SpidR: Learning Fast and Stable Linguistic Units for Spoken Language Models Without Supervision

Anonymous authors
Paper under double-blind review

Abstract

We introduce SpidR, a self-supervised speech representation model that efficiently learns strong representations for spoken language modeling. It is trained on unlabelled speech using a masked prediction objective combined with self-distillation and online clustering. The intermediate layers of the student model learn to predict assignments derived from the teacher intermediate layers. This learning objective stabilizes the online clustering procedure compared to previous approaches, resulting in higher-quality codebooks. SpidR outperforms previous state-of-the-art methods on downstream language modeling metrics while significantly reducing pretraining time, requiring only a day to pretrain with 16 GPUs instead of a week. We will open-source the training code and model checkpoints upon acceptance.

1 Introduction

Self-supervised learning (SSL) has revolutionized speech processing (Mohamed et al., 2022). Speech representations learned without supervision have gained traction in speech and audio processing, for two broad classes of downstream applications. In the first class, SSL is applied with large corpora of unlabelled speech, and the resulting representations are used as is or fine-tuned for specific classification or recognition tasks (e.g., automatic speech recognition, phoneme classification, or speaker identification). In the second class, representations learned with SSL are interpreted as a proxy for linguistic units, and are used to train *pure* spoken language models directly from audio for the purpose of generation (Arora et al., 2025).

Early research on spoken language modeling (SLM) (Lakhotia et al., 2021; Dunbar et al., 2021) based on automatically discovered discrete units has shown the best performance was obtained when these units were extracted from speech encoders for which phonetic information was the most easily accessible, as measured by phoneme classification of a linear probe or by computing the ABX discriminability of the representations (Schatz, 2016; Schatz et al., 2013). HuBERT (Hsu et al., 2021) was then the leading open model, and subsequent studies have therefore used it (Polyak et al., 2021; Kharitonov et al., 2022; Kreuk et al., 2022; Nguyen et al., 2023). Despite its transformative potential for speech applications, *pure* SLM (spoken language modeling from raw audio without any text) remains quite understudied, and spoken language models are far from being as advanced as text-based language models. These models still lag behind their text-based counterparts in terms of capturing semantics when trained with similar data quantity (Nguyen et al., 2020), with scaling laws up to three orders of magnitude slower (Cuervo & Marxer, 2024). To tackle this issue, recent efforts have focused more on bridging the gap between speech and text modalities, transferring the knowledge of large language models to speech (Hassid et al., 2023; Nguyen et al., 2025; Défossez et al., 2024; Cuervo et al., 2025), and achieving more favorable scaling laws (Maimon et al., 2025b).

This line of work has focused on improving the language model itself, but the performance gap between speech- and text-based language models also stems from a more fundamental issue: current speech units are less efficient for language modeling than text-derived units such as BPE tokens or even phonemic transcriptions. Speech units capture phonetic information (Choi et al., 2024), but they lack robustness to both acoustic variations (Gat et al., 2023) and contextual variations caused by coarticulation (Hallap et al., 2023). As a result, these units align more closely with contextual phone states (Young et al., 1994) than with actual

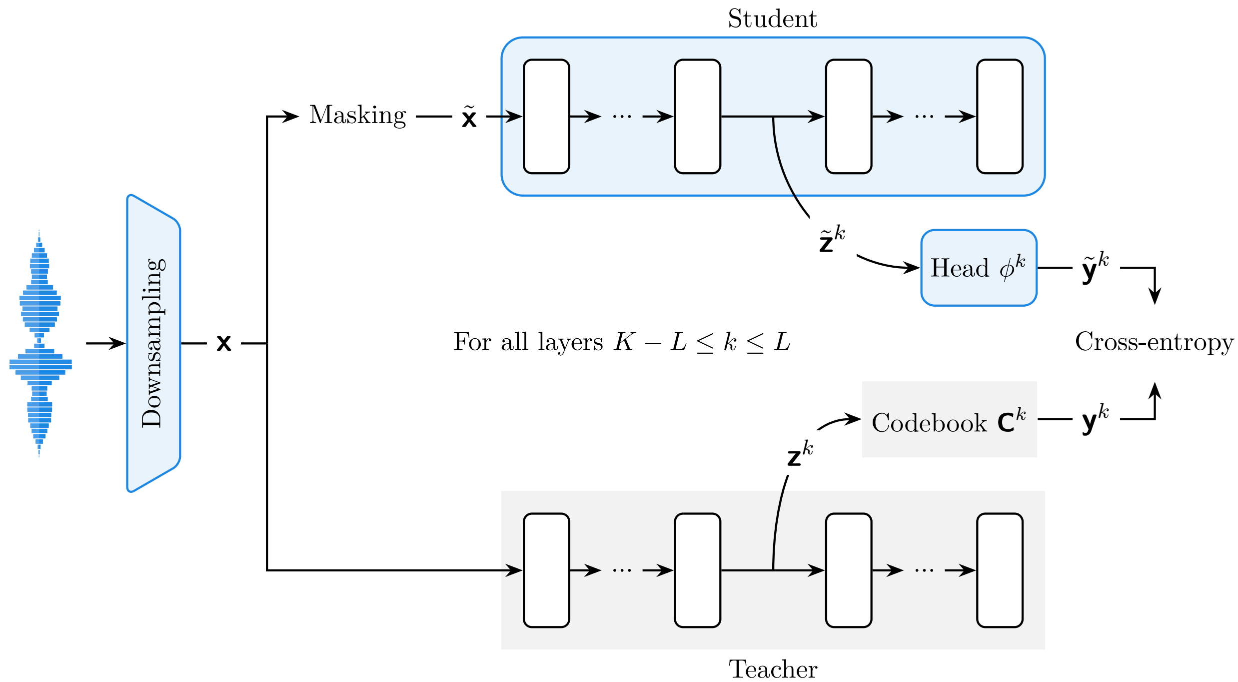


Figure 1: Architecture of SpidR. The downsampling module, a stack of convolutional layers, transforms the speech waveform into 20ms frames. The student and teacher are Transformers with $L = 12$ layers. For every layer k in the last $K = 8$ ones, the student predicts—through a prediction head ϕ^k—the nearest neighbor codebook assignment on the masked positions from the teacher at the same layer. The downsampling module, the student and the predictions heads are updated by gradient descent (in blue). The teacher is an exponential moving average (EMA) of the student, and the codebooks are updated with an EMA of the embeddings of the teacher (in gray).

linguistic units (Dunbar et al., 2022). To address these limitations, some approaches added a new step on the units derived from SSL models to make them larger and closer to syllables or words (Algayres et al., 2023; Baade et al., 2025; Visser & Kamper, 2025). However, training new SSL models from scratch is costly. For example, HuBERT is trained in several passes, demanding terabytes of storage to store the extracted features (Zanon Boito et al., 2024). It requires alternating between model training and clustering, with some manual decision to be made between each pass to select the intermediate layer used to compute the targets. DinoSR (Liu et al., 2023) is a recent single-pass alternative to HuBERT that demonstrates better phonetic discriminability, making it a suitable candidate for spoken language modeling. Its original implementation still requires a week of training, which limits the exploration of its training properties.

In this work, we introduce SpidR, a novel self-supervised method for speech representation that learns strong representations for spoken language modeling in a single pass. While inspired from DinoSR's architecture, SpidR's learning objective makes pretraining significantly more stable and resistant to codebook collapse. Our approach incorporates self-distillation and online clustering with pseudo-labels derived from codebooks at the intermediate layers of the teacher encoder. However, it differs from DinoSR in a key way. Instead of using only the student's final layer to predict the assignments for each teacher intermediate layer, we use the student's own intermediate representations. Our experimental results demonstrate that SpidR outperforms both HuBERT and DinoSR on zero-shot spoken language modeling metrics. We also release a minimal pure PyTorch codebase for training DinoSR or SpidR, using the latest advancements of PyTorch. With our implementation, full pretraining requires only one day of 16 GPUs—a substantial efficiency improvement over previous approaches that enables faster iteration.

