

000 001 002 003 004 005 006 007 008 009 010 011 012 013 014 015 016 017 018 019 020 021 022 023 024 025 026 027 028 029 030 031 032 033 034 035 036 037 038 039 040 041 042 043 044 045 046 047 048 049 050 051 052 053 PAY ATTENTION TO CTC: FAST AND ROBUST PSEUDO-LABELLING FOR UNIFIED SPEECH RECOGNITION

Anonymous authors

Paper under double-blind review

ABSTRACT

Unified Speech Recognition (USR) has emerged as a semi-supervised framework for training a single model for audio, visual, and audiovisual speech recognition, achieving state-of-the-art results on in-distribution benchmarks. However, its reliance on autoregressive pseudo-labelling makes training expensive, while its decoupled supervision of CTC and attention branches increases susceptibility to self-reinforcing errors, particularly under distribution shifts involving longer sequences, noise, or unseen domains. We propose CTC-driven teacher forcing, where greedily decoded CTC pseudo-labels are fed into the decoder to generate attention targets in a single forward pass. Although these can be globally incoherent, in the pseudo-labelling setting they enable efficient and effective knowledge transfer. Because CTC and CTC-driven attention pseudo-labels have the same length, the decoder can predict both simultaneously, benefiting from the robustness of CTC and the expressiveness of attention without costly beam search. We further propose mixed sampling to mitigate the exposure bias of the decoder relying solely on CTC inputs. The resulting method, USR 2.0, halves training time, improves robustness to out-of-distribution inputs, and achieves state-of-the-art results on LRS3, LRS2, and WildVSR, surpassing USR and modality-specific self-supervised baselines.

1 INTRODUCTION

Speech recognition includes multiple closely related tasks: automatic speech recognition (ASR) from audio (Abdel-Hamid et al., 2014; Chiu et al., 2018), visual speech recognition (VSR) from lip movements (Assael et al., 2016; Martinez et al., 2020), and audiovisual speech recognition (AVSR), which combines both modalities (Petridis et al., 2018; Ma et al., 2021). Despite their complementary nature, ASR, VSR, and AVSR have traditionally been studied in isolation (Baevski et al., 2020; Hsu et al., 2021; Assael et al., 2016), leading to separate models that are not only more cumbersome to deploy, but also forgo the advantages of sharing representations across modalities (Akbari et al., 2021; Girdhar et al., 2022). Recent work in audiovisual self-supervised learning (Shi et al., 2022a; Zhu et al., 2023; Lian et al., 2023) has explored unified pre-training across modalities but still often relies on separate fine-tuning stages for ASR, VSR, and AVSR, resulting in distinct models per task.

Unified Speech Recognition (USR) (Haliassos et al., 2024a) offers a promising step toward unification by incorporating unlabelled data during fine-tuning, thus reducing modality-specific sensitivity and enabling a single model to achieve state-of-the-art in-distribution performance across all tasks. Its decoupled two-branch architecture combines Connectionist Temporal Classification (CTC) (Graves et al., 2006) and attention-based (Chorowski et al., 2015) supervision, where each student branch is independently guided by pseudo-labels (PLs) from the corresponding branch of a momentum teacher (Tarvainen & Valpola, 2017). While effective in distribution, this design leaves USR’s attention decoder with two key limitations: (1) PL generation is a bottleneck, as its autoregressive (AR) decoding is slow and must run at every training step, and (2) decoupled supervision increases vulnerability to out-of-distribution (OOD) errors. That is, mistakes from AR decoding are compounded by the self-reinforcing training loop (Sohn et al., 2020): the student can be harmed by noisy AR PLs from the teacher, which can in turn degrade the teacher itself as it is a moving average of the student.

To address these limitations, we propose USR 2.0, which combines the efficiency and robustness of CTC with the expressiveness of attention-based decoding. As shown in Figure 1, CTC decoding is fast and robust to domain shift due to monotonic alignment and conditional independence assumptions,

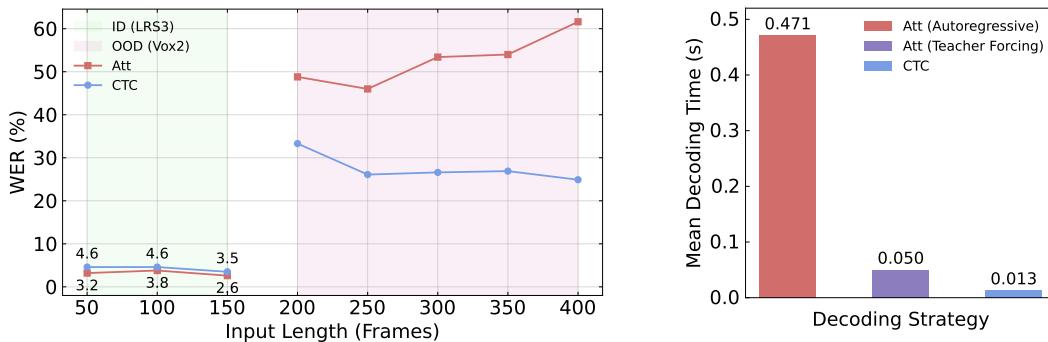


Figure 1: **CTC vs. attention-based decoding (greedy).** Left: AVSR word error rate (WER) of USR on in- (LRS3) and out-of-distribution (VoxCeleb2, automatically transcribed) samples. CTC decoding is notably more robust to domain shift and long sequences, while autoregressive attention-based decoding performs best in-distribution. Right: Decoding speed on a H200 GPU. CTC is $\sim 40\times$ faster than autoregressive decoding. While teacher forcing could significantly speed up attention-based decoding, it typically relies on ground-truth tokens unavailable during pseudo-labelling.

while attention-based decoding has higher in-distribution quality but depends on slow AR generation. USR 2.0’s core idea is CTC-driven teacher forcing: instead of decoding the teacher autoregressively, we feed its greedily decoded CTC outputs into the decoder to generate attention-based PLs in a single forward pass, removing the AR bottleneck. While this choice may appear counterintuitive—since the outputs can lack global coherence—a key insight is that in a pseudo-labelling setting coherence is unnecessary: teacher and student operate under the same forced inputs, making knowledge transfer effective. Further, because these attention-based PLs are derived from the CTC PLs (and are hence aligned), the student can predict both in a single decoder forward pass, coupling the two branches and improving robustness to OOD samples. Finally, as CTC-driven teacher forcing introduces a train–test mismatch (CTC inputs during training vs. AR tokens at inference), we mitigate it with a simple mixed sampling strategy that intermittently reintroduces AR decoding.

USR 2.0 improves robustness under domain shift, with large gains on long utterances, noisy audio, and cross-dataset evaluations across LibriSpeech (Panayotov et al., 2015), WildVSR (Djilali et al., 2024), and AVSpeech (Ephrat et al., 2018). It is also markedly less sensitive to beam size than USR, maintaining strong performance even under greedy decoding. These robustness improvements further translate to better *in-distribution* performance, as unlabelled data often comes from OOD sources (Chung et al., 2018; Shi et al., 2022a). As a result, USR 2.0 achieves state-of-the-art WER in various semi-supervised settings across ASR, VSR, and AVSR, *with nearly 2 \times faster training than USR*. Enabled by our findings, we scale USR 2.0 to a Huge model trained on ~ 2500 hours of unlabelled data, yielding WERs of 17.6% (VSR), 0.9% (ASR), and 0.8% (AVSR) on LRS3, along with state-of-the-art results on LRS2 and WildVSR (Appendix C.1), all using a single unified model.

2 RELATED WORK

CTC, attention, and joint training. Speech recognition typically relies on CTC (Graves et al., 2006; Baevski et al., 2020; Hsu et al., 2021) or attention-based encoder–decoder models (Chorowski et al., 2015; Chan et al., 2016; Radford et al., 2023). CTC enables efficient training and inference via monotonic alignment and conditional independence, improving robustness in noisy or out-of-distribution settings (Kim et al., 2017), whereas attention-based models capture richer dependencies through autoregressive decoding but may suffer from alignment errors and degrade under distribution shift (Watanabe et al., 2017). Joint CTC–attention training (Kim et al., 2017) supervises a shared encoder via separate branches, with inference performed by joint beam search and CTC rescoring (see Appendix B). While this strategy has been adopted in VSR and AVSR (Ma et al., 2022; 2023; Haliassos et al., 2022a; 2024a), its reliance on costly beam search limits its use in iterative pseudo-labelling. Non-autoregressive transformers (Ghazvininejad et al., 2019; Tian et al., 2020; Song et al., 2021) avoid AR decoding via parallel token generation but typically reduce accuracy, whereas our method applies CTC guidance to the attention-based decoder only during pseudo-labelling—where output sequence coherence is unnecessary—while retaining accurate AR beam search at inference.

108 **Self-supervised audiovisual speech models.** Audiovisual self-supervised learning has shown
 109 promise for speech recognition by learning contextualised representations from large-scale unlabelled
 110 audiovisual data (Shi et al., 2022a; Zhu et al., 2023; Lian et al., 2023). Pre-training typically relies on
 111 masked prediction (Baevski et al., 2022; He et al., 2022) or audiovisual correspondence (Chung &
 112 Zisserman, 2017; Morgado et al., 2021), followed by fine-tuning with a CTC layer and/or attention-
 113 based decoder (Cappellazzo et al., 2024; Rouditchenko et al., 2024). Although recent methods unify
 114 modalities in a single encoder during pre-training (Shi et al., 2022a; Zhu et al., 2023), the sensitivity
 115 of supervised fine-tuning to task-specific hyperparameters often complicates joint optimisation across
 116 modalities (Hsu & Shi, 2022; Shi et al., 2022a). As a result, fine-tuning is typically performed
 117 separately for ASR, VSR, and AVSR, increasing deployment costs (Haliassos et al., 2022a; 2024b)
 118 and forgoing the potential benefits of shared representations across modalities (Akbari et al., 2021).
 119

120 **Pseudo-labelling for speech recognition.** Pseudo-labelling is an effective strategy for semi-
 121 supervised speech recognition, where model-generated transcriptions for unlabelled data act as
 122 additional supervision (Sohn et al., 2020). Offline pseudo-labelling with a frozen model is effi-
 123 cient (Afouras et al., 2020; Ma et al., 2023), but requires external models and does not support
 124 iterative refinement. In contrast, self-training methods update pseudo-labels during training using
 125 the model itself, often with beam search (Park et al., 2020; Xu et al., 2020; Kahn et al., 2020).
 126 However, beam search is computationally expensive, especially at scale, prompting some works to
 127 adopt CTC-only supervision (Likhomanenko et al., 2020; Higuchi et al., 2021), at the cost of reduced
 128 sequence modelling capacity (Rouditchenko et al., 2023). USR (Haliassos et al., 2024a) shows that
 129 fine-tuning with unlabelled data using greedy CTC and attention-based pseudo-labelling mitigates
 130 the task-specific sensitivity observed in self-supervised approaches, enabling a *single* model for ASR,
 131 VSR, and AVSR. Building on this, we revisit USR’s decoupled pseudo-labelling scheme to improve
 132 training efficiency, robustness to distribution shifts, and in-distribution performance.
 133

3 BACKGROUND: USR

135 USR (Haliassos et al., 2024a) is a semi-supervised student-teacher framework for training a single
 136 model to perform ASR, VSR, and AVSR. It uses a shared encoder with two output heads—a CTC
 137 layer and an attention-based decoder—and leverages unlabelled data via iterative pseudo-labelling (see
 138 Figure 2, left). The student receives masked audio, video, and audiovisual inputs, and is trained to
 139 match the teacher’s outputs on unmasked audiovisual inputs, while also learning from ground-truth
 140 labels where available. Key components of USR are outlined below (more details in Appendix B.2).
 141

142 **Teacher–student setup.** Let θ_S and θ_T be the parameters of the student and teacher models. The
 143 student is optimised via gradient descent, and the teacher is updated as an exponential moving average
 144 of the student via $\theta_T \leftarrow \tau\theta_T + (1 - \tau)\theta_S$, where τ follows a cosine schedule from 0.998 to 1.
 145

146 **Pseudo-labelling.** Given an unlabelled audiovisual speech input represented as a feature sequence
 147 of length L , the teacher generates two types of pseudo-labels (PLs) via greedy decoding. The CTC
 148 head produces frame-level PLs \tilde{y}_t^{CTC} for $t \in [1, L]$, and the decoder (attention branch) autoregressively
 149 produces token-level PLs \tilde{y}_u^{Att} for $u \in [1, U_{\text{AR}}]$, where U_{AR} is the decoder output length:
 150

$$\tilde{y}_t^{\text{CTC}} = \arg \max_{y_t} P_{\text{CTC}}(y_t \mid x; \theta_T), \quad \tilde{y}_u^{\text{Att}} = \arg \max_{y_u} P_{\text{Att}}(y_u \mid \tilde{y}_{<u}^{\text{Att}}, x; \theta_T). \quad (1)$$

152 **Loss on unlabelled data.** The student is trained to match the teacher’s PLs for all masked modalities.
 153 Let $\hat{y}^{\omega, m}$ denote the student’s output from head $\omega \in \{\text{CTC}, \text{Att}\}$ for modality $m \in \{\text{A}, \text{V}, \text{AV}\}$. USR
 154 uses the cross-entropy loss (frame-wise or token-wise depending on the head ω):
 155

$$\mathcal{L}^{\omega, m} = \text{CE}(\hat{y}^{\omega, m}, \tilde{y}^{\omega}), \quad (2)$$

157 where we omit time and token indices t and u for ease of notation. The total unlabelled loss is a
 158 weighted combination across heads and modalities, and is jointly optimised with the supervised loss.
 159

160 **Limitations.** We argue that a key limitation of USR is its decoupled supervision: the student’s CTC
 161 and decoder branches are trained *independently* using their respective PLs. This design makes the
 162 attention decoder brittle on OOD inputs (e.g., noisy audio or long utterances), because (1) greedy

