Neurocomputing 506 (2022) 168–183

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Cross-modal distillation with audio–text fusion for fine-grained emotion classification using BERT and Wav2vec 2.0

Donghwa Kim [a,b], Pilsung Kang [a,*]

[a] School of Industrial & Management Engineering, Korea University, Republic of Korea
[b] Vision & NLP, Kakao Style, Republic of Korea

ARTICLE INFO

Article history:
Received 16 December 2021
Revised 14 June 2022
Accepted 12 July 2022
Available online 22 July 2022

Keywords:
Multi-class emotion classification
Knowledge distillation
Transformer
BERT
Wav2Vec 2.0
Contrastive learning

ABSTRACT

Fine-grained emotion classification for mood- and emotion-related physical-characteristics detection and its application to computer technology using biometric sensors has been extensively researched in the field of affective computing. Although text modality has achieved a considerably high performance from the perspective of sentiment analysis, which simply classifies a positive or negative label, fine-grained emotion classification requires additional information besides text. An audio feature can be adopted as the additional information as it is closely associated with text, and the characteristics of the changes in sound pulses can be employed in fine-grained emotion classification. However, the multimodal datasets related to fine-grained emotion are limited, and the scalability and efficiency are insufficient for multimodal training to be applied extensively via the self-supervised learning (Self-SL) approach, which can adequately represent modality. To address these limitations, we propose cross-modal distillation (CMD), which induces the feature spaces of student models with a few parameters while receiving those of the teacher models that can adequately express each modality based on Self-SL. The proposed CMD performs the mapping of a feature space between teacher-student models based on contrastive learning, while two attention mechanisms—cross-attention between audio and text features and self-attention for features in modality—are performed during knowledge distillation. Wav2vec 2.0 and BERT, which are already adequately trained for audio and text via Self-SL, were adopted as teacher models; audio–text transformer models were used as student models. Accordingly, the CMD-based representation learning applies a lightweight model for IEMOCAP, MELD, and CMU–MOSEI datasets with the task of multi-class emotion classification, while exhibiting better fine-grained emotion classification performance than benchmark models with a considerably low uncertainty for prediction.

1. Introduction

Unlike sentiment analysis, which classifies a given text into polarity labels (positive, neutral, and negative), fine-grained emotion classification aims at classifying the text into subdivided emotion labels (happy, anger, sad, and neutral) to better understand human emotions [1–3]. These emotion labels have characteristics related to a person's intention and mood; hence, studies are actively being conducted in the field of affective computing on using fine-grained emotion classification to detect physical characteristics and apply them to computer technology using biometric sensors [4–7]. In addition, beyond speech recognition, extensive research is being conducted on speech synthesis to represent detailed emotions of a person in the deep-learning-based emotional speech synthesis field [8–10]; these models can be applied to various application services, including artificial intelligence (AI) speakers or the Internet of Things (IoT), which further highlights the need for research on fine-grained emotion classification [11,12].

In sentiment analysis, which classifies polarities, previous studies have mostly focused on text modality, and natural language models with high-classification performance have been proposed [13–15]. However, there are cases in which additional information is required besides text to classify fine-grained emotions [16]. First, the intonation of a speech may vary depending on emotion expressions. For example, sarcastic sentences are used to emphasize the

Abbreviations: Semi-SL, Semi-Supervised Learning; Self-SL, Self-Supervised Learning; ST, Self-Training; CT, Co-Training; CMD, Cross-Modal Distillation.

* Corresponding author.
E-mail addresses: sol.ve@kakaostyle.com (D. Kim), pilsung_kang@korea.ac.kr (P. Kang).

https://doi.org/10.1016/j.neucom.2022.07.035
0925-2312/

contrary and induce a person to feel or appear foolish. Saying *"Very good; well done!"* when someone has clearly done wrong, or *"That's just what I needed today!"* when something bad has happened, has a negative connotation of "brooding," despite the positive context of the text modality. Second, when classifying emotions of relatively short sentences, ambiguity exists in emotion classification owing to the insufficient context information within the text. Therefore, it is difficult to accurately determine emotions solely based on text modality in short-answer questions. Accordingly, several studies have been conducted on audio–text fusion in which audio signals related to text are additionally taken into consideration [17–19]. Representation learning for audio–text features are essential in fine-grained emotion classification tasks, considering that audio features reflecting the changes in sound pulses can be additionally used with text features.

The collection of paired datasets comprising audio and text containing labels related to fine-grained emotions has been severely limited, which triggers difficulties in learning the characteristics and relationships of audio–text modality owing to a critically insufficient amount of data [20]. To date, the strategies that have been attempted in previous studies to address this limitation can be grouped into three categories. As the first strategy, the semi-supervised learning (Semi-SL) approach has been attempted to improve classification performance for cases with a large number of unlabeled data, in addition to a few labeled data [21]. Semi-SL approaches involve generating labels of unlabeled data by adopting the target distribution of a base model trained with a few labeled data. However, this training method becomes difficult owing to the poor performance of the initial base model and increased uncertainty of newly generated labels, as the entropy of the target distribution increases; time complexity is also large, as the model requires repeated training [22]. The second approach is the Self-SL approach for performing predictions by defining hidden labels in data without target labels. This approach includes two steps: (i) pretraining and (ii) fine-tuning [13,23]. In general, vectors are trained to increase the co-occurrence of the vectors (word or amplitude) sequentially appearing in text and audio modality in the pretraining step, while pretrained models are retrained in the fine-tuning step to ensure that the predictions of downstream tasks are performed adequately. Fine-tuning of the end-to-end learning is absolutely necessary for the models trained via Self-SL, if the difference between a hidden task of the pretraining step and a downstream task of the fine-tuning step is significant, or when the domain of training data changes [24]. Furthermore, scalability and memory efficiency are insufficient in fine-tuning multiple massive pretrained models simultaneously for multimodal datasets.

To resolve the above issues, we propose a cross-modal distillation (CMD) approach to learn the cross-modality interaction between acoustic and textual information. The proposed CMD method executes contrastive learning to ensure that the student model feature space that comprises a few parameters resembles the teacher model feature space that adequately represents the features of each modality. Moreover, two attention mechanisms (audio–text cross-attention and self-attention) were executed to enhance the representation for audio–text fusion. The proposed method consists of two steps. The first step is the *distillation step*, where the student models are updated to ensure that the feature vector of student models (audio–text transformers) with a few parameters becomes similar to the feature vectors of Wav2vec 2.0 and BERT (teacher models) that are adequately trained using Self-SL for audio and text, respectively. During the training process, audio–text transformers undergo cross-attention and self-attention sequentially to proceed with audio–text fusion. The cross-attention used in the distillation step pretrains the relationship and alignment between audio and text for multi-class emotion classification in the subsequent fine-tuning step. The second step, *fine-tuning step*, involves retraining using audio–text transformers (student models), in which the model parameters are updated to ensure that features can effectively classify emotion labels. Based on the results obtained from experiments conducted using IEMOCAP, MELD, and CMU–MOSEI datasets, the representation learning based on the proposed CMD not only used less model parameters but also yield higher fine-grained emotion classification performance than benchmark models, while exhibiting significantly low uncertainty for prediction results. Moreover, The effects of each modality on the proposed model were additionally investigated on using gradient weights of the modalities.

The main contributions of this paper can be summarized as follows:

- We propose knowledge distillation using a lightweight model to learn the cross-modality interaction between acoustic and textual information in the pretraining step.
- We inject two attention mechanisms (audio–text cross-attention and self-attention) that are executed in the distillation step to enhance the representation for audio–text fusion.
- Our proposed lightweight model outperforms benchmark models with a considerably low uncertainty for prediction in the task of multi-class emotion classification for EMOCAP, MELD, and CMU–MOSEI datasets.
- We provide the visual interpretation method of multimodal architecture by measuring the gradient weights of each modality's effect.

The remainder of this paper is organized as follows. Related studies that have performed fine-grained emotion classification using a few training labels are briefly introduced in Section 2. The proposed CMD is thoroughly described in Section 3. Data description, parameter setting, and performance measurements are presented in Section 4, while experimental results are analyzed in Section 5. Finally, in Section 6, a conclusion is provided and future research directions are proposed.

2. Literature review

The methods available for improving multi-class emotion classification performance using multimodal datasets with insufficient training labels can be grouped into three categories: Semi-SL, Self-SL, and knowledge distillation.

In previous studies on Semi-SL approaches, self-training (ST) and co-training (CT) were proposed for the emotion classification of multimodal datasets [25,26]. The ST approach estimates the target distribution of each unlabeled example using a classifier trained with a few labeled examples, and then generates the labels for the unlabeled examples. Then, considering the confidence score of the generated labels, the newly labeled examples are included in the set of labeled examples, and repeatedly used for updating the classifier. Similar to ST, the CT approach involves training models separately for two independent data, and then the labels generated in the target distribution of each model are cross-tossed, thereby receiving help from a different perspective. However, when the performance of initial base models is poor, uncertainty for label annotation increases with the entropy of the target distribution. Moreover, time complexity is large owing to newly repeated training for classifiers [22].

Previous studies on Self-SL approaches included two steps: pretext task for predicting hidden targets (pretraining step) and downstream task for emotion classification (fine-tuning step). Khare et al. [27] extracted features of multimodal datasets (vision and audio) for pretext tasks, using an attention-based transformer

encoder [28], and trained them to ensure that the feature vectors can be represented to better predict the randomly generated mask tokens in text data. Then, the feature vectors were fine-tuned in the downstream task step to enable trained models to adequately classify emotion datasets.

Siriwardhana et al. [29] adopted RoBERTa and speech BERT models for the feature representation of text and audio to perform the pretext task. Speech BERT involves performing vector quantization to change the audio signal of a continuous domain to a discrete domain using the pretrained VQ-wav2vec model [30], as well as training the discrete vectors based on masked language modeling of the BERT architecture [13]. The RoBERTa model [31] is a variant of BERT model in which the next sentence prediction (NSP) is omitted and a dynamic masking technique is added to generate the masked tokens per mini-batch. Similar to the BERT model, in the downstream task step, the parameters of the pretrained model are fine-tuned to classify emotions. However, these approaches require updating a large number of parameters in the fine-tuning step, and a domain shift for the source data can easily occur [24]. In addition, owing to the large pretrained model size of Self-SL models, they cannot be efficiently scaled when applied to multimodal datasets.

Knowledge distillation approaches are primarily researched in three aspects. First, response-based knowledge [32] delivers the output (softmax) layer of a teacher network that has already been well trained to the output layer of a student network being trained. The objective function of this approach is calculated based on the difference between two distributions of target classes computed from the student and teacher networks. Second, feature-based knowledge [33] is trained to minimize the difference between the two feature vectors computed from the student and teacher networks, where mean squared error (MSE) is commonly used as a loss function. Third, in addition to considering the similarity between the two feature vectors computed by the teacher-student network, relation-based knowledge [34] also preserves the similarity between observations used for the training data. For example, if a feature matrix is denoted as $\mathbf{h} \in \mathbb{R}^{n\times d}$, the similarity between observations is calculated as $\mathbf{h}^T \cdot \mathbf{h} = D$. Then, knowledge distillation is performed while minimizing the difference in the $D \in \mathbb{R}^{n\times n}$ matrix (MSE) calculated from student and teacher networks.

Several studies have adopted hierarchical feature extraction for emotions to learn the relevance of utterances. Nie et al. [35] proposed an incremental graph convolution network (I-GCN) for a combination of the temporal information of conversation and the semantic information of utterances. First, an utterance-level GCN (U-GCN) applies the transformer model to represent the correlations between the utterances, and then a speaker-level GCN (S-GCN) is used to represent the correlations between speakers and the utterances. This approach enables the feature representation of emotion to enhance the temporal and semantic relevance of utterances. In another study, Nie et al. [36] utilized traditional multi-LSTM encoders by combining text, audio, and video data to overcome the sparsity of emotion representing video frames. This framework is comprised of multiple hierarchical LSTM layers to fuse multimodal features in the order of text, audio, and video data using pretrained vectors.

