

Budgeting Speech-to-LLM ASR for Underrepresented Dutch Speech

Xinnian Zhao and Pu Wang and Hugo Van Hamme

Department of Electrical Engineering (ESAT), KU Leuven, Leuven, Belgium
{xinnian.zhao, pu.wang, hugo.vanhamme}@kuleuven.be

Abstract

Recent Speech-to-LLM ASR work often concludes that a lightweight adapter is sufficient, but this evidence mostly comes from high-resource English benchmarks. We examine whether this conclusion transfers to two Dutch settings: Dutch–Flemish regional and speaking-style variation in CGN, and native speech from children and adolescents in JASMIN-CGN. Across both datasets, a simple linear projection remains competitive with heavier adapters while being markedly cheaper to train. On CGN, we further find that adapting a mismatched speech foundation encoder is more beneficial than replacing the LLM with a Dutch-adapted alternative. These results suggest a practical adaptation strategy for the evaluated Dutch settings: start with robust frozen backbones and a linear adapter, and adapt the speech encoder only when a clear speech-domain mismatch remains.

1 Introduction

Large language models (LLMs) (Achiam et al., 2023; Grattafiori et al., 2024) have recently attracted substantial interest for their strong language understanding and generation capabilities. Automatic speech recognition (ASR), as a speech-to-text task, naturally aligns with the text generation interface of LLMs. As a result, recent work has moved beyond using LLMs only for rescoring or post-processing, and has increasingly explored end-to-end pipelines in which LLMs directly consume speech-derived representations. A practical and widely adopted formulation is the **SFE + adapter + LLM** framework, where a speech foundation encoder (SFE) extracts continuous speech representations and a lightweight adapter maps them into the LLM embedding space (Chen et al., 2023; Shu et al., 2023; Fathullah et al., 2024; Bai et al., 2024).

A central design choice in this framework is the adapter. It must bridge two mismatched interfaces

while retaining linguistic and acoustic information from long speech sequences and compressing them into a form that fits the LLM context budget. Accordingly, many prior studies have focused on designing more expressive adapters to improve alignment and efficiency (Li et al., 2023b; Tang et al., 2024; Wu et al., 2023; Yu et al., 2024; Fan et al., 2025). At the same time, recent evidence suggests that such complexity may not always be necessary. Ma et al. (2024) show that even a simple linear projection can be competitive when paired with a strong instruction-tuned LLM, and Verdini et al. (2025) systematically compare different SFEs, adapters, and LLMs, concluding that the choice of SFE often dominates downstream performance. Taken together, these findings suggest an appealing recipe for efficient Speech-to-LLM ASR: keep the backbone models frozen and adapt the system with only a lightweight adapter.

However, this conclusion is drawn mostly from high-resource English benchmarks, where both the SFE and the LLM are relatively well matched to the evaluation domain. It is less clear whether it transfers to underrepresented speech, where mismatch may arise on both sides of the pipeline: the SFE may have limited coverage of the target acoustic and phonetic distribution, and the LLM may be less familiar with the target language or domain. However, this conclusion is drawn mostly from high-resource English benchmarks, where both the SFE and the LLM are often relatively well matched to the evaluation data. It is less clear how well it transfers to Dutch speech with regional and age-related variation, which is less commonly examined in Speech-to-LLM studies. We consider two such settings: Dutch–Flemish regional and speaking-style variation in CGN, and native speech from children and adolescents in JASMIN-CGN. These settings introduce different combinations of pronunciation, acoustic, lexical, and interaction-style variation.

These observations motivate our main question:

how should a limited adaptation budget be allocated for the Dutch settings considered here? We first examine whether a more complex adapter is needed on CGN and JASMIN-CGN. We then use CGN to compare whether adaptation capacity is better spent on adapting the SFE or on using a Dutch-adapted LLM.

