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

Meta-Learning for Speech Emotion Recognition Considering Ambiguity of Emotion Labels

Takuya Fujioka[1], *Takeshi Homma*[1], *Kenji Nagamatsu*[1]

[1]Research & Development Group, Hitachi, Ltd., Tokyo, Japan.
{takuya.fujioka.qh, takeshi.homma.ps, kenji.nagamatsu.dm}@hitachi.com

Abstract

Emotion labels in emotion recognition corpora are highly noisy and ambiguous, due to the annotators' subjective perception of emotions. Such ambiguity may introduce errors in automatic classification and affect the overall performance. We therefore propose a dynamic label correction and sample contribution weight estimation model. Our model is based on a standard BLSTM model with attention with two extra parameters. The first learns a new corrected label distribution and aims to fix the inaccurate labels in the dataset. The other estimates the contribution of each sample to the training process and aims to ignore the ambiguous and noisy samples while giving higher weights to the clear ones. We train our model through an alternating optimization method, where in the first epoch we update the neural network parameters, and in the second we keep them fixed to update the label correction and sample importance parameters. When training and evaluating our model on the IEMOCAP dataset, we obtained a weighted accuracy (WA) and unweighted accuracy (UA) of 65.9% and 61.4%, respectively. This yielded an absolute improvement of 2.3% and 1.9%, respectively, compared to a BLSTM with attention baseline, trained on the corpus gold labels.

Index Terms: speech emotion recognition, meta-learning

1. Introduction

Automatic recognition of emotion is important to enable more natural and engaging communication between humans and machines. In this work we concentrate on emotion recognition from speech, which is the task of estimating the emotional content of a spoken utterance.

In the past, emotion recognition was performed by extracting a set of low-level features from each frame of an audio sample. These features were then aggregated through various statistical aggregation functions (mean, standard deviation, minimum, maximum, etc.) to a global utterance-level vector representation [1], to be finally fed through a shallow classifier such as support vector machines (SVMs) [2, 3]. However, in recent years, the accuracy of speech emotion recognition has dramatically improved with the introduction of deep neural networks (DNNs). Initial DNN-based models [4] were still based on the same utterance-level feature extraction. However, in subsequent approaches, speech features extracted from each frame were used as inputs of more complex neural network architectures such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), and the accuracy was further improved [5, 6, 7]. Recent years have seen the application of novel methods developed from other AI fields, such as self-attention models [8], connectionist temporal classification (CTC) [9] and dilated residual network (DRN) [10]. Even higher performance was achieved by using multi-modal information, such as audio and image together with speech [11].

While most of the studies concentrated on the development of more accurate classification models, there were other aspects of emotion classification regarding the data itself that could help improve the performance but were mostly ignored. In many datasets, the emotion labels are annotated based on human annotators' perceptions to emotion. Emotion perception is highly subjective [12]; therefore, the labels often contain some noise due to humans' decision ambiguities. For instance, an annotator may assign the label *neutral* not when the sample is actually neutral, but when he/she is unsure about the most appropriate emotion class. Likewise, he may mistakenly recognize some loud enthusiastic speech as angry, but actually it is happy. Training a model on such noisy labels is likely the cause of some performance degradation because the model may become confused and may not clearly distinguish one emotion from another.

Another important issue is that, in many emotion recognition datasets, the numbers of utterances for each emotional category are imbalanced. Generally, in a classification task using these category imbalanced datasets, the accuracy of the small class is decreased [13, 14], which in turn affects the overall accuracy. To overcome these problems, some methods have been proposed to use soft target approaches to correct the annotation ambiguities [15] or to augment the dataset with synthetic data to reduce the effect of the data imbalance [16]. However, the former method only performs a static label contribution estimation based on the original annotation data, while the latter method is complex and the generated data might still be affected by the original labeling noise. In other domains, such as image recognition, similar issues were tackled by performing a label update, not a priori but during training, by gradually tuning the estimation [17].