3. Method: cross-modal distillation

In this paper, the proposed CMD attempts to provide the context information of Wav2vec 2.0 and BERT (teacher models) that are well pretrained for large audio and text data to audio–text transformers (student models) with a few parameters. The architecture of the proposed CMD is presented in Fig. 1. The distillation method of contextual embeddings involves adopting contrastive learning to maximize the similarity between the two teacher–student vectors, and the model was designed to ensure that attention-based audio–text fusion occurs between the audio–text transformers. The proposed CMD can contribute to the improvement of contextual representation between audio and text, based on the model with a few parameters, and the performance improvement of emotion recognition can be achieved, provided the student models are fine-tuned on the downstream task.

3.1. Student models

Algorithm 1: Transformer block

1 n: number of samples, s: sequence length,
2 d: feature dimensions for input, d_h: feature dimensions for a head,
3 $\mathbf{x}^q \in \mathbb{R}^{n\times d\times s}$: collection of input embeddings for query,
4 $\mathbf{x}^k \in \mathbb{R}^{n\times d\times s}$: collection of input embeddings for key,
5 $\mathbf{x}^v \in \mathbb{R}^{n\times d\times s}$: collection of input embeddings for value,
6 $f_\theta : \mathbb{R}^{n\times d\times s} \rightarrow \mathbb{R}^{n\times d\times s}$, a parameterized function for a transformer block.
7 $\underline{f_\theta(\mathbf{x}^q, \mathbf{x}^k, \mathbf{x}^v)}$
input embeddings for query, key and value.
contextual embeddings.
8 $Q(\mathbf{x}_i^q) = W_q^T\mathbf{x}_i^q, \quad K(\mathbf{x}_i^k) = W_k^T\mathbf{x}_i^k, \quad V(\mathbf{x}_i^v) = W_v^T\mathbf{x}_i^v \qquad //W_q, W_k, W_v \in \mathbb{R}^{d\times d_h}$
9 $\alpha_{i,j} = \mathrm{softmax}_j\left(\frac{(Q(\mathbf{x}_i^q), K(\mathbf{x}_j^k))}{\sqrt{d_h}}\right) \qquad //\mathrm{softmax}_j(\mathbf{z}) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$
10 $\mathbf{h}_i = \sum_{j=1}^{n} \alpha_{i,j} V(\mathbf{x}_j^v) \qquad //\mathbf{h} \in \mathbb{R}^{n\times d_h\times s}$
11 $\mathbf{u}_i' = W_h^T \mathrm{Concat}(\mathbf{h}_i^1, \mathbf{h}_i^2, \ldots, \mathbf{h}_i^h) \qquad //W_h \in \mathbb{R}^{d\times d},\ \sum d_h = d$
12 $\mathbf{u}_i = \mathrm{LayerNorm}(\mathbf{x}_i^q + \mathbf{u}_i'; \gamma_1, \beta_1) \qquad //\gamma_1, \beta_1 \in \mathbb{R}^d$
13 $\mathbf{c}_i' = W_2^T \mathrm{ReLU}(W_1^T\mathbf{u}_i) \qquad //W_1 \in \mathbb{R}^{d\times m}, W_2 \in \mathbb{R}^{m\times d}$
14 $\mathbf{c}_i = \mathrm{LayerNorm}(\mathbf{u}_i + \mathbf{c}_i'; \gamma_2, \beta_2) \qquad //\gamma_2, \beta_2 \in \mathbb{R}^d$
15 **return c**

The student models of the proposed method employ audio–text transformers comprising transformer encoders. A transformer [28] is a methodology where sequential information is adequately represented using multi-head attention, as presented in Algorithm 1, which shows its computation process. The transformer consists of multiple blocks, where one transformer block parameterizes input embeddings into contextual embeddings (line 1–6 in Algorithm 1). The computation process of the transformer block comprises two steps. In the first step, query, key, and value vectors are converted for each head for $\mathbf{x}_i$ (line 7 in Algorithm 1). Subsequently, the attention score proposed in the transformer [28] is used as presented in Fig. 2. This score is calculated using the scaled dot product of Q and K and the softmax function, and it is applied to the value vector (line 8–10 in Algorithm 1). The attention vectors computed for each head are merged, and a linear transformation is applied to the merged vectors (line 11 in Algorithm 1). In the second step, the results obtained from the first step are sent to the feed-forward neural network to perform non-linear transformation of the feature representation (line 13 in Algorithm 1). Furthermore, to mitigate gradient vanishing and exploding, the output of each step is added to the input and normalized by the LayerNorm function [37] (line 12 and 14 in Algorithm 1). In this manner, when L number of transformer blocks is used, $f_{\theta_L} \circ \cdots \circ f_{\theta_1} \in \mathbb{R}^{n\times d\times s}$, is recursively denoted.

The audio–text transformers of the proposed CMD illustrated in Fig. 1 consist of two-step attention modules. In the first module, the cross-attention method between the audio and text generates a query vector in its own modality, and then performs multi-head attention by receiving key and value vectors from the other reference modality. The cross-attention method for audio transformers can be expressed as follows:

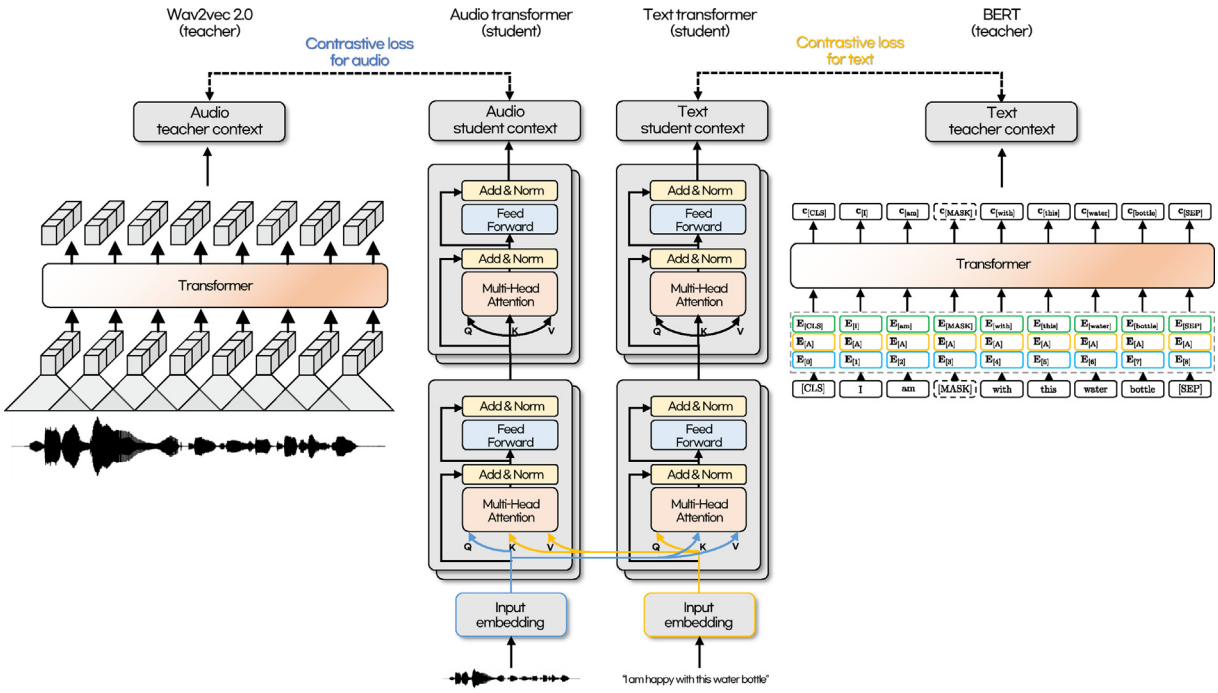

Fig. 1. Framework of CMD. During cross-attention module, the audio transformer receives key and value vectors from the text modality (yellow directions), while the text transformer receives key and value vectors from the audio modality (blue directions). In the CMD, self-attention is performed for each modality, and the similarity between the teacher and student context vectors is maximized in the output layer (dashed lines).

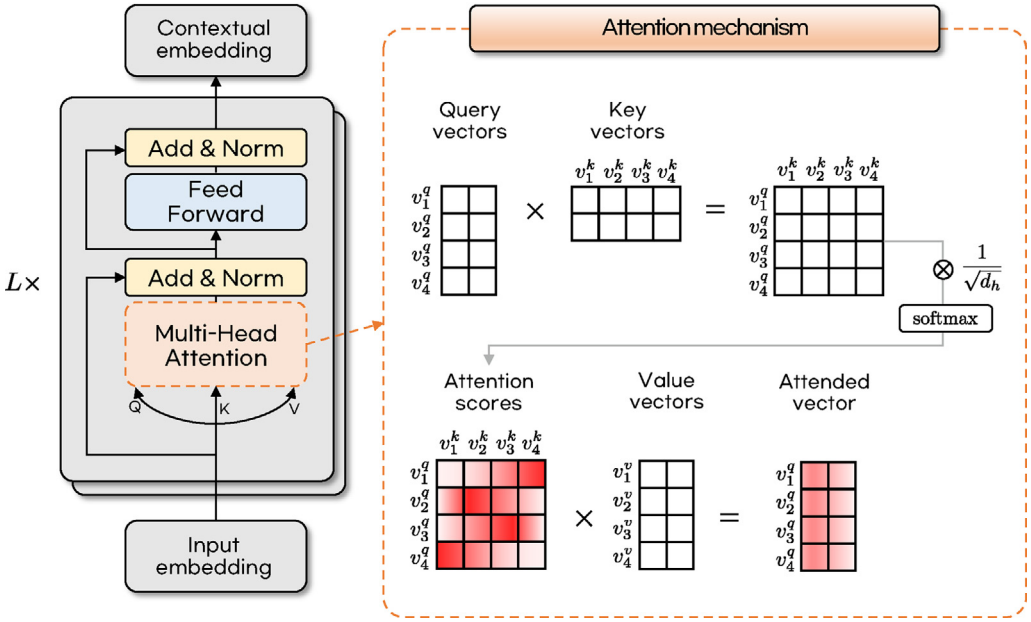

Fig. 2. Attention mechanism of the transformer encoder. v_i^q, v_i^k, and v_i^v represent the query, key, and value vectors in the i-th sequence, respectively; L denotes the number of encoder blocks.

$$f_\theta(\mathbf{x}^q_{\text{audio}}, \mathbf{x}^k_{\text{text}}, \mathbf{x}^v_{\text{text}}) = \mathbf{c}_{\text{audio}}. \tag{1}$$

Similarly, the cross-attention method for text transformers can be expressed as follows:

$$f_\theta(\mathbf{x}^q_{\text{text}}, \mathbf{x}^k_{\text{audio}}, \mathbf{x}^v_{\text{audio}}) = \mathbf{c}_{\text{text}}. \tag{2}$$

In the second module, the self-attention method involves generating the query, key, and value vectors in its own modality to proceed with multi-head attention. The proposed CMD applied the context vectors, $\mathbf{c}_{\text{audio}}$ and $\mathbf{c}_{\text{text}}$, that have been applied with cross-attention to the self-attention based transformer. Accordingly, the self-attention method for audio transformers can be expressed as follows:

$$f_\theta(\mathbf{c}^q_{\text{audio}}, \mathbf{c}^k_{\text{audio}}, \mathbf{c}^v_{\text{audio}}) = \mathbf{c}^s_{\text{audio}}. \tag{3}$$

Similarly, the self-attention method for text transformers can be expressed as follows:

$$f_\theta(\mathbf{c}^q_{\text{text}}, \mathbf{c}^k_{\text{text}}, \mathbf{c}^v_{\text{text}}) = \mathbf{c}^s_{\text{text}}. \tag{4}$$

Finally, the audio–text transformers applied with the two-step attention modules compute the contextual embeddings for audio and text, which are denoted as $\mathbf{c}^s_{\text{audio}}$ and $\mathbf{c}^s_{\text{text}}$, respectively.

