Deep Temporal Clustering Features for Speech Emotion Recognition

Wei-Cheng Lin[a], Carlos Busso[a,*]

[a]*Department of Electrical and Computer Engineering, The University of Texas at Dallas, 800 W. Campbell Road, Richardson, 75080, TX, USA*

Abstract

Deep clustering is a popular unsupervised technique for feature representation learning. We recently proposed the chunk-based DeepEmoCluster framework for *speech emotion recognition* (SER) to adopt the concept of deep clustering as a novel *semi-supervised learning* (SSL) framework, which achieved improved recognition performances over conventional reconstruction-based approaches. However, the vanilla DeepEmoCluster lacks critical sentence-level temporal information that is useful for SER tasks. This study builds upon the DeepEmoCluster framework, creating a powerful SSL approach that leverages temporal information within a sentence. We propose two sentence-level temporal modeling alternatives using either the *temporal-net* or the *triplet loss* function, resulting in a novel temporal-enhanced DeepEmoCluster framework to capture essential temporal information. The key contribution to achieving this goal is the proposed sentence-level uniform sampling strategy, which preserves the original temporal order of the data for the clustering process. An extra network module (e.g., gated recurrent unit) is utilized for the temporal-net option to encode temporal information across the data chunks. Alternatively, we can impose additional temporal constraints by using the triplet loss function while training the DeepEmoCluster framework, which does not increase model complexity. Our experimental results based on the MSP-Podcast corpus demonstrate that the proposed temporal-enhanced framework significantly outperforms the vanilla DeepEmoCluster framework and other existing SSL approaches in regression tasks for the emotional at-

*Corresponding author

Email addresses: wei-cheng.lin@utdallas.edu (Wei-Cheng Lin), busso@utdallas.edu (Carlos Busso)

tributes arousal, dominance, and valence. The improvements are observed in fully-supervised learning or SSL implementations. Further analyses validate the effectiveness of the proposed temporal modeling, showing (1) high temporal consistency in the cluster assignment, and (2) well-separated *emotional patterns* in the generated clusters.

Keywords: Deep clustering, Temporal modeling, Semi-supervised learning, Speech emotion recognition

---

1. Introduction

Advances in *speech emotion recognition* (SER) have opened new opportunities in *human computer interaction* (HCI), education, surveillance, and healthcare. To facilitate the deployment of SER solutions, the models need to be robust to new domains (Lee et al., 2021). Recently, *deep learning* (DL)-based SER models such as transformers (Tarantino et al., 2019; Goncalves and Busso, 2022; Wagner et al., 2023) or *long short-term memory* (LSTM) (Tao and Liu, 2018) have obtained state-of-the-art recognition performances compared to traditional two-step modeling approaches using handcrafted *high-level descriptors* (HLDs) features (Lin and Wei, 2005; Lee et al., 2011; Lotfian and Busso, 2018, 2019b). By jointly learning the feature extractor and the emotion classifier or regressor, the resulting approach typically leads to more powerful and discriminative feature representations for SER. A sufficient amount of data is the key to efficiently training a DL model. However, it is expensive to collect a large and high-quality human-labeled dataset. Therefore, developing new strategies to utilize unlabeled data while training DL models has become an important research direction.

Various studies have recently proposed leveraging self-supervised learning schemes for universal audio representations. The pre-defined self-supervision paradigm predicts the temporal relation of speech signals (e.g., predicting future frames from past frames (Schneider et al., 2019), or completing the masked frame clusters (Hsu et al., 2021)), which enables the model to take into consideration the overall audio sequence structure without requiring any human labels. This goal can be achieved by encouraging the models to optimize the triplet loss function (e.g., TRILL (Shor et al., 2020)), or a contrastive loss function (e.g., COLA (Saeed et al., 2021) and Wav2Vec (Baevski et al., 2020)). These strategies aim to increase the similarity of the learned representations if they have closer temporal relations. These models are easy to

pre-train on a large corpus (or multi-corpus setting (Baevski et al., 2020)) since they do not need supervision labels. These approaches result in robust general-purpose *deep features* that can be used for different speech tasks. The conventional approach to utilize deep features is to fine-tune these feature representations for the different downstream supervision tasks, such as *automatic speech recognition* (ASR) (Baevski et al., 2020), speaker identification (Saeed et al., 2021) or SER (Chou et al., 2022; Pepino et al., 2021)(Chen and Rudnicky, 2023). The use of deep features further improves the performances of DL-based models.

Unlike self-supervised learning, *semi-supervised learning* (SSL) relies on partial human knowledge (i.e., labeled data) to explore complementary task-specific information from an unlabeled data set to improve the model performance (Chapelle et al., 2006). In the SER field, one of the common approaches is self-labeled training (Triguero et al., 2015) or pseudo-labeling (Choi and Song, 2020; Zhu and Sato, 2021). A trained emotion classifier is used to recognize pseudo-labels with high confidence (i.e., low entropy) on the unlabeled data. The extra pseudo-labeled data is then added to the training set to retrain the classifier (Yarowsky, 1995; Zhu and Sato, 2021). However, this method might suffer from the problem of performance degradation due to mislabeled training samples produced by the pseudo-labeling process. An additional label correction framework is required to alleviate the error accumulation in the pseudo-labeling process (Zhang et al., 2018). Another conventional approach relies on a reconstruction-based network consisting of an encoder-decoder structure such as *autoencoder* (AE) (Deng et al., 2018), *variational autoencoder* (VAE) (Latif et al., 2018) or *ladder networks* (LadderNet) (Parthasarathy and Busso, 2020). These networks are trained to learn the bottleneck hidden embedding that can simultaneously reconstruct the input data and predict emotions, preserving emotional-relevant information with compact latent representation. Since the reconstruction process is unsupervised (i.e., reconstructing the input data itself), the model can adopt unlabeled data in the training process to extract additional prior knowledge of the data distribution. This method exploits unlabeled data without introducing potentially misleading emotion labels, which effectively improves recognition performances with reduced risks. However, reconstruction-based methods do not have strong geometric meaning when learning the latent representation, which might limit the intelligibility of the learned embeddings (e.g., why does a reconstruction task facilitate the SER task?). Also, a reconstruction task can be a simple SSL problem when the train set with labeled

data is large enough, limiting the contributions of the unlabeled data.

Deep clustering is a popular unsupervised technique for the feature representation learning of image classification (Caron et al., 2018; Guo et al., 2017). The training process is typically alternated between the update of network parameters and clustering assignments, leading to competitive performances compared to supervised learning approaches (Caron et al., 2018). Inspired by this concept, we proposed in our preliminary work a semi-supervised DeepEmoCluster framework to learn *deep clustering features* for SER tasks (Lin et al., 2021). The DeepEmoCluster model predicts the emotional task and K-means clustering classes at the same time, constraining the model to maximize the separation between clusters according to emotions. Similar to the AE (Deng et al., 2018) or VAE (Latif et al., 2018) approaches, DeepEmoCluster exploits the unlabeled data without directly introducing pseudo-emotion labels. Specifically, it leverages the cluster assignments obtained by the K-means classifier as the training target (i.e., pseudo-clustering labels) to perform cluster classification, which leads to better recognition performances than other existing reconstruction-based SSL frameworks in SER (Lin et al., 2021). In contrast to AE or VAE, the framework does not require a decoder. However, the original DeepEmoCluster framework learns the feature representation based on chunk-level data (i.e., the sentence is split into small segments referred to as chunks). The approach treats each data chunk as an independent training sample, which neglects the sentence-level temporal structure across consecutive chunks during the clustering process. Temporal modeling is a challenging but critical task when dealing with sequential data such as speech. Therefore, temporal modeling is essential to SER (Lin and Busso, 2022). Various studies have shown superior recognition performances if the model properly captures temporal information (Han et al., 2018; Lin and Busso, 2022; Kim et al., 2017; Ouyang et al., 2017).

This study builds upon the original DeepEmoCluster model, addressing in a principled manner its lack of temporal modeling capability. The novel formulation results in a temporal-enhanced DeepEmoCluster framework. Specifically, the approach modifies the sampling strategy to preserve complete sentence-level temporal information for the training process. This strategy allows us to introduce a *temporal-net* or *triplet loss* function, in addition to the original DeepEmoCluster model, for capturing the temporal relation across data chunks in a sentence. The temporal-net solution includes an additional network module (e.g., gated recurrent unit) for processing cross-chunk temporal information in the sentence. The triplet loss

function regularizes the model to increase the similarity of chunk-level representations that are closer in the temporal dimension without compromising extra model complexity during inferences.

We evaluate the proposed temporal-enhanced DeepEmoCluster framework with the MSP-Podcast corpus (Lotfian and Busso, 2019a), where the goal is to predict the emotional attributes for arousal, dominance, and valence. Our experimental results show that the proposed framework significantly outperforms the vanilla DeepEmoCluster and other reconstruction-based SSL baselines for all emotional attributes. If different temporal-net options are considered, we achieve the best *concordance correlation coefficient* (CCC) performances for arousal (0.5726), valence (0.1674), and dominance (0.4837) using mel-spectrogram as the inputs. We include analyses to understand the properties of the proposed temporal-enhanced DeepEmoCluster framework. First, we study the clusters. The analysis indicates that our proposed approach obtains high consistency in the clustering assignments for temporally close data chunks, which validates the effectiveness of the temporal modeling. Second, we obtain t-SNE (van der Maaten and Hinton, 2008) plots of the model embeddings. The results illustrate that the clusters are learned and grouped by considering the emotional information in the sentences (e.g., high, middle, and low arousal), improving the emotional dependency in the generated clusters. We also analyze the hyper-parameters of the framework to justify the choice of parameters used in the study. In summary, the main contributions of this study are:

- We propose a novel semi-supervised temporal-enhanced DeepEmoCluster framework, which achieves the best emotion attribute recognition performances among other existing SSL frameworks in SER.
- We find that the effectiveness of the temporal modeling approach can lead to high temporal consistency and well-separable emotional clusters, which lead to better SER performance.

The rest of the paper is organized as follows. Section 2 describes the research studies that are relevant to this work. Section 3 presents the proposed methodology, providing detailed explanations of the temporal-enhanced DeepEmoCluster framework. Section 4 describes the experimental settings, including the speech-emotional database, acoustic features, baseline models, and implementation details used to train and evaluate our approach. Section 5 discusses the experimental results and analysis, including the comprehensive performance comparison, the temporal consistency, the t-SNE

analysis, and the hyper-parameter analysis. Finally, Section 6 presents the concluding remarks and future research directions opened by this study.