Inspired by the achievements in image recognition [17], we propose a method to automatically tune the contribution of each data sample during training. We do this by alternately updating the parameters of a DNN emotion classification model, and then use the neural network prediction to correct the relative contribution and the target labels of each sample, to reduce the overall loss. The main purpose is to correct or ignore altogether the ambiguously labeled utterances, while giving higher importance to the clear and unambiguous ones. The results obtained in the interactive emotional dyadic motion capture (IEMOCAP) dataset [18] show that our proposed method is effective in removing the annotation noise. It achieves an improvement of 2.5% for weighted accuracy, and of 2.7% for unweighted accuracy compared to a state-of-the-art BLSTM model trained on the original labels only [7].

2. Methodology

Given an input audio sample $\mathbf{x}_n = [\mathbf{x}_{n,1}, \mathbf{x}_{n,2}, \cdots, \mathbf{x}_{n,T}]$, where n is the utterance index, $\mathbf{x}_{n,t}$ is the frame-based feature vector, and T is the total number of frames, we want

http://dx.doi.org/10.21437/Interspeech.2020-1082

to estimate the probabilities of each emotion category $\mathbf{y}_n = [y_{n,1}, y_{n,2}, \cdots, y_{n,C}]$, where C is the number of discrete emotion classes. We use a BLSTM model with attention [7] to perform the classification.

To improve the classification performance and reduce the ambiguities of the human-annotated labels during training, for each training speech sample we also learn two parameters: $\boldsymbol{l}_n = [l_{n,1}, l_{n,2}, \cdots, l_{n,C}]$, and w_n. $\boldsymbol{l}_n$ represents a new estimate of each sample emotion class, aiming to correct the ambiguities and inaccuracies during the training process, while through w_n we learn the contribution weight for each training utterance.

2.1. Emotion classification model

Fig. 1 shows the structure of our main BLSTM emotion recognition model, which closely follows the state-of-the-art by [7]. At first, we input the feature sequence $\mathbf{x}_n$ through a bi-directional LSTM (BLSTM), which yields $\boldsymbol{h}_n = [\boldsymbol{h}_{n,1}, \boldsymbol{h}_{n,2}, \cdots, \boldsymbol{h}_{n,T}]$ as the output. We then weigh the contribution of each frame through an attention layer, where its weights $\alpha_{n,t}$ are calculated as follows:

$$\alpha_{n,t} = \frac{\exp(\boldsymbol{h}_{n,t}\boldsymbol{u}^\top)}{\sum_{\tau=1}^{T}\exp(\boldsymbol{h}_{n,\tau}\boldsymbol{u}^\top)}. \quad (1)$$

In the equation above, $\boldsymbol{u} = [u_1, u_2, \cdots, u_C]$ are the learned attention parameters. The obtained attention weights $\alpha_{n,t}$ are used to calculate the weighted average over time of the BLSTM output vectors, to get a fixed-length utterance-level vector representation $\boldsymbol{h}'_n$. We get the output emotion probabilities $\mathbf{y}_n$ by applying a softmax layer to $\boldsymbol{h}'$:

$$\boldsymbol{h}'_n = \sum_{t=1}^{T}\alpha_{n,t}\boldsymbol{h}_{n,t}, \quad (2)$$

$$y_{n,c} = \frac{\exp(h'_{n,c})}{\sum_{c=1}^{C}\exp(h'_{n,c})}. \quad (3)$$

2.2. Update of target labels and contribution weights

Normally, an emotion classification system such as the one explained in the previous section, is trained from the gold standard labels $\mathbf{y}_n$, with all the training samples having the same contribution weight during training. This is however not ideal for emotion recognition since the human-annotated labels may be ambiguous or not precise. For instance, a sample can have been marked as "neutral" just because the annotators were unsure about the most appropriate emotion label, not because it was actually neutral.

We therefore correct these inaccuracies and ambiguities by learning two extra parameters. The first one is $\boldsymbol{l}_n = [l_{n,1}, l_{n,2}, \cdots, l_{n,C}] \in \{0, 1\}$, which is inspired by [17]. We initialize it with the one-hot gold standard emotion label. Through the learning process, this parameter is supposed to learn, for each training sample, the correct emotion distribution, eventually overriding the one previously assigned by the annotators. The second parameter, w_n, is the per-sample contribution weight. We initialize w_n by applying the same method proposed in [7], by taking the proportion of samples in each emotion category. In [7], w_n was proposed to address the class imbalance and prevent the recall degradation due to it, and w_n was kept fixed throughout the training. We instead update it during training, as we assume that the model would learn to give higher weights to clear samples, and lower weights to ambiguous samples that presumably would only add noise to the classifier.

