$$

Next, we need to repeat the previous steps, by replacing the model prediction in Eq 3.5 with the new pseudo-label.

$$c_{k,j}^{(i)} = \frac{\sum_{x_T \in D_T} \mathbb{1}\{\hat{y}_T^{(i-1)} = k\}\, f_{S_j}(x_T)}{\sum_{x_T \in D_T} \mathbb{1}\{\hat{y}_T^{(i-1)} = k\}} \tag{3.8}$$

where $\mathbb{1}[x = y]$ is 1 if x and y are equal, and otherwise 0. Then, using Eq 3.6 and 3.7 the pseudo-labels can be updated multiple times, but the first update is sufficient according to

[51, 2]. The final step is calculating the cross-entropy loss between the labels predicted by the target model and the pseudo labels.

$$\mathcal{L}_{pl} = -\mathbb{E}_{x_T \in D_T} \sum_{k=1}^{K} \mathbb{1}\{\hat{y}_T = k\} \log \delta_k(\theta_T(x_T)) \tag{3.9}$$

Self-Supervised Contrastive Loss with EEG-based Augmented Data As the number of data points in a leave-one-subject-out data set is limited, we suggest a two-step solution to increase the robustness of transferring knowledge from the source models to the target model.

As a first step, a novel EEG-based augmentation is used to generate new data points ($\hat{x}_T$) in the target domain [92]. By weakening randomly selected channels and bands, we encourage source feature extractors to be sensitive to both crucial and less crucial channels. Thus, they will become more robust to noise. We define a weakening probability p, which is the probability of choosing a channel-band to weaken. The magnitude of the selected channel-bands will be decreased by a coefficient α_{cb}.

Secondly, a self-supervised contrastive loss approximates data with its augmentation [14]. Let us consider a set $I \equiv \{x_T^1, ..., x_T^{N_T}, \hat{x}_T^1, ..., \hat{x}_T^{N_T}\}$ containing all target data points and their augmentations.

In this set, the index i is called anchor and the corresponding data point $j(i)$ is called positive, whereas the other $2(N_T - 1)$ points are considered as negative $A(i) \equiv I \setminus i$. Based on the above description, the contrastive loss would be:

$$\mathcal{L}_{con} = \sum_{m=1}^{N_s} \mathcal{L}_{con}^m = -\sum_{m=1}^{N_s} \sum_{i \in I} \log \frac{\exp\left(f_{S_m}(x_T^i) \cdot f_{S_m}(x_T^{j(i)})/\tau\right)}{\sum_{a \in A(i)} \exp\left(f_{S_m}(x_T^i) \cdot f_{S_m}(x_T^a)/\tau\right)} \tag{3.10}$$

To summarize, given N_s source models $\{\theta_{S_i}\}_{i=1}^{N_s} = \{f_{S_i} \circ g_{S_i}\}_{i=1}^{N_s}$ and target data set $D_T = \{x_T^i\}_{i=1}^{N_T}$, we fix the source classifiers $\{g_{S_i}\}_{i=1}^{N_s}$ and optimize $\{\alpha_i, f_{S_i}\}_{i=1}^{N}$ based on the following final objective:

$$\mathcal{L}_{tot} = \mathcal{L}_{IM} + \lambda_1 \mathcal{L}_{pl} + \lambda_2 \mathcal{L}_{con} \tag{3.11}$$

where λ_1 and λ_2 are hyperparameters. It is important to note that the $\mathcal{L}_{tot}$ should be minimized with the following condition:

$$\forall i \in \{1, 2, ..., N_s\} \qquad \alpha_i \geq 0, \quad \sum_{i=1}^{N_s} \alpha_i = 1 \tag{3.12}$$

Chapter 4

Experiments and Results

4.1 Datasets

We evaluated the proposed method using two public emotion recognition data sets provided by BCMI Laboratory: SEED and SEED-IV. Table 4.1 summarizes the technical specifications of the two datasets.

SEED [101] [20], SJTU Emotion EEG Dataset, contains the EEG data of 15 subjects in 3 sessions exposed to audio-visual stimuli. The stimuli are carefully selected film clips intended to produce positive, negative, and neutral emotions. Each film clip lasts about four minutes and is well-edited to evoke coherent emotions and maximize emotional impact. There are a total of 15 trials in each experiment. Each clip was preceded by a 5-second hint, followed by 45 seconds for self-assessment and 15 seconds for rest. It was determined that the subjects exhibited the expected emotion based on self-assessment. The order of presentation is arranged so that two film clips that target the same feeling are not shown consecutively. As the movie clip was viewed, the ESI Neuroscan system recorded EEG signals in 62 channels at a sampling rate of 1000 Hz.

SEED-IV [100] is an extension of the SEED dataset, which can also be used to evaluate EEG-based emotion recognition models. Unlike SEED, SEED-IV has four categories of emotions: happy, sad, fear, and neutral. The dataset consists of 15 subjects who watched six film clips for each emotion class in each session, resulting in 24 trials. Similar to SEED, the subjects were required to experiment three times on different days, while a 62-channel device was used to collect their EEG signals and eye movements.

