STYLECAP: AUTOMATIC SPEAKING-STYLE CAPTIONING FROM SPEECH BASED ON SPEECH AND LANGUAGE SELF-SUPERVISED LEARNING MODELS

Kazuki Yamauchi[⋆⋆], *Yusuke Ijima*[†], *Yuki Saito*[⋆]

[⋆]The University of Tokyo, Japan, [†]NTT Corporation, Japan

ABSTRACT

We propose *StyleCap*, a method to generate natural language descriptions of speaking styles appearing in speech. Although most of conventional techniques for para-/non-linguistic information recognition focus on the category classification or the intensity estimation of pre-defined labels, they cannot provide the reasoning of the recognition result in an interpretable manner. StyleCap is a first step towards an end-to-end method for generating speaking-style prompts from speech, i.e., *automatic speaking-style captioning*. StyleCap is trained with paired data of speech and natural language descriptions. We train neural networks that convert a speech representation vector into prefix vectors that are fed into a large language model (LLM)-based text decoder. We explore an appropriate text decoder and speech feature representation suitable for this new task. The experimental results demonstrate that our StyleCap leveraging richer LLMs for the text decoder, speech self-supervised learning (SSL) features, and sentence rephrasing augmentation improves the accuracy and diversity of generated speaking-style captions. Samples of speaking-style captions generated by our StyleCap are publicly available[1].

Index Terms— Speaking styles, Natural language descriptions, Self-supervised learning model, Large language models

1. INTRODUCTION

Speech contains not only linguistic information but also para-/non-linguistic information [1]. The latter information, which includes the speaker's emotion and identity, can add variations to the spoken (i.e., linguistic) content and enrich human speech communication. Hence, such information from speech needs to be automatically recognized to develop human-oriented spoken language processing technologies, such as a natural conversational agent. Benefiting from the development of sophisticated deep learning architectures and techniques, such as Transformer [2] and self-supervised learning (SSL) [3] for extracting meaningful feature representations from speech, the accuracy of para-/non-linguistic information recognition has been improved significantly [4]–[6]. Moreover, the knowledge of deep learning-based technologies to recognize this information can be shared with other speech generative tasks, such as expressive text-to-speech (TTS) [7] and emotional voice conversion [8].

Another crucial factor for better para-/non-linguistic information recognition is the explainability of the recognition results. Some applications such as healthcare will require not only the recognition result but also its reasoning [9], which should be interpretable for users. However, most existing techniques focus on the category classification or the intensity estimation of labels defined by authors or dataset developers. One possible approach to solve this issue would be representing the evidence, i.e., para-/non-linguistic information in speech, with a natural language description. Several studies have attempted to acquire the relationship between para-/non-linguistic information and natural language descriptions [10], [11]. However, these methods still output class labels such as emotional states.

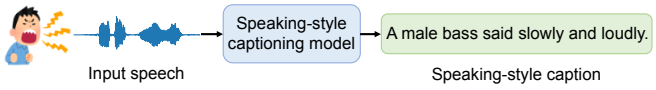

Fig. 1. Concept of automatic speaking-style captioning.

From a different viewpoint, generating natural language descriptions from input audio/image data can be regarded as a *captioning* task. For instance, audio captioning [12] and image captioning [13] are tasks to describe content information (objects and its behavior) in audio and image data, respectively. In these tasks, the remarkable progress of large-scale pre-trained deep neural network (DNN) models [14]–[16], has achieved the audio/image captioning performance better than conventional methods. Unlike these tasks, this paper focuses on para-/non-linguistic information rather than content information in speech. However, since this information cannot be directly observed in input speech [17], generating captions of such information can be regarded as a new challenge.