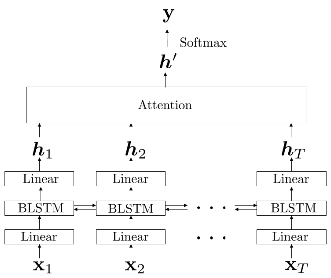


Figure 1: *Structure of BLSTM model with attention.*

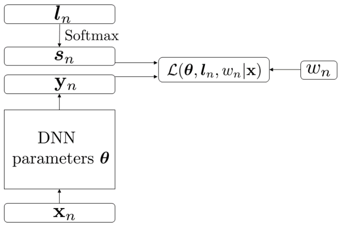

l_n
Softmax
s_n
$\mathbf{y}_n$
$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{l}_n, w_n|\mathbf{x})$
w_n
DNN
parameters $\boldsymbol{\theta}$
$\mathbf{x}_n$

Figure 2: *Overview of proposed training framework.*

To update those parameters, and apply them in the classification process, we designed the framework shown in Fig. 2. The overall model loss function is defined as $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{l}_n, w_n|\mathbf{x}_n)$:

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{l}_n, w_n|\mathbf{x}_n) = -\frac{\sum_{c=1}^{C} s_{n,c}\log y_{n,c}}{w_n}, \quad (4)$$

where $\boldsymbol{s}_n$ is the mapping of $\boldsymbol{l}_n$ by softmax function to make it a probability distribution over emotions:

$$s_{n,c} = \frac{\exp l_{n,c}}{\sum_{\gamma=1}^{C}\exp l_{n,\gamma}}. \quad (5)$$

It is worth noticing that in the cross entropy function, we do not use the gold standard labels but only use the new learned emotion representation $\boldsymbol{l}_n$.

The model parameters $\boldsymbol{\theta}$, $\boldsymbol{l}_n$ and w_n are updated through an alternating optimization process, shown in Algorithm 1. Alternately, we first update $\boldsymbol{\theta}$, keeping $\boldsymbol{l}_n$ and w_n fixed, through one epoch. In the second step, we update $\boldsymbol{l}_n$ and w_n, keeping the BLSTM network weights fixed, through another one epoch. To avoid the algorithm converging on very high values of w_n when minimizing the loss function, we scale the value of w_n after each update to maintain the following constraint:

$$\frac{\sum_{n=1}^{N} w_n}{N} = 1. \quad (6)$$

Algorithm 1 Alternating optimization algorithm

```
for $i \leftarrow 1$ to $num\_epochs$ do
    update $\boldsymbol{\theta}^{(i+1)}$ using $\boldsymbol{L}^{(i)}$ and $\boldsymbol{w}^{(i)}$
    update $\boldsymbol{L}^{(i+1)}$ and $\boldsymbol{w}^{(i+1)}$ using $\boldsymbol{\theta}^{(i+1)}$
end for
```

3. Experiments

3.1. Corpus

To evaluate the performance of the proposed learning method, we use the IEMOCAP dataset [18], one of the most commonly used benchmark datasets in emotion recognition tasks. The corpus is organized into 5 sessions, in each of which two actors performed a conversation. The total number of speakers in the corpus is 10. We only considered the samples belonging to the four emotion categories of *happiness*, *sadness*, *neutral* and *anger*, to keep the analysis consistent with previous works [6, 7, 8, 9, 10, 11, 15, 16]. The number of utterances in each emotional class of each speaker is shown in Table 1. We performed a leave-one-speaker-out 10-fold cross-validation using a leave-one-out strategy [10]. We applied early-stopping criteria in all conditions to minimize the loss of the validation set [15].