4.2 Preprocessing

In EEG emotion studies, researchers tend to extract frequency-based features instead of analyzing raw time series EEG data [34, 102]. To have a fair comparison with other methods, we used the preprocessed EEG data which is published by BCMI Laboratory [1]. In this preprocessing scheme, raw EEG signals are sliced into nonoverlapping 1-second and 4-second segments, for SEED and SEED-IV datasets, respectively. Afterward, the EEG data are downsampled to

[1] https://bcmi.sjtu.edu.cn/

Table 4.1: Technical Comparison Between SEED and SEED-IV

Item	SEED	SEED-IV
EEG device	ESI NeuroScan	ESI NeuroScan
Emotions	Happy, Sad, Neutral	Happy, Sad, Neutral, Fear
# of channels	62	62
# of recording session per subject	3	3
# of subjects	15	15
# of trials per session	15	24
Trial length	Approx. 2 minutes	Approx. 4 minutes

200 Hz, and a bandpass filter between 1 Hz and 75 Hz is applied to remove noise and artifacts. According to [20], differential entropy (DE) is better suited to recognizing emotions than the traditional frequency-domain feature, named energy spectrum (ES). In order to calculate the frequency domain features, a 512-point short-time Fourier transform with a non-overlapping Hanning window of 1s was used. Considering five frequency bands: δ waves (1 $\sim$ 3 Hz), θ waves (4 $\sim$ 7 Hz), α waves (8 $\sim$ 13 Hz), β waves (14 $\sim$ 31 Hz), and γ waves (31 $\sim$ 50Hz), the differential entropy (DE) [19] features within each segment at the mentioned frequency bands can be calculated as follows:

$$\text{DE} = -\int f(\mathbf{x}) \log(f(\mathbf{x})) d\mathbf{x} \tag{4.1}$$

where x is a one-channel EEG signal. Assuming that the EEG data obeys the Gaussian distribution $N(\mu,\ \sigma^2)$ on each frequency band [76], the DE features can be extracted by the following formulation:

$$\text{DE} = -\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{(\mathbf{x}-\mu)^2}{2\sigma^2}} \log\left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{(\mathbf{x}-\mu)^2}{2\sigma^2}}\right) d\mathbf{x} = \frac{1}{2}\log(2\pi e\sigma^2) \tag{4.2}$$

Lastly, DE features form a (62×5-D) vector that will be smoothed using moving averages and linear dynamic systems (LDS) [77, 101, 20]. The calculated DE features as well as the raw EEG data are published by BCMI Labratory[2].

4.3 Implementation Details

We are using the Pytorch platform for all preprocessing, model training, and evaluation, with GPU (Tesla T4) acceleration provided by Google Colab for the training phase and inference phase. Training the source models takes an average of 2.6 seconds and 1.9 seconds for SEED and SEED-IV, respectively, while adapting the models to the target takes an average of 49.97 seconds and 30.33 seconds, respectively. All models consist of an attention layer and two fully connected layers each followed by a LeakyReLU layer as the feature extractor and a fully

[2]https://bcmi.sjtu.edu.cn/

connected layer as the classifier. Despite having the same architecture in our setup, each model can be designed differently. The only requirement for the adaptation phase is the same feature dimension d_i, which is 32-D in our case.

For fair comparison and consistency, we follow the leave-one-subject-out setup (LOSO) similar to [13, 42]. In this setup, a subject is selected as the target and all the rest as sources, then the calculated accuracy will be averaged on all scenarios.

As mentioned in Section. 3.2, the first step is to train all source models on its data set. Training is done using the Adam [37] optimizer with an initial learning rate of 0.01 and a 10-epoch learning period. During the target adaptation phase, all source models are aggregated according to 3.2.2. While the classifiers are fixed, the Adam optimizer is used to train the feature extractors and aggregation weights for five epochs. The initial learning rate for feature extractor parameters is 0.001, whereas the aggregation weights are learned at a learning rate of 0.01.

4.4 Results

In Table 4.2, the performance of our proposed method as well as several baselines on SEED and SEED-IV is presented. In this table, methods are categorized into three groups. First, "Multiple(w)" methods use the data of all source domains in the adaptation phase. On the other hand, in "Single(w/o)" methods, a model is pre-trained on each source domain, then each of these models will be separately adapted to the target domain without accessing their source data. Finally, in the last category, "Multiple(w/o)", none of the methods use source domain data, they only adapt the aggregated knowledge learned by source models to the target domain.

In addition to typical domain adaption methods like TCA, KPCA, and TPT, a few other baselines are also used to be compared without our method. A description of these baselines and their specific structures follows:

DAN: According to [42], there are three fully connected are used as the feature extractor and two layers as the classifier. The node numbers are 128, 128, 64, 64, and K from the input to the output, respectively. The MK-MMD loss is calculated at the end of the feature extractor.

DANN: In [42], the feature extractor contains two layers with 128 nodes, and the classifier includes three layers with node numbers 64, 64, and K.

DCORAL: For this baseline, we use the same number of layers and parameters as DAN and DANN, and correlation alignment (CORAL) is employed to adapt the knowledge to the target model.

MS-MDA: [13] contains of three components: Common Feature Extractor (CFE), Domain-specific Feature Extractor (DSFE), and Domain-specific Classifier (DSC). While the CFE module is three shared fully connected layers, both the DSFE, and the DSC are composed of a single linear layer.

BiDANN: BiDANN [47] framework consists of two feature extractors (left and right hemisphere), local, global discriminator, and a classifier.

Source-best/worst: Source-best and source-worst refer to the average of the best and the worst accuracy of source models, respectively, without any adaptation phase.

SHOT-best/worst: After adapting sources separately using SHOT [51] method, SHOT-best and SHOT-worst show the average of the highest and lowest accuracy.

SHOT: Assuming that all source subjects are from a single domain, the SHOT method adapts the knowledge of the source model to the target domain.

DECISION: [2] proposed a general unsupervised multi-source domain adaptation that does not use source data in the adaptation phase.

Since many previous studies have not published their code, we have reported their results. The performances of DANN [23], BiDANN [47], and other typical deep learning domain adaptation methods like TCA [67], KPCA [75], and TPT[74] have been borrowed from[42]. However, we have reproduced and verified the reported performance of the DAN[59], DCORAL [83], and MS-MDA [13] methods. Finally, in this study, a variety of baselines, including source-best/worst, shot-best/worst, DECISION, and SHOT, have been developed and implemented.

