Enhancing Speech Emotion Recognition through Knowledge Distillation

Trung Minh Nguyen
AiTA Lab,
Dept. of Computing Fundamental
FPT University
Ho Chi Minh City, Vietnam
trungnmse182406@fpt.edu.vn

Phuong-Nam Tran
Networking Intelligence Lab,
Dept. of Computer Science and Engineering
Kyung Hee University
Yongin-si, Gyeonggi-do, Republic of Korea
tpnam0901@khu.ac.kr

Duc Ngoc Minh Dang*
AiTA Lab,
Dept. of Computing Fundamental
FPT University
Ho Chi Minh City, Vietnam
ducdnm2@fe.edu.vn

***Abstract*—The importance of Speech Emotion Recognition (SER) is growing across diverse applications, which has resulted in the development of multiple methodologies and models to improve SER performance. Nevertheless, some modern SER models require significant processing resources and exhibit poor performance, making them unsuitable for real-time applications. To address this, we propose a novel approach that leverages Knowledge Distillation (KD) to create lightweight student models derived from the 3M-SER architecture. Our method focuses on compressing the text embedding component by replacing BERT$_{\text{BASE}}$ with smaller variants while maintaining VGGish for audio embedding. Experiments conducted on the IEMOCAP dataset demonstrate that our student model, which reduces model size by up to 44.9%, achieves performance remarkably close to that of the teacher model while improving inference time by up to 40.2% when trained with KD. These results underscore the effectiveness of KD in creating efficient and accurate SER models suitable for resource-constrained environments and real-time applications. Our work contributes to the ongoing effort to make advanced SER technology more accessible and deployable in practical settings.**

***Index Terms*—Speech Emotion Recognition, Knowledge Distillation**

I. Introduction

In recent years, Speech Emotion Recognition (SER) has garnered significant attention and undergone substantial development, especially its potential to evaluate customer emotions during online service consultations. In addition, SER also has diverse applications in healthcare, entertainment, e-learning, and human-computer interaction [1], which take advantage of emotion recognized by the deep learning model through voice analysis to enhance the user experience.

In the early stage, SER systems mainly relied on traditional audio features such as zero crossing rate, pitch, and Mel-Frequency Cepstral Coefficients (MFCCs) to recognize emotions [2]. With the introduction of deep learning, the performance of emotion recognition is improved by passing these features through a deep network such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). For example, CNNs can be applied to MFCCs to extract intricate patterns related to emotional states as used in [3]. In addition, Zhang *et al.* [4] proposed a method that merges a two-branch CNN to capture deep features from both audio and MFCCs. In 2017, Google created a new audio model called VGGish [5], which uses a CNN-based approach to convert audio signals into vector features in a latent space. This transformation involves converting each second of audio into a spectrogram image using log Mel-spectrograms.

Another approach in SER is to focus on extracting text features from spoken language. In text-based SER, several techniques are employed to enhance emotion recognition. One such technique is word embeddings, which capture the meaning of words in their context, allowing for a more nuanced understanding of the text. Traditional word embeddings like Word2Vec [6] and GloVe [7] generate a single, static vector for each word. This vector remains the same regardless of the context in which the word appears, which can limit their ability to differentiate between different meanings of the same word based on context. Contextual embeddings have been introduced to address the limitations of traditional word embeddings. These embeddings generate dynamic vectors that vary depending on the word's context, capturing more nuanced and contextually relevant meanings. Among the most advanced and influential techniques in this domain is BERT [8] (Bidirectional Encoder Representations from Transformers). By utilizing a self-attention mechanism and pre-training on large text datasets, BERT learns contextual feature vectors for words and sentences. This enables BERT to understand complex language patterns and extract meaningful textual embeddings, enhancing the accuracy and effectiveness of emotion recognition tasks.