3.2. Experimental setup

We extracted 32-dimensional acoustic features from the raw audio samples using the openSMILE toolkit [19]: 12-dimensional mel-frequency cepstral coefficients (MFCCs), loudness, fundamental frequency (F_0), voicing probability, zero-crossing rate, and their first order derivatives. The frame length and frame shift were set to 25 ms and 10 ms, respectively. All features were normalized by mean and standard deviation calculated over all of the utterance features in the training set.

The emotion classification model was composed of a fully-connected layer with rectified linear unit (ReLU), a BLSTM layer and a fully-connected layer. The numbers of hidden units were 512, 128 and 4, respectively. We applied dropout to all the layers; the dropout rate was 0.5. We used Adam [20] as an optimization algorithm.

We evaluated our model using two common evaluation measures in the previous works: weighted accuracy (WA) and unweighted accuracy (UA). We also calculated per-class precision, recall and F1-score, to get a performance estimate over each individual emotion class.

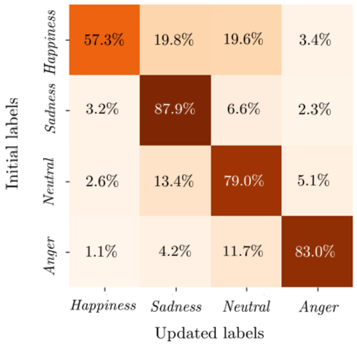
We compare our model (**BLSTM + ATT + $\boldsymbol{L}$ + $\boldsymbol{w}$**) against the following baselines:

- **BLSTM + ATT**: our reimplementation of the attention based BLSTM model as proposed in [7].
- **BLSTM + ATT + Oversampling/Undersampling**: same as the above, but applying oversampling or undersampling instead of $\boldsymbol{w}$ to address the class imbalance problem. In this baseline, utterances are randomyly resampled in each epoch.
- **BLSTM + ATT + $\boldsymbol{L}$**: the full model and training algorithm, but only updating $\boldsymbol{L}$, while keeping $\boldsymbol{w}$ fixed.
- **BLSTM + ATT + $\boldsymbol{w}$**: the full model and training algorithm, but only updating $\boldsymbol{w}$, while keeping $\boldsymbol{L}$ equal to the gold standard labels.
- **BLSTM + ATT + $\boldsymbol{L}$ + $\boldsymbol{w}$ pretrained**: similar to the full model, but the neural network was first pretrained using the gold standard labels.
- **Soft-target**: the soft label method proposed in [15].
- **Cycle-GAN**: the data augmentation method proposed in [16].

3.3. Results

Table 1: *Number of utterances in each emotion class and speaker.*

Speaker	*Happiness*	*Sadness*	*Neutral*	*Anger*
Ses01F	69	78	171	147
Ses01M	66	116	213	82
Ses02F	70	113	135	67
Ses02M	47	84	227	70
Ses03F	80	172	130	92
Ses03M	55	133	190	148
Ses04F	31	62	76	205
Ses04M	34	81	182	122
Ses05F	77	132	221	78
Ses05M	66	113	163	92
Total	595	1084	1708	1103



Initial labels	*Happiness*	*Sadness*	*Neutral*	*Anger*
Happiness	57.3%	19.8%	19.6%	3.4%
Sadness	3.2%	87.9%	6.6%	2.3%
Neutral	2.6%	13.4%	79.0%	5.1%
Anger	1.1%	4.2%	11.7%	83.0%

Figure 3: *Percentages of labels updated for each emotion category in* ***BLSTM + ATT + L + w****. Most of the updates affect the* happiness *class, which is often changed to* sadness *or* neutral.

The final results are shown in Table 2. For the Soft-target and Cycle-GAN baselines, we show the reported results from the original papers, since in the former case we were unable to replicate the same results, and the latter used a different methodology. Our proposed model performed the best in terms of WA, and the second best in terms of UA, achieving 65.9% and 61.4%, respectively. This yields an absolute improvement of +2.3% and +1.9%, respectively, over the **BLSTM + ATT** baseline. It is also worth noticing that the originally reported WA and UA in [7] are 63.5% and 58.8%, respectively. These values are not significantly different from the ones obtained by our reimplementation. The soft-target baseline achieved a 2.3% higher UA than our proposed method. However, they only used one-fold cross validation instead of ten-fold, and they did not report the performance for the individual emotion classes [15]; therefore, the results are not fully comparable.

