2025 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW) | 979-8-3315-1931-5/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICASSPW65056.2025.11010998

Performance Evaluation of SLAM-ASR: The Good, the Bad, the Ugly, and the Way Forward

Shashi Kumar[♣,1,2], Iuliia Thorbecke[♣,1,3], Sergio Burdisso[♣,1], Esaú Villatoro-Tello[♣,1], Manjunath K E[4], Kadri Hacioğlu[4]
Pradeep Rangappa[1], Petr Motlicek[1,5], Aravind Ganapathiraju[4] and Andreas Stolcke[4]

[1]*Idiap Research Institute, Switzerland*; [2]*EPFL, Switzerland*; [3]*University of Zurich, Switzerland*;
[4]*Uniphore, U.S.A. & India*; [5]*Brno University of Technology, Czech Republic*

***Abstract*—Recent research has demonstrated that training a linear connector between speech foundation encoders and large language models (LLMs) enables this architecture to achieve strong ASR capabilities. Despite the impressive results, it remains unclear whether these simple approaches are robust enough across different scenarios and speech conditions, such as domain shifts and speech perturbations. In this paper, we address these questions by conducting various ablation experiments using a recent and widely adopted approach called SLAM-ASR. We present novel empirical findings that offer insights on how to effectively utilize the SLAM-ASR architecture across a wide range of settings. Our main findings indicate that SLAM-ASR exhibits poor performance in cross-domain evaluation settings. Additionally, speech perturbations on in-domain data, such as changes in speech rate or additive noise, can significantly degrade performance. Our findings offer critical insights for fine-tuning and configuring robust LLM-based ASR models, tailored to different data characteristics and computational resources.**

***Index Terms*—ASR, LLMs, embeddings, speech-to-text alignment, foundation models.**

I. Introduction

Enabling large language models (LLMs) to "comprehend" non-textual modalities has received substantial attention recently. For instance, in [1] the authors trained a projection layer to align the outputs of a visual encoder with an LLM. In the context of automatic speech recognition (ASR), some early methods utilize a cascaded approach, where speech is first transcribed using an automated ASR system, followed by processing the resulting text with an LLM to enhance the transcription accuracy [2]–[5], or extracting further knowledge from automatic transcripts for downstream tasks [6], [7]. However, cascaded approaches have several limitations, including error propagation and lack of valuable paralinguistic information conveyed by acoustics, such as prosody and speaker characteristics.

Recently, systems that integrate robust speech encoders with instruction-tuned LLMs through a connector/projector layer have been proposed as end-to-end ASR solutions [8]–[12], henceforth referred to as LLM-based ASR systems. Intuitively, the main task of the connector/projector is to learn how to transform acoustic embeddings from speech encoders into speech representations (tokens) that are meaningful within the LLM's embedding space. These representations are then combined with text instructions (*i.e.*, prompts) and fed into an LLM to generate various predictions, such as transcription, emotion classification, language identification, and named entity recognition. Three immediate advantages of such architectures are: (a) computational efficiency, as the entire system can be adapted to new tasks by adopting parameter-efficient approaches, and/or by training the projector layer; (b) efficient use of the vast corpora used in the pretraining of foundation models; and (c) enhanced generalization capability, as LLMs can leverage prompts for zero-shot or in-context learning to handle unseen tasks effectively [9].

♣ Corresponding authors: {*shashi.kumar, iuliia.nigmatulina, sergio.burdisso, esau.villatoro*}*@idiap.ch*

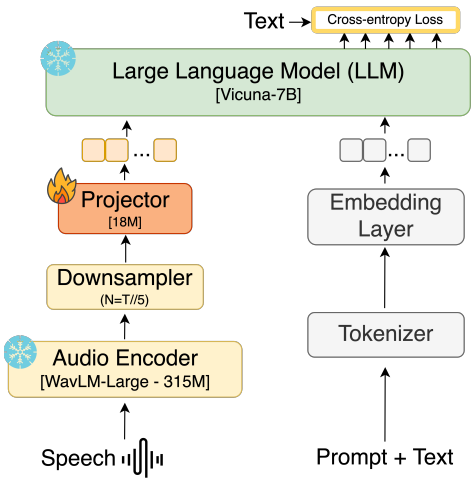


Fig. 1: SLAM-ASR pipeline. The selected models and the number of parameters for the performed experiments appear between brackets.