2 Related Work

Self-supervised speech representation learning. Self-supervised learning for speech evolved from early autoregressive models (van den Oord et al., 2019; Schneider et al., 2019; Chung & Glass, 2020) to predominantly bidirectional masked prediction approaches (Devlin et al., 2019) that leverage surrounding unmasked context. The wav2vec 2.0 (Baevski et al., 2020) model is trained with contrastive learning between the contextual representations and quantized units. Its architecture also established a standard backbone that has been adopted by most subsequent approaches, the key differentiation between models lying in how they compute the self-supervised loss and derive training targets. HuBERT (Hsu et al., 2021) introduced an iterative approach where pseudo-targets are obtained from a previous iteration of the model, alternating between clustering and pretraining. Baevski et al. (2022) use self-distillation to derive the targets, an approach followed by DinoSR (Liu et al., 2023) but with discrete targets instead of continuous embeddings. Our work builds directly on DinoSR, maintaining the established architecture while focusing specifically on improving the stability of the learning objective rather than architectural innovations. The pseudo-targets in data2vec, DinoSR and our work are derived from intermediate layers, an approach also explored by Chung et al. (2021); Wang et al. (2022). Another line of research has focused on enhancing robustness through additional training objectives—addressing acoustic (Chen et al., 2022; Chang & Glass, 2024) or speaker variations (Chang et al., 2023). Particularly relevant to our goals, Chang et al. (2024) fine-tune HuBERT to learn codebooks optimized for spoken language modeling. We focus in this work on single-pass pretraining without additional fine-tuning steps, making our approach complementary to these specialized adaptation methods.

Efficient speech representation learning. With the increasing cost to train self-supervised speech models, researchers have explored various approaches to simplify the training procedure and accelerate training time. For instance, Baevski et al. (2023) improve the sample efficiency of data2vec by training with multiple masked versions of the same sample. For HuBERT specifically, several efficiency improvements have been proposed: Lin et al. (2023) and Yang et al. (2023) replace the learned downsampling module by mel-filterbanks and use a cross-entropy loss, while Chen et al. (2023a) use an existing ASR model to extract the targets for the first training iteration instead of MFCC features. Yang et al. (2025) take a different approach by replacing the encoder with a Zipformer (Yao et al., 2024). It's worth noting that training HuBERT also requires terabytes of available storage to save the high-dimensional embeddings extracted between each iteration (Zanon Boito et al., 2024)—a inherant limitation that architectural changes alone cannot address. Additionally, most models derived from wav2vec 2.0, including HuBERT, were originally pretrained using the fairseq library (Ott et al., 2019). While fairseq initially provided essential solutions for distributed training, mixed precision, etc., these features now exist natively in PyTorch (Ansel et al., 2024), and fairseq is no longer maintained. Our streamlined PyTorch-native implementation of SpidR and DinoSR reduces compute requirements, enables faster iteration during development, and provides a hackable foundation for future research without legacy dependencies.

Spoken Language Modeling. Generative text pretraining has inspired a new family of speech generation models. By proposing to quantize self-supervised representations, Lakhotia et al. (2021) rephrased speech generation as a language modeling task. The discrete tokens function as phonetic units, due to their accessible phonetic information (Nguyen et al., 2022; Sicherman & Adi, 2023; Yeh & Tang, 2024), and serve as inputs to train a Transformer decoder. Borsos et al. (2023) combined these units with audio codec tokens (Zeghidour et al., 2022; Défossez et al., 2023) to capture finer acoustic details. Non-phonetic information has also been incorporated with phonetic units to capture style or prosody (Kharitonov et al., 2022; Nguyen et al., 2025). Chen et al. (2025); Zhang et al. (2024) and Défossez et al. (2024) only use units from audio codecs, with the latter two training their codecs with distillation from a SSL model. Despite their ability to learn linguistic structures (Dunbar et al., 2021), purely speech-based models have exhibited limited factual knowledge and reasoning abilities. This prompted the development of hybrid speech-text models (Hassid et al., 2023; Nguyen et al., 2025; Défossez et al., 2024; Cuervo et al., 2025). A parallel research direction focuses on improving the speech units themselves (Algayres et al., 2023; Baade et al., 2025), as speech representations with more accessible phonetic information significantly improves linguistic knowledge (Poli et al., 2024). In our work, we deliberately focus on pure spoken language modeling from raw audio to isolate and evaluate the specific contributions of our speech encoder.

3 Method

As illustrated in figure 1, SpidR leverages self-distillation and online clustering, making predictions at multiple layers of the network. It is based on DinoSR, but with a novel learning objective. The student layers directly predict the assignment given by the corresponding codebook, instead of having multiple prediction heads at the end of the student encoder, which results in more stable training runs.

We first extract feature frames $\mathbf{x} = (\boldsymbol{x}_1, ..., \boldsymbol{x}_n)$ from a speech utterance, with $\boldsymbol{x}_i \in \mathbb{R}^d$, using a convolutional block. We sample a random mask $M \subset \{1, ..., n\}$, with the sampling procedure from Baevski et al. (2020), and build $\tilde{\mathbf{x}}$, a corrupted version of $\mathbf{x}$ where for each $i \in M$, $\boldsymbol{x}_i$ has been replaced by a learned mask embedding. The student encoder is a Transformer (Vaswani et al., 2017) with L layers, trained to predict the pseudo-labels derived from a teacher at the masked positions. Let $\tilde{\mathbf{z}}^k = (\boldsymbol{z}_1^k, ..., \boldsymbol{z}_n^k)$ be the output of the student encoder at layer k from $\tilde{\mathbf{x}}$. As in previous works, this is the output of the feed-forward network in the Transformer block, before the final residual connection and layer normalization. The prediction is done at the last K layers of the encoder. The labels prediction at frame i and intermediate layer k is

$$\tilde{\boldsymbol{y}}_i^k = \phi^k(\tilde{\boldsymbol{z}}_i^k) \in (0,1)^V, \tag{1}$$

where ϕ^k is the prediction head at layer k, with $L - K \leq k \leq L$, and V is the number of labels. The prediction head is made of a single linear projection followed by a softmax. To derive the pseudo-labels, we first feed the unmasked frames $\mathbf{x}$ to the teacher. Let $\mathbf{z}^k$ be the output of the teacher encoder at intermediate layer k after instance normalization. The one-hot target label at frame i and layer k is

$$\boldsymbol{y}_i^k \in \{0,1\}^V \text{ where for } 1 \leq v \leq V, (\boldsymbol{y}_i^k)_v = \begin{cases} 1 & \text{if } v = \arg\min_{1 \leq u \leq V} \|\boldsymbol{z}_i^k - \mathbf{C}_u^k\|_2 \\ 0 & \text{otherwise} \end{cases}, \tag{2}$$

where $\mathbf{C}^k$ is the codebook associated to layer k, with V codewords. The model is trained to predict the target labels from the teacher on the masked positions by minimizing the cross-entropy

$$-\frac{1}{|M| \cdot K} \sum_{\substack{i \in M \\ L-K \leq k \leq L}} \boldsymbol{y}_i^k \log \tilde{\boldsymbol{y}}_i^k, \tag{3}$$

The teacher is updated with an exponential moving average (EMA) of the student: the update at step t is $\theta_{\text{teacher}} \leftarrow \beta_t \theta_{\text{teacher}} + (1 - \beta_t)\theta_{\text{student}}$. Following Liu et al. (2023) and Baevski et al. (2022), the positional embeddings of the teacher are copied from the student, not updated by EMA. All activated codewords are updated with an EMA of the teacher output embeddings:

$$\begin{aligned} \boldsymbol{s}_v^k &\leftarrow \begin{cases} \tau \boldsymbol{s}_v^k + (1-\tau) \sum_{i:(\boldsymbol{y}_i^k)_v = 1} \boldsymbol{z}_i^k & \text{if } \{i \mid (\boldsymbol{y}_i^k)_v = 1\} \neq \emptyset, \\ \boldsymbol{s}_v^k & \text{otherwise,} \end{cases} \\ n_v^k &\leftarrow \begin{cases} \tau n_v^k + (1-\tau) \sum_{i:(\boldsymbol{y}_i^k)_v = 1} 1 & \text{if } \{i \mid (\boldsymbol{y}_i^k)_v = 1\} \neq \emptyset, \\ n_v^k & \text{otherwise,} \end{cases} \\ \mathbf{C}_v^k &\leftarrow \frac{\boldsymbol{s}_v^k}{n_v^k}, \end{aligned} \tag{4}$$

where $\boldsymbol{s}_v^k$ is initialized randomly and n_v^k to 1, and τ is a constant decay parameter. Note that with this update procedure, all embeddings $\boldsymbol{z}_i^k$ are used to update the codewords, but the non-activated codewords do not move. The main change from Liu et al. (2023) is that the predictions are now aligned with the target layer. The output of layer k of the student is used to predict the label derived from layer k of the teacher, whereas DinoSR uses only the output of the last layer of the student encoder with $\tilde{\boldsymbol{y}}_i^k = \phi^k(\tilde{\boldsymbol{z}}_i^L)$.