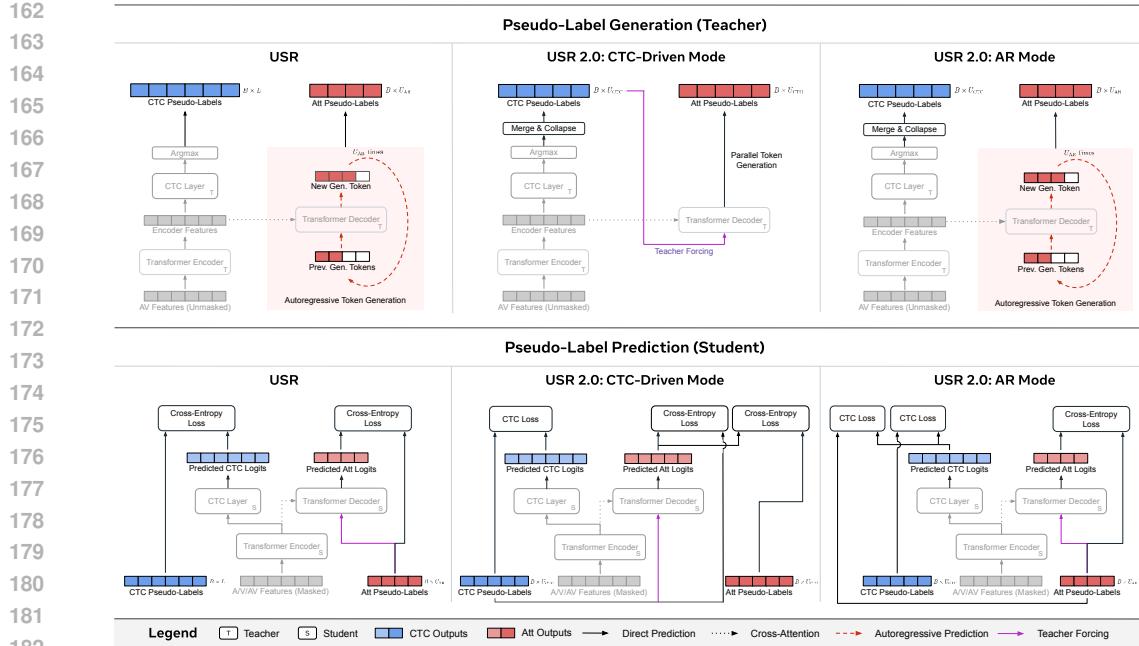


Figure 2: **Pseudo-labelling in USR and USR 2.0** In USR (left), the teacher generates CTC and attention-based pseudo-labels (PLs) from unmasked audiovisual inputs: CTC PLs are generated in parallel, while attention-based PLs require autoregressive decoding. The student predicts each target type independently, leading to decoupled supervision. **USR 2.0** introduces two modes for tighter integration. In CTC-driven mode (centre), attention-based PLs are generated by feeding collapsed CTC PLs into the decoder via teacher forcing, avoiding autoregression. The student decoder predicts both types of PLs. In AR mode (right), the teacher operates autoregressively as in USR, and the student’s CTC branch predicts both CTC and attention-based PLs. USR 2.0 alternates between modes at each iteration: CTC-driven mode improves efficiency and robustness, while AR mode mitigates exposure bias. Note: B refers to batch size, and typically, $U_{CTC}, U_{AR} \ll L$ (not to scale in the figure).

autoregressive (AR) decoding can fail and produce cascading errors (Kim et al., 2017; Fan et al., 2018), and (2) errors are reinforced in the training loop, as noisy AR PLs supervise the student, which can then degrade the teacher via EMA updates. We validate this effect empirically in Section 5.

Equally problematic is the computational cost of generating attention-based PLs: because they are produced autoregressively, one forward pass is required per output token. Despite key–value caching, this makes pseudo-labelling slow, especially since decoding is performed at every training step. This inefficiency becomes a major bottleneck for scaling to larger models and longer sequences, and it limits the practical utility of semi-supervised training on large unlabelled corpora.

4 PROPOSED METHOD

To overcome these limitations, we propose USR 2.0 for faster and more robust pseudo-labelling. The key idea is CTC-driven teacher forcing, where greedily decoded CTC PLs replace slow AR decoding to generate attention targets in a single forward pass. This design both removes the AR bottleneck and transfers the robustness of CTC to the decoder via joint CTC-attention PL prediction. To address the resulting train–test discrepancy, we propose a simple mixed sampling strategy, intermittently reintroducing AR decoding. Together, these components preserve the modelling flexibility of attention while improving efficiency and robustness under distribution shift. An overview is in Figure 2.

4.1 CTC-DRIVEN TEACHER FORCING

We propose to leverage the teacher’s CTC outputs to guide the generation of attention-based PLs. Given an unlabelled audiovisual input x , we first greedily decode the teacher’s CTC head, then apply

216 a merge-and-collapse operation by removing blank tokens and merging repeated symbols:
 217

$$218 \quad \tilde{y}_t = \arg \max_{y_t} P_{\text{CTC}}(y_t \mid x; \theta_T), \quad \tilde{y}^{\text{CTC}} = \text{collapse}(\tilde{y}_{1:L}) \quad (3)$$

219 where $\text{collapse}(\cdot)$ denotes the standard CTC post-processing operation (Graves et al., 2006). This
 220 produces a sequence \tilde{y}^{CTC} of length $U_{\text{CTC}} \ll L$, which we then feed as input to the teacher decoder:
 221

$$222 \quad \tilde{y}_u^{\text{Att}} = \arg \max_{y_u} P_{\text{Att}}(y_u \mid \tilde{y}_{<u}^{\text{CTC}}, x; \theta_T), \quad u = 1, \dots, U_{\text{CTC}}, \quad (4)$$

223 replacing the autoregressive conditioning on past attention outputs with the *fixed* CTC token prefix
 224 $\tilde{y}_{<u}^{\text{CTC}}$. This removes the sequential dependency, allowing the entire sequence of attention-based PLs
 225 to be generated in *parallel* with a single forward pass. As a result, the modelling strength of attention-
 226 based pseudo-labelling is preserved while avoiding the high cost and fragility of autoregression.
 227

228 **Aligned targets.** Since the attention-based PLs are generated by conditioning the decoder directly
 229 on the CTC pseudo-labels, they inherit the same sequence length U_{CTC} . This shared alignment ensures
 230 that both pseudo-label types correspond position-wise, allowing the student decoder to predict them
 231 simultaneously in a single forward pass under teacher forcing on the CTC PLs. This design lets
 232 the decoder inherit robustness from predicting CTC pseudo-labels while retaining flexibility from
 233 predicting attention pseudo-labels.

234 **Global coherence.** In CTC-driven teacher forcing, each decoder output \tilde{y}_u^{Att} is conditioned on the
 235 fixed prefix of CTC tokens $\tilde{y}_{<u}^{\text{CTC}}$ rather than past decoder outputs $\tilde{y}_{<u}^{\text{Att}}$. A natural concern is that the
 236 resulting sequence \tilde{y}^{Att} may lack coherence (see Appendix C.4 for discussion). While this prevents
 237 its use for *inference-time* decoding, the student decoder learns effectively during *self-training* with
 238 token-wise cross-entropy because of matched conditioning: teacher and student are conditioned on
 239 the same CTC-derived sequence, which is itself coherent. The student is trained to predict the teacher
 240 decoder’s most likely next token under this shared input. **The student decoder therefore learns a**
 241 **stable mapping from a coherent CTC prefix to the teacher’s conditionally valid next-token prediction.**
 242 **Because inference is autoregressive and relies on this learned prefix-to-next-token relation, global**
 243 **incoherence of the teacher-generated sequence does not hinder learning.**

244 4.2 MIXED SAMPLING

245 While CTC-driven teacher forcing allows attention-based PLs to be generated efficiently, it introduces
 246 a mismatch: the decoder is trained using inputs derived from the teacher’s CTC predictions, whereas
 247 at inference time it autoregressively conditions on its own past outputs. This discrepancy can be
 248 seen as a form of “exposure bias,” which can hurt performance (Ranzato et al., 2015). **In addition to**
 249 **this input-side mismatch, CTC’s conditional-independence assumptions may also occasionally yield**
 250 **imperfect CTC predictions on challenging, in-distribution segments, and these errors can propagate**
 251 **through the teacher to produce weaker attention-based pseudo-labels.** To address these issues, we
 252 introduce a simple yet effective mixed sampling strategy, similar to scheduled sampling (Bengio et al.,
 253 2015) but for a different type of exposure bias. At each training step, we randomly sample a mode:
 254 with probability 0.5 we use the CTC-driven mode, otherwise we adopt the standard AR mode¹.

255 In **CTC-driven mode**, the student CTC branch outputs $\hat{y}^{\text{CTC},m}$ for modality m , while the decoder
 256 outputs $\hat{y}^{\text{Att},m}$, conditioned on the same CTC PLs \tilde{y}^{CTC} used to condition the teacher. The losses are:

$$257 \quad \mathcal{L}^{\text{CTC},m} = \text{CTC}(\hat{y}^{\text{CTC},m}, \tilde{y}^{\text{CTC}}), \quad \mathcal{L}^{\text{Att},m} = 0.5 \text{CE}(\hat{y}^{\text{Att},m}, \tilde{y}^{\text{Att}}) + 0.5 \text{CE}(\hat{y}^{\text{Att},m}, \tilde{y}^{\text{CTC}}). \quad (5)$$

258 The decoder is supervised with both the teacher’s attention-based pseudo-labels \tilde{y}^{Att} and the CTC
 259 targets \tilde{y}^{CTC} to leverage the strengths of both PL types. The student’s CTC branch is supervised only
 260 with \tilde{y}^{CTC} , since \tilde{y}^{Att} may lack coherence (as explained) and could degrade CTC training.

261 In **AR mode**, the attention PLs \tilde{y}^{Att} are generated via standard AR decoding to alleviate train-test
 262 mismatch. The student decoder is trained to predict \tilde{y}^{Att} , while the CTC branch is jointly supervised
 263 with both \tilde{y}^{CTC} and \tilde{y}^{Att} , as both now represent coherent sequences. The corresponding losses are:

$$264 \quad \mathcal{L}^{\text{CTC},m} = 0.5 \text{CTC}(\hat{y}^{\text{CTC},m}, \tilde{y}^{\text{CTC}}) + 0.5 \text{CTC}(\hat{y}^{\text{CTC},m}, \tilde{y}^{\text{Att}}), \quad \mathcal{L}^{\text{Att},m} = \text{CE}(\hat{y}^{\text{Att},m}, \tilde{y}^{\text{Att}}). \quad (6)$$

265 ¹We also experimented with an adaptive sampling schedule (Appendix C.2), where the probability of AR
 266 pseudo-labelling increases over training. This strategy performed similarly to the fixed 0.5 probability used in
 267 USR 2.0, so we adopt the simpler approach in the main method.

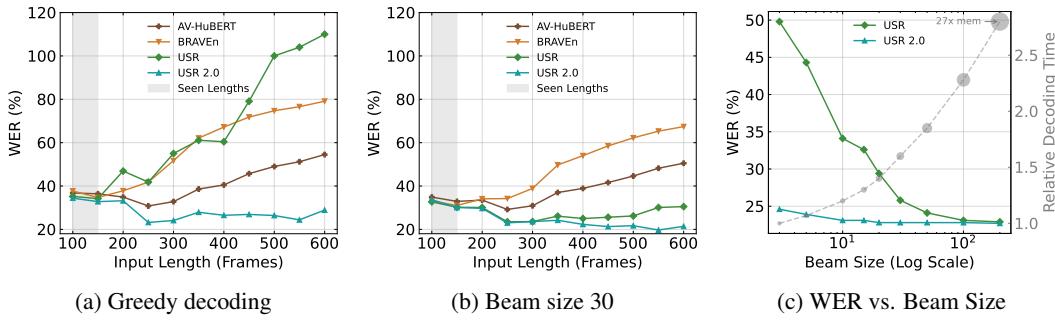


Figure 3: **Robustness to long utterances.** (a) Greedy decoding: USR 2.0 maintains robustness to longer input lengths, significantly outperforming other models. (b) Beam search (beam size = 30, joint CTC–attention decoding) improves USR robustness but still lags behind USR 2.0. (c) Increasing beam size reduces the WER gap between USR and USR 2.0, but at a significant computational and memory cost. The size of the markers corresponds to the relative memory cost with batched beam search.

Note that only the CTC branch receives both types of pseudo-labels as supervision: \hat{y}^{CTC} and \hat{y}^{Att} may differ in length, making it infeasible to supervise the decoder on both in a single forward pass.

4.3 IMPLEMENTATION DETAILS

Our implementation follows USR, with the main difference being our proposed pseudo-labelling strategy. We summarise the setup below; full details, including model sizes and hyperparameters, are given in Appendix A.

Architecture. We use a transformer encoder–decoder with modality-specific ResNet-18 backbones (He et al., 2016; Stafylakis & Tzimiropoulos, 2017) for audio and video. The audio encoder subsamples to match the visual frame rate (25 fps). Audio and visual features are projected via linear layers into a shared transformer dimension, while audiovisual features are formed by concatenation and applying a separate projection (Haliassos et al., 2024a). The resulting representations are processed by a transformer encoder with relative positional embeddings (Dai et al., 2019). We experiment with four model sizes (Base, Base+, Large, Huge); details are in Table 5.

Loss weights. We follow joint CTC–attention training (Kim et al., 2017), combining the CTC loss (weight 0.1) with attention-based cross-entropy loss using label smoothing 0.1 (Szegedy et al., 2016). Following USR, we use modality-specific weights of 0.3 (visual) and 0.7 (audio and audiovisual), and unlabelled-to-labelled loss ratios of 0.97 (visual) and 0.75 (audio and audiovisual).

Thresholding. We adopt confidence-based filtering as in USR, with a threshold of 0.8. For collapsed CTC PLs, sequence-level confidence is computed as the average log-probability over token predictions.

Inference. Unless otherwise stated, inference uses joint CTC–attention decoding via ESPnet (Watanabe et al., 2018), with beam size 40 and CTC weight 0.1. We use a 1,000-token SentencePiece vocabulary (Kudo & Richardson, 2018) trained on the labelled data.