In [8], the authors proposed attaching an audio encoder (comprising 36 Conformer [13] layers) to an LLM (Llama-7B [14]) to perform the ASR task. Embeddings generated by the audio encoder are stacked and projected onto the LLM's input embedding space, which is further trained using the low-rank adaptation (LoRA) approach [15]. Similarly, in SpeechVerse [9] and SpeechLLM [16], the authors describe robust multitask training and curriculum learning frameworks that combine pretrained speech and text foundation models via a small set of learnable parameters. Contrary to [8], these approaches applied a 1-D convolution module that operates over the audio feature sequence to ensure compatibility with the LLM. In the end, only the convolutional downsampling module and the LoRA adapter are trained.

Recently, the SLAM-ASR architecture was introduced as "an embarrassingly simple approach for large language models with robust ASR capabilities" [12]. The authors argue that elaborate neural architecture designs are unnecessary, showing that a simple composition of off-the-shelf speech encoders and LLMs—using only a simple trainable projector to connect them—is sufficient (Figure 1).

Despite the impressive results reported for SLAM-ASR [12], it remains unclear whether this novel, yet simple solution, is "the way to go" for LLM-based ASR systems. This paper aims to address this question and performs three ablation experiments on the SLAM-ASR architecture to evaluate how strong its claimed ASR capabilities really are. We study and analyze the performance of SLAM-ASR on three well-known benchmark datasets and one private dataset, as well as its performance under extreme, yet plausible conditions normally addressed by traditional ASR models. This allows insights into the

Authorized licensed use limited to: Brno University of Technology. Downloaded on February 17,2026 at 15:35:12 UTC from IEEE Xplore. Restrictions apply.

structure and organization of the knowledge represented within the SLAM-ASR architecture, providing transparency and interpretability of the network's behavior.

Overall, our work makes three main contributions that we hope will support the methodological decisions of future researchers working on LLM-based ASR in the SLAM-ASR paradigm: *(i)* we show empirically that SLAM-ASR has a big dependence on the data used for training the projector (*i.e.*, overfitting), resulting in a model that lacks robustness when training and test data are mismatched; *(ii)* we show how sensitive the SLAM-ASR architecture is to temporal and noise perturbations, unlike its ASR counterpart, and *(iii)* we performed an exhaustive analysis, both quantitative and qualitative, of what type of alignments the projector layer is learning and, we show how such an alignment can be improved to give better ASR performance.

II. Methods

A. SLAM-ASR

The SLAM-ASR architecture consists of a speech encoder, followed by a fixed downsampler, then a trainable projector, with the final network being an instruction-tuned LLM accepting text embedding (prompt) and audio representations coming from the projector (Figure 1). One advantage of this implementation is its flexibility, making it possible to use different existing pretrained speech encoders, as well as a variety of LLMs. According to the original work [12] and confirmed by our experiments, the combination of *WavLM-large* [17] as a speech encoder and *Vicuna-7B* as an LLM achieves the best performance. Thus, in the experiments reported here, we always use the same *WavLM-large+Vicuna-7B* combination as the original work. The key element of the SLAM-ASR architecture is that the only trained element is the *projector* [1], a single hidden layer followed by a ReLU activation and a regression layer:

$$\mathbf{E}_i = Linear(ReLU(Linear(\mathbf{Z}_i))) \tag{1}$$

where $\mathbf{Z}_i$ is the ith downsampled audio feature, consisting of the concatenation of k consecutive frames (output by WavLM) in the feature dimension, and $\mathbf{E}_i$ has the same dimension as the LLM input embedding. We refer to $\mathbf{E}_i$ as the *speech token embedding* of the ith audio feature. Finally, an input audio consisting of the n downsampled audio features $\mathbf{Z}_0 \cdots \mathbf{Z}_n$ is given to the instruction-tuned LLM using the following prompt:[2]

$\mathbf{E}_0 \cdots \mathbf{E}_n$<s>USER: Transcribe speech to text.
ASSISTANT:{*transcript*}</s>

where {*transcript*} is the text corresponding to the audio, which is either provided during training or forced to be generated by the LLM at inference time.

