EmoSLLM: Parameter-Efficient Adaptation of LLMs for Speech Emotion Recognition

Hugo Thimonier[1,*] **Antony Perzo**[1] **Renaud Seguier**[1, 2]
[1] Emobot, Paris, France
{name}.{surname}@emobot.fr
[2] CentraleSupélec, IETR (UMR CNRS 6164), Rennes, France
[*] Corresponding author.

Abstract

Emotion recognition from speech is a challenging task that requires capturing both linguistic and paralinguistic cues, with critical applications in human-computer interaction and mental health monitoring. Recent works have highlighted the ability of Large Language Models (LLMs) to perform tasks outside of the sole natural language area. In particular, recent approaches have investigated coupling LLMs with other data modalities by using pre-trained backbones and different fusion mechanisms. This work proposes a novel approach that fine-tunes an LLM with audio and text representations for emotion prediction. Our method first extracts audio features using an audio feature extractor, which are then mapped into the LLM's representation space via a learnable interfacing module. The LLM takes as input (1) the transformed audio features, (2) additional features in the form of natural language (e.g., the transcript), and (3) a textual prompt describing the emotion prediction task. To efficiently adapt the LLM to this multimodal task, we employ Low-Rank Adaptation (LoRA), enabling parameter-efficient fine-tuning. Experimental results on standard emotion recognition benchmarks demonstrate that our model outperforms all but one existing Speech-Text LLMs in the literature, while requiring less than half the parameters of competing approaches.

1 Introduction

Predicting the emotion conveyed in audio is a critical task with many healthcare applications. For instance, tracking a patient's emotional fluctuations throughout the day can offer psychiatrists valuable insights into conditions such as depression—a disorder characterized by persistent sadness, irritability, and apathy [1]. As a result, continuous and non-invasive emotion monitoring could significantly improve diagnostic accuracy and treatment personalization.

The widespread adoption of smartphones among both minors and adults [11, 25] has enabled scalable, real-time monitoring of behavioral and emotional health. Among the modalities accessible through smartphones, speech is particularly informative due to its rich linguistic and paralinguistic content [29, 10]. These cues have been linked to various mental health conditions, and numerous studies have explored speech emotion recognition (SER) as a proxy for psychological well-being [37, 12].

SER has been addressed lately by leveraging feature representations coming from models trained for different tasks [8, 20, 21, 2, 5, 3, 13]. For instance, [33] fine-tune HuBERT [13] and Wav2Vec2.0 [3] for the task of SER [32]. Other approaches consider the use of frozen self-supervised models as feature extractors to train a supervised classifier [28, 17] by solely adding a linear layer on top of the self-supervised model. While promising, these approaches often rely exclusively on speech-related information.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Multimodal Representation Learning for Healthcare.

Given the recent discoveries on the strong capacities of LLMs for multimodal tasks, research has been oriented towards leveraging LLMs for other modalities, including audio. In particular, different overlapping lines of works have been considered: LLMs that *speak*, LLMs that *listen*, and LLMs that can do both. Relevant to the present work is LLMs that *listen*, which describe LLMs that can take as input both natural language and audio features [34, 16, 18, 9, 30, 7, 27].

Current state-of-the-art LLM-based approaches for Speech Emotion Recognition, such as SIFT-LLM [27] and SALOMONN [30], demonstrate impressive performance but rely on models with over 7 billion parameters. This makes them impractical for privacy-sensitive, on-device deployment—an essential consideration when handling highly personal data like a user's emotional state over time.

In the present work, we propose a parameter-efficient approach LLM-based approach for speech emotion recognition. We build on [31] and use as a downsampling module QPMapper, as it is lightweight and has shown strong performance for visual and audio data inclusion in LLMs. We rely on WavLM [4] as the audio feature extractors and experiment using Llama3.2-3B-Instruct [24]. We train our model using a 3-step learning curriculum. In the first phase, we treat automatic speech recognition (ASR) as a proxy task to align the audio representations with the LLM embedding space. During this phase, the audio encoder and LLM are frozen, and only the QPMapper is updated. In the second phase, we continue training on the ASR task but enable fine-tuning of the LLM via Low-Rank Adaptation (LoRA) [15]. Finally, in the third phase, we introduce the SER task to specialize the model for emotion recognition, further fine-tuning the LLM with LoRA while continuing to update the weights of the downsampling module.