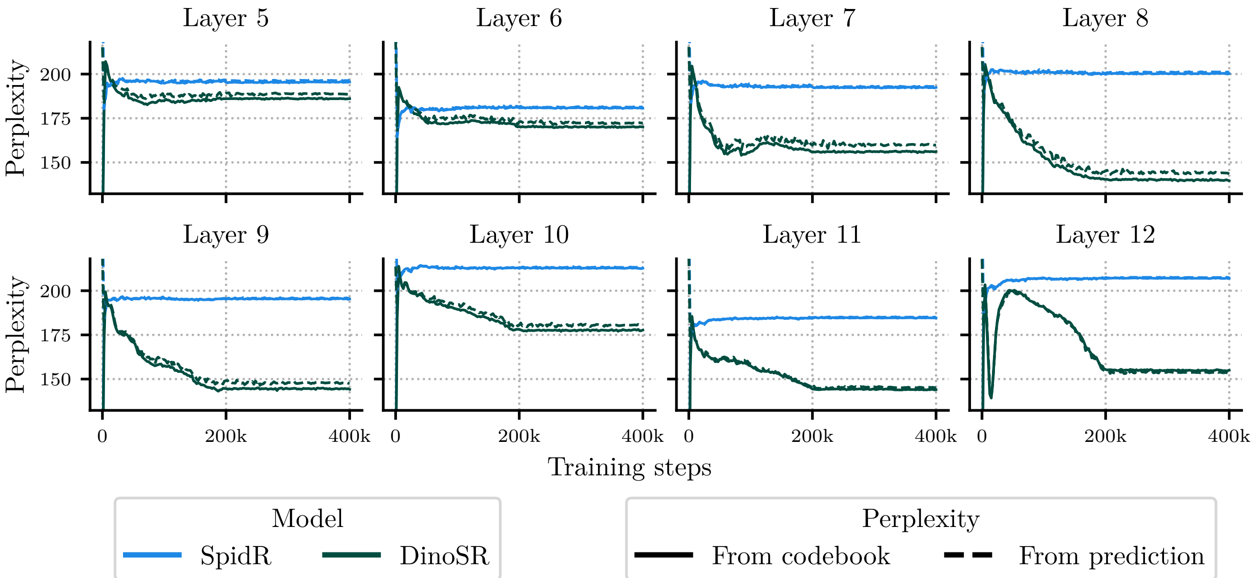


Figure 2: Codebook and prediction perplexities during training for SpidR and DinoSR on LibriSpeech `dev-clean`, with $K = 8$ codebooks. For each layer k, the codebook perplexity is computed over each batch with $\boldsymbol{p} = \boldsymbol{y}^k$ and then averaged across the dataset. The prediction perplexity uses $\boldsymbol{p} = \tilde{\boldsymbol{y}}^k$.

Baevski et al. (2022; 2023) also used the intermediate representations to train a SSL speech model, but in their case there was only one prediction head at the top of the student, trained to predict the average of the representations of the last K layers of the teacher. Our approach is reminiscent of the deep supervision literature (Lee et al., 2015), but in a self-supervised learning context.

4 Experiments

We pretrain SpidR, compare its training stability to DinoSR in section 4.2, and evaluate the phonetic and word discriminability of its representations in section 4.3. We then extract discrete tokens and train spoken language models. In section 4.4, we show the improvement of SpidR on zero-shot spoken language modeling task over other SSL encoders in identical conditions. Finally, in section 4.5 we compare the training time of SpidR to that of HuBERT, DinoSR, and previous work on efficient SSL.

4.1 Setup

Pretraining. The architecture follows the standard backbone from Baevski et al. (2020), and we make minimal changes from DinoSR. The model has a feature extractor with seven temporal convolutions and a projection layer, downsampling the 16 kHz input speech to 50Hz features of dimension $d = 768$. The student and teacher are Base size Transformer encoders with $L = 12$ layers. The prediction is done at the top $K = 8$ layers, using codebooks with $V = 256$ codewords. We pretrain with 960 hours of speech from LibriSpeech (Panayotov et al., 2015). To maintain a fair comparison with DinoSR, we keep the same total batch size of 63 minutes of audio across 16 GPUs. The codebook decay parameter is kept constant: $\tau = 0.9$. The student encoder and feature extractor are optimized with AdamW[1] (Loshchilov & Hutter, 2019) for 400k steps. We use the same learning rate scheduler as Liu et al. (2023), with a warmup from 5×10^{-6} to 5×10^{-4} within the first 12k steps, held constant until mid-training, and then exponentially decayed to 5×10^{-6}. We freeze the feature extractor after 200k steps. See appendix A in appendix for more details on the model and the hyperparameters.

[1]Previous work in SSL for speech (Baevski et al., 2020; Hsu et al., 2021; Liu et al., 2023) reported using Adam, but the Adam optimizer in fairseq that was used is actually implemented as AdamW.

Table 1: Zero-shot evaluation of self-supervised speech representations (in %, chance level 50% for ABX). All models are trained on LibriSpeech 960h. For each model, the selected layer is the one with the lowest average ABX. The best scores are in **bold** and second best are underlined.

Model	Layer	ABX within speaker ↓		ABX across speaker ↓		MAP words ↑	
		dev-clean	dev-other	dev-clean	dev-other	dev-clean	dev-other
wav2vec 2.0	6	4.47	5.63	5.25	7.82	44.81	31.92
HuBERT	11	<u>3.38</u>	<u>4.26</u>	<u>4.01</u>	<u>6.49</u>	46.07	33.37
data2vec	4	4.41	5.49	5.07	7.40	39.34	27.90
data2vec 2.0	1	5.13	5.77	5.72	7.53	**69.38**	<u>53.49</u>
DinoSR	5	4.05	5.11	4.72	7.29	63.02	45.86
DinoSR†	5	4.29	5.56	5.22	8.56	51.35	33.74
SpidR	6	**3.32**	**3.74**	**3.66**	**4.95**	<u>66.50</u>	**55.26**

† Our re-implementation.

We found during preliminary experiments that the norm of the weights of the Q, K, V projections in the attention layers could increase along training, and potentially lead to spikes in the loss and model collapse. Removing the biases in those layers fixed this issue, with no negative impact. We also modify the schedule of the decay parameter of the teacher β_t. Instead of the warmup-and-constant schedule of Baevski et al. (2022) and Liu et al. (2023), we take a smoother approach and set the decay at step t to be $\beta_t = 1 - (1 - \beta_0) \exp(-t/T)$, where $T = 10000$ is a timescale parameter and $\beta_0 = 0.999$. See appendix C in appendix for an ablation from DinoSR to SpidR.