B. Baseline model

For the sake of a fair comparison, we will use the *WavLM-large* [17] based model as our baseline ASR system since, as in the original work, is also used as the speech encoder in the SLAM-ASR model. More precisely, the baseline simply consists of adding a linear layer on top of the *WavLM-large*, trained using Connectionist Temporal Classification (CTC) [18] loss, as a typical End-to-End (E2E) ASR model. We refer to this model simply as "ASR".

[1]Although referred to as a 'linear projector' in the original work, it employs a non-linear activation function (ReLU) as shown in Eq. 1. In this work, we refer to it simply as *projector* to avoid confusion.

[2]In the original SLAM-ASR paper, speech embeddings are placed after "USER:", but in the released code, they are prepended to the input; we adopt the latter prompt format .

TABLE I: Cross-domain performance comparison between the baseline (ASR) and SLAM-ASR in terms of *word error rate* (WER). ∞ denotes WER values greater than 100.

Training	**Evaluation Datasets** (WER (↓))			
Data	*LibriSpeech*	*CallHome*	*ContactCenter*	*CommonVoice*
ASR				
LibriSpeech	**2.6**	38.5	31.8	26.2
CallHome	15.9	**25.9**	30.5	48.9
ContactCenter	17.1	37.2	**17.0**	44.1
SLAM-ASR				
LibriSpeech	**2.6**	∞	69.7	51.5
CallHome	61.8	**35.5**	44.5	∞
ContactCenter	60.4	∞	**13.8**	67.7

III. Experimental Setup

A. Datasets

We selected three well-known benchmark datasets—LibriSpeech, CallHome, and CommonVoice—due to their distinct characteristics and the unique challenges they present. Additionally, we experimented with a private dataset, ContactCenter, composed of multidomain contact center conversations.

LibriSpeech consists of read English speech derived from audiobooks with 1000 hours of speech sampled at 16 kHz [19]. We trained our models using the 960h training partition, while all the evaluations were performed in the LibriSpeech *test-clean* partition.

CallHome English (LDC97S42) contains spontaneous telephone conversations between multiple speakers, comprising 12.5h of transcribed training speech and 1.5h of test data. This dataset poses challenges due to its conversational nature, known to be difficult for ASR, with a large number of short segments.

CommonVoice comprises several thousand hours of crowdsourced audio in more than 100 languages [20], featuring significant variability in speakers, accents, speaking styles, and background noise, among others. We used the English test partition from CommonVoice-v11, containing 27h, but only as test set, to evaluate robustness.

ContactCenter contains 48h of training and 6h of test stereo-audio/transcript data from contact center conversations between agents and customers. We upsampled audio from 8 kHz to 16 kHz to align with the SLAM-ASR model's requirements.

B. Training setup and technical details

For the experiments on LibriSpeech, we use the publicly available checkpoint from SLAM-ASR[3], while we trained the projectors for CallHome and ContactCenter datasets in-house. To ensure that our results are reproducible and comparable to SLAM-ASR results, in all our experiments, we followed the setup from the original work [12]: the projector was trained for 3 epochs, with early stopping based on cross-entropy loss on the dev set, using AdamW [21] with a learning rate $\gamma = 10^{-4}$ and a batch size of 4. The speech encoder produces output at 50 Hz and the downsampling rate is set to $k = 5$, leading to the downsampled audio features $\mathbf{Z}_i$ having a rate of 10 speech token embeddings $\mathbf{E}_i$ per second. The projector hidden layer dimension is set to 2048 and beam search with a beam size of 4 is used.