Recently, researchers have explored combining audio and text features for more robust SER systems. The SERVER model [9] which integrates BERT [8] for text feature extraction and VGGish [5] for audio feature extraction. While SERVER achieves better accuracy by considering both audio and text, it doesn't fully explore their interconnections, which might limit its potential. To address this, Tran *et al.* introduced enhancements such as 3M-SER [10] further to refine the integration of audio and text features and integrating feature loss function [11] into 3M-SER to capture more nuanced emotional information by adjusting fusion feature vectors. Expanding on these developments, the MERSA [12] model incorporates self-align embeddings into the feature extractor and utilizes cross-

* *Corresponding author: Duc Ngoc Minh Dang (ducdnm2@fe.edu.vn)*

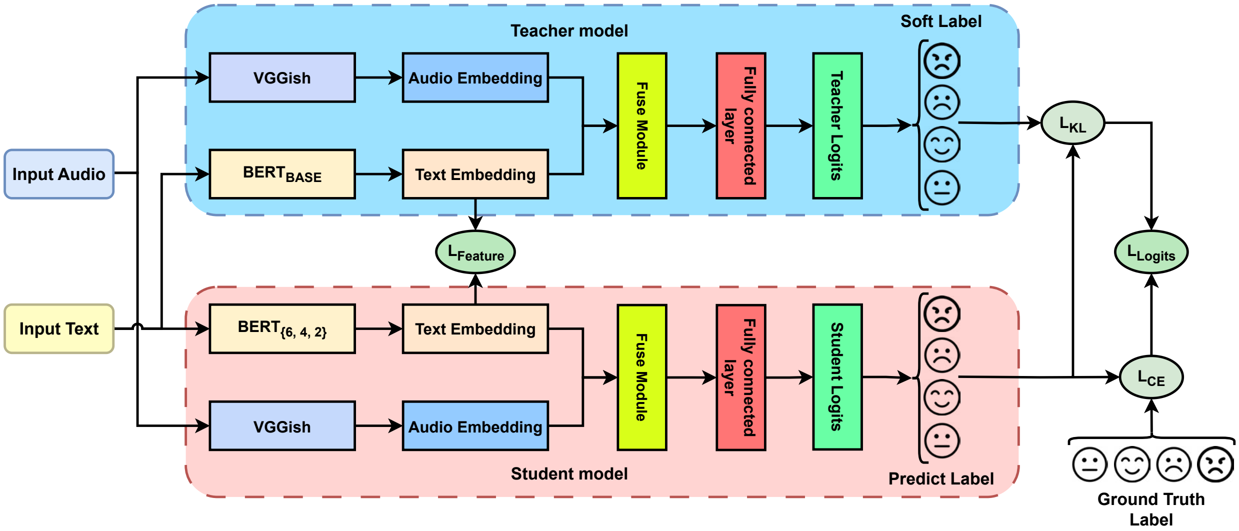


Fig. 1: Knowledge Distillation process for Multi-modal Speech Emotion Recognition

attention to combine text and audio features. By integrating multi-modal information more effectively, MERSA enhances the model's ability to process and respond to textual and auditory inputs coherently. This improvement contributes to better performance in tasks that require the integration of both modalities.

While models such as SERVER, 3M-SER, and MERSA have shown significant improvements in accuracy, they often result in high model complexity and high computational cost, which limits their practical application in real-time scenarios. To address this problem, we propose leveraging Knowledge Distillation (KD) [13] to enhance the smaller variants of the 3M-SER model. KD is a technique that transfers knowledge from a large model (teacher) to a smaller model (student), aiming to minimize the student's error and achieve a balance between model size and performance. By applying KD, our approach provides a compact and efficient model for real-world emotion detection tasks without compromising accuracy.

Our contributions can be summarized as follows:

- We replace $\text{BERT}_{\text{BASE}}$ as a text embedding extractor in the teacher model 3M-SER with its smaller variants by reducing the number of hidden layers while retaining VGGish for extracting audio embedding to build out student models.
- We use KD with logits-based and feature-based loss to transfer knowledge from the teacher to student models, allowing the smaller models to achieve comparable performance.
- Our student models achieve up to a 44.9% reduction in total parameters and a 40.2% reduction in inference time while maintaining performance closely aligned with the teacher model in SER tasks.

The rest of this paper is organized as follows: Section II provides a detailed description of the architecture of our student models and outlines the knowledge distillation process employed during their training. In Section III, we present the experimental results and analysis conducted on the SER dataset, evaluating the performance of the proposed model. Finally, in Section IV, we summarize our findings and discuss potential future directions for this research.

