INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

Meta Multi-task Learning for Speech Emotion Recognition

Ruichu Cai[1], *Kaibin Guo*[1], *Boyan Xu*[1*], *Xiaoyan Yang*[2], *Zhenjie Zhang*[2*]

[1]School of Computer Science, Guangdong University of Technology, China
[2]Singapore R&D, Yitu Technology Pte Ltd., Singapore

cairuichu@gmail.com, hyskid801@gmail.com, hpakyim@gmail.com
yangxiaoyan@gmail.com, zhangzhenjie@gmail.com

Abstract

Most existing Speech Emotion Recognition (SER) approaches ignore the relationship between the categorical emotional labels and the dimensional labels in valence, activation or dominance space. Although multi-task learning has recently been introduced to explore such auxiliary tasks of SER, existing approaches only share the feature extractor under the traditional multi-task learning framework and can not efficiently transfer the knowledge from the auxiliary tasks to the target task. In order to address these issues, we propose a Meta Multi-task Learning method for SER by combining the multi-task learning with meta learning. Our contributions include: 1) to model the relationship among auxiliary tasks, we extend the task generation of meta learning to the form of multiple tasks, and 2) to transfer the knowledge from the auxiliary tasks to the target task, we propose a tuning-based transfer training mechanism in the meta learning framework. The experiments on IEMOCAP show that our approach outperforms the state-of-the-art solution (UA: 70.32%, WA: 76.64%).

Index Terms: speech emotion recognition, meta multi-task learning, transfer learner

1. Introduction

Speech Emotion Recognition (SER) aims to detect the speaker's emotions, which plays an essential role in human-computer interaction, such as customer service calls [1, 2]. In recent years, benefiting from the development of deep learning technologies, the performance of SER has significantly improved. Both convolutional neural networks(CNNs)[3] and recurrent neural networks(RNNs)[4] have been applied to extract emotional features from either time domain or frequency domain. For example, Mao et al. propose to use CNNs to learn salient features through sparse auto-encoders and discriminative feature analysis[3]. Lee et al. consider SER as a sequence-to-sequence task and adopt BiLSTM to predict emotional labels in each time step[4]. Recently, a CNN-LSTM hybrid framework, taking advantage of both CNNs and RNNs, has become popular for SER[5, 6]. The works mentioned above only consider extracting features from spectrograms for SER, failing to take into account other factors such as ways of expression.

However, studies [7, 8] have shown that different emotions have different characteristics in valence-activation space. Human emotions are complex and related to the way humans express emotions[9, 10], such as valence (V, positive or negative), activation (A, calm or excited), and dominance (D, passive or aggressive). They represent how humans behave in different emotion eliciting events, and the relation of which can be shown in Fig. 1. The main challenges of SER lie in identifying the relationships between these factors and human emotions.

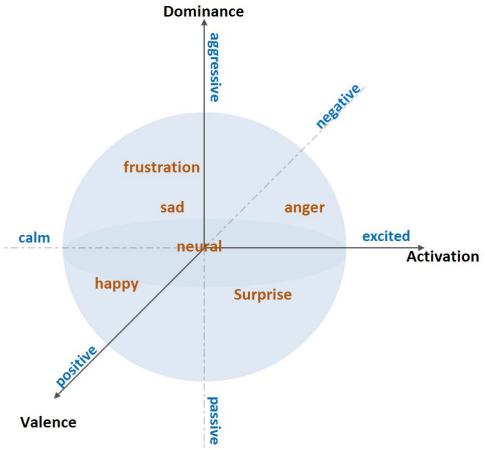


Figure 1: *The distribution of categorical emotion in Valence-Activation-Dominance space.*

Recently, a number of research works attempt to apply multi-task learning on SER to leverage information of V/A/D. Xia et al. use activation and valence information to help recognize categorical emotional labels based on the deep belief network with multi-task learning[11]. In [12], Neumann et al. combine an attentive convolutional neural network with autoencoder in multi-task learning to recognize the emotion and values of valence and activation at the same time. Although promising results were reported, there are two limitations rooted in multi-task learning-based solutions. First, such methods do not model well the inherent relationships between auxiliary tasks. Second, original multi-task learning methods only share information at the feature extractor level from auxiliary tasks to the target one, which can not transfer knowledge in the learner level of auxiliary tasks. The key to tackle these limitations is to design a model that can better transfer knowledge among auxiliary tasks of SER based on a multi-task learning framework.

