Representing Speech Through Autoregressive Prediction of Cochlear Tokens

Greta Tuckute[1], *Klemen Kotar*[2], *Evelina Fedorenko*[1,4], *Daniel L. K. Yamins*[2,3]

[1]Department of Brain and Cognitive Sciences & McGovern Institute for Brain Research, MIT, USA
[2]Department of Computer Science & Wu Tsai Neurosciences Institute, Stanford University, USA
[3]Department of Psychology, Stanford University, USA
[4]Program in Speech and Hearing Bioscience and Technology, Harvard University, USA

gretatu@mit.edu, klemenk@stanford.edu, evelina9@mit.edu, yamins@stanford.edu

Abstract

We introduce **AuriStream**, a biologically inspired model for encoding speech via a two-stage framework inspired by the human auditory processing hierarchy. The first stage transforms raw audio into a time-frequency representation based on the human cochlea, from which we extract discrete **cochlear tokens**. The second stage applies an autoregressive sequence model over the cochlear tokens. AuriStream learns meaningful phoneme and word representations, and state-of-the-art lexical semantics. AuriStream shows competitive performance on diverse downstream SUPERB speech tasks. Complementing AuriStream's strong representational capabilities, it generates continuations of audio which can be visualized in a spectrogram space and decoded back into audio, providing insights into the model's predictions. In summary, we present a two-stage framework for speech representation learning to advance the development of more human-like models that efficiently handle a range of speech-based tasks[1]

Index Terms: speech perception, computational paralinguistics, human-inspired modeling

1. Introduction

Humans possess a remarkable ability to perform a wide range of tasks on speech inputs, from recognizing words in noise to separating speakers' voices and interpreting emotional tone. These processes are carried out by the human ear and networks of biological neurons. However, developing artificial neural networks that mirror the human ability to flexibly and efficiently understand and interact with the world through speech remains a significant challenge[1, 2, 3]. **To bridge this gap, we introduce AuriStream, a biologically-inspired model that learns versatile speech representations through a simple and scalable autoregressive prediction objective on a time-frequency representation inspired by the human cochlea**.

1.1. Related Work: Speech Representation Learning

Speech representation models, also known as audio encoders, broadly transform audio signals into discrete tokens or continuous embeddings for various downstream audio tasks [3]. **One popular approach is neural audio codecs**, which learn compressed representations by retaining the essential information for audio reconstruction, enabling them to recover the original signal from the learned codes [4, 5, 6, 7, 8, 9, 10, 11]. These audio codes can then serve as representation for downstream audio tasks [12, 2, 10]. However, although these models retain high-fidelity information about acoustic details due to the reconstruction objective, learning the appropriate acoustic invariances remains a challenge [2]. Further, high-fidelity signal reconstruction is unlikely to be a biologically plausible objective; instead, human speech perception mechanisms critically abstract away from the low-level acoustics and learn robust invariances over the audio signal [13, 14]. **A second popular approach is prediction-based modeling**, where models are trained to predict features derived from the raw waveform [15, 16, 17] or a time-frequency representation of audio [18, 19, 20, 21]. These prediction-based speech models broadly fall into two categories: autoregressive models, which predict future frames [18, 19, 20, 21], and masked prediction models, which predict masked frames from surrounding frames [22, 15, 16] (analogous to the causal and bi-directional prediction approaches in language modeling). The learned representations are then applied to various downstream audio tasks, for instance language modeling [15, 23, 24, 25]. One of the most widely used predictive models is HuBERT [15], which adapts the bi-directional BERT [26] objective for speech representation learning using self-generated *k*-means pseudolabels. **A third common approach is contrastive learning**, in which frames from different audio samples are pushed together or pulled apart in the embedding space according to a specified objective [27, 28, 29, 30]. One popular model is wav2vec2 [28] which contrasts masked-out audio segments from distractors in combination with an auxiliary objective. Although the contrastive approach can yield powerful representations, it requires heuristics to define positive and negative samples, implicitly enforcing which aspects of the audio signal are retained—potentially building in incorrect assumptions. Moreover, contrastive objectives often rely on directly contrasting embeddings across hundreds or thousands of samples simultaneously, which arguably is not a biologically plausible operation.

Although these three speech representation learning strategies are distinct, their objectives can be combined and augmented with additional heuristics. For instance, a state-of-the-art model, WavLM [17], combines the HuBERT prediction objective [15] with a noisy input transformation. However, as with most ensemble models, these performance gains come at the cost of additional hand-crafted complexity.

1.2. Our Approach: A Two-Stage Framework for Autoregressive Prediction on Biologically-Inspired Inputs

Unlike past approaches, our proposed framework does not rely on signal-reconstruction objectives (used by neural codec models), non-causal prediction objectives (used in bi-directional prediction models), or intra-batch contrasting of samples (used in many contrastive models). Instead, our framework takes inspiration from the human auditory processing hierarchy and op-

[1]Website and model weights:
https://tukoresearch.github.io/auristream-speech/

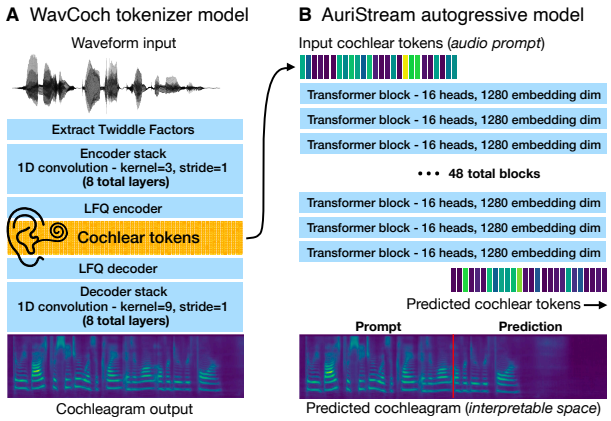

Figure 1: ***Schematic of speech representation framework***. *Description of the steps of the framework can be found in the Introduction (1.2) and Methods (2).*

erates in two stages:

The first stage is **WavCoch**, a model that transforms the raw audio into a time-frequency representation based on the human cochlea (Figure 1A). This approach bears some resemblance to neural audio codecs; however, instead of reconstructing the *same* signal, we predict a *different* audio representation–the time-frequency cochleagram–one known to be computed within the human auditory processing hierarchy [31, 32, 33, 34]. We read out the representations from an intermediate bottleneck stage of WavCoch which effectively discretizes the audio representations. We refer to these intermediate representations as **cochlear tokens** (Figure 1A).