II. Methodology

A. Knowledge Distillation in Multi-modal SER

This section presents our technique for using KD to transfer knowledge in a multi-modal SER system, as illustrated in Figure 1. The following subsection details the architectures of the teacher and student models. As shown in Figure 1, the teacher and student models first process the input separately. We soften the teacher outputs because they contain more information about the relative probabilities of different classes, providing a richer signal than complex labels. Next, the loss function is applied to transfer knowledge from the teacher model to the student model. It consists of two components: logits-based loss and feature-based loss. The logits-based loss component compares the probability distributions of the teacher and student models and the student's predictions with the ground truth labels to encourage the student to mimic the teacher's predictions while learning from the true labels. The feature-based loss component compares the text embeddings of the teacher and student models. This encourages the student model to learn similar internal representations as the teacher. We aim to transfer the teacher's ability to extract meaningful text representations to the student model by aligning these intermediate features. Finally, these loss components are combined to form the total loss function.

B. Teacher and student models

In this study, we choose the 3M-SER model [10] as a teacher model based on its performance and its system designed to leverage the strengths of both textual and audio embeddings to classify emotional states.

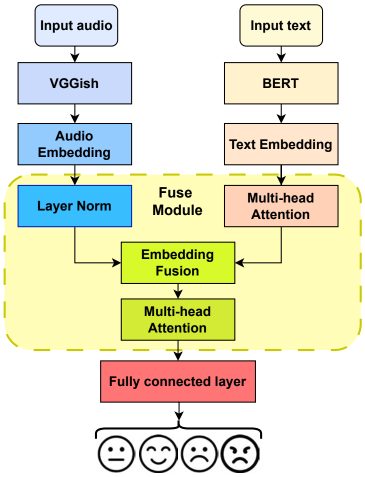


Fig. 2: Architecture of the 3M-SER model

As shown in Figure 2, the 3M-SER model employs a dual-stream architecture, which processes text and audio inputs independently before fusing them. To process the audio input, 3M-SER employs VGGish [5]. This CNN model has been pre-trained on extensive audio datasets to transform audio signals into log-mel spectrograms and effectively extract important audio features. For text processing, the model utilizes BERT [8] to comprehend the meaning of words in different contexts, resulting in robust text feature extraction. Once the audio and text features are extracted, 3M-SER employs multi-head attention to fuse these features. This fusion process utilizes the attention mechanism, which allows the model to focus on different aspects of the audio and text features. By doing so, the model can identify the most relevant information for accurate emotion detection. Furthermore, 3M-SER incorporates layer normalization in the audio feature to stabilize the fusion process between the text and audio features. This technique ensures a smooth integration of information from both modalities, enhancing the overall performance of the system as discussed in [10].

To address the computational challenges of large-scale SER models and to create a more efficient system for real-time applications, we propose a student model derived from the 3M-SER [10] teacher model. The primary breakthrough in our student model is in the text embedding component. The teacher model utilizes BERT, a large-scale language model consisting of multiple layers of Transformer blocks. Specifically, the teacher model uses BERT$_{\text{BASE}}$, which contains 12 layers. In contrast, our student model investigates the use of a smaller BERT by reducing its hidden layers to 6, 4, and 2, resulting in the creation of BERT$_6$, BERT$_4$, and BERT$_2$, while retaining the original hidden size of 768 and 12 attention heads. Table I provides a detailed comparison of these variants regarding hidden layers and total parameters. For audio feature extraction, our student models continue using the VGGish component in the teacher model.

TABLE I: Comparison of BERT and its lightweight variants

Text embedder	Hidden layers	Params (M)
BERT$_{\text{BASE}}$	12	85.65
BERT$_6$	6	43.12
BERT$_4$	4	28.94
BERT$_2$	2	14.77

C. Training objective

When employing knowledge distillation for training a student model, we incorporate two types of loss: logits-based and feature-based loss.