3.2. Teacher models

For the teacher models of the CMD model proposed in this paper, pretrained Wav2vec 2.0 [23] and BERT [13] that can adequately represent the contextual information of audio and text modalities were adopted accordingly. As aforementioned, both models are transformer encoders that predict masked targets using contextual embeddings.

3.2.1. BERT

BERT [13] trains the mask language modeling (MLM) task to predict masked words (tokens) in a sentence, including the next sentence prediction (NSP) task for predicting the next sentence using the transformer encoders. Fig. 3 presents the architecture of the BERT model.

3.2.2. Wav2vec 2.0

Wav2vec 2.0 [23] employs the MLM technique of BERT for raw speech. As illustrated in Fig. 4, this method consists of two tasks. (1) Contrastive task: contrastive learning is executed, such that the masked input embedding in a certain position is trained to attract the true unmasked feature (positive samples) of the same position, while repelling the unmasked features (negative samples) positioned in different masked time steps. (2) Quantization task: true unmasked features in the masked time steps are applied with the argmax function of the quantization module and represented with a discrete space. To secure the diversity of quantization during this process, the entropy of the averaged softmax distribution is maximized, such that the given unmasked features are evenly distributed in the discrete space. The input embedding of Wav2vec 2.0 comprises multiple blocks where the input data of each block are sequentially applied with 1D convolution [38], LayerNorm [37], and GELU activation function [39]. Then, the input embedding is encoded by the transformer model to be trained as contextual embeddings for predicting contrastive and quantization tasks. Total loss $\mathscr{L}_{\text{wav2}}$ comprises the sum of the individual loss of the minimum contrastive loss ($\mathscr{L}_c$) and the maximum perplexity of quantization ($\mathscr{L}_q$); α_q denotes the hyperparameter adjusting the trade-off between the two losses:

$$\mathscr{L}_{\text{wav2}} = \mathscr{L}_c + \alpha_q \mathscr{L}_q. \tag{5}$$

Specifically, if the masked context feature at a specific step $\mathbf{c}_t$ and the true unmasked and quantized feature $\tilde{\mathbf{q}}_t$ at the same step are positive samples, and $K+1$ number of quantized candidate features consisting of the positive sample and K negative samples [40] are denoted as $\mathbf{Q}_t$, then $\mathscr{L}_c$ can be defined as follows:

$$\mathscr{L}_c = -\log \frac{\exp(sim(\mathbf{c}_t, \tilde{\mathbf{q}}_t)/\kappa)}{\sum_{\tilde{\mathbf{q}} \sim \mathbf{Q}_t} \exp(sim(\mathbf{c}_t, \tilde{\mathbf{q}})/\kappa)}, \quad sim(\mathbf{a}, \mathbf{b}) = \mathbf{a}^T\mathbf{b}/||\mathbf{a}||||\mathbf{b}||. \tag{6}$$

In this equation, the cosine similarity $sim(\mathbf{a}, \mathbf{b})$ indicates the similarity between positive and negative samples calculated with unmasked and quantized features, and the similarity distribution is adjusted by the temperature scale κ [41] (conventionally adjusted, such that distribution becomes more pointed [23]). To denote the perplexity of the quantization module, if the target size of the quantized distribution is V_e entries, and the number of quantized distributions is G codebooks, $\mathscr{L}_q$ can be defined as follows [23,42].

$$\mathscr{L}_q = \frac{1}{GV_e} \sum_{g=1}^{G} \sum_{v_e=1}^{V_e} \bar{p}_{p,v_e} \log \bar{p}_{g,v_e}. \tag{7}$$

In this equation, $\bar{p}_{g,v_e}$ denotes the entropy of the averaged softmax distribution over entries in each codebook across a batch.

3.3. Cross-modal distillation

3.3.1. Feature-based distillation with audio–text fusion

Inspired by the contrastive learnings [23,43], the proposed CMD extracts context embeddings from Wav2vec 2.0 and BERT (teacher

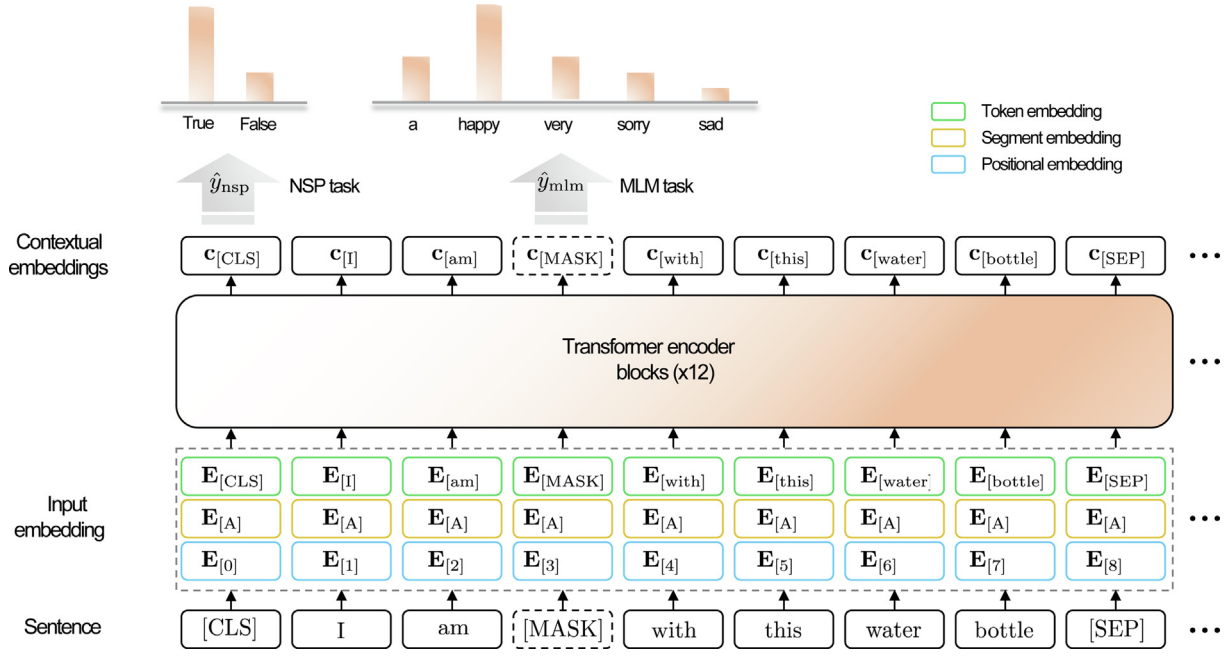

Fig. 3. BERT architecture. The input embedding which consists of token, segment, and positional embeddings is encoded into contextual embedding by transformer blocks. The contextual embedding is trained to accurately predict the true tokens of the masked tokens and whether the next sentence is sequentially correct or not.

Algorithm 2: CMD algorithm

// Definition of the functions
1 Wav2vec 2.0: pretrained teacher model for audio,
2 BERT: pretrained teacher model for text,
3 audio–text transformers: student models for audio and text,
4 g: an encoder for linear embedding.
// Definition of the dimensions
5 n: batch size, M: number of model updates,
6 $\mathbf{x}_{\text{audio}} \in \mathbb{R}^{n \times s_{\text{a}}}$: raw data for audio, $\mathbf{x}_{\text{text}} \in \mathbb{R}^{n \times s_{\text{t}}}$: raw data for text,
7 s_{a}: sequence length for audio signal, s_{t}: sequence length for tokens,
8 d_{a}: output dimensions for Wav2vec2, d_{t}: output dimensions for BERT,
9 d_{k}: output dimensions for audio–text transformers, d_{z}: latent dimensions for $g(\cdot)$,
10 Θ: updated parameters of audio–text transformers and $g(\cdot)$.
11 **CMD**$(\mathbf{x}_{\text{audio}}, \mathbf{x}_{\text{text}})$
Input : raw data for audio and text
Output: distilled audio–text transformers
12 **repeat**
13 $\mathbf{c}^t_{\text{audio}} \leftarrow$ Wav2vec 2.0$(\mathbf{x}_{\text{audio}})$ // $\mathbf{c}^t_{\texttt{audio}} \in \mathbb{R}^{n \times s_a \times d_a}$
14 $\mathbf{c}^t_{\text{text}} \leftarrow$ BERT$(\mathbf{x}_{\text{text}})$ // $\mathbf{c}^t_{\texttt{text}} \in \mathbb{R}^{n \times s_t \times d_t}$
15 $\mathbf{c}^s_{\text{audio}}, \mathbf{c}^s_{\text{text}} \leftarrow$ audio–text transformers$(\mathbf{x}_{\text{audio}}, \mathbf{x}_{\text{text}})$ // $\mathbf{c}^s_{\texttt{audio}} \in \mathbb{R}^{n \times s_a \times d_k}$, // $\mathbf{c}^s_{\texttt{text}} \in \mathbb{R}^{n \times s_t \times d_k}$
16 $\mathbf{z}^t_{\text{audio}}, \mathbf{z}^s_{\text{audio}} \leftarrow g(\mathbf{c}^t_{\text{audio}}), g(\mathbf{c}^s_{\text{audio}})$ // $\mathbf{z}^t_{\texttt{audio}}, \mathbf{z}^s_{\texttt{audio}} \in \mathbb{R}^{n \times s_a \times d_z}$
17 $\mathbf{z}^t_{\text{text}}, \mathbf{z}^s_{\text{text}} \leftarrow g(\mathbf{c}^t_{\text{text}}), g(\mathbf{c}^s_{\text{text}})$ // $\mathbf{z}^t_{\texttt{text}}, \mathbf{z}^s_{\texttt{text}} \in \mathbb{R}^{n \times s_t \times d_z}$
18 $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{clr}}(\mathbf{z}^t_{\text{audio}}, \mathbf{z}^s_{\text{audio}}) + \mathcal{L}_{\text{clr}}(\mathbf{z}^t_{\text{text}}, \mathbf{z}^s_{\text{text}})$ // $\mathcal{L}_{\texttt{clr}}$ in Eq.(9)
19 $\Theta \leftarrow \Theta - \zeta \frac{\partial \mathcal{L}_{\text{total}}}{\partial \Theta_1}$ // ζ: a learning rate.
20 **until** $m \rightarrow M$
21 **return** audio–text transformers

models) in which the weight is fixed, and then performs offline distillation to deliver context information to audio–text transformers (student models). Fig. 1 illustrates the training process of the CMD, which is comprehensively suggested in Algorithm 2. First, audio and text are input as raw data to generate contextual embeddings of Wav2vec 2.0 and BERT (teacher models), respectively (line 13–14 in Algorithm 2). Similarly, contextual embeddings are generated in audio–text transformers (student models) (line 15 in Algorithm 2) where the input embeddings of the student models are used with the same input encoder structure of the teacher models. After applying a linear transformation to the generated context vectors, contrastive learning is performed to ensure teacher and student contexts become similar for each modality within a batch (line 16–18 in Algorithm 2). Fig. 5 presents the process of computing similarity between $\mathbf{z}$ vectors per time step in which the (i,j)-th entry of the matrix for the contrastive loss can be expressed as follows:

$$\ell_{ij} = -\log \frac{\exp(s_{ij}/\kappa)}{\sum_{k=1}^{2n} \mathbb{1}_{[\mathbb{k} \neq \mathbb{i}]} \exp(s_{\mathbb{i}\mathbb{k}}/\kappa)}, \qquad \ell \in \mathbb{R}^{2n \times 2n}. \tag{8}$$

In the equation, s_{ij} denotes the (i,j)-th similarity. Therefore, the contrastive loss for $\mathbf{z}^t$ and $\mathbf{z}^s$ can be defined as follows:

$$\mathcal{L}_{\text{clr}}(\mathbf{z}^t, \mathbf{z}^s) = \frac{1}{2n} \sum_{k=1}^{n} [\ell_{k,n+k} + \ell_{n+k,n}]. \tag{9}$$

The total contrastive loss of each modality is defined as $\mathcal{L}_{\text{total}}$ and the gradient-descent algorithm [44] is used to train the model (line 19 in Algorithm 2).