We compare our model, coined **Emo**tion **S**peech **L**arge **L**anguage **M**odel, to existing text-audio language models for the task of speech emotion recognition. EmoSLLM achieves competitive SER performance, outperforming all but one existing text-audio model while maintaining a substantially smaller parameter footprint. This demonstrates its potential for privacy-preserving, on-device emotion recognition. In addition, we carefully design prompts to guide the language model's reasoning over the audio representations, which we find to be essential for improving emotion recognition accuracy in low-resource settings.

2 Method

2.1 Architecture

Audio encoder To extract semantically useful features from an audio signal, we rely on a pretrained audio feature extractor. Let $\mathbf{x}$ denote an audio signal, and f_{AE} denote the audio feature extractor.

$$\mathbf{h}_{AE} = f_{AE}(\mathbf{x}; \theta_{AE}) \in \mathbb{R}^{n \times d_{AE}}, \tag{1}$$

where d_{AE} is the hidden dimension of the audio encoder, and n the sequence length of the model's output.

Downsampling module We adopt a Query Pooling Mapper (QPMapper) module, previously shown to perform well on image modalities [31]. In a nutshell, this module adds n_q learnable queries, $\mathbf{q} \in \mathbb{R}^{n_q \times d_{AE}}$, to the original sequence $\mathbf{h}_{AE}$. This concatenated sequence is then passed through a transformer encoder, and the output queries' representations are kept as the downsampled audio representation. Additionally, this downsampling module g serves to project the audio features into the dimensional space of the language model's representations. Thus,

$$\mathbf{h}_{ds} = g(\mathbf{h}_{AE}; \theta_{ds}) \in \mathbb{R}^{n_q \times d_{LLM}},$$

where n_q is a hyperparameter.

Large language model Let f_{LLM} denote the large language model that will serve for the causal generation. It inputs a concatenated sequence comprised of: (i) the output of the downsampling module, $\mathbf{h}_{ds} \in \mathbb{R}^{n_q \times d_{LLM}}$, (ii) an embedded vectorized text prompt describing the task $\mathbf{p} \in \mathbb{R}^{n_p \times d_{LLM}}$ and (iii) possibly some textual information extracted from the audio signal also embedded and vectorized, e.g. a transcript, $\mathbf{z} \in \mathbb{R}^{n_z \times d_{LLM}}$. Thus, the probability distribution over the output from EmoSLLM can be expressed as:

$$\text{EmoSLLM}(\mathbf{x}, \mathbf{p}, \mathbf{z}) = f_{LLM}\left([\mathbf{h}_{ds}, \mathbf{p}, \mathbf{z}]; \theta_{LLM}\right). \tag{2}$$

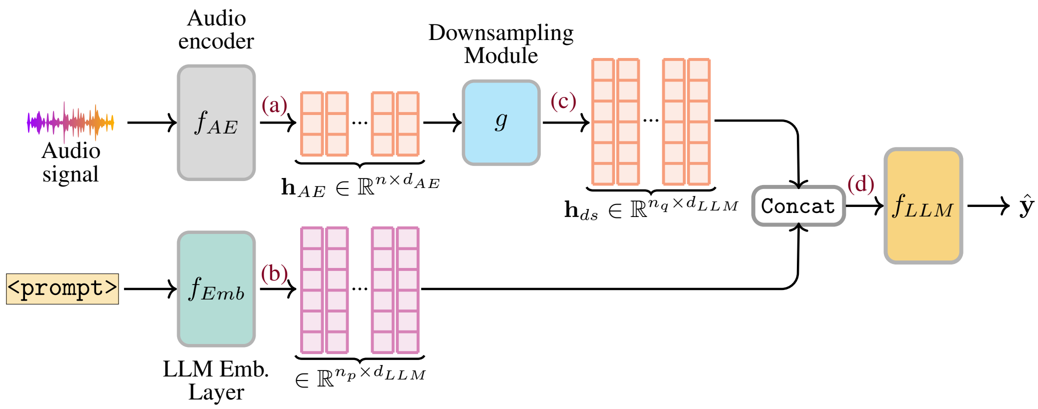

Figure 1: **EmoSLLM Pipeline**. In step (a), the audio signal is fed to a pretrained audio encoder to obtain a vectorized embedded representation $\mathbf{h}_{AE}$ of dimension $\mathbb{R}^{n \times d_{AE}}$. In step (b) a text prompt is fed to the embedding module of an LLM to output a vectorized embedded representation $\mathbf{p}$ of dimension $\mathbb{R}^{n_p \times d_{LLM}}$. In step (c) $\mathbf{h}_{AE}$ is fed to a downsampling module and the obtained sequence $\mathbf{h}_{ds}$ is of dimension $\mathbb{R}^{n_q \times d_{LLM}}$. In step (d) $\mathbf{h}_{ds}$ and $\mathbf{p}$ are concatenated and fed to the LLM, f_{LLM} that predicts the target given a task (ASR or SER).