To answer these questions, we revisit Speech-to-LLM ASR from the perspective of *adaptation budgeting*. We study two Dutch benchmarks that are rarely centered in prior Speech-to-LLM analyses: CGN for Dutch–Flemish regional and speaking-style variation, and JASMIN-CGN for speech from children and adolescents. Rather than proposing another adapter architecture, our goal is to derive practical guidance for deploying Speech-to-LLM ASR in underrepresented speech settings with limited annotation resources and compute budget. The main contributions of this paper are as follows:

- We show that for Dutch regional-accent and child speech, a simple linear projection achieves WER comparable to more complex adapters while being substantially more training-efficient.
- We show that, in this underrepresented setting, improving the SFE yields larger gains than replacing the LLM with a Dutch-adapted alternative.
- We provide an adaptation protocol that approaches full Whisper-Large fine-tuning at much lower training cost, offering a practical recipe for underrepresented Speech-to-LLM ASR.

2 Method

2.1 Speech-to-LLM ASR

We use the standard **SFE + adapter + LLM** architecture shown in Figure 1. Given speech input

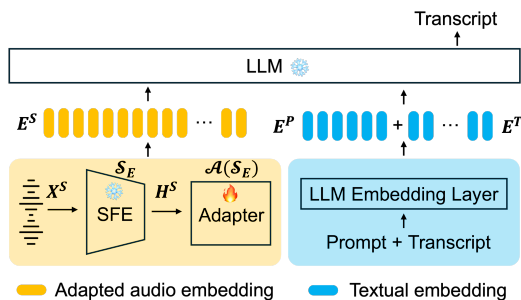


Figure 1: Speech-to-LLM ASR with an SFE, an adapter, and an LLM.

Table 1: Trainable adapter parameters (M) for each frozen SFE. Counts include all trainable components of the adapter and are rounded to the nearest 0.1M.

Adapter	mHuBERT	XLSR-53 /	
		Whisper-M	Whisper-L
LP	8.1M	18.9M	21.5M
CNN	26.8M	27.3M	27.8M
Q-Former	30.2M	30.4M	30.7M
Window-QF	30.2M	30.4M	30.7M

$\mathbf{X}^S$, the SFE $\mathcal{S}_E$ produces frame-level representations $\mathbf{H}^S = \mathcal{S}_E(\mathbf{X}^S)$. Because these sequences are too long for direct LLM consumption, an adapter $\mathcal{A}$ both down-samples and projects them into the LLM embedding space, yielding $\mathbf{E}^S = \mathcal{A}(\mathbf{H}^S)$. We then concatenate speech embeddings with a short text prompt and train the LLM to generate the transcript. During training, the loss is computed only on the target transcript tokens. This formulation allows the LLM to treat speech input as a soft prompt. Unless otherwise stated, we freeze the SFE and the LLM and update only the adapter.

2.2 Model Choices

We consider multilingual SFEs from two families: supervised Whisper-Medium and Whisper-Large (OpenAI, 2022; Radford et al., 2023), and self-supervised XLSR-53 and mHuBERT (FaceBook, 2020; Babu et al., 2022; UTTER, 2024; Boito et al., 2025). For fair comparison, all features are used at 50 Hz.

For the adapter, we compare four representative designs: linear projection (LP), a 1D CNN adapter, Q-Former (Li et al., 2023a), and a windowed Q-Former. LP and CNN use fixed 5:1 temporal down-sampling. For the global Q-Former, we use $Q = 80$ trainable queries following the Speech-to-LLM configuration of Yu et al. (2024). Window-QF applies the same query-based pooling within non-overlapping windows, with its window size and stride chosen to match the 5:1 compression ratio of LP and CNN. The number of trainable parameters depends on both the adapter architecture and the output dimensionality of the SFE. Table 1 reports the parameter counts for each SFE–adapter configuration. XLSR-53 and Whisper-Medium have the same output dimensionality and therefore use adapters with the same number of trainable parameters.