Discrete units. We extract the embeddings from the layer with the best phonetic discriminability. The output representations of this layer are then quantized to derive the discrete units. We consider two quantization methods. We first use vector quantization with K-means clustering (Nguyen et al., 2020; Lakhotia et al., 2021), training it with the `train-clean-100` subset of LibriSpeech. For DinoSR and SpidR, we also consider using the codebook predictions, by taking the assignment made by the prediction heads from the student encoder ϕ_k and selecting the label for which the probability is maximum. We deduplicate the tokens before passing them to the language model.

Spoken language models. The SLMs are OPT-125M models (Zhang et al., 2022), trained on the 6k hours subset of Libri-Light (Kahn et al., 2020) using the SlamKit framework (Maimon et al., 2025a). We train on one GPU with a batch size of 16, with 4 gradient accumulation steps, and a context length of 2048 for 100k steps. The other training parameters follow the defaults of OPT-125M. The selected checkpoint in the one with the lowest validation loss.

4.2 Training stability

Our motivation for changing DinoSR's learning objective was to stabilize the training procedure. We found in preliminary studies that the online clustering of DinoSR tended to collapse, as tracked by the codebook and prediction head perplexities. The perplexity $2^{H(\boldsymbol{p})}$, with $H(\boldsymbol{p}) = -\sum_{v \in V} \boldsymbol{p}_v \log_2 \boldsymbol{p}_v$ the entropy, measures the diversity of codewords used by the model, with $\boldsymbol{p}_v$ being the probability of the assignment v. The codebook perplexity at layer k is measured with $\boldsymbol{p} = \boldsymbol{y}^k \in \{0, 1\}^V$, and the prediction head perplexity with $\boldsymbol{p} = \tilde{\boldsymbol{y}}^k \in (0, 1)^V$. With a perplexity of V, all codewords are used equally.

In figure 2, we compare the codebook and prediction perplexities of DinoSR and SpidR during training. The perplexities are computed on LibriSpeech `dev-clean` over each batch, using the same batch size as in pretraining, and then averaged across the dataset. Liu et al. (2023) report that DinoSR has a much higher perplexity than other online clustering methods, such as VQ-APC (Chung et al., 2020) and Co-training APC (Yeh & Tang, 2022). However, DinoSR is still prone to codebook collapse, especially in the last layers. In DinoSR, the output of the last layer $\tilde{\boldsymbol{z}}^L$ is given to all heads ϕ^k to derive the pseudo-labels from the

Table 2: Zero-shot discrete units quality and spoken language modeling metrics from HuBERT, DinoSR, and SpidR (in %, chance level 50%, except for PNMI). The speech encoders are trained on LibriSpeech 960h and the language models on Libri-Light 6k. The vocabulary size is $V = 256$. For each model, the selected layer is the one with the lowest average ABX on continuous embeddings. The best scores are in **bold** and second best are <u>underlined</u>.

Model	Layer	Units	Discrete units quality		Language modeling			
			ABX ↓	PNMI ↑	sWUGGY ↑ all	sWUGGY ↑ in-vocab	sBLIMP ↑	tSC ↑
HuBERT	11	K-means	7.32	**0.636**	65.15	73.21	55.68	67.66
DinoSR	5	Codebook	7.89	0.620	57.50	60.35	55.97	68.04
		K-means	10.81	0.588	55.76	57.81	54.14	64.72
DinoSR†	5	Codebook	7.70	0.614	59.44	62.61	54.64	65.95
		K-means	10.61	0.589	55.59	57.27	53.40	62.43
SpidR	6	Codebook	**6.32**	0.602	<u>70.24</u>	<u>80.45</u>	**57.48**	<u>69.37</u>
		K-means	<u>7.10</u>	<u>0.633</u>	**71.90**	**82.33**	<u>57.09</u>	**70.55**

† Our re-implementation.

intermediate layers of the teacher. The codebook assignments information for all K layers must be linearly extractable from $\tilde{\boldsymbol{z}}^L$. SpidR is more straightforward: $\tilde{\boldsymbol{z}}^k$ is used to predict the assignments from layer k. This result suggests that our training objective reduces the distribution shift between the embeddings and the codebooks, a challenge frequently encountered in neural networks with vector quantization (Huh et al., 2023).

4.3 Evaluation of the learned speech representations

In order to train a spoken language model, we derive discrete units from the representations of the SSL model. For successful language modeling, the units need to encode the underlying linguistic content, not the speaker information or the acoustic background. Therefore, we want the model to have highly accessible phonetic and word information in its representations, and a well clustered representation space. Following previous work (Nguyen et al., 2020), we evaluate the SSL models with metrics computing the discriminability of the embeddings. This evaluation is then used to select the target layer for spoken language modeling (Lakhotia et al., 2021).

The first metric of interest is the ABX discriminability over phonemes (Schatz, 2016). It measures how well triphones differing only by the central phone (like /bag/ and /beg/) are discriminated in the embedding space by comparing the distances between two instances X and A of the same triphone to the distance between X and another triphone B. The test is successful if the representations of X and A are closer than those of X and B. In the *within* speaker task, A, B and X are from the same speaker, whereas in the *across* speaker task, A and B are from the same speaker and X from another one. We use the implementation of Poli et al. (2025) to compute ABX scores. It fixes issues with the slicing of features that existed in the Libri-Light version, which explains the differences with the scores reported by Liu et al. (2023).

In addition to the ABX, which operates at the triphone level, we evaluate embedding discriminability at the word level. An ABX task where A and X are instances of the same word and B is from a different word would be too easy in most cases. Instead, we opt for a more challenging metric: Mean Average Precision (MAP) over words (Carlin et al., 2011). This retrieval task requires that, for each word, the closest embeddings correspond to other instances of the same word. Unlike ABX, which uses Dynamic Time Warping to handle duration differences between speech segments, we average word representations over the time axis. Following Algayres et al. (2020), we use MAP@R (Musgrave et al., 2020) and get the final score by averaging over all words, where R is the number of other instances of a given query word, and

$$\text{MAP@}R = \frac{1}{R}\sum_{i=1}^{R} P(i), \text{ where } P(i) = \begin{cases} \text{precision at } i & \text{if the } i\text{-th retrieval is correct,} \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

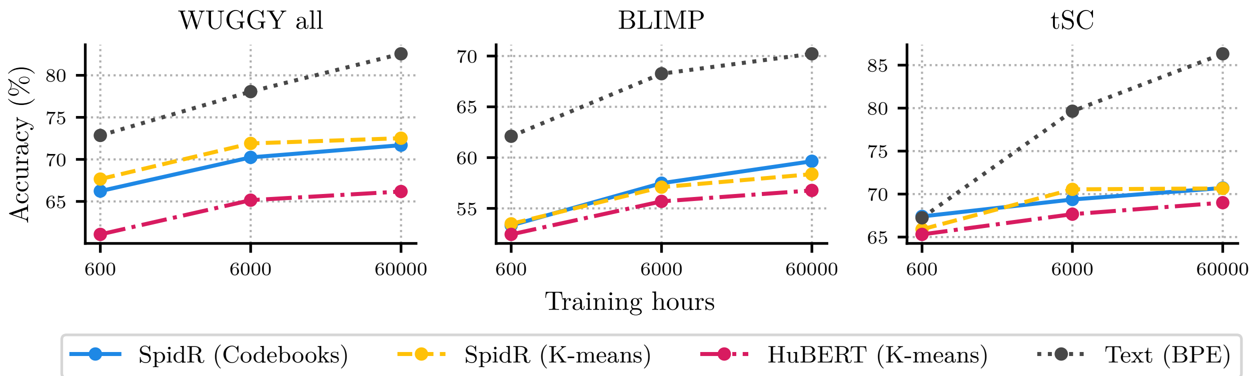


Figure 3: Data scaling results for a 125M parameters OPT model trained on Libri-Light, with different discrete units encoders. Zero-shot accuracy in %, chance level 50%. The speech encoders have $V = 256$ units. The log-likelihoods are normalized by the number of tokens, except for WUGGY with text.