Therefore, we propose a new learning algorithm called *Meta Multi-task Learning* (MMTL) that smoothly combines meta learning techniques and multi-task learning techniques. *Meta Multi-task Learning* consists of two stages: *Multi-train Stage* and *Knowledge Transfer Stage*. In the *Multi-train Stage*, instead of using different learners in native multi-task learning, we share the same meta learner in all auxiliary tasks. In the *Knowledge Transfer Stage*, we build a transfer learner on top of the meta learner to transfer knowledge from auxiliary tasks to the target one. A tuning-based training mechanism is proposed to train the two types of learners alternately and cooperatively. It includes two steps: 1) training the transfer learner with

* Corresponding author

Copyright © 2020 ISCA

http://dx.doi.org/10.21437/Interspeech.2020-2624

a large learning rate on the target task; 2) fine-tuning the transfer learner and meta-learner with small learning rates on the target task.

The core contributions are summarized as follows:

- To the best of our knowledge, this is the first work to apply meta learning techniques for speech emotion recognition;
- In *Multi-train Stage*, we propose a hybrid learning method that combines meta learning techniques and multi-task learning techniques to share knowledge among auxiliary tasks.
- In *Knowledge Transfer Stage*, we propose a tuning-based transfer training mechanism to transfer the knowledge from auxiliary tasks to the target task.

2. Meta Multi-task learning

Meta learning mainly consists of three types of approaches, which are metric-based [13, 14], model-based [15, 16], and optimization-based [17, 18]. Among these approaches, the optimization-based Model-Agnostic Meta Learning(MAML) [17] can be combined with any other model that allows gradient-based optimization. Thus, we adopt MAML as the basic meta learning method to provide initialization of parameters for new tasks.

The proposed MMTL iteratively employ the *Multi-train Stage* and the *Knowledge Transfer Stage* to alternately update the model on auxiliary tasks and SER task, as shown in Fig. 2. In the *Multi-train Stage*, as shown in the upper part of Fig. 2, the meta learner is trained using the average loss on the V/A/D auxiliary tasks. In the *Knowledge Transfer Stage*, as shown in the lower part of Fig. 2, the transfer learner is initialized with the meta learner trained in the *Multi-train Stage*, and further trained using the loss on the SER task. Following, we will provide the details of the model, including how to generate tasks for the two-stage process, how to model the relationship among auxiliary tasks in the *Multi-train Stage*, and how to transfer the knowledge from the auxiliary tasks to the target task in the *Knowledge Transfer Stage*.

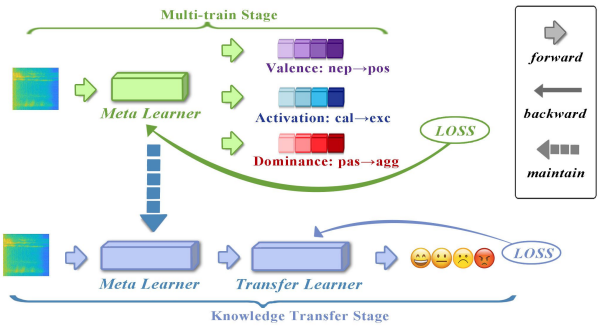


Figure 2: *An iteration of MMTL: includes the Multi-train Stage and the Knowledge Transfer Stage.*