5 OUT-OF-DISTRIBUTION RESULTS

Out-of-distribution robustness is critical not only for deployment but also for semi-supervised learning, since unlabelled data often comes from domains outside the labelled distribution. Better robustness yields higher-quality pseudo-labels and thus more reliable training. To assess USR 2.0’s robustness, we evaluate the Base model under long utterances, noisy audio, and unseen datasets, comparing against USR and strong self-supervised baselines AV-HuBERT (Shi et al., 2022a) and BRAVE (Haliassos et al., 2024b). All experiments are conducted in the low-resource setting (Shi

324 **Table 1: Comparison of WER (%) across different SNR levels (dB) for ASR and AVSR on the**
 325 **LRS3 test set (beam size 30). None of the methods saw noise-augmented samples during training.**

Method	All Samples					Samples with > 100 Frames				
	10 dB	5 dB	0 dB	-5 dB	Avg	10 dB	5 dB	0 dB	-5 dB	Avg
ASR										
AV-HuBERT	8.4	18.9	54.2	94.6	44.0	6.5	16.2	49.3	97.2	42.3
BRAVE _n	7.2	16.8	50.1	99.2	43.3	6.4	13.8	49.2	108.4	44.5
USR	5.8	14.3	48.5	104.4	43.3	4.9	12.6	48.1	111.7	44.3
USR 2.0	5.2	13.4	44.0	94.4	39.3	3.8	10.6	42.8	98.3	38.9
AVSR										
AV-HuBERT	5.8	10.5	24.3	56.9	24.4	4.4	8.0	18.0	48.7	19.8
BRAVE _n	7.2	14.0	32.2	54.6	27.0	6.5	12.7	30.9	55.5	26.4
USR	4.0	6.4	14.8	36.5	15.4	2.7	4.6	11.6	29.1	12.0
USR 2.0	3.7	5.6	14.0	33.1	14.1	2.6	4.3	10.4	26.0	10.8

340 et al., 2022a), where the labelled set is the 30-hour “trainval” partition of LRS3 (Afouras et al., 2018)
 341 and the remaining LRS3 samples are treated as unlabelled data.

343 5.1 ROBUSTNESS TO LONG UTTERANCES

344 We first evaluate robustness to long sequences using \sim 2,000 automatically transcribed samples from
 345 VoxCeleb2 (Chung et al., 2018), treating Whisper (Radford et al., 2023), trained on far more data, as
 346 an oracle for WER. The test set is bucketed into \sim 200-sample groups by input length (50–600 frames,
 347 in 50-frame increments). The longest labelled training utterance is 155 frames, while unlabelled
 348 samples reach 600, so inputs beyond 155 represent a distributional shift relative to the labelled data.
 349

350 **Greedy decoding.** Figure 3a shows WER versus input length under greedy (attention-based)
 351 decoding, which is important for pseudo-labelling. All methods perform similarly for inputs under
 352 150 frames, but WER rises sharply beyond that for BRAVE_n, AV-HuBERT, and especially USR,
 353 likely due to compounded autoregressive errors. While longer sequences should theoretically help by
 354 providing more context, distributional mismatch dominates. In contrast, USR 2.0 remains stable due
 355 to its integrated CTC and attention supervision, which mitigates autoregressive drift.
 356

357 **Beam search.** We next evaluate joint CTC–attention decoding with beam size 30. For AV-HuBERT,
 358 which lacks CTC, we use attention-only decoding. As shown in Figure 3b, USR benefits strongly
 359 from beam search, as CTC scoring helps correct decoder errors. However, performance still drops
 360 beyond 400 frames, suggesting that inference-time correction does not fully compensate for a poorly
 361 trained decoder. In contrast, USR 2.0 remains strong, indicating greater underlying robustness.
 362

363 **Beam size.** Finally, we vary the beam size and plot WER for USR and USR 2.0 across all samples
 364 (Figure 3c). At small beams, USR 2.0 clearly outperforms USR, highlighting its robustness under fast
 365 decoding. As beam size increases, the gap narrows, with very large beams compensating for USR’s
 366 weaker decoder at the cost of significantly higher latency and memory. Overall, USR 2.0 performs
 367 far better with small beams, making it well suited for pseudo-labelling and low-latency applications.
 368

369 5.2 ROBUSTNESS TO NOISE

370 In Table 1, we evaluate generalisation to noise on LRS3 with additive babble noise from NOI-
 371 SEX (Varga & Steeneken, 1993) at SNRs from 10 dB to -5 dB. As most LRS3 test samples are short,
 372 we also report results on those longer than 100 frames to better reflect real-world inputs. Importantly,
 373 none of the models were trained with noise augmentation, making this a zero-shot evaluation.
 374

375 For ASR, USR 2.0 clearly outperforms all baselines at moderate noise levels (10 to 0 dB), showing
 376 strong robustness. At -5 dB SNR, where the audio distribution shifts substantially, its performance is
 377 comparable to AV-HuBERT. On longer samples, the relative gap between USR 2.0 and USR slightly
 widens, suggesting that USR 2.0’s decoder may be more resilient to accumulated errors over time.
 378

378
 379
 380
 381
 382
 383
 384
 385
 386
 387
 388
 389
 390
 391
 392
 393
 394
 395
 396
 397
 398
 399
 400
 401
 402
 403
 404
 405
 406
 407
 408
 409
 410
 411
 412
 413
 414
 415
 416
 417
 418
 419
 420
 421
 422
 423
 424
 425
 426
 427
 428
 429
 430
 431
 Table 2: **In-distribution comparisons with self-supervised methods.** LRS3 WER (%) results for low- (left) and high-resource (right) labelled data settings. Best results in **bold**. We specify model size and pre-training dataset for each setting in **bold**. “Shared params” denotes whether a unified model is used. *Trained on LRS2+LRS3 (labelled) and English-only VoxCeleb2+AVSpeech (unlabelled).

Method	Shared params	V	A	AV	Method	Shared params	V	A	AV																																																																																																									
Base, LRS3																																																																																																																		
AV-HuBERT	✗	51.8	4.9	4.7	AV-HuBERT	✗	34.8	2.0	1.8																																																																																																									
RAVEEn	✗	47.0	4.7	-	VATLM	✗	34.2	-	1.7																																																																																																									
AV-data2vec	✗	45.2	4.4	4.2	RAVEEn	✗	33.1	1.9	-																																																																																																									
BRAVEEn	✗	43.4	4.0	4.0	AV-data2vec	✗	32.9	1.7	1.4																																																																																																									
USR	✓	36.0	3.2	3.0	Lip2Vec	✗	34.1	-	-																																																																																																									
USR 2.0	✓	36.2	3.0	2.9	BRAVEEn	✗	28.8	1.4	-																																																																																																									
Base+, LRS3+Vox2																																																																																																																		
AV-HuBERT	✗	46.1	4.6	4.0	USR	✓	26.5	1.6	1.3																																																																																																									
RAVEEn	✗	40.2	3.8	-	USR 2.0	✓	24.8	1.4	1.2																																																																																																									
AV-data2vec	✗	37.8	3.7	3.3	Large, LRS3+Vox2																																																																																																													
BRAVEEn	✗	35.1	3.0	-	USR	✓	28.4	2.6	2.5	AV-HuBERT	✗	28.6	1.3	1.4	USR 2.0	✓	26.4	2.5	2.4	VATLM	✗	28.4	-	1.2	Large, LRS3+Vox2										AV-HuBERT	✗	32.5	2.9	3.3	RAVEEn	✗	28.2	1.4	-	RAVEEn	✗	32.5	2.7	-	AV-data2vec	✗	28.5	1.3	1.3	AV-data2vec	✗	30.8	2.7	2.7	Lip2Vec	✗	26.0	-	-	BRAVEEn	✗	30.8	2.3	-	BRAVEEn	✗	26.6	1.2	-	USR	✓	26.9	2.4	2.4	u-HuBERT	✓	29.1	1.5	1.3	USR 2.0	✓	23.7	2.3	2.2	USR	✓	22.3	1.2	1.1	Huge*										USR 2.0	✓	17.6	0.9	0.8					
USR	✓	28.4	2.6	2.5	AV-HuBERT	✗	28.6	1.3	1.4																																																																																																									
USR 2.0	✓	26.4	2.5	2.4	VATLM	✗	28.4	-	1.2																																																																																																									
Large, LRS3+Vox2																																																																																																																		
AV-HuBERT	✗	32.5	2.9	3.3	RAVEEn	✗	28.2	1.4	-																																																																																																									
RAVEEn	✗	32.5	2.7	-	AV-data2vec	✗	28.5	1.3	1.3																																																																																																									
AV-data2vec	✗	30.8	2.7	2.7	Lip2Vec	✗	26.0	-	-																																																																																																									
BRAVEEn	✗	30.8	2.3	-	BRAVEEn	✗	26.6	1.2	-																																																																																																									
USR	✓	26.9	2.4	2.4	u-HuBERT	✓	29.1	1.5	1.3																																																																																																									
USR 2.0	✓	23.7	2.3	2.2	USR	✓	22.3	1.2	1.1																																																																																																									
Huge*																																																																																																																		
USR 2.0	✓	17.6	0.9	0.8																																																																																																														

Consistent with prior works (Ma et al., 2021; Haliassos et al., 2024b;a), AVSR outperforms ASR under noise by leveraging visual information. Across all SNRs, the pseudo-labelling methods (USR and USR 2.0) significantly outperform the self-supervised baselines (BRAVEEn and AV-HuBERT). Interestingly, the improvement of USR 2.0 over USR remains stable across sequence lengths, suggesting that its AVSR gains are less dependent on input duration than in ASR.

5.3 ROBUSTNESS TO OOD DATASETS

We evaluate robustness to OOD datasets with substantial gaps between training and test distributions, including shifts in vocabulary, speaker accents, background noise, and recording conditions. We use LibriSpeech test-clean (Panayotov et al., 2015) for ASR, WildVSR (Djilali et al., 2024) for VSR, and 1,000 manually filtered AVSpeech (Ephrat et al., 2018) samples for AVSR, transcribed using Whisper (Radford et al., 2023). As shown in Table 3, USR 2.0 outperforms all baselines by a wide margin under greedy decoding across all tasks, highlighting the robustness of its improved decoder to diverse, real-world shifts.

6 IN-DISTRIBUTION RESULTS

We now evaluate performance on the LRS3 benchmark, which serves as an *in-distribution* setting where the labelled and test data come from the same domain. However, because the unlabelled data often comes from other sources (e.g., VoxCeleb2), pseudo-labelling robustness remains critical.

Low- and high-resource settings. Table 2 presents WERs for low- (30h) and high-resource (433h) settings (Shi et al., 2022a), comparing against self-supervised methods, including RAVEEn (Haliassos et al., 2022a), AV-data2vec (Lian et al., 2023), VATLM (Zhu et al., 2023), Lip2Vec (Djilali et al., 2023), and u-HuBERT (Hsu & Shi, 2022). USR 2.0 matches or outperforms the state of the art, even

Table 3: **Performance on OOD datasets:** LibriSpeech (LibriS), WildVSR, and AVSpeech (AVS). We report WER (%) under greedy decoding.

Method	LibriS	WildVSR	AVS
AV-HuBERT	29.1	82.4	26.0
BRAVEEn	38.4	81.2	44.6
USR	25.3	80.0	34.7
USR 2.0	15.4	73.7	25.0

432
 433 **Table 4: AVSR WER (%) for different pseudo-**
 434 **label types** for each branch in CTC-driven and
 435 AR modes. Default settings are in gray .

CTC Head Preds		Decoder Preds		ID	OOD
CTC PL	Att PL	CTC PL	Att PL		
<i>CTC-Driven Mode</i>					
✓	✗	✓	✓	3.2	24.2
✓	✗	✗	✓	3.3	35.1
✓	✓	✓	✓	3.2	24.3
✓	✗	✓	✗	3.6	25.5
<i>AR Mode</i>					
✓	✓	✗	✓	2.9	40.1
✓	✗	✗	✓	3.0	45.1
✗	✓	✗	✓	2.8	52.3

448 methods that train separate models per task. Gains over USR are more pronounced with VoxCeleb2
 449 pre-training, particularly for VSR, which relies more heavily on pseudo-labels. These results support
 450 our hypothesis that improving pseudo-labelling benefits even in-distribution performance.

451 **Training efficiency.** Figure 5 plots WER for VSR
 452 against wall-clock training time for USR and USR 2.0
 453 across different model scales and pre-training datasets.
 454 We observe approximately 2× faster training for each set-
 455 ting, driven by two key factors: (i) faster training steps due
 456 to CTC-driven teacher forcing; and (ii) faster convergence,
 457 requiring fewer epochs (50 vs. 75; see Appendix C.5).
 458 These gains make USR 2.0 substantially more efficient for
 459 semi-supervised training at scale.

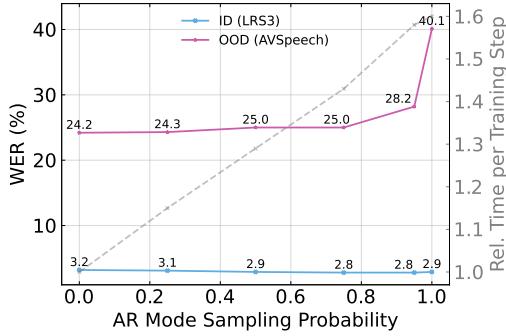
460 **Scaling.** Building on these results, we scale USR 2.0 to
 461 a Huge model trained on the combined LRS2 (Son Chung
 462 et al., 2017) and LRS3 labelled sets, along with a large
 463 unlabelled corpus from English-only VoxCeleb2 and
 464 AVSpeech. The model achieves WERs of 17.6%, 0.9%,
 465 and 0.8% on VSR, ASR, and AVSR, respectively, demon-
 466 strating USR 2.0’s scalability. Comparisons with additional methods are in Appendix C.1.

469 7 ABLATIONS

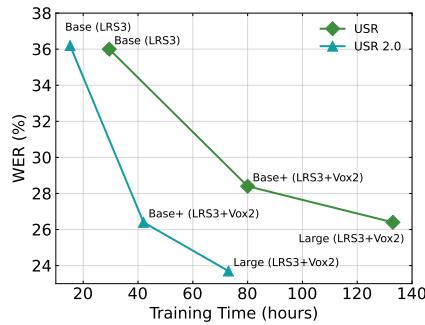
471 We ablate the choice of pseudo-label (PL) targets for each branch under the two modes: CTC-driven
 472 and AR. To isolate their effects, we disable AR mode when evaluating CTC-driven mode, and vice
 473 versa. All experiments use the low-resource setup with the Base model (see Section 5). We report
 474 AVSR WER on both in-distribution (ID; LRS3 test) and out-of-distribution (OOD; AVSpeech) data.
 475 More ablations are given in Appendix C.2.