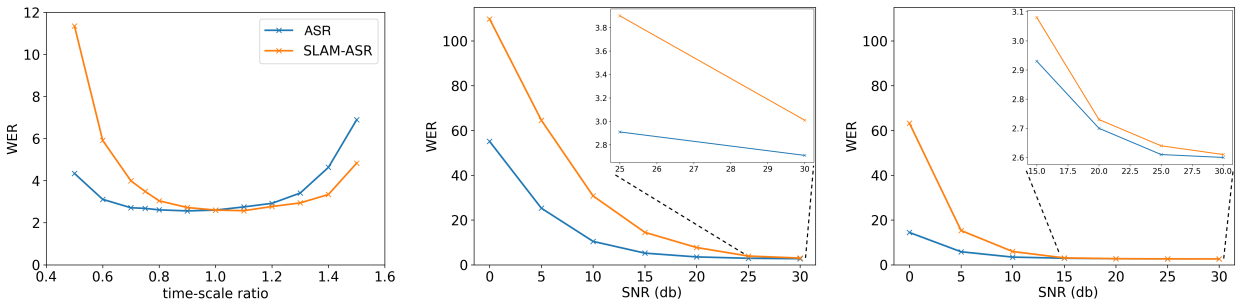
IV. Experiments and Results

A. Cross-domain robustness evaluation

Since the projector is the only trainable component in SLAM-ASR, we hypothesize it may be prone to overfitting the domain seen

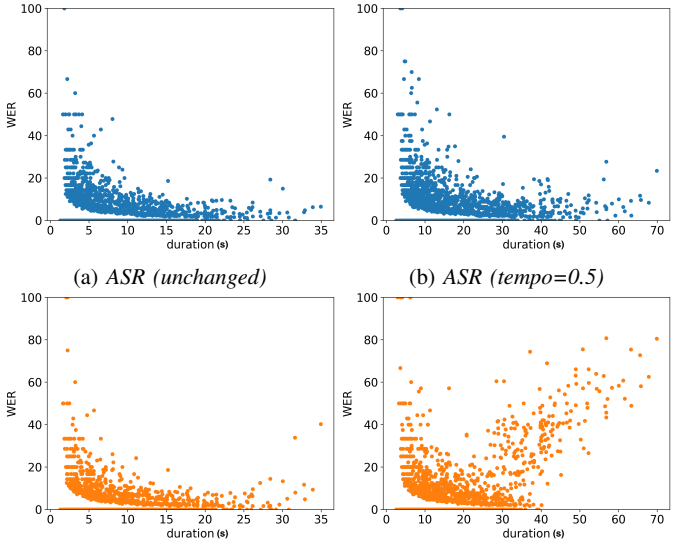
[3]https://github.com/X-LANCE/SLAM-LLM

Authorized licensed use limited to: Brno University of Technology. Downloaded on February 17,2026 at 15:35:12 UTC from IEEE Xplore. Restrictions apply.



(a) *Tempo* (b) *Babble noise* (c) *Music noise*

Fig. 2: Analysis of the impact of tempo (speech rate) and noise on WER.



(a) *ASR (unchanged)* (b) *ASR (tempo=0.5)*

(c) *SLAM-ASR (unchanged)* (d) *SLAM-ASR (tempo=0.5)*

Fig. 3: Scatter plots of WER versus speech duration for SLAM-ASR (bottom) and ASR baseline (top) on the LibriSpeech *test-clean* set: unchanged (left) and half-speed (right).

during training. This could limit its generalization to unseen domains or tasks when compared to standard ASR models, as highlighted in [22]. Thus, the goal of this experiment is to evaluate the robustness of SLAM-ASR in cross-domain scenarios. More precisely, we assess the performance degradation of SLAM-ASR when trained on one dataset, but evaluated on different, mismatched datasets (Section III-A). We compare this degradation to that of standard ASR under the same conditions.

From the results, shown in Table I, it is clear that SLAM-ASR shows much greater performance degradation than its ASR counterpart when decoding audio from datasets different from those used in training, consistently across all datasets. Additionally, SLAM-ASR exhibits exceptionally high WER (denoted by ∞) for certain dataset combinations. Upon inspection, we identified that this behavior was mainly due to a substantial increase in insertion errors, caused by LLM hallucinations spiraling out of control.