In this paper, we propose a deep learning technique for describing para-/non-linguistic information such as speaking style in speech with natural language, called *automatic speaking-style captioning* (Fig. 1). One way to train a DNN-based speaking-style captioning model is to use a sufficiently large dataset including many pairs of speech and sentences that describe various speaking styles, but such datasets have yet to be constructed. Instead, we combine the existing LibriTTS and PromptSpeech corpora and build a multi-speaker speech corpus paired with the natural language instructions of speaking styles (e.g., pitch, and speed) to train an end-to-end speaking-style captioning model consisting of a speech encoder and a text decoder. Inspired by the DNN-based method for image captioning [18], our proposed method, *StyleCap*, predicts prefix vectors fed into a large language model (LLM)-based text decoder from fixed-length speech representation vectors extracted by the speech encoder. In this paper, we experimentally explore an appropriate text decoder and speech representation suitable for this new task. In addition, since one speaking style can be described in various ways, automatic speaking-style captioning involves learning one-to-many mapping. To overcome this difficulty, we also introduce a simple data augmentation method, sentence rephrasing augmentation, that rephrases sentences using an LLM. We conduct an automatic style-captioning experiment to assess the effectiveness of the proposed method, using evaluation metrics commonly utilized in other natural language generation tasks. The experimental results demonstrate that our StyleCap leveraging richer LLMs for the text decoder, speech self-supervised learning (SSL) features, and sentence rephrasing augmentation improves the accuracy and diversity of generated speaking-style captions.

[⋆]This author performed the work while at NTT Corporation as an intern.

[1]https://ntt-hilab-gensp.github.io/icassp2024stylecap/

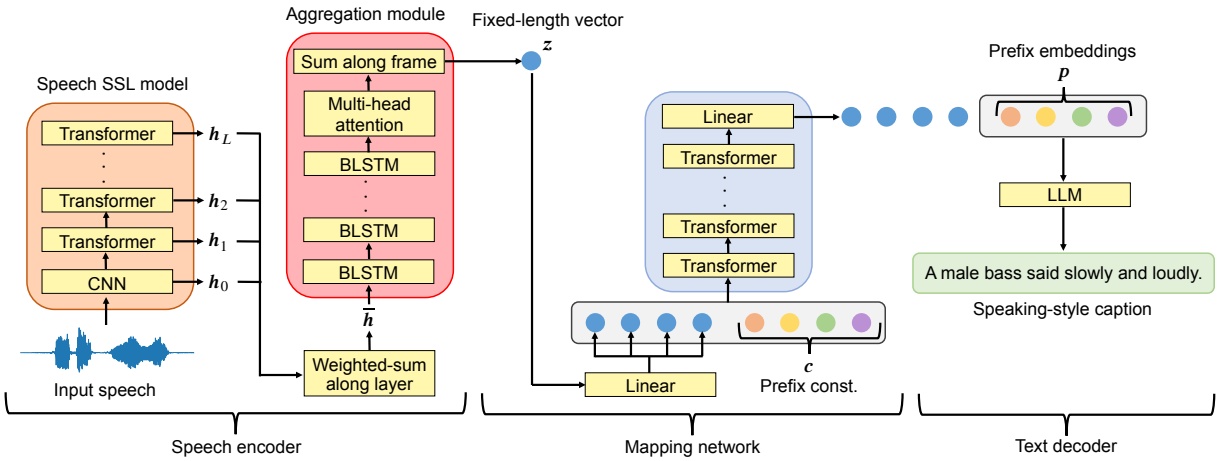

Fig. 2. DNN architecture of StyleCap, end-to-end model that automatically describes speaking style of input speech with natural language.

2. METHODS

2.1. Task overview

We define *automatic speaking-style captioning*, a novel task that explains the speaking style of input speech with a natural language descriptions. Specifically, a captioning model takes a single speaker's speech sample as the input and generates a sentence describing the speaker's speaking style, for instance, *"The speaker's sound height is normal, but the speed is very fast, and the volume is very low."*

Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be a speech waveform and a corresponding speaking-style caption (i.e., token sequence), respectively. Because the lengths of $\boldsymbol{x}$ and $\boldsymbol{y}$ are different, one can introduce a sequence-to-sequence model widely used in spoken language processing tasks, such as Transformer [2], into this captioning task. In Section 3, we evaluate the speaking-style captioning performance of this naive Transformer-based encoder-decoder model.