The cochlear tokens serve as input to the second stage, **AuriStream**, which is an autoregressive sequence model, trained to predict the upcoming cochlear tokens (Figure 1B). Because the cochlear tokens were derived from a waveform-to-cochleagram transformation, these predicted tokens can naturally be decoded into the cochleagram, and then into audio, enabling inspection and interpretability.

In sum, we formulate speech representation learning as a simple yet powerful autoregressive prediction task over biologically-realistic inputs—cochlear tokens. Our framework yields representations from which phonemes, word forms, and word meanings (lexical semantics) can be decoded at competitive levels (Section 3.1), achieving state-of-the-art performance on lexical semantics. The learned representations also serve as a powerful backbone for various downstream SUPERB speech tasks [1] (Section 3.2). Finally, unlike comparison models, AuriStream *generates* continuations of audio that can be visualized in a cochleagram space, offering insights into the model's predictions (Section 3.3).

2. Methods

2.1. Input Tokens: WavCoch

We propose **WavCoch**, a model that efficiently tokenizes audio by transforming waveforms into cochleagrams, loosely mimicking the function of the human cochlea [31, 34]. The purpose of WavCoch is to extract discrete tokens from continuous audio signals to serve as the input to AuriStream. WavCoch is a causal encoder-decoder network with 8 encoder layers (1D convolution with kernel 3) and 8 decoder layers (1D convolution with kernel 9) with a total of 11.1M parameters (see Figure 7.1A). It takes as input 5s clips of mono audio waveforms sampled at 16kHz and is trained to predict the cochleagram representation of this audio clip. The target cochleagram [35, 36, 34] consists of 211 frequency bins and 988 temporal steps [34] (for comparison to mel-spectrograms, see Appendix 7.3). To obtain discrete representations, we place a 13-bit LFQ [37] bottleneck layer in the middle of the model, discretizing the embeddings into one of 8,192 units (corresponding to a 13-bit code, 2^{13}) denoted as **cochlear tokens**. We train WavCoch on the 960-hour LibriSpeech [38] dataset for 200k steps using the AdamW optimizer with a peak learning rate of 1e-4 with a 2,000-step warmup, and a cosine decay schedule. For further details and vocabulary size ablations, see Appendix 7.1 and 7.2.

2.2. Sequence Modeling: AuriStream

AuriStream is a GPT-style autoregressive Transformer [39] trained to predict the next cochlear token in a sequence (see Figure 7.1B). We train two versions: AuriStream-100M (100.7M parameters), with 12 layers, 12 attention heads and an embedding size of 784; and AuriStream-1B (970.1M parameters) with 48 layers, 16 attention heads, and an embedding size of 1,280. Both use SiLU activations [40], RMSNorms [41], and a vocabulary of 8,192 cochlear tokens. The AuriStream model takes as input the cochlear token sequence produced by WavCoch and predicts the next token in the sequence using a context window of 4,096 tokens (approximately 20s of speech). We utilize a learned positional embedding and compute the cross-entropy loss between the predicted logits and the true next token in the sequence. We train both AuriStream models on the 60k hour LibriLight [42] dataset for 500k steps using the AdamW optimizer with a peak learning rate of 3e-4 with a 2,000-step warmup, and a cosine decay schedule.

2.3. Evaluation Metrics

2.3.1. Phoneme/Word Linear Probing

To probe for phoneme and word identity representation, we use the TIMIT dataset [43] consisting of approximately five hours of audio recordings with ground truth phoneme- and word-boundaries. We use the train and complete test sets with exclusion of the "SA" sentences for train and test sets that are non-overlapping in sentences and speakers. For phoneme classification, we followed the standard protocol of collapsing the TIMIT phoneme labels from 60 to 39 classes [44]. We embed the audio clip up to and including the target phoneme/word jointly, then extract just the embeddings of the time bins corresponding to that phoneme/word for probing. We use the scikit-learn LogisticRegression multiclass classifier [45]. The reported values are weighted accuracy scores as the classes are imbalanced.

2.3.2. Lexical Semantic Similarity (sSIMI)

We use the "sSIMI" lexical semantics benchmark developed for the ZeroSpeech 2021 challenge [46]. The benchmark consists of pairs of words with ground truth human similarity judgments (on a 0 and 10 scale) collected from behavioral experiments. For instance, a pair of words such as "water" and "river" have a human similarity score of 9.8, while a pair like "festival" and "whiskers" have a score of 0.2. The benchmark contains two audio subsets: i) a natural subset with word pairs present in LibriSpeech [38], and ii) a synthetic subset with all pairs. The sSIMI score is computed as the Spearman correlation between the cosine distance of model embeddings for word pairs and the

true human similarity scores, multiplied by 100.

2.3.3. Obtaining Model Embeddings

We obtain model embeddings for phoneme/word probing (Section 2.3.1) and lexical similarity (Section 2.3.2) by pooling the embeddings of all the tokens associated with the corresponding temporal section of the audio via ground-truth phoneme or word boundaries. For the pooling operation, we tested mean/max/min pooling across the temporal dimension. To select the best layer for decoding, we evaluate the phoneme/word probing performance on a subset of the TIMIT set (the top 10 phonemes/words in the TIMIT test set). For the sSIMI benchmark, we select the best layer on the independent "dev" set.