Logits-based loss (L_{Logits}) aims to transfer the softened logits from the teacher to the student model while learning from the true labels. To soften the model logits, we take temperature scaling on the logits, a widely used calibration technique [14], which is calculated as follows:

$$P_k = softmax(Z_k/T) \tag{1}$$

where, P_k and Z_k is the soften logits and logits of k-th sample of our model, respectively, and T is the temperature scaling parameter.

To transfer the softened logits from the teacher model to the student model, we utilize the Kullback-Leibler divergence loss function, denoted as L_{KL}. This loss function measures the similarity between the emotional probability distributions of the soft labels obtained from the teacher model and the soft predictions generated by the student model. By minimizing this loss, we encourage the student model to generate probability distributions that closely resemble those of the teacher, therefore successfully transferring the teacher's knowledge to the student. The L_{KL} is defined as follows:

$$L_{KL} = \frac{1}{N}\sum_{k=1}^{N} KL(P_k^t \parallel P_k^s) \tag{2}$$

where N represents the number of samples, P_k^t and P_k^s are respectively the softened probability distribution of k-th sample from the teacher and student model calculated using the Eq. 1, $KL(.)$ denotes the Kullback-Leibler divergence of two components.

Although the L_{KL} helps student models replicate the teacher's output, it is challenging for the student model to reproduce the same features as the teacher due to architectural differences. Moreover, the convergence point of student models may differ from that of the teacher model, leading to poor performance. To address these challenges, cross-entropy loss (L_{CE}) is also employed in the training process of student models. This loss function supports student models in finding their optimal convergence point. The student models can learn more effectively from the provided examples by directly measuring the disparity between the predicted probabilities and the actual labels. We can define L_{CE} as follows:

$$L_{CE} = -\frac{1}{N}\sum_{k=1}^{N} y_k \log y_k^s \tag{3}$$

where y_k^s is the predicted probability of the k-th sample using the softmax function generated by the student model, and y_k is the ground truth label of the k-th sample.

From Eqs. 2 and 3, the L_{Logits} can be defined as follows:

$$L_{Logits} = \alpha L_{KL} + (1 - \alpha) L_{CE} \tag{4}$$

where α is a weighting factor that balances the two components.

Feature-based loss ($L_{Feature}$) ensures that the student model's text embeddings closely align with the teacher's. We use the mean squared error (MSE) between the teacher's and student's text feature embeddings. This MSE function measures the average squared difference between the text embeddings produced by the teacher and student models. By minimizing this loss, we encourage the student model to learn text representations similar to the teacher model. The $L_{Feature}$ is defined as follows:

$$L_{Feature} = \frac{1}{N} \sum_{k=1}^{N} \frac{\| f_k^t - f_k^s \|^2}{h} \tag{5}$$

where h is the size (number of elements) of the text embedding vector, f_k^t and f_k^s represent the text embedding generated by the teacher and student models at k-th sample, respectively.

By using both L_{Logits} and $L_{Feature}$, we encourage the student model to not only replicate the final output of the teacher and learn from truth labels but also acquire comparable intermediate representations. The total loss function (L_{Total}) for training the student model is a combination of the logits-based and feature-based loss, which is defined as follows:

$$L_{Total} = L_{Logits} + L_{Feature} \tag{6}$$

D. Performance metrics

Evaluating the performance of a SER system involves using various metrics to assess its accuracy and effectiveness comprehensively. In this study, we utilize four key metrics:

1) Accuracy (ACC): This fundamental metric measures the proportion of correct predictions out of the total predictions made. It is calculated as follows:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{7}$$

where TP, TN, FP, and FN represent true positive, true negative, false positive, and false negative samples, respectively.

