Emotion information recovery potential of wav2vec2 network fine-tuned for speech recognition task

Tilak Purohit*†, Mathew Magimai.-Doss *
*Idiap Research Institute, Martigny, Switzerland, †EPFL, École polytechnique fédérale de Lausanne, Switzerland
tilak.purohit@{idiap,epfl}.ch

***Abstract*—Fine-tuning has become a norm to achieve state-of-the-art performance when employing pre-trained networks like foundation models. These models are typically pre-trained on large-scale unannotated data using self-supervised learning (SSL) methods. The SSL-based pre-training on large-scale data enables the network to learn the inherent structure/properties of the data, providing it with capabilities in generalization and knowledge transfer for various downstream tasks. However, when fine-tuned for a specific task, these models become task-specific. Finetuning may cause distortions in the patterns learned by the network during pre-training. In this work, we investigate these distortions by analyzing the network's information recovery capabilities by designing a study where speech emotion recognition is the target task and automatic speech recognition is an intermediary task. We show that the network recovers the task-specific information but with a shift in the decisions also through attention analysis, we demonstrate some layers do not recover the information fully.**

***Index Terms*—Foundation Models, wav2vec2.0, Finetuning, Domain adaptation, Speech Emotion Recognition, ASR**

I. Introduction

When employing Foundation Models (FMs) [1] for the downstream tasks, there are two prevalent approaches: (a) full fine-tuning, involving the updating/tuning of all model parameters, and (b) linear probing, where the entire network is frozen, and only the last linear layer (known as the 'head') is tuned for the target task. In the Independent and Identically Distributed (IID) setting, it is known that fine-tuning outperforms linear probing [2], [3]. Therefore, usually fine-tuning becomes the de facto approach to yield state-of-the-art performance. Despite the prevalent usage of fine-tuning for adapting a FM, a comprehensive understanding of this process is still underway and actively being investigated by the machine learning community [4]–[6].

It is well-known that when a model undergoes self-supervised pretraining, employing either contrastive loss [7] or reconstruction loss, it learns a robust representation of the input modality (e.g., speech). This quality makes pretrained models a valuable choice as a 'universal' feature extractor [8], [9]. However, after adapting to a specific task, these networks often acquire specialization for that adapted task. It was observed that the fine-tuning process has the potential to distort the patterns learned by the pretrained model on a large corpus, resulting in a decline in the quality of the model's generated outputs [6], [10]–[12]. We question, does performing a single round of fine-tuning on a pre-trained network distort its representations to the extent that it becomes unsuitable for further fine-tuning for tasks in different domains?

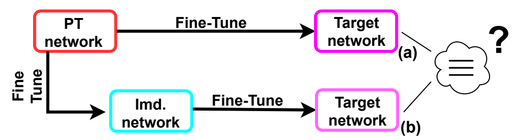

Fig. 1: *Diagram illustrates two pathways to attain the target network, with the same target task. We ask if these target networks (a & b) are equivalent. 'PT' refers to the pre-trained SSL network, and 'Imd.' denotes the intermediary network.*

Researchers in speech community have utilized different speech FMs [13]–[15] and investigated how representations evolve across layers [16]–[19]. Whereas this work aims to investigate the information recovery potential of the FMs. Information recovery, in this context, pertains to the network's capability to attain a comparable state of information encoding. This comparability is evaluated when the pretrained network is directly adapted for a target task or if the pretrained network is initially adapted to an auxiliary task before undergoing further adaptation for the target task, as illustrated in Figure 1.

Previously, it was observed that while fine-tuning the pretrained speech FM, Wav2vec2.0 [13] for Automatic Speech Recognition (ASR) and utilizing these ASR-based embeddings for modeling Emotion Recognition (ER), the incremental improvement in Word Error Rate (WER) achieved through the utilization of more data for fine-tuning ASR corresponds to a gradual decrease in the encoding of paralinguistic information [17], [20], [21]. This corresponds to the fact that as the model gets more task (ASR)-specific it loses paralinguistic feature encoding properties. We utilize these findings to design our study for investigating information recovery in FMs.

II. Methodology and Study design

Figure 2 illustrates our methodology for examining information recovery in the Foundation Models (FMs). In this study, we propose to systematically model and analyze the representations derived from three specific systems:

1) Modeling the 'universal' embeddings derived from the pre-trained FM for the target task.
2) Fine-tuning the pre-trained FM to an intermediary task-specific system, and subsequently extracting and utilizing these representations for the target task.