In terms of per-class performance, our model achieves an F1-score of 35.7%, 67.1%, 63.3% and 75.9% for the *happiness*, *sadness*, *neutral* and *anger* classes, respectively, with absolute improvements of +0.5%, +2.1%, +2.4% and +1.9%, respectively. The lower improvement in the F1-score of *happiness* is compensated by a significant improvement in precision of +14.6%.

3.4. Discussion

By looking at the results in Table 2, it clearly emerges how our proposed model achieves a much better performance than just applying some simple imbalance corrections such as data undersampling or oversampling. In terms of performance, the introduction of per-sample importance weighting $\boldsymbol{w}$ had a slightly higher effect than emotion correction parameter $\boldsymbol{L}$ presumably because it is less sensitive to errors. $\boldsymbol{w}$ had the main effect of improving the precision in *happiness*, and of improving the precision in *anger*, while the main contribution of $\boldsymbol{L}$ was to improve the recall for *neutral* samples. Pretraining the model with the original labels did not seem to work better than starting by immediately updating $\boldsymbol{L}$ and $\boldsymbol{w}$, presumably due to a greater

Table 2: *Results, percentage, over each method. P: precision, R: recall, F1: F1-score, WA: weighted accuracy, UA: unweighted accuracy. *: reported values in the original papers.*

	Happiness			*Sadness*			*Neutral*			*Anger*				
Method	P	R	F1	P	R	F1	P	R	F1	P	R	F1	WA	UA
BLSTM + ATT [7]	38.8	35.3	35.2	**64.5**	68.2	65.0	**65.1**	58.3	60.9	73.0	76.1	74.0	63.6	59.5
BLSTM + ATT + Oversampling	42.6	36.0	**36.5**	62.8	62.2	61.4	63.7	59.6	60.1	74.3	78.6	75.5	63.5	59.1
BLSTM + ATT + Undersampling	35.4	34.7	34.4	64.5	63.1	62.9	64.9	61.2	62.4	73.3	74.4	72.7	63.2	58.4
Soft-target [15]	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	62.6*	**63.7***
Cycle-GAN [16]	N/A	**54***	N/A	N/A	69*	N/A	N/A	51*	N/A	N/A	69*	N/A	N/A	60.4*
BLSTM + ATT + $\boldsymbol{L}$	44.9	24.1	28.4	62.8	70.2	65.3	63.0	**66.4**	**63.9**	75.9	75.9	75.1	64.7	59.2
BLSTM + ATT + $\boldsymbol{w}$	45.4	30.5	34.7	61.9	71.7	65.3	63.8	63.7	63.1	**78.3**	77.2	**77.2**	65.2	60.8
BLSTM + ATT + $\boldsymbol{L}$ + $\boldsymbol{w}$ pretrained	51.0	22.6	29.4	62.7	68.8	65.1	58.9	65.4	61.6	75.6	74.9	74.7	64.4	57.9
BLSTM + ATT + $\boldsymbol{L}$ + $\boldsymbol{w}$	**53.4**	30.2	35.7	62.8	**74.1**	**67.1**	64.1	63.7	63.3	75.7	77.6	75.9	**65.9**	61.4

Table 3: *Mean values of the contribution weights w_n before and after training in **BLSTM + ATT + $\boldsymbol{L}$ + $\boldsymbol{w}$**. A lower value means higher importance to the loss function. The model assigns a much lower importance to the happiness label, which is presumably the most ambiguous also given the lower overall performance.*

	Mean values of contribution weights w_n			
	Happiness	*Sadness*	*Neutral*	*Anger*
Initial w_n	0.53	0.97	1.52	0.98
Learned w_n	1.32	0.92	1.08	0.84