2.2 Optimization objective

For each task, we provide a set of 10 prompts, selected randomly for each sample in a batch. We display in sections B.2 and B.3 examples of each prompt for both SER and ASR. For each task $t \in [\text{ASR}, \text{SER}]$, we train the LLM for next-token prediction in an auto-regressive manner as in standard LLM training schedules. Formally, let $\mathbf{p}^t$ denote a prompt for task t uniformly sampled on $\mathcal{P}^t$ the set of prompts for task t, and $\mathcal{D}^t$ the set of datasets used for task t. Each sample can be represented as a tuple $(\mathbf{x}^t, \mathbf{p}^t, \mathbf{z}^t, \mathbf{y}^t)$, where $\mathbf{x}^t$ is the audio waveform, $\mathbf{p}^t$ the sampled prompt for the corresponding task, $\mathbf{z}^t$ some additional information relevant for task t and $\mathbf{y}^t$ the label to be predicted. The probability of predicting the label $\mathbf{y}^t$ is modeled as

$$p(\mathbf{y}^t \mid \mathbf{x}^t, \mathbf{p}^t, \mathbf{z}^t; \Theta) = \text{EmoSLLM}(\mathbf{x}^t, \mathbf{p}^t, \mathbf{z}^t), \tag{3}$$

where $\Theta = \{\theta_{AE}, \theta_{ds}, \theta_{LLM}\}$. The LLM can attend to all tokens in the concatenated sequence $[\mathbf{x}^t, \mathbf{p}^t, \mathbf{z}^t]$ and is trained to leverage the audio tokens to minimize the negative likelihood given the probability modeled in Eq. (3).

3 Experiments

3.1 Experimental settings

Dataset For ASR training, we rely on the Librispeech dataset [26] as well as MSP-Podcast [23] since the transcript is also provided. Regarding SER, we only rely on the MSP-Podcast dataset for training. We evaluate the ability of EmoSLLM on the SER task on the `test1` share of MSP-Podcast.

Training settings We use AdamW [22] as the optimizer with learning rate $5 \cdot 10^{-4}$ and weight decay 0.01. We also rely on linear scheduling with a warm-up on 10% of the phase's training steps. We use WavLM [4] as the audio feature extractor and Llama 3.2-3B-Instruct [24] as the foundation language model. The LoRA adapters are added to the attention and FFN layers of the LLM while contrary to [9] we do not add LoRA adapters to the audio encoder and keep it frozen. LoRA adapter's rank is set to 8 α equal to 16. For the downsampling module, implemented as a QPMapper, we use 32 learnable queries, 2 transformer layers with 8 attention heads each, and an embedding dimension of 768. The output of the downsampling module is then mapped to the dimension of the LLM using a learned linear layer. We set the effective batch size to 512 for all three phases.

Benchmark To ensure a rigorous evaluation, we benchmark our approach against existing Speech-Text LLMs that incorporate SER capabilities. For instance, we compare to SALMONN-7B [30],

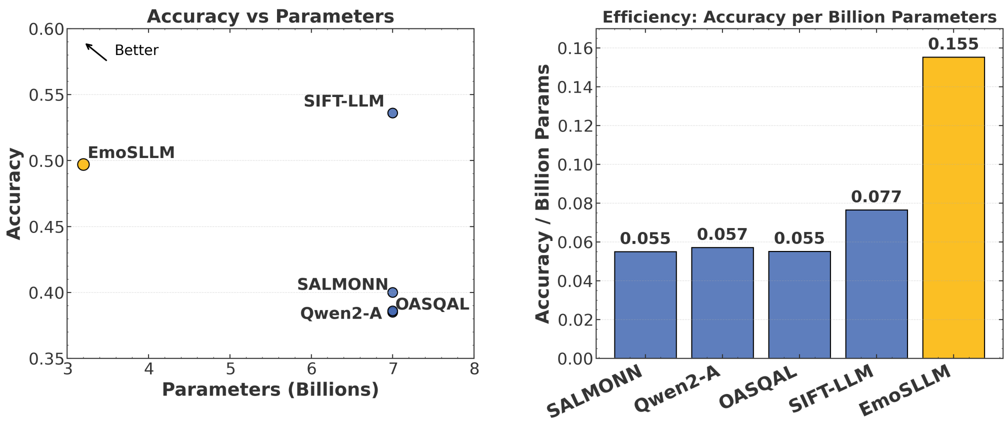

Figure 2: **Performance Comparison**. Performance comparison with existing Audio-Language Models that perform speech emotion recognition, Qwen2-Audio-Instruct [6] (Qwen2-A), OASQA-LLM [27] (OASQAL), and SIFT-LLM [27].