2.2. StyleCap: end-to-end automatic speaking-style captioning

StyleCap incorporates two SSL models for speech and text processing and achieves the end-to-end generative modeling of speaking-style prompts. Fig. 2 shows an overview of StyleCap, which is inspired by ClipCap [18], an existing model for automatic image captioning. Specifically, StyleCap consists of three DNNs: speech encoder, text decoder, and mapping network.

The speech encoder extracts a fixed-length feature vector from an input speech waveform. Considering the success in various spoken language processing tasks, we incorporate a speech SSL model into the speech encoder. First, all hidden vectors of the SSL model, $\boldsymbol{h}_l = [\boldsymbol{h}_{l,1}, \ldots, \boldsymbol{h}_{l,T}]^\top$ $(l = 1, \ldots, L)$, are extracted from a speech waveform $\boldsymbol{x}$. Then, the weighted-sum of the hidden layer outputs along layers for each frame index t, i.e., $\bar{\boldsymbol{h}}_t = \sum_l w_l \boldsymbol{h}_{l,t}$, is taken, where w_l is a trainable weight coefficient of the lth hidden layer output. Finally, the hidden vector sequence, $\bar{\boldsymbol{h}} = [\bar{\boldsymbol{h}}_1, \ldots, \bar{\boldsymbol{h}}_T]^\top$, is encoded by an aggregation module consisting of a stack of bidirectional long short-term memory (BLSTM) and multi-head attention (MHA) layers, and the outputs are then taken to be summed along the frame direction to obtain a D_{z}-dimensional fixed-length speech feature vector $\boldsymbol{z}$.

The text decoder leverages a pre-trained LLM and generates the speaking-style caption of the given speech waveform. Similar to the ClipCap text decoder, it first takes $K \times D_{\mathrm{w}}$ prefix embeddings $\boldsymbol{p} = [\boldsymbol{p}_1^\top, \ldots, \boldsymbol{p}_K^\top]^\top$ predicted by the mapping network from the feature vector $\boldsymbol{z}$, where K and D_{w} denote the prefix length and the word embedding dimensionality of the LLM, respectively. Then, the decoder autoregressively generates the tokens of speaking-style captions using the prefix embeddings as conditional vectors.

The mapping network projects the speech encoder output $\boldsymbol{z}$ onto the word embedding space, i.e., prefix embeddings $\boldsymbol{p}$. It consists of a stack of Transformer layers and $K \times D_{\mathrm{z}}$ trainable prefix constant parameters $\boldsymbol{c} = [\boldsymbol{c}_1^\top, \ldots, \boldsymbol{c}_K^\top]^\top$ to retrieve meaningful feature representation from $\boldsymbol{z}$ through MHA layers.

2.3. Data augmentation by rephrasing sentences using an LLM

Because one speaking style can be described in various ways, automatic speaking-style captioning essentially requires learning a one-to-many mapping similar to TTS. To mitigate the difficulty, we introduce sentence rephrasing augmentation using an LLM to increase the diversity of speaking-style captions in the training data. Specifically, we first ask the pre-trained Llama 2-Chat (7B)[2] [19], which is an LLM optimized for dialogue use cases, to generate five sentences from one given sentence on the basis of the following prompt: *"Rewrite the following sentence that describes someone's style of speaking in a different way, keeping the meaning of the original description. Original Description: [subject to be rephrased]."* Then, we select one sentence from the five rephrased sentences on the basis of their BERTScore [20] values: i.e., we pick on the sentence with the highest score if the score is higher than 0.80. For instance, the description *"His sound height is normal, but the speed is very fast, and the volume is very low."* can be rephrased as *"Despite his normal height, his sound is incredibly fast and surprisingly quiet."*