Table 4: *Mean values of the contribution weights w_n before and after training in **BLSTM + ATT + $\boldsymbol{L}$ + $\boldsymbol{w}$** in the balanced dataset evaluation. In this evaluation, the mean values of $\boldsymbol{w}$ were less changed than those of the imbalanced dataset evaluation.*

	Mean values of contribution weights w_n			
	Happiness	*Sadness*	*Neutral*	*Anger*
Initial w_n	1.00	1.00	1.00	1.00
Learned w_n	1.02	0.98	1.06	0.94

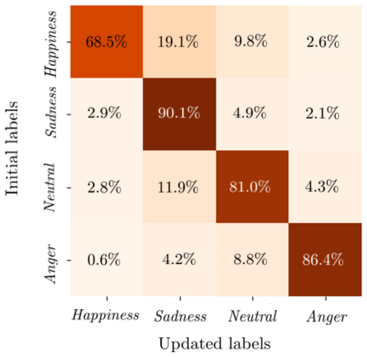


Figure 4: *Percentages of labels update for each emotion category in **BLSTM + ATT + $\boldsymbol{L}$ + $\boldsymbol{w}$** in the balanced dataset evaluation. The characteristics of the $\boldsymbol{L}$ transition are similar to those of the imbalanced dataset evaluation.*

learning bias over incorrect and ambiguous gold labels.

It is interesting to notice how these two parameters affect the various emotion classes after training. Table 3 shows the change of $\boldsymbol{w}$, a lower value means a greater weight of the loss function in eq. 4. The weight given to *happiness* samples, initially the less numerous class, was greatly reduced during training. By looking at the final precision and recall for this class, this seems to be a consequence of the very high ambiguity of the *happiness* annotations; the classifiers had great difficulty in clearly distinguishing *happiness* from other classes.

Likewise, we observed a similar behavior regarding the $\boldsymbol{L}$ parameter. Figure 3 shows the amount of label updates as learned by $\boldsymbol{L}$ during training. Only in around half of the cases was the label *happiness* kept; it was often changed into *sadness* or *neutral*. Besides *happiness*, in around 10% of the cases, *anger* was updated to *neutral*; *neutral* was updated to *sadness*. These latter changes are likely due to the aforementioned subjectivity of the emotion, and of the boundaries between them, which are leading to ambiguous choices.

To investigate the effect of updating $\boldsymbol{L}$ and $\boldsymbol{w}$ in more detail, we evaluated **BLSTM + ATT + $\boldsymbol{L}$ + $\boldsymbol{w}$** and **BLSTM + ATT** using the balanced dataset. In this dataset, the numbers of utterances in each emotion class is the same for each speaker. These utterances are extracted from the IEMOCAP dataset. As a result, $\boldsymbol{w}$ was updated as shown in Table 4, and $\boldsymbol{L}$ was updated as shown in Figure 4. In this evaluation, $\boldsymbol{w}$ was less changed through model training than in that of the imbalance dataset evaluation, and the characteristics of the $\boldsymbol{L}$ transition are similar to those of the imbalance dataset evaluation, especially regarding *happiness*, *anger* and *neutral*. This result supports the motivation of label correction by $\boldsymbol{L}$ and handling imbalance by $\boldsymbol{w}$. Besides, **BLSTM + ATT + $\boldsymbol{L}$ + $\boldsymbol{w}$** scored 61.5% and 59.4% for WA and UA, respectively, while **BLSTM + ATT** scored 60.1% and 56.9% for WA and UA, respectively. The contribution of updating $\boldsymbol{L}$ and $\boldsymbol{w}$ to improve WA and UA is also confirmed in this evaluation.

4. Conclusions

We have proposed a novel meta-learning approach, built on top of a traditional BLSTM with attention classifier, to address the issue of labeling inaccuracy and ambiguity in speech emotion recognition. Our proposed method is effective for dynamically updating each sample label during training, and learning an estimate of each sample contribution to reduce the relative weight of ambiguous utterances. We obtained an overall performance of 65.9% and 61.4%, for weighted and unweighted accuracy, respectively, on the IEMOCAP dataset, giving an absolute improvement of 2.5% and 2.7%, respectively over the same BLSTM model trained on the original gold labels. We also showed how our proposed framework clearly managed to reduce the importance of the most ambiguous label (*happiness*), and fix the initial label annotation to the most appropriate classes for each sample, thus improving the classification performance.