3.3.2. Fine-tuning on the downstream task

The emotion labels are fine-tuned to verify whether the distilled audio–text transformers contributes to the improvement in the multiclass emotion classification performance. Algorithm 3 summarizes the fine-tuning process. First, the raw data of audio and text are encoded to $\mathbf{c}^s_{\text{audio}}$ and $\mathbf{c}^s_{\text{text}}$ (context embeddings) using the distilled audio–text transformers generated in Algorithm 2. Subsequently, average pooling over the time steps is applied to ensure that the sequential vectors of audio and text with different lengths are represented as vectors with the same lengths for emotion classification (line 8 in Algorithm 3).

The pooled vectors go through fully-connected layers comprising linear embeddings and an ReLU activation function, and the output logits are normalized by the temperature scaling κ to adjust a target distribution with the softmax function (line 9–10 in Algorithm 3). The final predictive probability is defined as the average of the output probability formed for each modality, and the cross-entropy $\mathcal{L}_{ce}$ indicating the difference from the target label is calculated (line 11–12 in Algorithm 3). Moreover, the gradients of $\mathcal{L}_{ce}$ with respect to the parameters for the audio–text transformers and

Algorithm 3: Fine-tuning for multi-class emotion classification

// Definitions
n: batch size, c: target size, M: number of model updates,
$\mathbf{x}_{\text{audio}} \in \mathbb{R}^{n \times s_a}$: raw data for audio, $\mathbf{x}_{\text{text}} \in \mathbb{R}^{n \times s_t}$: raw data for text,
$\mathbf{y}$: emotion labels corresponding to $(\mathbf{x}_{\text{audio}}, \mathbf{x}_{\text{text}})$,
Θ : updated parameter for audio–text transformers and fully-connected layers $\mathcal{F}$.

Fine-tune$(\mathbf{x}_{\text{audio}}, \mathbf{x}_{\text{text}})$
Input : raw data for audio and text with the emotion labels
Output: audio–text transformers for emotion classification

repeat
  $\mathbf{c}^s_{\text{audio}}, \mathbf{c}^s_{\text{text}} \leftarrow$ audio–text transformers$(\mathbf{x}_{\text{audio}}, \mathbf{x}_{\text{text}})$
  $\mathbf{c}^\mu_{\text{audio}}, \mathbf{c}^\mu_{\text{text}} \leftarrow \text{AvgPool}(\mathbf{c}^s_{\text{audio}}),\ \text{AvgPool}(\mathbf{c}^s_{\text{text}})$ // $\mathbf{c}^\mu_{\text{audio}}, \mathbf{c}^\mu_{\text{text}} \in \mathbb{R}^{n \times d_k}$
  $\hat{\mathbf{y}}_{\text{audio}} = \text{softmax}(\mathcal{F}(\mathbf{c}^\mu_{\text{audio}})/\kappa)$ // $\hat{\mathbf{y}}_{\text{audio}} \in \mathbb{R}^{n \times c}, \kappa \in \mathbb{R}$
  $\hat{\mathbf{y}}_{\text{text}} = \text{softmax}(\mathcal{F}(\mathbf{c}^\mu_{\text{text}})/\kappa)$ // $\hat{\mathbf{y}}_{\text{text}} \in \mathbb{R}^{n \times c}, \kappa \in \mathbb{R}$
  $\hat{\mathbf{y}} = (\hat{\mathbf{y}}_{\text{audio}} + \hat{\mathbf{y}}_{\text{text}})/2$
  $\mathcal{L}_{ce} = -\frac{1}{nc}\sum_{i=1}^{n}\sum_{j=1}^{c}\mathbf{y}_{ij}\log\hat{\mathbf{y}}_{ij}$
  $\Theta \leftarrow \Theta - \zeta\frac{\partial\mathcal{L}_{ce}}{\partial\Theta}$ // ζ_2: a learning rate.
until $m \rightarrow M$
return audio–text transformers

fully-connected layers $\mathcal{F}$ are applied to the gradient-decent algorithm to train the model (line 13 in Algorithm 3).

4. Experiments

4.1. Data description

The IEMOCAP,[1] MELD,[2] and CMU–MOSEI[3] datasets, which contain the paired audio and text data for multi-class emotion classification, are commonly used to verify the model performance in different data domains. The number of data samples and their brief descriptions are provided in Table 1. Similar to the experimental setting of a previous study [45], the emotion labels of each data comprised four classes, including "happiness," "anger," "sadness," and "neutral." In the IEMOCAP dataset, the emotion label of each data was voted multiple times by three experimenters, and the emotion label that received two or more votes was adopted as the ground truth of that specific data. The train, validation, and test rates of the IEMOCAP dataset were 60%, 20% and 20%, respectively, and a predetermined partitioned index was used for the MELD and CMU–MOSEI datasets. In the CMU–MOSEI dataset, the emotion label voted multiple times with the maximum annotation rate of 1 or greater among "happiness," "anger," and "sadness" was adopted as the ground truth, and it was labeled as "neutral" if the annotation rates of the emotion labels were all zero. Data collected in different languages other than English, and audio files without text were excluded in the CMU–MOSEI dataset.

4.2. Experimental settings

Batch training. In the distillation and fine-tuning steps, the batch size was set dynamically based on the accumulated total of the frames generated in each audio file. For example, audio data can be formed into batches for up to 20 s if the accumulated sum of audio frames is allowed up to 320,000 batch frames for the audio data with a 16,000 sampling rate. Using the same method, the batches comprising text data, including audio, were generated. In our experiment, the accumulated batch frames were set to 320,000, 640,000 and 1,280,000, and were explored using the grid search method.

Settings in the distillation step. In the audio–text transformers employed in the proposed CMD, the number of transformer blocks (L) for cross- and self-attention was 3, the output dimension (d_k) of each block was 32, and the number of heads (h) was set to 2 as default. The input embedding of the student models used the same weights and structure as the input embedding of the teacher models. In addition, 1D-convolution with channel 32 was performed once for the input embedding. The initial value of the learning rate ζ was set to 10^{-4}, and different learning rates were adopted per epoch using a warmup scheduler [46]. Among a total of 50 epochs, the learning rate was gradually increased until 10 epochs (warmup), and then decreased by 0.1 per epoch for the remaining 40 epochs (decay). For the teacher models of the proposed CMD, a bert-base-uncased[4] architecture consisting of 12 transformer blocks with an output dimension (d_t) of 768 was adopted as the pretrained model of BERT, and a Wav2vec 2.0 Base[5] architecture consisting of 12 transformer blocks with an output dimension (d_a) of 768 was used as the pretrained model of Wav2vec 2.0. In contrastive learning, the dimension (d_z) of linear embedding for matching between teacher and student models for each modality was set to 32, and the temperature scaling κ was explored using the grid search method between 0.01, 0.05, and 0.1. The final model was applied with the model weights of a checkpoint with the smallest validation loss.

Settings in the fine-tuning step. To perform multi-class emotion classification, the distilled audio–text transformers were adopted as the initial model, and the audio–text transformers were fine-tuned by connecting two fully-connected layers comprising 32 dimensions for target prediction. The accumulated batch frames and tem-

[1] https://sail.usc.edu/iemocap/.
[2] https://github.com/declare-lab/MELD.
[3] https://github.com/A2Zadeh/CMU-MultimodalSDK.
[4] https://github.com/google-research/bert.
[5] https://github.com/pytorch/fairseq/tree/master/examples/wav2vec.

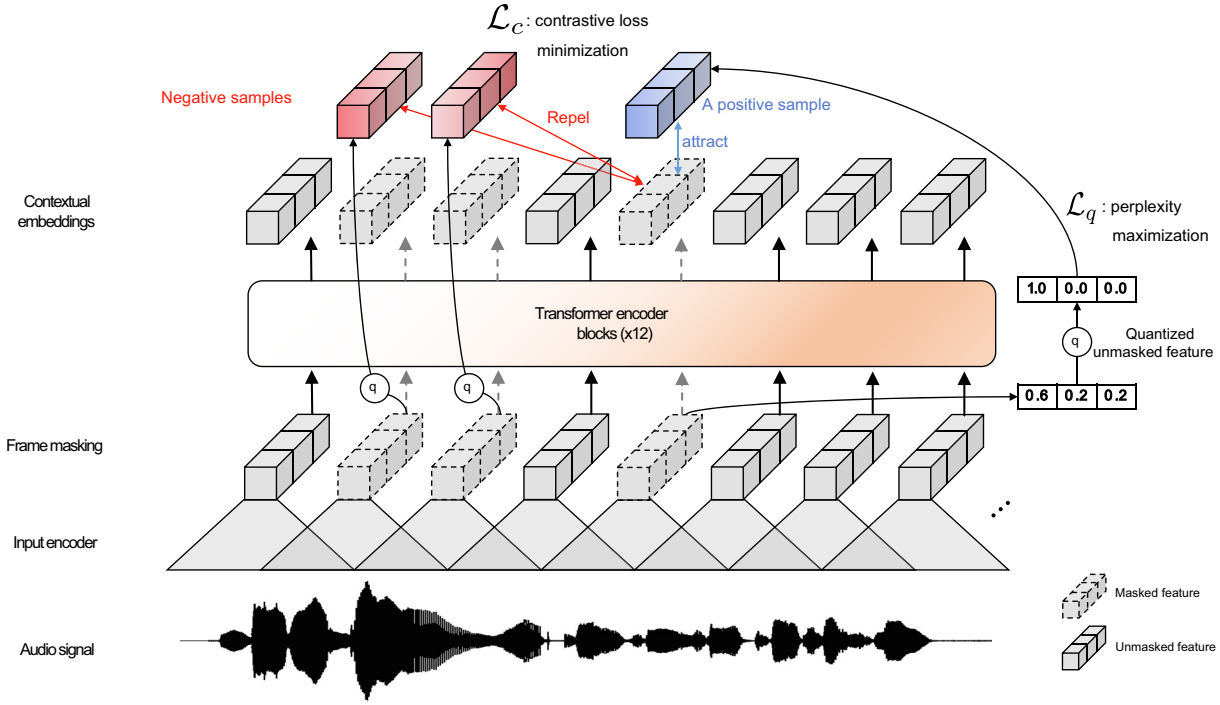

Fig. 4. Wav2vec 2.0 architecture. The input encoder consists of 1D convolution, LayerNorm, and GELU activation functions. The input embedding is encoded into a contextual embedding by transformer blocks, and contextual embedding is trained to attract the quantized features of the positive sample while repelling the quantized features of negative samples. Furthermore, the entropy of the softmax distribution is maximized to secure the diversity of the quantized features.

