Enhancing Visual Speech Recognition by Integrating Language Models for Improved Lip Reading

Sofia Dominguez and Daria Kryvosheieva
Massachusetts Institute of Technology
Cambridge, MA, USA
domingc7@mit.edu, daria_k@mit.edu

Abstract

Lip reading, or visual speech recognition (VSR), is essential for enhancing communication in hearing-impaired individuals and improving speech comprehension in noisy environments [3, 17]. However, current VSR systems struggle with the inherent ambiguity of visually similar lip movements and the scarcity of large, high-quality training datasets. To address these issues, we propose enhancing the state-of-the-art model, Auto-AVSR [15], with two key modifications: (1) integrating an LLM into the decoding process to decrease ambiguity by utilizing context, and (2) fine-tuning on previously unused datasets to increase training diversity. With these changes, our approach aims to advance lip reading performance, which allows for more reliable VSR applications.

1. Introduction

Lip reading, also known as visual speech recognition (VSR), is the problem of understanding spoken words given a video recording of the speaker's lips. Applications of lip reading include communication enhancement for deaf and hearing-impaired individuals [3], speech comprehension improvement for people with normal hearing in noisy environments [17], and criminal investigation [22]. Current state-of-the-art AI systems for lip reading perform far from perfectly, making 10-20 errors for every 100 words, due to challenges such as inherent ambiguity of certain lip movements and limited training data.

Inherent ambiguity Distinct words that look identical on the lips are called *homophenes*. Estimates suggest that every English word has an average of 10 to 100 homophenes [21]. For example, in the homophene set *pat/bat/mat*, the voiceless/voiced/nasal distinction in the initial consonant is lost. Disambiguating homophenes requires leveraging context [13, 23].

Limited training data The sizes of curated lip reading datasets are measured in hundreds of hours—a very small amount compared to datasets for related tasks like speaker identification (thousands of hours) and audio-based speech recognition (tens of thousands of hours). Most of the work in sentence-level lip reading has focused on just two datasets: LRS2 [2] (225h of BBC programs) and LRS3 [1] (438h of TED talks). Some works tried to adapt datasets originally intended for other tasks [15] or scrape YouTube videos [4, 20], but these approaches make it difficult to ensure that the speaker's lips are always clearly visible.

We propose a method that builds upon the existing state-of-the-art architecture Auto-AVSR [15] by taking dedicated measures to resolve the two aforementioned issues. To address inherent ambiguity, we will integrate the Auto-AVSR pipeline with a language model, taking advantage of its ability to model context. To address insufficient training data, we will fine-tune the pre-trained Auto-AVSR on additional datasets.

2. Related Work

2.1. Current state of the art

Auto-AVSR [15] is presently the best-performing model on the LRS2 benchmark with a 14.6% word error rate (WER) and the third-best-performing model on LRS3 with a 19.1% WER. As shown in Figure 1, the architecture consists of a ResNet, a Conformer (convolution-augmented Transformer) encoder, a Transformer decoder, and a Connectionist Temporal Classification (CTC) projection layer. The ResNet module was pre-trained on LRW [5] (~160h of isolated words), while the entire pipeline was trained on a combination of LRS2, LRS3, and adaptations of AVSpeech (1323h) [9] and VoxCeleb2 (1307h) [6] auto-labelled with a speech recognition model.

The output of the model consists of two parts. The Cross Entropy (CE) loss corresponds to the auto-regressive part of the model, and it's trained by passing as arguments the

first n tokens of the label as well as the video, and trying to predict token $n+1$. The other part corresponds to the CTC loss, which takes the video as input and predicts a probability distribution of possible tokens at each time step, and uses dynamic programming to convert this into a distribution of possible transcripts.

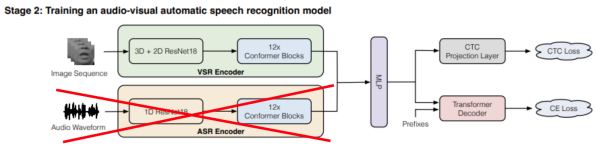

Figure 1. Auto-AVSR architecture. We do not use the audio encoder (crossed out) because we consider the visual-only variant of the lip reading task without audio cues.

Chang et al. [4] proposed a simpler architecture, consisting merely of a linear projection layer followed by a Conformer encoder and an LSTM decoder, and trained it on 100,000 hours of YouTube videos. The resulting model ranked first on LRS3 with a 12.8% WER.