2.1. Task generation for two-stage process

In this section, we formulate the task generation for two-stage process. Let $\mathcal{T} = \{\mathcal{T}_1, \mathcal{T}_2, ..., \mathcal{T}_m\}$ denote the training set obtained by sampling a fixed number of samples from m different speakers in each training step. And $X = \{X_1, X_2, ..., X_m\}$ denote the input features of $\mathcal{T}$. In SER, the target task is to detect the emotional label Y_i^Q of the ith speaker given X_i. Similarly, the auxiliary tasks are to detect labels $\{Y_i^{S,V}, Y_i^{S,A}, Y_i^{S,D}\}$ in valence, activation, and dominance domains of the ith speaker given X_i. For each speaker's sample set $\mathcal{T}_i$, we define the auxiliary tasks D_i^S and the target task D_i^Q as follows:

$$\mathcal{D}_i^{S,j} = (X_i, Y_i^{S,j}) \tag{1}$$

$$\mathcal{D}_i^{S} = (\mathcal{D}_i^{S,1}, \mathcal{D}_i^{S,2}, ..., \mathcal{D}_i^{S,k}) \tag{2}$$

$$\mathcal{D}_i^{Q} = (X_i, Y_i^{Q}) \tag{3}$$

where $\mathcal{D}_i^{S,j}$ represents the jth auxiliary task.

Algorithm 1 Meta Multi-task Learning

Require: $p(\mathcal{T})$: distribution over training set
Require: α, β, γ: step size hyperparameters
while not converge **do**
  Sample speakers $\mathcal{T}_i \sim p(\mathcal{T})$
  for all $\mathcal{T}_i$ **do**
    for $j = 1$ to k **do**
      Evaluate $\nabla_\theta \mathcal{L}_i^{S,j}(\theta)$ on $\mathcal{D}_i^{S,j}$
    end for
    Compute ${\theta'}_i$ based on equation 4
    Evaluate $\nabla_\phi \mathcal{L}_i^{Q}(\phi)$ on $\mathcal{D}_i^{Q}$
    Compute ${\theta'}_i$ based on equation 5
  end for
  Update θ and ϕ based on equation (6) and (7)
end while

2.2. Multi-train stage

In the *Multi-train Stage*, we opt to employ CNN-LSTM based SER framework as the meta learner, which performed well in SER task. The meta learner is trained on the auxiliary tasks D_i^S of each speaker $\mathcal{T}_i$. We use θ to denote the parameters of the meta learner. Different from multi-task learning, all auxiliary tasks share the same meta learner. We compute the gradient $\nabla_\theta \mathcal{L}_i^{S,j}(\theta)$ on each auxiliary task j. The parameters θ_i of the meta learner is updated by iterating over the ith speakers as follows:

$${\theta'}_i = \theta - \frac{\alpha}{k} \sum_{j=1}^{k} \nabla_{\theta_i} \mathcal{L}_i^{S,j}(\theta_i), \tag{4}$$

where α is the learning rate of meta learner. Notice that for each speaker $\mathcal{T}_i$ we maintain its own updated parameter θ_i' from original parameter θ.

In multi-task learning, auxiliary tasks usually share one common feature extractor but with task-specific classifiers. In MMTL, instead of using task-specific classifiers, the *Multi-train Stage* shares one meta learner which includes one common feature extractor and one classifier. Similar to multi-task learning, the common meta learner can learn how to extract the salient acoustic features for identifying valence, activation, and dominance information from the audio. The sharing of classifiers in the meta learner probably leads to sub-optimal performance in classification on each auxiliary task, but doing so enables the meta learner to learn the common knowledge between the auxiliary tasks. After the *Multi-train Stage*, the meta learner can get a good initialization of parameter for the *Knowledge Transfer Stage*.

2.3. Knowledge Transfer Stage

In *Knowledge Transfer Stage*, we propose a tuning-based transfer training mechanism to transfer the knowledge maintained

in the meta learner, which is well trained in the last stage by auxiliary tasks. A Fully-connected layer followed by the meta learner is implemented to transfer knowledge from auxiliary tasks to target task, named transfer learner.