The intermediate layer chosen for each model is the one with the lowest average ABX error rate, which is not necessarily the best layer in terms of MAP (see figure 7 in appendix). As shown in table 1, SpidR outperforms baseline SSL models on both metrics. For all models, we computed the ABX using the angular distance on the representations of the intermediate layers, contrary to Liu et al. (2023) who used the prediction heads with the KL-symmetric distance for DinoSR. See appendix B.1 in appendix for additionnal discriminibility results and appendix B.2 for a visualization of the learned embeddings.

4.4 Evaluation of downstream spoken language modeling

Evaluation metrics. In order to assess the role of the speech encoder in spoken language modeling, we consider three standard tasks. At the lexical level, sWUGGY (Nguyen et al., 2020) evaluates the ability of the network to assign a higher probability to the true word than to a matching non-word. We also report results for "in-vocab" pairs, keeping only the words present in LibriSpeech. At the syntactic level, in sBLIMP, the network has to decide which sentence is grammatically correct, given minimal sentence pairs. Spoken StoryCloze (Mostafazadeh et al., 2017; Hassid et al., 2023) measures the ability of the model to choose the correct continuation of the beginning of a short story. We report the results for the "Topic" version (tSC), based on simpler negative examples. Following previous works, the log-likelihoods are normalized by the number of tokens.

Comparison against other speech encoders. To evaluate the contribution of units from SpidR for spoken language modeling, we compare in table 2 SLMs trained with units from HuBERT, DinoSR or SpidR on those three metrics. We keep a vocabulary size of $V = 256$ for all models to allow for exact comparison between units derived from K-means and from the codebook predictions. We also add an analysis of the discrete units' quality with the ABX on one-hot tokens, as well as the Phone Normalized Mutual Information (PNMI) (Hsu et al., 2021). The alignements used for PNMI are those from the ZeroSpeech 2021 challenge (Nguyen et al., 2020). Both metrics indicate how well the units correlate with the underlying phonemes. See appendix B.3 in appendix for a layer-wise analysis of the discrete units quality and downstream spoken language modeling results. SpidR outperforms all other encoders on SLM metrics and on ABX on discrete units.

Data scaling analysis. To assess how the advantage of SpidR over other SSL models generalizes across different training conditions, we compare the scaling properties of SLMs trained with HuBERT or SpidR across varying data quantities in figure 3. We train SLMs on three dataset sizes: the 600h subset of Libri-Light, the 6k subset, or the full 60k dataset. We maintain the same hyperparameters as before, and we train for 500k steps when using the Libri-Light dataset instead of 100k steps. Additionally, we train a topline text LM using BPE tokens from the original books read, ensuring exact dataset matching between text and spoken LMs. The transcriptions are from Kang et al. (2024); we use the standard OPT tokenizer. Apart from the

Table 3: Zero-shot spoken language modeling results (in %, chance level 50%) for ∼150M parameters models trained on Libri-Light 6k from HuBERT or SpidR discrete units, across different number of units. Results for models based on HuBERT are from Chang et al. (2024); Messica & Adi (2024). The best scores are in **bold** and second best are underlined.

Num. units	Model	Units	sWUGGY ↑ all	sWUGGY ↑ in-vocab	sBLIMP ↑	tSC ↑
50	HuBERT	K-means	-	67.48	52.42	66.27
	HuBERT	Spin_{50}	58.90	63.52	<u>59.38</u>	65.85
	HuBERT	$\text{DC-Spin}_{50,4096}$	<u>65.05</u>	<u>73.51</u>	**60.15**	<u>69.91</u>
	SpidR	K-means	**68.77**	**78.32**	58.20	**71.62**
100	HuBERT	K-means	-	67.75	51.96	67.18
	HuBERT	Spin_{100}	65.28	73.25	<u>59.97</u>	68.25
	HuBERT	$\text{DC-Spin}_{100,4096}$	<u>68.04</u>	<u>78.47</u>	**61.35**	<u>70.18</u>
	SpidR	K-means	**70.77**	**81.39**	59.17	**70.23**
200	HuBERT	K-means	-	71.88	52.43	67.55
	HuBERT	Spin_{200}	68.95	78.19	**62.55**	69.64
	HuBERT	$\text{DC-Spin}_{200,4096}$	<u>70.79</u>	<u>80.59</u>	<u>62.13</u>	<u>69.21</u>
	SpidR	K-means	**71.63**	**82.49**	58.05	**70.28**
500	HuBERT	K-means	66.74	74.72	55.54	63.23
	HuBERT	Spin_{500}	70.03	79.31	<u>60.08</u>	67.45
	HuBERT	$\text{DC-Spin}_{500,4096}$	**71.48**	**81.38**	**60.84**	<u>67.50</u>
	SpidR	K-means	<u>70.14</u>	<u>80.32</u>	56.87	**69.27**

vocabulary size, all training hyperparameters match those of the SLMs. We evaluate the text LM on the original text versions of WUGGY, BLIMP and tSC. On WUGGY, we do not normalize log-likelihoods for the text LM since non-words are segmented into more tokens by the tokenizer. Across all conditions, SpidR consistently outperforms HuBERT on all metrics, whether using codebook predictions or K-means clustering. However, it does not change the scaling properties: text LMs trained under the same conditions achieve both better performance and superior scaling, particularly on tSC.

Across number of units. Finally, we investigate the role of the number of units in the spoken LM in table 3. We train SLMs on SpidR units derived from K-means with vocabulary sizes in $\{50, 100, 200, 500\}$ under the same conditions as above. We compare the zero-shot scores to HuBERT-based models from Chang et al. (2024); Messica & Adi (2024). Those works use `transformer_lm_big` from fairseq (Ott et al., 2019) with 150M parameters, whereas we use the OPT-125M architecture. All language models are trained on Libri-Light 6k. The advantage of SpidR over HuBERT remains consistent across different vocabulary sizes. We also compare against units derived from HuBERT with Spin or DC-Spin. These approaches aim to improve speaker invariance and speech tokenization by learning auxiliary codebooks using swapped prediction. SpidR with standard K-means clustering matches the performance of HuBERT with DC-Spin units, with the latter showing advantages on sBLIMP, while SpidR performs better on the other metrics.

4.5 Codebase and pretraining time

In addition to learning strong phonetic representations, SpidR was designed with practical considerations in mind: reducing computational costs and simplifying the training pipeline. We developed a minimal PyTorch codebase compatible with the latest PyTorch features, with model implementations based on HuBERT from torchaudio (Hwang et al., 2023).

Table 4: Pretraining compute footprint of SpidR against other SSL models operating at 50Hz. We report the pretraining times in the default settings given in the corresponding papers. k2SSL Zipformer is trained using labels from the first iteration of HuBERT, and Academic HuBERT with labels from E-branchformer (Kim et al., 2023).

Model	GPUs	Steps	Pretraining time	GPU hours
HuBERT (Hsu et al., 2021)	A100 ×32	650k	62 hr	1984
DinoSR (Liu et al., 2023)	V100 ×16	400k	180 hr	2880
data2vec 2.0 (Baevski et al., 2023)	A100 ×16	50k	43 hr	688
MelHuBERT (Lin et al., 2023)	RTX 3090 ×1	630k	300 hr	300
Academic HuBERT (Chen et al., 2023a)	A100 ×8	1760k	240 hr	1920
k2SSL Zipformer (Yang et al., 2025)	V100 ×8	225k	64 hr	513
SpidR and DinoSR (our reimplem.)	A100 ×16	400k	23 hr	369

We re-implemented DinoSR in this codebase, reducing training time from the reported 180 hours to just 70 hours on 16 V100 GPUs under identical settings to Liu et al. (2023). We further optimized the codebase for full compatibility with `torch.compile` (Ansel et al., 2024) and minimized host-device synchronization points. Since `torch.compile` merges native PyTorch modules and functions into optimized kernels, this results in significant throughput improvements. As shown in table 4, SpidR can be pretrained in under a day on 16 A100 GPUs. With 32 A100 GPUs, training SpidR only takes 14 hours (maintaining the same total batch size), compared to 62 hours for HuBERT. The single-pass training of SpidR also eliminates the feature extraction and label computation steps required by HuBERT, removing common engineering challenges. Figure 4 shows pretraining times for SpidR across different hardware configurations (4, 8, and 16 A100 or H100 GPUs) with constant total batch size. Using `torch.compile` provides approximately a 20% speedup in pretraining time. We open-source both the final checkpoints and the codebase.