477 **CTC-driven mode (Table 4).** By default, the CTC head predicts CTC PLs, while the decoder
 478 predicts both CTC and attention-based PLs. Removing CTC supervision from the decoder harms
 479 OOD performance, confirming its value for robustness. Dropping attention-based targets reduces ID
 480 performance, underscoring their importance for sequence modelling. Thus, both are crucial. Interest-
 481 ingly, supervising the CTC head with attention-based PLs does not noticeably degrade performance,
 482 suggesting that AR inconsistencies may be offset by higher-quality conditional predictions.

483 Finally, conditioning the decoder on CTC pseudo-labels in this mode avoids conditioning solely
 484 on AR-generated pseudo-labels, which can be error-prone under domain shift due to cascaded
 485 autoregressive mistakes. This effect is reflected in the large OOD gap between rows 2 of “CTC-
 486 Driven Mode” (35.1%) and “AR Mode” (45.1%).



443 **Figure 4: AVSR WER (%) vs. AR mode sam-
 444 pling probability** on in-distribution (LRS3) and
 445 out-of-distribution (AVSpeech) samples.



560 **Figure 5: VSR WER vs. training time**
 561 for the low-resource setting.

486 **AR mode (Table 4).** Here, the default assigns only attention-based PLs to the decoder (due to the
 487 target length mismatch) and both PL types to the CTC head. Removing attention-based targets from
 488 the CTC head reduces performance. Using only attention-based PLs in both branches degrades OOD
 489 performance, reaffirming the role of CTC PLs for robustness. Overall, ID and OOD trends are largely
 490 uncorrelated, and AR performs significantly worse than CTC-driven mode on OOD data.
 491

492 **Effect of mixed sampling probability (Figure 4).** We evaluate the effect of varying the probability
 493 of sampling the AR mode during training. As this probability increases, ID performance improves,
 494 likely due to reduced exposure bias. OOD performance remains relatively stable across most of
 495 the range but degrades sharply when approaching AR-only training. Training time per iteration
 496 also increases steadily with higher AR probability due to the cost of AR decoding. **The sampling
 497 probability therefore controls the balance between ID accuracy, OOD robustness, and training
 498 efficiency, and may be tuned to match application needs, with our default of 0.5 offering a strong
 499 overall balance.**
 500

501 8 CONCLUSION

502 We presented USR 2.0, which addresses two key limitations of USR that prevent scaling to larger
 503 models and diverse unlabelled data: the computational cost of autoregressive pseudo-labelling and the
 504 decoupled design that reduces robustness to out-of-distribution inputs. By introducing CTC-driven
 505 teacher forcing and mixed sampling, USR 2.0 combines the *efficiency* and *robustness* of CTC with
 506 the *modelling flexibility* of attention-based decoding. This design improves generalisation to long
 507 utterances, noisy conditions, and unseen datasets, while reducing training time by nearly half. These
 508 robustness gains also translate to improved in-distribution performance, since unlabelled data often
 509 comes from out-of-distribution sources. Scaling both model and corpus, USR 2.0 achieves state-of-
 510 the-art performance across ASR, VSR, and AVSR with a *single* model. Beyond USR, our insights
 511 can be applied to other speech recognition tasks, including audio-only, streaming, and multilingual
 512 ASR, which are active research areas where large unlabelled datasets are common. **More broadly, the
 513 CTC-driven teacher-forcing paradigm could be applied to other sequence-to-sequence settings where
 514 the input and output follow the same temporal order but lack explicit frame-level alignment, such as
 515 handwriting recognition, music transcription, or DNA/protein sequencing. We hope our insights will
 516 prove useful for future work on scalable sequence-to-sequence self-training beyond speech.**
 517

518 REFERENCES

519 Ossama Abdel-Hamid, Abdel-rahman Mohamed, Hui Jiang, Li Deng, Gerald Penn, and Dong Yu.
 520 Convolutional neural networks for speech recognition. *IEEE/ACM Transactions on audio, speech,
 521 and language processing*, 22(10):1533–1545, 2014.

523 Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. Lrs3-ted: a large-scale dataset for
 524 visual speech recognition. *arXiv preprint arXiv:1809.00496*, 2018.

525 Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. Asr is all you need: Cross-modal
 526 distillation for lip reading. In *ICASSP 2020-2020 IEEE International Conference on Acoustics,
 527 Speech and Signal Processing (ICASSP)*, pp. 2143–2147. IEEE, 2020.

529 Hassan Akbari, Liangzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing
 530 Gong. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text.
 531 *Advances in Neural Information Processing Systems*, 34:24206–24221, 2021.

532 Yannis M Assael, Brendan Shillingford, Shimon Whiteson, and Nando De Freitas. Lipnet: End-to-end
 533 sentence-level lipreading. *arXiv preprint arXiv:1611.01599*, 2016.

535 Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework
 536 for self-supervised learning of speech representations. *Advances in neural information processing
 537 systems*, 33:12449–12460, 2020.

538 Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. Data2vec: A
 539 general framework for self-supervised learning in speech, vision and language. In *International
 Conference on Machine Learning*, pp. 1298–1312. PMLR, 2022.

540 Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence
 541 prediction with recurrent neural networks. *Advances in neural information processing systems*, 28,
 542 2015.

543 Umberto Cappellazzo, Minsu Kim, Honglie Chen, Pingchuan Ma, Stavros Petridis, Daniele Falavigna,
 544 Alessio Brutti, and Maja Pantic. Large language models are strong audio-visual speech recognition
 545 learners. *arXiv preprint arXiv:2409.12319*, 2024.

546 William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals. Listen, attend and spell: A neural network
 547 for large vocabulary conversational speech recognition. In *2016 IEEE international conference on
 548 acoustics, speech and signal processing (ICASSP)*, pp. 4960–4964. IEEE, 2016.

549 Oscar Chang, Hank Liao, Dmitriy Serdyuk, Ankit Shahy, and Olivier Siohan. Conformer is all you
 550 need for visual speech recognition. In *ICASSP 2024-2024 IEEE International Conference on
 551 Acoustics, Speech and Signal Processing (ICASSP)*, pp. 10136–10140. IEEE, 2024.

552 Chung-Cheng Chiu, Tara N Sainath, Yonghui Wu, Rohit Prabhavalkar, Patrick Nguyen, Zhifeng
 553 Chen, Anjuli Kannan, Ron J Weiss, Kanishka Rao, Ekaterina Gonina, et al. State-of-the-art
 554 speech recognition with sequence-to-sequence models. In *2018 IEEE international conference on
 555 acoustics, speech and signal processing (ICASSP)*, pp. 4774–4778. IEEE, 2018.

556 Jan K Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio.
 557 Attention-based models for speech recognition. *Advances in neural information processing
 558 systems*, 28, 2015.

559 Joon Son Chung and Andrew Zisserman. Out of time: automated lip sync in the wild. In *Computer
 560 Vision–ACCV 2016 Workshops: ACCV 2016 International Workshops, Taipei, Taiwan, November
 561 20-24, 2016, Revised Selected Papers, Part II 13*, pp. 251–263. Springer, 2017.

562 Joon Son Chung, Arsha Nagrani, and Andrew Zisserman. Voxceleb2: Deep speaker recognition.
 563 *arXiv preprint arXiv:1806.05622*, 2018.

564 Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov.
 565 Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint
 566 arXiv:1901.02860*, 2019.

567 Yasser Abdelaziz Dahou Djilali, Sanath Narayan, Haithem Boussaid, Ebtesam Almazrouei, and
 568 Merouane Debbah. Lip2vec: Efficient and robust visual speech recognition via latent-to-latent
 569 visual to audio representation mapping. In *Proceedings of the IEEE/CVF International Conference
 570 on Computer Vision*, pp. 13790–13801, 2023.

571 Yasser Abdelaziz Dahou Djilali, Sanath Narayan, Eustache LeBihan, Haithem Boussaid, Ebtesam
 572 Almazrouei, and Merouane Debbah. Do vsr models generalize beyond lrs3? In *Proceedings of the
 573 IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 6635–6644, 2024.

574 Ariel Ephrat, Inbar Mosseri, Oran Lang, Tali Dekel, Kevin Wilson, Avinatan Hassidim, William T
 575 Freeman, and Michael Rubinstein. Looking to listen at the cocktail party: A speaker-independent
 576 audio-visual model for speech separation. *arXiv preprint arXiv:1804.03619*, 2018.

577 Angela Fan, Mike Lewis, and Yann Dauphin. Hierarchical neural story generation. *arXiv preprint
 578 arXiv:1805.04833*, 2018.

579 Marjan Ghazvininejad, Omer Levy, Yinhuan Liu, and Luke Zettlemoyer. Mask-predict: Parallel
 580 decoding of conditional masked language models. *arXiv preprint arXiv:1904.09324*, 2019.

581 Rohit Girdhar, Mannat Singh, Nikhila Ravi, Laurens van der Maaten, Armand Joulin, and Ishan
 582 Misra. Omnivore: A single model for many visual modalities. In *Proceedings of the IEEE/CVF
 583 Conference on Computer Vision and Pattern Recognition*, pp. 16102–16112, 2022.

584 Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola,
 585 Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet
 586 in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

594 Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal
 595 classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings*
 596 of the 23rd international conference on Machine learning, pp. 369–376, 2006.

597

598 Alexandros Haliassos, Konstantinos Vougioukas, Stavros Petridis, and Maja Pantic. Lips don't lie:
 599 A generalisable and robust approach to face forgery detection. In *Proceedings of the IEEE/CVF*
 600 conference on computer vision and pattern recognition, pp. 5039–5049, 2021.

601 Alexandros Haliassos, Pingchuan Ma, Rodrigo Mira, Stavros Petridis, and Maja Pantic. Jointly learn-
 602 ing visual and auditory speech representations from raw data. *arXiv preprint arXiv:2212.06246*,
 603 2022a.

604

605 Alexandros Haliassos, Rodrigo Mira, Stavros Petridis, and Maja Pantic. Leveraging real talking faces
 606 via self-supervision for robust forgery detection. In *Proceedings of the IEEE/CVF conference on*
 607 *computer vision and pattern recognition*, pp. 14950–14962, 2022b.

608 Alexandros Haliassos, Rodrigo Mira, Honglie Chen, Zoe Landgraf, Stavros Petridis, and Maja Pantic.
 609 Unified speech recognition: A single model for auditory, visual, and audiovisual inputs. *arXiv*
 610 *preprint arXiv:2411.02256*, 2024a.

611

612 Alexandros Haliassos, Andreas Zinonos, Rodrigo Mira, Stavros Petridis, and Maja Pantic. Braven:
 613 Improving self-supervised pre-training for visual and auditory speech recognition. In *ICASSP*
 614 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),
 615 pp. 11431–11435. IEEE, 2024b.

616 Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image
 617 recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*,
 618 pp. 770–778, 2016.

619

620 Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked
 621 autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer*
 622 *vision and pattern recognition*, pp. 16000–16009, 2022.

623 Yosuke Higuchi, Niko Moritz, Jonathan Le Roux, and Takaaki Hori. Momentum pseudo-labeling for
 624 semi-supervised speech recognition. *arXiv preprint arXiv:2106.08922*, 2021.

625

626 Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text
 627 degeneration. *arXiv preprint arXiv:1904.09751*, 2019.

628 Wei-Ning Hsu and Bowen Shi. u-hubert: Unified mixed-modal speech pretraining and zero-shot
 629 transfer to unlabeled modality. *Advances in Neural Information Processing Systems*, 35:21157–
 630 21170, 2022.

631

632 Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhota, Ruslan Salakhutdinov,
 633 and Abdelrahman Mohamed. Hubert: Self-supervised speech representation learning by masked
 634 prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*,
 635 29:3451–3460, 2021.

636 Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q Weinberger. Deep networks with
 637 stochastic depth. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The*
 638 *Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pp. 646–661. Springer, 2016.

639

640 Jacob Kahn, Ann Lee, and Awni Hannun. Self-training for end-to-end speech recognition. In *ICASSP*
 641 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),
 642 pp. 7084–7088. IEEE, 2020.

643 Suyoun Kim, Takaaki Hori, and Shinji Watanabe. Joint ctc-attention based end-to-end speech
 644 recognition using multi-task learning. In *2017 IEEE international conference on acoustics, speech*
 645 *and signal processing (ICASSP)*, pp. 4835–4839. IEEE, 2017.

646

647 Taku Kudo and John Richardson. Sentencepiece: A simple and language independent subword
 tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*, 2018.

648 Jiachen Lian, Alexei Baevski, Wei-Ning Hsu, and Michael Auli. Av-data2vec: Self-supervised
 649 learning of audio-visual speech representations with contextualized target representations. *arXiv*
 650 *preprint arXiv:2302.06419*, 2023.

651 Tatiana Likhomanenko, Qiantong Xu, Jacob Kahn, Gabriel Synnaeve, and Ronan Collobert. slimipl:
 652 Language-model-free iterative pseudo-labeling. *arXiv preprint arXiv:2010.11524*, 2020.

653 Xubo Liu, Egor Lakomkin, Konstantinos Vougioukas, Pingchuan Ma, Honglie Chen, Ruiming Xie,
 654 Morrie Doulaty, Niko Moritz, Jachym Kolar, Stavros Petridis, et al. Synthvsr: Scaling up visual
 655 speech recognition with synthetic supervision. In *Proceedings of the IEEE/CVF Conference on*
 656 *Computer Vision and Pattern Recognition*, pp. 18806–18815, 2023.

657 Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv*
 658 *preprint arXiv:1608.03983*, 2016.

659 Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint*
 660 *arXiv:1711.05101*, 2017.

661 Pingchuan Ma, Stavros Petridis, and Maja Pantic. End-to-end audio-visual speech recognition with
 662 conformers. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and*
 663 *Signal Processing (ICASSP)*, pp. 7613–7617. IEEE, 2021.

664 Pingchuan Ma, Stavros Petridis, and Maja Pantic. Visual speech recognition for multiple languages
 665 in the wild. *Nature Machine Intelligence*, 4(11):930–939, 2022.