2. Related Work

The temporal-enhanced DeepEmoCluster is a novel semi-supervised SER modeling framework that combines the techniques of deep clustering and temporal modeling. This section discusses previous studies proposed in the SER field working on clustering approaches (Sec. 2.1) and temporal modeling (Sec. 2.2), respectively.

2.1. Clustering Approaches in SER

In the SER field, the clustering technique is typically applied as a data preprocessing step to identify *keyframes* (or *keysegment*) when using *convolutional neural networks* (CNNs) or LSTM (Hajarolasvadi and Demirel, 2019; Mustaqeem et al., 2020). The K-means clustering algorithm is adopted for grouping similar feature frames, where the representative frame (keyframe) is determined by the distance to the cluster centroids (Mustaqeem et al., 2020). This approach can effectively remove redundant frames in the utterance (i.e., frames having similar acoustic contents due to the overlaps between the analysis windows), which significantly reduces the computational cost while still preserving the model performance (Hajarolasvadi and Demirel, 2019). Some studies implicitly learn the feature clusters by combining different network modules. Jalal et al. (2019) proposed a framework that utilizes convolution capsules (Sabour et al., 2017) to summarize different temporal information after a feature encoding layer implemented with *bidirectional long short-term memory* (Bi-LSTM). Their t-SNE analysis demonstrated that the distinct temporal features (or clusters) are learned by different capsules, leading to improvements in SER performance.

In contrast to conventional clustering techniques, deep clustering feature representation learning explicitly treats the clustering assignments as the classification target of a self-supervision task (Caron et al., 2018; Van Gansbeke et al., 2020; Asano et al., 2020; Lin et al., 2021; Hsu et al., 2021). The training of the clustering task (i.e., pseudo-label assignment) and the main classification task (i.e., update of model parameters) are alternating until the model converges. The major challenge of the framework is to prevent the model from learning the trivial solution of assigning all samples to a single cluster. This issue is prevented with ad-hoc techniques such as entropy

regularization (Van Gansbeke et al., 2020; Asano et al., 2020) or uniform sampling (Caron et al., 2018). The entropy regularization encourages the model to maximize an additional entropy term in the loss function, which aims to spread the predictions of cluster assignment uniformly across the classes. The uniform sampling strategy balances the label distribution over classes (i.e., cluster assignments) by upsampling and downsampling the data points contained in the clusters with fewer and more samples, respectively. This approach is equivalent to weighting the contribution of the loss function based on the inverse size of each cluster set. The DeepEmoCluster framework (Lin et al., 2021) adopted a similar deep clustering approach implemented with a uniform sampling strategy, forming a semi-supervised framework to learn an emotional cluster representation for speech. However, the uniform sampling strategy of the DeepEmoCluster framework is based on chunk-level data, which does not preserve the temporal order of the chunks in the sentence. In fact, data chunks are randomly sampled without considering their original temporal order in the sentence, which makes the introduction of an additional sentence-level temporal modeling module infeasible. To overcome this issue, we modify the sampling strategy to preserve the temporal order of the data chunks in the sentence to capture and model sentence-level temporal information, extending the framework to temporal-enhanced DeepEmoCluster. We describe the sampling process in Section 3.3.

2.2. Temporal Modeling in SER

Emotion is not uniformly conveyed across time (Lin and Busso, 2023; Busso and Narayanan, 2006; Lotfian and Busso, 2019b). Therefore, SER needs to capture and summarize dynamic changes in the emotions conveyed in the acoustic frames across time. Conventionally, this goal can be achieved with a *recurrent neural network* (RNN)-based model (e.g., LSTM) using an attention mechanism, where the recurrent units encode temporal information, and the jointly trainable attention weights help the model to emphasize or omit frames depending on their emotional contents (Mirsamadi et al., 2017; Huang and Narayanan, 2017). However, frame-level features contain redundant information and typically result in long sequence input (e.g., a few seconds of a sentence create hundreds of frames). Long sequences cause degeneration of RNN-based models due to the vanishing gradient problem (Trinh et al., 2018). They also impose an increase in the computational complexity of the model while using transformer-based models (Kitaev et al., 2020). To resolve the problem, Lin and Busso (2022, 2020) proposed a

framework to split sentences into a fixed number of data chunks with the same duration, performing chunk-level attention instead of frame-level attention. This approach significantly reduces the complexity of the attention models and improves overall SER performance. Various similar approaches have also been proposed to avoid directly modeling long sequence inputs by splitting a sentence into smaller data chunks (or segments) for building chunk-level models (Tarantino et al., 2019; Sahoo et al., 2019). During the training stage, data chunks are regarded as independent training samples sharing the same sentence-level label (Han et al., 2014). After the training, the model relies on statistical decision pooling such as majority voting rule or mean pooling layer to aggregate the final sentence-level prediction (Bitouk et al., 2010). The proposed temporal-enhanced DeepEmoCluster framework follows the same chunk-level modeling scheme, capturing sentence-level temporal information with either the temporal-net or triplet loss function.

3. Proposed Methodology

Our approach builds upon the vanilla DeepEmoCluster framework presented in our preliminary study (Lin et al., 2021), enabling the model to capture in a principled way temporal information. This section describes the preparation of data chunks used in this study (Sec. 3.1), and the original DeepEmoCluster framework (Sec. 3.2). Then, we present the two major contributions proposed in this study to build the temporal-enhanced DeepEmoCluster: 1) the uniform sampling strategy to prevent the trivial clustering solution while preserving the sentence-level temporal order between data chunks (Sec. 3.3), and 2) the model constraints using either the temporal-net or the triplet loss function to capture the sentence-level temporal information (Sec. 3.4).

3.1. Dynamic Chunk-Based Segmentation

The input of the DeepEmoCluster framework is the data chunks split by using the dynamic chunk segmentation approach proposed by Lin and Busso (2022). We illustrate the segmentation process in Figure 1(a). This approach segments a sentence-level data $\mathbf{X}$ with an arbitrary length into a fixed number of data chunks $\{x_1, x_2, \ldots, x_C\}$ with the same size, regardless of its duration. The variable C is the number of chunks, and w_c is the duration of the chunks. This goal is achieved by dynamically adjusting the shifting size Δc_i between data chunks according to their sentence duration. Equation 1 provides the

Algorithm 1: The complete training procedure for the semi-supervised DeepEmoCluster model.

Initialize:
$\Phi(\cdot)$: the feature extractor
$f_{cltr}(\cdot)$: the cluster classifier
$f_{emo}(\cdot)$: the emotion regressor
$\mathbf{x}_{iU}$: *i*-th batch data chunks from the unlabeled set $\mathbb{U}$
$\mathbf{x}_{iL}$: *i*-th batch data chunks from the labeled set $\mathbb{L}$
$\mathbf{y}_{emo,iL}$: *i*-th batch emotion labels from set $\mathbb{L}$
Main:
for *epoch = 1, 2, . . . , E* **do**
    % pseudo-labeling for $\mathbb{U}$
    $\mathbf{h}_{iU} = \Phi(\mathbf{x}_{iU})$, $\forall$ i
    $\mathbf{y}_{cltr,iU} \leftarrow Kmeans(\mathbf{h}_{iU})$, $\forall$ i
    % Stage I: self-supervised path
    for *i = 1, 2, . . . I* **do**
        $\hat{\mathbf{y}}_{cltr,iU} = f_{cltr}(\Phi(\mathbf{x}_{iU}))$
        $\mathcal{J}$ = CE($\hat{\mathbf{y}}_{cltr,iU}$, $\mathbf{y}_{cltr,iU}$)
        update $\Phi(\cdot)$ *and* $f_{cltr}(\cdot)$ *by* $\mathcal{J}$
    end
    % pseudo-labeling for $\mathbb{L}$
    $\mathbf{h}_{iL} = \Phi(\mathbf{x}_{iL})$, $\forall$ i
    $\mathbf{y}_{cltr,iL} \leftarrow Kmeans(\mathbf{h}_{iL})$, $\forall$ i
    % Stage II: jointly optimized path
    for *i = 1, 2, . . . I* **do**
        $\hat{\mathbf{y}}_{cltr,iL} = f_{cltr}(\Phi(\mathbf{x}_{iL}))$
        $\hat{\mathbf{y}}_{emo,iL} = f_{emo}(\Phi(\mathbf{x}_{iL}))$
        $\mathcal{J}$ = CE($\hat{\mathbf{y}}_{cltr,iL}$, $\mathbf{y}_{cltr,iL}$) +
            [1 - CCC($\hat{\mathbf{y}}_{emo,iL}$, $\mathbf{y}_{emo,iL}$)]
        update $\Phi(\cdot)$, $f_{cltr}(\cdot)$ *and* $f_{emo}(\cdot)$ *by* $\mathcal{J}$
    end
end

key formula for this segmentation. The shifting size Δc_i is a function of the sentence duration T_i, where the overlap between consecutive chunks is

smaller for longer sentences. The number C is determined by Equation 2, which requires two predefined parameters: $T_{\max}$, maximum duration of a sentence in the dataset, and w_c, the desired chunk window length.

$$\Delta c_i = \frac{T_i - w_c}{C - 1} \tag{1}$$

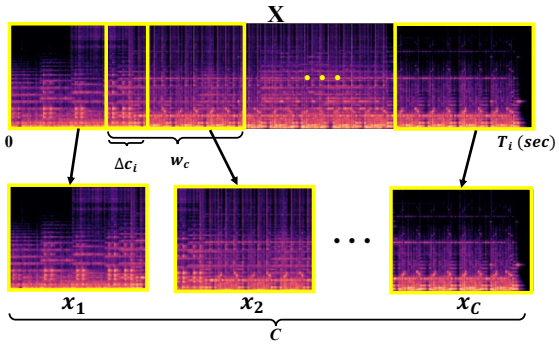
$$C = \left\lceil \frac{T_{\max}}{w_c} \right\rceil \tag{2}$$