As the results are shown in Table 4.2, in spite of the huge difference between the most and the least accurate source models, our method increases accuracy of emotion recognition on SEED and SEED-IV by 4.4 and 5.7 percent, respectively, over the best adapted source models (SHOT-best). In addition, since each AMFDA source model is trained only with its corresponding data, it is less generalized than SHOT's model, which is developed with the entire dataset. However, AMFDA could outperform the SHOT method in the adaptation phase since it restricts negative transfer via joint adaptation of the source models and the assembled weights. Finally, while AMFDA preserves privacy by requiring no access to the source data, it performs at par with the unsupervised domain adaptation emotion recognition state-of-the-art.

4.5 Ablation and Analysis

4.5.1 Comparison between UDA methods with AMFDA

Unsupervised domain adaptation (UDA) methods like MS-MDA generally require both source and target data for adapting the knowledge from source domains to the model. When there are a lot of source domains, computing costs may be significantly higher. For example, in our problem setting, the source data consist of the EEG data of 14 subjects and the target data is the EEG data of one subject.

In contrast to MS-MDA, which prepares the target model in a single step, AMFDA is a two-step process. Although this difference makes comparing computation costs difficult, we can compare total computation costs. Using Tesla T4 GPUs, the total training process for MS-MDA takes 1195 seconds, while for AMFDA, the source training takes 36.4 seconds, and the adaptation phase takes 49.97 seconds. By comparing these to computation costs, we can see that our proposed method is much more efficient.

Table 4.2: Comparison cross-subject classification accuracies (mean ± std (%)) of different methods on SEED and SEED-IV. Multiple and Single denotes the methods which uses multiple and single sources, respectively, for domain adaptation, while (w) and (w/o) are abbreviations of with source data and without source data respectively

	Methods	SEED	SEED-IV
Multiple(w)	TCA [42]	64.00 ± 14.66	53.97 ± 8.05
	KPCA [42]	69.02 ± 9.25	51.76 ± 12.89
	TPT [42]	75.17 ± 12.83	52.43 ± 14.43
	DAN [42]	83.81 ± 8.56	58.87 ± 8.13
	DANN [42]	79.19 ± 13.14	54.63 ± 8.03
	DCORAL	66.39 ± 7.55	51.85 ± 7.30
	MS-MDA [13]	82.67 ± 9.51	$\mathbf{67.96 \pm 11.94}$
	BiDANN [47]	$\mathbf{83.28 \pm 9.10}$	-
Single(w/o)	Source-best	73.19 ± 7.71	52.02 ± 4.28
	Source-worst	44.75 ± 6.95	25.84 ± 7.66
	SHOT-best	76.93 ± 6.79	56.05 ± 7.25
	SHOT-worst	46.45 ± 9.75	23.89 ± 7.06
Multiple(w/o)	SHOT [51]	78.20 ± 5.83	60.77 ± 11.47
	DECISION [2]	77.78 ± 7.32	60.07 ± 10.43
	AMFDA (Ours)	$\mathbf{81.37 \pm 7.94}$	$\mathbf{61.79 \pm 9.44}$

4.5.2 Contribution of each component

We performed an ablation study to investigate the impact of each component in our model and each term in the adaptation objective function in our method. The performance of several variations of our proposed method is compared in Table 4.3. According to the results, adding a channel-wise attention layer to the beginning of the feature extractor will improve classification accuracy. In addition, contrastive learning is effective during the adaptation phase, even if it is not helpful during the source training phase.

4.5.3 Analysis on the learned weights

According to Section 3.2.2, the proposed emotion recognition optimizes the feature extractors of sources $\{f_S^i\}_{i=1}^N$, as well as weights $\{\alpha_i\}_{i=1}^N$ in the adaptation phase. To understand the impact of the weights, we recommend freezing the feature extractors and only optimizing the weights. As expected, this setup performs better than trivially assigning equal weights to all source models, as shown in Table 4.4.

Figure 4.1 contains two line graphs. The first one demonstrates the classification accuracy of each source model on the target domain without adaptation. While the other shows the assigned learned weights to each source model after the adaptation. Although the patterns of these two graphs are not highly correlated, higher weights indicate which source models perform more effectively on the target domain. As a result, the target model relies more on source models with higher weights. It can therefore be used as a proxy indicator when selecting

Table 4.3: Ablation study of our method on SEED and SEED-IV. T-Con and S-Con refer models which are trained using contrastive loss in the target training and source training phase, respectively. C-Attention indicate that the attention layer used in the feature extractor is channel-wise, while E-attention refers to electrode-wise attention.

T-Con	S-Con	E-Attention	C-Attention	SEED	SEED-IV
✗	✗	✗	✗	78.68 ± 8.27	61.22 ± 8.49
✓	✗	✗	✗	79.07 ± 7.76	60.36 ± 9.46
✗	✓	✗	✗	74.63 ± 10.29	52.98 ± 11.36
✗	✗	✓	✗	78.22 ± 7.60	60.76 ± 9.64
✗	✗	✗	✓	79.52 ± 6.63	61.59 ± 9.35
✓	✗	✗	✓	$\mathbf{81.37 \pm 7.94}$	$\mathbf{61.79 \pm 9.44}$

Table 4.4: Performance on freezing backbone network on SEED and SEED-IV. AMFDA-weight is optimized solely over source weights and performs better than uniform weighting consistently.

Method	SEED	SEED-IV
Source-Ens	40.06 ± 6.49	21.19 ± 0.26
AMFDA-weights	$\mathbf{77.99 \pm 7.12}$	$\mathbf{54.80 \pm 6.60}$

new models.