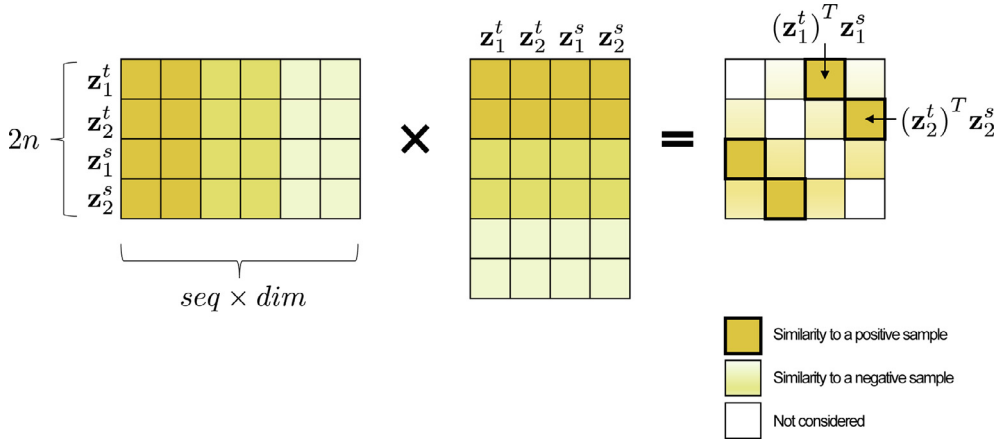

Fig. 5. Process of computing the similarity between $\mathbf{z}$ vectors generated in teacher and student models, as an example of Eq. (8). The example illustrates the case where $n = 2$ (batch size), $seq = 3$ (sequence length), and $dim = 2$ (dimensions). The diagonal of the similarity matrix is not considered for training; the similarities $(\mathbf{z}_1^{t\,T}\mathbf{z}_1^s)$ and $(\mathbf{z}_2^{t\,T}\mathbf{z}_2^s)$ derived from the same raw data are maximized, while the similarities $(\mathbf{z}_1^{t\,T}\mathbf{z}_2^t)$, $(\mathbf{z}_1^{t\,T}\mathbf{z}_2^s)$, $(\mathbf{z}_2^{t\,T}\mathbf{z}_1^s)$, and $(\mathbf{z}_1^{s\,T}\mathbf{z}_2^s)$ derived from different raw dat.a are minimized.

perature scaling κ used for fine-tuning were set identical to the distillation step where the best hyperparameters were explored. Based on the evaluation metrics of previous studies [45,47–49], the binary F1-score of a positive class was used to compare the experiment models. The F1-score can be defined as follows:

$$\begin{aligned} \text{F1} - \text{score} &= 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}, \\ \text{precision} &= \frac{tp}{tp+fp}, \ \text{recall} = \frac{tp}{tp+fn}, \\ tp &: \text{true positive}, \ fp : \text{false positive}, \ fn : \text{false negative}. \end{aligned} \tag{10}$$

Learning Strategies. A comparative experiment was conducted to verify the performance of the proposed CMD, as presented in Table 2. First, compared to mel-frequency cepstral coefficient (MFCC[6]) and GLOVE[7] used in a previous study [45], the encoding performance of Wav2Vec 2.0 and BERT and the effects of CMD were verified (multimodal case). Considering actual problems regarding data collection, a comparative experiment was conducted for the case where only one modality is used, and text data generated from automatic speech recognition (ASR) model consisting of Wav2Vec 2.0 architecture was applied to the text modality of the proposed CMD (unimodal case). The experiment was repeated 10 times to confirm

[6] In this paper, the MFCC was created by torchaudio library with n_fft_size = 400 and n_mfcc = 80.

[7] https://nlp.stanford.edu/data/glove.840B.300d.zip.

Table 1
Data description

Name	Description	No. of data
IEMOCAP	Audio and text data collected by having actors read and act scripted scenarios with emotions.	7,487
MELD	1400 dialogues extracted from the TV series, Friends. The dialogues include audio and text of various speakers.	11,350
CMU–MOSEI	Audio and text data collected from videos of random topics and including more than 1,000 YouTube speakers.	6,228

Table 2
Learning Strategies.

Case	Strategy	Description
multimodal	mfcc + glove [45]	Features of MFCC and GLOVE are used as input embedding of multimodal transformer [45]
	wav2 + bert	Contextual embeddings of Wav2vec 2.0 and BERT are used as input embedding of multimodal transformer [45]
	wav2 + bert + CMD	Our proposed CMD comprising early fusion between audio and text modality is used with Wav2vec 2.0 and BERT
unimodal	mfcc [45]	MFCC features are used as input embedding of a transformer
	glove [45]	GLOVE features are used as input embedding of a transformer
	wav2	Contextual embeddings of Wav2vec 2.0 are used as input embedding of a transformer
	bert	Contextual embeddings of BERT are used as input embedding of a transformer
	wav2 + distill	Distillation is applied to only audio using Wav2vec 2.0
	bert + distill	Distillation is applied to only text using BERT
	bert(asr)+distill	Audio data is converted to text data using ASR model, and distillation is applied to only text using BERT
	wav2 + bert(asr) +CMD	After converting audio to text using ASR model, our proposed CMD comprising early fusion between audio and text modality is used with Wav2vec 2.0 and BERT

the variability of the model.

5. Results

Model Selection For the batch frames used in the experiment, 1,280,000 was selected for all datasets, while the temperature scaling was set to 0.05 for IEMOCAP and MELD, and 0.01 for MOSEI. The batch frames are related to negative samples of contrastive learning in which the number of negative samples increases as the batch size increases because the number of positive samples is fixed to 1, as batch frames increase. Therefore, using more negative samples for training contributed to the improvement in the distillation performance. In general, the temperature scale improved performance as the value was smaller; it became regularized by increasing the sparsity of probability distribution for feature matching. Early fusion, which is illustrated in Fig. 1, was selected as the fusion method for the proposed CMD because when the variance of emotion classification performance was compared based on repeated experiments, as shown in Fig. A.8, the variance of the F1 scores of early fusion (cross-attention → self-attention) was significantly lower than that of late fusion (self-attention → cross-attention). the variance of F1-scores of early fusion (cross-attention → self-attention) was significantly lower than that of late fusion (self-attention → cross-attention). The most significant difference between the early and late fusion is the emergence of fusion in low- or high-level representation. We found that the fusion in the low-level representation of the early fusion method was more stable than the fusion in the high-level representation of the late fusion method when the relationship between the audio and text was learned.

Multimodal Performance. Fig. 6 presents a boxplot from the repeated experiments on emotion classification performance when data with both audio and text modality are assumed to be given. It is drawn as a subplot of each type of data; the y-axis of each subplot represents the macro-average F1-scores from the repeated experiments, while the x-axis denotes the experimental levels of two benchmark models and the model that adopted the proposed

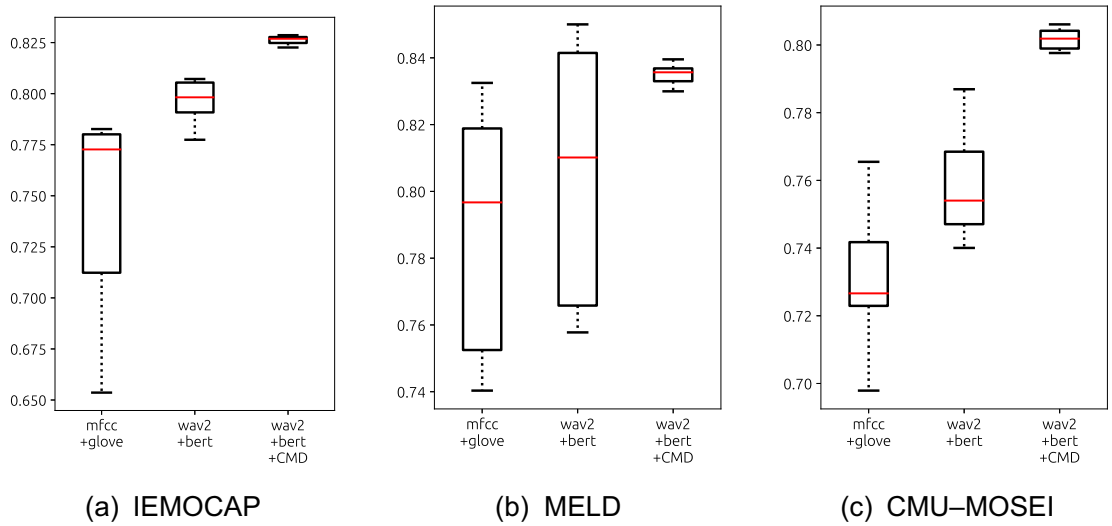

Fig. 6. Experimental results for multimodal cases in Table 2; boxplot of F1-scores calculated from data from repea.ted experiments.

Table 3
Average F1-scores with standard deviations for emotion labels based on the *multimodal* models. The first column in the table presents the data used in the experiments, and the second column shows the models used for learning. From the third to the sixth column, F1-scores of emotion labels are shown. Macro-average F1–score (Macro–F1), which is used as an integrated metric is presented in the seventh column. The experimental results present the mean (standard deviation) of 10 values deduced from the repeated experiments.

Data	Learning Strategy	Neutral	Happy	Sad	Anger	Macro–F1
IEMOCAP	mfcc + glove	0.7346 (0.046)	0.7614 (0.0576)	0.663 (0.0786)	0.8161 (0.0354)	0.7438 (0.0521)
	wav2 + bert	0.7956 (0.0084)	0.8162 (0.0136)	0.7107 (0.022)	0.861 (0.0144)	0.7959 (0.0111)
	wav2 + bert(CMD)	**0.8108 (0.0057)**	**0.8662 (0.0045)**	**0.7481 (0.0043)**	**0.8818 (0.0062)**	**0.8267 (0.0027)**
MELD	mfcc + glove	0.6909 (0.0824)	0.8042 (0.0417)	0.8606 (0.0091)	0.7991 (0.0215)	0.7887 (0.0361)
	wav2 + bert	0.7159 (0.0598)	0.8089 (0.0481)	0.8714 (0.017)	0.8221 (0.034)	0.8046 (0.0384)
	wav2 + bert(CMD)	**0.7714 (0.0075)**	**0.8402 (0.0081)**	**0.8788 (0.0029)**	**0.846 (0.0071)**	**0.8341 (0.0047)**
CMU–MOSEI	mfcc + glove	0.5881 (0.0448)	0.5684 (0.0354)	0.8852 (0.0113)	0.8844 (0.0192)	0.7315 (0.019)
	wav2 + bert	0.6354 (0.0255)	0.6076 (0.0335)	0.8891 (0.0097)	0.9008 (0.0077)	0.7582 (0.0158)
	wav2 + bert(CMD)	**0.6809 (0.0076)**	**0.6826 (0.0089)**	**0.9194 (0.0039)**	**0.9242 (0.0045)**	**0.8018 (0.0032)**

CMD. Furthermore, The results are presented in a boxplot to examine the prediction performance and variance. The calculated macro-average F1-scores, as well as the mean and standard deviation of F1-scores for the emotion labels, are presented in Table 3. Consequently, in addition to reducing variance in prediction performance, the proposed wav2 + bert(CMD) also exhibited a better emotion classification performance than mfcc + glove and wav2 + bert.