3. EXPERIMENTS

3.1. Dataset

We used PromptSpeech[3], which includes natural language instructions (style prompt) of various speaking styles. This dataset was constructed for PromptTTS [21] that can synthesize speech in accordance with a style prompt. We used the training subset of PromptSpeech real version, which includes 26,588 human-annotated style prompts of multi-speakers' speech samples from LibriTTS [22]. We divided the 26,588 prompts into training (24,953 by 1,113 speakers), development (857 by 40), and test (778 by 38) subsets and paired them with the corresponding speech samples from LibriTTS. Although PromptSpeech also includes categorical style factors that indicate class labels regarding gender, pitch, speed, and volume, we did not use them to train the captioning models.

[2] https://huggingface.co/meta-llama/Llama-2-7b-chat

[3] https://speechresearch.github.io/promptts

Table 1. Automatic speaking-style captioning results. AM is the aggregation module. B@4, R, M, BS, C, and S denote BLEU@4, ROUGE-L, METEOR, BERTScore, CIDEr-D, and SPICE scores, respectively. distinct-1/-2 of reference captions are 0.020 and 0.071, respectively.

(a) Results without sentence rephrasing augmentation.

Model	Speech encoder	B@4↑	R↑	M↑	BS↑	C↑	S↑	distinct-1↑	distinct-2↑
Transformer-based encoder-decoder	Mel-spectrogram	0.163	0.352	0.320	0.817	2.171	0.273	0.019	0.049
	x-vector	0.096	0.269	0.248	0.799	1.289	0.209	0.015	0.039
	WavLM	0.253	0.475	0.456	0.850	3.239	0.419	0.022	0.064
StyleCap w/ GPT-2 (proposed)	Mel-spectrogram + AM	0.178	0.381	0.357	0.827	2.295	0.316	0.020	0.057
	x-vector	0.085	0.273	0.255	0.800	1.138	0.214	0.013	0.032
	WavLM + AM	0.228	0.433	0.410	0.839	2.868	0.370	0.022	0.064
StyleCap w/ Llama 2 (proposed)	Mel-spectrogram + AM	0.160	0.358	0.332	0.821	2.109	0.295	0.022	0.066
	x-vector	0.076	0.262	0.239	0.799	1.107	0.213	0.016	0.042
	WavLM + AM	**0.273**	**0.497**	**0.469**	**0.855**	**3.471**	**0.434**	**0.023**	**0.073**

(b) Results with sentence rephrasing augmentation.

Model	Speech encoder	B@4↑	R↑	M↑	BS↑	C↑	S↑	distinct-1↑	distinct-2↑
Transformer-based encoder-decoder	Mel-spectrogram	0.140	0.332	0.303	0.814	1.847	0.239	0.018	0.046
	x-vector	0.071	0.244	0.212	0.792	1.046	0.191	0.012	0.027
	WavLM	0.246	0.464	0.441	0.848	3.172	0.404	0.021	0.059
StyleCap w/ GPT-2 (proposed)	Mel-spectrogram + AM	0.164	0.368	0.334	0.822	2.122	0.294	0.021	0.063
	x-vector	0.068	0.260	0.237	0.798	0.895	0.210	0.013	0.033
	WavLM + AM	0.239	0.470	0.439	0.848	3.056	0.403	0.022	0.068
StyleCap w/ Llama 2 (proposed)	Mel-spectrogram + AM	0.165	0.353	0.327	0.818	2.131	0.279	0.024	0.065
	x-vector	0.084	0.259	0.237	0.796	1.157	0.212	0.014	0.034
	WavLM + AM	**0.279**	**0.507**	**0.479**	**0.857**	**3.594**	**0.447**	**0.027**	**0.079**