B. Speech perturbation ablations

In the previous subsection, SLAM-ASR demonstrated limited cross-dataset generalization capabilities. This suggests that the model's claimed strong speech recognition performance may be partially due to a reliance on dataset-specific acoustic and speaker characteristics, which it exploits as shortcuts to map audio features $\mathbf{Z}_i$ to speech token embeddings $\mathbf{E}_i$. To further investigate this hypothesis, we designed a series of ablation experiments incorporating various speech perturbation techniques applied to the original evaluation audio. Specifically, we evaluate how different levels of perturbations impact the performance of SLAM-ASR. We used the SLAM-ASR model trained on LibriSpeech, enabling a comparison of the results reported in the original SLAM-ASR paper [12] with those obtained in this section, when the test waveforms are altered.

1) Tempo perturbation: In SLAM-ASR, the projector must learn to map each audio feature $\mathbf{Z}_i$ to the LLM's input space, specifically the (sub)word embedding space. We hypothesize that the learned mapping could be influenced by the speaking rates present in the training data. Intuitively, the average speaking rate affects the average number of frames into which words are split and, consequently, the average number of audio feature $\mathbf{Z}$s to be mapped to the words' corresponding $\mathbf{E}$s. Therefore, we assess the robustness of SLAM-ASR to time-scale modifications of the test audio. In particular, the time-scale was modified by simply manipulating the hop length using the well-established PSOLA method [23], [24] which modifies the prosody of natural speech while retaining a high level of naturalness.

Figure 2a shows the change in WER performance across time-scale ratios ranging from 0.5 to 1.5, in increments of 0.1. As the speaking rate increases (*i.e.*, $ratio > 1$), both models exhibit a similar decline in performance. However, when the rate slows down (*i.e.*, $ratio < 1$), SLAM-ASR's WER rises sharply, reaching nearly 12% at a ratio of 0.5 (half the normal rate), in contrast to its ASR counterpart. These results suggest that SLAM-ASR struggles to handle an increase in the number of frames that must be mapped to the same number of (sub)words. This phenomenon may occur because a greater mismatch between the number of frames and the LLM's input wordpieces makes the mapping from $\mathbf{Z}$ to the corresponding $\mathbf{E}$ significantly more challenging. To further support this hypothesis, in Figure 3 we plot utterance-level WER values against speech duration. We can see that SLAM-ASR's WER starts to increase linearly when audio inputs are longer than $\approx$ 30 seconds (*i.e.* longer than 300 $\mathbf{Z}$s audio features), unlike ASR, which maintains a consistent WER pattern regardless of rate changes[4]

2) Noise augmentation: Given that the LLM and speech encoder are fixed in SLAM-ASR, the projector must be robust to relatively small changes in audio feature $\mathbf{Z}_i$ for accurate mapping to the LLM's embedding space. We hypothesize that the projector's tendency to find shortcuts for this mapping may significantly impact SLAM-ASR's performance, even with simple additive noise in the audio. Therefore, we add noise to the waveforms [25] to assess SLAM-

[4]Note that this behavior is unlikely due to reaching the maximum LLM context-length, 4K for Vicuna-7B, since the longer audio samples generate only around 700 speech tokens.

Authorized licensed use limited to: Brno University of Technology. Downloaded on February 17,2026 at 15:35:12 UTC from IEEE Xplore. Restrictions apply.

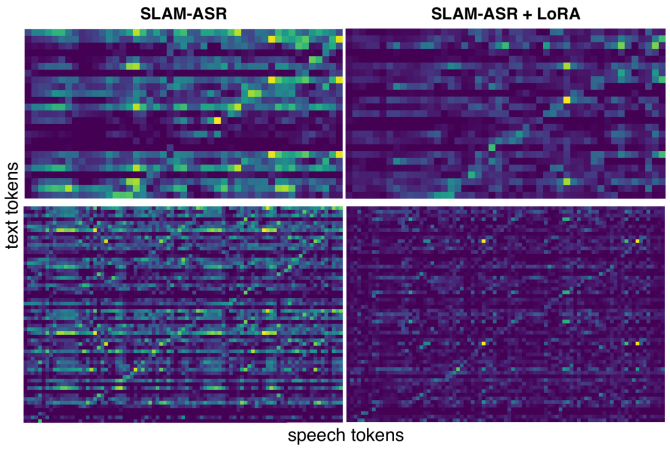


Fig. 4: The pairwise cosine similarity between every pair of speech and text token embeddings for two test examples before (left side) and after using LoRA (right side) in the LLM. Colors range from purple (-1, low similarity) through green (0, neutral) to yellow (+1, high similarity), using the viridis colormap.