The mfcc + glove and wav2 + bert approaches both reused the *fixed vectors* calculated by the pretrained models as the input of transformers for emotion classification prediction. The wav2 + bert exhibited a more outstanding performance than mfcc + glove in the representation of a feature encoder. mfcc + glove and wav2 + bert both had high variability in prediction performance when fixed vectors were used, while the proposed wav2 + bert(CMD) was adopted to propagate supervisory signals for a target label to the lowest level of features (*end-to-end learning*), with the high performance and the smallest variance. Out-of-memory usually occurs when multiple large pretrained models are used; however, the proposed offline distillation method achieved high performance and learning stability while minimizing the number of model parameters. Regarding the ratio of compressed parameters in the proposed CMD, distilled parameters were compressed by approximately 95% (94 M → 5 M) of Wav2vec 2.0 and by 78% (109 M → 24 M) of BERT.

Unimodal Performance. Fig. 7 presents a boxplot of results from repeated experiments on emotion classification under the assumption that data with solely one modality were used. It is presented as a subplot of each type of data, where the y-axis of each subplot represents the macro-average F1-scores from the repeated experiments, while the x-axis denotes a total of eight experimental levels, which include unimodal models (mfcc, glove, wav2, bert), unimodal models distilled in CMD manner (wav2 + distill and bert + distill), a unimodal model distilled in CMD manner using text generated from Wav2vecCTC[8] that is used to achieve ASR (bert(asr)+distill), and a multimodal model executing CMD in a multimodal dataset form by combining the generated text and audio (wav2 + bert(asr)+CMD).

The calculated macro-average F1-scores drawn in the boxplots, including the mean and standard deviation of the F1-scores of the emotion labels, are presented in Table 4. As demonstrated in the results of (a) IEMOCAP and (c) CMU–MOSEI, the feature encoder performance of wav2 and bert was further improved than mfcc and glove, while the models that used the proposed contrastive learning to only one unimodal case (wav2 + distill and bert + distill) exhibited better performance and lower variability than the models (wav2 and bert) that did not perform distillation. The bert(asr) +distill case that used the ASR model contains uncertainty in its text generation, thus exhibiting poorer performance and higher variability than the bert + distill case. However, the best performance was achieved in the wav2 + bert(asr)+CMD case, where the generated text and audio were applied to the proposed CMD. In the results of (b) MELD, performance degradation was observed in the bert(asr)+distill case, where the ASR model was adopted, and performance improvement was minimal in the wav2 + bert(asr) +CMD case. The reason can be attributed to the text generation performance of the ASR model. Table 5 presents the results obtained from evaluating the text generation performance of ASR based on character-level error rate (CER) and word-level error rate (WER) [51]. The MELD dataset contains twice as much CER and WER as IEMOCAP and CMU–MOSEI, thus exhibiting relatively poorer ASR performance. Hence, owing to the low quality of the generated texts, the performance improvement with the ASR model can be considered limited. Table A.8 provides examples of actual and predicted text of the ASR model. In specific audio files, the uncertainty of ASR was observed in parts with background noise or involvement of multi-speakers. Provided ASR performance is guaranteed, the learning models can be sufficiently improved using the proposed CMD.

Importance of the modalities. The models used for multi-class emotion classification consist of transformers for audio and text modalities. To elucidate the features that are more important between audio and text, the gradients of the target function with respect to logits for audio and text were adopted. Table 6 presents the comparison of the order relationship of the gradients with respect to audio and text modality, and the modality with relatively greater gradients was defined as the dominant modality. For example, $\frac{\partial \mathscr{L}_{ce}}{\partial \mathbf{y}_{\text{audio}}} > \frac{\partial \mathscr{L}_{ce}}{\partial \mathbf{y}_{\text{text}}}$ is dominant for audio, otherwise, it is dominant for text. The third column in Table 6 shows the ratio of the dominant modality for each data where text modality contributes to the emotion classification for all datasets by 76.2% and 100%, at a minimum and maximum, respectively. Considering that the performance improvement from bert + distill to wav2 + bert + distill in IEMOCAP, MELD, and CMU–MOSEI is 4.69%, 0.58%, 2.42%, respectively, in Fig. 6, the text feature($\mathbf{c}_{\text{text}}$) computed by the cross-attention method $f_\theta(\mathbf{x}^q_{\text{text}}, \mathbf{x}^k_{\text{audio}}, \mathbf{x}^v_{\text{audio}})$ contributed more to emotion classification than the audio feature($\mathbf{c}_{\text{audio}}$). Furthermore, when the mean and standard deviation of the length of text tokens were calculated per group in each dominant modality, audio modality contributed more to emotion classification as the sentences become shorter. MELD, with its short average length of raw text tokens, exhibited the highest rate of dominance to audio at 23.8%. In contrast, CMU–MOSEI, which has a long average length of raw text tokens, was only dominant for text.

Visualization. Because the previously explained text features contributed significantly to emotion classification, we performed visualization to identify the tokens in the text that are crucial. Gradient-based visualization [52], which is frequently used for

[8] Wav2vecCTC is the Wav2vec 2.0 finetuned on Librispeech 960 h based on CTC loss [50], and it converts the output of Wav2vec 2.0 into text using a Viterbi decoder.

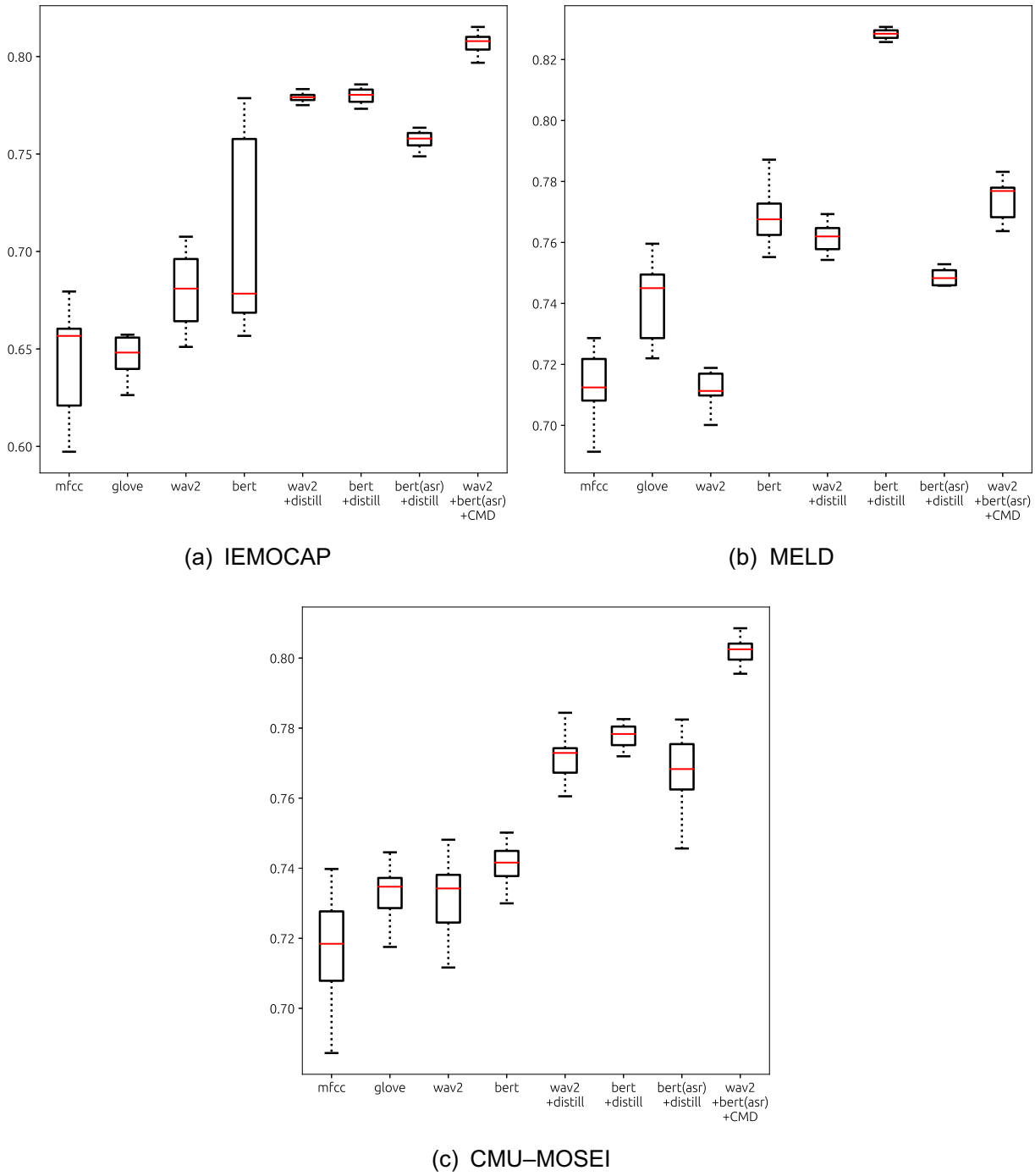

Fig. 7. Experimental results of unimodal cases suggested in Table 2; boxplot of F1-scores calculated from data from repea.ted experiments.

localization, was performed. Table 7 presents the localized texts for each emotion label for IEMOCAP. The red gradation in the table expresses the relative importance of the tokens given for each emotion. The significance is defined as the gradients of the target function with respect to the contextual embedding of text tokens and can be expressed in an equation as $\frac{\partial \mathcal{L}_{ce}}{\partial \mathbf{c}_{\text{text}}}$. Consequently, awesome/great/right/good, etc. tokens were selected for the happy label, while hell/goddamn/hate/insulting, etc. tokens were selected for the angry label. In IEMOCAP, happy (86.62%) and anger (88.18%) F1-scores have a better performance than sad (74.81%) and neutral (81.08%) because large numbers of words directly related to happy and anger emotions were included in the sentences. In contrast, ambiguous tokens were mostly selected for sad and neutral labels, where "try to do something" and "i didn't come here to get" for the sad label and "okay?", "bad thing good

Table 4
Average F1-scores with standard deviations for emotion labels based on the *unimodal models*. The first column in the table shows the data used in the experiments, and the second column shows the models used for learning. From the third to the sixth column, F1-scores of emotion labels are presented. Macro-average f1–score (Macro–f1), which is used as an integrated metric, is shown in the seventh column. The experimental results present the mean (standard deviation) of 10 values deduced from the repeated experiments.