3.2. The DeepEmoCluster Framework

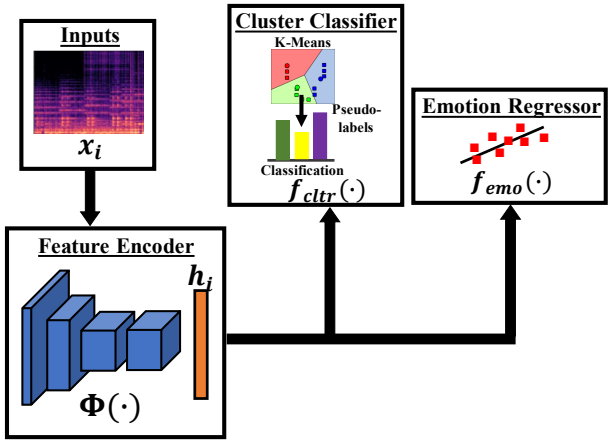
The DeepEmoCluster framework uses the classification of assignments of data-generated clusters as an auxiliary task. The model consists of three subnetwork modules, illustrated in Figure 1(b): feature encoder, cluster classification, and emotion regression. The feature encoder $\Phi(\cdot)$ maps the input data x_i into the hidden representation h_i, which is the input of the cluster classification and emotion regression blocks. The cluster classifier $f_{cltr}(\cdot)$, where *cltr* stands for *cluster*, predicts the clustering assignments where the labels are directly obtained from the data. The emotion regressor $f_{emo}(\cdot)$, where *emo* stands for *emotion*, predicts the score of the emotional attributes (i.e., arousal, dominance, and valence). The DeepEmoCluster framework trains the model by alternating between the clustering and classification processes. Algorithm 1 shows the complete training process for the DeepEmoCluster framework. In every training epoch, the K-means clustering algorithm is first implemented using the hidden outputs of $\Phi(\cdot)$ (i.e., feature encoder) to determine the pseudo-labels for the cluster classifier. Then, the model is jointly trained to optimize a multitask loss function that combines the *cross-entropy* (CE) loss for the cluster classifier ($\mathcal{J}_{cltr} = CE$), and the *concordance correlation coefficient* (CCC) function for the emotion regressor ($\mathcal{J}_{emo} = 1 - CCC$). Equation 3 shows the loss function. For the data without emotion labels (i.e., unlabeled set $\mathbb{U}$), the model only optimizes the CE loss and freezes the update of the emotion regressor weights ($f_{emo}(\cdot)$). The iteration between unsupervised and supervised stages results in an SSL framework.

$$\mathcal{J} = \mathcal{J}_{cltr} + \mathcal{J}_{emo} = CE + (1 - CCC) \tag{3}$$

The DeepEmoCluster framework utilizes the chunk-level modeling scheme



(a) Segmentation of Data Chunks



(b) Vanilla DeepEmoCluster Framework

Figure 1: The vanilla DeepEmoCluster framework proposed in Lin et al. (2021), which adopts a dynamic chunk segmentation approach to split the data into chunks, which are used as the input of the model.

(Sec. 3.1) to train the model. In our original implementation presented in our preliminary study (Lin and Busso, 2022), the DeepEmoCluster framework treated each data chunk as an independent training sample that shares the same sentence-level label assigned to the sentence. The results of the chunk-level predictions were then aggregated to obtain a sentence-level prediction using the mean pooling operation. We refer to this model as the vanilla DeepEmoCluster framework.

Since the vanilla DeepEmoCluster framework treats the data chunks as independent training samples, its uniform sampling strategy is straightfor-

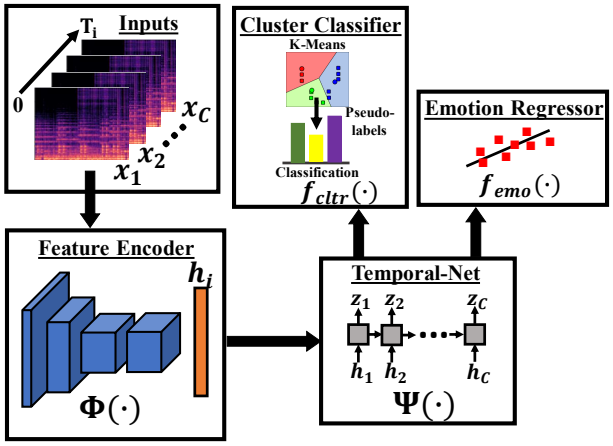
ward. Assume the training set contains N sentences, each sentence is split into C data chunks resulting in a total of $N \times C$ samples for training the chunk-level model. Then, we create cluster sets with the chunks assigned to the same clustering group. We randomly sample a fixed size of $\frac{N \times C}{Q}$ data points for each cluster, where Q represents the total number of clusters. If the cluster set does not have enough data points, we sample with replacement to reach the target size (i.e., upsampling). This sampling approach guarantees a balanced class distribution during training, preventing the model from converging to a trivial solution that assigns all samples to a few clusters (Caron et al., 2018). However, the random sampling process loses the temporal order between data chunks, making the framework unable to capture sentence-level temporal information.

3.3. Sentence-level Uniform Sampling

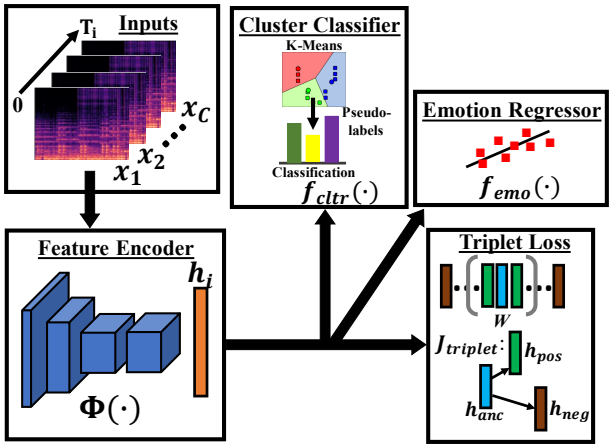
This study modifies the sampling strategy to preserve the sentence-level temporal order, while still keeping a balanced cluster class distribution. The proposed sampling approach is implemented at the sentence level instead of at the chunk-level. The key idea is to assign each sentence a sampling weight. First, we compute a weight for each chunk, which is derived from the chunk-level cluster distribution. We use the inverse size of each cluster set as the weight of the cluster. For instance, if a cluster set has a smaller size (i.e., minority class), the sample weight of the data chunks coming from this cluster set is larger (i.e., upsampling). Then, the weights across the chunks of a sentence are aggregated. The sentence-level sampling weight is simply the summation of these chunk-level weights for the sentence, following the clustering assignments. This sentence-level sampling approach preserves the temporal order of the data chunks since the sentence always has the C chunks in order (see the input block for the model in Fig. 2). It also balances the class distribution with the aforementioned cluster weighting approach. Notice that the approach does not guarantee having a strictly uniform distribution across classes due to the stochastic nature of the sampling process. However, studies have shown that a moderately relaxed non-uniform class distribution is closer to practical scenarios and does not result in non-trivial clustering solutions (Asano et al., 2020).

3.4. Temporal Modeling for Deep Clustering Framework

With the sentence-level sampling strategy, we obtain the full C data chunks of the sentence, arranged by their temporal orders. Therefore, we



(a) Temporal-Net DeepEmoCluster



(b) Triplet DeepEmoCluster

Figure 2: The two proposed sentence-level temporal modeling approaches to achieve the temporal-enhanced DeepEmoCluster framework. Both methods rely on the modified sentence-level uniform sampling strategy (Sec. 3.3) to incorporate either an additional temporal-net module or the triplet loss function in the training pipeline.

can perform additional temporal modeling across data chunks in the deep clustering process. We explore two approaches to encode temporal information in the DeepEmoCluster model.

Temporal-Net DeepEmoCluster (Fig. 2(a)): We can directly add an extra network module to capture the temporal information. More specifically,

the feature encoder $\Phi(\cdot)$ extracts the chunk-level representations h_i of the input data chunks x_i (Eq. 4). Then, we introduce the temporal-net $\Psi(\cdot)$ to encode the temporal information across these C consecutive representations, resulting in the hidden outputs z_i (Eq. 5) for the upcoming cluster classifier $f_{cltr}(\cdot)$ and emotion regressor $f_{emo}(\cdot)$. There are various options to implement the temporal-net, since the goal of the model is to just process cross-chunk information. One option is the *gated recurrent unit* (GRU), which dynamically tracks temporal changes with recurrent connections. Another option is the 2D-CNN, which extracts spatial-temporal features. We can also implement this network with the transformer architecture, which uses positional encoding to specify the temporal order of the sequence for self-attention modeling. We explore these three options to implement the temporal-net. We refer to these networks as *Temp-GRU*, *Temp-CNN*, and *Temp-Trans*, respectively. Notice that the complexity of temporal-net does not need to be large, since the feature encoder $\Phi(\cdot)$ has summarized the frame-level features into chunk-level representations, which significantly reduces the temporal length (i.e., one representation per chunk, instead of one representation per frame). We describe the detailed temporal-net architectures in Section 4.3.

$$h_i = \Phi(x_i), i = 1, 2, \ldots, C \tag{4}$$

$$\{z_1, z_2, \ldots, z_C\} = \Psi(\{h_1, h_2, \ldots, h_C\}) \tag{5}$$

Triplet DeepEmoCluster (Fig. 2(b)): Another alternative way to efficiently learn temporal information without adding extra model complexity (as done with the temporal-net module) is by introducing the triplet loss function during training. Similar to conventional approaches (Shor et al., 2020; Saeed et al., 2021; Baevski et al., 2020), we design the triplets by finding an anchor, a positive sample, and a negative sample based on the temporal proximity. Specifically, the positive sample is determined by the samples that are temporally close to the anchor within a specified window length W, and the negative sample is determined by the samples outside of that window, as shown in the triplet loss block in Figure 2(b). The triplet loss function encourages the model to learn similar representations for temporally close samples and dissimilar representations for distant samples. Note that

our triplet loss function is computed at the chunk level (i.e., h_i from Eq. 4) rather than at the raw acoustic frame level. This implementation significantly reduces the number of triplet pairs needed for training. Equation 6 indicates the loss function,

$$\mathcal{J}_{triplet} = \max\{\mathrm{d}(h_{anc}, h_{pos}) - \mathrm{d}(h_{anc}, h_{neg}) + \delta, 0\} \tag{6}$$

where $\mathrm{d}(\cdot)$ is the L2 distance between two vectors, and δ is a nonnegative margin set as a hyper-parameter. Finally, the triplet loss term is added to the original DeepEmoCluster loss (Eq. 3) with a weighting control factor γ, resulting in the overall loss function in Equation 7 for joint optimization. Therefore, the model is trained to maximize the separation between clusters based on emotion contents with additional temporal consistency constraints. We refer to this approach as *Temp-Triplet*.