4.5.4 Confusion Matrix Visualization

To further explore the performance of the AMFDA method, we computed the confusion matrices for SEED and SEED-IV datasets in Figure 4.2. In a confusion matrix, rows indicate the target data's actual labels, and columns indicate the predicted labels. As a result, diagonal elements represent correct predictions, while off-diagonal elements represent incorrect predictions. It is important to note that the SEED and SEED-IV datasets are balanced, meaning that each output class (or target class) is represented by the same number of input samples.

Accordingly, the presented confusion matrices demonstrate that positive emotions (Happy class) are more recognizable than negative (Sad and Fear classes) and neutral emotions. Moreover, in the SEED dataset, the Neutral-Sad classes and the Neutral-Happy classes are more likely to be misclassified than the Happy-Sad classes. The SEED-IV dataset contains two negative classes, Fear and Sad, which are more susceptible to misclassification.

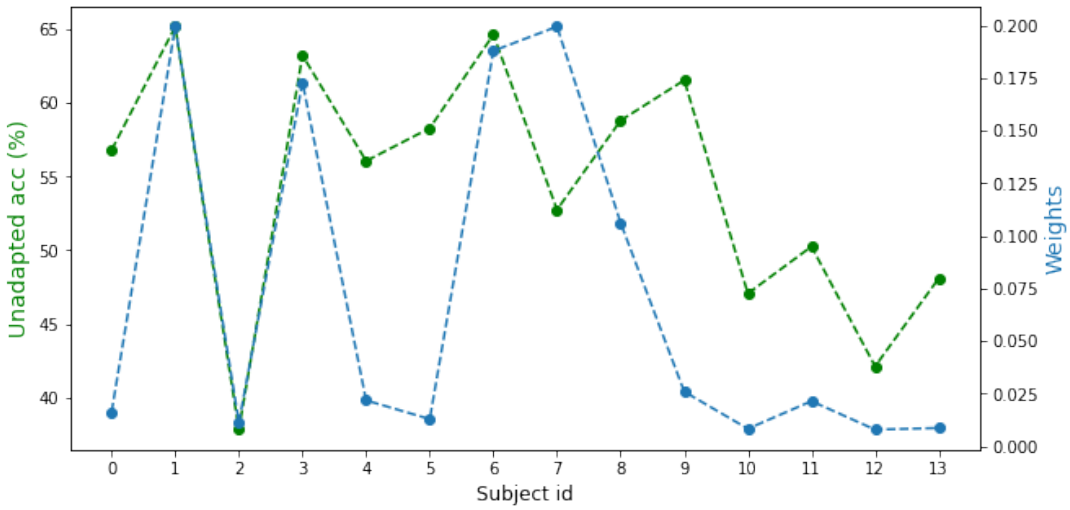
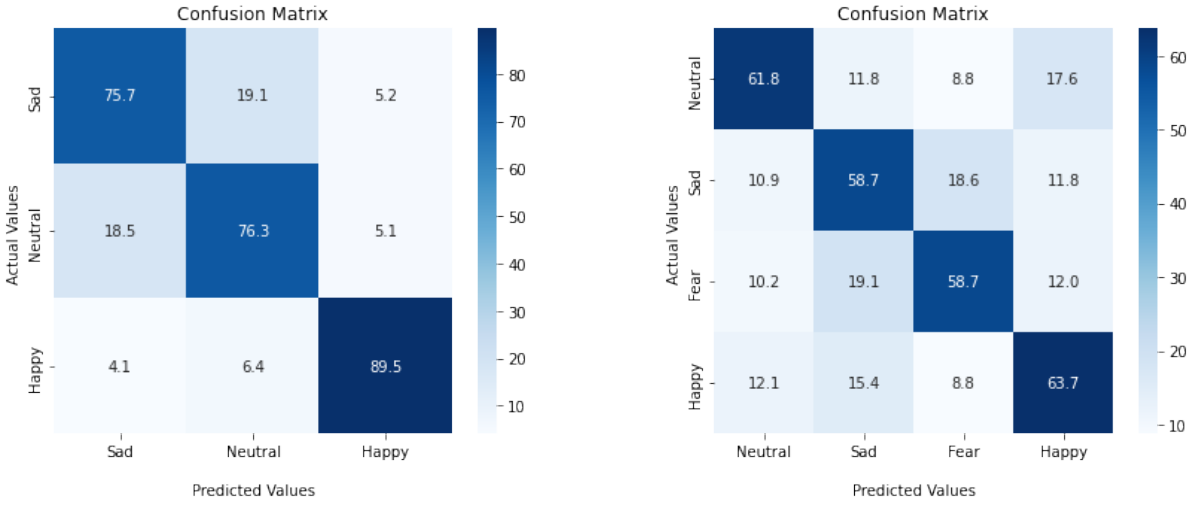


Figure 4.1: For the 15th subject on the SEED dataset, the weights learned by our framework correlate positively with the unadapted source model performance.



(a) SEED dataset

(b) SEED-IV dataset

Figure 4.2: The confusion matrices of the subject-dependent EEG emotion recognition results using the AMFDA method on the SEED and SEED-IV datasets

Chapter 5

Discussion & Conclusion

5.1 Discussions

This thesis aims to propose AMFDA, an EEG-based emotion recognition source-free domain adaptation method that can be applied to multiple source domain situations. The proposed setup merely requires well-trained private source models to be jointly adapted to the target domain. By incorporating the differences between various sources/subjects, the AMFDA method can achieve comparable performance in emotion recognition tasks to state-of-the-art unsupervised domain adaptation methods. In addition, we investigated the impact of channel-wise attention modules on emotion recognition models and contrastive learning's effectiveness in generalizing models to channel-wise noise. In the end, we hope that the findings of this study can serve as inspiration for future research on affective computing based on EEG.