This work was funded by the SNSF through the Bridge Discovery project EMIL: Emotion in the loop - a step towards a comprehensive closed-loop deep brain stimulation in Parkinson's disease (grant no. $40B2-0\_194794$).

3) Further adapting/fine-tuning the intermediary task-specific network acquired in the previous step to the target task and utilizing the corresponding representations.

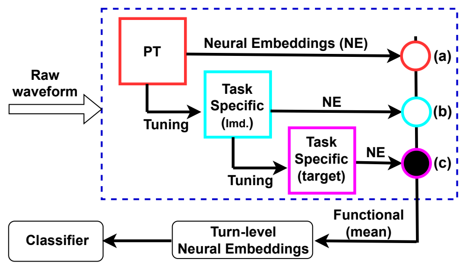

Fig. 2: *Proposed systems: (a) generates pre-trained embeddings, (b) generates intermediary (Imd.) task-specific representation, and (c) generates target task representation. The circles (○) indicate the switching system, while the filled circle (●) denotes the activated switch, directed towards the classifier block.*

In this case study we examine our hypothesis, by considering ASR as our intermediary task and ER as our target task and probe the following questions:

1) Is the loss of paralinguistic information a permanent effect, or is there a possibility of its recovery through additional fine-tuning of the ASR network for the ER task? Furthermore, does this result in the same encoding of paralinguistic information as we would observe if we had fine-tuned the pretrained network directly for the target task?
2) If recovery is possible, what are the implications on the decision-making capabilities of the system compared to the network directly finetuned for the target task?

III. Dataset and Protocols

The Interactive Emotion Dyadic Motion Capture: The Interactive Emotion Dyadic Motion Capture (IEMOCAP) [22] features ten actors (5 male and 5 female) participating in five dyadic sessions. These sessions involve both improvised and scripted scenarios designed to evoke emotional expressions. For consistency with previous studies, we categorized emotions into four classes: *angry*, *happy*, *neutral*, and *sad*. Notably, we merged samples from the *excited* class with the *happy* class, encompassing a total of 5531 utterances.

MSP-Improv Database: The MSP-IMPROV [23] corpus comprises recordings from six spontaneous dyadic sessions involving twelve actors (6 male and 6 female) affiliated with the University of Texas at Dallas. The database is known for its emphasis on naturalness in the recorded interactions. In line with prior studies, we adopt four emotion categories: *angry*, *happy*, *neutral*, and *sad* comprising of a total 7798 utterances.

To adhere to previous works, for each corpus, we use the protocols that have been used in the literature. More precisely, we use the leave-one-session-out methodology. That is, for testing the 'k'-th session, we trained the model on the remaining sessions. We evaluate the performance of the SER systems in terms of Unweighted average recall (UAR) and Weighted average recall (WAR).

IV. Systems and Results

(a) Handcrafted feature representation: We employ ComParE features [24]. Two configurations of ComParE features are utilized: ComParE$_{LLD}$, comprising 130 low-level descriptors ($LLDs$) and their delta functions for frame-level representation, and ComParE$_{LLD\times F}$, consisting of 6373 static turn-level features derived from computing functionals (statistics) over LLD contours. Additionally, we apply the Bag-of-Audio-Words (BoAW) approach implemented in the openXBOW toolkit [25] to extract turn-level representations from the ComParE$_{LLD}$ frame-level representation. In the BoAW approach, 1000 codebook vectors were created, with 500 for the 65 LLDs and 500 for the delta coefficients of 65 LLDs. This system is denoted as BoAW(ComParE$_{LLD}$).

(b) FM based representation: We leverage Wav2vec2.0 [13] representations. The Wav2vec2.0 model adopts a contrastive learning approach, combining it with masking techniques. In this study, we employ the base variant of the model, which includes 12 transformer encoder layers, 768-dimensional hidden states, and 8 attention heads, totaling 95 million parameters. The model underwent pre-training using 960 hours of audio data from the Librispeech corpus [26]. For this study, we investigate four variations of Wav2vec2.0 :

1) The default pre-trained network, identified as PT.
2) PT fine-tuned specifically for our target task of emotion recognition, denoted by SER.
3) PT fine-tuned for the intermediate ASR task with three configurations, each based on the amount of data used for the fine-tuning process: (a) Fine-tuned with 10 minutes of LibriSpeech data (ASR10), (b) Fine-tuned with 100 hours of LibriSpeech data (ASR100), and (c) Fine-tuned with 960 hours of LibriSpeech data (ASR960).
4) Networks derived from the intermediate ASR task further adapted for our target task (ER), labeled as ASR(x)→SER, where x represents the different ASR models as mentioned above.