Data	Learning Strategy	Neutral	Happy	Sad	Anger	Macro–F1
IEMOCAP	mfcc	0.6295 (0.0566)	0.6787 (0.0174)	0.5224 (0.0728)	0.7444 (0.0631)	0.6438 (0.0271)
	glove	0.673 (0.0236)	0.6693 (0.048)	0.5106 (0.0639)	0.7537 (0.0103)	0.6517 (0.0298)
	wav2	0.6901 (0.0129)	0.6763 (0.0118)	0.5772 (0.0364)	0.7771 (0.0335)	0.6802 (0.0197)
	bert	0.7086 (0.0412)	0.7199 (0.0607)	0.6117 (0.0703)	0.7832 (0.0373)	0.7058 (0.0503)
	wav2(distill)	0.7667 (0.0082)	0.7986 (0.0072)	0.6801 (0.0081)	0.8704 (0.0039)	0.779 (0.0029)
	bert(distill)	0.7578 (0.0149)	0.8293 (0.0102)	0.6906 (0.0058)	0.8413 (0.0025)	0.7798 (0.0063)
	bert(asr)+(distill)	0.7334 (0.0116)	0.7873 (0.0101)	0.6785 (0.0056)	0.8261 (0.0069)	0.7563 (0.0066)
	wav2 + bert(asr)+(CMD)	**0.7847 (0.0072)**	**0.8458 (0.013)**	**0.7222 (0.0069)**	**0.8733 (0.0036)**	**0.8065 (0.0059)**
MELD	mfcc	0.4945 (0.0288)	0.7344 (0.0106)	0.8496 (0.0181)	0.7721 (0.0099)	0.7126 (0.0119)
	glove	0.5965 (0.046)	0.7585 (0.0305)	0.8422 (0.0182)	0.779 (0.0094)	0.7441 (0.0188)
	wav2	0.5064 (0.0236)	0.7306 (0.01)	0.8551 (0.0058)	0.762 (0.0155)	0.7135 (0.0084)
	bert	0.6553 (0.0273)	0.7699 (0.0191)	0.8575 (0.0048)	0.7904 (0.0102)	0.7683 (0.0096)
	wav2(distill)	0.6187 (0.0116)	0.7498 (0.0051)	0.8621 (0.0011)	0.8139 (0.0079)	0.7611 (0.0043)
	bert(distill)	0.768 (0.0068)	0.8304 (0.0075)	0.8766 (0.0023)	0.8382 (0.0067)	0.8283 (0.0025)
	bert(asr)+(distill)	0.5661 (0.0246)	0.7674 (0.0153)	0.8668 (0.0044)	0.783 (0.0037)	0.7458 (0.0106)
	wav2 + bert(asr)+(CMD)	**0.6278 (0.0188)**	**0.7775 (0.0123)**	**0.8697 (0.0042)**	**0.821 (0.0061)**	**0.774 (0.0067)**
CMU–MOSEI	mfcc	0.5704 (0.0421)	0.5293 (0.0567)	0.8805 (0.0177)	0.8861 (0.015)	0.7166 (0.0154)
	glove	0.5929 (0.0258)	0.5591 (0.0166)	0.8909 (0.0065)	0.889 (0.0097)	0.733 (0.0077)
	wav2	0.5971 (0.0338)	0.5385 (0.0524)	0.8892 (0.0228)	0.8892 (0.0146)	0.7285 (0.0169)
	bert	0.6197 (0.0203)	0.571 (0.0218)	0.888 (0.0153)	0.885 (0.0138)	0.7409 (0.0071)
	wav2(distill)	0.6366 (0.0132)	0.6206 (0.023)	0.9097 (0.0048)	0.9181 (0.0027)	0.7713 (0.0072)
	bert(distill)	0.6606 (0.0056)	0.6226 (0.0148)	0.9116 (0.0007)	0.9157 (0.0048)	0.7776 (0.0053)
	bert(asr)+(distill)	0.6426 (0.0176)	0.6221 (0.0231)	0.8959 (0.0317)	0.9096 (0.0047)	0.7675 (0.0111)
	wav2 + bert(asr)+(CMD)	**0.6827 (0.0064)**	**0.6855 (0.0098)**	**0.9142 (0.0091)**	**0.9232 (0.0047)**	**0.8014 (0.0048)**

Table 5
CER and WER performance of ASR model for each data.

	CER	WER
IEMOCAP	12.71%	22.75%
MELD	32.23%	45.76%
CMU–MOSEI	13.84%	23.64%

Table 6
Relative importance of the modality that employed the proposed wav2 + bert + CMD (third and fourth columns present the dominant modality ratio and the mean and standard deviation of the text token length for the dominant group, respectively).

Data	Dominant modality	Selection ratio	$\mu(\pm\sigma)$
IEMOCAP	Audio	9.5%	5.38($\pm$3.48)
	Text	90.5%	16.85($\pm$13.04)
MELD	Audio	23.8%	5.47($\pm$4.03)
	Text	76.2%	14.65($\pm$8.87)
CMU–MOSEI	Audio	0.0%	-
	Text	100.0%	28.70($\pm$22.70)

thing", and "oh" for the neutral label were localized, and the context of the sentences had indirect effects for emotion classification.

6. Conclusions

Fine-grained emotion classification is a field of research required to understand the characteristics related to a person's intentions or moods, in which emotion classification models can be applied to computer technologies that employ biometric sensors to detect physical characteristics. However, there are cases in which audio modality is required in addition to text to classify fine-grained emotions, and the collection of audio and text paired datasets related to fine-grained emotions is severely limited. To address this limitation, we proposed CMD, which can be extensively applied to various modalities, while applying Self-SL approaches that can adequately express audio and text modalities. The proposed CMD method applies contrastive learning to ensure that the feature space of student models comprising a few parameters becomes similar to the feature space of teacher models that can adequately represent the features of each modality. Regarding the representation learning used for CMD, cross- and self-attention are sequentially performed for audio and text data, while Wav2vec 2.0 and BERT deliver context information to the audio–text transformers.

The distilled audio–text transformers trained by the proposed CMD achieved the best fine-grained emotion classification performance among other benchmark models in the experiments conducted on multimodal datasets. Furthermore, the proposed model adopted a few parameters and exhibited low variability for the results of repeated experiments. When the effects of the distilled audio–text transformers and the emotion labels were examined using the gradient-based method, the results became more dependent on text modality as the sentences become longer and on audio modality as the sentences become shorter. More specifically, when the localization results of the text's sentence tokens were examined, the localized tokens captured the words expressing emotions to correspond to the meaning of the target labels. Consequently, the proposed model exhibited outstanding emotion classification performance and reproducibility of predictions, even with a few parameters, and is highly applicable to affective computing, which understands the characteristics related to a person's intention or mood. Moreover, the pretrained cross-modal model is guaranteed to perform well on different types of downstream tasks because it is pretrained to ensure that contextual features can be represented by the interaction between acoustic and textual information, regardless of the multi-class emotion classification task.

In addition, future research directions related to the proposed method are suggested. The prediction model proposed from the experimental results of this paper can be considered to be biased toward text modality. Hence, the performance of the proposed model can be further enhanced if the representation of audio features with relatively weaker influence can be learned more effectively. For example, the elimination of background noise, data augmentation, and multi-speaker voice filtering can be alternatives. Second, the proposed CMD was assumed to be given the data where audio and text are unaligned; hence, input embeddings can be applied by merging audio and text vectors with the same length

Table 7
Localized texts of IEMOCAP using wav2 + bert + CMD

Target Label	Localized Sentences
Happy	that ' ll be awesome . this is going to be so great . i have so many great ideas . u . s . c . has a major in that right . oh good . [laughter] oh good idea .
Angry	you have a business here . what the hell is this ? goddamn it aug ##ie don ' t ask me that . i hate it when you ask me that . you always ask me that . it ' s insulting . oh yes i am . . . oh yes i am . let go of me . you let go . you . . . ur ##gh . . . you ' re a cruel friend . i lo ##ath ##e and i hate you . marry you again . . . ha . . . never never never never . i hope you die in torment . you beast . stop it . stop it . i hate you . look at you you ' re shaking .
Sad	and try to do something . i mean - okay okay let ' s keep our voices down okay ? i didn ' t come here to get in yelling match . i just . . . i just basically lost someone that was really close to me . no no it ' s nothing like that . you know . i went to work with dad and i started that whole rat race again . . .
Neutral	you okay ? oh it ' s not bad thing it ' s good thing . hmm . okay yeah . i would imagine . oh . . . well . yeah of course .

if there is an alignment or a segment coordinate between audio frames and text tokens. Positional embeddings derived from the coordinates can also be adopted as additional input embeddings.

CRediT authorship contribution statement

Donghwa Kim: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Visualization, Writing – original draft. **Pilsung Kang:** Project administration, Supervision, Writing – review & editing, Resources, Funding acquisition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (NRF-2022R1A2C2005455). This work as also supported by the Korea Institute for Advancement of Technology (KIAT) grant funded by the Korea Government (MOTIE) (P0008691, The Competency Development Program for Industry Specialist).

Appendix A. The supplementary experimental results

Table A.8 and Fig. A.8.

Table A.8
The evaluation of generated text. For each dataset, comparison of actual and predicted values by the ASR model, which is Wav2vec 2.0 finetuned on Librispeech 960 h based on CTC loss [50].

Dataset	Sample Index	Value	Sentence
IEMOCAP	1	actual	Oh I know I know.
		predicted	All i know i know so
	2	actual	My name is Brent Caption
		predicted	Might have made his brench caption
	3	actual	Well don't you understand that this is-
		predicted	Well don't you understand thatthis is
	4	actual	I worked really hard I'll say I worked really hard. You know I I'm not bragging or anything like that. But-
		predicted	I worked really hard i'll say i worked really hard you know i'm not bragging or anything like that buta
	5	actual	You guys have been-
		predicted	Yo go amin on o thet pers
MELD	1	actual	"Oh my God, he's lost it. He's totally lost it."
		predicted	Ty god he's lost it he's totally lost im
	2	actual	"Or! Or, we could go to the bank, close our accounts and cut them off at the source."
		predicted	Or we could go to the bank close our accounts and cut them off at the sorce
	3	actual	You're a genius!
		predicted	Your a genious
	4	actual	"Y'know, he hums when he pees!"
		predicted	Ow he comes when he peas
	5	actual	Hey!
		predicted	A atin
CMU–MOSEI	1	actual	My name is Dr. Erma Jean Sims I'm a member of the faculty of the School of Education at Sonoma State University."
		predicted	My name is doctor irmigene simms i'm a member of the faculty in the school of education at zenoma state university
	2	actual	"It's important for teachers to understand their legal, ethical, and professional obligations in an educational setting."
		predicted	It's important for teachers to understand their legal ethical and professional obligations in an educational setting
	3	actual	Noncompliance can result in legal liability to you and to the school district in which you are employed.
		predicted	Non compliance can result in legal liability to you and to the school district in which you are employed
	4	actual	"is, you can say, hey I really like baby skin, they are so soft, they don't have any hair on their face"
		predicted	Is even say hey you're not i really like babyes kid they're so soft they don't have any hair on their face it's so
	5	actual	"It's so fun, and it's a great workout"
		predicted	It's so fine and it's a great workout

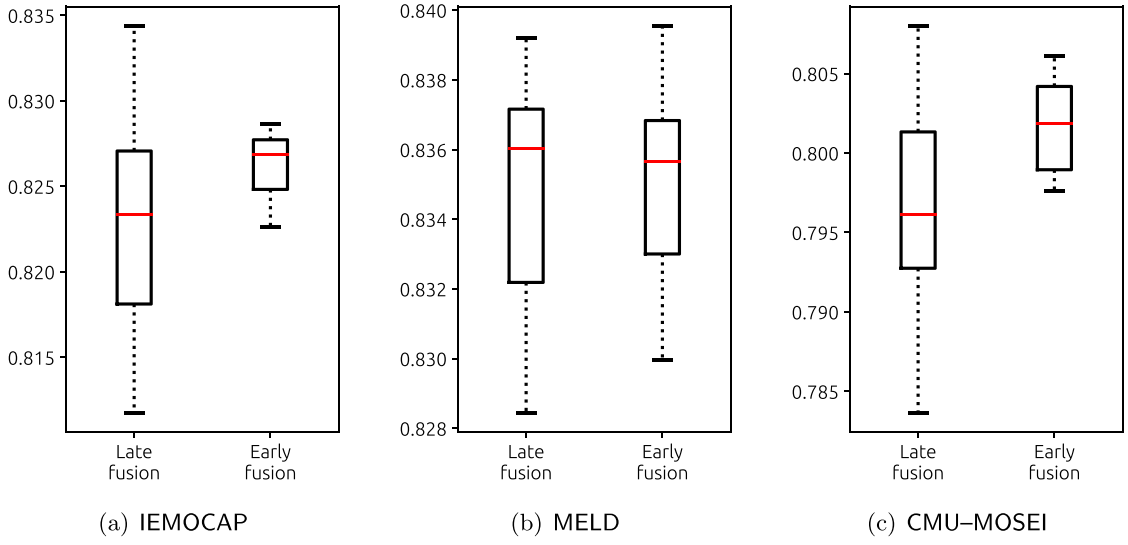

Fig. A.8. Comparison between early fusion and late fusion of audio–text transformers. Boxplot of the results obtained from repeated experiments on emotion classification according to the fusion method. Here, a subplot of each type of data is presented, and the y-axis of each subplot represents the macro-average F1-scores from the repeated experiments. In the x-axis, early fusion indicates the cases in which the attention mechanism are adopted in the order of cross- and self-attention modules, while late fusion indicates the cases in which the attention mechanism are utilized in the order of self- and cross-attention modules. In the experimental results obtained using (a) IEMOCAP and (c) CMU–MOSEI, the architecture of early fusion exhibits a significantly lower variability of prediction than late fusion, whereas no significant difference is observed between the fusion methods in (b) MELD containing noises (e.g., background noise or multi-speaker intervention).