666 Pingchuan Ma, Alexandros Haliassos, Adriana Fernandez-Lopez, Honglie Chen, Stavros Petridis,
 667 and Maja Pantic. Auto-avsr: Audio-visual speech recognition with automatic labels. In *ICASSP*
 668 *2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*,
 669 pp. 1–5. IEEE, 2023.

670 Takaki Makino, Hank Liao, Yannis Assael, Brendan Shillingford, Basilio Garcia, Otavio Braga, and
 671 Olivier Siohan. Recurrent neural network transducer for audio-visual speech recognition. In *2019*
 672 *IEEE automatic speech recognition and understanding workshop (ASRU)*, pp. 905–912. IEEE,
 673 2019.

674 Brais Martinez, Pingchuan Ma, Stavros Petridis, and Maja Pantic. Lipreading using temporal
 675 convolutional networks. In *ICASSP 2020-2020 IEEE International Conference on Acoustics,*
 676 *Speech and Signal Processing (ICASSP)*, pp. 6319–6323. IEEE, 2020.

677 Pedro Morgado, Nuno Vasconcelos, and Ishan Misra. Audio-visual instance discrimination with
 678 cross-modal agreement. In *Proceedings of the IEEE/CVF conference on computer vision and*
 679 *pattern recognition*, pp. 12475–12486, 2021.

680 Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus
 681 based on public domain audio books. In *2015 IEEE international conference on acoustics, speech*
 682 *and signal processing (ICASSP)*, pp. 5206–5210. IEEE, 2015.

683 Daniel S Park, Yu Zhang, Ye Jia, Wei Han, Chung-Cheng Chiu, Bo Li, Yonghui Wu, and Quoc V Le.
 684 Improved noisy student training for automatic speech recognition. *arXiv preprint arXiv:2005.09629*,
 685 2020.

686 Stavros Petridis, Themis Stafylakis, Pingchuan Ma, Feipeng Cai, Georgios Tzimiropoulos, and Maja
 687 Pantic. End-to-end audiovisual speech recognition. In *2018 IEEE international conference on*
 688 *acoustics, speech and signal processing (ICASSP)*, pp. 6548–6552. IEEE, 2018.

689 KR Prajwal, Triantafyllos Afouras, and Andrew Zisserman. Sub-word level lip reading with visual
 690 attention. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*,
 691 pp. 5162–5172, 2022.

692 Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever.
 693 Robust speech recognition via large-scale weak supervision. In *International Conference on*
 694 *Machine Learning*, pp. 28492–28518. PMLR, 2023.

702 Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. Sequence level training
 703 with recurrent neural networks. *arXiv preprint arXiv:1511.06732*, 2015.

704

705 Andrew Rouditchenko, Ronan Collobert, and Tatiana Likhomanenko. Av-cpl: Continuous pseudo-
 706 labeling for audio-visual speech recognition. *arXiv preprint arXiv:2309.17395*, 2023.

707

708 Andrew Rouditchenko, Yuan Gong, Samuel Thomas, Leonid Karlinsky, Hilde Kuehne, Rogerio Feris,
 709 and James Glass. Whisper-flamingo: Integrating visual features into whisper for audio-visual
 710 speech recognition and translation. *arXiv preprint arXiv:2406.10082*, 2024.

711

712 Dmitriy Serdyuk, Otavio Braga, and Olivier Siohan. Transformer-based video front-ends for audio-
 713 visual speech recognition for single and multi-person video. *arXiv preprint arXiv:2201.10439*,
 714 2022.

715

716 Bowen Shi, Wei-Ning Hsu, Kushal Lakhotia, and Abdelrahman Mohamed. Learning audio-visual
 717 speech representation by masked multimodal cluster prediction. *arXiv preprint arXiv:2201.02184*,
 718 2022a.

719

720 Bowen Shi, Wei-Ning Hsu, and Abdelrahman Mohamed. Robust self-supervised audio-visual speech
 721 recognition. *arXiv preprint arXiv:2201.01763*, 2022b.

722

723 Brendan Shillingford, Yannis Assael, Matthew W Hoffman, Thomas Paine, Cian Hughes, Utsav
 724 Prabhu, Hank Liao, Hasim Sak, Kanishka Rao, Lorrayne Bennett, et al. Large-scale visual speech
 725 recognition. *arXiv preprint arXiv:1807.05162*, 2018.

726

727 Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Do-
 728 gus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning
 729 with consistency and confidence. *Advances in neural information processing systems*, 33:596–608,
 730 2020.

731

732 Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman. Lip reading sentences in
 733 the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp.
 734 6447–6456, 2017.

735

736 Xingchen Song, Zhiyong Wu, Yiheng Huang, Chao Weng, Dan Su, and Helen Meng. Non-
 737 autoregressive transformer asr with ctc-enhanced decoder input. In *ICASSP 2021-2021 IEEE
 738 International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5894–5898.
 739 IEEE, 2021.

740

741 Themos Stafylakis and Georgios Tzimiropoulos. Combining residual networks with lstms for
 742 lipreading. *arXiv preprint arXiv:1703.04105*, 2017.

743

744 Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking
 745 the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer
 746 vision and pattern recognition*, pp. 2818–2826, 2016.

747

748 Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency
 749 targets improve semi-supervised deep learning results. *Advances in neural information processing
 750 systems*, 30, 2017.

751

752 Zhengkun Tian, Jiangyan Yi, Jianhua Tao, Ye Bai, Shuai Zhang, and Zhengqi Wen. Spike-
 753 triggered non-autoregressive transformer for end-to-end speech recognition. *arXiv preprint
 754 arXiv:2005.07903*, 2020.

755

756 Andrew Varga and Herman JM Steeneken. Assessment for automatic speech recognition: Ii. noisex-
 757 92: A database and an experiment to study the effect of additive noise on speech recognition
 758 systems. *Speech communication*, 12(3):247–251, 1993.

759

760 Shinji Watanabe, Takaaki Hori, Suyoun Kim, John R Hershey, and Tomoki Hayashi. Hybrid
 761 ctc/attention architecture for end-to-end speech recognition. *IEEE Journal of Selected Topics in
 762 Signal Processing*, 11(8):1240–1253, 2017.

756 Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson
757 Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, Adithya Renduchintala,
758 and Tsubasa Ochiai. ESPnet: End-to-end speech processing toolkit. In *Proceedings of the 19th*
759 *Annual Conference of International Speech Communication Association (INTERSPEECH)*, pp.
760 2207–2211, 2018.

761 Ronald J Williams and David Zipser. A learning algorithm for continually running fully recurrent
762 neural networks. *Neural computation*, 1(2):270–280, 1989.

763
764 Qiantong Xu, Tatiana Likhomanenko, Jacob Kahn, Awni Hannun, Gabriel Synnaeve, and Ronan
765 Collobert. Iterative pseudo-labeling for speech recognition. *arXiv preprint arXiv:2005.09267*,
766 2020.

767
768 Qiushi Zhu, Long Zhou, Ziqiang Zhang, Shujie Liu, Binxing Jiao, Jie Zhang, Lirong Dai, Daxin Jiang,
769 Jinyu Li, and Furu Wei. Vatlm: Visual-audio-text pre-training with unified masked prediction for
770 speech representation learning. *IEEE Transactions on Multimedia*, 2023.

771

772

773

774

775

776

777

778

779

780

781

782

783

784

785

786

787

788

789

790

791

792

793

794

795

796

797

798

799

800

801

802

803

804

805

806

807

808

809

810 A EXPERIMENTAL SETUP
811812 A.1 DATASETS
813814 **LRS3.** The LRS3 dataset (Afouras et al., 2018) consists of approximately 430 hours of transcribed
815 TED Talk videos, featuring thousands of speakers and an open vocabulary. Videos are automatically
816 segmented into sentence-level utterances, with audio and visual streams synchronised via facial
817 tracking. The dataset exhibits rich variability in lighting, speaking styles, and head poses. The test
818 set, used for evaluation only, contains around 1 hour of speech by speakers held out from training.
819 LRS3 is released under the Creative Commons BY-NC-ND 4.0 license.820 **LRS2.** Sourced from British television broadcasts, LRS2 (Son Chung et al., 2017) provides 223
821 hours of audiovisual content with sentence-level transcriptions. While similar in structure to LRS3,
822 the video content in LRS2 is more controlled, with generally better lighting and resolution. The
823 test set contains roughly 30 minutes of clips spoken by unseen speakers and is frequently used for
824 benchmarking visual and audiovisual speech models. It is available for academic, non-commercial
825 research use.826 **WildVSR.** WildVSR (Djilali et al., 2024) is a challenging VSR benchmark collected from open-
827 domain video content. Designed to test generalisation, it includes more extreme conditions than
828 LRS3 or LRS2, such as diverse ethnicities, head motions, and video resolutions. It comprises around
829 5 hours of test material without overlap with LRS3 and is used exclusively for evaluation. WildVSR
830 is licensed under CC BY-NC-ND 4.0.831 **AVSpeech.** AVSpeech (Ephrat et al., 2018) contains over 4,700 hours of face-aligned speech
832 segments harvested from over 150,000 YouTube videos. Its scale and speaker diversity make it
833 valuable for self- or semi-supervised learning. We follow prior work (Ma et al., 2023) in filtering
834 the dataset to retain 1,323 hours of only English-speaking videos. AVSpeech is licensed under the
835 Creative Commons Attribution 4.0 International License (CC BY 4.0).836 **VoxCeleb2.** VoxCeleb2 (Chung et al., 2018) comprises over 2,400 hours of unconstrained, in-the-
837 wild recordings from 6,000+ speakers. The data spans multiple languages and noisy environments
838 such as interviews and live performances. We use an English-only subset curated in prior work (Shi
839 et al., 2022a), which includes 1,326 hours of utterances. VoxCeleb2 is distributed under the CC
840 BY-NC-ND 4.0 license.841 **LibriSpeech.** We include the test-clean set of LibriSpeech (Panayotov et al., 2015) as an additional
842 audio-only evaluation benchmark. It consists of 5.4 hours of read speech from audiobook recordings,
843 featuring high-quality audio with minimal background noise. LibriSpeech is released under the CC
844 BY 4.0 license.845 The lists of sampled videos used for out-of-distribution evaluation on AVSpeech and VoxCeleb2 are
846 included in the README .md of the source code provided in the supplementary material.847 A.2 EXTERNAL CODEBASES
848849 For the robustness experiments in Section 5, we directly ran the official codebases of prior methods
850 to ensure fair and consistent comparisons. Specifically, we used:851

- AV-HuBERT (Shi et al., 2022a): https://github.com/facebookresearch/av_hubert
- BRAVEEn (Haliassos et al., 2024b): <https://github.com/ahaliassos/raven>
- USR (Haliassos et al., 2024a): <https://github.com/ahaliassos/usr>

852 Where available, we used the officially released pre-trained models. In cases where no suitable model
853 was provided (i.e., audiovisual BRAVEEn, audio-only AV-HuBERT, and audiovisual AV-HuBERT
854 without noise augmentations), we trained the models ourselves using the official hyperparameters
855 and training protocols.

864 Table 5: **Model hyperparameters.** Parameter count includes encoder, decoder, and feature extractors.
865

	Base	Base+	Large	Huge
Total parameters (M)	86	171	503	953
Transformer blocks (enc / dec)	12 / 6	12 / 6	24 / 9	36 / 9
Attention dim	512	768	1024	1280
MLP size	2048	3072	4096	5120
Heads	8	12	16	16

873 Table 6: **Training hyperparameters** for different pre-training datasets and model sizes. “All” refers
874 to the combination of LRS2, LRS3, VoxCeleb2, and AVSpeech. Additional hyperparameters are
875 provided in Appendix A.5.
876

Hyperparameter	LRS3 (Base)	LRS3+VoxCeleb2	All (Huge)
Learning rate	3e-3	2e-3	7e-4
Weight decay	0.04	0.04	0.04
(β_1, β_2)	(0.9, 0.98)	(0.9, 0.98)	(0.9, 0.98)
Drop path rate	0.1	0.1 (Base), 0.2 (Large)	0.3
Frames per GPU (lab./unlab.)	600 / 4400	300 / 3200	—
Frames per GPU (high-resource)	—	1200 / 3200	700 / 2400

885
886 For all other baselines and state-of-the-art comparisons, we report numbers as provided in the original
887 publications.
888889
890

A.3 PRE-PROCESSING

891
892 Video frames are first stabilised to reduce jitter, then cropped to a 96×96 region centred on the
893 speaker’s mouth, and finally converted to grayscale. This pipeline follows prior work (Shi et al.,
894 2022a;b; Haliassos et al., 2022a; 2024b;a). No pre-processing is applied to the raw audio.895
896

A.4 MODEL VARIANTS

897
898 We employ four model sizes: Base, Base+, Large, and Huge. Each variant shares the same feature
899 extractors, with the encoder and decoder size being the primary differentiating factors. Configurations
900 are summarised in Table 5. Note that Base+ corresponds to the Base model used in related works (Shi
901 et al., 2022a; Lian et al., 2023; Zhu et al., 2023).902
903

A.5 TRAINING HYPERPARAMETERS

904
905 **Optimiser and scheduling.** Models are trained for 50 epochs using AdamW (Loshchilov & Hutter,
906 2017), with a 15-epoch linear warmup (Goyal et al., 2017) followed by cosine decay (Loshchilov &
907 Hutter, 2016). Drop path regularisation (Huang et al., 2016) and gradient clipping (threshold 3.0) are
908 applied for stability.909
910 **Augmentations.** Video augmentations include random cropping (88×88) and horizontal flipping
911 with probability 0.5, applied consistently across frames. Both modalities also use temporal zero-
912 masking for the student model to enforce context understanding: video inputs are masked with max
913 duration 0.4s, audio with 0.6s, following Ma et al. (2022); Haliassos et al. (2022a). Teacher inputs
914 are unmasked to preserve PL quality.915
916 **Training resources.** We fix seed to 42. Training takes ~ 1 day for Base (8 H200 GPUs), ~ 2 days
917 for Base+ (32 GPUs), and ~ 3 days for Large (32 GPUs). Base uses LRS3, Base+ and Large use
918 LRS3+VoxCeleb2, and Huge uses LRS2+LRS3+VoxCeleb2+AVSpeech over ~ 4 days on 64 GPUs.
919 Frame counts per GPU are tuned for utilisation.