5.2 Applications

Using EEG-based emotion recognition, individuals can gain valuable insights into their emotional states and the ability to accurately identify emotions based on brain activity has significant potential applications in fields such as medicine, marketing, psychology, neuroscience, and artificial intelligence. Some potential applications of this technology include assessing and monitoring emotional states in individuals with mental disorders, utilizing neurofeedback therapy to help individuals regulate their emotional states, and adapting consumer products such as virtual reality systems and video games to the user's emotional state. Furthermore, it can also be used to assess the effectiveness of marketing and advertising campaigns by analyzing consumer emotional responses and identifying and addressing emotional challenges in education.

5.3 Limitations

While our proposed method addresses EEG-based emotion classification without accessing the source data, several limitations can be addressed in future work. Since each subject is treated as a source in the proposed method for emotion recognition, there are as many source models

as there are subjects. In this case, the computation cost increases with the number of subjects. Additionally, the performance of the target model is influenced by the performance of the source models. Thus, the target model will perform better if the source models have higher classification capabilities.

5.4 Future Research

To overcome the limitations previously described, the current work could substantially benefit from two types of solutions. Firstly, by investigating ways to improve the generalizability of each source model to an unseen new source domain, it is possible to formulate more robust source models for the target model. Secondly, investigating ways to consolidate all source models into one target model while minimizing the influence of non-relevant source domains may improve domain adaptation.

Bibliography

[1] Lyubomir I Aftanas, Natalya V Reva, Anton A Varlamov, Sergey V Pavlov, and Victor P Makhnev. Analysis of evoked eeg synchronization and desynchronization in conditions of emotional activation in humans: temporal and topographic characteristics. *Neuroscience and behavioral physiology*, 34:859–867, 2004.

[2] Sk Miraj Ahmed, Dripta S Raychaudhuri, Sujoy Paul, Samet Oymak, and Amit K Roy-Chowdhury. Unsupervised multi-source domain adaptation without access to source data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10103–10112, 2021.

[3] Soraia M Alarcao and Manuel J Fonseca. Emotions recognition using eeg signals: A survey. *IEEE Transactions on Affective Computing*, 10(3):374–393, 2017.

[4] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Vaughan. A theory of learning from different domains. *Machine Learning*, 79:151–175, 2010.

[5] John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman. Learning bounds for domain adaptation. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007.

[6] Léon Bottou. Stochastic gradient descent tricks. In *Neural networks: Tricks of the trade*, pages 421–436. Springer, 2012.

[7] John Bridle, Anthony Heading, and David MacKay. Unsupervised classifiers, mutual information and 'phantom targets. In J. Moody, S. Hanson, and R.P. Lippmann, editors, *Advances in Neural Information Processing Systems*, volume 4. Morgan-Kaufmann, 1991.

[8] Jennifer C Britton, K Luan Phan, Stephan F Taylor, Robert C Welsh, Kent C Berridge, and Israel Liberzon. Neural correlates of social and nonsocial emotions: An fmri study. *Neuroimage*, 31(1):397–409, 2006.

[9] Hanshu Cai, Zhidiao Qu, Zhe Li, Yi Zhang, Xiping Hu, and Bin Hu. Feature-level fusion approaches based on multimodal eeg data for depression recognition. *Information Fusion*, 59:127–138, 2020.

[10] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 132–149, 2018.

[11] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European conference on computer vision (ECCV)*, pages 132–149, 2018.

[12] Xin Chai, Qisong Wang, Yongping Zhao, Yongqiang Li, Dan Liu, Xin Liu, and Ou Bai. A fast, efficient domain adaptation technique for cross-domain electroencephalography (eeg)-based emotion recognition. *Sensors*, 17(5):1014, 2017.

[13] Hao Chen, Ming Jin, Zhunan Li, Cunhang Fan, Jinpeng Li, and Huiguang He. Ms-mda: Multisource marginal distribution adaptation for cross-subject and cross-session eeg emotion recognition. *Frontiers in Neuroscience*, 15, 2021.

[14] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, pages 1597–1607. PMLR, 2020.

[15] Ronan Collobert, Fabian Sinz, Jason Weston, Léon Bottou, and Thorsten Joachims. Large scale transductive svms. *Journal of Machine Learning Research*, 7(8), 2006.

[16] Lina Deng, Xiaoliang Wang, Frank Jiang, and Robin Doss. Eeg-based emotion recognition via capsule network with channel-wise attention and lstm models. *CCF Transactions on Pervasive Computing and Interaction*, 3(4):425–435, 2021.

[17] Ning Ding, Yixing Xu, Yehui Tang, Chao Xu, Yunhe Wang, and Dacheng Tao. Source-free domain adaptation via distribution estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7212–7222, 2022.

[18] Jiahua Dong, Zhen Fang, Anjin Liu, Gan Sun, and Tongliang Liu. Confident anchor-induced multi-source free domain adaptation. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 2848–2860. Curran Associates, Inc., 2021.

[19] Ruo-Nan Duan, Jia-Yi Zhu, and Bao-Liang Lu. Differential entropy feature for eeg-based emotion classification. In *2013 6th International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 81–84. IEEE, 2013.

[20] Ruo-Nan Duan, Jia-Yi Zhu, and Bao-Liang Lu. Differential entropy feature for EEG-based emotion classification. In *6th International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 81–84. IEEE, 2013.

[21] Amit Etkin, Tobias Egner, and Raffael Kalisch. Emotional processing in anterior cingulate and medial prefrontal cortex. *Trends in cognitive sciences*, 15(2):85–93, 2011.

[22] Abolfazl Farahani, Sahar Voghoei, Khaled Rasheed, and Hamid R Arabnia. A brief review of domain adaptation. *Advances in Data Science and Information Engineering*, pages 877–894, 2021.