3.2. Experimental conditions

As for the speech encoder, we used three types of speech feature representations: mel-spectrogram, speaker embeddings for speaker verification, and hidden vectors of a speech SSL model for the performance comparison. We extracted 80-dimensional mel-spectrograms from speech waveforms with 10 ms frame shift. We also used WavLM BASE+[4] [23] as the speech SSL model and extracted weighted-sum representation for each frame. To obtain a fixed-length vector representation for the mel-spectrogram and speech SSL model, a stack of 4-layer BLSTMs and MHA with 8 heads was used as an aggregation module. As for the speaker embeddings, 512-dimensional x-vectors [24] obtained from the pre-trained WavLM BASE+ for speaker verification[5] was used. As for the text decoder, we used two types of pre-trained LLMs: GPT-2[6] [25] and Llama 2 (7B)[7] [19]. The former is the same setting as ClipCap [18], while the latter can be regarded as the richer one. The numbers of model parameters for GPT-2 and Llama 2 were 125M and 7B, respectively. The dimensions of word embeddings for each model were 768 and 4,096, respectively. The mapping network consisted of 8-layer Transformer-encoders. The prefix length and dropout ratio were set to 40 and 0.2, respectively. These parameters were empirically determined. During the model training, we only trained the mapping network, the aggregation module in the speech encoder, and the weight coefficient for each layer in WavLM (i.e., w_l) used in the SSL feature aggregation process. The model parameters of LLMs and WavLM were frozen. All modules are trained in an end-to-end manner using the cross entropy loss between the generated captions and the ground-truth captions. Each model was trained without sentence rephrasing augmentation at 20 epochs or with the augmentation at 10 epochs because the augmentation doubles the data size. The batch size was set to 16.

We also trained a naive Transformer-based encoder-decoder model as the baseline system. The input features were extracted by the almost same speech encoder as those explained in the previous paragraph, but the aggregation module was replaced by an attention layer to align the lengths of speech features and tokens in a generated caption. The encoder had 12 encoder blocks. The decoder had a token embedding layer followed by 6 decoder blocks. We adopted MHA with 4 heads of 256 dimensions for both encoder and decoder. The implementation was based on Huggingface Transformers[8].

As evaluation metrics for captioning accuracy, we used BLEU [26], ROUGE-L [27], METEOR [28], BERTScore [20], CIDEr-D [29], and SPICE [30], between generated and annotated captions referring to other natural language generation tasks. We also used distinct-1/-2 [31] to evaluate the diversity of generated captions.

3.3. Experimental results

Table 1(a) shows experimental results without sentence rephrasing augmentation. First, a performance comparison by the speech encoder difference showed WavLM performed the best in this as well as other speech tasks. On the other hand, x-vectors did not work well because they were mainly trained to represent speaker characteristics rather than speaking styles such as speech rhythm [32]. In addition, the use of Llama 2 for the text decoder performed better than GPT-2 when employing WavLM for the speech encoder, which demonstrates that leveraging of richer LLM-based text decoders is a crucial factor to improve the captioning performance of StyleCap. On the other hand, Llama 2 did not necessarily improve the performance when mel-spectrogram was used as the input feature. One reason for this would be overfitting to the training data. The word embedding dimension of Llama 2 was 4,096-dimensional embeddings, which greatly outnumbers that of GPT-2 (768). Therefore, the speech encoder with the mel-spectrogram input might suffer from the high dimensionality of Llama 2 word embeddings and

[4] https://huggingface.co/microsoft/wavlm-base-plus

[5] https://huggingface.co/microsoft/wavlm-base-plus-sv

[6] https://huggingface.co/gpt2

[7] https://huggingface.co/meta-llama/Llama-2-7b

[8] https://huggingface.co/docs/transformers/model_doc/speech_to_text

Table 2. The accuracy (%) of style factor classifications with embedding for each speech encoder. P, S, and V indicate Pitch, Speed, and Volume respectively.