2.3.4. Speech processing Universal PERformance Benchmark (SUPERB)

We evaluate AuriStream on the SUPERB benchmark which contains 15 tasks, categorized into five aspects of speech: content, speaker, semantics, paralinguistics, and generation. We report values on a subset of six tasks spanning all five categories. We refer to the original paper for additional details on the benchmark [1]. Scores for the comparison models were obtained from the SUPERB leaderboard.

3. Results

3.1. AuriStream Embeddings Contain Information about Phoneme Identity, Word Identity, and Lexical Semantics

To first assess whether AuriStream representations contain information about phoneme and word identity, we trained linear classifiers on the phonemes and words from the TIMIT train set [43] and evaluated the classifiers on the test set with non-overlapping sentences and speakers. We compared AuriStream to five state-of-the-art speech representation models (see details in Appendix 7.4). As shown in Table 1, for phoneme decoding, AuriStream-1B's performance was very close to state-of-the-art models HuBERT-xl and WavLM-large. The error patterns of AuriStream were sensible. For instance, the phoneme cluster "er" was often confused with "r", or "ah" with "ih" (see Appendix 7.5). For word decoding, AuriStream-1B surpassed wav2vec-large, however, AuriStream fell short of HuBERT and WavLM. We hypothesize that the subpar word decoding performance of AuriStream relative to these models is due to the fact that HuBERT and its derivative models (WavLM) were exposed to global clustering operations aimed at discovering word-like units. In contrast, AuriStream did not undergo any such global operations. Finally, we emphasize that decoding performance for both phonemes and words scales well with AuriStream size.

Second, we evaluate whether AuriStream learns representations of word meanings (lexical semantics). This benchmark (sSIMI) measures the correlation between embeddings of audio corresponding to pairs of words (e.g., "water" and "river") and human similarity judgments. Prior studies have described speech models' performance on this task as "modest" [46]. As shown in Table 2, both AuriStream-100M and AuriStream-1B outperform the other models on the natural and synthetic data subsets, and AuriStream-1B outperformed all other models on the synthetic set. Performance also improves with model scale. These findings demonstrate that a simple autoregressive prediction objective can lead to state-of-the-art representations for lexical semantics.

Table 1: ***Linear probing performance for phonemes or words on the TIMIT dataset***. *Reported values are weighted accuracy scores of the best layer (see Section 2.3.3) on the TIMIT test set with non-overlapping sentences uttered by non-overlapping speakers relative to the train set.*

Dataset	Params	Hours	Phoneme	Word
HuBERT-base	97M	1K	0.83	0.75
HuBERT-xl	1000M	60K	**0.91**	**0.85**
wav2vec2-large	317M	60K	0.76	0.43
WavLM-base	97M	1K	0.85	0.76
WavLM-large	317M	94K	0.90	0.84
AuriStream-100M	101M	60K	0.82	0.45
AuriStream-1B	970M	60K	0.88	0.65

Table 2: ***Semantic similarity scores on the ZeroSpeech 2021 Lexical Semantic Benchmark***. *Reported values are Spearman correlations (multiplied by 100 per [46]) of the best layer (see Section 2.3.3) between the embeddings for pairs of words and human similarity judgments. Scores are obtained on the test sets of two subsets: LibriSpeech Audio and Synthetic Audio.*

Dataset	Params	Hours	LibriSpeech ↑	Synthetic ↑
HuBERT-base	97M	1K	6.10	7.48
HuBERT-xl	1000M	60K	7.81	10.37
wav2vec2-large	317M	60K	6.41	7.19
WavLM-base	97M	1K	8.29	9.41
WavLM-large	317M	94K	10.50	10.37
AuriStream-100M	101M	60K	10.63	10.12
AuriStream-1B	970M	60K	**12.52**	**10.64**

3.2. AuriStream Serves as a Strong Backbone for Downstream Audio Tasks

Having established that AuriStream representations encode meaningful phoneme, word, and lexical semantics information, we investigated whether the frozen representations of AuriStream would serve as powerful features for training decoders across various audio tasks. To do so, we leveraged six tasks from the SUPERB benchmark, spanning all five major task categories defined in the benchmark [1]. As shown in Table 3, AuriStream-1B outperformed APC and vq-wav2vec—two models most similar to AuriStream—while performing competitively against state-of-the-art models on most tasks. In particular, AuriStream-1B showed strong performance on automatic speech recognition (ASR), intent classification (IC), and speech separation (SS). In contrast, AuriStream-1B had subpar performance on keyword spotting (KS) compared to other similarly sized models. Although WavLM-large—a model which contains many hand-designed heuristics such as noise addition during training and k-means clustering—dominates in all categories, AuriStream comes close to matching its performance on several tasks, demonstrating that it learns versatile representations for diverse downstream audio tasks. Importantly, AuriStream's favorable scaling behavior indicates strong potential for further improvements with more parameters and training data.

Table 3: ***Model performance on SUPERB tasks***. *Reported values are obtained by training a downstream task decoder on top of a frozen model backbone [1]. ASR = automatic speech recognition, IC = intent classification, KS = keyword spotting, SID = speaker identification, ER = emotion recognition, SS = speaker separation.*

Setting	ASR ↓	IC ↑	KS ↑	SID↑	ER ↑	SS ↑
HuBERT-base	6.42	98.34	96.30	81.42	64.92	9.36
HuBERT-large	3.62	98.76	95.29	90.33	67.62	10.45
wav2vec2-large	3.75	95.28	96.66	86.14	65.64	10.02
WavLM-base	6.21	98.42	96.79	84.51	65.94	10.37
WavLM-large	**3.44**	**99.31**	**97.86**	**95.49**	**70.62**	**11.19**
vq-wav2vec	17.71	85.68	93.38	38.80	58.24	8.16
APC	21.28	74.69	91.01	60.42	59.33	8.92
AuriStream-100M	7.80	92.00	93.96	79.10	59.32	9.05
AuriStream-1B	4.20	98.01	95.25	81.14	67.47	10.07

3.3. AuriStream Learns Short- and Long-Range Speech Statistics

In this final section, we leverage the fact that AuriStream was trained to perform predictions in a space that can be visualized and interpreted (the time-frequency cochleagram image) to ask whether it learns speech statistics without ground-truth phoneme, word, or task labels. We hypothesize that learned speech statistics should manifest in two distinct modes: *At short timescales*, when provided with sufficient context (such as the first part of a common word), the model should complete the cochleagram in a way that aligns with the remainder of the word. In contrast, *at longer timescales*, the model's predictions should diverge, reflecting the variability of plausible words that could follow any given phoneme or word. To test this hypothesis, we provided AuriStream with variable-length sequences of ground-truth audio clips from the TIMIT test set (out-of-distribution for AuriStream) and qualitatively analyzed the resulting model completions (Figure 2).