[23] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

[24] Muhammad Ghifary, W Bastiaan Kleijn, Mengjie Zhang, David Balduzzi, and Wen Li. Deep reconstruction-classification networks for unsupervised domain adaptation. In *European Conference on Computer Vision*, pages 597–613. Springer, 2016.

[25] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[26] Arthur Gretton, Karsten Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alex Smola. A kernel method for the two-sample-problem. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2006.

[27] Jiang Guo, Darsh J Shah, and Regina Barzilay. Multi-source domain adaptation with mixture of experts. *arXiv preprint arXiv:1809.02256*, 2018.

[28] Zhipeng He, Yongshi Zhong, and Jiahui Pan. An adversarial discriminative temporal convolutional network for eeg-based cross-domain emotion recognition. *Computers in Biology and Medicine*, 141:105048, 2022.

[29] Bo Hjorth. Eeg analysis based on time domain properties. *Electroencephalography and clinical neurophysiology*, 29(3):306–310, 1970.

[30] Judy Hoffman, Mehryar Mohri, and Ningshan Zhang. Algorithms and theory for multiple-source adaptation. *Advances in Neural Information Processing Systems*, 31, 2018.

[31] Weihua Hu, Takeru Miyato, Seiya Tokui, Eiichi Matsumoto, and Masashi Sugiyama. Learning discrete representations via information maximizing self-augmented training. In *International Conference on Machine Learning*, pages 1558–1567. PMLR, 2017.

[32] Spiros V. Ioannou, Amaryllis T. Raouzaiou, Vasilis A. Tzouvaras, Theofilos P. Mailis, Kostas C. Karpouzis, and Stefanos D. Kollias. Emotion recognition through facial expression analysis based on a neurofuzzy network. *Neural Networks*, 18(4):423–435, 2005. Emotion and Brain.

[33] Noppadon Jatupaiboon, Setha Pan-Ngum, and Pasin Israsena. Emotion classification using minimal eeg channels and frequency bands. *The 2013 10th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, pages 21–24, 2013.

[34] Robert Jenke, Angelika Peer, and Martin Buss. Feature extraction and selection for emotion recognition from eeg. *IEEE Transactions on Affective Computing*, 5:327–339, 2014.

[35] Xiang Jie, Rui Cao, and Li Li. Emotion recognition based on the sample entropy of eeg. *Bio-medical materials and engineering*, 24(1):1185–1192, 2014.

[36] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4893–4902, 2019.

[37] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014.

[38] Andreas Krause, Pietro Perona, and Ryan Gomes. Discriminative clustering by regularized information maximization. In J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems*, volume 23. Curran Associates, Inc., 2010.

[39] Eleni Kroupi, Ashkan Yazdani, and Touradj Ebrahimi. Eeg correlates of different emotional states elicited during watching music videos. In *Affective Computing and Intelligent Interaction: Fourth International Conference, ACII 2011, Memphis, TN, USA, October 9–12, 2011, Proceedings, Part II*, pages 457–466. Springer, 2011.

[40] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

[41] Ilja Kuzborskij and Francesco Orabona. Stability and hypothesis transfer learning. In Sanjoy Dasgupta and David McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 942–950, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR.

[42] He Li, Yi-Ming Jin, Wei-Long Zheng, and Bao-Liang Lu. Cross-subject emotion recognition using deep adaptation networks. In Long Cheng, Andrew Chi Sing Leung, and Seiichi Ozawa, editors, *Neural Information Processing*, pages 403–413, Cham, 2018. Springer International Publishing.

[43] Jinpeng Li, Shuang Qiu, Changde Du, Yixin Wang, and Huiguang He. Domain adaptation for eeg emotion recognition based on latent representation similarity. *IEEE Transactions on Cognitive and Developmental Systems*, PP:1–1, 10 2019.

[44] Rui Li, Qianfen Jiao, Wenming Cao, Hau-San Wong, and Si Wu. Model adaptation: Unsupervised domain adaptation without source data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9641–9650, 2020.

[45] Xiang Li, Dawei Song, Peng Zhang, Guangliang Yu, Yuexian Hou, and Bin Hu. Emotion recognition from multi-channel eeg data through convolutional recurrent neural network. In *2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 352–359, 2016.

[46] Xiang Li, Yazhou Zhang, Prayag Tiwari, Dawei Song, Bin Hu, Meihong Yang, Zhigang Zhao, Neeraj Kumar, and Pekka Marttinen. Eeg based emotion recognition: A tutorial and review. *ACM Computing Surveys (CSUR)*, 2022.

[47] Yang Li, Wenming Zheng, Yuan Zong, Zhen Cui, Tong Zhang, and Xiaoyan Zhou. A bi-hemisphere domain adversarial neural network model for eeg emotion recognition. *IEEE Transactions on Affective Computing*, 12(2):494–504, 2021.

[48] Yang Li, Wenming Zheng, Yuan Zong, Zhen Cui, Tong Zhang, and Xiaoyan Zhou. A bi-hemisphere domain adversarial neural network model for eeg emotion recognition. *IEEE Transactions on Affective Computing*, 12(2):494–504, 2021.

[49] Yitong Li, David E Carlson, et al. Extracting relationships by multi-domain matching. *Advances in Neural Information Processing Systems*, 31, 2018.

[50] Zhunan Li, Enwei Zhu, Ming Jin, Cunhang Fan, Huiguang He, Ting Cai, and Jinpeng Li. Dynamic domain adaptation for class-aware cross-subject and cross-session eeg emotion recognition. *IEEE Journal of Biomedical and Health Informatics*, pages 1–10, 2022.

[51] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 6028–6039. PMLR, 2020.