Speech encoder	Gender	P	S	V	Avg.
Mel-spectrogram + AM	93.8	62.1	67.8	54.7	69.6
x-vector	94.0	40.5	44.0	49.4	57.0
WavLM + AM	91.0	61.0	85.2	69.9	76.8

Table 3. Evaluation scores with changing prefix length.

Prefix length	1	2	5	10	40	60
METEOR	0.419	0.437	0.468	0.464	**0.479**	0.459
BERTScore	0.839	0.845	0.853	0.853	**0.857**	0.852

result in overfitting. In contrast, WavLM prevented such overfitting thanks to pre-training using a massive amount of speech data. In summary, StyleCap employing WavLM and Llama 2 outperformed naive Transformer-based encoder-decoder models.

Table 1(b) shows experimental results with sentence rephrasing augmentation. Overall tendencies were similar to the case without the augmentation. We can see that sentence rephrasing augmentation can improve the captioning performance of StyleCap employing WavLM. In other words, sentence rephrasing augmentation can make StyleCap to generate more diverse and accurate speaking-style captions, as shown in the improved distinct-1/-2 and other captioning results. These results indicate that sentence rephrasing augmentation is effective to deal with the difficulty of speaking-style captioning, i.e., learning one-to-many mapping. In contrast, the performance of Transformer-based encoder-decoder models could not be improved.

3.4. Discussion

3.4.1. Analysis of StyleCap behavior

To further understand the methods, we performed analysis from two aspects: the property of fixed-length vectors extracted by the speech encoders and the captioning performance from the vectors.

We first performed style factor classification using the learned fixed-length vectors. As described in Sect. 3.1, PromptSpeech also includes class labels indicating four style factors: gender, pitch, speed, and volume. The numbers of class labels for gender and the others are two (male/female), and three (low/mid/high), respectively. We trained a simple linear layer to classify the four style factors from a fixed-length vector extracted by the speech encoder of each trained StyleCap. In this analysis, trained StyleCap employing Llama 2 for the text decoder and sentence rephrasing augmentation for the training were used. Table 2 lists the classification accuracy. We can see that all speech encoders can classify each style factor to some extent. This implies that StyleCap can acquire speaking-style-related information from speech with natural language descriptions only. In other words, such information can be obtained without class labels indicating style factors. Comparing the speech encoders, WavLM achieved the highest classification performance, while x-vector could not capture para-linguistic information except for gender. Similar tendency was also reported in the previous work [32].

We next analyzed the relationship between the captioning performance and the fixed-length vector extracted by the speech encoder employing WavLM. Fig. 3 shows the violin plot of METEOR aggregated by the style factor classification performance (i.e., the number of correct predictions from 1 to 4)[9] described in the previous paragraph. From the results, we can see that the captioning

[9]We did not observe the complete misclassification whose number of correct predictions was 0 in the experiments.

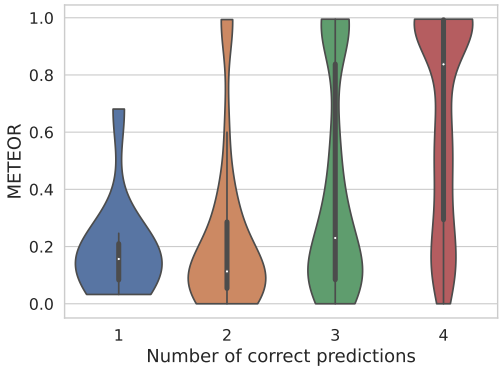

Fig. 3. Violin plot of METEOR by the performance of style factor classifications.

performance strongly depends on the style factor classification performance. Especially, the misclassification of even one factor can considerably degrade the captioning performance. Similar tendencies were also found in terms of other metrics such as BERTScore. These results indicate that capturing adequate para-/non-linguistic information by the speech encoder is an essential factor to further improve performance. Although WavLM and the simple aggregation module were used in this paper, another possible approach is to use CLAP [11], [15] trained from various para-/non-linguistic information.