Let ϕ denote the parameters of the transfer learner. In tuning-based transfer training mechanism, we first fix the updated parameters θ_i' of the meta learner and train the parameters ϕ of the transfer learner only on the target task. A large learning rate β is utilized to better train the transfer leaner from scratch, and the parameters ϕ is updated as follows:

$$\phi'_i = \phi - \beta \nabla_\phi \mathcal{L}_i^Q(\phi). \tag{5}$$

Notice that for each speaker $\mathcal{T}_i$ we maintain its own updated parameter ϕ_i' from original parameter ϕ.

After each speaker sample set $\mathcal{T}_i$ have finished the above update process, we can obtain the updated parameter vector θ'_i of the meta learner and ϕ'_i the transfer learner of different speakers, as shown in Eq. 4 and 5. To fine-tune with a small learning rate γ, the original parameter θ of meta learner and ϕ of transfer learner will be finally updated as follows:

$$\begin{aligned}\theta' &= \theta - \gamma \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \nabla_\theta \mathcal{L}_i^Q(\theta'_i, \phi'_i) \\ &= \theta - \gamma \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \nabla_\theta \mathcal{L}_i^Q(\theta - \frac{\alpha}{k} \sum_{j=1}^{k} \nabla_\theta \mathcal{L}_i^{S,j}(\theta), \phi'_i) \quad (6) \\ \phi' &= \phi - \gamma \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \nabla_\phi \mathcal{L}_i^Q(\theta'_i, \phi'_i) \\ &= \phi - \gamma \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \nabla_\phi \mathcal{L}_i^Q(\theta'_i, \phi - \beta \nabla_\phi \mathcal{L}_i^Q(\phi)) \quad (7)\end{aligned}$$

2.4. Model Summarization

The training process of MMTL is outlined in Algorithm 1. Lines 4-7 describe the *Multi-train Stage*, where we calculate the loss of the auxiliary tasks for all speakers and update their own parameters of the meta learner. Line 8-9 describe the *Knowledge Transfer Stage*, where we compute the loss of the target task for all speakers and update their own parameters of the transfer learner. Line 11 aggregates the loss in the above two stages and updates the original parameters of the meta learner and the transfer learner. Lines 2-11 are repeated until the model converges.

The testing process of MMTL is different from the training process in that there is no *Knowledge Transfer Stage* in the testing process. Given a target speaker, the meta learner of the *Multi-train Stage* is first fine-tuned with the small learning rate γ on the auxiliary tasks of the target speaker. Second, we apply the model, including both the meta learner and the transfer learner, to predict the SER labels of the target speaker.

3. Experiments

3.1. Experimental setup

3.1.1. Data processing

We evaluate our MMTL method on the widely used dataset, IEMOCAP[19]. This dataset contains five sessions. Each session is recorded by a male speaker and a female speaker in improvisations or scripted scenarios. Following the previous work [20, 21], we use 4 categorical emotional labels from utterances of improvised recording, including happiness, sadness, neural, and anger. Further, utterances are also annotated into dimensional labels in valence, activation, and dominance space.

Table 1: *Architectures of meta learner and transfer learner.*

Learner	Layer type	Parameters
Meta Learner	Conv_1	12×16, 16, stride 1
	Maxpooling_1	2×2, stride 2
	Conv_2	8×12, 24, stride 1
	Max pooling_2	2×2, stride 2
	Conv_3	5×7, 32, stride 1
	Max pooling_3	2×2, stride 2
	BiLSTM	1 layer, 128 dim
	FC_1	1 layer, 64 dim
	FC_2	1 layer, 4 dim
Transfer Learner	FC_3	1 layer, 4 dim