ASR-based fine-tuned model checkpoints were retrieved from HuggingFace, ASR10 [27] with a WER of 57.81%, ASR100 [28] with a WER of 6.1%, and ASR960 [29] with a WER of 3.4% on the clean set of Librispeech. To fine-tune Wav2vec2.0 for Speech Emotion Recognition (SER), we follow the default S3PRL [8] configuration with minor adjustments. The learning rate is set to 1.0×10^{-5}, using the cross-entropy loss function, the batch size is 4, gradient accumulation is configured at 8, and a random seed value of 1337 is utilized. During Wav2vec2 fine-tuning, the convolution-based encoder blocks are kept frozen, while all the 12 transformer encoder blocks are fine-tuned.

Support Vector Machine (SVM) was utilized as a classifier. We performed hyperparameter tuning for the classifiers associated with handcrafted features using the grid search. For neural embeddings we maintain a consistent linear kernel for SVM, but optimize the values of C and γ parameters through grid search, employing a 5-fold cross-validation split. This approach ensures a fair comparison across various embedding spaces.

TABLE I: *Comparison of different feature representations for emotion recognition on two evaluation corpora.*

Feature representation	Dim.	Evaluation corpus			
		Iemocap (4-class)		Msp-Improv (4-class)	
		UAR ↑	WAR ↑	UAR ↑	WAR ↑
G-1: Baseline Features					
ComParE$_{LLD \times F}$	6373	58.00	56.51	43.10	55.90
BoAW(ComParE$_{LLD}$)	500/500	57.67	56.62	43.30	55.60
G-2: Pretrained network embeddings					
PT	768	56.76	56.26	47.01	58.49
G-3: Task specific fine-tuning network embeddings					
SER	768	**64.98**	**63.89**	**56.54**	63.41
ASR10	768	55.18	53.59	41.42	58.10
ASR100	768	60.03	58.09	49.25	60.44
ASR960	768	49.38	49.34	36.51	62.22
G-4: 2 step fine-tuning network embeddings					
ASR10→SER	768	60.93	59.52	52.80	59.91
ASR100→SER	768	64.59	63.68	56.29	**63.99**
ASR960→SER	768	63.57	62.56	55.54	62.72

A. System performance

From Table I it is evident that the SER network outperforms other systems across both databases, as anticipated due to its specific optimization for the ER task. There is a significant enhancement in ER task performance when transitioning from ASR10 to ASR100, characterized by a lower WER for ASR100. However, a subsequent decline in ER performance is observed with ASR960, even with it being a superior ASR system with the lowest WER. This observation aligns with findings reported in prior literature [20]. Upon comparing the performance of PT and ASR100, it becomes evident that the incorporation of phonetic information in the ASR network is, to some extent, more beneficial than the vanilla pre-trained network for the ER task. But, as the ASR network is further optimized for improved ASR performance, there is a notable decline in SER performance. This reaffirms that as the model becomes more optimized for ASR, it tends to lose paralinguistic information which might be undesirable for ASR. Examining the G-4 section of Table I, all the three ASR systems, when further fine-tuned for the SER task, achieved comparable performance to SER in the G-3 section of the table. We do not observe any performance gain through using ASR-based initialization in a two-step fine-tuning process. Nevertheless, the network proficiently regains emotion information, with ASR100→SER demonstrating the most effective recovery, while ASR10→SER exhibits the least recovery. It is worth mentioning that both direct fine-tuning and two-step fine-tuning outperform handcrafted features in terms of performance. Additionally, our results align with previously reported figures in the literature for both Iemocap [17], [21], [30] and MSP [20], [23], [31]. It is important to emphasize that our primary focus is not on competing for state-of-the-art results. In this study, we compare and analyze various systems using a similar parameter setup to explore the information recovery potential of FMs.