Qwen2-Audio-7B-Instruct [6], OASQA-LLM [27] and SIFT-LLM [27] using the test1 share of MSP-Podcast and the unweighted accuracy metric following previous work [6, 27, 30, 27].

3.2 Results

Figure 2 (left) illustrates the performance of EmoSLLM in relation to existing methods, plotted against their respective parameter counts. Our results demonstrate that EmoSLLM achieves notable performance despite its smaller number of parameters. EmoSLLM significantly outperforms SALMONN [30], Qwen2-Audio-Instruct [6], and OASQAL [27], while having less than half (3.2B) the number of parameters than competing methods (7B+). While SIFT-LLM exhibits superior performance in emotion prediction, this advantage comes with substantially higher computational requirements. We attribute the performance gap between EmoSLLM and SIFT-LLM to two primary factors. First, the backbone LLM used in their approach, Qwen2.5-7B-instruct [36], has a significantly larger number of parameters than ours. Second, SIFT-LLM's multi-task training regime exposes it to significantly more diverse data, enabling better cross-modal feature learning and more generalizable representations. This multi-task approach may create synergy where emotion recognition benefits from related speech understanding tasks. Despite these advantages of SIFT-LLM, it is noteworthy that EmoSLLM achieves competitive performance while maintaining a significantly smaller parameter footprint, suggesting greater computational efficiency as shown on Figure 2 (right). See Appendix A for ablation studies motivating the overall EmoSLLM approach.

4 Conclusion

This paper introduced EmoSLLM, a novel and computationally efficient approach for speech emotion recognition (SER) that effectively integrates audio and text modalities using LLMs. Our experimental results on standard SER benchmarks demonstrate that EmoSLLM outperforms most existing Speech-Text LLMs in the literature while requiring significantly fewer parameters and less training time. This highlights EmoSLLM's effectiveness and paves the way for more efficient and privacy-preserving applications in areas like human-computer interaction and mental health monitoring.

Limitations and future work While EmoSLLM shows strong performance and efficiency, it is still surpassed by SIFT-LLM. This is likely due to SIFT-LLM benefiting from a larger backbone LLM and exposure to a significantly greater volume of multi-task training data. This suggests that even with parameter efficiency, the scale of the base LLM and training data diversity remain crucial. Furthermore, achieving true on-device deployment for multimodal LLMs still requires substantially reducing the overall parameter count. Future work could explore the integration of smaller backbone LLMs, model compression techniques such as quantization, or extending EmoSLLM to handle a broader range of multimodal inputs.

Acknowledgement

This work was granted access to the HPC resources of IDRIS under the allocation 2025-AD010616049 made by GENCI.

References

[1] Ali J. Alsaad, Yusra Azhar, and Yasser Al Nasser. *Depression in Children*. StatPearls Publishing, Treasure Island (FL), 2023.

[2] Alexei Baevski, Steffen Schneider, and Michael Auli. vq-wav2vec: Self-supervised learning of discrete speech representations. In *International Conference on Learning Representations*, 2020.

[3] Alexei Baevski, Henry Zhou, Abdel rahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *ArXiv*, abs/2006.11477, 2020.

[4] Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, Jian Wu, Long Zhou, Shuo Ren, Yanmin Qian, Yao Qian, Jian Wu, Michael Zeng, and Furu Wei. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *CoRR*, abs/2110.13900, 2021.

[5] Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, Jian Wu, Long Zhou, Shuo Ren, Yanmin Qian, Yao Qian, Micheal Zeng, and Furu Wei. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16:1505–1518, 2021.

[6] Yunfei Chu, Jin Xu, Xiaohuan Zhou, Qian Yang, Shiliang Zhang, Zhijie Yan, Chang Zhou, and Jingren Zhou. Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models. *arXiv preprint arXiv:2311.07919*, 2023.