2) Balanced Accuracy: This metric overcomes the limitations of ACC by treating each class equally, making it better suited for imbalanced datasets. Balanced Accuracy ($BACC$) ensures fair performance measurement across all classes, defined as the average recall for each class:

$$BACC = \frac{1}{2} \left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP} \right) \tag{8}$$

3) Macro F1-Score: This is another metric that is particularly useful for imbalanced datasets. It independently calculates the unweighted mean of precision and recall scores for each class. The formula of Macro F1-Score ($MF1$) is as follow:

$$MF1 = \frac{1}{C} \sum_{i=1}^{C} F1_i \tag{9}$$

where C is total number of classes, $F1_i$ is the F1-Score of i-th class. The F1-Score for each class is the harmonic mean of precision and recall:

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{10}$$

4) Weighted F1-Score: This metric is similar to the Macro F1-Score metric but considers the support (the number of true instances) of each class. It is computed as:

$$WF1 = \frac{1}{C} \sum_{i=1}^{C} w_i \cdot F1_i \tag{11}$$

where w_i is the proportion of the true instances of i-th class relative to the total instances. This metric ensures that the F1-Score of each class contributes proportionally to its occurrence in the dataset, providing a more realistic evaluation for imbalanced datasets.

III. Result and Discussion

A. Dataset

To assess the performance of our model, we utilized the Interactive Emotional Dyadic Motion Capture (IEMOCAP) [15] dataset for training and testing. This dataset comprises approximately 12 hours of audiovisual data, including recorded speech and transcriptions. Our study focused on four primary emotional states: neutral, sadness, happiness, and anger. The dataset consisted of 1,708 neutral samples, 1,084 sadness samples, 1,636 happiness samples, and 1,103 anger samples, divided into the train, validation, and test datasets. The distribution of these emotional states can be visualized in Figure 3.

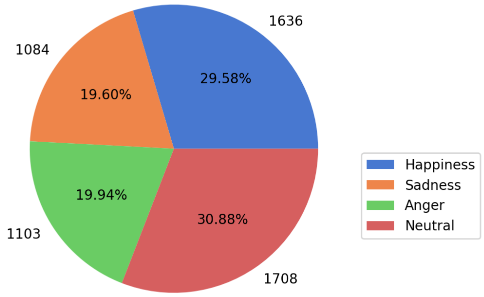


Fig. 3: Distribution of emotional states in IEMOCAP dataset

TABLE II: Performance and efficiency comparison of teacher and student models on IEMOCAP dataset

Methods	**Total**	**Inference**		**Validation (%)**				**Test (%)**		
	Params (M)	**Time (ms)**	$BACC$	ACC	$MF1$	$WF1$	$BACC$	ACC	$MF1$	$WF1$
BERT$_{\text{BASE}}$-VGGish (Teacher)	157.91	32.80	81.80	80.32	80.80	80.15	81.24	80.69	80.93	80.69
BERT$_6$-VGGish	115.38	23.38	78.12	78.31	78.83	78.48	76.32	76.71	77.06	76.78
BERT$_6$-VGGish + KD			**80.36**	**79.32**	**79.88**	**79.18**	**78.92**	**78.52**	**78.71**	**78.45**
BERT$_4$-VGGish	101.21	21.46	78.60	77.51	78.13	77.30	76.93	76.53	76.65	76.49
BERT$_4$-VGGish + KD			**78.85**	**78.31**	**78.93**	**78.41**	**79.03**	**78.34**	**78.88**	**78.35**
BERT$_2$-VGGish	87.03	19.60	78.19	77.11	77.85	76.96	76.90	76.90	76.83	76.73
BERT$_2$-VGGish + KD			**79.48**	**77.91**	**78.53**	**77.56**	**79.26**	**78.16**	**78.34**	**77.93**

B. Experimental setup

All the experiments are conducted on Kaggle using two NVIDIA T4 GPUs with 16 GB VRAM. The Stochastic Gradient Descent [16] optimizer is utilized with an initial learning rate of 10^{-6}. After every 30 epochs, the learning rate is reduced by a factor of 10. For the logits-based loss function, The optimal hyperparameters based on [13] are utilized which set the temperature scaling of 2 and the logits-based loss weighting factor to 0.5. The training process consisted of two stages: transfer learning and fine-tuning. In the transfer learning stage, the model is initialized with pre-trained weights and trained for 50 epochs. After that, the best weight on the evaluation dataset obtained from transfer learning is used for continuing training in the fine-tuning phase. During fine-tuning, we unfroze all layers of the text processing component. However, the audio processing component remained frozen, meaning its weights were not updated during fine-tuning. This approach allowed us to focus on improving the text-processing part of the model while keeping the audio-processing part consistent.