918 **Initialisation.** All models are initialised from the same self-supervised checkpoints as USR. For
 919 Huge, we pre-train using the Large USR hyperparameters from Haliassos et al. (2024a) before
 920 semi-supervised fine-tuning.
 921

922 **Summary.** Table 6 lists key hyperparameters. Training configs, dataset prep, and evaluation code
 923 are included in the supplementary material.
 924

925 B SUPPLEMENTARY METHODOLOGICAL BACKGROUND

927 B.1 CTC AND ATTENTION-BASED MODELS

929 **Connectionist Temporal Classification (CTC).** CTC (Graves et al., 2006) models the probability
 930 of a target sequence $y = (y_1, \dots, y_U)$ given input frames x by marginalising over all monotonic
 931 alignment paths π :

$$932 P_{\text{CTC}}(y \mid x) = \sum_{\pi \in \mathcal{B}^{-1}(y)} \prod_{t=1}^T P(\pi_t \mid x), \quad (7)$$

935 where π_t is the label (or blank) at timestep t and \mathcal{B} collapses repeats and blanks to recover y . This
 936 formulation (i) enforces a left-to-right monotonic alignment, since paths are built only by inserting
 937 blanks or repeats (never by reordering labels), and (ii) assumes each π_t is conditionally independent
 938 of $\pi_{<t}$ given x , thereby allowing the path probability to factorise as $\prod_t P(\pi_t \mid x)$. This explicit
 939 alignment mechanism tends to be robust under distribution shifts, but the conditional independence
 940 assumption limits modelling capabilities (Watanabe et al., 2017).

941 **Attention-based encoder–decoder models.** In contrast, attention-based encoder–decoder models
 942 define the conditional probability:

$$944 P_{\text{Att}}(y \mid x) = \prod_{u=1}^U P(y_u \mid y_{<u}, x). \quad (8)$$

946 Unlike CTC, attention-based decoders do not assume conditional independence across timesteps,
 947 allowing them to model long-range dependencies and complex token interactions. However, they do
 948 not enforce monotonic alignment and must instead learn alignment patterns purely from data. As
 949 a result, they are highly expressive but often brittle under distribution shifts, such as longer input
 950 sequences or noisy data, where learned alignments may fail to generalise.

951 During training, these models are typically trained using teacher forcing (Williams & Zipser, 1989),
 952 where the ground-truth tokens $y_{<u}$ are fed as inputs to the decoder instead of its own predictions.
 953 This stabilises learning and significantly accelerates computation by enabling parallel sequence
 954 processing in transformer-based architectures. However, at inference time, the model must operate
 955 autoregressively, meaning each predicted token must be fed back into the decoder for the next step.
 956

957 **Joint CTC–attention training and decoding.** To leverage the strengths of both CTC and attention-
 958 based models, hybrid architectures incorporate a joint CTC–attention framework (Watanabe et al.,
 959 2017). This approach optimises a multi-task objective:

$$960 \mathcal{L} = \lambda \mathcal{L}_{\text{CTC}} + (1 - \lambda) \mathcal{L}_{\text{Att}}, \quad (9)$$

961 where \mathcal{L}_{CTC} is the CTC loss (typically applied to a projection of the transformer encoder outputs),
 962 \mathcal{L}_{Att} is the attention-based cross-entropy loss (applied to the decoder outputs), and λ is a tunable
 963 hyperparameter balancing the two objectives.

964 At inference time, joint decoding is often performed using beam search, combining CTC and attention
 965 scores:

$$966 S(y') = \alpha \log P_{\text{CTC}}(y' \mid x) + (1 - \alpha) \log P_{\text{Att}}(y' \mid x), \quad (10)$$

967 where y' is a candidate hypothesis and α controls the relative contribution of the CTC branch. The
 968 CTC probabilities help constrain the beam, reducing alignment errors in attention-based decoding
 969 and improving robustness to distribution shifts.

971 While effective, beam search is computationally expensive and thus impractical in iterative self-
 972 training setups, where decoding must be performed repeatedly over large unlabelled datasets.

972 B.2 ADDITIONAL DETAILS ON USR
973

974 While the main paper focuses on our new pseudo-labelling generation and prediction strategy, this
975 section supplements the brief recap in Section 3 with components of USR that are not the focus in
976 USR 2.0, specifically the confidence filtering strategy (excluded from the main text for simplicity)
977 and a more detailed formulation of the semi-supervised loss. For clarity, we describe the setting using
978 one labelled and one unlabelled sample per iteration (with audio, visual, and audiovisual views); in
979 practice, batches of b_l labelled and b_u unlabelled samples are used, typically with $b_u > b_l$.
980

981 **Shared parameters.** USR is a single model trained to perform ASR, VSR, and AVSR jointly. Aside
982 from modality-specific feature extractors (audio and visual frontends), the remaining components
983 (encoder and decoder heads) are shared across modalities. These shared components operate on token
984 sequences, allowing audio, visual, and audiovisual inputs to be processed in a unified transformer
985 architecture after appropriate modality-specific projections. A key benefit of this design in the
986 self-training setup is that pseudo-labels only need to be generated once per unlabelled audiovisual
987 sample. These shared pseudo-labels can then supervise all three modality-specific inputs (audio-only,
988 video-only, and audiovisual), amortising the cost of pseudo-label generation.
989

990 **Confidence filtering.** To reduce the impact of erroneous pseudo-labels during self-training, USR
991 applies confidence-based filtering, discarding low-confidence predictions from the teacher model.
992 Let $p^{\text{CTC}} \in [0, 1]^{L \times |\mathcal{V}|}$ and $p^{\text{Att}} \in [0, 1]^{U_{\text{AR}} \times |\mathcal{V}|}$ denote the softmax outputs of the teacher’s CTC
993 and attention-based heads, respectively, where \mathcal{V} is the subword vocabulary. Frame- and token-wise
994 masks are obtained as follows:
995

$$r_t^{\text{CTC}} = \mathbb{1}(\max_k p_{t,k}^{\text{CTC}} \geq \tau), \quad r_u^{\text{Att}} = \mathbb{1}(\max_k p_{u,k}^{\text{Att}} \geq \tau), \quad (11)$$

996 with confidence threshold $\tau = 0.8$.
997

998 In USR 2.0, the CTC PLs are token-level (rather than frame-level) due to the collapse operation, so
999 it is difficult to use the frame-wise confidence scores $c_t^{\text{CTC}} = \max_k p_{t,k}^{\text{CTC}}$ directly in the CTC loss
1000 function. Instead, we compute a single sequence-level confidence:
1001

$$\text{conf}(\hat{y}^{\text{CTC}}) = \exp \left(\frac{1}{L} \sum_{t=1}^L \log c_t^{\text{CTC}} \right). \quad (12)$$

1002 The CTC sequence-level mask is then given by
1003

$$r^{\text{CTC}} = \mathbb{1}(\text{conf}(\hat{y}^{\text{CTC}}) \geq \tau). \quad (13)$$

1004 This reflects the fact that in USR 2.0, CTC PL filtering must occur at the sequence level due to the
1005 way the CTC loss is computed.
1006

1007 **Unlabelled losses.** Let $\hat{y}^{\omega, m}$ denote the student predictions from head $\omega \in \{\text{CTC}, \text{Att}\}$ for modality
1008 $m \in \{\text{A}, \text{V}, \text{AV}\}$. In USR, the cross-entropy losses on unlabelled data (with confidence masking) are
1009

$$\mathcal{L}_u^{\text{CTC}, m} = \frac{1}{L} \sum_{t=1}^L r_t^{\text{CTC}} \text{CE}(\hat{y}_t^{\text{CTC}, m}, \tilde{y}_t^{\text{CTC}}), \quad (14)$$

$$\mathcal{L}_u^{\text{Att}, m} = \frac{1}{U_{\text{AR}}} \sum_{u=1}^{U_{\text{AR}}} r_u^{\text{Att}} \text{CE}(\hat{y}_u^{\text{Att}, m}, \tilde{y}_u^{\text{Att}}). \quad (15)$$

1010 In USR 2.0, the unlabelled losses from equation 5–equation 6 are similarly masked using r^{CTC} and
1011 r_u^{Att} .
1012

1013 The CTC and attention unlabelled losses are then combined per modality using:
1014

$$\mathcal{L}_u^m = \lambda_{\text{CTC}} \mathcal{L}_u^{\text{CTC}, m} + (1 - \lambda_{\text{CTC}}) \mathcal{L}_u^{\text{Att}, m}, \quad \lambda_{\text{CTC}} \in [0, 1]. \quad (16)$$

1015 **Labelled losses.** For labelled samples, USR employs the standard joint CTC–attention objective
1016 described in Section B. The resulting per-modality labelled loss is denoted by \mathcal{L}_l^m .
1017

1026 Table 7: **Comparisons with the state-of-the-art on LRS3.** *Labels include automatic transcriptions
 1027 from ASR models trained on large-scale, often non-public datasets. “Shared params” denotes whether
 1028 a unified model is used at inference. “ST” denotes offline self-training. [†]Uses Whisper as its encoder,
 1029 trained on ~ 680 k hours of transcribed speech data (Radford et al., 2023).

Method	Labelled hours	Unlabelled hours	Lang. model	Shared params	WER (%)		
					V	A	AV
Supervised*							
V2P	3,886	-	✗	✗	55.1	-	-
RNN-T	31,000	-	✗	✓	33.6	4.8	4.5
VTP	2,676	-	✓	✗	30.7	-	-
Llama-AVSR [†]	680,000	-	✓	✗	24.0	0.8	0.8
Auto-AVSR	1,902	-	✓	✗	23.5	1.0	1.0
Auto-AVSR	3,448	-	✓	✗	19.1	1.0	0.9
ViT3D-CM	90,000	-	✗	✗	17.0	-	1.6
SynthVSR	6,720	-	✓	✗	16.9	-	-
LP Conf	100,000	-	✗	✗	12.8	-	0.9
Whisper-Flamingo [†]	680,000	-	✓	✗	-	0.7	0.8
Self/semi-supervised							
AV-HuBERT w/ ST	433	1,326	✗	✗	28.6	-	-
RAVEEn w/ ST	433	1,326	✓	✗	23.1	1.4	-
BRAVEEn w/ ST	433	2,649	✓	✗	20.1	1.1	-
USR	433	1,326	✓	✓	21.5	1.2	1.1
USR 2.0	656	2,649	✗	✓	17.6	0.9	0.8

1052 **Final semi-supervised objective.** Let $w_m \in [0, 1]$ denote the per-modality weight and $\gamma_m \in [0, 1]$
 1053 control the balance between unlabelled and labelled losses. The full loss is:

$$\mathcal{L}_{\text{semi}} = \sum_{m \in \{\text{A, V, AV}\}} w_m [\gamma_m \mathcal{L}_{\text{u}}^m + (1 - \gamma_m) \mathcal{L}_{\text{l}}^m]. \quad (17)$$

1057 As described in Section 4.3, we set:

$$w_{\text{A}} = w_{\text{AV}} = 0.7, \quad w_{\text{V}} = 0.3, \quad \gamma_{\text{A}} = \gamma_{\text{AV}} = 0.75, \quad \gamma_{\text{V}} = 0.97. \quad (18)$$

C ADDITIONAL RESULTS

C.1 FULL STATE-OF-THE-ART COMPARISONS

1066 Tables 7, 8, and 9 present comparisons with prior work on LRS3, LRS2, and WildVSR. We report
 1067 results from strong supervised systems trained with vastly more labelled or automatically transcribed
 1068 data, including V2P (Shillingford et al., 2018), RNN-T (Makino et al., 2019), VTP (Prajwal et al.,
 1069 2022), Llama-AVSR (Cappellazzo et al., 2024), Auto-AVSR (Ma et al., 2023), ViT3D-CM (Serdyuk
 1070 et al., 2022), SynthVSR (Liu et al., 2023), LP Conf (Chang et al., 2024), Whisper-Flamingo (Rou-
 1071 ditchenko et al., 2024), CM-seq2seq (Ma et al., 2021), and CM-aux (Ma et al., 2022). All results
 1072 for USR 2.0 refer to our Huge model, trained on LRS2 and LRS3 labelled data and unlabelled
 1073 English-only VoxCeleb2 and AVSpeech, *without* the use of an external language model.

1074 USR 2.0 achieves strong performance across all modalities. On LRS2 and LRS3, it outperforms
 1075 Auto-AVSR, which leverages automatic transcriptions from external ASR models, and is on par
 1076 with Whisper-Flamingo and Llama-AVSR, which initialise from Whisper models trained on 680,000
 1077 hours of transcribed speech data. On WildVSR (Table 9), which contains video clips with high visual
 1078 variability, USR 2.0 obtains state-of-the-art performance, surpassing prior heavily supervised systems.
 1079 This highlights its robustness to in-the-wild conditions such as variable head pose, lighting, and
 background.

1080
 1081
 1082
 1083 **Table 8: Comparisons with the state-of-the-art on LRS2.** *Labels include automatic transcriptions
 1084 from ASR models trained on large-scale, often non-public datasets. “Shared params” denotes whether
 1085 a unified model is used at inference. “ST” denotes offline self-training. [†]Uses Whisper as its encoder,
 1086 trained on ~680k hours of transcribed speech data (Radford et al., 2023).

Method	Labelled hours	Unlabelled hours	Lang. model	Shared params	WER (%)		
					V	A	AV
Supervised*							
CM-seq2seq	380	-	✓	✗	37.9	3.9	3.7
CM-aux	1,459	-	✓	✗	25.5	-	-
VTP	698	-	✓	✗	28.9	-	-
VTP	2,676	-	✓	✗	22.6	-	-
Auto-AVSR	818	-	✓	✗	27.9	2.6	-
Auto-AVSR	3,448	-	✓	✗	14.6	1.5	1.5
Whisper-Flamingo [†]	680,000	-	✓	✗	-	1.3	1.4
Self/semi-supervised							
Uni-AVSR	223	60,000	✗	✗	43.2	2.7	2.6
LiRA	223	433	✓	✗	38.8	-	-
RAVEn	223	1,759	✗	✗	23.2	2.5	-
RAVEn w/ ST	223	1,759	✓	✗	17.9	2.3	-
USR	223	1,759	✓	✓	15.4	1.9	1.9
USR 2.0	656	2,649	✗	✓	12.6	1.3	1.3

1106
 1107
 1108
 1109
 1110
 1111 **Table 9: Comparisons with the state-of-the-art on WildVSR.** *Labels include automatic transcriptions
 1112 from ASR models trained on large-scale, often non-public datasets. “Shared params” denotes whether
 1113 a unified model is used at inference. “ST” denotes offline self-training.