To first test whether AuriStream learns speech structure at short timescales, we prompted the model with the first phoneme of a common word (e.g., "she", starting with the phoneme "sh"), and evaluate predictions from this first phoneme across different speakers in the TIMIT test set. As shown in Figure 2A, the model learns to consistently complete the phoneme with an "iy" phoneme, resulting in the word "she". Conversely, when a phoneme has several likely continuations, the model learns to complete the phoneme with different words. For instance, when prompted with the initial phoneme cluster ("wa") of the words "water" and "wash" from two different speakers (Figure 2B), AuriStream sometimes predicts the remainder of the true word and other times generates a different completion consistent with the initial phoneme cluster. In one example, AuriStream's prediction for Speaker 2's utterance "wash" appears more similar to Speaker 1's ground-truth utterance of the word "water" than to its own ground-truth word ("wash") (Figure 2B), indicating that AuriStream learns to complete phoneme prompts with different plausible word continuations. These visualizations suggest that AuriStream learns the statistical regularities of how phonemes combine to form words, demonstrating knowledge of speech structure at short timescales.

Second, to evaluate the diversity of longer-range predictions, we prompted AuriStream with the first 2.5 seconds of TIMIT audio clips (Figure 2C). AuriStream predicts several

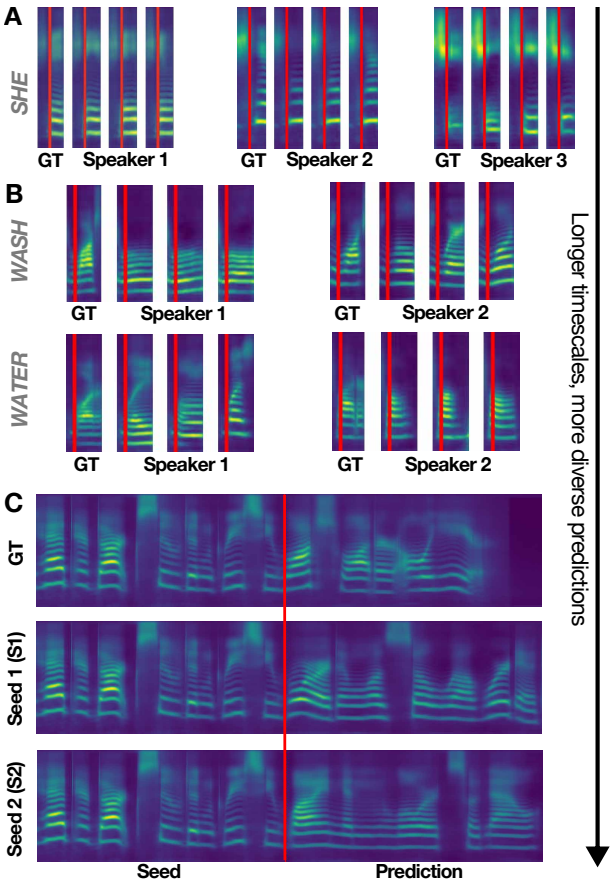

Figure 2: ***Cochleagram predictions by AuriStream-1B. A.*** *AuriStream-1B is prompted with the first phoneme of the word "she" (left of red vertical line) and predicts the word completion (right of red line) across three seeds. The ground-truth (GT) cochleagram is shown in the first column.* ***B.*** *AuriStream is prompted with the first phoneme of the words "wash" and "water".* ***C.*** *AuriStream is prompted with the first 2.5 seconds of an audio clip (red line) from the TIMIT test set and predicts the remaining part of the clip across two different seeds.*

seconds of plausible continuations as inspected visually, and audibly, since cochleagram representations can be inverted into audio. These continuations often sound very plausible given the topic of the prompt. We observe that the continuations degrade over time, however, we emphasize that the purpose of AuriStream is not to be a *language* model, but a *speech* representation model—the fact that it can perform rudimentary language modeling is a serendipitous side effect of the training objective, which points to the fact that learning patterns in speech, and producing language may be operationalized under a unified objective, albeit perhaps requiring additional mechanisms for enforcing longer-range coherence. Additional audio samples available at: https://tukoresearch.github.io/auristream-speech/ (also see details in Appendix 7.6).

4. Conclusion

We introduced AuriStream, a self-supervised speech representation model that achieves competitive phoneme and word decoding, state-of-the-art lexical semantic representations, and serves as a strong representational backbone for various audio

tasks. A key strength of our framework is the use of cochlear tokens: a biologically inspired and highly efficient token representation (around 200 tokens per second of audio) that fits within the context window of a standard Transformer, effectively leveraging the power of autoregressive modeling. While WavCoch is conceptually similar to neural codec approaches [4, 6, 8, 9, 10], its novelty lies in learning to transform one representation into *another* representation through a discrete quantization bottleneck (instead of auto-encoding, as done in related approaches)—an approach we denote as "Transformation Imitation". Finally, unlike prior speech representation models such as HuBERT and wav2vec2 [15, 28], AuriStream can also *generate* audio, enabling both embedding extraction and audio generation. In addition, AuriStream enables the visualization and interpretation of audio predictions through the cochleagram space, a capability that many audio models lack, making AuriStream less of a "black box".