	Reference text:	YES THAT WOULD BE THREE ONE SEVEN FIVE FIVE FIVE EIGHT THREE THREE
Speech tokens	**SLAM-ASR**	ū ụ ún uto uto uto uto uto \r \r ɲo ◌ヤ ◌uto 前 typeof ォ ☒ typeof ˙ ◌ ◌uto ɹ typeof ɾân uto uto ⍰ 田 lei wy ⍰ 分 안 ɹientes tw 分 前 laquelle Jahr 이 ◆ Jahr ɣ ɾosis 개 ォ ᵤ ᵤ lei ˊ人 \r \r
	SLAM-ASR + LoRA	hinlmmy Z **YessssS** S SGS **that Will** B **be** llS S Th **Three** l**one one**S S**77**anmu llllllF **Five** Fiveu O O F **Five Five** Fiveiivu lil□llP **Five** Fiveive 了**88**y P li Th **Three**y S Th **Three**ylee

Fig. 5: Ground truth transcript compared to learned speech tokens for SLAM-ASR and SLAM-ASR+LoRA for the same input audio.

ASR's robustness to noise. We use music and speech (babble) noise files from the MUSAN [26] corpus.

Figures 2b and 2c show the WER when babble and music noise, respectively, are added, resulting in a Signal-to-Noise Ratio (SNR) ranging from 0 to 30 decibels (dB) in 5 dB increments. As the noise increases (30dB $\rightarrow$ 0dB), the performance of SLAM-ASR degrades quite rapidly in contrast to its ASR counterpart. With babble noise in particular, even a small amount of noise (*e.g.*, 20 dB) causes substantial degradation in SLAM-ASR's performance, while the ASR model experiences only minimal degradation. These results suggest SLAM-ASR struggles with minor audio changes unseen in training.

C. Speech-to-text alignment analysis

Previous research has suggested that the LLM-based ASR task can be viewed as a regurgitation activity, where the language model is responsible for refining and reproducing information in the same order as it appears in the audio encoder's output sequence [8]. Thus, if the projector in SLAM-ASR can provide a sequence of embeddings monotonically aligned with the text embeddings, the LLM-based ASR problem reduces to a repetition task, which should not require the full capacity of an LLM. Nevertheless, a key aspect of the SLAM-ASR architecture is that both the speech encoder and the LLM are frozen, which contrasts with the current trend in LLM-based ASR research [8]–[12], where adapters (*e.g.*, LoRA [15]) are used to fine-tune the LLM. Hence, we hypothesize that SLAM-ASR will face more difficulties in learning the alignment of speech tokens and text tokens compared to methods that also fine-tune the LLM. To validate our hypothesis, we conducted an additional experiment on the ContactCenter dataset using the LoRA adapter to fine-tune the LLM alongside training the projector of SLAM-ASR. We then computed the cosine similarity between each possible pair of speech tokens and text tokens embedding of the LLM. Figure 4 shows the resultant alignment plots for two randomly selected samples. As can be inferred from the similarity plots, the task of aligning audio to text is harder for the original SLAM-ASR setup (left-side plots).

Additionally, to assess what the projector is learning, we mapped the learned speech tokens to their closest token from the LLM's vocabulary. One example from this exploration is shown in Figure 5. Note that the output from SLAM-ASR is mostly gibberish, while the tokens retrieved by SLAM-ASR+LoRA are much more aligned with the reference text. Overall, the results from this ablation raise the question whether freezing the LLM is advisable, given the clear advantage of using LoRA for better alignments and the improvement in performance for in-domain and cross-domain scenarios.[5]

V. Conclusions

We have investigated what is and what is not advisable when working with a recent, widely adopted LLM-based ASR solution, *i.e.*, SLAM-ASR. By ablation, we were able to identify the good, the bad, and the ugly aspects of this ASR paradigm in which both the speech foundation model and the LLM are frozen, and connected through a trainable projector.