C. Comparisons and Results

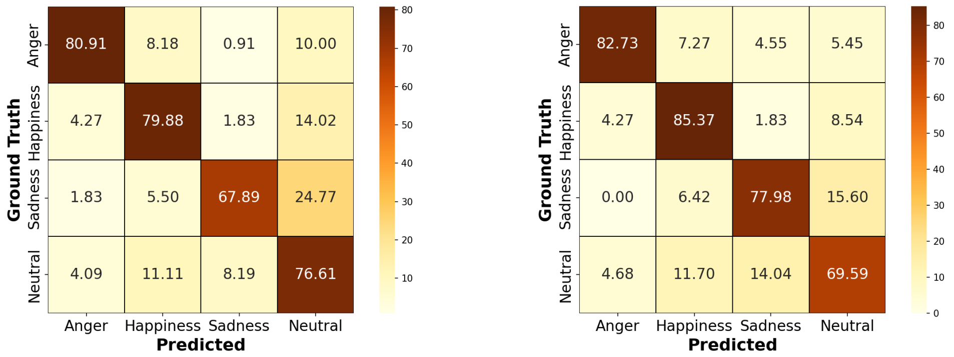
Table II presents the performance comparison between the teacher model, 3M-SER, and our proposed student models trained on the IEMOCAP dataset. As we progressively compressed the hidden layers of the BERT$_{\text{BASE}}$ models to reduce their size, a corresponding decline in performance was observed. This decline is expected, as the reduction in model complexity limits the ability of the student models to capture and represent rich features. However, after applying the KD technique, which incorporates logits-based and feature-based loss, we observed a significant improvement in the performance of these compressed student models on both the validation and test datasets. Specifically, our student models using BERT$_6$, BERT$_4$, and BERT$_2$ for text embedding with KD achieved performance improvements of 1.59%, 1.37%, and 1.21%, respectively, on average across all performance metrics for the validation and test datasets, nearly matching the performance of the teacher model. This demonstrates that KD effectively transfers knowledge from the teacher model to the student models, enabling them to retain critical information and maintain a higher level of performance despite their reduced size.

On the other hand, BERT$_2$-VGGish has roughly half the number of parameters compared to the teacher model, with 87.03 million versus 157.91 million parameters. Despite this reduction, BERT$_2$-VGGish maintains commendable performance compared with the teacher model. Furthermore, it achieves an impressive average inference time of 19.60 milliseconds per sample on the test dataset, compared to 32.80 milliseconds for the teacher model. In this experiment, the design of the student model only considers the size of the hidden layer of the text embedding. The feature size is fixed to match the teacher's feature size to ensure compatibility during the training of the student model. This constraint makes it challenging to reduce the model's parameters while maintaining its performance.

Figure 4 presents the confusion matrices on the test dataset for BERT$_6$-VGGish student model. The results demonstrate that integrating KD into BERT$_6$-VGGish student model improves the overall classification performance across most emotion categories. Specifically, the BERT$_6$-VGGish method with KD shows higher accuracy in predicting ”Anger”, ”Happiness”, and ”Sadness” emotions, with reduced misclassification rates in these categories. However, the baseline model without KD outperforms in recognizing ”Neutral” emotions, suggesting that while KD enhances the model's ability to distinguish more distinct emotional states, it may introduce some ambiguity in classifying more subtle emotions like ”Neutral”.

IV. Conclusions and Future Work

In conclusion, this study presents a highly effective method for multi-modal SER by employing KD, resulting in notable improvements in both model accuracy and computational efficiency. We leverage the 3M-SER model (using the BERT$_{\text{BASE}}$-VGGish method) as the teacher model and replace BERT$_{\text{BASE}}$ with its lightweight variants such as BERT$_6$, BERT$_4$, and BERT$_2$ for text embedding to build out student models. Experiments on the IEMOCAP dataset show that our student models, which reduce model size by up to 44.9%, achieves performance comparable to the teacher model while improving inference time by up to 40.2% when trained with KD. This combination of performance and compactness emphasizes the promise of KD in constructing more smaller yet powerful SER models, making them highly suitable for deployment in real-



(a) Without Knowledge Distillation.