Limitations of our work exist. One limitation is that AuriStream is trained on English speech, restricting analyses to tasks and materials in English [47, 48]. Another limitation is that AuriStream is trained exclusively on read speech from LibriLight, limiting ecological validity. Extending training to more naturalistic and developmentally plausible data [49, 50] is a key future direction. More broadly, although AuriStream is not a fully biologically realistic model, it constitutes a critical step in the right direction—and we hope that it will serve as a valuable model for the emerging field of "NeuroAI" which aims to understand biological and artificial intelligence by linking the representations and computations in artificial models to neural activity in the brain [51, 52, 53, 54, 55, 56, 57].

5. Acknowledgements

G.T. acknowledges support from The K. Lisa Yang ICoN Center and McGovern Institute for Brain Research. E.F. acknowledges support from McGovern Institute for Brain Research, the Department of Brain and Cognitive Sciences, MIT's Quest for Intelligence, and the Simons Foundation. K.K. and D.L.K.Y. acknowledge support from the Simons Foundation (543061), National Science Foundation (CAREER grant 1844724), Office of Naval Research (MURI S5847 and 1141386 - 493027). We thank the Stanford HAI, Stanford Data Sciences and the Marlowe team, and the Google TPU Research Cloud team for computing support.

6. References

[1] S.-w. Yang, P.-H. Chi, Y.-S. Chuang, C.-I. J. Lai, K. Lakhotia, Y. Y. Lin, A. T. Liu, J. Shi, X. Chang, G.-T. Lin *et al.*, "Superb: Speech processing universal performance benchmark," *arXiv preprint arXiv:2105.01051*, 2021.

[2] H. Wu, X. Chen, Y.-C. Lin, K. Chang, J. Du, K.-H. Lu, A. H. Liu, H.-L. Chung, Y.-K. Wu, D. Yang *et al.*, "Codec-superb@ slt 2024: A lightweight benchmark for neural audio codec models," *arXiv preprint arXiv:2409.14085*, 2024.

[3] A. Mehrish, N. Majumder, R. Bhardwaj, R. Mihalcea, and S. Poria, "A review of deep learning techniques for speech processing," 2023. [Online]. Available: https://arxiv.org/abs/2305.00359

[4] N. Zeghidour, A. Luebs, A. Omran, J. Skoglund, and M. Tagliasacchi, "Soundstream: An end-to-end neural audio codec," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 495–507, 2021.

[5] A. D'efossez, J. Copet, G. Synnaeve, and Y. Adi, "High fidelity neural audio compression," *ArXiv*, vol. abs/2210.13438, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:253097788

[6] D. Yang, S. Liu, R. Huang, J. Tian, C. Weng, and Y. Zou, "Hifi-codec: Group-residual vector quantization for high fidelity audio codec," *arXiv preprint arXiv:2305.02765*, 2023.

[7] R. Kumar, P. Seetharaman, A. Luebs, I. Kumar, and K. Kumar, "High-fidelity audio compression with improved rvqgan," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[8] S. Ji, M. Fang, Z. Jiang, R. Huang, J. Zuo, S. Wang, and Z. Zhao, "Language-codec: Reducing the gaps between discrete codec representation and speech language models," *arXiv preprint arXiv:2402.12208*, 2024.

[9] X. Zhang, D. Zhang, S. Li, Y. Zhou, and X. Qiu, "Speechtokenizer: Unified speech tokenizer for speech language models," in *The Twelfth International Conference on Learning Representations*, 2024.

[10] J. Kim, K. Lee, S. Chung, and J. Cho, "Clam-tts: Improving neural codec language model for zero-shot text-to-speech," *arXiv preprint arXiv:2404.02781*, 2024.

[11] R. Langman, A. Jukić, K. Dhawan, N. R. Koluguri, and B. Ginsburg, "Spectral codecs: Spectrogram-based audio codecs for high quality speech synthesis," *arXiv preprint arXiv:2406.05298*, 2024.

[12] C. Wang, S. Chen, Y. Wu, Z.-H. Zhang, L. Zhou, S. Liu, Z. Chen, Y. Liu, H. Wang, J. Li, L. He, S. Zhao, and F. Wei, "Neural codec language models are zero-shot text to speech synthesizers," *ArXiv*, vol. abs/2301.02111, 2023.

[13] K. Okada, F. Rong, J. Venezia, W. Matchin, I.-H. Hsieh, K. Saberi, J. T. Serences, and G. Hickok, "Hierarchical organization of human auditory cortex: evidence from acoustic invariance in the response to intelligible speech," *Cerebral Cortex (New York, N.Y.: 1991)*, vol. 20, no. 10, pp. 2486–2495, Oct. 2010.

[14] C. Tang, L. Hamilton, and E. Chang, "Intonational speech prosody encoding in the human auditory cortex," *Science*, vol. 357, no. 6353, pp. 797–801, 2017.

[15] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[16] Y.-A. Chung, Y. Zhang, W. Han, C.-C. Chiu, J. Qin, R. Pang, and Y. Wu, "W2v-BERT: Combining Contrastive Learning and Masked Language Modeling for Self-Supervised Speech Pre-Training," Sep. 2021, arXiv:2108.06209 [cs, eess]. [Online]. Available: http://arxiv.org/abs/2108.06209

[17] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao *et al.*, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.

[18] P. Michel, O. Räsänen, R. Thiolliere, and E. Dupoux, "Blind phoneme segmentation with temporal prediction errors," *arXiv preprint arXiv:1608.00508*, 2016.

[19] Y.-A. Chung, W.-N. Hsu, H. Tang, and J. Glass, "An unsupervised autoregressive model for speech representation learning," *arXiv preprint arXiv:1904.03240*, 2019.

[20] C. Shain and M. Elsner, "Acquiring language from speech by learning to remember and predict," in *Proceedings of the 24th Conference on Computational Natural Language Learning*, 2020, pp. 195–214.