[7] Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen2-audio technical report, 2024.

[8] Yu-An Chung, Wei-Ning Hsu, Hao Tang, and James Glass. An unsupervised autoregressive model for speech representation learning. In *Interspeech*, 2019.

[9] Nilaksh Das, Saket Dingliwal, S. Ronanki, Rohit Paturi, David Huang, Prashant Mathur, Jie Yuan, Dhanush Bekal, Xing Niu, Sai Muralidhar Jayanthi, Xilai Li, Karel Mundnich, Monica Sunkara, Sundararajan Srinivasan, Kyu J Han, and Katrin Kirchhoff. Speechverse: A large-scale generalizable audio language model. *ArXiv*, abs/2405.08295, 2024.

[10] Hongwei Ding and Yang Zhang. Speech prosody in mental disorders. *Annual Review of Linguistics*, 9(Volume 9, 2023):335–355, 2023. ISSN 2333-9691.

[11] Linda Fischer-Grote, Oswald D Kothgassner, and Anna Felnhofer. Risk factors for problematic smartphone use in children and adolescents: a review of existing literature. *neuropsychiatrie*, 33(4):179, 2019.

[12] Lasse Hansen, Yan-Ping Zhang, Detlef Wolf, Konstantinos Sechidis, Nicolai Ladegaard, and Riccardo Fusaroli. A generalizable speech emotion recognition model reveals depression and remission. *Acta Psychiatrica Scandinavica*, 145(2):186–199, 2022.

[13] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 29:3451–3460, October 2021.

[14] Wei-Ning Hsu, Anuroop Sriram, Alexei Baevski, Tatiana Likhomanenko, Qiantong Xu, Vineel Pratap, Jacob Kahn, Ann Lee, Ronan Collobert, Gabriel Synnaeve, and Michael Auli. Robust wav2vec 2.0: Analyzing domain shift in self-supervised pre-training. In *Interspeech 2021*, pages 721–725, 2021.

[15] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[16] Shujie Hu, Long Zhou, Shujie Liu, Sanyuan Chen, Lingwei Meng, Hongkun Hao, Jing Pan, Xunying Liu, Jinyu Li, Sunit Sivasankaran, Linquan Liu, and Furu Wei. WavLLM: Towards robust and adaptive speech large language model. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 4552–4572, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

[17] Karim M. Ibrahim, Antony Perzo, and Simon Leglaive. Towards improving speech emotion recognition using synthetic data augmentation from emotion conversion. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 10636–10640, 2024.

[18] Zhifeng Kong, Arushi Goel, Rohan Badlani, Wei Ping, Rafael Valle, and Bryan Catanzaro. Audio flamingo: A novel audio language model with few-shot learning and dialogue abilities. In *Forty-first International Conference on Machine Learning*, 2024.

[19] Yuanchao Li, Yuan Gong, Chao-Han Huck Yang, Peter Bell, and Catherine Lai. Revise, reason, and recognize: Llm-based emotion recognition via emotion-specific prompts and asr error correction. In *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2025.

[20] Alexander H. Liu, Yu-An Chung, and James R. Glass. Non-autoregressive predictive coding for learning speech representations from local dependencies. In *Interspeech*, 2020.

[21] Andy T. Liu, Shang-Wen Li, and Hung yi Lee. Tera: Self-supervised learning of transformer encoder representation for speech. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:2351–2366, 2020.

[22] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

[23] Reza Lotfian and Carlos Busso. Building naturalistic emotionally balanced speech corpus by retrieving emotional speech from existing podcast recordings. *IEEE Transactions on Affective Computing*, 10(4):471–483, 2019.

[24] AI @Meta. The llama 3 herd of models, 2024.

[25] Jay A. Olson, Dasha A. Sandra, Élissa S. Colucci, Alain Al Bikaii, Denis Chmoulevitch, Johnny Nahas, Amir Raz, and Samuel P.L. Veissière. Smartphone addiction is increasing across the world: A meta-analysis of 24 countries. *Computers in Human Behavior*, 129:107138, 2022. ISSN 0747-5632.

[26] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: An asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210, 2015.

[27] Prabhat Pandey, Rupak Vignesh Swaminathan, K V Vijay Girish, Arunasish Sen, Jian Xie, Grant P. Strimel, and Andreas Schwarz. Sift-50m: A large-scale multilingual dataset for speech instruction fine-tuning, 2025.