$$\mathcal{J}_{total} = \mathcal{J} + \gamma \times \mathcal{J}_{triplet} \tag{7}$$

4. Experimental Settings

4.1. The MSP-Podcast Corpus

We use version 1.8 of MSP-Podcast (Lotfian and Busso, 2019a) corpus to evaluate our approach. The dataset collects spontaneous emotional speech segments from various publicly available audio-sharing websites. The original full podcast conversations are segmented by a series of automatic steps such as speaker diarization, and noise and music detection to obtain recordings containing a single speaker, with high *signal-to-noise ratios* (SNRs), and no background music. We rely on the emotional retrieval strategy proposed by Mariooryad et al. (2014) to prioritize samples to be annotated with emotional labels that are likely to have emotional content. These segments are processed to have a duration ranging from 2.75 to 11 seconds. Every segment is rated by at least five different annotators for categorical descriptors (e.g., anger, happiness, and sadness) and emotional attributes (arousal, valence, and dominance). This study only focuses on attribute-based descriptors for arousal (calm to active), valence (negative to positive), and dominance (weak to strong), where the labels are sentence-level scores ranging from 1 to 7. The ground truth is the average of the scores provided by the evaluators to

Table 1: The VGG-16 feature encoder model used in this study.

Layer	Channels/Nodes	Kernel	Stride	Activation
Input	1	N/A	N/A	N/A
CNN-block (×2)	32	(3, 3)	1	ReLU
CNN-block (×2)	64	(3, 3)	2	ReLU
CNN-block (×3)	128	(3, 3)	2	ReLU
Flatten	N/A	N/A	N/A	N/A
Linear	256	N/A	N/A	ReLU
Dropout	$p = 0.5$	N/A	N/A	N/A

a speech recording. We formulate the SER task as a regression problem, similar to previous studies (Abdelwahab and Busso, 2018, 2019).

The dataset defines the partitions for the train (44,879 speech turns), development (7,800 speech turns), and test (15,326 speech turns) sets. The sets aim to have speaker-independent partitions where data from one speaker is only included in one set. We use the train set to build our models, optimizing performance and hyper-parameters on the development set. The final SER evaluation results are based on the test set. We utilize segments that have not been retrieved and annotated in the corpus to create the unlabeled pool (over 600K speech turns). If the semi-supervised learning scheme is used in the experiment, we randomly sample a subset from this pool to serve as the additional unlabeled set.

4.2. Acoustic Features

We use the same acoustic feature set used in the original DeepEmoCluster model (Lin et al., 2021), which corresponds to a 128D Mel-spectrogram. We extract this feature vector using a 32 ms window size with a 16 ms window hop size (i.e., 50% overlaps). We perform the z-normalization approach on these features, where the normalization parameters (mean and standard deviation) are estimated over the entire train set.

4.3. Hyper-parameter, Model Implementation and Baseline Settings

We follow the original settings of the DeepEmoCluster framework (Lin et al., 2021). We split every sentence into C=11 data chunks using the chunk size w_c=1 sec, since the maximum sentence duration of the dataset is $T_{\max} =$ 11 secs. The model architecture of $\Phi(\cdot)$ uses the VGG-16 structure (Simonyan

and Zisserman, 2015). Table 1 shows the details of the VGG-16 model. We use *stochastic gradient descent* (SGD) with a learning rate set to 0.001 for the VGG-16 encoder. We use Adam with a learning rate set to 0.0005 for the output layers. The $f_{cltr}(\cdot)$ and $f_{emo}(\cdot)$ networks are implemented with two *fully connected* (FC) output layers, each of them implemented with 256-nodes using the *rectified linear unit* (ReLU) as the activation function.

The three alternative approaches for the temporal-net $\Psi(\cdot)$ are implemented with different settings. The *Temp-GRU* model uses a single bi-directional GRU layer using the tanh activation. The *Temp-CNN* model has two 2D-CNN blocks of convolutional weights with 16 channels, each of them with a 5×5 kernel. We also use ReLU as the activation function and batch normalization layers. The *Temp-Trans* model uses a single-head transformer encoder with positional encoding. The hidden nodes of the network layers are set to 256, implemented with dropout with a rate set to 0.3.

We set the window length to W=4 for the *Temp-Triplet* approach, which is used to define positive and negative samples. Therefore, the positive samples are located within ± 2 chunks of the anchor. The margin (δ) for the triplet loss function is set to 1 (Eq. 6). To efficiently train the model, we randomly sample 10 triplets per sentence and re-sample another 10 pairs for each training iteration instead of considering the full set of triplets. Since we do not observe significant performance differences when we increase the number of triplets during training, we conclude that the model can infer temporal relationships based on this setting. We set the weighting factor γ=1 for the triplet loss function (Eq. 7). The cluster number (Q) for the K-means algorithm is fixed to 10 for the DeepEmoCluster-based frameworks. We randomly sample 15K speaking segments from the unlabeled pool (Sec. 4.1) as the unlabeled set $\mathbb{U}$ for the SSL experiments. We present a hyper-parameter analysis for γ, Q, and the size of $\mathbb{U}$ in Section 5.4. For a fair comparison, all the models are trained with the same training and evaluation configurations using CCC as the performance metric, a batch size of 64, and a maximum of 30 epochs. We save the best models with an early stopping criterion based on the development set performance (i.e., CCC). The final reported model results are averaged over five running trials with different network initializations. We test on four randomly split sub-test sets from the original test set. This setting results in 20 values (i.e., 4 subsets $\times$ 5 trials), which are used to statistically analyze and compare the results using a two-tailed t-test, asserting statistical significance if p-value < 0.05. The models are implemented in PyTorch.

For the baseline models, we consider the vanilla DeepEmoCluster (Lin et al., 2021). In addition, we also implement conventional reconstruction-based SSL frameworks for SER using AE, VAE, and LadderNet. These models have an encoder-decoder architecture with different input reconstruction strategies. AE directly reconstructs the input data from the bottleneck embeddings using the *mean square error* (MSE) loss. Deng et al. (2018) used this strategy for SER tasks. We refer to this baseline as *CNN-AE*. VAE utilizes reparametrized latent vectors, sampling from learnable Gaussian distributions to reconstruct the input vector. The approach uses the *evidence lower bound* (ELBO) as a cost function. Latif et al. (2018) used VAE for SER tasks. We refer to this baseline as *CNN-VAE*. LadderNet is similar to the AE approach, but it introduces random noise perturbations to intermediate encoder layers, which imposes additional denoising constraints for the reconstruction process. They also have lateral connections across intermediate representations between the encoder and the decoder. Several studies have used LadderNet for SER tasks (Parthasarathy and Busso, 2020; Huang et al., 2018; Parthasarathy and Busso, 2018; Goncalves and Busso, 2023; Leem et al., 2021; Reddy Naini et al., 2023). We refer to this baseline as *CNN-LadderNet*. All these models are attached with an emotion regressor to the bottleneck layer for predicting the emotional attribute score. We implement these models to have the same VGG-16 feature encoder $\Phi(\cdot)$ structure described in Table 1. The decoder layers $\Phi^{-1}(\cdot)$ have the reverse structure implemented with an upsampling process using a transposed convolution operator. These baselines are ideal to have a fair comparison with the proposed model since they have the same model complexity and predicting pipeline during the inference stage (i.e., input data only passes through the feature encoder and emotion regressor to produce recognition results). We also compare our approach with a standard supervised regression model, referred to as *CNN-regressor*. This model serves as a naïve baseline that does not include extra self-supervision tasks for leveraging unlabeled data (i.e., input reconstruction or cluster classification).

5. Experimental Results

5.1. Performance Comparison

Tables 2 and 3 summarize the performance comparison of different approaches implemented as a *fully-supervised learning* (FSL) problem (i.e., training with labeled data) and SSL problem (i.e., training with labeled

Table 2: Summary of FSL results on the test set of the MSP-Podcast corpus. The symbol $*$ indicates that the results of the proposed approaches are significantly better than the results achieved by the reconstruction-based baselines (i.e., the CNN-AE, CNN-VAE, and CNN-LadderNet). The symbol † indicates that the results for the proposed approaches are significantly better than all the baselines, including the vanilla DeepEmoCluster (two-tailed t-test, p-value < 0.05). Bold values indicate the best performance for each attribute.

Approach (FSL)	**Aro.-CCC**	**Val.-CCC**	**Dom.-CCC**
CNN-regressor	0.5151	0.1222	0.4334
CNN-AE(Deng et al., 2018)	0.5114	0.1302	0.4226
CNN-VAE(Latif et al., 2018)	0.5155	0.1465	0.4252
CNN-LadderNet(Parthasarathy and Busso, 2020)	0.5212	0.1184	0.4257
vanilla DeepEmoCluster(Lin et al., 2021)	0.5463	0.1260	0.4618
Temp-GRU	$\mathbf{0.5722}^{\dagger *}$	$0.1573^{\dagger *}$	0.4656^{*}
Temp-CNN	$0.5589^{\dagger *}$	0.1514	$\mathbf{0.4705}^{\dagger *}$
Temp-Trans	$0.5636^{\dagger *}$	$\mathbf{0.1586}^{\dagger *}$	0.4638^{*}
Temp-Triplet	0.5515^{*}	0.1508	$0.4701^{\dagger *}$

and unlabeled data), respectively. The unlabeled set has 15K speaking turns. The symbol $*$ indicates that the proposed approach is significantly better than the reconstruction-based baselines (i.e., CNN-AE, CNN-VAE, and CNN-LadderNet). The symbol † indicates that our approach is significantly better than all the baselines including the vanilla DeepEmoCluster approach.