[21] Y.-A. Chung, H. Tang, and J. Glass, "Vector-quantized autoregressive predictive coding," *arXiv preprint arXiv:2005.08392*, 2020.

[22] P.-H. Chi, P.-H. Chung, T.-H. Wu, C.-C. Hsieh, Y.-H. Chen, S.-W. Li, and H.-y. Lee, "Audio albert: A lite bert for self-supervised learning of audio representation," in *2021 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2021, pp. 344–350.

[23] K. Lakhotia, E. Kharitonov, W.-N. Hsu, Y. Adi, A. Polyak, B. Bolte, T.-A. Nguyen, J. Copet, A. Baevski, A. Mohamed *et al.*, "On generative spoken language modeling from raw audio," *Transactions of the Association for Computational Linguistics*, vol. 9, pp. 1336–1354, 2021.

[24] Z. Borsos, R. Marinier, D. Vincent, E. Kharitonov, O. Pietquin, M. Sharifi, D. Roblek, O. Teboul, D. Grangier, M. Tagliasacchi *et al.*, "Audiolm: a language modeling approach to audio generation," *IEEE/ACM transactions on audio, speech, and language processing*, vol. 31, pp. 2523–2533, 2023.

[25] G.-W. Wu, G.-T. Lin, S.-W. Li, and H.-y. Lee, "Improving textless spoken language understanding with discrete units as intermediate target," *arXiv preprint arXiv:2305.18096*, 2023.

[26] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proceedings of NAACL-HLT 2019*, Jun. 2019.

[27] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[28] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "Wav2vec 2.0: a framework for self-supervised learning of speech representations," in *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*, Dec. 2020.

[29] A. Saeed, D. Grangier, and N. Zeghidour, "Contrastive learning of general-purpose audio representations," *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3875–3879, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:224814216

[30] S. Bhati, J. Villalba, P. Żelasko, L. Moro-Velazquez, and N. Dehak, "Unsupervised speech segmentation and variable rate representation learning using segmental contrastive predictive coding," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 2002–2014, 2022.

[31] K. Wang and S. Shamma, "Self-normalization and noise-robustness in early auditory representations," *IEEE transactions on speech and audio processing*, vol. 2, no. 3, pp. 421–435, 1994.

[32] K. N. Ochsner and S. Kosslyn, *The Oxford Handbook of Cognitive Neuroscience, Volume 1: Core Topics*. Oxford University Press, 12 2013. [Online]. Available: https://doi.org/10.1093/oxfordhb/9780199988693.001.0001

[33] T. Chi, P. Ru, and S. A. Shamma, "Multiresolution spectrotemporal analysis of complex sounds," *The Journal of the Acoustical Society of America*, vol. 118, no. 2, pp. 887–906, Aug. 2005.

[34] J. Feather, G. Leclerc, A. Mądry, and J. H. McDermott, "Model metamers reveal divergent invariances between biological and artificial neural networks," *Nature Neuroscience*, vol. 26, no. 11, pp. 2017–2034, 2023.

[35] B. R. Glasberg and B. C. J. Moore, "Derivation of auditory filter shapes from notched-noise data," *Hearing Research*, vol. 47, no. 1, pp. 103–138, Aug. 1990. [Online]. Available: https://www.sciencedirect.com/science/article/pii/037859559090170T

[36] J. H. McDermott and E. P. Simoncelli, "Sound texture perception via statistics of the auditory periphery: evidence from sound synthesis," *Neuron*, vol. 71, no. 5, pp. 926–940, Sep. 2011.

[37] L. Yu, J. Lezama, N. B. Gundavarapu, L. Versari, K. Sohn, D. Minnen, Y. Cheng, V. Birodkar, A. Gupta, X. Gu, A. G. Hauptmann, B. Gong, M.-H. Yang, I. Essa, D. A. Ross, and L. Jiang, "Language model beats diffusion – tokenizer is key to visual generation," *ArXiv*, 2023, iCLR 2024. [Online]. Available: https://arxiv.org/abs/2310.05737

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

[39] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," 2018, technical report.

[40] D. Hendrycks and K. Gimpel, "Gaussian error linear units (gelus)," *ArXiv*, 2016. [Online]. Available: https://arxiv.org/abs/1606.08415

[41] B. Zhang and R. Sennrich, "Root mean square layer normalization," *ArXiv*, 2019, neurIPS 2019. [Online]. Available: https://arxiv.org/abs/1910.07467

[42] J. Kahn, M. Rivière, W. Zheng, E. Kharitonov, Q. Xu, P.-E. Mazar'e, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen, T. Likhomanenko, G. Synnaeve, A. Joulin, A. rahman Mohamed, and E. Dupoux, "Libri-light: A benchmark for asr with limited or no supervision," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7669–7673, 2019.

[43] J. S. Garofolo, "Timit acoustic phonetic continuous speech corpus," *Linguistic Data Consortium, 1993*, 1993.

[44] K.-F. Lee and H.-W. Hon, "Speaker-independent phone recognition using hidden markov models," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 11, pp. 1641–1648, 1989.

[45] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, A. Müller, J. Nothman, G. Louppe, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[46] T. A. Nguyen, M. de Seyssel, P. Rozé, M. Rivière, E. Kharitonov, A. Baevski, E. Dunbar, and E. Dupoux, "The zero resource speech benchmark 2021: Metrics and baselines for unsupervised spoken language modeling," *arXiv preprint arXiv:2011.11588*, 2020.

[47] D. E. Blasi, J. Henrich, E. Adamou, D. Kemmerer, and A. Majid, "Over-reliance on english hinders cognitive science," *Trends in cognitive sciences*, vol. 26, no. 12, pp. 1153–1170, 2022.

[48] S. Feng, B. M. Halpern, O. Kudina, and O. Scharenborg, "Towards inclusive automatic speech recognition," *Computer Speech & Language*, vol. 84, p. 101567, 2024.