2.2. Usage of language models

Ma et al. [14] integrated a custom-trained sequence-to-sequence language model (LM) into the decoding process by defining the beam search score to equal the weighted sum of the CTC score and the score under the LM. Prajwal et al. [19] did the same with an off-the-shelf GPT2 LM. Yeo et al. [23] developed a pipeline consisting of an AV-HuBERT-large visual encoder and a LLaMA-2-7B LM, where the LM is fine-tuned to map the intermediate output of the visual encoder to the final output.

3. Methodology

We make two primary modifications to the Auto-AVSR pipeline to improve performance: (1) integrating a large-scale language model (LM) during decoding, and (2) fine-tuning the model on new curated datasets that differ from the original training distribution. Training and evaluation are carried out on 1 A100 GPU provided by MIT's OpenMind computing cluster. Model quality is measured using the Word Error Rate (WER), defined as the total number of errors (insertions, deletions, and substitutions) divided by the number of words in the ground-truth transcript.

3.1. Language Model in Decoding Process

We introduce an LM into decoding similarly to [14]. We use **Gemma-3-12B** developed by Google. To our knowledge, this is the largest and newest LM used for lip reading so far. Larger [12] and newer [16] LMs have been shown to perform better across a wide variety of tasks, so we expect that this model's scale and novelty, unprecedented for the lip reading task, will ensure advanced context-modeling capability and ultimately lead to superior performance. A further advantage of Gemma-3-12B is free access via Hugging Face.

The decoding process in a VSR model tries to find the most likely sentence y given the input video x. In particular, we want

$$y^* = \arg\max_y p_\theta(y|x)\,,$$

where θ is the VSR model. However, calculating this would require searching over the space of all possible sentences, which is infeasible. For this reason, Auto-AVSR uses a process called beam search that can provide an approximate solution [10, 15].

In beam search, we iteratively construct a sentence y one token at a time. We start with an empty string y_0, and at every step we choose a token ϵ to append to the string. This is done greedily; we choose the tokens such that the conditional probability of the new string $p_\theta(y_0+\epsilon|x, y_0)$ is highest. To prevent the issues that may arise from using a greedy strategy, we keep a list of the k best hypotheses so far instead of just the best. This number k is called the beam size, which we set to 20 in all experiments.

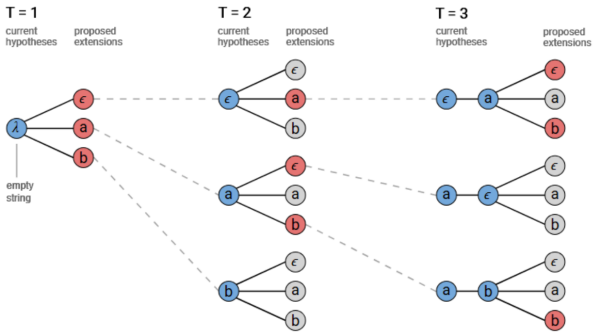

Figure 2. Beam search process with beam size 3. At each step, many extensions ϵ are proposed but only the best three are kept.

To introduce an LM into the decoding process, we modify the beam search so that instead of choosing tokens that maximize the probability $p_\theta(y_0+\epsilon|x, y_0)$, we choose those that maximize a weighted product

$$p_\theta(y_0+\epsilon|x, y_0)^{1-\alpha} \cdot p_\varphi(y_0+\epsilon|y_0)^{\alpha}\,, \qquad (1)$$

where φ is the LM, and $\alpha \in (0, 1)$ is a parameter that we will find through experiments.

This computation can be done efficiently because running an auto-regressive LM in a sentence y_0 will output a vector of probabilities $p_\varphi(y_0+\epsilon'|y_0)$ for every ϵ' in the LM's vocabulary. Since the LM's vocabulary and the VSR vocabulary are different, we decided to tokenize ϵ into LM tokens, and then replace the term $p_\varphi(y_0+\epsilon|y_0)$ with $\prod_{\epsilon'} p_\varphi(y_0+\epsilon'|y_0)$ in Eq. (1), where the product is done

over the LM tokens ϵ' that compose ϵ. This means that we only need to run the language model once on each step of the beam search.

In practice, log-probabilities are computed instead, and the term $\log p_\theta(y_0 + \epsilon | x, y_0)$ is split into a part that corresponds to CTC loss and a part that corresponds to CE loss. We choose a weight $\beta = 0.1$ for the CTC score and a weight $\gamma = 1 - \beta - \alpha$ for the CE score, and with this the final inference objective can be written as

$$\gamma \cdot \ell_{CE} + \beta \cdot \ell_{CTC} + \alpha \cdot \ell_{LM}$$

where ℓ_{CE} corresponds to the Cross Entropy score, and analogous for the other two terms.