The implementations of the proposed temporal-enhanced DeepEmoCluster framework systematically obtain significantly better performances than the baselines under FSL or SSL training strategies. With different implementations of the proposed approach, we obtain the best CCC performance for arousal (0.5726), valence (0.1674), and dominance (0.4837). More importantly, if we compare the results of the Temp-Triplet model and the vanilla DeepEmoCluster using an FSL strategy (Table 2), we can see that the critical role leading to the improvements is the sentence-level temporal modeling rather than simply increasing the model complexity (i.e., they have the same model architecture, differing only on the triplet loss function added in the Temp-Triplet model during the training stage and the sentence-level uniform sampling strategy).

We observe a general trend while considering additional unlabeled data leading to better performance (i.e., Table 3), which shows the benefit of using SSL frameworks. However, the improvements in the reconstruction-

Table 3: Summary of SSL-15K results on the test set of the MSP-Podcast corpus. The symbol $*$ indicates that the results of the proposed approaches are significantly better than the results achieved by the reconstruction-based baselines (i.e., the CNN-AE, CNN-VAE, and CNN-LadderNet). The symbol † indicates that the results for the proposed approaches are significantly better than all the baselines, including the vanilla DeepEmoCluster (two-tailed t-test, p-value < 0.05). Bold values indicate the best performance for each attribute.

Approach (SSL-15K)	**Aro.-CCC**	**Val.-CCC**	**Dom.-CCC**
CNN-AE(Deng et al., 2018)	0.5051	0.1362	0.4248
CNN-VAE(Latif et al., 2018)	0.5306	0.1260	0.4429
CNN-LadderNet(Parthasarathy and Busso, 2020)	0.5201	0.1428	0.4362
vanilla DeepEmoCluster(Lin et al., 2021)	0.5553	0.1530	0.4734
Temp-GRU	$0.5660^{\dagger *}$	0.1576^{*}	$\mathbf{0.4837}^{\dagger *}$
Temp-CNN	$0.5650^{\dagger *}$	0.1523^{*}	0.4678^{*}
Temp-Trans	$\mathbf{0.5726}^{\dagger *}$	$\mathbf{0.1674}^{\dagger *}$	0.4763^{*}
Temp-Triplet	0.5581^{*}	0.1535^{*}	0.4627^{*}

based baselines are relatively limited. The average relative performance improvements across emotional attributes from FSL to SSL is 2.2% for reconstruction-based approaches (CNN-AE, CNN-VAE, CNN-LadderNet), and 2.7% for cluster-based approaches (DeepEmoCluster, Temp-GRU, Temp-CNN, Temp-Trans, Temp-Triplet). A potential explanation for this result is that the input reconstruction task becomes trivial if we have sufficient training data (i.e., MSP-Podcast corpus v1.8 has more than 100 hours of audio data). The decoder may already be well-trained with the labeled set, diminishing the role of the unlabeled data. In contrast, the proposed frameworks utilize a harder cluster classification task, grouping and separating deep features based on different acoustic conditions (e.g., speaker traits or microphone settings). Studies have shown that a harder self-supervision or pretraining task can increase the model regularization and lead to better generalization performance (Zhang et al., 2021; Wang et al., 2021; Li et al., 2020). Our approach provides a favorable way to leverage unlabeled data. As an aside, we notice that it is challenging to predict valence using spectrogram features. A similar performance trend has been reported in other studies (Yan et al., 2022; Wagner et al., 2023). The proposed framework can easily adopt other acoustic features such as the Wav2Vec (Schneider et al., 2019) or low-level descriptors (Schuller et al., 2013) to further increase recognition performances.

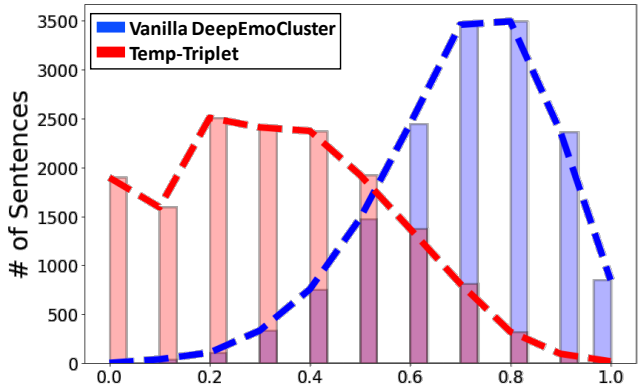
The Temp-Triplet is the most efficient method to incorporate sentence-

level temporal modeling from our implementation. For example, it is approximately 33% more efficient than the Temp-Trans model in terms of the number of parameters. In fact, it has the same model architecture as the vanilla DeepEmoCluster during inference. The differences between these two approaches are solely coming from the training stage, with the addition of the triplet loss function (Sec. 3.4), and the sentence-level uniform sampling (Sec. 3.3). Therefore, it is straightforward and fair to analyze the Temp-Triplet and the vanilla DeepEmoCluster model to obtain further insights into the temporal modeling. By comparing the Temp-Triplet and vanilla DeepEmoCluster methods, Section 5.2 focuses the analysis on temporal consistency, and Section 5.3 focuses the analysis on clustering representation.

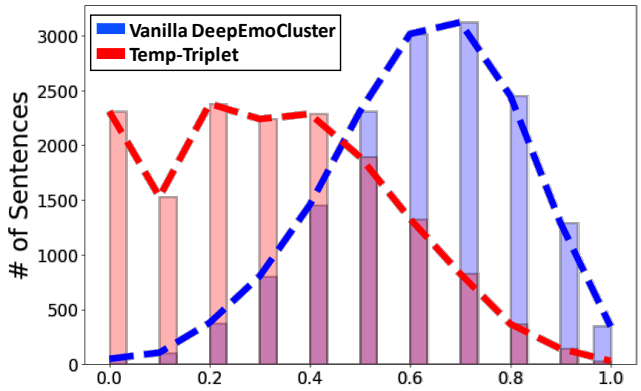
5.2. Analysis of Temporal Consistency

It is expected that nearby data chunks should be assigned to similar clusters since they contain partial overlaps (Eq. 1) and are temporally close to each other. We refer to this property as *temporal consistency* in the clustering process. A strength of our proposed approach is the improved temporal consistency in the assignment of clusters. As mentioned in the previous section, it is straightforward to validate the effectiveness of the temporal modeling by comparing the Temp-Triplet and vanilla DeepEmoCluster models, since they have the same architecture.

We validate the effectiveness of the temporal modeling from the chunk-level clustering assignment perspective. Notice that the clustering assignment is based on the hidden chunk-level representations h_i (Eq. 4). We measure the cluster transition between data chunks within a sentence to verify this property. We calculate this metric as the ratio between the chunk-level cluster assignment transitions within a sentence and $C - 1$, where C is the total number of chunks. In this metric, the numerator increases by one every time two consecutive chunks are assigned to different clusters. A higher ratio indicates a lower temporal consistency since the chunk-level cluster assignments keep jumping across chunks within a sentence. Figures 3(a) and 3(b) illustrate the analysis results based on the test set of the MSP-Podcast corpus under FSL and SSL-15K setups, respectively. As we can see, the vanilla DeepEmoCluster approach always shows a high transition ratio (blue bars are located on high values of this ratio), demonstrating low temporal consistency due to the lack of proper temporal modeling. In contrast, the proposed Temp-Triplet model consistently obtains better temporal consistency (red bars are located on low values of the ratio), validating the



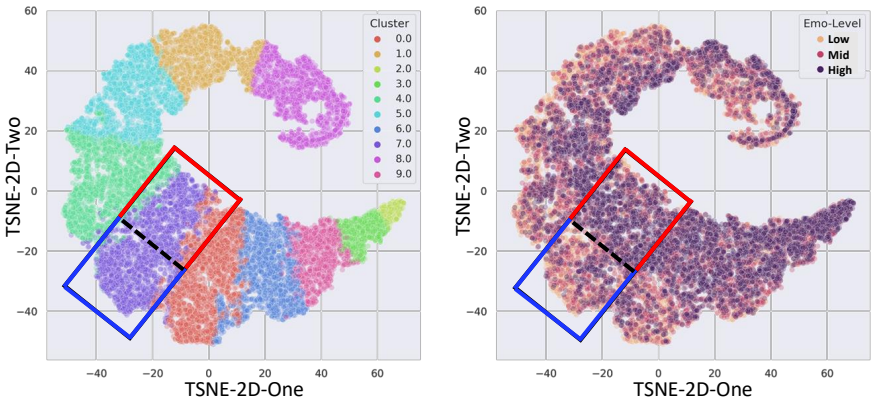
(a) Fully-Supervised Learning (FSL)



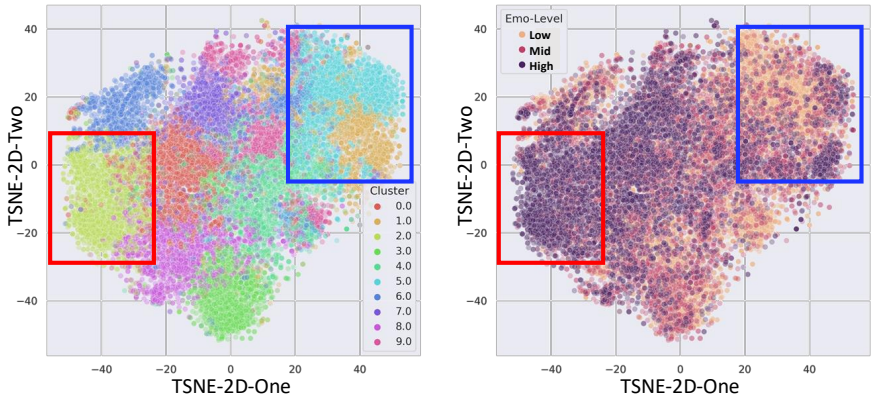
(b) Semi-Supervised Learning (SSL-15K)

Figure 3: The transition ratio of the chunk-level cluster assignments within a sentence under FSL and SSL settings (test set of the MSP-Podcast corpus). Lower ratio values indicate better temporal consistency as the cluster assignments for the chunks within a sentence are more stable.

effectiveness of the temporal modeling. Interestingly, the use of unlabeled data in the vanilla DeepEmoCluster method improves the temporal consistency of the model (see the left-shifting trend of the blue color distribution from Fig. 3(a) to Fig. 3(b)). We can also observe a general trend that a higher temporal consistency of the clusters (i.e., lower transition ratio) leads to better recognition performances when we compare Figure 3, Table 2 and Table 3. The peak ratio in the figures is 0.8 (Fig. 3(a)) for the FSL approach and 0.6 (Fig. 3(b)) for the SSL approach. This shift leads to improvements