[49] J. Sullivan, M. Mei, A. Perfors, E. Wojcik, and M. C. Frank, "Saycam: A large, longitudinal audiovisual dataset recorded from the infant's perspective," *Open mind*, vol. 5, pp. 20–29, 2021.

[50] A. Warstadt, L. Choshen, A. Mueller, A. Williams, E. Wilcox, and C. Zhuang, "Call for Papers – The BabyLM Challenge: Sample-efficient pretraining on a developmentally plausible corpus," *Arxiv*, 2023, publisher: arXiv Version Number: 1. [Online]. Available: https://arxiv.org/abs/2301.11796

[51] A. J. E. Kell, D. L. Yamins, E. N. Shook, S. V. Norman-Haignere, and J. H. McDermott, "A task-optimized neural network replicates human auditory behavior, predicts brain responses, and reveals a cortical processing hierarchy," *Neuron*, vol. 98, no. 3, pp. 630–644.e16, May 2018.

[52] J. Millet, C. Caucheteux, P. Orhan, Y. Boubenec, A. Gramfort, E. Dunbar, C. Pallier, and J.-R. King, "Toward a realistic model of speech processing in the brain with self-supervised learning," in *Advances in Neural Information Processing Systems 35 (NeurIPS 2022)*, Jun. 2022.

[53] Y. Li, G. K. Anumanchipalli, A. Mohamed, J. Lu, J. Wu, and E. F. Chang, "Dissecting neural computations of the human auditory pathway using deep neural networks for speech," *bioRxiv*, 2022, publisher: Cold Spring Harbor Laboratory.

[54] G. Tuckute, J. Feather, D. Boebinger, and J. H. McDermott, "Many but not all deep neural network audio models capture brain responses and exhibit correspondence between model stages and brain regions," *Plos Biology*, vol. 21, no. 12, p. e3002366, 2023.

[55] S. R. Oota, V. Agarwal, M. Marreddy, M. Gupta, and R. S. Bapi, "Speech taskonomy: Which speech tasks are the most predictive of fmri brain activity?" in *INTERSPEECH 2023-24th INTERSPEECH Conference*, 2023, pp. 5167–5171.

[56] G. Tuckute, N. Kanwisher, and E. Fedorenko, "Language in brains, minds, and machines," *Annual Review of Neuroscience*, vol. 47, no. 2024, pp. 277–301, 2024.

[57] O. Moussa and M. Toneva, "Brain-tuned speech models better reflect speech processing stages in the brain," *arXiv preprint arXiv:2506.03832*, 2025.

[58] J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex fourier series," *Mathematics of Computation*, vol. 19, pp. 297–301, 1965. [Online]. Available: https://api.semanticscholar.org/CorpusID:121744946

[59] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2015, pp. 5206–5210, iSSN: 2379-190X.

7. Appendix

7.1. WavCoch Architecture Details

As shown in Figure 1A, the raw waveform (shape: $1 \times 80{,}000$ for 5s of mono audio sampled at 16kHz) is first transformed into the time–frequency domain via a fixed-kernel discrete Fourier transform implemented as a bank of 1D convolutional filters (window size 1,001 samples, hop length 80 samples). The filter weights—the complex sinusoidal basis functions (or Twiddle Factors [58]) of the discrete Fourier transform—slide over the signal to produce a spectral representation with one feature vector every 5ms. Second, each 5ms temporal step of this frequency representation is passed through an 8-layer encoder stack (each layer is a 1D convolution with 512 channels, kernel size 3, stride 1, ReLU nonlinearities), yielding a sequence of 512-dimensional embeddings. Third, these embeddings are then passed through a 13-dimensional LFQ bottleneck [37], which effectively binarizes the representation. We read out the activations of this bottleneck as a 13-bit binary code which can be interpreted as one of $2^{13} = 8,192$ discrete tokens. We determined that 13-bits is the optimal vocabulary size by ablating vocabulary sizes and evaluating out-of-distribution performance on cochleagram reconstruction error and phoneme cluster purity; 12-bit and 14-bit codes yielded inferior performance (see full ablation details in Appendix 7.2). Fourth, the output of the LFQ bottleneck is passed through a decoder stack (each layer is a 1D convolution with 211 channels, kernel size 9, stride 1, ReLU nonlinearities). This decoder output corresponds to the frequencies in the cochleagram representation [34], which the model is supervised to match via L2 error. An auxiliary entropy penalty with a weight of 0.001 is applied at the LFQ bottleneck to encourage diversity, in line with [37]. Thus, for every 5 seconds of audio, WavCoch extracts a sequence of 988 integers in the range [0, 8192) through the LFQ bottleneck, denoted as cochlear tokens, to feed into AuriStream (illustrated in Figure 1B).

7.2. WavCoch Vocabulary Size Ablations

We performed ablations to identify the optimal vocabulary size of the WavCoch model. We trained variants of WavCoch using a vocabulary size of 4,096, 8,192, and 16,384 (12-, 13- and 14-bit codes, respectively) on the LibriSpeech960 dataset [59]. For each of these models, we evaluated the cochleagram reconstruction L2 error and phoneme cluster purity on an out-of-distribution test set (TIMIT test set [43]). Phoneme cluster purity was defined as *purity = (count of most associated phoneme for token i) / (total counts for token i)* providing an intuitive metric for how consistently a given token aligns with a specific phoneme. Figure I shows that a vocabulary size of 8,192 (13-bit code) yields both the lowest reconstruction error and the highest phoneme cluster purity.

7.3. WavCoch Target Representations: Cochleagram vs. Mel Spectrogram

To evaluate the impact of using the biologically-inspired cochleagram representation [31, 34] as the WavCoch prediction target as opposed to the more standard deep learning practice of using a mel-spectrogram, we trained a version of WavCoch using mel-spectrograms (80 mel bins and 5ms temporal bins) as prediction targets. Both cochleagram- and mel-based WavCoch models were trained on the publicly available LibriSpeech960 dataset [59], consisting of 960 hours of speech recordings. Since the L2 reconstruction error is not directly comparable between a cochleagram and a mel-spectrogram, we investigated two proxy measures of representational quality: i) The number of unique codes utilized in the quantized representation ("codebook usage"), and ii) Phoneme cluster purity (defined as *purity = (count of most associated phoneme for token i) / (total counts for token i)*) Both metrics were computed on the out-of-distribution TIMIT test set [43] and are reported in Table I.