TABLE II: *Comparision of decision missmatch between the predictions of* SER *and* ASR(X)→SER *network.*

Feature representation	Iemocap	Msp-Improv
	Miss-Match %	Miss-Match %
ASR10→SER	35.18	34.03
ASR100→SER	27.57	25.10
ASR960→SER	29.66	28.89

V. Analysis

A. Effects of two-step fine-tuning on decision outcomes

To further investigate the information recovery within the two-step fine-tuned systems (ASR(x)→SER), we examined the decision outcomes of these systems. We computed the decision mismatches between the predicted labels of the SER network and the ASR(x)→SER networks. The mismatch values are presented in Table II. At first glance, it is noticeable that there is a mismatch of more than 25% for all the networks. It is interesting to point out, in spite of the seemingly close UAR values between ASR100→SER (64.59%) and SER (64.98%) in Table I, there exists a more than 25% discrepancy in their decision outcomes. Consequently, we do observe a discernible shift in the decision properties.

B. Latent space analysis

In addition to examining the decision outcomes, we analyzed the embedding space by comparing the last layer representations of SER and ASR(x)→SER systems. For this analysis we resorted to the cosine distance formulation, which helps measure the similarity between two non-zero vectors. We calculated the cosine distance between the embeddings generated from SER and ASR(x)→SER systems for all data points in both the corpora used in the study. Figure 3 showcases the distribution of cosine distances for all data points using a line-joined histogram plot. The markers (e.g. -●-) on the curves in the subplots of Figure 3 represent the center of each of the 10 bins for their respective systems. The distribution of cosine distances between SER and ASR100→SER (denoted by -×- curve) reveals that the majority of data points exhibit lower distances, indicating a higher degree of similarity. In contrast, the diamond head curve (-♦-) representing cosine distance distribution between SER and ASR960→SER shows lower similarity. These observations align with the results presented in Table I. To better assess the alignment of cosine distance with the decisions, we computed decision matches between the predictions of SER and ASR(X)→SER for data points falling within a defined distance threshold. This threshold is represented by a dashed magenta line in the subplots of Figure 3,

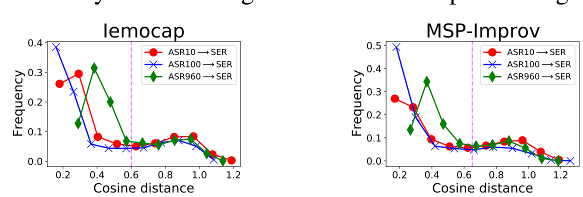

Fig. 3: *Distribution of cosine distance values computed for the last layer representation between* SER *and* ASR(x)→SER *network, for all the data points in the corpus. Vertical dashed magenta line indicates the threshold value.*

(a) Comparing: SER with ASR100 & ASR100→SER

(b) Comparing: SER with ASR960 & ASR960→SER

Fig. 4: *Bhattacharyya distance comparison (using Equation1) for the attention heads at different layers for different systems.*

with values set at 0.6 for Iemocap and 0.65 for Msp-Improv. The threshold is chosen to encompass more than 70% of the total data points, while maintaining a small cosine distance for analysis. Table III provides details on the percentage of data points falling within the cosine distance threshold for different systems, along with the decision matches between the SER prediction and ASR(X)→SER prediction for those encompassing data points. The values in the match percentage column of Table III suggest that the networks can effectively recover the information to a considerable extent. It is important to highlight that there were a few data points where the cosine distance exceeded one, indicating complete dissimilarity. Upon examining the decision matches, no matches were found for these points. Instances where the cosine distance exceeded one were primarily associated with ASR10→SER (-•- curve), constituting 3.6% of the total data from Iemocap and 5.5% from Msp-Improv. Conversely, the least occurrences were noted with ASR100→SER (-×- curve), accounting for 1.5% in Iemocap and 2% in Msp-Improv.

TABLE III: *% of data falling within the cosine distance threshold, along with the corresponding decision match %.*

	Iemocap		Msp-Improv	
Feature representation	**Data %**	**Match %**	**Data %**	**Match %**
ASR10→SER	70.85	89.00	70.00	85.63
ASR100→SER	75.55	90.69	81.98	88.75
ASR960→SER	70.01	94.07	73.04	89.69