3.2. Fine-Tuning on Curated Datasets

To improve generalization and reduce WER, we fine-tune Auto-AVSR on a curated train set built from approximately 75% of the videos in the TCD-TIMIT [11] and WildVSR [8] datasets, with the remaining 25% of each used for evaluation. The selection was randomized to ensure speaker diversity across splits, and fine-tuning was conducted for 10 epochs. To prevent catastrophic forgetting (the model forgetting its original training data), which is an issue that been observed in the literature since the 1980s [18], we choose a learning rate of $5 \cdot 10^{-6}$, 20 times smaller than the original learning rate. Furthermore, we freeze the ResNet18 encoder during training and only fine-tune the Conformer layers, the CTC projection, and the Transformer Decoder.

Figure 3. Sample image from a processed dataset. The VSR model is trained to predict text given a video of the lips of a person.

The datasets we use contain high-quality transcripts and controlled recording conditions, which means we will not need to deal with issues like invisibility of the speaker's lips, large distance of the speaker's face from the camera, or errors in transcripts. This allows the model to learn from cleaner supervision and more consistent lip motion-video alignments. TCD-TIMIT consists of high-quality video footage of 62 speakers reading a total of 6913 phonetically rich sentences, while WildVSR features visually challenging videos, often with partial occlusions, varied lighting, and natural head motion. Although it was made by closely following the LRS3 dataset creation processes, its data distribution is different, which can be seen by the fact that many publicly available VSR models perform poorly in this dataset [8]. These datasets provide a training distribution that is significantly different from LRS2 and LRS3, and by fine tuning the Auto-AVSR model on them we hope to improve the overall performance of the model.

We also considered using GRID [7], which consists of recordings of 1000 sentences spoken by 34 speakers. However, GRID sentences have limited vocabulary and follow a rigid templatic structure, which caused symptoms of severe overfitting: the model started reproducing the same structure even when tested on unrelated data.

To increase effective data coverage, we perform standard augmentations such as horizontal flipping, random cropping, and temporal jittering. These augmentations teach the model to be invariant to small shifts in lip motion and facial position. Together, TCD-TIMIT and WildVSR yield 43.4 hours of video; augmentations approximately double this amount.

Initial results show that fine-tuning without an LM already improves performance: WER dropped from 0.3918 (baseline model without any changes) to 0.3778. These results support our hypothesis that domain-specific fine-tuning on curated data can offer meaningful gains, even in the absence of additional architectural changes.

3.3. Domain-Adapting the Language Model

After initial experiments, we observed that incorporating the language model (LM) into the decoding process yielded no significant improvement in performance. To address this, we performed domain adaptation by fine-tuning the LM on the transcripts of the training videos.

The domain adaptation was carried out over 3 epochs with a learning rate of 10^{-4}, aligning the LM more closely with the distribution of our target data before its use in decoding.

3.4. Data Limitations and Evaluation Strategy

Due to access restrictions, we were unable to use LRS2 or LRS3—the two most common benchmarks in lip reading—for training or evaluation. LRS3 is no longer publicly available, and usage of LRS2 requires permission, which we could not obtain on time. Therefore, we used the 25% holdout portion of TCD-TIMIT and WildVSR as our test set. This still provides reliable information about relative model performance because we can still compare our model against the unmodified one, evaluating both on the same dataset.

For evaluation, we compute WER using standard alignment metrics that account for insertions, deletions, and substitutions. Beam size is fixed at 20 in all decoding runs. We evaluate both with and without LM integration to isolate the effects of each modification.

4. Experimental Results

We ran 8 experiments in total. The baseline uses the original Auto-AVSR model with no LM. Our best attempt uses the fine-tuned Auto-AVSR with no LM. Two more attempts use the fine-tuned Auto-AVSR with different weights of the LM, and the remaining four use the fine-tuned Auto-AVSR with the domain-adapted LM.

We found that fine-tuning the VSR model noticeably improves performance. Adding an LM, on the other hand, does not increase performance but does not significantly decrease it either: performance with both Auto-AVSR fine-tuning and LM is worse than performance with the Auto-AVSR fine-tuning and no LM, but still better than the baseline. The domain-adapted language model proves slightly better than the off-the-shelf one, but not significantly so.

Tables 1-3 and Figure 4 show the WER scores we obtained in different experimental scenarios.

Fine-tuning	WER
No	0.3918
Yes	**0.3778**

Table 1. Auto-AVSR without LM, with and without fine-tuning.

LM weight	WER
0.005	0.3781
0.01	0.3784

Table 2. Fine-tuned Auto-AVSR, LM without domain adaptation.