For the LLM, we compare LLaMA-3.1-8B-Instruct (Meta, 2024; Grattafiori et al., 2024) and the Dutch-adapted ChocoLlama-8B-

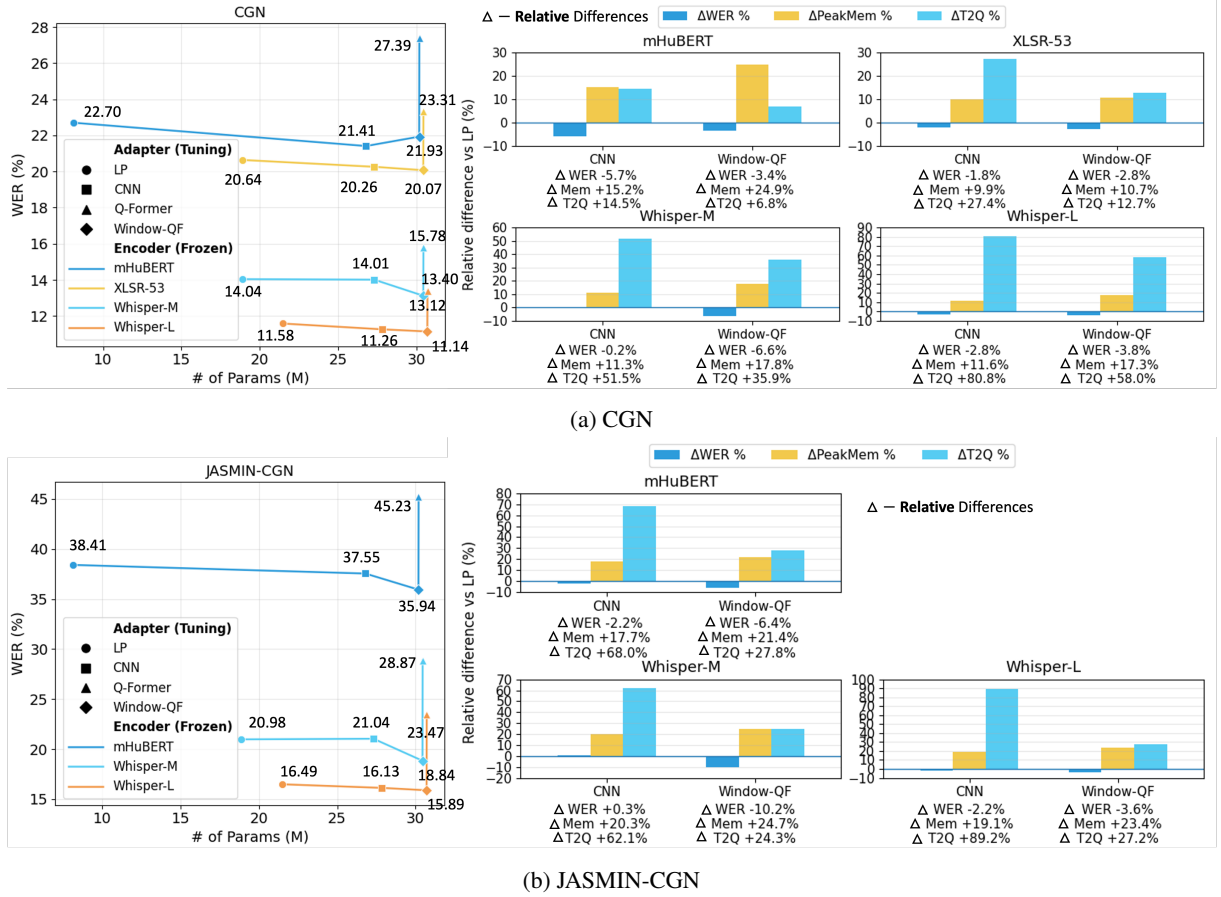


Figure 2: LP remains competitive on both CGN and JASMIN-CGN, while heavier adapters increase peak GPU memory and time-to-quality (T2Q; wall-clock training time from initialization to the checkpoint attaining the lowest validation WER) more than they improve WER.