5. Acknowledgements

The authors would like to thank Dr. Dario Bertero for his valuable suggestions to improve the manuscript.

6. References

[1] B. Schuller, S. Steidl, A. Batliner, F. Burkhardt, L. Devillers, C. Müller, and S. S. Narayanan, "The interspeech 2010 paralinguistic challenge," in *INTERSPEECH*, 2010.

[2] B. Schuller, D. Arsic, F. Wallhoff, and G. Rigoll, "Emotion recognition in the noise applying large acoustic feature sets," in *Speech Prosody*, 2006.

[3] A. Álvarez, I. Cearreta, J. M. López, A. Arruti, B. Lazkano, E. Sieera, and N. Garay, "Feature subset selection based on evolutionary algorithms for automatic emotion recognition in spoken spanish and standard basque language," in *International conference on Text, Speech and Dialogue*, 2006.

[4] A. Stuhlsatz, C. Meyer, T. Eyben, F. Zielke, and B. Meier, G. Schuller, "Deep neural networks for acoustic emotion recognition: raising the benchmarks," in *ICASSP*, 2011.

[5] K. Han, D. Yu, and I. Tashev, "Speech emotion recognition using deep neural network and extreme learning machine," in *INTERSPEECH*, 2014.

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

[7] S. Mirsamadi, E. Barsoum, and C. Zhang, "Automatic speech emotion recognition using recurrent neural networks with local attention," in *ICASSP*, 2017.

[8] Y. Li, T. Zhao, and T. Kawahara, "Improved end-to-end speech emotion recognition using self attention mechanism and multitask learning," in *INTERSPEECH*, 2019.

[9] Z. Zhao, Z. Bao, Z. Zhang, N. Cummins, H. Wang, and B. Schuller, "Attention-enhanced connectionist temporal classification for discrete speech emotion recognition," in *INTERSPEECH*, 2019.

[10] R. Li, Z. Wu, J. Jia, S. Zhao, and H. Meng, "Dilated residual network with multi-head self-attention for speech emotion recognition," in *ICASSP*, 2019.

[11] J. Li and C. Lee, "Attentive to individual: a multimodal emotion recognition network with personalized attention profile," in *INTERSPEECH*, 2019.

[12] E. Douglas-Cowie, L. Cevillers, J. Martin, R. Cowie, S. Savvidou, S. Abrilian, and C. Cox, "Multimodal databases of everyday emotion: facing up to complexity," in *INTERSPEECH*, 2005.

[13] X. Liu, J. Wu, and Z. Zhou, "Exploratory undersampling for class-imbalance learning," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 39, no. 2, pp. 539–550, 2009.

[14] Z. Sun, Q. Song, and X. Zhu, "Using coding-based ensemble learning to improve software defect prediction," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 42, no. 6, pp. 1806–1817, 2012.

[15] A. Ando, S. Kobayashi, H. Kamiyama, R. Masumura, Y. Ijima, and Y. Aono, "Soft-target training with ambiguous emotional utterances for DNN-based speech emotion classification," in *ICASSP*, 2018.

[16] F. Bao, M. Neumann, and N. T. Vu, "CycleGAN-based emotion style transfer as data augmaentation for speech emotion recognition," in *INTERSPEECH*, 2019.

[17] D. Tanaka, D. Ikami, T. Yamasaki, and K. Aizawa, "Joint optimization framework for learning with noisy labels," in *CVPR*, 2018.

[18] C. Busso, M. Bulut, C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J. N. Chang, S. Lee, and S. S. Narayanan, "IEMOCAP: Interactive emotional dyadic motion capture database," in *Language resources and evaluation*, vol. 42, no. 4, 2008, pp. 335–359.

[19] F. Eyben, M. öllmer, and B. Schuller, "Opensmile: The munich versatile and fast open-source audio feature extractor," in *ACM Multimedia*, 2010, pp. 1459–1462.

[20] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.