Method	Labelled hours	Unlabelled hours	Lang. model	Shared params	WER (%)
Supervised*					
CM-seq2seq	1,459	-	✓	✗	58.4
VTP	698	-	✓	✗	75.6
VTP	2,676	-	✓	✗	68.7
Auto-AVSR	661	-	✓	✗	62.3
Auto-AVSR	1,759	-	✓	✗	49.3
Auto-AVSR	3,448	-	✓	✗	38.6
Self/semi-supervised					
AV-HuBERT	433	1,326	✗	✗	51.7
AV-HuBERT w/ ST	433	1,326	✓	✗	48.7
RAVEn	433	1,326	✗	✗	52.2
RAVEn w/ ST	433	1,326	✓	✗	46.7
USR	433	1,326	✓	✓	46.4
USR 2.0	656	2,649	✗	✓	38.5

1134 Table 10: **Ablation results.** We report WER (%) for VSR (V), ASR (A), and AVSR (AV) on
 1135 in-distribution test data (LRS3) and AVSR on out-of-distribution data (sampled, automatically
 1136 transcribed AVSpeech). Default settings for USR 2.0 are highlighted in gray .
 1137

1138 (a) Weight on attention vs. CTC pseudo-labels for
 1139 decoder loss in CTC-driven mode.

Weight	ID		OOD	
	V	A	AV	AV
0	36.7	3.7	3.6	25.5
0.25	37.1	3.5	3.4	24.2
0.5	36.6	3.3	3.2	24.2
0.75	36.4	3.2	3.2	28.1
1	36.5	3.3	3.3	35.1

1140 (b) Weight on attention vs. CTC pseudo-labels for
 1141 CTC loss in AR mode.

Weight	ID		OOD	
	V	A	AV	AV
0	36.3	3.2	3.0	45.1
0.25	36.7	3.2	2.9	41.9
0.5	36.5	3.1	2.9	40.1
0.75	36.2	3.0	2.9	44.4
1	36.1	3.0	2.8	52.3

1142 (c) Effect of CTC merge & collapse on pseudo-labels
 1143 in USR.

Method	ID		OOD	
	V	A	AV	AV
Without	36.0	3.2	3.0	34.7
With	36.2	3.2	2.9	36.9

1144 (d) AR mode sampling schedule.

Schedule	ID		OOD	
	V	A	AV	AV
Constant	36.2	3.0	2.9	25.0
Linear	36.5	2.9	2.9	24.8
Cosine	36.6	3.0	2.9	25.4

1145 These results demonstrate the benefits of scaling both model capacity and unlabelled data within
 1146 a unified semi-supervised learning framework. While USR 2.0 already outperforms many task-
 1147 specific baselines, we anticipate further improvements through continued scaling of compute and
 1148 data. Importantly, all results are obtained using a *single* shared model for ASR, VSR, and AVSR,
 1149 without relying on task-specific tuning or external language models.

1150 C.2 ADDITIONAL ABLATIONS

1151 We conduct additional ablations on the Base model in the low-resource setting, using LRS3 as the
 1152 unlabelled dataset. We evaluate in-distribution performance on the LRS3 test set for VSR, ASR, and
 1153 AVSR, and out-of-distribution performance on the sampled AVSpeech test set (AVSR with greedy
 1154 decoding). Results are shown in Table 10.

1155 **Loss weights for auxiliary pseudo-labels.** Recall that in CTC-driven mode, the decoder is trained
 1156 to match both CTC and attention-based pseudo-labels. Similarly, in AR mode, the CTC branch is
 1157 trained to predict both pseudo-label types. By default, the coefficients balancing the contribution
 1158 of these losses are set to 0.5 in both cases (see equation 5–equation 6 in the main text). We ablate
 1159 different values of these coefficients from $\{0, 0.25, 0.5, 0.75, 1\}$ for each prediction head.

1160 As shown in Table 10(a), assigning greater weight to CTC pseudo-labels in the decoder loss generally
 1161 improves out-of-distribution robustness but can degrade in-distribution accuracy, especially for
 1162 ASR and AVSR, which are more sensitive to pseudo-label noise. This trade-off highlights the
 1163 complementary inductive biases of CTC and attention: CTC promotes monotonic alignment and
 1164 stability, while attention models long-range dependencies that are more beneficial in clean, in-
 1165 distribution settings.

1166 Interestingly, Table 10(b) shows that combining CTC and attention-based pseudo-labels in the CTC
 1167 loss of the AR mode improves OOD performance over using either type alone, suggesting that the
 1168 two supervision signals reinforce each other when used in the CTC branch.

1169 **CTC merge & collapse.** USR uses a cross-entropy loss directly on frame-level pseudo-labels and
 1170 predictions, without collapsing repeated tokens or removing blanks. In contrast, USR 2.0 applies
 1171 the standard CTC post-processing step (merging repeats and deleting blanks) to convert frame-level
 1172 outputs into token-level pseudo-labels. To isolate the effect of this change, we compare training USR

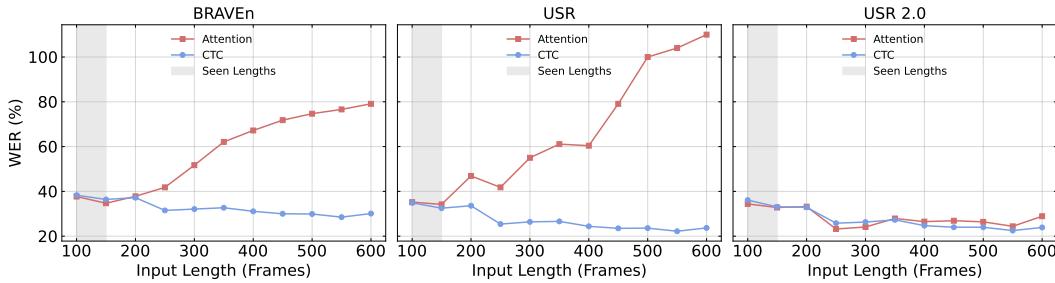


Figure 6: **WER vs. input length** on VoxCeleb2 (automatically transcribed using Whisper). Models are trained in the low-resource LRS3 setup, where labelled utterances are no longer than around 150 video frames. We evaluate CTC-only and attention-only greedy decoding for BRAVEn, USR, and USR 2.0.

with and without merge & collapse. As shown in Table 10(c), this choice has negligible impact on performance in USR. The benefit of collapsing in USR 2.0 stems not from improved label quality, but from enabling compatibility with teacher forcing in the decoder: it significantly reduces sequence length and maps the targets into the subword token space required by the decoder input.

Sampling schedule. USR 2.0 alternates between CTC-driven and AR modes when training on unlabelled data. By default, we use a constant sampling probability of 0.5 for AR mode throughout training. A natural hypothesis is to begin with more frequent use of the more robust CTC-driven mode, then gradually increase the use of AR mode to improve expressiveness. To test this, we evaluate both linear and cosine schedules for the AR sampling probability, increasing from 0 to 1 over the course of training. As shown in Table 10(d), neither schedule outperforms the simpler constant strategy, suggesting that fixed mixing is sufficient in this setting.

C.3 OUT-OF-DISTRIBUTION ROBUSTNESS

Length generalisation of decoding branches. Figure 6 evaluates the robustness of CTC-only and attention-only decoding across input lengths for BRAVEn, USR, and USR 2.0. We use the Base models trained in the low-resource setting (on LRS3), where the longest labelled utterances span roughly 150 frames. Evaluation is performed on out-of-distribution samples from VoxCeleb2, grouped into length buckets (e.g., 150–200, 200–250, etc), as in Figure 3 from the main text.

All models perform stably under CTC-only decoding, likely due to the robustness of monotonic alignment. However, attention-only decoding reveals notable differences. Both BRAVEn and USR exhibit sharp increases in WER as input length grows, reflecting the decoder’s fragility on long, out-of-distribution utterances. Notably, USR deteriorates more severely than BRAVEn, likely due to its self-training loop, where autoregressive pseudo-labels are generated by a teacher that is itself derived from the student, compounding early decoding errors. In contrast, USR 2.0 maintains significantly better performance, suggesting that its training strategy (i.e., the transfer of alignment information from the CTC branch to the decoder) successfully stabilises attention-based predictions.

Qualitative examples. To illustrate this effect, Table 11 presents qualitative decoding examples on long, out-of-distribution utterances. Transformer decoders are known to struggle with such inputs under greedy decoding, often producing truncated or repetitive outputs (Fan et al., 2018; Holtzman et al., 2019). These issues are amplified in pseudo-labelling setups, where standard remedies like beam search with length penalties are too costly to apply. In USR, we observe such failures on utterances longer than those seen during supervised training, typically manifesting as omissions or degeneracies.

USR 2.0 mitigates these issues by jointly supervising the decoder with both CTC and attention-based pseudo-labels. This promotes more stable and complete sequence generation, even under greedy decoding. As shown in Table 11, USR 2.0 produces fluent hypotheses, while USR suffers from repetition and content loss. These findings echo our quantitative results in Figures 6 and 3 of the main text, where USR 2.0 maintains robustness on long utterances.

1242

1243

1244

1245

1246

1247

1248

Table 11: **Qualitative analysis of USR vs. USR 2.0 on long, out-of-distribution utterances.** We show examples where USR either omits substantial segments (top) or degenerates into repetitions (bottom), both of which USR 2.0 resolves. Segments omitted by USR but correctly transcribed by USR 2.0 are highlighted in **green**; repetition errors by USR are highlighted in **red**.

(a) Omission Failures	
1249 GT	because with them we can break this cycle of suffering without them the cycle will continue I'll leave you with the word that we use at the end of every yoga class which also illustrates this fundamental truth I really believe in that we are all one
1250 USR	because with them we can break this cycle of every yoga class which also illustrates this fundamental truth I really believe in is that we are all one
1251 USR 2.0	because with them we can break this cycle of suffering without them the cycle will continue I'll leave you with the word that we use at the end of every yoga class which also illustrates this fundamental truth I really believe in is that we are all one
<hr/>	
1252 GT	so the evidence is there and it's going to be designed to interpret it correctly it will understand for example that books are very biased in the evidence they contain they only talk about kings and princes and elite white male people doing stuff so it's a complicated problem but as it learns more about our objectives it will become more and more
1253 USR	so it's a complicated problem but as it learns more about our objectives it will become more and more
1254 USR 2.0	so the evidence is there it's going to be designed to interpret it correctly it will understand for example that books are very biased in the evidence they contain they only talk about kings and princes and elite white male people doing stuff so it's a complicated problem but as it learns more about our objectives it will become more and more
<hr/>	
1255 GT	you may find these stories inspirational I have to be perfectly honest I find them daunting a little bit unsettling because I look at my own life and I ask is this going to happen to me and if it does will I recognize it and if I recognize it will I have the guts to make the leap myself so I took a look at some of these leaps into the void because if super successful peoples' success in fact does not hinge upon living on the edge
1256 USR	so I took a look at some of these leaps into the void because if super successful people success in fact does not hinge upon living on the edge
1257 USR 2.0	you may find these stories inspirational have to be perfectly honest I find them daunting a little bit unsettling because I look at my own life and I ask is this going to happen to me and if it does will I recognize it and if I recognize it will I have the guts to make the leap myself so I took a look at some of these leaps into the void because if super successful people's success in fact does not hinge upon living on the edge
<hr/>	
(b) Repetition Failures	
1258 GT	here's the striking thing though here's what Shery says Shery says yeah it's an incredibly painful experience but she wouldn't have it any other way because what it's taught her is
1259 USR	here's the striking thing though here's what Sharry says Sharry says Sharry says Sharry says Sharry says Sharry says Sharry says Sharry says...
1260 USR 2.0	here's the striking thing though here's what Shary says Shary says yes incredibly painful experience but she wouldn't have it any other way because it's what it's taught her is
<hr/>	
1261 GT	at the pebble on the bottom of the stream the stream is continuously moving and turbulent and that makes it very difficult to see the pebble on the bottom of the stream very much in the same way it's very difficult to see astronomical sources
1262 USR	at the pubble on the bottom of stream the stream the stream the stream...
1263 USR 2.0	at the puble on the bottom of stream the stream is continuously moving and turbulent and that makes it very difficult to see the puble on the bottom of the stream very much in the same way it's very difficult to see astronomical sources
<hr/>	
1264 GT	your grandmother was on TEDx Youth and that was like so cool back then I want to go like I went to this and I did that and I saw this I want to tell them stories
1265 USR	your grandmother was on TEDx Youth and that was like so cool back then see I want to go like I went to this and I did that and I saw this and I saw this and I saw this and I saw this...
1266 USR 2.0	your grandmother was on TEDx Youth and that was like so cool back then see I want to go like I went to this and I did that and I saw this and I wanted to tell them story

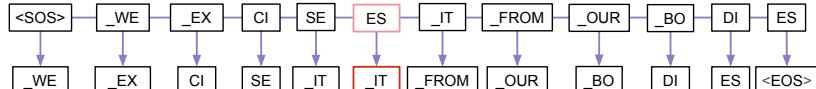
1296
1297**Example 1**

1298

Ground-truth sentence WE EXCISE IT FROM OUR BODIES

1299

CTC pseudo-labels



1300

CTC-driven attention
pseudo-labels

1303

1304

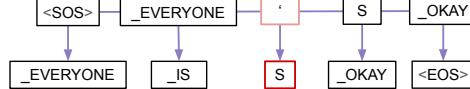
Example 2

1305

Ground-truth sentence: EVERYONE IS OKAY

1306

CTC pseudo-labels



1307

CTC-driven attention
pseudo-labels

1312

1313

Example 3

1314

Ground-truth sentence I DO WANT TO TALK ABOUT IT

1315

CTC pseudo-labels



1316

CTC-driven attention
pseudo-labels

1320

Figure 7: **Examples of AR inconsistencies** in decoder outputs under CTC-driven teacher forcing. In each case, the decoder is conditioned on CTC-derived prefixes, sometimes leading to repeated or malformed outputs at the sequence level, even though each token prediction is plausible given its input. These artefacts do not affect training, as the student learns to match the teacher’s outputs under the same input.