As the values in valence, activation, and dominance are continuous, we first discretize them. For values in (0, 2), we map them to 0. Values in [2, 3) are mapped to 1. Values in [3, 4) and [4, 6) are mapped to 2 and 3 respectively. As for audio processing, we first split the utterances into segments of equal length, the duration of which is not more than 3 seconds. The label of the original utterance is given to each segment. After processing, we obtain 4432 samples[1] in total. Second, for each segment, we apply a sequence of overlapping hamming windows with a frame size of 10 ms and a window size of 200 ms. We compute the DFT of length 800 and keep the results with frequency in the range of 0-4KHz. The resulting spectrogram of each segment is an $N \times M$ matrix, where $N \leq 300$ and $M = 200$ representing the time and frequency dimensions of the spectrogram respectively. Third, we convert the spectrogram to log-power-spectrum with normalization. Last, we apply zero paddings along the time dimension of the spectrum to make $N = 300$.

In the testing phase, for each utterance, we compute the posterior probabilities of each segment and average them to get the prediction for the utterance. The label with the highest score is selected as the prediction result of the utterance.

3.1.2. Evaluation Metric

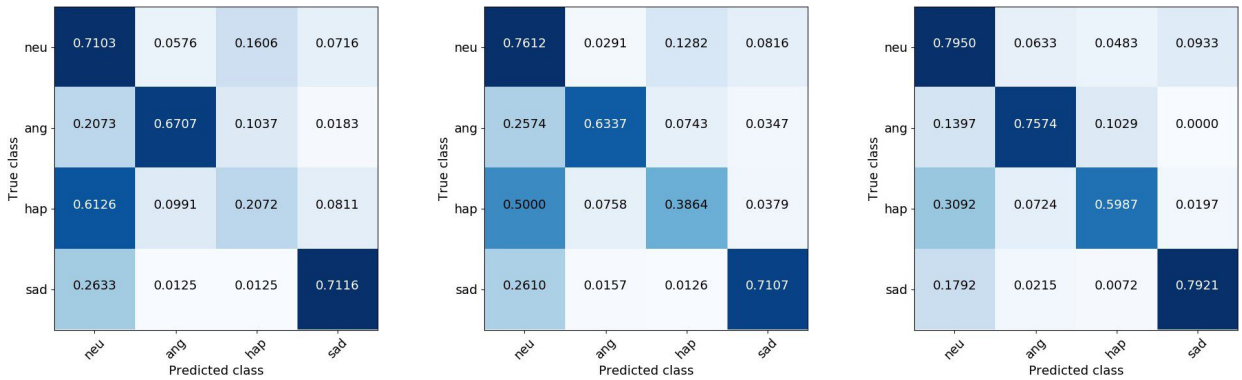
We adopt **unweighted accuracy** (UA, the average of class accuracy of test set) and **weighted accuracy** (WA, the accuracy of the test set) as metrics to evaluate SER performance. We carry out five-fold cross-validation where data from four sessions are used for training, one speaker of the last session for evaluation, and the other for testing.

3.1.3. Model configuration

The meta learner consists of three layers of CNNs, one layer of Bi-LSTM followed by two layers of Fully-connected layers(FCs). A max pooling layer is added after each CNN layer to halve the output both in frequency and time domains. Readers may refer to [5] for more details of the structure of the meta learner. The transfer learner consists of one layer of FC. The Architecture are listed in table 1.

We use cross entropy as the loss function for each task. However, class imbalance exists within and among tasks. To address this problem, we assign a weight to each task for each speaker as follows. Assume each speaker $\mathcal{T}_i$ has N_i samples,

[1] neural: 2016, anger: 585, happiness: 525, sadness: 1306



(a) CNN-LSTM (b) CNN-LSTM+MTL (c) MMTL(Ours)

Figure 3: *The confusion matrix of CNN-LSTM, CNN-LSTM+MTL and MMTL(Ours).*

among which $N_{i,c}$ samples are with labels c. We define

$$w_{i,c} = \frac{N_i}{N_{i,c}}$$

$$w'_{i,c} = \frac{w_{i,c}}{\max_c w_{i,c}}$$

as the loss weight. We use Adam as the optimizer and set the learning rates as: $\alpha = 0.01, \beta = 0.01, \gamma = 0.001$.