The Good: SLAM-ASR is efficient, both in computation and data usage, while delivering competitive in-domain performance. Its flexibility allows easy integration of various speech encoders and LLMs, the only trainable component being a simple projector, a great advantage when computational resources or data are scarce.

The Bad: Like many other LLM-based ASR systems, SLAM-ASR tends to strongly overfit the training domain. This is a clear disadvantage compared to traditional ASR models, which have better generalization capabilities and are less sensitive to domain shifts and speech perturbation. Our experiments show that SLAM-ASR can easily go off the rails and hallucinate words (*e.g.*, repetitions or unrelated text) when simple perturbations are applied to the speech signal.

The Ugly: Unlike LLM-based ASR approaches that use LoRA for fine-tuning the LLM, the SLAM-ASR architecture lacks clear evidence that the projector is learning an alignment between speech and text, rather than some other spurious correlation. Our analysis shows that the alignment is less evident without LoRA, resulting in gibberish output when speech tokens are mapped to text tokens.

The Way Forward: Overall, our ablations suggest that SLAM-ASR should be trained and used with in-domain data for inference. However, if the data is very noisy, such as in CallHome, a traditional ASR model is the better choice. Similarly, our experimental analysis suggests that LoRA adapters should be considered for improving the alignment of speech and text tokens, as well as for better performance of SLAM-ASR in both in-domain and cross-domain scenarios. We recognize that many factors can influence the learning capabilities of LLM-based ASR models; however, we believe that our insights will benefit the research community working in similar areas.

Acknowledgments

This work was supported by the Idiap & Uniphore collaboration project. Part of the work was also supported by EU Horizon 2020 project ELOQUENCE[6] (grant number 101070558).

[5]SLAM-ASR+LoRA experiments using ContactCenter as training and evaluation data (in-domain setup) improved the WER from 13.8 to 11.6. For the cross-domain setup, *i.e.*, (*train*) ContactCenter $\rightarrow$ (*test*) LibriSpeech, WER improves from 60.4 to 22.5. We had to switch the default fp16 to fp32 precision for training with LoRA, as the default setting did not converge.

[6]https://eloquenceai.eu/

Authorized licensed use limited to: Brno University of Technology. Downloaded on February 17,2026 at 15:35:12 UTC from IEEE Xplore. Restrictions apply.

REFERENCES

[1] D. Zhu, J. Chen, X. Shen, X. Li, and M. Elhoseiny, "MiniGPT-4: Enhancing Vision-Language Understanding with Advanced Large Language Models," *arXiv preprint arXiv:2304.10592*, 2023.

[2] W. R. Huang, C. Allauzen, T. Chen, K. Gupta, K. Hu, J. Qin, Y. Zhang, Y. Wang, S.-Y. Chang, and T. N. Sainath, "Multilingual and Fully Non-Autoregressive ASR with Large Language Model Fusion: A Comprehensive Study," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 13 306–13 310.

[3] Y. Li, Y. Wu, J. Li, and S. Liu, "Prompting Large Language Models for Zero-Shot Domain Adaptation in Speech Recognition," in *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2023, pp. 1–8.

[4] R. Ma, M. Qian, P. Manakul, M. Gales, and K. Knill, "Can Generative Large Language Models Perform ASR Error Correction?" *arXiv preprint arXiv:2307.04172*, 2023.

[5] C.-H. H. Yang, Y. Gu, Y.-C. Liu, S. Ghosh, I. Bulyko, and A. Stolcke, "Generative Speech Recognition Error Correction With Large Language Models and Task-Activating Prompting," in *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2023, pp. 1–8.

[6] I. Nigmatulina, J. Zuluaga-Gomez, A. Prasad, S. Saeed Sarfjoo, and P. Motlicek, "A two-step approach to leverage contextual data: Speech recognition in air-traffic communications," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 6282–6286.