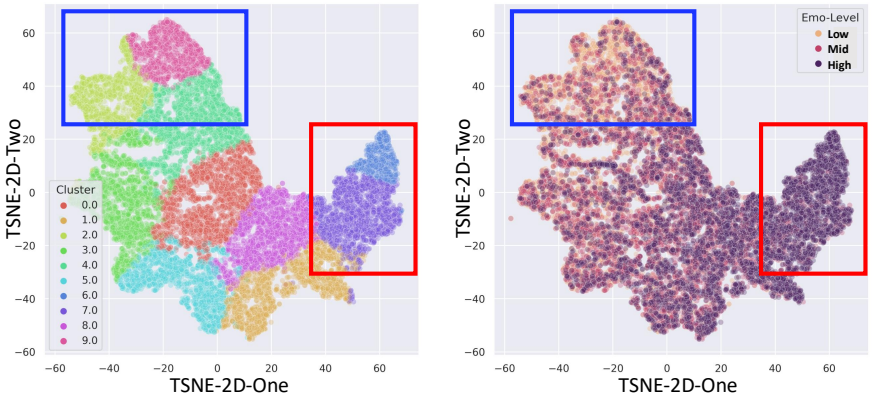
(a) t-SNE plot of the vanilla DeepEmoCluster (FSL)



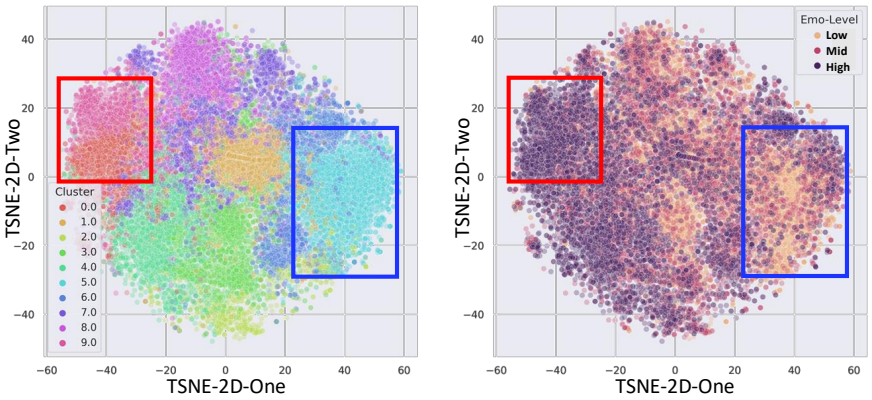
(b) t-SNE plot of the Temp-Triplet (FSL)

Figure 4: The t-SNE plots of hidden representations for the Temp-Triplet and vanilla DeepEmoCluster models trained under an FSL setup (test set of the MSP-Podcast corpus). For each case, the left-hand side figures provide the clustering assignment groups, and the right-hand side figures provide the corresponding ground-truth emotional levels aggregated into low (1-3), middle (3-5) and high (5-7) scores. The figure shows the t-SNE plots for arousal. The blue and red frames represent a low arousal region and a high arousal region, respectively.

in the performance for both methods, when we compare Table 2 (FSL) and Table 3 (SSL).



(a) t-SNE plot of the vanilla DeepEmoCluster (SSL-15K)



(b) t-SNE plot of the Temp-Triplet (SSL-15K)

Figure 5: The t-SNE plots of hidden representations for the Temp-Triplet and vanilla DeepEmoCluster models trained under an SSL-15K setup (test set of the MSP-Podcast corpus). For each case, the left-hand side figures provide the clustering assignment groups, and the right-hand side figures provide the corresponding ground-truth emotional levels aggregated into low (1-3), middle (3-5) and high (5-7) scores. The figure shows the t-SNE plots for arousal. The blue and red frames represent a low arousal region and a high arousal region, respectively.

5.3. Analysis of Clustering Representation

This section compares the hidden representations learned by the Temp-Triplet and vanilla DeepEmoCluster models using t-SNE plots (van der Maaten and Hinton, 2008). Our focus is on visualizing the emotional content of the chunks assigned to the clusters. The t-SNE technique projects the high

dimensional hidden embeddings (i.e., the h_i in Eq. 4) into a 2D subspace for better visualization of the learned feature representation created by a model. This goal is achieved by minimizing the *Kullback-Leibler divergence* (KLD) between the two sets of feature distributions (i.e., high-dimensional versus low-dimensional) to preserve the most prominent global data structure. Figures 4 and 5 visualize the plots under FSL and SSL setups, respectively. The Figures show the t-SNE plots for the vanilla DeepEmoCluster (Figs. 4(a) and 5(a)), and Temp-Triplet (Figs 4(b) and 5(b)) models. The left-hand side of each subfigure indicates the cluster assignments coded in different colors. The right-hand side of each subfigure indicates its corresponding ground-truth emotional levels (three bins). We select arousal as the emotional attribute in our illustration for simplicity since valence and dominance also show similar trends. Considering that the arousal values range from 1 to 7, the emotion levels in the right-hand side figure represent low (1-3), middle (3-5), and high (5-7) levels of arousal. The combination of these figures (i.e., clusters and emotional content) helps us monitor the emotional content contained in each cluster. The data points in the plots are based on the test set of the MSP-Podcast corpus. Each data point is the mapping of a hidden embedding (i.e., h_i in Eq. 4) in the t-SNE space. We assign the sentence-level label to each of the chunks within the sentence. We highlight some regions in the figures to facilitate the interpretation. The blue and red rectangles show low (tan color in right-hand side figure) and high (dark purple in right-hand side figure) arousal regions, respectively. The corresponding clusters are highlighted in the left-hand side figures.

Figure 4 shows that the Temp-Triplet model (Fig. 4(b)) learns to group similar emotions. This is exactly the desired objective of the DeepEmoCluster framework, which aims to obtain highly separable clusters according to the emotional content of the sentences. If the clusters were unrelated to the emotional content, the auxiliary task of recognizing the cluster assignment would not lead to SER improvements. In contrast, the vanilla DeepEmoCluster (Fig. 4(a)) shows some clusters with mixed emotional content, limiting the model's accuracy. To quantify the separability of features, we finetune a simple linear classification head for the three emotion-level classes based on the feature encoder that was trained (frozen). As expected, the clusters that show well-separated emotions (i.e., the proposed Temp-Triplet model) also obtain improved classification accuracy, improving the macro F1-score from 0.445 (Fig. 4(a)- right) to 0.472 (Fig. 4(b)- right).

Interestingly, Figure 5 shows a similar emotional grouping trend when us-

Table 4: Hyper-parameter analysis for different sizes of the unlabeled set $\mathbb{U}$ (development set). We evaluate the Temp-Trans approach with 15K, 50K, and 100K sentences, using a fixed number of clusters (Q=10).

Temp-Trans	**Aro.-CCC**	**Val.-CCC**	**Dom.-CCC**
SSL-15K	**0.5726**	**0.1674**	**0.4763**
SSL-50K	0.5605	0.1395	0.4749
SSL-100K	0.5593	0.1134	0.4579

Table 5: Hyper-parameter analysis for different numbers of clusters Q (development set). We evaluate the Temp-Trans approach with 10, 30, and 50 clusters, using a fixed size for the unlabeled set ($\mathbb{U} = 100K$).

Temp-Trans	**Aro.-CCC**	**Val.-CCC**	**Dom.-CCC**
Q=10	0.5593	0.1134	0.4579
Q=30	**0.5762**	**0.1563**	**0.4847**
Q=50	0.5726	0.1365	0.4765

ing the vanilla DeepEmoCluster framework under an SSL setting (Fig. 5(a)). We observe that the emotional content within the clusters is more consistent. The improved clusters are responsible for the improvement in CCC performances achieved by the vanilla DeepEmoCluster implemented using the SSL setting (Table 3) over the results achieved with the FSL setting (Table 2). The t-SNE plots for the Temp-Triplet model show similar consistency in the emotional content of the clusters. The proposed temporal modeling approach can effectively capture sentence-level temporal information. This capability directly contributes toward the model learning separable emotional clusters, leading to better SER performances.

5.4. Hyper-Parameter Analysis

There are three critical parameters of the proposed temporal-enhanced DeepEmoCluster framework: 1) the size of the unlabeled set $\mathbb{U}$ in the SSL scheme, 2) the number of clusters Q, and 3) the weighting factor γ in the Temp-Triplet approach.

In general, we observe the Temp-Trans model achieves the best recognition performances according to Table 2 and Table 3. Therefore, we select the Temp-Trans approach as the representative method to perform the hyper-parameter analysis for $\mathbb{U}$ and Q. Tables 4 and 5 show the analysis results for $\mathbb{U}$ and Q, respectively. Table 4 shows a general trend, where lower perfor-

Table 6: Hyper-parameter analysis for the weighting factor γ of the Temp-Triplet approach (Eq. 7). The results are reported on the development set of the MSP-Podcast corpus. The models are trained using a fixed size for the unlabeled set ($\mathbb{U} = 15K$), and a fixed number of clusters (Q=10).

Temp-Triplet	**Aro.-CCC**	**Val.-CCC**	**Dom.-CCC**
γ=0.5	0.5492	0.1394	0.4623
γ=1.0	0.5581	**0.1535**	0.4627
γ=2.0	**0.5632**	0.1438	**0.4697**
γ=4.0	0.5549	0.1389	0.4662

mances are observed when adding more unlabeled data under a fixed number of clusters (Q=10). However, when we start increasing the cluster number Q corresponding to the larger unlabeled data size, Table 5 shows a positive trend in the performance. This result suggests a relationship between the hyper-parameters $\mathbb{U}$ and Q. It is necessary to adjust the model's clustering capacity with respect to the size of $\mathbb{U}$. By increasing the cluster number Q, the model obtains higher clustering capacity to encode more complex acoustic conditions, such as speaker traits, microphone settings, and potentially environmental noises included in the unlabeled set. Using a limited number of clusters for a large unlabeled data set might force the model to encode *mixed* features within each cluster, causing additional confusion and inferior performance. However, the best value for Q is a finetuned parameter that depends on the size of $\mathbb{U}$, where a higher number of clusters is not necessarily better. In the case when $\mathbb{U}$=100K, using 30 clusters is better than using 50 clusters (Table 5).