C. Attention head analysis

We extract self-attention weights for each head in every layer of the Wav2vec2.0 model for a particular input audio. This results in a 2D float-type array of shape $N \times N$, where N is the frame-wise sequence length of the input audio. This 2D representation is referred to as the self-attention map (SAM). Each row in the SAM is a probability distribution representing the attention logits for a specific element to all other elements in the sequence. For our analysis of SAMs, we make use of Equation 1, which computes the Bhattacharyya distance (BC) [32], providing a symmetric distance measure between the rows of the SAMs generated by different systems to be compared. Each unit/cell in Figure 4 subplots represents the normalized distance between the rows of SAMs from 2 different systems referred to as sys (E.g. ASR and SER) and are computed using Equation 1, where l and h refers to the transformer layers (= 12) and attention heads (= 12) in Wav2vec2.0 respectively.

$$Dis_{l,h} = \frac{\sum_{i=0}^{N} BC(A_{l,h,i}^{sys}, A_{l,h,i}^{sys})}{N} \tag{1}$$

In Figure 4, subplots (a) and (b) reveal that the attention behaves similarly in the initial layers, as indicated by the low distance values depicted in the plots. However, it is in the last few layers where the attention mechanism becomes more task-specific, as evidenced by the higher distance values. In Figure 4, subplot (a), when comparing ASR100 and SER, we observe that in the last layer, some attention blocks exhibit high distance values, while others have lower distance values. In contrast, in Figure 4, subplot (b), when comparing ASR960 and SER, we see that the last layer attention heads have consistently high distance values. This observation might explain the results in Table I, where ASR100 yields better results compared to ASR960 and even the PT network. It suggests that some attention heads in ASR100 focus on phonetic information, while others concentrate on emotion-related information. For ASR960, the attention does not correspond well to emotional information, as indicated by the high distance values. Upon further fine-tuning of these ASR(X) networks for the ER task (ASR(X)→SER), we observe that the attention heads in the last layer begin to behave similarly to those in the SER network. This is evident from the lower distance values in Figure 4, rightmost subplots for both (a) and (b). It is worth noting that even after adapting the network for the ER task some intermediary layers (e.g., 7, 8) still exhibit high distance values, showing the information was not restored completely; this requires further probing.

VI. Conclusion

Our study explored information recovery potential of FMs using SER and ASR tasks as the target and intermediary tasks, respectively. Initially, the evaluation of intermediary network representations for the target task uncovered an inverse relationship. As the network excelled in the ASR task, it exhibited a decline in SER discrimination properties. However, fine-tuning the intermediary networks for the target task successfully recovered SER information, achieving performance levels comparable (similar) to a network directly tuned for the target task. Despite similar overall performance, we identified disparities in decision-making capabilities between the networks (SER and ASR(X)→SER). Future investigations will explore information recovery using diverse tasks, and the interplay between different learning rates. Additionally, we aim to conduct a layer-wise analysis with different FMs to further enhance our understanding of information recovery in these systems.

REFERENCES

[1] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill *et al.*, "On the opportunities and risks of foundation models," *arXiv preprint arXiv:2108.07258*, 2021.

[2] S. Kornblith, J. Shlens, and Q. V. Le, "Do better imagenet models transfer better?" in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 2661–2671.

[3] X. Zhai, J. Puigcerver, A. Kolesnikov, P. Ruyssen, C. Riquelme, M. Lucic, J. Djolonga, A. S. Pinto, M. Neumann, A. Dosovitskiy *et al.*, "A large-scale study of representation learning with the visual task adaptation benchmark," *arXiv preprint arXiv:1910.04867*, 2019.

[4] A. Kumar, A. Raghunathan, R. Jones, T. Ma, and P. Liang, "Fine-tuning can distort pretrained features and underperform out-of-distribution," *arXiv preprint arXiv:2202.10054*, 2022.

[5] J. Mukhoti, Y. Gal, P. H. Torr, and P. K. Dokania, "Fine-tuning can cripple your foundation model; preserving features may be the solution," *arXiv preprint arXiv:2308.13320*, 2023.

[6] H. Zheng, L. Shen, A. Tang, Y. Luo, H. Hu, B. Du, and D. Tao, "Learn from model beyond fine-tuning: A survey," *arXiv preprint arXiv:2310.08184*, 2023.

[7] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *IEEE computer society conference on computer vision and pattern recognition (CVPR)*, 2005, pp. 539–546.

[8] S.-w. Yang, P.-H. Chi, Y.-S. Chuang, C.-I. J. Lai, K. Lakhotia, Y. Y. Lin, A. T. Liu, J. Shi, X. Chang, G.-T. Lin *et al.*, "SUPERB: Speech Processing Universal PERformance Benchmark," in *Proc. of Interspeech*, 2021, pp. 1194–1198.