1321

1322

1323

1324

1325

1326

1327

1328

1329

C.4 CTC-CONDITIONED SEQUENCE-LEVEL INCONSISTENCIES

1330

1331

When using CTC-driven teacher forcing to generate decoder targets for self-training, we observe that the resulting attention-based outputs may exhibit *sequence*-level inconsistencies, even though each individual token prediction is *conditionally* valid. Specifically, the decoder is conditioned on the sequence of subwords predicted by the CTC branch, and the decoder produces the most plausible next token given each CTC-derived prefix. If the decoder and CTC branches diverge on certain tokens, the resulting attention-based output may contain repeated or malformed phrases when interpreted as a complete sequence.

1332

1333

Figure 7 illustrates examples of this phenomenon. In the first case, the CTC output incorrectly includes an extra subword (“ES” after “EXCISE”). When conditioned on a prefix ending in “EXCISE”, the decoder correctly predicts “IT”. However, for the subsequent prediction, the decoder sees the CTC-based prefix ending in “EXCISEES” and again predicts “IT”, which is a reasonable output under the given context. This results in a duplicated “IT” in the final sequence. In another case, the CTC branch outputs “EVERYONE’S”, leading the decoder to generate “IS” when conditioned on the prefix “EVERYONE”, followed by “S” under the prefix “EVERYONE” , resulting in the malformed word “ISS”. Finally, we show an example where the CTC output incorrectly predicts “DON’T” instead of “DO”, prompting the decoder to generate “DO’T” due to the prefix “DON”.

1334

1335

1336

1337

1338

1339

1340

1341

1342

1343

1344

1345

1346

1347

1348

1349

We emphasise again that these prefix-conditioned artefacts do *not* undermine the training process. The student decoder is trained to predict the teacher’s outputs under the *same* CTC-driven inputs, making the decoder’s predictions consistent *given* its inputs, even if they appear inconsistent at the sequence level.

1350 Table 12: **WER (%) for USR 2.0 under different decoding strategies at inference.** We compare
 1351 CTC-only decoding, attention-only greedy decoding, joint CTC + attention decoding with beam
 1352 search (beam size 30), and **CTC-driven teacher forcing**.

1353

1354

1355

1356

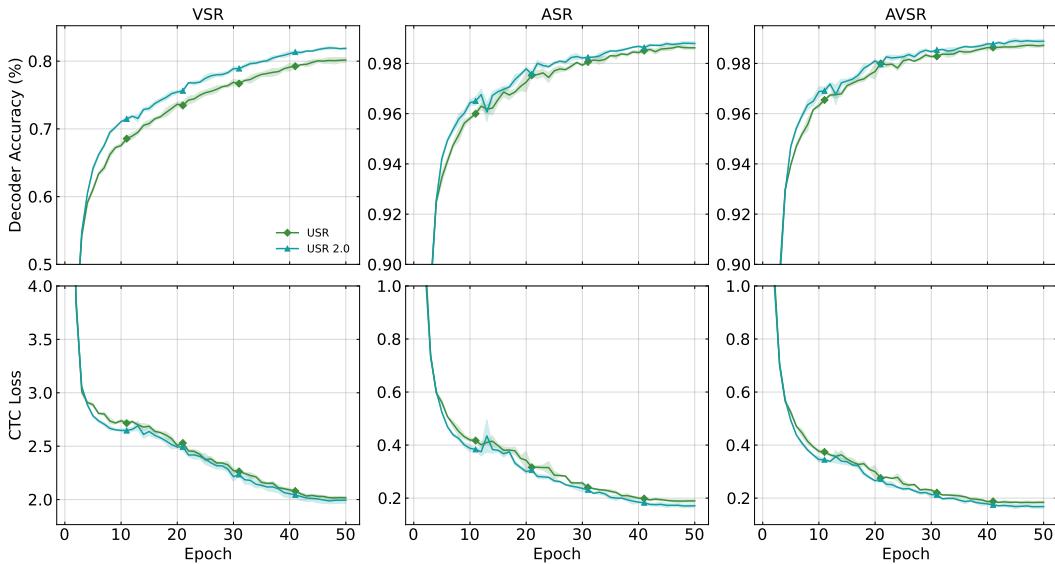
1357

1358

1359

1360

Decoding Method	V	A	AV
CTC-only (greedy)	45.6	4.1	4.0
Attention-only (greedy)	41.8	3.8	3.9
CTC + attention (beam size 30)	36.2	3.0	2.9
CTC-driven teacher forcing	47.1	4.2	4.1



1379 Figure 8: **Validation curves for USR and USR 2.0.** Top row shows decoder accuracy curves; bottom
 1380 row shows CTC validation loss. Each curve shows mean \pm std over three seeds.

1381

1382

1383

C.5 IN-DISTRIBUTION BEHAVIOUR

1384

Impact of decoding strategy at inference (in-distribution). Table 12 presents WER results for USR 2.0 under different decoding strategies on the in-distribution LRS3 low-resource benchmark. We observe that attention-only greedy decoding outperforms CTC-only decoding across ASR, VSR, and AVSR, reflecting the decoder’s strong sequence modelling capabilities. As expected, the best performance is achieved using joint CTC + attention decoding with beam search (albeit at a much higher computational and memory cost). On the other hand, as we have seen in Section C.3, attention decoding tends to degrade on out-of-distribution inputs such as longer utterances, while CTC-only decoding remains stable. This contrast underscores the complementary strengths of the two branches.

1392

1393

1394

1395

1396

1397

We also evaluated whether CTC-driven teacher forcing can be used as an inference-time decoding strategy. As shown in Table 12, this approach yields higher WER than standard greedy or beam-search decoding. This is expected: the benefit of CTC-driven teacher forcing arises during self-training from matched teacher–student conditioning. At inference, where the model must generate coherent outputs autoregressively, CTC-driven decoding does not perform better than conventional CTC-based or AR methods.

1398

1399

1400

1401

1402

1403

Validation curves and convergence. Figure 8 compares the validation dynamics of USR and USR 2.0 across VSR, ASR, and AVSR, using Base models trained in the low-resource setting over 50 epochs with LRS3 as the unlabelled data source (in-distribution setting). We report decoder accuracy (top row) and CTC validation loss (bottom row), averaged over three random seeds. Across all tasks, USR 2.0 consistently achieves higher decoder accuracy and lower CTC loss throughout training, indicating more effective learning within fewer total epochs.

1404
 1405 Table 13: **Training Duration Comparison.** We report LRS3 low-resource WER (%) results for 50-
 1406 and 75-epoch training. Mean and standard deviation over three random seeds are reported for the
 1407 50-epoch runs.

Method	Epochs	V	A	AV
USR	50	37.6 ± 0.7	3.4 ± 0.1	3.2 ± 0.1
USR 2.0	50	36.2 ± 0.5	3.0 ± 0.1	2.9 ± 0.1
USR	75	36.0	3.2	3.0
USR 2.0	75	36.1	3.0	2.9

1414
 1415 As shown in Table 13, extending training for USR to 75 epochs allows it to close much of the gap in
 1416 this in-distribution setting (where even the unlabelled data is in distribution), suggesting that USR 2.0
 1417 reaches comparable or better performance with fewer epochs. This contributes to its overall training
 1418 efficiency, which stems from both faster training steps and a reduced number of total steps (see
 1419 Figure 4 in the main text).

1421 C.6 FAILURE CASES

1423 Table 14 shows representative failure cases for the Huge USR 2.0 model on the LRS3 test set.
 1424 Although the model performs strongly across modalities, some error patterns remain. First, VSR
 1425 can produce visually plausible but semantically incorrect substitutions (e.g., “break” vs. “prod”,
 1426 “conscious” vs. “cautious”), which ASR and AVSR typically correct. Second, there are cases where
 1427 ASR can drift semantically (e.g., “It wasn’t a treatment” vs. “They want a treatment”), but the
 1428 audiovisual model resolves the ambiguity, illustrating how visual cues can compensate for uncertain
 1429 acoustics. Finally, a small number of errors persist across all modalities. These tend to be phonetically
 1430 understandable (e.g., “Signfeld” vs. “Seinfeld”, or “had” vs. “ought”).

1432 D DISCUSSION / LIMITATIONS

1433 Despite its advantages, USR 2.0 also presents limitations. First, while our proposed modifications
 1434 significantly reduce training time compared to USR (nearly halving it), they do not match the
 1435 efficiency of self-supervised approaches that rely on a fully supervised fine-tuning stage (Shi et al.,
 1436 2022a; Haliassos et al., 2022a; 2024b). In low-resource settings, where the labelled dataset is small,
 1437 such methods benefit from training the model on limited labelled data, making them substantially
 1438 faster in wall-clock time. In contrast, USR and USR 2.0 iteratively optimise on both labelled and
 1439 unlabelled data through self-training, leading to longer overall training durations. However, this
 1440 trade-off enables unified learning across ASR, VSR, and AVSR tasks. By leveraging pseudo-labelling
 1441 across modalities, USR 2.0 reduces hyperparameter sensitivity and achieves strong performance
 1442 on all tasks using a single model, without sacrificing *inference* speed and with significantly lower
 1443 deployment cost compared to training and maintaining separate models.

1444 Second, we observe that the benefit of additional unlabelled data is more pronounced for VSR
 1445 than for ASR or AVSR. This is likely due to the higher WERs in VSR, which allow it to benefit
 1446 from increased data volume even if pseudo-labels are noisy. In contrast, ASR and AVSR, which
 1447 already perform well in low-resource settings, may be bottlenecked by label quality rather than
 1448 quantity. As a result, greedy decoding (used here to keep pseudo-labelling computationally feasible)
 1449 may limit further improvements in ASR and AVSR, where higher-quality supervision could be
 1450 more beneficial. We believe that developing strategies for improving pseudo-label quality without
 1451 significantly compromising efficiency is a challenging but important area for future work.

1452 Third, CTC-driven teacher forcing is specifically designed for iterative self-training, where the teacher
 1453 evolves over time and pseudo-labels must be generated efficiently at every iteration. In this regime,
 1454 matching the teacher and student conditioning is more important than global sequence coherence,
 1455 and the method works well in practice. However, in non-self-training scenarios, such as offline
 1456 pseudo-labelling with a frozen pre-trained model or inference-time decoding, coherence becomes
 1457 desirable. In these settings, standard autoregressive or beam-search decoding is more appropriate, as
 1458 they (though slower) are run only once and do not dominate training cost.

1458 **Table 14: Representative failure cases for USR 2.0 on the LRS3 test set.** We show outputs from
 1459 the Huge model. Correct fragments are shown in green; errors are in red.
 1460

1461	GT	We push and pull and poke and prod things
1462	V	We push and pull and poke and break things
1463	A	We push and pull and poke and prod things
1464	AV	We push and pull and poke and prod things
1465		
1466	GT	You're more cautious
1467	V	You're more conscious
1468	A	You're more cautious
1469	AV	You're more cautious
1470		
1471	GT	And one activist actually found a contract from a western company for the sale of surveillance
1472	V	And one activist asked if I could contract from a western company for sale surveillance
1473	A	And one activist actually found a contract from a western company for the sale of surveillance
1474	AV	And one activist actually found a contract from a western company for the sale of surveillance
1475		
1476	GT	They want a treatment
1477	V	They want that treatment
1478	A	It wasn't a treatment
1479	AV	They want a treatment
1480		
1481	GT	I said we work together
1482	V	I think we work at the
1483	A	I said we worked together
1484	AV	I said we work together
1485		
1486	GT	What difference does it make if they talk like Jerry Seinfeld
1487	V	What difference does it make if they talk like church they fail
1488	A	What difference does it make if they talk like Jerry Signfeld
1489	AV	What difference does it make if they talk like Jerry Signfeld
1490		
1491	GT	That's something we ought to celebrate
1492	V	That's something we can't celebrate
1493	A	That's something we had to celebrate
1494	AV	That's something we had to celebrate
1495		
1496		
1497		
1498	<h2>E BROADER IMPACT</h2>	
1499		

1500 Unified speech recognition systems like USR 2.0 have the potential to expand accessibility across
 1501 a wide range of real-world settings. ASR can facilitate communication for individuals with motor
 1502 impairments or reading difficulties, while VSR / AVSR offer an alternative for users with aphonia or
 1503 in environments with poor audio quality. Moreover, the techniques developed here may contribute to
 1504 broader advances in multimodal machine learning, including audiovisual forensics. For instance, it
 1505 has been shown that the ability to accurately interpret audiovisual speech can support the detection of
 1506 manipulated media, such as deepfakes (Haliassos et al., 2021; 2022b).

1507 However, this technology is not without risks. Audiovisual speech recognition may be misused for
 1508 mass surveillance or covert monitoring through video feeds, particularly in contexts where individuals
 1509 have not consented to such analysis. This raises important questions around privacy, informed
 1510 consent, and the need for regulatory oversight. Additionally, as with most large-scale machine
 1511 learning systems, performance disparities may arise if models are trained on imbalanced datasets
 that underrepresent certain demographic groups. This could lead to inequitable outcomes, with

1512 worse recognition rates for users based on gender, age, accent, or ethnicity, which warrants careful
1513 consideration.

1514

1515

1516 LLM USAGE DECLARATION

1517

1518 In accordance with the ICLR 2026 policy on large language models (LLMs), we disclose that LLMs
1519 were used solely for polishing the language of this paper. Specifically, we employed an LLM to
1520 rephrase certain sentences for clarity and conciseness. No part of the paper's technical content,
1521 experimental design, analysis, or conclusions was generated by an LLM.

1522

1523

1524

1525

1526

1527

1528

1529

1530

1531

1532

1533

1534

1535

1536

1537

1538

1539

1540

1541

1542

1543

1544

1545

1546

1547

1548

1549

1550

1551

1552

1553

1554

1555

1556

1557

1558

1559

1560

1561

1562

1563

1564

1565