The parameter γ controls the contribution of the temporal information term for the Temp-Triplet model (i.e., the triplet loss). The analysis of γ is useful for understanding how the temporal information impacts the model performance. Table 6 shows the results. The performance drops if we reduce the value for γ. This result verifies the importance of considering temporal information. However, the model trained with a higher value for γ also decreases the recognition performances (i.e., see the decreasing trend of arousal, dominance, and valence when γ=4). This result suggests that a balanced weight between temporal and emotional information is needed to achieve the best model performances.

6. Conclusions and Future Work

This study proposed the temporal-enhanced DeepEmoCluster framework that incorporates temporal constraints in the use of clusters as an auxiliary task for SSL tasks. The temporal constraints are imposed using two alternative sentence-level temporal modeling approaches: 1) temporal-net, and 2) triplet loss function. The temporal-net method consists of an additional network module (i.e., Temp-GRU, Temp-CNN, and Temp-Trans) that encodes the temporal information across data chunks in the sentence. The triplet loss function leverages the temporal relationship between data chunks to learn the hidden representation with additional temporal constraints (i.e., Temp-Triplet). We propose a sentence-level uniform sampling strategy to preserve the original sentence-level temporal orders of the data chunks in the sentence, enabling the use of either the temporal-net or the triplet loss function. Our experimental results based on the MSP-Podcast corpus demonstrated that the temporal-enhanced models consistently outperform the vanilla DeepEmoCluster and other existing reconstruction-based semi-supervised frameworks (AE, VAE, LadderNet) in SER tasks for all the emotion attributes (i.e., arousal, dominance, and valence). These improvements are observed when the models are implemented with either FSL or SSL training schemes. The analysis demonstrated that the temporal-enhanced model obtains a high temporal consistency property, where nearby chunks are assigned to similar clusters. The analysis also reveals that the temporal-enhanced DeepEmoCluster framework leads to clusters with well-defined emotional patterns, validating the effectiveness of the proposed temporal modeling approaches. We conclude that these properties lead to performance improvements.

Our main goal for our future research is to extend this research direction to a multimodal formulation. We aim to derive a complete clustering representation of emotions, under a multimodality scenario (e.g., speech, spoken contents, and facial expression). Since humans convey emotions in a multimodality manner, the modeling of emotional clusters should not be limited to a single channel. An open question is how to model the cross-modality temporal relationships during the clustering process. We expect to achieve this goal with an advanced temporal modeling approach that considers the synchronization and characteristics of the expression and perception of emotions across different modalities. Another research direction is to obtain feature representations that learn to cluster the data into meaningful multimodal emotional clusters.

Acknowledgment

This study was funded by the National Science Foundation (NSF) under grant CNS-2016719. Github: https://github.com/winston-lin-wei-cheng/Temporal-Enhanced-DeepEmoCluster

References

Abdelwahab, M., Busso, C., 2018. Study of dense network approaches for speech emotion recognition, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2018), IEEE, Calgary, AB, Canada. pp. 5084–5088. doi:10.1109/ICASSP.2018.8461866.

Abdelwahab, M., Busso, C., 2019. Active learning for speech emotion recognition using deep neural network, in: International Conference on Affective Computing and Intelligent Interaction (ACII 2019), Cambridge, UK. pp. 441–447. doi:10.1109/ACII.2019.8925524.

Asano, Y., Patrick, M., Rupprecht, C., Vedaldi, A., 2020. Labelling unlabelled videos from scratch with multi-modal self-supervision, in: Advances in Neural Information Processing Systems (NeurIPS 2020), Virtual. pp. 4660–4671.

Baevski, A., Zhou, Y., Mohamed, A., Auli, M., 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations, in: Advances in Neural Information Processing Systems (NeurIPS 2020), Virtual. pp. 12449–12460.

Bitouk, D., Verma, R., Nenkova, A., 2010. Class-level spectral features for emotion recognition. Speech Communication 52, 613–625. doi:10.1016/j.specom.2010.02.010.

Busso, C., Narayanan, S., 2006. Interplay between linguistic and affective goals in facial expression during emotional utterances, in: 7th International Seminar on Speech Production (ISSP 2006), Ubatuba-SP, Brazil. pp. 549–556.

Caron, M., Bojanowski, P., Joulin, A., Douze, M., 2018. Deep clustering for unsupervised learning of visual features, in: Ferrari, V., Hebert,

M., Sminchisescu, C., Weiss, Y. (Eds.), European Conference on Computer Vision (ECCV 2018). Springer Berlin Heidelberg, Munich, Germany. volume 11217 of *Lecture Notes in Computer Science*, pp. 139–156. doi:10.1007/978-3-030-01264-9_9.

Chapelle, O., Schlkopf, B., , Zien, A., 2006. Semi-Supervised Learning. MIT Press, Cambridge, Mass., USA.

Chen, L.W., Rudnicky, A., 2023. Exploring Wav2vec 2.0 fine tuning for improved speech emotion recognition, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2023), Rhodes Island, Greece. pp. 1–5. doi:10.1109/ICASSP49357.2023.10095036.

Choi, D.Y., Song, B.C., 2020. Semi-supervised learning for continuous emotion recognition based on metric learning. IEEE Access 8, 113443–113455. doi:10.1109/ACCESS.2020.3003125.

Chou, H.C., Lin, W.C., Lee, C.C., Busso, C., 2022. Exploiting annotators' typed description of emotion perception to maximize utilization of ratings for speech emotion recognition, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2022), Singapore. pp. 7717–7721. doi:10.1109/ICASSP43922.2022.9746990.

Deng, J., Xu, X., Zhang, Z., Frühholz, S., Schuller, B., 2018. Semisupervised autoencoders for speech emotion recognition. IEEE/ACM Transactions on Audio, Speech, and Language Processing 26, 31–43. doi:10.1109/TASLP.2017.2759338.

Goncalves, L., Busso, C., 2022. Improving speech emotion recognition using self-supervised learning with domain-specific audiovisual tasks, in: Interspeech 2022, Incheon, South Korea. pp. 1168–1172. doi:10.21437/Interspeech.2022-11012.

Goncalves, L., Busso, C., 2023. Learning cross-modal audiovisual representations with ladder networks for emotion recognition, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2023), Rhodes island, Greece. pp. 1–5. doi:10.1109/ICASSP49357.2023.10096138.

Guo, X., Liu, X., Zhu, E., Yin, J., 2017. Deep clustering with convolutional autoencoders, in: Liu, D., Xie, S., Li, Y., Zhao, D., El-Alfy, E. (Eds.), International conference on neural information processing (ICONIP 2017). Springer Berlin Heidelberg, Guangzhou, China. volume 10635 of *Lecture Notes in Computer Science*, pp. 373–382. doi:10.1007/978-3-319-70096-0_39.

Hajarolasvadi, N., Demirel, H., 2019. 3D CNN-based speech emotion recognition using K-means clustering and spectrograms. Entropy 21, 479. doi:10.3390/e21050479.

Han, K., Yu, D., Tashev, I., 2014. Speech emotion recognition using deep neural network and extreme learning machine, in: Interspeech 2014, Singapore. pp. 223–227.

Han, W., Ruan, H., Chen, X., Wang, Z., Li, H., Schuller, B., 2018. Towards temporal modelling of categorical speech emotion recognition, in: Interspeech 2018, Hyderabad, India. pp. 932–936. doi:10.21437/Interspeech.2018-1858.

Hsu, W.N., B. Bolte, Y.H.H.T., Lakhotia, K., Salakhutdinov, R., Mohamed, A., 2021. HuBERT: Self-supervised speech representation learning by masked prediction of hidden units. IEEE/ACM Transactions on Audio, Speech, and Language Processing 29, 3451–3460. doi:10.1109/TASLP.2021.3122291.

Huang, C.W., Narayanan, S.S., 2017. Deep convolutional recurrent neural network with attention mechanism for robust speech emotion recognition, in: IEEE International Conference on Multimedia and Expo (ICME 2017), Hong Kong, China. pp. 583–588. doi:10.1109/ICME.2017.8019296.

Huang, J., Li, Y., Tao, J., Lian, Z., Niu, M., Yi, J., 2018. Speech emotion recognition using semi-supervised learning with ladder networks, in: Asian Conference on Affective Computing and Intelligent Interaction (ACII Asia 2018), Beijing, China. pp. 1–5. doi:10.1109/ACIIAsia.2018.8470363.

Jalal, M., Loweimi, E., Moore, R., Hain, T., 2019. Learning temporal clusters using capsule routing for speech emotion recognition, in: Interspeech 2019, Graz, Austria. pp. 1701–1705. doi:10.21437/Interspeech.2019-3068.

Kim, J., Englebienne, G., Truong, K., Evers, V., 2017. Deep temporal models using identity skip-connections for speech emotion recognition, in: ACM international conference on Multimedia (MM 2017), Mountain View, CA, USA. pp. 1006–1013. doi:10.1145/3123266.3123353.

Kitaev, N., Kaiser, Ł., Levskaya, A., 2020. Reformer: The efficient transformer, in: International Conference on Learning Representations (ICLR 2020, Addis Ababa Ethiopia. pp. 1–12.

Latif, S., Rana, R., Qadir, J., Epps, J., 2018. Variational autoencoders for learning latent representations of speech emotion, in: Interspeech 2018, Hyderabad, India. pp. 3107–3111. doi:10.21437/Interspeech.2018-1568.

Lee, C.C., Mower, E., Busso, C., Lee, S., Narayanan, S., 2011. Emotion recognition using a hierarchical binary decision tree approach. Speech Communication 53, 1162–1171. doi:10.1016/j.specom.2011.06.004.

Lee, C.C., Sridhar, K., Li, J.L., W.-C.Lin, Su, B.H., Busso, C., 2021. Deep representation learning for affective speech signal analysis and processing: Preventing unwanted signal disparities. IEEE Signal Processing Magazine 38, 22–38. doi:10.1109/MSP.2021.3105939.

Leem, S.G., Fulford, D., Onnela, J.P., Gard, D., Busso, C., 2021. Separation of emotional and reconstruction embeddings on ladder network to improve speech emotion recognition robustness in noisy conditions, in: Interspeech 2021, Brno, Czech Republic. pp. 2871–2875. doi:10.21437/Interspeech.2021-1438.