Instruct (Meeus, 2024; Meeus et al., 2024). This lets us test whether language-specific LLM adaptation helps once the speech side is already strong.

3 Experimental Setup

3.1 Dataset

CGN (Oostdijk, 2000) covers Dutch and Flemish speech across multiple speaking styles. We use a 341-hour subset with a 2:1 Dutch–Flemish ratio, excluding spontaneous components c, d, and f, resulting in 341/35/70 hours for train/val/test. We treat this benchmark as a regional-accent setting: the language is shared, but pronunciation and speaking style vary across regions.

JASMIN-CGN (Cucchiari et al., 2006) extends CGN to non-standard Dutch speech. We use native Dutch speech from children aged 7–11 and adolescents aged 12–16, including both read and interactive recordings, and construct speaker-disjoint 80/10/10 train/validation/test splits. After timestamp-based segmentation, the final durations are 21.0/2.5/2.6 hours. Compared with CGN, this

setting introduces age-related variation in acoustics and pronunciation, together with potential differences in lexical choice and interaction style.

3.2 Implementation

We run two experiment groups. First, we freeze both the SFE and the LLM and train only the adapter, which isolates the effect of adapter capacity. Then, we compare spending adaptation budget on the SFE versus the LLM. For SFE adaptation, we fine-tune the encoder on CGN using a restricted target vocabulary derived from the LLM tokenizer (16.5k tokens out of 128.3k). For Whisper, we keep the original Whisper decoder during fine-tuning to preserve its pretrained linguistic modeling.

All models are trained on a single NVIDIA A100 (80 GB) with batch size 16, gradient accumulation 2, up to 40k steps, AdamW, peak learning rate 1×10^{-4} , 1k warmup steps, and early stopping on validation WER. We report WER, peak GPU memory, and time-to-quality (T2Q), defined as the wall-clock time from initialization to the checkpoint with the lowest validation WER.

4 Results

4.1 Adapter Capacity

In this section, we use ChocoLlama-8B-Instruct as the frozen LLM and tune adapters of varying capacity on top of frozen Whisper-Medium, Whisper-Large, mHuBERT, and XLSR-53 encoders on both CGN and JASMIN-CGN.

Figure 2 (left) compares LP, CNN, Q-Former, and Window-QF across encoder backbones. Overall, the simple LP achieves WER comparable to the stronger complex adapters. The absolute differences among the best-performing adapters are small, ranging from 0.44 to 1.29 WER points on CGN and from 0.60 to 2.47 on JASMIN-CGN. By contrast, the global Q-Former is consistently weaker. In our configuration, its fixed $Q = 80$ queries yield an average output rate of approximately 4 Hz on the segmented utterances, whereas LP, CNN, and Window-QF produce duration-dependent sequences at approximately 10 Hz. The stronger performance of Window-QF is consistent with the hypothesis that retaining finer temporal resolution is useful. However, output rate, fixed versus duration-dependent sequence length, and pooling scope are confounded in this comparison; our results therefore do not establish output rate as the sole cause of the gap.

We further compare peak GPU memory and T2Q. Since LP, CNN, and Window-QF are the strongest adapters, Figure 2 (right) focuses on their **relative** changes in WER, peak memory, and T2Q with respect to LP. The main pattern is clear: more complex adapters incur substantially higher memory and training cost, while their WER gains are modest and often inconsistent across backbones.

This pattern is consistent across the two evaluated datasets: increasing adapter complexity yields limited returns despite their different speech conditions. A simple linear adapter therefore offers the most favorable accuracy–efficiency trade-off in our experiments.

4.2 Speech Foundation Encoders vs LLMs

We next evaluate whether limited adaptation budget is better allocated to the SFE or to the LLM. Because ChocoLlama-8B-Instruct is trained primarily on Dutch adult-oriented text, we restrict this comparison to CGN.