[28] Leonardo Pepino, Pablo Riera, and Luciana Ferrer. Emotion Recognition from Speech Using wav2vec 2.0 Embeddings. In *Interspeech*, pages 3400–3404, 2021.

[29] Katarzyna Pisanski, Valentina Cartei, Carolyn McGettigan, Jordan Raine, and David Reby. Voice modulation: A window into the origins of human vocal control? *Trends in Cognitive Sciences*, 20(4):304–318, 2016. ISSN 1364-6613.

[30] Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun MA, and Chao Zhang. SALMONN: Towards generic hearing abilities for large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

[31] Théophane Vallaeys, Mustafa Shukor, Matthieu Cord, and Jakob Verbeek. Improved baselines for data-efficient perceptual augmentation of LLMs, 2024.

[32] Johannes Wagner, Andreas Triantafyllopoulos, Hagen Wierstorf, Maximilian Schmitt, Felix Burkhardt, Florian Eyben, and Björn W. Schuller. Dawn of the transformer era in speech emotion recognition: Closing the valence gap. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9):10745–10759, 2023.

[33] Yingzhi Wang, Abdelmoumene Boumadane, and Abdelwahab Heba. A fine-tuned wav2vec 2.0/hubert benchmark for speech emotion recognition, speaker verification and spoken language understanding. *CoRR*, abs/2111.02735, 2021.

[34] Jian Wu, Yashesh Gaur, Zhuo Chen, Long Zhou, Yimeng Zhu, Tianrui Wang, Jinyu Li, Shujie Liu, Bo Ren, Linquan Liu, and Yu Wu. On decoder-only architecture for speech-to-text and large language model integration, 2023.

[35] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[36] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025.

[37] Tsung-Hsien Yang, Chung-Hsien Wu, Kun-Yi Huang, and Ming-Hsiang Su. Detection of mood disorder using speech emotion profiles and lstm. In *2016 10th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, pages 1–5, 2016.

A Discussion

A.1 Large Language Model

To investigate the impact of both the LLM's architecture and training strategy, as well as parameter count, we compare the performance obtained by EmoLLM when using Qwen3-4B [35] and Llama 3.2-3B-Instruct [24] as f_{LLM}.

A.2 Training curriculum

We rely on a three-stage curriculum learning framework. First, we train the model only on the ASR task following previous work [16, 9]. In this first phase (P1), the audio feature extractor and LLM are frozen, while the downsampling module's weights are the only components updated. This phase aims to learn an effective mapping from the audio representation space to the LLM's embedding space, allowing the model to leverage the LLM's semantic capabilities [9, 27]. In the second phase (P2), we introduce Low-Rank Adaptation (LoRA) adapters to the LLM and continue training on the ASR task. Here, both the downsampling module and the LLM (via LoRA) are fine-tuned jointly, enabling the model to begin adapting to audio-conditioned language tasks. Finally, in the last phase (P3) we introduce the SER objective and train the model to perform both ASR and emotion recognition simultaneously.

A.3 Additional features

Previous work [19] has demonstrated that adding paralinguistic audio features may boost emotion recognition using text-only LLMs. [19] only rely on the audio transcript and curated prompts for emotion recognition, in which case including additional paralinguistic information in the audio logically boosts the performance for emotion recognition. In our case, some paralinguistic information is likely already contained in the audio tokens. Nevertheless, we investigate whether adding this information in the form of textual tokens in $\mathbf{z}$ might enhance our model's performance. We include the following paralinguistic features: loudness, average pitch, pitch range, jitter and shimmer. Also, following [19] we include in $\mathbf{z}$ the gender of the speaker. Rather than directly providing the value of the features, we provide the binned paralinguistic features in three classes `['low', 'medium', 'high']` that each represent a third of the values based on the training set. We include those tokens by sampling among 5 introductory sentences and randomizing the order in which the paralinguistic features are provided. We display in table 1, the performance of EmoLLM-base when trained with $\mathbf{z}$ including the additional features and with an empty $\mathbf{z}$. We observe that including paralinguistic features in natural language inputs improves the performance of emotion prediction, increasing accuracy from 45.8% to 46.9%, a gain of 1.1 percentage points.

Table 1: Performance comparison between performance with and without paralinguistic features in natural languages on test1 from MSP-Podcast [23].

Add. features	Accuracy (↑)
×	0.458
✓	**0.469**

Table 2: Performance comparison between n-shot hinting on test1 from MSP-Podcast [23].