Li, C., Li, X., Zhang, L., Peng, B., Zhou, M., Gao, J., 2020. Self-supervised pre-training with hard examples improves visual representations. ArXiv e-prints (arXiv:2012.13493) , 1–10doi:10.48550/arXiv.2012.13493, arXiv:2012.13493.

Lin, W.C., Busso, C., 2020. An efficient temporal modeling approach for speech emotion recognition by mapping varied duration sentences into fixed number of chunks, in: Interspeech 2020, Shanghai, China. pp. 2322–2326. doi:10.21437/Interspeech.2020-2636.

Lin, W.C., Busso, C., 2022. Chunk-level speech emotion recognition: A general framework of sequence-to-one dynamic temporal modeling. IEEE

Transactions on Affective Computing Early Access. doi:10.1109/TAFFC.2021.3083821.

Lin, W.C., Busso, C., 2023. Sequential modeling by leveraging non-uniform distribution of speech emotion. IEEE/ACM Transactions on Audio, Speech, and Language Processing 31, 1087–1099. doi:10.1109/TASLP.2023.3244527.

Lin, W.C., Sridhar, K., Busso, C., 2021. DeepEmoCluster: A semi-supervised framework for latent cluster representation of speech emotions, in: IEEE international conference on acoustics, speech and signal processing (ICASSP 2021), Toronto, ON, Canada. pp. 7263–7267. doi:10.1109/ICASSP39728.2021.9414035.

Lin, Y.L., Wei, G., 2005. Speech emotion recognition based on HMM and SVM, in: International Conference on Machine Learning and Cybernetics (ICMLC 2005), Guangzhou, China. pp. 4898–4901. doi:10.1109/ICMLC.2005.1527805.

Lotfian, R., Busso, C., 2018. Predicting categorical emotions by jointly learning primary and secondary emotions through multitask learning, in: Interspeech 2018, Hyderabad, India. pp. 951–955. doi:10.21437/Interspeech.2018-2464.

Lotfian, R., Busso, C., 2019a. Building naturalistic emotionally balanced speech corpus by retrieving emotional speech from existing podcast recordings. IEEE Transactions on Affective Computing 10, 471–483. doi:10.1109/TAFFC.2017.2736999.

Lotfian, R., Busso, C., 2019b. Lexical dependent emotion detection using synthetic speech reference. IEEE Access 7, 22071–22085. doi:10.1109/ACCESS.2019.2898353.

Mariooryad, S., Lotfian, R., Busso, C., 2014. Building a naturalistic emotional speech corpus by retrieving expressive behaviors from existing speech corpora, in: Interspeech 2014, Singapore. pp. 238–242.

Mirsamadi, S., Barsoum, E., Zhang, C., 2017. Automatic speech emotion recognition using recurrent neural networks with local attention, in:

IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2017), New Orleans, LA, USA. pp. 2227–2231. doi:10.1109/ICASSP.2017.7952552.

Mustaqeem, Sajjad, M., Kwon, S., 2020. Clustering-based speech emotion recognition by incorporating learned features and deep BiLSTM. IEEE Access 8, 79861–79875. doi:10.1109/ACCESS.2020.2990405.

Ouyang, X., Kawaai, S., Goh, E., Shen, S., Ding, W., Ming, H., Huang, D.Y., 2017. Audio-visual emotion recognition using deep transfer learning and multiple temporal models, in: ACM International Conference on Multimodal Interaction (ICMI 2017), Glasgow, UK. pp. 577–582. doi:10.1145/3136755.3143012.

Parthasarathy, S., Busso, C., 2018. Ladder networks for emotion recognition: Using unsupervised auxiliary tasks to improve predictions of emotional attributes, in: Interspeech 2018, Hyderabad, India. pp. 3698–3702. doi:10.21437/Interspeech.2018-1391.

Parthasarathy, S., Busso, C., 2020. Semi-supervised speech emotion recognition with ladder networks. IEEE/ACM Transactions on Audio, Speech, and Language Processing 28, 2697–2709. doi:10.1109/TASLP.2020.3023632.

Pepino, L., Riera, P., Ferrer, L., 2021. Emotion recognition from speech using Wav2vec 2.0 embeddings, in: Interspeech 2021, Brno, Czech Republic. pp. 3400–3404. doi:10.21437/Interspeech.2021-703.

Reddy Naini, A., Kohler, M., Busso, C., 2023. Unsupervised domain adaptation for preference learning based speech emotion recognition, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2023), Rhodes island, Greece. pp. 1–5. doi:10.1109/ICASSP49357.2023.10094301.

Sabour, S., Frosst, N., Hinton, G., 2017. Dynamic routing between capsules, in: In Advances in Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA. pp. 1–11.

Saeed, A., Grangier, D., Zeghidour, N., 2021. Contrastive learning of general-purpose audio representations, in: IEEE International Conference

on Acoustics, Speech and Signal Processing (ICASSP 2021), Toronto, ON, Canada. pp. 3875–3879. doi:10.1109/ICASSP39728.2021.9413528.

Sahoo, S., Kumar, P., Raman, B., Pratim Roy, P., 2019. A segment level approach to speech emotion recognition using transfer learning, in: Palaiahnakote, S., Sanniti di Baja, G., Wang, L., Yan, W. (Eds.), Asian Conference on Pattern Recognition (ACPR 2019). Springer Berlin Heidelberg, Auckland, New Zealand. volume 12047 of *Lecture Notes in Computer Science*, pp. 435–448. doi:10.1007/978-3-030-41299-9_34.

Schneider, S., Baevski, A., Collobert, R., Auli, M., 2019. Wav2vec: Unsupervised pre-training for speech recognition, in: Interspeech 2019, Graz, Austria. pp. 3465–3469. doi:10.21437/Interspeech.2019-1873.

Schuller, B., Steidl, S., Batliner, A., Vinciarelli, A., Scherer, K., Ringeval, F., Chetouani, M., Weninger, F., Eyben, F., Marchi, E., Mortillaro, M., Salamin, H., Polychroniou, A., Valente, F., Kim, S., 2013. The INTERSPEECH 2013 computational paralinguistics challenge: Social signals, conflict, emotion, autism, in: Interspeech 2013, Lyon, France. pp. 148–152.

Shor, J., Jansen, A., Maor, R., Lang, O., Tuval, O., Quitry, F., Tagliasacchi, M., Shavitt, I., Emanuel, D., Haviv, Y., 2020. Towards learning a universal non-semantic representation of speech, in: Interspeech 2020, Shanghai, China. pp. 140–144. doi:10.21437/Interspeech.2020-1242.

Simonyan, K., Zisserman, A., 2015. Very deep convolutional networks for large-scale image recognition, in: International Conference on Learning Representations (ICLR 2015), San Juan, Puerto Rico. pp. 1–10.

Tao, F., Liu, G., 2018. Advanced LSTM: a study about better time dependency modeling in emotion recognition, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2018), Calgary, AB, Canada. pp. 2906–2910. doi:10.1109/ICASSP.2018.8461750.

Tarantino, L., Garner, P., Lazaridis, A., 2019. Self-attention for speech emotion recognition, in: Interspeech 2019, Graz, Austria. pp. 2578–2582. doi:10.21437/Interspeech.2019-2822.

Triguero, I., , García, S., Herrera, F., 2015. Self-labeled techniques for semi-supervised learning: taxonomy, software and empirical study. Knowledge and Information systems 42, 245–284. doi:10.1007/s10115-013-0706-y.

Trinh, T., Dai, A., Luong, T., Le, Q., 2018. Learning longer-term dependencies in RNNs with auxiliary losses, in: International Conference on Machine Learning (ICML 2018), Stockholm, Sweden. pp. 4965–4974.

van der Maaten, L., Hinton, G., 2008. Visualizing data using t-SNE. Journal of Machine Learning Research 9, 2579–2605.

Van Gansbeke, W., Vandenhende, S., Georgoulis, S., Proesmans, M., Van Gool, L., 2020. SCAN: Learning to classify images without labels, in: Vedaldi, A., Bischof, H., Brox, T., Frahm, J. (Eds.), European Conference on Computer Vision (ECCV 2020). Springer Berlin Heidelberg, Glasgow, UK. volume 12355 of *Lecture Notes in Computer Science*, pp. 268–285. doi:10.1007/978-3-030-58607-2_16.

Wagner, J., Triantafyllopoulos, A., Wierstorf, H., Schmitt, M., Burkhardt, F., Eyben, F., Schuller, B., 2023. Dawn of the transformer era in speech emotion recognition: Closing the valence gap. IEEE Transactions on Pattern Analysis and Machine Intelligence 45, 10745–10759. doi:10.1109/TPAMI.2023.3263585.

Wang, L., Luc, P., Recasens, A., Alayrac, J.B., van den Oord, A., 2021. Multimodal self-supervised learning of general audio representations. ArXiv e-prints (arXiv:2104.12807) , 1–6doi:10.48550/arXiv.2104.12807, arXiv:2104.12807.

Yan, T., Meng, H., Parada-Cabaleiro, E., Tao, J., Schuller, B., 2022. A residual multi-scale convolutional transformer network with chunk-level log-mel spectrograms for speech emotion recognition. TechRxiv , 1–17doi:10.36227/techrxiv.21309600.v1, arXiv:21309600.

Yarowsky, D., 1995. Unsupervised word sense disambiguation rivaling supervised methods, in: ACM Association for Computational Linguistics (ACL 1995), Cambridge, MA, USA. pp. 189–196.

Zhang, P., Wu, M., Dinkel, H., Yu, K., 2021. DEPA: Self-supervised audio embedding for depression detection, in: ACM International Conference on

Multimedia (MM 2021), Virtual Event China. pp. 135–143. doi:10.1145/3474085.3479236.

Zhang, Z., Han, J., Deng, J., Xu, X., Ringeval, F., Schuller, B., 2018. Leveraging unlabeled data for emotion recognition with enhanced collaborative semi-supervised learning. IEEE Access 6, 22196–22209. doi:10.1109/ACCESS.2018.2821192.

Zhu, Z., Sato, Y., 2021. Speech emotion recognition using semi-supervised learning with efficient labeling strategies, in: IEEE Automatic Speech Recognition and Understanding Workshop (ASRU 2021), Cartagena, Colombia. pp. 358–365. doi:10.1109/ASRU51503.2021.9687938.