Considering the high computation overhead of MAML which requires to compute the Hessian Matric, we replace it in our algorithm with Reptile [18] that approximates MAML by only computing the first-order gradient. Implementation is available at https://github.com/kidconan/MMTL.

3.2. Results

We divide the experiments into two parts. In the first part, we compare our algorithm MMTL with the state-of-the-art baselines as listed in Table 2. We also extend [5] by applying multi-task learning (MTL) on it, i.e., CNN-LSTM+MTL in Table 2, which allows the CNNs to be shared among the auxiliary tasks and be fine-tuned on the auxiliary tasks during testing. The acoustic feature size of CNN-LSTM and CNN-LSTM+MTL is set to be equivalent to that of MMTL, while other baseline methods use their original settings. In the second part, we examine the influence of auxiliary task selection on the performance of SER by varying the combinations of the auxiliary tasks.

Fig. 3 shows the confusion matrices of CNN-LSTM [5], CNN-LSTM+MTL, and our algorithm MMTL. MMTL significantly improves the recognition accuracy of various emotion labels over the two baselines, especially for the "happiness" (hap) label, the number of which is smaller. For semantically opposite "happiness" and "sadness" emotions, our algorithm can distinguish them better.

Table 2 shows the results of our algorithm and the state-of-the-art model structures on IEMOCAP. It can be observed that our algorithm performs much better than the state-of-the-art model structures. This shows the effectiveness of our algorithm and the important guiding role of auxiliary task information in predicting on the main task.

Varying the combinations of auxiliary tasks, Table 3 shows that task selection has a greater impact on the algorithm performance. On the one hand, activation and dominance have a mutual exclusion effect, so when auxiliary tasks include both activation and dominance, the algorithm's effectiveness will be reduced. On the other hand, the combination of valence and activation or dominance can improve the algorithm's recognition efficiency. This experimental study also provides us a heuristic strategy for the selection of auxiliary tasks.

Table 2: *Comparisons with baseline models on IEMOCAP.*

Model	**UA**	**WA**
CNN-LSTM [5]	56.60%	65.80%
CNN-LSTM+angular softmax loss [20]	64.16%	68.74%
CNN+GRU+SeqCap [21]	59.71%	72.23%
CNN+GAP+Attention [22]	68.06%	71.75%
Self Attention [23]	63.80%	68.10%
CNN-LSTM+MTL	64.31%	68.84%
MMTL(Ours)	**70.32%**	**76.64%**

Table 3: *Comparison of task selection on IEMOCAP.*

Model	**UA**	**WA**
MMTL(V)	65.60%	71.99%
MMTL(A)	58.28%	65.71%
MMTL(D)	54.25%	66.98%
MMTL(V+D)	73.22%	75.83%
MMTL(A+D)	55.26%	67.25%
MMTL(V+A)	**74.61%**	**77.74%**
MMTL(V+A+D)	70.32%	76.64%

4. Conclusions

In this paper, we propose a *Meta Multi-task Learning*(MMTL) method for Speech Emotion Recognition. MMTL takes advantage of both meta learning and multi-task learning through well designed two-stage process including *Multi-train Stage* and *Knowledge Transfer Stage*. The core idea is to model the relationship among auxiliary tasks and transfer the knowledge to target task. The experimental results on IEMOCAP show MMTL achieves the highest WA and UA compared with the state-of-the-art methods. As future work, we will extend MMTL to the selectively transfer case by adaptively employing the relevant auxiliary tasks of the target task.

5. References

[1] B. Li, D. Dimitriadis, and A. Stolcke, "Acoustic and lexical sentiment analysis for customer service calls," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 5876–5880.

[2] D. Morrison, R. Wang, and L. C. De Silva, "Ensemble methods for spoken emotion recognition in call-centres," *Speech communication*, vol. 49, no. 2, pp. 98–112, 2007.