References

[1] E. Cambria, D. Das, S. Bandyopadhyay, A. Feraco, Affective computing and sentiment analysis, in: A practical guide to sentiment analysis, Springer, 2017, pp. 1–10.

[2] K. Mouthami, K.N. Devi, V.M. Bhaskaran, Sentiment analysis and classification based on textual reviews, in: 2013 international conference on Information communication and embedded systems (ICICES), IEEE, 2013, pp. 271–276.

[3] E. Guzman, W. Maalej, How do users like this feature? A fine grained sentiment analysis of app reviews, in: 2014 IEEE 22nd international requirements engineering conference (RE), IEEE, 2014, pp. 153–162.

[4] T.-P. Jung, T.J. Sejnowski, et al., Multi-modal approach for affective computing, in: 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), IEEE, 2018, pp. 291–294.

[5] R.E.S. Panda, R. Malheiro, B. Rocha, A.P. Oliveira, R.P. Paiva, Multi-modal music emotion recognition: A new dataset, methodology and comparative analysis, in: 10th International Symposium on Computer Music Multidisciplinary Research (CMMR 2013), 2013, pp. 570–582.

[6] J.-H. Lee, H.-J. Kim, Y.-G. Cheong, A multi-modal approach for emotion recognition of tv drama characters using image and text, in: 2020 IEEE International Conference on Big Data and Smart Computing (BigComp), IEEE, 2020, pp. 420–424.

[7] S. Moncrieff, S. Venkatesh, G. West, S. Greenhill, Multi-modal emotive computing in a smart house environment, Pervasive Mobile Comput. 3 (2) (2007) 74–94.

[8] Y. Lei, S. Yang, L. Xie, Fine-grained emotion strength transfer, control and prediction for emotional speech synthesis, in: 2021 IEEE Spoken Language Technology Workshop (SLT), IEEE, 2021, pp. 423–430.

[9] T. Li, S. Yang, L. Xue, L. Xie, Controllable emotion transfer for end-to-end speech synthesis, in: 2021 12th International Symposium on Chinese Spoken Language Processing (ISCSLP), IEEE, 2021, pp. 1–5.

[10] N. Tits, K.E. Haddad, T. Dutoit, Asr-based features for emotion recognition: A transfer learning approach, arXiv preprint arXiv:1805.09197.

[11] Y.-S. Seo, J.-H. Huh, Automatic emotion-based music classification for supporting intelligent iot applications, Electronics 8 (2) (2019) 164.

[12] L.Y. Mano, B.S. Faiçal, L.H. Nakamura, P.H. Gomes, G.L. Libralon, R.I. Meneguete, P.R. Geraldo Filho, G.T. Giancristofaro, G. Pessin, B. Krishnamachari, et al., Exploiting iot technologies for enhancing health smart homes through patient identification and emotion recognition, Comput. Commun. 89 (2016) 178–190.

[13] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805.

[14] C. Sun, L. Huang, X. Qiu, Utilizing bert for aspect-based sentiment analysis via constructing auxiliary sentence, arXiv preprint arXiv:1903.09588.

[15] H. Xu, B. Liu, L. Shu, P.S. Yu, Bert post-training for review reading comprehension and aspect-based sentiment analysis, arXiv preprint arXiv:1904.02232.

[16] N. Majumder, S. Poria, H. Peng, N. Chhaya, E. Cambria, A. Gelbukh, Sentiment and sarcasm classification with multitask learning, IEEE Intell. Syst. 34 (3) (2019) 38–43.

[17] J. Bhaskar, K. Sruthi, P. Nedungadi, Hybrid approach for emotion classification of audio conversation based on text and speech mining, Proc. Comput. Sci. 46 (2015) 635–643.

[18] G. Xu, W. Li, J. Liu, A social emotion classification approach using multi-model fusion, Future Gener. Comput. Syst. 102 (2020) 347–356.

[19] A. Houjeij, L. Hamieh, N. Mehdi, H. Hajj, A novel approach for emotion classification based on fusion of text and speech, in: 2012 19th International Conference on Telecommunications (ICT), IEEE, 2012, pp. 1–6.

[20] A.B. Zadeh, P.P. Liang, S. Poria, E. Cambria, L.-P. Morency, Multimodal language analysis in the wild: Cmu-mosei dataset and interpretable dynamic fusion graph, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2018, pp. 2236–2246.

[21] O. Chapelle, B. Scholkopf, A. Zien, Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews], IEEE Trans. Neural Networks 20(3) (2009) 542–542.

[22] D. Kim, D. Seo, S. Cho, P. Kang, Multi-co-training for document classification using various document representations: Tf–idf, lda, and doc2vec, Inf. Sci. 477 (2019) 15–29.

[23] A. Baevski, H. Zhou, A. Mohamed, M. Auli, wav2vec 2.0: A framework for self-supervised learning of speech representations, arXiv preprint arXiv:2006.11477.

[24] H. Alwassel, D. Mahajan, B. Korbar, L. Torresani, B. Ghanem, D. Tran, Self-supervised learning by cross-modal audio-video clustering, arXiv preprint arXiv:1911.12667.

[25] Z. Zhang, F. Ringeval, B. Dong, E. Coutinho, E. Marchi, B. Schüller, Enhanced semi-supervised learning for multimodal emotion recognition, in: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2016, pp. 5185–5189.

[26] S. Li, Z. Wang, G. Zhou, S.Y.M. Lee, Semi-supervised learning for imbalanced sentiment classification, in: Twenty-Second International Joint Conference on Artificial Intelligence, 2011.

[27] A. Khare, S. Parthasarathy, S. Sundaram, Self-supervised learning with cross-modal transformers for emotion recognition, in: 2021 IEEE Spoken Language Technology Workshop (SLT), IEEE, 2021, pp. 381–388.

[28] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in neural information processing systems, 2017, pp. 5998–6008.

[29] S. Siriwardhana, A. Reis, R. Weerasekera, S. Nanayakkara, Jointly fine-tuning bert-like self supervised models to improve multimodal speech emotion recognition, arXiv preprint arXiv:2008.06682.

[30] A. Baevski, S. Schneider, M. Auli, vq-wav2vec: Self-supervised learning of discrete speech representations, arXiv preprint arXiv:1910.05453.

[31] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, arXiv preprint arXiv:1907.11692.

[32] G. Hinton, O. Vinyals, J. Dean, Distilling the knowledge in a neural network, arXiv preprint arXiv:1503.02531.

[33] N. Komodakis, S. Zagoruyko, Paying more attention to attention: improving the performance of convolutional neural networks via attention transfer, ICLR (2017).

[34] F. Tung, G. Mori, Similarity-preserving knowledge distillation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 1365–1374.

[35] W. Nie, R. Chang, M. Ren, Y. Su, A. Liu, I-gcn: Incremental graph convolution network for conversation emotion detection, IEEE Trans. Multimedia.

[36] W. Nie, Y. Yan, D. Song, K. Wang, Multi-modal feature fusion based on multi-layers lstm for video emotion recognition, Multimedia Tools Appl. 80 (11) (2021) 16205–16214.

[37] J. Lei Ba, J.R. Kiros, G.E. Hinton, Layer normalization, ArXiv e-prints (2016) arXiv–1607.

[38] R. Wan, S. Mei, J. Wang, M. Liu, F. Yang, Multivariate temporal convolutional network: A deep neural networks approach for multivariate time series forecasting, Electronics 8 (8) (2019) 876.

[39] D. Hendrycks, K. Gimpel, Gaussian error linear units (gelus), arXiv preprint arXiv:1606.08415.

[40] M. Gutmann, A. Hyvärinen, Noise-contrastive estimation: A new estimation principle for unnormalized statistical models, in: Proceedings of the thirteenth international conference on artificial intelligence and statistics, JMLR Workshop and Conference Proceedings, 2010, pp. 297–304.

[41] C. Guo, G. Pleiss, Y. Sun, K.Q. Weinberger, On calibration of modern neural networks, in: International Conference on Machine Learning, PMLR, 2017, pp. 1321–1330.

[42] S. Dieleman, A. v. d. Oord, K. Simonyan, The challenge of realistic music generation: modelling raw audio at scale, arXiv preprint arXiv:1806.10474.

[43] T. Chen, S. Kornblith, M. Norouzi, G. Hinton, A simple framework for contrastive learning of visual representations, in: International conference on machine learning, PMLR, 2020, pp. 1597–1607.

[44] S.-I. Amari, Backpropagation and stochastic gradient descent method, Neurocomputing 5 (4–5) (1993) 185–196.

[45] Y.-H.H. Tsai, S. Bai, P.P. Liang, J.Z. Kolter, L.-P. Morency, R. Salakhutdinov, Multimodal transformer for unaligned multimodal language sequences, in: Proceedings of the conference. Association for Computational Linguistics. Meeting, Vol. 2019, NIH Public Access, 2019, p. 6558.

[46] P. Goyal, P. Dollár, R. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola, A. Tulloch, Y. Jia, K. He, Accurate, large minibatch sgd: Training imagenet in 1 hour, arXiv preprint arXiv:1706.02677.

[47] S. Poria, E. Cambria, D. Hazarika, N. Majumder, A. Zadeh, L.-P. Morency, Context-dependent sentiment analysis in user-generated videos, in: Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1: Long papers), 2017, pp. 873–883.

[48] Y. Wang, Y. Shen, Z. Liu, P.P. Liang, A. Zadeh, L.-P. Morency, Words can shift: Dynamically adjusting word representations using nonverbal behaviors, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 7216–7223.

[49] Y.-H.H. Tsai, P.P. Liang, A. Zadeh, L.-P. Morency, R. Salakhutdinov, Learning factorized multimodal representations, arXiv preprint arXiv:1806.06176.

[50] A. Graves, S. Fernández, F. Gomez, J. Schmidhuber, Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks, in: Proceedings of the 23rd international conference on Machine learning, 2006, pp. 369–376.

[51] D. Jan, L. Zeevaert, Receptive multilingualism: Linguistic analyses, language policies and didactic concepts, vol. 6, John Benjamins Publishing, 2007.

[52] R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-cam: Visual explanations from deep networks via gradient-based localization, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 618–626.

Donghwa Kim is an AI researcher in the Kakao Style Corp, Republic of Korea. He received B.S. in Industrial Engineering at Seoul National University of Science and Technology, and Ph.D in Industrial and Management Engineering at Korea University. His main research interest is representation learning for unstructured data and applying them to solve engineering problems in the field of image processing, language modeling, and speech recognition. He has published a number of papers on related topics in leading journals such as Information Sciences and IEEE Transactions on Semiconductor Manufacturing.

Pilsung Kang is an associate professor in the School of Industrial and Management Engineering, Korea University, Republic of Korea. He received B.S. and Ph.D in Industrial Engineering at Seoul National University. His main research interest is developing machine learning algorithms and applying them to solve engineering problems in the field of manufacturing, security, IT service, and healthcare. He has published a number of papers on related topics in leading journals such as Information Sciences and Pattern Recognition.