Hint	Accuracy (↑)
0-shot	0.458
1-shot	0.473
2-shot	**0.474**

Few-shot format hinting We also investigate whether providing during phase P3, n examples of the expected output structure in the user prompt, might help enhance the performance of EmoLLM-base. See appendix B.6 for an example of such prompt strategy. We provide in table 2 the performance of EmoLLM-base trained with $\mathbf{z}$ including n examples in the user prompt for $n \in [0, 1, 2]$. We chose to keep n small as we expect the marginal gain to be quite small for higher values while increasing the computational cost. We observe a significant difference between EmoLLM-base without any hint (0-shot) and its performance when enhanced with the 1-shot and 2-shot hinting strategies as they display a respective gain of 1.5 and 1.6 percentage points over the 0-shot approach. Since 1-shot and 2-shot hinting provide similar performance, we chose to keep 1-shot hinting in our main approach as it involves a lower computational cost.

A.4 Joint prediction

Training During phase P3 training, when the emotion is available for a sample, we provide a prompt that asks for joint prediction. In other words, the model is asked to perform simultaneously the ASR and SER task. We believe that this could only be beneficial as it ensures that the model uses both semantic, linguistic and paralinguistic features to form its emotion prediction. See appendix B.4 for an illustrative example.

Inference The first approach, referred to as `SER-only`, involves prompting the LLM exclusively for the SER task without any auxiliary information. To assess the utility of providing transcript information as contextual hints, we explore two additional approaches. First, we consider providing the transcript in the user prompt, introduced by `"Use the following transcript to help you predict the emotion:"`, we refer to this approach as `Prompt-hint`. Note that this approach was never used during the training phase. Second, we consider providing the same user prompt as the ones seen during training, but we provide the beginning of the answer to the LLM and ask it to complete it. In other words, we ask the LLM to perform auto-regressive generation where its context contains the user prompt followed by the beginning of the assistant's answer, `"| ASR: <transcript> | Emotion:"`. We refer to this last approach as EmoLLM. See section B.7 for examples of such prompts. Comparison of the performance between these approaches is displayed in Table 3.

Table 3: **Prompt Strategies**. Performance comparison of different prompting strategies during inference on the test1 split of MSP-Podcast [23].

Prompt Strategy	Accuracy (↑)
`SER-only`	0.417
`Prompt-hint`	0.431
EmoLLM	**0.497**

Overall, we find that incorporating the transcript into the LLM's input significantly enhances SER accuracy. Both `Prompt-hint` and EmoLLM outperform the `SER-only` baseline. However, providing the transcript within the assistant's response, as done in EmoLLM, proves more effective than embedding it in the user prompt. Specifically, EmoLLM achieves an accuracy of 0.497, compared to 0.431 for `Prompt-hint`. The LLM's unfamiliarity with user prompts containing transcripts, since it was not exposed to such prompts during training, likely contributes to this performance gap.

A.5 Audio Encoder

We assess the impact of the choice of backbone audio encoder by replacing WavLM [4] with Robust wav2vec 2.0 [14]. The alternative model is trained using the same hyperparameters, training curriculum, and prompting strategy as the original configuration. Table 4 compares the performance of EmoLLM when using WavLM versus Robust wav2vec 2.0 as the pretrained audio encoder. While both encoders yield strong results, WavLM consistently outperforms Robust wav2vec 2.0 in this setup. However, it is important to note that the hyperparameters were optimized for WavLM and may not be ideal for wav2vec 2.0, potentially limiting the latter's performance.

Table 4: **Audio Encoder**. Performance comparison between pretrained audio encoders in EmoLLM on test1 from MSP-Podcast [23].

Audio Encoder	Accuracy (↑)
wav2vec 2.0	0.471
wavLM	**0.497**

B Prompts

We provide in this section a more comprehensive description of the prompt structures used to train EmoLLM.

B.1 System prompt

We carefully design a system that details to the LLM the task at hand, while providing useful information about the expected input and output structures. We provide hereafter a snippet of the curated system prompt.