LM weight	WER
0.001	0.3780
0.005	0.3779
0.01	0.3785
0.05	0.3802

Table 3. Fine-tuned Auto-AVSR, LM with domain adaptation.

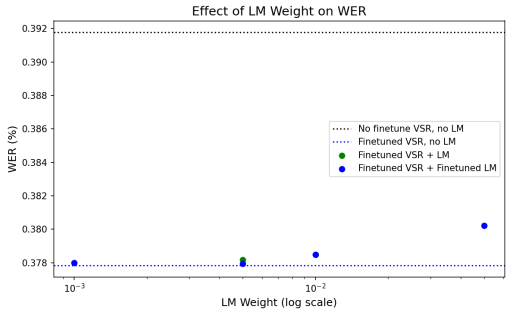

Figure 4. WER scores for all the experiments we conducted. The best model is the fine-tuned Auto-AVSR without LM. Adding a LM neither increases performance nor induces significant damage.

5. Discussion

Experiments showed that using a language model in the decoding process did not improve performance, and instead made the WER score increase slightly. This outcome suggests several possible explanations.

One contributing factor may be that the LM lacks access to the visual context of the input video. In the beam search process, we compute a weighted product involving $p_\varphi(y_0 + \epsilon \mid y_0)$, which represents the likelihood of a sentence continuation under the LM (Eq. (1)). However, this quantity is not conditioned on the video input, and as a result, it could favor linguistically sound sequences that are inconsistent with the visual features.

Related to the above, the term involving $p_\varphi(y_0 + \epsilon \mid y_0)$ represents the likelihood of the continuation as given by the training data of the LM, which generally consists of text from all the internet. This may not be ideal, since transcript is usually different from generic internet text.

As evidence for this hypothesis, we observed that fine tuning in the transcripts of the videos increased performance. This suggests that the model starts evaluating sentences conditioned that it's a transcript, instead of very diverse text from the internet. Further improvements could be expected by utilizing a model that is fine tuned in a larger or more representative dataset of transcripts.

Another possible reason is the way we are implementing the LM in the beam search. Eq. (1) is given in terms of a token ϵ in the VSR vocabulary. However, we are computing this probability by partitioning ϵ into LM tokens ϵ', and multiplying the probability of each token given the previous sentence y_0. This is not ideal since it assumes conditional independence of each part ϵ' given y_0, however, computing the true probability $p_\varphi(y_0 + \epsilon \mid y_0)$ would be computationally infeasible, as it would require to run the LM more than once at each step of the beam search. We note that our current implementation already takes around 1.5h to run on a single A100 GPU.

A potentially more effective approach would be to train a VSR model with the same vocabulary as the LM (or a subset of it), so that Eq. (1) can be calculated directly and there's no need to use approximations.

6. Individual Contributions

Sofia Dominguez: developed the code to integrate the LM into the decoding process, ran and debugged initial experiments with a small subset of data.

Daria Kryvosheieva: loaded and preprocessed datasets, ran fine-tuning and evaluation jobs on the OpenMind computing cluster.

Both team members contributed equally to writing the project report.

7. Conclusion

We presented two targeted modifications to the Auto-AVSR pipeline aimed at improving performance on the lip reading task: (1) integration of a large-scale language model into the decoding process, and (2) fine-tuning on curated datasets with high-quality annotations. Both strategies were motivated by key limitations in current VSR systems—namely, ambiguity in lip movements and insufficient labeled data.

Our experiments show that fine-tuning on diverse, controlled datasets such as TCD-TIMIT and WildVSR improves the model's generalization, reducing WER from 0.391 to 0.377. This confirms that even modest amounts of high-quality data can yield measurable gains when chosen carefully. Although the addition of the Gemma-3-12B language model did not significantly improve performance in isolation, it provides a flexible framework for future work and enables principled decoding under a probabilistic objective.

Overall, our findings demonstrate that strategic use of curated data and integration of modern language models can meaningfully enhance the performance of state-of-the-art lip reading systems, even without changes to the core architecture. Future work may focus on better alignment between VSR and LM tokenizations, dynamic adjustment of beam search parameters, and fine-tuning the language model jointly with the decoder for deeper integration.