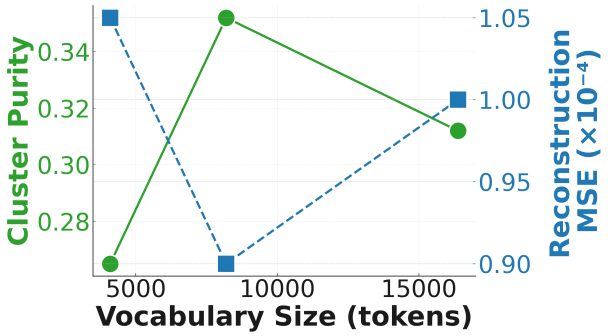

Figure I: ***Evaluation of WavCoch trained with different vocabulary sizes.*** *We plot the L2 cochleagram reconstruction error (blue) and the phoneme cluster purity (green) on the out-of-distribution TIMIT test set.*

Table I: ***Evaluation of WavCoch trained with different prediction targets.*** *Codebook usage and phoneme cluster purity evaluated on the out-of-distribution TIMIT test set.*

Target	Codebook Usage ↑	Cluster Purity ↑
Cochleagram	**8,172**	**0.3517**
Mel-Spectrogram	8,151	0.3473

First, in terms of codebook usage, we found that the WavCoch model trained with the cochleagram target utilized slightly more codes than the model trained with the mel-spectrogram target to represent out-of-distribution speech data (TIMIT test [43]). Second, the cochleagram-based WavCoch model achieved a slightly higher average phoneme cluster purity on the TIMIT test set than the mel-spectrogram model. While these differences are relatively small, they suggest that the cochleagram representation performs at least as well as, if not slightly better than the mel-spectrogram in this setting.

Beyond the quantitative analyses reported in Table I, we prefer the cochleagram over the mel-spectrogram representation for conceptual reasons: The ultimate goal of our framework is to move towards more biologically plausible speech models, and the cochleagram is more aligned with this goal.

7.4. Comparison Models

AuriStream is compared to five state-of-the-art speech representation models using the HuggingFace Transformers package: HuBERT-base (identifier: *facebook/hubert-base-ls960*), HuBERT-xl (identifier: *facebook/hubert-xlarge-ll60k*), wav2vec2-large (identifier: *facebook/wav2vec2-large*), WavLM-base (identifier: *microsoft/wavlm-base*), and WavLM-large (identifier: *microsoft/wavlm-large*). For the SUPERB benchmark, we additionally compare against two smaller models which share some similarity to AuriStream, specifically,

APC and vq-wav2vec.

7.5. Confusion Matrix for Phoneme Decoding

Figure II shows the phoneme confusion matrix for AuriStream-1B in the linear decoding task (see Section 3.1). The error patterns were sensible: for instance, "er" was often confused with "r", or "ah" with "ih".

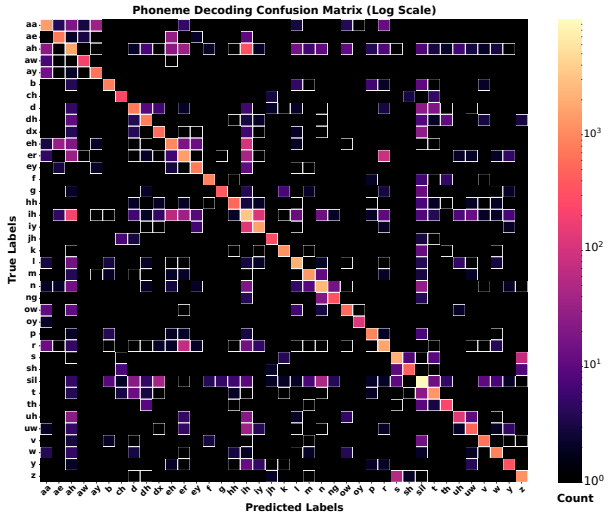

Figure II: ***Confusion matrix for phoneme decoding.*** *The plot shows which phonemes were confused with each other from the AuriStream-1B model on the TIMIT test set. The plot is shown on a log colorscale to better highlight the mismatches between true and predicted labels.*

7.6. Sonifying AuriStream Predictions through Cochleagram Inversion

We investigate AuriStream's predictions by inverting the cochleagrams into audible waveforms. To this end, we developed a simple per-sample optimization procedure that constructs a waveform that matches the cochleagram prediction. Specifically, we optimize a tensor of shape $(1 \times 80{,}000)$—initialized with random numbers from a normal distribution with mean 0 and variance 1—representing the waveform input to make its cochleagram representation match the cochleagram predicted by WavCoch (via L2 error). We backpropagate through the cochleagram transformation and use the Adam optimizer with a learning rate of 1e-2. Note that this optimization procedure is not a learned vocoder model, but a simple procedure which converts the output of WavCoch, the cochleagrams, into audible sound (conceptually similar to Griffin-Lim algorithm).

Several audible samples of speech generations from AuriStream-1B are available at the following link: https://tukoresearch.github.io/auristream-speech/. Please access the page using Google Chrome as we have seen some cases in which Safari and Firefox are not properly loading these videos.

We observed that on short timescales, the model produces reasonable completions, but the longer the completion, the more the predictions drift away from being plausible. We want to emphasize that the purpose of AuriStream is not to be a language model, but a speech representation model—the fact that it can perform rudimentary language modeling is a serendipitous side effect of the training objective, which points to the fact that learning patterns in speech, and producing language may be operationalized under a unified objective. These findings serve as great motivating factors for follow-up work, which will attempt to stabilize speech generations with longer-term coherence, building on the foundation laid out in this paper.