[3] Q. Mao, M. Dong, Z. Huang, and Y. Zhan, "Learning salient features for speech emotion recognition using convolutional neural networks," *IEEE Transactions on Multimedia*, vol. 16, pp. 2203–2213, 2014.

[4] J. Lee and I. Tashev, "High-level feature representation using recurrent neural network for speech emotion recognition," in *INTERSPEECH*, 2015.

[5] A. Satt, S. Rozenberg, and R. Hoory, "Efficient emotion recognition from speech using deep learning on spectrograms." in *INTERSPEECH*, 2017, pp. 1089–1093.

[6] J. Kim and R. A. Saurous, "Emotion recognition from human speech using temporal information and deep learning," in *INTERSPEECH*, 2018.

[7] M. Chen, X. He, J. Yang, and H. Zhang, "3-d convolutional recurrent neural networks with attention model for speech emotion recognition," *IEEE Signal Processing Letters*, vol. 25, pp. 1440–1444, 2018.

[8] M. Neumann and N. T. Vu, "Attentive convolutional neural network based speech emotion recognition: A study on the impact of input features, signal length, and acted speech," in *INTERSPEECH*, 2017.

[9] L. Shu, J. Xie, M. Yang, Z. Li, Z. Li, D. Liao, X. Xu, and X. Yang, "A review of emotion recognition using physiological signals," *Sensors*, vol. 18, no. 7, p. 2074, 2018.

[10] M. El Ayadi, M. S. Kamel, and F. Karray, "Survey on speech emotion recognition: Features, classification schemes, and databases," *Pattern Recognition*, vol. 44, no. 3, pp. 572–587, 2011.

[11] R. Xia and Y. P. Liu, "A multi-task learning framework for emotion recognition using 2d continuous space," *IEEE Transactions on Affective Computing*, vol. 8, pp. 3–14, 2017.

[12] M. Neumann and N. T. Vu, "Improving speech emotion recognition with unsupervised representation learning on unlabeled speech," *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7390–7394, 2019.

[13] J. Snell, K. Swersky, and R. S. Zemel, "Prototypical networks for few-shot learning," in *NIPS*, 2017.

[14] F. Sung, Y. Yang, N. Zhang, T. Xiang, P. H. S. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1199–1208, 2017.

[15] D. Ha, A. Dai, and Q. V. Le, "Hypernetworks," *arXiv preprint arXiv:1609.09106*, 2016.

[16] N. Mishra, M. Rohaninejad, X. Chen, and P. Abbeel, "Meta-learning with temporal convolutions," *arXiv preprint arXiv:1707.03141*, vol. 2, no. 7, 2017.

[17] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 1126–1135.

[18] A. Nichol, J. Achiam, and J. Schulman, "On first-order meta-learning algorithms," *arXiv preprint arXiv:1803.02999*, 2018.

[19] C. Busso, M. Bulut, C.-C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J. N. Chang, S. Lee, and S. S. Narayanan, "Iemocap: Interactive emotional dyadic motion capture database," *Language resources and evaluation*, vol. 42, no. 4, p. 335, 2008.

[20] Z. Li, L. He, J. Li, L. Wang, and W.-Q. Zhang, "Towards discriminative representations and unbiased predictions: Class-specific angular softmax for speech emotion recognition," *Proc. Interspeech 2019*, pp. 1696–1700, 2019.

[21] X. Wu, S. Liu, Y. Cao, X. Li, J. Yu, D. Dai, X. Ma, S. Hu, Z. Wu, X. Liu, and H. M. Meng, "Speech emotion recognition using capsule networks," *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6695–6699, 2019.

[22] P. Li, Y. Song, I. V. McLoughlin, W. Guo, and L. Dai, "An attention pooling based representation learning method for speech emotion recognition," in *INTERSPEECH*, 2018.

[23] L. Tarantino, P. N. Garner, and A. Lazaridis, "Self-attention for speech emotion recognition," *Proc. Interspeech 2019*, pp. 2578–2582, 2019.