We first keep ChocoLlama frozen and adapt the SFE together with a linear adapter. Figure 3 reports three settings: (i) zero-shot SFE performance, (ii)

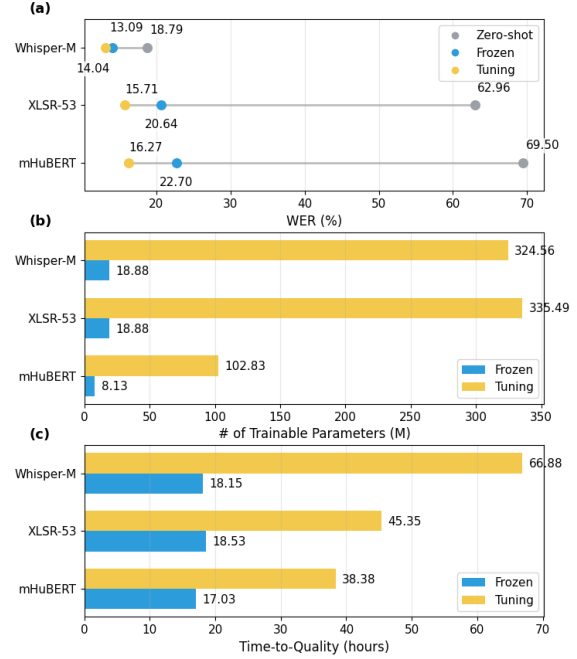


Figure 3: Effect of SFE adaptation on CGN. (a) WER (%) improvements from zero-shot to frozen to SFE-tuned settings. (b, c) corresponding training cost.

adapter-only tuning with a frozen SFE, and (iii) further fine-tuning of the SFE. For XLSR-53 and mHuBERT, zero-shot decoding is obtained by attaching a CTC head. We find that weaker and more mismatched encoders benefit substantially from SFE adaptation, with gains of up to 6.5 absolute WER points. In contrast, for Whisper, which is already relatively robust in the zero-shot setting (18.79 WER), additional SFE fine-tuning yields only a 0.95 absolute WER improvement while requiring about $4\times$ more training time. Thus, SFE adaptation is most worthwhile when the encoder is clearly out of domain.

We then fix Whisper-Large and compare the LLM using the same linear adapter with either LLaMA-3.1-8B-Instruct or ChocoLlama-8B-Instruct. The bottom two rows of Table 2 show that LLaMA-3.1-8B-Instruct achieves slightly lower WER than the Dutch-adapted LLM (11.33 vs. 11.58). One explanation is that LLaMA-3.1 provides sufficient Dutch subword coverage, while its broader instruction tuning better supports decoding from speech-derived embeddings. Thus, explicit Dutch adaptation of the LLM does not necessarily improve ASR performance in this setup. Overall, the CGN results indicate that adapting a clearly mismatched SFE is more beneficial than replacing the LLM with the Dutch-adapted alternative considered here.

Table 2: WER-training efficiency comparison on CGN.

Model	WER(%)	T2Q(h)	Params
mHuBERT (full fine-tuning with the CTC decoder)	24.63	21.35	94.7
XLSR-53 (full fine-tuning with the CTC decoder)	22.56	26.82	316.6
Whisper-Large (full fine-tuning Whisper encoder-decoder)	10.04	75.22	1550.0
Whisper-Large + LP (adapter-only tuning) + Llama3.1-8B-Instruct	11.33	18.58	18.9
Whisper-Large + LP (adapter-only tuning) + ChocoLlama-8B-Instruct	11.58	18.58	18.9

4.3 Comparison to SOTA Speech Foundation Models

We further compare this lightweight approach with full fine-tuning of speech foundation models. Table 2 summarizes WER together with T2Q and the number of trainable parameters.