(b) With Knowledge Distillation.

Fig. 4: Confusion matrices for BERT$_6$-VGGish student model

world applications where speed and resource efficiency are crucial.

Our results demonstrate the successful development of lightweight student models that maintain the accuracy of the original 3M-SER while substantially improving their speed and suitability for real-time applications. However, further improvements are needed to meet the full requirements of real-time applications. Future research will focus on reducing the number of parameters in both the audio and text extraction components. Additionally, reducing the text feature size will be explored to further decrease the text extraction model size. Furthermore, experimentation with different preprocessing techniques will be conducted to identify optimal solutions for processing continuous audio input, which is crucial for real-time SER applications. Addressing these aspects will further enhance the speed and efficiency of our models, making them more suitable for real-time deployment.

References

[1] A. Mikuckas, I. Mikuckiene, A. Venčkauskas, E. Kazanavicius, R. Lukas, and I. Plauska, "Emotion recognition in human computer interaction systems," *Elektronika ir Elektrotechnika*, vol. 20, pp. 51–56, 12 2014.

[2] M. J. Al-Dujaili and A. Ebrahimi-Moghadam, "Speech emotion recognition: A comprehensive survey," *Wireless Personal Communications*, vol. 129, no. 4, pp. 2525–2561, Apr 2023.

[3] A. M. Badshah, J. Ahmad, N. Rahim, and S. W. Baik, "Speech emotion recognition from spectrograms with deep convolutional neural network," in *2017 International Conference on Platform Technology and Service (PlatCon)*, 2017, pp. 1–5.

[4] J. Zhao, X. Mao, and L. Chen, "Learning deep features to recognize speech emotion using merged deep cnn," *IET Signal Processing*, vol. 12, 02 2018.

[5] S. Hershey, S. Chaudhuri, D. P. W. Ellis, J. F. Gemmeke, A. Jansen, C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. Weiss, and K. Wilson, "Cnn architectures for large-scale audio classification," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017.

[7] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Conference on Empirical Methods in Natural Language Processing*, vol. 14, 01 2014, pp. 1532–1543.

[6] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in Neural Information Processing Systems*, vol. 26, 10 2013.

[8] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *North American Chapter of the Association for Computational Linguistics*. Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186.

[9] N. T. Pham, D. N. M. Dang, B. N. H. Pham, and S. D. Nguyen, "Server: Multi-modal speech emotion recognition using transformer-based and vision-based embeddings," in *Proceedings of the 2023 8th International Conference on Intelligent Information Technology*, 2023, pp. 234–238.

[10] P.-N. Tran, T.-D. T. Vu, D. N. M. Dang, N. T. Pham, and A.-K. Tran, "Multi-modal speech emotion recognition: Improving accuracy through fusion of vggish and bert features with multi-head attention," in *Industrial Networks and Intelligent Systems*, N.-S. Vo and H.-A. Tran, Eds. Cham: Springer Nature Switzerland, 2023, pp. 148–158.

[11] P.-N. Tran, T.-D. T. Vu, N. Truong Pham, H. Dang-Ngoc, and D. N. M. Dang, "Comparative analysis of multi-loss functions for enhanced multi-modal speech emotion recognition," in *2023 14th International Conference on Information and Communication Technology Convergence (ICTC)*, 2023, pp. 425–429.

[12] Q. B. Le, K. Tuan Trinh, N. D. Hung Son, P.-N. Tran, C. T. Nguyen, and D. N. M. Dang, "Mersa: Multimodal emotion recognition with self-align embedding," in *2024 International Conference on Information Networking (ICOIN)*, 2024, pp. 500–505.

[13] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015.

[14] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. PMLR, 06–11 Aug 2017, pp. 1321–1330.

[15] 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, pp. 335–359, 2008.

[16] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *Proceedings of the 30th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, S. Dasgupta and D. McAllester, Eds., vol. 28, no. 3. Atlanta, Georgia, USA: PMLR, 17–19 Jun 2013, pp. 1139–1147.