[52] Jian Liang, Dapeng Hu, Yunbo Wang, Ran He, and Jiashi Feng. Source data-absent unsupervised domain adaptation through hypothesis transfer and labeling transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[53] Kristen A Lindquist and Lisa Feldman Barrett. A functional architecture of the human brain: emerging insights from the science of emotion. *Trends in Cognitive Sciences*, 16(11):533–540, 2012.

[54] Jin Liu, Xinke Shen, Sen Song, and Dan Zhang. Domain adaptation for cross-subject emotion recognition by subject clustering. In *2021 10th International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 904–908. IEEE, 2021.

[55] Yisi Liu and Olga Sourina. Real-time fractal-based valence level recognition from eeg. In *Transactions on computational science XVIII: special issue on Cyberworlds*, pages 101–120. Springer, 2013.

[56] Yong-Jin Liu, Minjing Yu, Guozhen Zhao, Jinjing Song, Yan Ge, and Yuanchun Shi. Real-time movie-induced discrete emotion recognition from eeg signals. *IEEE Transactions on Affective Computing*, 9(4):550–562, 2018.

[57] Yuang Liu, Wei Zhang, and Jun Wang. Source-free domain adaptation for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1215–1224, 2021.

[58] Yuang Liu, Wei Zhang, Jun Wang, and Jianyong Wang. Data-free knowledge transfer: A survey. *arXiv preprint arXiv:2112.15278*, 2021.

[59] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In *International Conference on Machine Learning*, pages 97–105. PMLR, 2015.

[60] Yun Luo and Bao-Liang Lu. Eeg data augmentation for emotion recognition using a conditional wasserstein gan. In *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 2535–2538. IEEE, 2018.

[61] Yun Luo, Si-Yang Zhang, Wei-Long Zheng, and Bao-Liang Lu. Wgan domain adaptation for eeg-based emotion recognition. In *International Conference on Neural Information Processing*, pages 275–286. Springer, 2018.

[62] Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation with multiple sources. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems*, volume 21. Curran Associates, Inc., 2008.

[63] Douglas M McNair, Maurice Lorr, Leo F Droppleman, et al. Eits manual for the profile of mood states. 1971.

[64] Angeliki Metallinou, Sungbok Lee, and Shrikanth Narayanan. Audio-visual emotion recognition using gaussian mixture models for face and voice. In *2008 Tenth IEEE International Symposium on Multimedia*, pages 250–257, 2008.

[65] Toshimitsu Musha, Yuniko Terasaki, Hasnine A Haque, and George A Ivamitsky. Feature extraction from eegs associated with emotions. *Artificial Life and Robotics*, 1(1):15–19, 1997.

[66] Milan Paluš. Nonlinearity in normal human eeg: cycles, temporal asymmetry, nonstationarity and randomness, not chaos. *Biological cybernetics*, 75(5):389–396, 1996.

[67] Sinno Jialin Pan, Ivor W Tsang, James T Kwok, and Qiang Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210, 2010.

[68] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1406–1415, 2019.

[69] Panagiotis C Petrantonakis and Leontios J Hadjileontiadis. Emotion recognition from eeg using higher order crossings. *IEEE Transactions on information Technology in Biomedicine*, 14(2):186–197, 2009.

[70] Sayan Rakshit, Biplab Banerjee, Gemma Roig, and Subhasis Chaudhuri. Unsupervised multi-source domain adaptation driven by deep adversarial ensemble learning. In *German Conference on Pattern Recognition*, pages 485–498. Springer, 2019.

[71] M Rizon, Murugappan M, Raghul Nagarajan, and Sazali Yaacob. Asymmetric ratio and fcm based salient channel selection for human emotion detection using eeg. *WSEAS Transactions on Signal Processing*, 4, 01 2008.

[72] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-assisted Intervention*, pages 234–241. Springer, 2015.

[73] Elham S Salama, Reda A El-Khoribi, Mahmoud E Shoman, and Mohamed A Wahby Shalaby. Eeg-based emotion recognition using 3d convolutional neural networks. *International Journal of Advanced Computer Science and Applications*, 9(8), 2018.

[74] Enver Sangineto, Gloria Zen, Elisa Ricci, and Nicu Sebe. We are not all equal: Personalizing models for facial expression analysis with transductive parameter transfer. In *Proceedings of the 22nd ACM International Conference on Multimedia*, pages 357–366, 2014.

[75] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural computation*, 10(5):1299–1319, 1998.

[76] Li-Chen Shi, Ying-Ying Jiao, and Bao-Liang Lu. Differential entropy feature for eeg-based vigilance estimation. In *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 6627–6630. IEEE, 2013.

[77] Li-Chen Shi and Bao-Liang Lu. Off-line and on-line vigilance estimation based on linear dynamical system and manifold learning. In *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, pages 6587–6590. IEEE, 2010.

[78] Yuan Shi and Fei Sha. Information-theoretical learning of discriminative clusters for unsupervised domain adaptation. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, ICML'12, page 1275–1282, Madison, WI, USA, 2012. Omnipress.

[79] Tengfei Song, Wenming Zheng, Peng Song, and Zhen Cui. Eeg emotion recognition using dynamical graph convolutional neural networks. *IEEE Transactions on Affective Computing*, 11(3):532–541, 2018.

[80] Olga Sourina and Yisi Liu. A fractal-based algorithm of emotion recognition from eeg using arousal-valence model. In *Biosignals*, pages 209–214, 2011.

[81] C.J. Stam. Nonlinear dynamical analysis of eeg and meg: Review of an emerging field. *Clinical Neurophysiology*, 116(10):2266–2301, 2005.

[82] Baochen Sun, Jiashi Feng, and Kate Saenko. Correlation alignment for unsupervised domain adaptation. In *Domain Adaptation in Computer Vision Applications*, pages 153–171. Springer, 2017.