Whereas the full-fine-tuning baselines adapt the entire speech model, our **SFE + LP + LLM** system updates only the LP adapter while keeping both the SFE and the LLM frozen. Despite this extreme parameter efficiency, the linear-adapter Speech-to-LLM system clearly outperforms fully fine-tuned mHuBERT and XLSR-53. It also remains close to full Whisper-Large fine-tuning (11.33 vs. 10.04 WER), while requiring more than $4\times$ less training time and about $82\times$ fewer trainable parameters. Overall, lightweight Speech-to-LLM adaptation provides a highly favorable accuracy-efficiency trade-off for underrepresented ASR.

5 Discussion and Limitations

Interpreting the Q-Former result. The global Q-Former follows prior Speech-to-LLM work and provides a compact, fixed-length interface to the LLM. On our segmented utterances, $Q = 80$ yields an average output rate of approximately 4 Hz, compared with approximately 10 Hz for LP, CNN, and Window-QF. Window-QF’s stronger performance is consistent with the benefit of preserving finer temporal detail. However, the variants also differ in fixed versus duration-dependent sequence length and global versus local pooling. The present comparison therefore cannot isolate temporal resolution as the sole cause; doing so would require varying Q under a more tightly rate-matched setup.

Dependence on backbone generation. Our conclusions are conditional on the evaluated backbones. The return from SFE adaptation decreases as the frozen SFE becomes stronger: mHuBERT and XLSR-53 gain up to 6.5 absolute WER points, whereas Whisper gains only 0.95 points at roughly four times the training time. As future SFEs become more robust to regional, age-related, and

recording variation, SFE fine-tuning may become less valuable and adapter-only training more attractive. Conversely, stronger lexical or text-domain shifts may increase the value of LLM adaptation. We therefore view the protocol as diagnostic rather than universal: adapt the SFE only when frozen performance reveals a clear speech-side gap.

Acoustic versus lexical mismatch in JASMIN-CGN. JASMIN-CGN is not a purely acoustic-shift benchmark. Its child/adolescent read and interactive speech may differ from adult Dutch in vocal characteristics and pronunciation, but also in lexical choice, disfluency, sentence structure, and interaction style. Because we report aggregate WER and do not vary the LLM on JASMIN-CGN, we cannot separate speech-side from language-side errors. The JASMIN-CGN results therefore support the adapter efficiency conclusion under a combined mismatch, whereas the SFE-versus-LLM comparison is restricted to CGN.

6 Conclusion

We studied adaptation budgeting for Speech-to-LLM ASR on CGN and JASMIN-CGN. Across both datasets, a linear adapter provided the most favorable accuracy–efficiency trade-off, with heavier adapters increasing training cost more than they improved WER. On CGN, adapting a clearly mismatched speech encoder was more effective than replacing the LLM with a Dutch-adapted alternative, whereas the robust Whisper encoder benefited little from further adaptation. These findings suggest a practical strategy for the evaluated Dutch settings: begin with frozen backbones and a linear adapter, and allocate additional capacity to the speech encoder only when its frozen performance indicates a clear mismatch.

7 Acknowledgment

This research was supported by the Research Foundation Flanders (FWO) under grant S004923N of the Strategic Basic Research (SBO) programme.