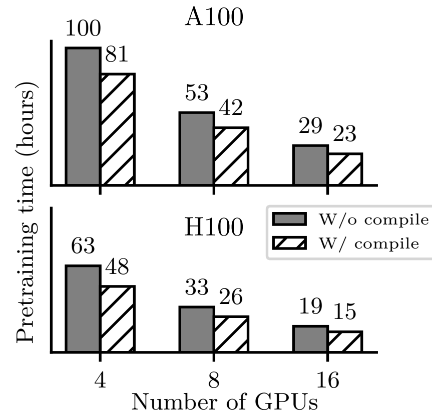


Figure 4: Approximate pretraining time for various hardware configurations with constant total batch size.

5 Conclusion

We presented SpidR: a self-supervised speech representation model that efficiently learns strong representations for spoken language modeling. We demonstrated that its learning objective, adapted from DinoSR, enables stable training and produces representations with salient phonetic information. Spoken language models using units from SpidR consistently outperform those based on HuBERT and DinoSR.

This work focused exclusively on English, and only with data from LibriVox audiobooks. Major multilingual SSL models are based on either wav2vec 2.0 (Conneau et al., 2021; Babu et al., 2022; Pratap et al., 2024) or HuBERT/WavLM (Chen et al., 2023b; 2024; Zanon Boito et al., 2024), and require massive computational resources for training. SpidR offers a solution for learning strong representations much faster, serving as foundation for future models and making approaches in other languages or multilingual settings more accessible due to reduced computational cost. Future work will focus on scaling the speech encoder to more data and languages while ensuring robustness to diverse acoustic conditions, with the goal of building a speech encoder capable of learning linguistic representations from ecological speech.

References

Robin Algayres, Mohamed Salah Zaiem, Benoît Sagot, and Emmanuel Dupoux. Evaluating the reliability of acoustic speech embeddings. In *Interspeech 2020*, pp. 4621–4625, 2020. doi:10.21437/Interspeech.2020-2362.

Robin Algayres, Yossi Adi, Tu Nguyen, Jade Copet, Gabriel Synnaeve, Benoît Sagot, and Emmanuel Dupoux. Generative spoken language model based on continuous word-sized audio tokens. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 3008–3028, dec 2023. doi:10.18653/v1/2023.emnlp-main.182. URL https://aclanthology.org/2023.emnlp-main.182/.

Jason Ansel, Edward Yang, Horace He, Natalia Gimelshein, Animesh Jain, Michael Voznesensky, Bin Bao, Peter Bell, David Berard, Evgeni Burovski, Geeta Chauhan, Anjali Chourdia, Will Constable, Alban Desmaison, Zachary DeVito, Elias Ellison, Will Feng, Jiong Gong, Michael Gschwind, Brian Hirsh, Sherlock Huang, Kshiteej Kalambarkar, Laurent Kirsch, Michael Lazos, Mario Lezcano, Yanbo Liang, Jason Liang, Yinghai Lu, C. K. Luk, Bert Maher, Yunjie Pan, Christian Puhrsch, Matthias Reso, Mark Saroufim, Marcos Yukio Siraichi, Helen Suk, Shunting Zhang, Michael Suo, Phil Tillet, Xu Zhao, Eikan Wang, Keren Zhou, Richard Zou, Xiaodong Wang, Ajit Mathews, William Wen, Gregory Chanan, Peng Wu, and Soumith Chintala. Pytorch 2: Faster machine learning through dynamic python bytecode transformation and graph compilation. In *Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2*, ASPLOS '24, pp. 929–947, 2024. ISBN 9798400703850. doi:10.1145/3620665.3640366. URL https://doi.org/10.1145/3620665.3640366.

Siddhant Arora, Kai-Wei Chang, Chung-Ming Chien, Yifan Peng, Haibin Wu, Yossi Adi, Emmanuel Dupoux, Hung-Yi Lee, Karen Livescu, and Shinji Watanabe. On The Landscape of Spoken Language Models: A Comprehensive Survey, apr 2025. URL https://arxiv.org/abs/2504.08528.

Alan Baade, Puyuan Peng, and David Harwath. SyllableLM: Learning coarse semantic units for speech language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=dGSOn7sdWg.

Arun Babu, Changhan Wang, Andros Tjandra, Kushal Lakhotia, Qiantong Xu, Naman Goyal, Kritika Singh, Patrick von Platen, Yatharth Saraf, Juan Pino, Alexei Baevski, Alexis Conneau, and Michael Auli. XLS-R: Self-supervised cross-lingual speech representation learning at scale. In *Interspeech 2022*, pp. 2278–2282, 2022. doi:10.21437/Interspeech.2022-143.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. Wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations. In *Advances in Neural Information Processing Systems*, volume 33, pp. 12449–12460, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/92d1e1eb1cd6f9fba3227870bb6d7f07-Abstract.html.

Alexei Baevski, Arun Babu, Wei-Ning Hsu, and Michael Auli. Efficient Self-supervised Learning with Contextualized Target Representations for Vision, Speech and Language. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 1416–1429, jul 2023. URL https://proceedings.mlr.press/v202/baevski23a.html.

Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. data2vec: A general framework for self-supervised learning in speech, vision and language. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 1298–1312, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/baevski22a.html.

Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Dominik Roblek, Olivier Teboul, David Grangier, Marco Tagliasacchi, and Neil Zeghidour. AudioLM: A language modeling approach to audio generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:2523–2533, 2023. doi:10.1109/TASLP.2023.3288409.

Michael A. Carlin, Samuel Thomas, Aren Jansen, and Hynek Hermansky. Rapid evaluation of speech representations for spoken term discovery. In *Interspeech 2011*, pp. 821–824, 2011. doi:10.21437/Interspeech.2011-304.

Heng-Jui Chang and James Glass. R-spin: Efficient speaker and noise-invariant representation learning with acoustic pieces. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 642–662, jun 2024. doi:10.18653/v1/2024.naacl-long.36. URL `https://aclanthology.org/2024.naacl-long.36/`.

Heng-Jui Chang, Alexander H. Liu, and James Glass. Self-supervised fine-tuning for improved content representations by speaker-invariant clustering. In *Interspeech 2023*, pp. 2983–2987, 2023. doi:10.21437/Interspeech.2023-847.

Heng-Jui Chang, Hongyu Gong, Changhan Wang, James Glass, and Yu-An Chung. DC-Spin: A speaker-invariant speech tokenizer for spoken language models, 2024. URL `https://arxiv.org/abs/2410.24177`.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, Jian Wu, Long Zhou, Shuo Ren, Yanmin Qian, Yao Qian, Jian Wu, Michael Zeng, Xiangzhan Yu, and Furu Wei. WavLM: Large-Scale Self-Supervised Pre-Training for Full Stack Speech Processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1505–1518, oct 2022. ISSN 1941-0484. doi:10.1109/JSTSP.2022.3188113.

Sanyuan Chen, Chengyi Wang, Yu Wu, Ziqiang Zhang, Long Zhou, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, Lei He, Sheng Zhao, and Furu Wei. Neural codec language models are zero-shot text to speech synthesizers. *IEEE Transactions on Audio, Speech and Language Processing*, 33:705–718, 2025. doi:10.1109/TASLPRO.2025.3530270.

William Chen, Xuankai Chang, Yifan Peng, Zhaoheng Ni, Soumi Maiti, and Shinji Watanabe. Reducing barriers to self-supervised learning: HuBERT pre-training with academic compute. In *Interspeech 2023*, pp. 4404–4408, 2023a. doi:10.21437/Interspeech.2023-1176.