References

[1] Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. Lrs3-ted: a large-scale dataset for visual speech recognition, 2018. 1

[2] Triantafyllos Afouras, Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman. Deep audio-visual speech recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):8717–8727, 2022. 1

[3] Lynne E. Bernstein, Nicole Jordan, Edward T. Auer, and Silvio P. Eberhardt. Lipreading: A review of its continuing importance for speech recognition with an acquired hearing loss and possibilities for effective training. *American Journal of Audiology*, 31(2):453–469, 2022. 1

[4] Oscar Chang, Hank Liao, Dmitriy Serdyuk, Ankit Shah, and Olivier Siohan. Conformer is all you need for visual speech recognition. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 10136–10140, 2024. 1, 2

[5] Joon Son Chung and Andrew Zisserman. Lip reading in the wild. In *Computer Vision – ACCV 2016*, pages 87–103, Cham, 2016. Springer International Publishing. 1

[6] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman. Voxceleb2: Deep speaker recognition. In *Interspeech 2018*, pages 1086–1090, 2018. 1

[7] Martin Cooke, Jon Barker, Stuart Cunningham, and Xu Shao. An audio-visual corpus for speech perception and automatic speech recognition. *The Journal of the Acoustical Society of America*, 120(5):2421–2424, 2006. _eprint: https://pubs.aip.org/asa/jasa/article-pdf/120/5/2421/13697743/2421_1_online.pdf. 3

[8] Yasser Abdelaziz Dahou Djilali, Sanath Narayan, Eustache Le Bihan, Haithem Boussaid, Ebtessam Almazrouei, and Merouane Debbah. Do vsr models generalize beyond lrs3? *arXiv preprint arXiv:2311.14063*, 2023. 3

[9] Ariel Ephrat, Inbar Mosseri, Oran Lang, Tali Dekel, Kevin Wilson, Avinatan Hassidim, William T. Freeman, and Michael Rubinstein. Looking to listen at the cocktail party: a speaker-independent audio-visual model for speech separation. *ACM Trans. Graph.*, 37(4), 2018. 1

[10] Awni Hannun. Sequence modeling with ctc. *Distill*, 2017. https://distill.pub/2017/ctc. 2

[11] Naomi Harte and Eoin Gillen. Tcd-timit: An audio-visual corpus of continuous speech. *IEEE Transactions on Multimedia*, 17(5):603–615, 2015. 3

[12] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models, 2020. 2

[13] Minsu Kim, Jeong Hun Yeo, and Yong Man Ro. Distinguishing homophenes using multi-head visual-audio memory for lip reading. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(1):1174–1182, 2022. 1

[14] Pingchuan Ma, Stavros Petridis, and Maja Pantic. Visual speech recognition for multiple languages in the wild. *Nature Machine Intelligence*, 4(11):930–939, 2022. 2

[15] Pingchuan Ma, Alexandros Haliassos, Adriana Fernandez-Lopez, Honglie Chen, Stavros Petridis, and Maja Pantic. Auto-avsr: Audio-visual speech recognition with automatic labels. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023. 1, 2

[16] Nestor Maslej, Loredana Fattorini, Raymond Perrault, Yolanda Gil, Vanessa Parli, Njenga Kariuki, Emily Capstick, Anka Reuel, Erik Brynjolfsson, John Etchemendy, Katrina Ligett, Terah Lyons, James Manyika, Juan Carlos Niebles, Yoav Shoham, Russell Wald, Tobi Walsh, Armin Hamrah, Lapo Santarlasci, Julia Betts Lotufo, Alexandra Rome, Andrew Shi, and Sukrut Oak. Artificial intelligence index report 2025, 2025. 2

[17] M. McClain, K. Brady, M. Brandstein, and T. Quatieri. Automated lip-reading for improved speech intelligibility. In *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages I–701, 2004. 1

[18] Michael McCloskey and Neal J. Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of Learning and Motivation*, pages 109–165. Academic Press, 1989. 3

[19] K R Prajwal, Triantafyllos Afouras, and Andrew Zisserman. Sub-word Level Lip Reading With Visual Attention . In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5152–5162, Los Alamitos, CA, USA, 2022. IEEE Computer Society. 2

[20] Dmitriy Serdyuk, Otavio Braga, and Olivier Siohan. Transformer-based video front-ends for audio-visual speech

recognition for single and muti-person video. In *Interspeech 2022*, pages 2833–2837, 2022. 1

[21] Nancy Tye-Murray, Mitchell Sommers, and Brent Spehar. Auditory and visual lexical neighborhoods in audiovisual speech perception. *Trends in Amplification*, 11(4):233–241, 2007. 1

[22] R v. Luttrell & Ors, 2004. EWCA Crim 1344. 1

[23] Jeonghun Yeo, Seunghee Han, Minsu Kim, and Yong Man Ro. Where visual speech meets language: VSP-LLM framework for efficient and context-aware visual speech processing. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 11391–11406, Miami, Florida, USA, 2024. Association for Computational Linguistics. 1, 2