3.4.2. Ablation study for mapping network

We finally performed an ablation study for the mapping network. To investigate the performance by changing the prefix length (K) of the mapping network, we set the prefix length to 1, 2, 5, 10, 40, and 60, respectively. In this experiment, StyleCap employing WavLM, Llama 2, and sentence rephrasing augmentation was used. Table 3 shows METEOR and BERTScore only by changing the prefix length due to the space limitation. As we can see, the performance is getting better with the longer prefix length. The shorter prefix lengths, especially 1 and 2, tends to worsen the performances because the number of trainable model parameters are limited. In addition, the too longer prefix also degrades the performance due to overfitting to training data. These tendencies were similar to ClipCap [18].

4. CONCLUSIONS

In this paper, we proposed StyleCap, a method to generate natural language descriptions of speaking styles appearing in speech, which we call the automatic speaking-style captioning task. As a first step to this end, we explored an appropriate large language model (LLM)-based text decoder and speech feature representation suitable for this task. The experimental results demonstrated that our StyleCap leveraging richer LLMs for the text decoder, speech self-supervised learning (SSL) features, and sentence rephrasing augmentation improved the accuracy and diversity of generated speaking-style captions. Although this paper focused on speaking styles appearing in speech, we believe that the proposed approach will be easily applicable for other para-/non-linguistic information such as emotional states and mental illnesses. Applying it to such information, which includes constructing pairs of speech and natural language descriptions, is for future work. We will also explore more suitable evaluation metrics for this task to growth the research area regarding the speaking style captioning. The evaluation metric using a larger LLM [33] including the prompt engineering for each task would be a possible approach.

Acknowledgements: This work was supported by JST, ACT-X Grant Number JPMJAX23CB, Japan.

References

[1] A. S. Cowen, H. A. Elfenbein, P. Laukka, *et al.*, "Mapping 24 emotions conveyed by brief human vocalization," *American Psychologist*, vol. 74, no. 6, pp. 698–712, 2019.

[2] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need," in *Proc. NIPS*, 2017.

[3] J. Shor, A. Jansen, W. Han, *et al.*, "Universal paralinguistic speech representations using self-supervised conformers," in *Proc. ICASSP*, 2022, pp. 3169–3173.

[4] M. B. Akçay and K. Oğuz, "Speech emotion recognition: Emotional models, databases, features, preprocessing methods, supporting modalities, and classifiers," *Speech Communication*, vol. 116, pp. 56–76, 2020.

[5] Z. Bai and X.-L. Zhang, "Speaker recognition based on deep learning: An overview," *Neural Networks*, vol. 140, pp. 65–99, 2021.

[6] D. M. Low, K. H. Bentley, and S. S. Ghosh, "Automated assessment of psychiatric disorders using speech: A systematic review," *Laryngoscope Investigative Otolaryngology*, vol. 5, no. 1, pp. 96–116, 2020.

[7] Y. Guo, C. Du, X. Chen, *et al.*, "EmoDiff: Intensity controllable emotional text-to-speech with soft-label guidance," in *Proc. ICASSP*, 2023.

[8] K. Zhou, B. Sisman, R. Rana, *et al.*, "Emotion intensity and its control for emotional voice conversion," *IEEE Transactions on Affective Computing*, vol. 14, no. 1, pp. 31–48, 2023.

[9] S. Bharati, M. R. H. Mondal, and P. Podder, "A review on explainable artificial intelligence for healthcare: Why, how, and when?" *IEEE Transactions on Artificial Intelligence*, pp. 1–15, 2023.

[10] H. Dhamyal, B. Elizalde, S. Deshmukh, *et al.*, "Describing emotions with acoustic property prompts for speech emotion recognition," *arXiv preprint arXiv:2211.07737*, 2022.