[9] A. Mohamed, H.-y. Lee, L. Borgholt, J. D. Havtorn, J. Edin, C. Igel, K. Kirchhoff, S.-W. Li, K. Livescu, L. Maaløe *et al.*, "Self-supervised speech representation learning: A review," *IEEE Journal of Selected Topics in Signal Processing*, 2022.

[10] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," in *Psychology of learning and motivation*. Elsevier, 1989, vol. 24, pp. 109–165.

[11] R. M. French, "Catastrophic forgetting in connectionist networks," *Trends in cognitive sciences*, vol. 3, pp. 128–135, 1999.

[12] K. Lee, K. Lee, J. Shin, and H. Lee, "Overcoming catastrophic forgetting with unlabeled data in the wild," in *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 312–321.

[13] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in neural information processing systems*, vol. 33, pp. 12 449–12 460, 2020.

[14] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao *et al.*, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, pp. 1505–1518, 2022.

[15] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[16] A. Pasad, J.-C. Chou, and K. Livescu, "Layer-wise analysis of a self-supervised speech representation model," in *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2021, pp. 914–921.

[17] L. Pepino, P. Riera, and L. Ferrer, "Emotion Recognition from Speech Using wav2vec 2.0 Embeddings," in *Proc. of Interspeech*, 2021, pp. 3400–3404.

[18] A. Pasad, C.-M. Chien, S. Settle, and K. Livescu, "What do self-supervised speech models know about words?" *Transactions of the Association for Computational Linguistics*, vol. 12, pp. 372–391, 2024.

[19] W. Wu, C. Zhang, and P. C. Woodland, "Self-supervised representations in speech-based depression detection," in *Proc. of ICASSP*, 2023, pp. 1–5.

[20] T. Purohit, B. Vlasenko, and M. Magimai.-Doss, "Implicit phonetic information modeling for speech emotion recognition," in *Proc. of Interspeech*, 2023.

[21] Y. Li, Y. Mohamied, P. Bell, and C. Lai, "Exploration of a self-supervised speech model: A study on emotional corpora," in *IEEE Spoken Language Technology Workshop (SLT)*, 2023, pp. 868–875.

[22] C. Busso, M. Bulut, C.-C. Lee, A. Kazemzadeh, E. Mower *et al.*, "IEMOCAP: Interactive emotional dyadic motion capture database," *Language resources and evaluation*, vol. 42, pp. 335–359, 2008.

[23] C. Busso, S. Parthasarathy, A. Burmania, M. AbdelWahab, N. Sadoughi, and E. M. Provost, "MSP-IMPROV: An acted corpus of dyadic interactions to study emotion perception," *IEEE Transactions on Affective Computing*, vol. 8, pp. 67–80, 2017.

[24] B. Schuller, S. Steidl, A. Batliner, A. Vinciarelli, K. Scherer, F. Ringeval, M. Chetouani, F. Weninger, F. Eyben, E. Marchi *et al.*, "The INTERSPEECH 2013 computational paralinguistics challenge: Social signals, conflict, emotion, autism," in *Proc. of Interspeech*, 2013.

[25] M. Schmitt and B. Schuller, "openXBOW - Introducing the Passau Open-Source Crossmodal Bag-of-Words Toolkit," *Journal of Machine Learning Research*, 2017.

[26] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," in *Proc. of ICASSP*, 2015, pp. 5206–5210.

[27] Huggingface, "w2v2-libri-10min," https://huggingface.co/Xinnian/w2v2-libri-10min.

[28] ——, "wav2vec2-base-100h," https://huggingface.co/facebook/wav2vec2-base-100h.

[29] ——, "wav2vec2-base-960h," https://huggingface.co/facebook/wav2vec2-base-960h.

[30] T. Purohit, S. Yadav, B. Vlasenko, S. P. Dubagunta, and M. Magimai.-Doss, "Towards Learning Emotion Information from Short Segments of Speech," in *Proc. of ICASSP*, 2023, pp. 1–5.

[31] M. Neumann and N. T. Vu, "Improving speech emotion recognition with unsupervised representation learning on unlabeled speech," in *Proc. of ICASSP*, 2019, pp. 7390–7394.

[32] A. Bhattacharyya, "On a measure of divergence between two statistical populations defined by their probability distribution," *Bulletin of the Calcutta Mathematical Society*, vol. 35, pp. 99–110, 1943.