William Chen, Jiatong Shi, Brian Yan, Dan Berrebbi, Wangyou Zhang, Yifan Peng, Xuankai Chang, Soumi Maiti, and Shinji Watanabe. Joint prediction and denoising for large-scale multilingual self-supervised learning. In *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 1–8, 2023b. doi:10.1109/ASRU57964.2023.10389735.

William Chen, Wangyou Zhang, Yifan Peng, Xinjian Li, Jinchuan Tian, Jiatong Shi, Xuankai Chang, Soumi Maiti, Karen Livescu, and Shinji Watanabe. Towards robust speech representation learning for thousands of languages. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 10205–10224, nov 2024. doi:10.18653/v1/2024.emnlp-main.570. URL `https://aclanthology.org/2024.emnlp-main.570/`.

Kwanghee Choi, Ankita Pasad, Tomohiko Nakamura, Satoru Fukayama, Karen Livescu, and Shinji Watanabe. Self-supervised speech representations are more phonetic than semantic. In *Interspeech 2024*, pp. 4578–4582, 2024. doi:10.21437/Interspeech.2024-1157.

Yu-An Chung and James Glass. Generative pre-training for speech with autoregressive predictive coding. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3497–3501, 2020. doi:10.1109/ICASSP40776.2020.9054438.

Yu-An Chung, Hao Tang, and James Glass. Vector-quantized autoregressive predictive coding. In *Interspeech 2020*, pp. 3760–3764, 2020. doi:10.21437/Interspeech.2020-1228.

Yu-An Chung, Yu Zhang, Wei Han, Chung-Cheng Chiu, James Qin, Ruoming Pang, and Yonghui Wu. w2v-BERT: Combining contrastive learning and masked language modeling for self-supervised speech pre-training. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 244–250, 2021. doi:10.1109/ASRU51503.2021.9688253.

Alexis Conneau, Alexei Baevski, Ronan Collobert, Abdelrahman Mohamed, and Michael Auli. Unsupervised cross-lingual representation learning for speech recognition. In *Interspeech 2021*, pp. 2426–2430, 2021. doi:10.21437/Interspeech.2021-329.

Santiago Cuervo and Ricard Marxer. Scaling properties of speech language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 351–361, nov 2024. doi:10.18653/v1/2024.emnlp-main.21. URL https://aclanthology.org/2024.emnlp-main.21/.

Santiago Cuervo, Adel Moumen, Yanis Labrak, Sameer Khurana, Antoine Laurent, Mickael Rouvier, and Ricard Marxer. Text-speech language models with improved cross-modal transfer by aligning abstraction levels, 2025. URL https://arxiv.org/abs/2503.06211.

Maureen de Seyssel, Marvin Lavechin, Yossi Adi, Emmanuel Dupoux, and Guillaume Wisniewski. Probing phoneme, language and speaker information in unsupervised speech representations. In *Interspeech 2022*, pp. 1402–1406, 2022. doi:10.21437/Interspeech.2022-373.

Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High fidelity neural audio compression. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=ivCd8z8zR2. Featured Certification, Reproducibility Certification.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, jun 2019. doi:10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423/.

Ewan Dunbar, Mathieu Bernard, Nicolas Hamilakis, Tu Anh Nguyen, Maureen De Seyssel, Patricia Rozé, Morgane Rivière, Eugene Kharitonov, and Emmanuel Dupoux. The Zero Resource Speech Challenge 2021: Spoken Language Modelling. In *Interspeech 2021*, pp. 1574–1578, aug 2021. doi:10.21437/Interspeech.2021-1755.

Ewan Dunbar, Nicolas Hamilakis, and Emmanuel Dupoux. Self-Supervised Language Learning From Raw Audio: Lessons From the Zero Resource Speech Challenge. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1211–1226, oct 2022. ISSN 1941-0484. doi:10.1109/JSTSP.2022.3206084.

Alexandre Défossez, Laurent Mazaré, Manu Orsini, Amélie Royer, Patrick Pérez, Hervé Jégou, Edouard Grave, and Neil Zeghidour. Moshi: A speech-text foundation model for real-time dialogue, 2024. URL https://arxiv.org/abs/2410.00037.

Itai Gat, Felix Kreuk, Tu Anh Nguyen, Ann Lee, Jade Copet, Gabriel Synnaeve, Emmanuel Dupoux, and Yossi Adi. Augmentation invariant discrete representation for generative spoken language modeling. In *Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023)*, pp. 465–477, jul 2023. doi:10.18653/v1/2023.iwslt-1.46. URL https://aclanthology.org/2023.iwslt-1.46/.

Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang. Conformer: Convolution-augmented transformer for speech recognition. In *Interspeech 2020*, pp. 5036–5040, 2020. doi:10.21437/Interspeech.2020-3015.

Mark Hallap, Emmanuel Dupoux, and Ewan Dunbar. Evaluating context-invariance in unsupervised speech representations. In *INTERSPEECH 2023*, pp. 2973–2977, aug 2023. doi:10.21437/Interspeech.2023-1862.

Michael Hassid, Tal Remez, Tu Anh Nguyen, Itai Gat, Alexis Conneau, Felix Kreuk, Jade Copet, Alexandre Defossez, Gabriel Synnaeve, Emmanuel Dupoux, et al. Textually pretrained speech language models. *Advances in Neural Information Processing Systems*, 36: 63483–63501, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/c859b99b5d717c9035e79d43dfd69435-Abstract-Conference.html.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460, 2021. ISSN 2329-9304. doi:10.1109/TASLP.2021.3122291.

Minyoung Huh, Brian Cheung, Pulkit Agrawal, and Phillip Isola. Straightening out the straight-through estimator: Overcoming optimization challenges in vector quantized networks. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 14096–14113, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/huh23a.html.

Jeff Hwang, Moto Hira, Caroline Chen, Xiaohui Zhang, Zhaoheng Ni, Guangzhi Sun, Pingchuan Ma, Ruizhe Huang, Vineel Pratap, Yuekai Zhang, Anurag Kumar, Chin-Yun Yu, Chuang Zhu, Chunxi Liu, Jacob Kahn, Mirco Ravanelli, Peng Sun, Shinji Watanabe, Yangyang Shi, and Yumeng Tao. Torchaudio 2.1: Advancing speech recognition, self-supervised learning, and audio processing components for pytorch. In *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 1–9, 2023. doi:10.1109/ASRU57964.2023.10389648.

J. Kahn, M. Rivière, W. Zheng, E. Kharitonov, Q. Xu, P.E. Mazaré, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen, T. Likhomanenko, G. Synnaeve, A. Joulin, A. Mohamed, and E. Dupoux. Libri-light: A benchmark for ASR with limited or no supervision. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7669–7673, 2020. doi:10.1109/ICASSP40776.2020.9052942.

Wei Kang, Xiaoyu Yang, Zengwei Yao, Fangjun Kuang, Yifan Yang, Liyong Guo, Long Lin, and Daniel Povey. Libriheavy: A 50,000 hours ASR corpus with punctuation casing and context. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 10991–10995, 2024. doi:10.1109/ICASSP48485.2024.10447120.

Eugene Kharitonov, Ann Lee, Adam Polyak, Yossi Adi, Jade Copet, Kushal Lakhotia, Tu Anh Nguyen, Morgane Riviere, Abdelrahman Mohamed, Emmanuel Dupoux, and Wei-Ning Hsu. Text-free prosody-aware generative spoken language modeling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8666–8681, may 2022. doi:10.18653/v1/2022.acl-long.593. URL https://aclanthology.org/2022.acl-long.593/.

Kwangyoun Kim, Felix Wu, Yifan Peng, Jing Pan, Prashant Sridhar, Kyu J. Han, and Shinji Watanabe. E-branchformer: Branchformer with enhanced merging for speech recognition. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pp. 84–91, 2023. doi:10.1109/SLT54892.2023.10022656.