[83] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European Conference on Computer Vision*, pages 443–450. Springer, 2016.

[84] Wei Tao, Chang Li, Rencheng Song, Juan Cheng, Yu Liu, Feng Wan, and Xun Chen. Eeg-based emotion recognition via channel-wise attention and self attention. *IEEE Transactions on Affective Computing*, pages 1–1, 2020.

[85] Hoda Tavakkoli and Ali Motie Nasrabadi. A spherical phase space partitioning based symbolic time series analysis (spsp—stsa) for emotion recognition using eeg signals. *Frontiers in Human Neuroscience*, 16, 2022.

[86] Michel Valstar, Björn Schuller, Kirsty Smith, Florian Eyben, Bihan Jiang, Sanjay Bilakhia, Sebastian Schnieder, Roddy Cowie, and Maja Pantic. Avec 2013: the continuous audio/visual emotion and depression recognition challenge. In *Proceedings of the 3rd ACM International Workshop on Audio/visual Emotion Challenge*, pages 3–10, 2013.

[87] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th International Conference on Machine Learning*, pages 1096–1103, 2008.

[88] Haotian Wang, Wenjing Yang, Zhipeng Lin, and Yue Yu. Tmda: Task-specific multi-source domain adaptation via clustering embedded adversarial training. In *2019 IEEE International Conference on Data Mining (ICDM)*, pages 1372–1377. IEEE, 2019.

[89] David Watson, Lee Anna Clark, and Auke Tellegen. Development and validation of brief measures of positive and negative affect: the panas scales. *Journal of personality and social psychology*, 54(6):1063, 1988.

[90] Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big data*, 3(1):1–40, 2016.

[91] Garrett Wilson and Diane J. Cook. A survey of unsupervised deep domain adaptation. *ACM Trans. Intell. Syst. Technol.*, 11(5), jul 2020.

[92] Kun Xia, Lingfei Deng, Wlodzislaw Duch, and Dongrui Wu. Privacy-preserving domain adaptation for motor imagery-based brain-computer interfaces. *IEEE Transactions on Biomedical Engineering*, 69(11):3365–3376, 2022.

[93] Ruijia Xu, Ziliang Chen, Wangmeng Zuo, Junjie Yan, and Liang Lin. Deep cocktail network: Multi-source unsupervised domain adaptation with category shift. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3964–3973, 2018.

[94] Tong Zhang, Wenming Zheng, Zhen Cui, Yuan Zong, and Yang Li. Spatial–temporal recurrent neural network for emotion recognition. *IEEE Transactions on Cybernetics*, 49(3):839–847, 2019.

[95] Zhi Zhang, Sheng-hua Zhong, and Yan Liu. Ganser: A self-supervised data augmentation framework for eeg-based emotion recognition. *IEEE Transactions on Affective Computing*, 2022.

[96] Han Zhao, Shanghang Zhang, Guanhang Wu, José MF Moura, Joao P Costeira, and Geoffrey J Gordon. Adversarial multiple source domain adaptation. *Advances in Neural Information Processing Systems*, 31, 2018.

[97] Li-Ming Zhao, Xu Yan, and Bao-Liang Lu. Plug-and-play domain adaptation for cross-subject eeg-based emotion recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 863–870, 2021.

[98] Sicheng Zhao, Bo Li, Pengfei Xu, and Kurt Keutzer. Multi-source domain adaptation in the deep learning era: A systematic survey. *arXiv preprint arXiv:2002.12169*, 2020.

[99] Sicheng Zhao, Guangzhi Wang, Shanghang Zhang, Yang Gu, Yaxian Li, Zhichao Song, Pengfei Xu, Runbo Hu, Hua Chai, and Kurt Keutzer. Multi-source distilling domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12975–12983, 2020.

[100] W. Zheng, W. Liu, Y. Lu, B. Lu, and A. Cichocki. Emotionmeter: A multimodal framework for recognizing human emotions. *IEEE Transactions on Cybernetics*, pages 1–13, 2018.

[101] 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*, 7(3):162–175, 2015.

[102] 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*, 7:1–1, 09 2015.

[103] Wei-Long Zheng and Bao-Liang Lu. Personalizing eeg-based affective models with transfer learning. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, page 2732–2738. AAAI Press, 2016.

[104] Wei-Long Zheng, Jia-Yi Zhu, Yong Peng, and Bao-Liang Lu. Eeg-based emotion classification using deep belief networks. In *2014 IEEE international conference on multimedia and expo (ICME)*, pages 1–6. IEEE, 2014.

[105] Peixiang Zhong, Di Wang, and Chunyan Miao. Eeg-based emotion recognition using regularized graph neural networks. *IEEE Transactions on Affective Computing*, 2020.

[106] Yongchun Zhu, Fuzhen Zhuang, and Deqing Wang. Aligning domain-specific distribution and classifier for cross-domain classification from multiple sources. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5989–5996, 2019.

[107] Yukun Zuo, Hantao Yao, and Changsheng Xu. Attention-based multi-source domain adaptation. *IEEE Transactions on Image Processing*, 30:3793–3803, 2021.

Curriculum Vitae

Name:	Amir Hesam Salimnia
Post-Secondary Education and Degrees:	B.Sc. in Electrical Engineering Minor in Computer Engineering 2016 - 2020 University of Tehran Tehran, Iran
Honours and Awards:	Vector Scholarship In Artificial Intelligence 2020 Silver Medal in Iranian National Physics Olympiad 2015 Member of National Iranian Elites Foundation 2015 - Present
Related Work Experience:	Teaching Assistant The University of Western Ontario 2021 - Present Teaching Assistant University of Tehran 2018 - 2021