System prompt

```
{
"role":
    "system",
"content":
    "You are a highly capable assistant specialized in audio processing tasks.
    You receive inputs containing audio token representations followed by text
    instructions, and return structured answer.

    You may be asked to perform:
      1. **Automatic Speech Recognition (ASR)** — transcribe the spoken content.
      2. **Speech Emotion Recognition (SER)** — identify the emotion expressed
    in the audio.

    Follow one of the two output formats:
    - For ASR-only tasks:
          '| ASR: <transcription> |'
    - For SER-only tasks:
          '| Emotion: <emotion code> |'
    For tasks involving both ASR and SER, use the following format:
          '| ASR: <transcription> | Emotion: <emotion code> |'

    Emotion must be provided as a single letter chosen from the following emotion
    codes:
      - A: Angry
      - S: Sad
      - H: Happy
      - U: Surprise
      - F: Fear
      - D: Disgust
      - C: Contempt
      - N: Neutral
      - O: Other
    (...)"
}
```

B.2 Automatic Speech Recognition (ASR) prompt

As previously discussed in the main section of the paper, for each sample we select a prompt among a curated selection of prompts detailing the expected task at hand. We provide hereafter an example an ASR prompt used during training.

ASR prompt

```
{
"role":
    "user",
"content":
    "You will now perform the following audio-based task.
    Task: **Automatic Speech Recognition (ASR)**.
    Transcribe the preceding audio into written text."
}
```

B.3 Speech Emotion Recognition (SER) prompt

We provide hereafter an example of a vanilla SER prompt used during training.

SER prompt

```
{
"role":
    "user",
"content":
    "You will now perform the following audio-based task.
    Task: **Speech Emotion Recognition**.
    Classify the tone of the speaker in the preceding audio."
}
```

B.4 Joint decoding prompt

We provide hereafter an example of a vanilla joint decoding prompt used during training.

Joint decoding prompt

```
{
"role":
    "user",
"content":
    "Perform the following audio-based tasks in the order as described.
      1. Task: **Automatic Speech Recognition (ASR)**.
      Identify and write down the words spoken in the preceding audio.
      2. Task: **Speech Emotion Recognition**.
      Analyze the audio and determine the emotional state of the speaker."
}
```

B.5 Supplementary features hinting

We provide a variety of supplementary features to guide the LLM in its prediction. We include features ranging from the gender of the person speaking, to some paralinguistic features. Supplementary feature can be combined or taken in isolation. We hereafter provide an example of how the supplementary features are included in the prompts as shown in sections B.2, B.3 and B.4.

Supplementary feature prompt

```
{
"role":
    "user",
"content":
    " (...)
    The speaker in this audio is $gender.
    Here's a breakdown of paralinguistic cues in the audio:
      - loudness: 'low'
      - pitch: 'medium'
      - pitch range: 'high'
      - jitter: 'low'
      - shimmer: 'high'
      (...)"
}
```

B.6 Example hinting

We also include the possibility of including n examples of answers in the format we are expecting. We include hereafter an example for example hinting in the case of joint decoding. Note that we also include this for SER-only and ASR-only prompts.

Few shot example prompt

```
{
"role":
    "user",
"content":
    "(...)
    Here are some examples of the expected output:
      - '| ASR: You're such a disappointment. | Emotion: C |'
      - '| ASR: I'm speechless... didn't expect that. | Emotion: U |',
      - '| ASR: I can't shake this feeling of dread. | Emotion: F |'"
    (...)"
}
```

B.7 Transcript hinting

As mentioned in section A.4, in inference we consider two alternatives to include the transcript as a hint to guide the LLM to make its prediction on the emotion class.

Prompt-hint First, we consider including the transcript in the user prompt with an introductory sentence. This approach is never used in training.

`Prompt-hint` prompt

```
{
"role":
    "user",
"content":
    "(...)
    Use the following transcript to help you classify the emotion:
    <transcript>
    (...)"
}
```

EmoLLM For our main pipeline, we consider an alternative that most resembles the task that the LLM performs in the training phase. We use as a user prompt the joint-decoding prompt as detailed in section B.4, and ask the LLM to auto-regressively generate tokens given the first part of the answer that contains the true transcript.

EmoSLLM prompt

```
{
"role":
    "user",
"content":
    <Joint decoding prompt>
},
{
"role":
    "assistant",
"content":
    "| ASR: <transcript> | Emotion:"
}
```

C Compute

We display in table 5 the compute hours required to train the different models to which we compare our method. The displayed hours are directly extracted from the original papers.

Table 5: Comparison of the total training hours and training hours specific to SER.

Model	Total training hours	SER training hours
SALMONN [30]	$\sim$ 4400	5
Qwen2-Audio-7B-Instruct [6]	$\sim$ 146500	$\sim$ 1000
SIFT-LLM [27]	173483	237
EmoLLM (Ours)	$\sim$ 320	$\sim$ 180