[7] J. Zuluaga-Gomez, S. S. Sarfjoo, A. Prasad, I. Nigmatulina, P. Motlicek, K. Ondrej, O. Ohneiser, and H. Helmke, "Bertraffic: Bert-based joint speaker role and speaker change detection for air traffic control communications," in *2022 IEEE Spoken Language Technology Workshop (SLT)*, 2023, pp. 633–640.

[8] Y. Fathullah, C. Wu, E. Lakomkin, J. Jia, Y. Shangguan, K. Li, J. Guo, W. Xiong, J. Mahadeokar, O. Kalinli *et al.*, "Prompting Large Language Models with Speech Recognition Abilities," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 13 351–13 355.

[9] N. Das, S. Dingliwal, S. Ronanki, R. Paturi, D. Huang, P. Mathur, J. Yuan, D. Bekal, X. Niu, S. M. Jayanthi *et al.*, "SpeechVerse: A Large-scale Generalizable Audio Language Model," *arXiv preprint arXiv:2405.08295*, 2024.

[10] C. Tang, W. Yu, G. Sun, X. Chen, T. Tan, W. Li, L. Lu, M. Zejun, and C. Zhang, "SALMONN: Towards Generic Hearing Abilities for Large Language Models," in *The Twelfth International Conference on Learning Representations*, 2024.

[11] J. Wu, Y. Gaur, Z. Chen, L. Zhou, Y. Zhu, T. Wang, J. Li, S. Liu, B. Ren, L. Liu *et al.*, "On decoder-only architecture for speech-to-text and large language model integration," in *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2023, pp. 1–8.

[12] Z. Ma, G. Yang, Y. Yang, Z. Gao, J. Wang, Z. Du, F. Yu, Q. Chen, S. Zheng, S. Zhang *et al.*, "An Embarrassingly Simple Approach for LLM with Strong ASR Capacity," *arXiv preprint arXiv:2402.08846*, 2024.

[13] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu *et al.*, "Conformer: Convolution-augmented Transformer for Speech Recognition," *arXiv preprint arXiv:2005.08100*, 2020.

[14] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "LLaMA: Open and Efficient Foundation Language Models," *arXiv preprint arXiv:2302.13971*, 2023.

[15] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "LoRA: Low-Rank Adaptation of Large Language Models," in *International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=nZeVKeeFYf9

[16] S. Rajaa and A. Tushar, "SpeechLLM: Multi-Modal LLM for Speech Understanding." [Online]. Available: https://github.com/skit-ai/SpeechLLM

[17] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao *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.

[18] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[19] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[20] R. Ardila, M. Branson, K. Davis, M. Kohler, J. Meyer, M. Henretty, R. Morais, L. Saunders, F. Tyers, and G. Weber, "Common Voice: A Massively-Multilingual Speech Corpus," in *Proceedings of the Twelfth Language Resources and Evaluation Conference*, 2020, pp. 4218–4222.

[21] I. Loshchilov and F. Hutter, "Decoupled Weight Decay Regularization," in *International Conference on Learning Representations*, 2019. [Online]. Available: https://openreview.net/forum?id=Bkg6RiCqY7

[22] M. Li, C.-T. Do, S. Keizer, Y. Farag, S. Stoyanchev, and R. Doddipatla, "WHISMA: A Speech-LLM to Perform Zero-shot Spoken Language Understanding," *arXiv preprint arXiv:2408.16423*, 2024.

[23] F. Charpentier and E. Moulines, "Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones," in *First European Conference on Speech Communication and Technology (Eurospeech 1989)*, 1989, pp. 2013–2019.

[24] Y. Jadoul, B. Thompson, and B. de Boer, "Introducing Parselmouth: A Python interface to Praat," *Journal of Phonetics*, vol. 71, pp. 1–15, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0095447017301389

[25] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2017, pp. 5220–5224.

[26] D. Snyder, G. Chen, and D. Povey, "MUSAN: A Music, Speech, and Noise Corpus," *arXiv preprint arXiv:1510.08484*, 2015.

Authorized licensed use limited to: Brno University of Technology. Downloaded on February 17,2026 at 15:35:12 UTC from IEEE Xplore. Restrictions apply.