[11] Y. Pan, Y. Hu, Y. Yang, *et al.*, "GEmo-CLAP: Gender-attribute-enhanced contrastive language-audio pretraining for speech emotion recognition," *arXiv preprint arXiv: 2306.07848*, 2023.

[12] X. Mei, X. Liu, M. D. Plumbley, *et al.*, "Automated audio captioning: An overview of recent progress and new challenges," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2022, no. 1, pp. 1–18, 2022.

[13] L. Xu, Q. Tang, J. Lv, *et al.*, "Deep image captioning: A review of methods, trends and future challenges," *Neurocomputing*, vol. 546, p. 126 287, 2023.

[14] A. Radford, J. W. Kim, C. Hallacy, *et al.*, "Learning transferable visual models from natural language supervision," in *Proc. ICML*, 2021, pp. 8748–8763.

[15] E. Benjamin, D. Soham, M. A. Ismail, *et al.*, "CLAP: Learning audio concepts from natural language supervision," in *Proc. ICASSP*, 2023.

[16] T. Brown, B. Mann, N. Ryder, *et al.*, "Language models are few-shot learners," in *Proc. NeurIPS*, 2020.

[17] S. G. Koolagudi and K. S. Rao, "Emotion recognition from speech: A review," *International journal of speech technology*, vol. 15, pp. 99–117, 2012.

[18] R. Mokady, A. Hertz, and A. H. Bermano, "ClipCap: CLIP prefix for image captioning," *arXiv preprint arXiv: 2111.09734*, 2021.

[19] H. Touvron, L. Martin, K. Stone, *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.

[20] T. Zhang, V. Kishore, F. Wu, *et al.*, "BERTScore: Evaluating text generation with BERT," in *Proc. ICLR*, 2020.

[21] Z. Guo, Y. Leng, Y. Wu, *et al.*, "PromptTTS: Controllable text-to-speech with text descriptions," in *Proc. ICASSP*, 2023.

[22] H. Zen, V. Dang, R. Clark, *et al.*, "LibriTTS: A corpus derived from LibriSpeech for text-to-speech," in *Proc. INTERSPEECH*, 2019, pp. 1526–1530.

[23] S. Chen, C. Wang, Z. Chen, *et al.*, "WavLM: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.

[24] D. Snyder, D. Garcia-Romero, G. Sell, *et al.*, "X-Vectors: Robust dnn embeddings for speaker recognition," in *Proc. ICASSP*, 2018, pp. 5329–5333.

[25] A. Radford, J. Wu, R. Child, *et al.*, "Language models are unsupervised multitask learners," 2019.

[26] K. Papineni, S. Roukos, T. Ward, *et al.*, "BLEU: A method for automatic evaluation of machine translation," in *Proc. ACL*, 2002, pp. 311–318.

[27] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Proc. Workshop on Text Summarization Branches Out*, 2004, pp. 74–81.

[28] S. Banerjee and A. Lavie, "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments," in *Proc. ACL*, 2005, pp. 65–72.

[29] R. Vedantam, C. L. Zitnick, and D. Parikh, "CIDEr: Consensus-based image description evaluation," in *Proc. CVPR*, 2015, pp. 4566–4575.

[30] P. Anderson, B. Fernando, M. Johnson, *et al.*, "SPICE: Semantic propositional image caption evaluation," in *Proc. ECCV*, 2016, pp. 382–398.

[31] J. Li, M. Galley, C. Brockett, *et al.*, "A diversity promoting objective function for neural conversation models," in *Proc. ACL*, 2016, pp. 110–119.

[32] K. Fujita, T. Ashihara, H. Kanagawa, *et al.*, "Zero-shot text-to-speech synthesis conditioned using self-supervised speech representation model," in *Proc. ICASSP SASB Workshop*, 2023.

[33] Y. Liu, D. Iter, Y. Xu, *et al.*, "G-Eval: NLG evaluation using GPT-4 with better human alignment," *arXiv preprint arXiv:2303.16634*, 2023.