References

- Josh Achiam, Steven Adler, Sandhini Agarwal, and 1 others. 2023. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*.
- Arun Babu, Changan Wang, Andros Tjandra, and 1 others. 2022. XLS-R: Self-supervised cross-lingual speech representation learning at scale. In *Proceedings of Interspeech*, pages 2278–2282.
- Ye Bai, Jingping Chen, Jitong Chen, and 1 others. 2024. SEED-ASR: Understanding diverse speech and contexts with LLM-based speech recognition. *arXiv preprint arXiv:2407.04675*.
- Marcely Zanon Boito, Vivek Iyer, Nikolaos Lagos, and 1 others. 2025. mHuBERT-147: A compact multilingual HuBERT model. In *Proceedings of Interspeech*, pages 3939–3943.
- Feilong Chen, Minglun Han, Haozhi Zhao, and 1 others. 2023. X-LLM: Bootstrapping advanced large language models by treating multi-modalities as foreign languages. *arXiv preprint arXiv:2305.04160*.
- Catia Cucchiari, H Van Hamme, O van Herwijnen, and Felix Smits. 2006. Jasmin-cgn: Extension of the spoken dutch corpus with speech of elderly people, children and non-natives in the human-machine interaction modality.
- FaceBook. 2020. facebook/wav2vec2-large-xlsr-53. <https://huggingface.co/facebook/wav2vec2-large-xlsr-53>. Hugging Face model repository.
- Ruchao Fan, Bo Ren, Yuxuan Hu, and 1 others. 2025. AlignFormer: Modality matching can achieve better zero-shot instruction-following speech-LLM. *IEEE Journal of Selected Topics in Signal Processing*.
- Yassir Fathullah, Chunyang Wu, Egor Lakomkin, and 1 others. 2024. Prompting large language models with speech recognition abilities. In *ICASSP*, pages 13351–13355. IEEE.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. In *Neural Information Processing Systems*. Curran Associates.
- Junnan Li, Dongxu Li, Silvio Savarese, and 1 others. 2023a. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International Conference on Machine Learning (ICML)*, pages 19730–19742. PMLR.
- Yuang Li, Yu Wu, Jinyu Li, and Shujie Liu. 2023b. Prompting large language models for zero-shot domain adaptation in speech recognition. In *ASRU*, pages 1–8. IEEE.
- Ziyang Ma, Guanrou Yang, Yifan Yang, and 1 others. 2024. An embarrassingly simple approach for LLM with strong ASR capacity. *arXiv preprint arXiv:2402.08846*.
- Matthieu Meeus. 2024. Chocollama/llama-3-chocollama-8b-instruct. <https://huggingface.co/ChocoLlama/LLama-3-ChocoLlama-8B-instruct>. Hugging Face model repository.
- Matthieu Meeus, Anthony Rathé, François Remy, Pieter Delobelle, Jens-Joris Decorte, and Thomas Demeester. 2024. Chocollama: Lessons learned from teaching llamas dutch. *arXiv preprint arXiv:2412.07633*.
- Meta. 2024. meta-llama/llama-3.1-8b. <https://huggingface.co/meta-llama/LLama-3.1-8B>. Hugging Face model repository.
- Nelleke Oostdijk. 2000. The spoken dutch corpus: Overview and first evaluation. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, pages 887–894.
- OpenAI. 2022. openai/whisper-large-v3. <https://huggingface.co/openai/whisper-large-v3>. Hugging Face model repository.
- Alec Radford, Jong Wook Kim, Tao Xu, and 1 others. 2023. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning (ICML)*, pages 28492–28518. PMLR.
- Yu Shu, Siwei Dong, Guangyao Chen, and 1 others. 2023. LLASM: Large language and speech model. *arXiv preprint arXiv:2308.15930*.
- Changli Tang, Wenyi Yu, Guangzhi Sun, and 1 others. 2024. Salmonn: Towards generic hearing abilities for large language models. In *Proceedings of the International Conference on Learning Representations (ICLR)*.
- UTTER. 2024. utter-project/mhubert-147. <https://huggingface.co/utter-project/mHuBERT-147>. Hugging Face model repository.
- Francesco Verdini, Pierfrancesco Melucci, Stefano Perna, and 1 others. 2025. How to connect speech foundation models and large language models? what matters and what does not. In *Proceedings of Interspeech*, pages 1813–1817.
- Jian Wu, Yashesh Gaur, Zhuo Chen, and 1 others. 2023. On decoder-only architecture for speech-to-text and large language model integration. In *ASRU*, pages 1–8. IEEE.
- Wenyi Yu, Changli Tang, Guangzhi Sun, and 1 others. 2024. Connecting speech encoder and large language model for ASR. In *ICASSP*, pages 12637–12641. IEEE.