Felix Kreuk, Adam Polyak, Jade Copet, Eugene Kharitonov, Tu Anh Nguyen, Morgan Rivière, Wei-Ning Hsu, Abdelrahman Mohamed, Emmanuel Dupoux, and Yossi Adi. Textless speech emotion conversion using discrete & decomposed representations. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 11200–11214, dec 2022. doi:10.18653/v1/2022.emnlp-main.769. URL https://aclanthology.org/2022.emnlp-main.769/.

Kushal Lakhotia, Eugene Kharitonov, Wei-Ning Hsu, Yossi Adi, Adam Polyak, Benjamin Bolte, Tu-Anh Nguyen, Jade Copet, Alexei Baevski, Abdelrahman Mohamed, and Emmanuel Dupoux. On Generative Spoken Language Modeling from Raw Audio. *Transactions of the Association for Computational Linguistics*, 9:1336–1354, 2021. doi:10.1162/tacl_a_00430.

Chen-Yu Lee, Saining Xie, Patrick Gallagher, Zhengyou Zhang, and Zhuowen Tu. Deeply-supervised nets. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, volume 38 of *Proceedings of Machine Learning Research*, pp. 562–570, 09–12 May 2015. URL https://proceedings.mlr.press/v38/lee15a.html.

Tzu-Quan Lin, Hung-yi Lee, and Hao Tang. Melhubert: A simplified hubert on mel spectrograms. In *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 1–8. IEEE, 2023. URL https://arxiv.org/abs/2211.09944.

Alexander H. Liu, Heng-Jui Chang, Michael Auli, Wei-Ning Hsu, and Jim Glass. DinoSR: Self-Distillation and Online Clustering for Self-supervised Speech Representation Learning. *Advances in Neural Information Processing Systems*, 36:58346–58362, dec 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/b6404bf461c3c3186bdf5f55756af908-Abstract-Conference.html.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Gallil Maimon, Avishai Elmakies, and Yossi Adi. Slamming: Training a speech language model on one GPU in a day. In *Findings of the Association for Computational Linguistics: ACL 2025*, 2025a. URL https://arxiv.org/abs/2502.15814.

Gallil Maimon, Michael Hassid, Amit Roth, and Yossi Adi. Scaling analysis of interleaved speech-text language models, 2025b. URL https://arxiv.org/abs/2504.02398.

Shoval Messica and Yossi Adi. NAST: Noise aware speech tokenization for speech language models. In *Interspeech 2024*, pp. 4169–4173, 2024. doi:10.21437/Interspeech.2024-288.

Abdelrahman Mohamed, Hung-yi Lee, Lasse Borgholt, Jakob D. Havtorn, Joakim Edin, Christian Igel, Katrin Kirchhoff, Shang-Wen Li, Karen Livescu, Lars Maaløe, Tara N. Sainath, and Shinji Watanabe. Self-Supervised Speech Representation Learning: A Review. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1179–1210, oct 2022. ISSN 1941-0484. doi:10.1109/JSTSP.2022.3207050.

Nasrin Mostafazadeh, Michael Roth, Annie Louis, Nathanael Chambers, and James Allen. LSDSem 2017 shared task: The story cloze test. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*, pp. 46–51, apr 2017. doi:10.18653/v1/W17-0906. URL https://aclanthology.org/W17-0906/.

Kevin Musgrave, Serge Belongie, and Ser-Nam Lim. A metric learning reality check. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXV 16*, pp. 681–699. Springer, 2020. URL https://www.ecva.net/papers/eccv_2020/papers_ECCV/html/5170_ECCV_2020_paper.php.

Tu Anh Nguyen, Maureen de Seyssel, Patricia Rozé, Morgane Rivière, Evgeny Kharitonov, Alexei Baevski, Ewan Dunbar, and Emmanuel Dupoux. The zero resource speech benchmark 2021: Metrics and baselines for unsupervised spoken language modeling, 2020. URL https://arxiv.org/abs/2011.11588.

Tu Anh Nguyen, Benoit Sagot, and Emmanuel Dupoux. Are discrete units necessary for spoken language modeling? *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1415–1423, 2022. doi:10.1109/JSTSP.2022.3200909.

Tu Anh Nguyen, Wei-Ning Hsu, Antony D'Avirro, Bowen Shi, Itai Gat, Maryam Fazel-Zarani, Tal Remez, Jade Copet, Gabriel Synnaeve, Michael Hassid, Felix Kreuk, Yossi Adi, and Emmanuel Dupoux. Expresso: A benchmark and analysis of discrete expressive speech resynthesis. In *Interspeech 2023*, pp. 4823–4827, 2023. doi:10.21437/Interspeech.2023-1905.

Tu Anh Nguyen, Benjamin Muller, Bokai Yu, Marta R. Costa-jussa, Maha Elbayad, Sravya Popuri, Christophe Ropers, Paul-Ambroise Duquenne, Robin Algayres, Ruslan Mavlyutov, Itai Gat, Mary Williamson, Gabriel Synnaeve, Juan Pino, Benoît Sagot, and Emmanuel Dupoux. SpiRit-LM: Interleaved spoken and written language model. *Transactions of the Association for Computational Linguistics*, 13:30–52, 2025. doi:10.1162/tacl_a_00728. URL https://aclanthology.org/2025.tacl-1.2/.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pp. 48–53, jun 2019. doi:10.18653/v1/N19-4009. URL https://aclanthology.org/N19-4009/.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: An ASR corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5206–5210, 2015. doi:10.1109/ICASSP.2015.7178964.

Maxime Poli, Emmanuel Chemla, and Emmanuel Dupoux. Improving spoken language modeling with phoneme classification: A simple fine-tuning approach. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 5284–5292, nov 2024. doi:10.18653/v1/2024.emnlp-main.302. URL https://aclanthology.org/2024.emnlp-main.302/.

Maxime Poli, Emmanuel Chemla, and Emmanuel Dupoux. fastabx: A library for efficient computation of ABX discriminability, 2025. URL https://arxiv.org/abs/2505.02692.

Adam Polyak, Yossi Adi, Jade Copet, Eugene Kharitonov, Kushal Lakhotia, Wei-Ning Hsu, Abdelrahman Mohamed, and Emmanuel Dupoux. Speech resynthesis from discrete disentangled self-supervised representations. In *Interspeech 2021*, pp. 3615–3619, 2021. doi:10.21437/Interspeech.2021-475.

Vineel Pratap, Andros Tjandra, Bowen Shi, Paden Tomasello, Arun Babu, Sayani Kundu, Ali Elkahky, Zhaoheng Ni, Apoorv Vyas, Maryam Fazel-Zarandi, Alexei Baevski, Yossi Adi, Xiaohui Zhang, Wei-Ning Hsu, Alexis Conneau, and Michael Auli. Scaling speech technology to 1,000+ languages. *Journal of Machine Learning Research*, 25(97):1–52, 2024. URL http://jmlr.org/papers/v25/23-1318.html.

Kaizhi Qian, Yang Zhang, Heting Gao, Junrui Ni, Cheng-I Lai, David Cox, Mark Hasegawa-Johnson, and Shiyu Chang. ContentVec: An improved self-supervised speech representation by disentangling speakers. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 18003–18017, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/qian22b.html.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine Mcleavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 28492–28518, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/radford23a.html.

Mirco Ravanelli, Titouan Parcollet, Adel Moumen, Sylvain de Langen, Cem Subakan, Peter Plantinga, Yingzhi Wang, Pooneh Mousavi, Luca Della Libera, Artem Ploujnikov, Francesco Paissan, Davide Borra, Salah Zaiem, Zeyu Zhao, Shucong Zhang, Georgios Karakasidis, Sung-Lin Yeh, Pierre Champion, Aku Rouhe, Rudolf Braun, Florian Mai, Juan Zuluaga-Gomez, Seyed Mahed Mousavi, Andreas Nautsch, Ha Nguyen, Xuechen Liu, Sangeet Sagar, Jarod Duret, Salima Mdhaffar, Gaëlle Laperrière, Mickael Rouvier, Renato De Mori, and Yannick Estève. Open-source conversational AI with speechbrain 1.0. *Journal of Machine Learning Research*, 25(333):1–11, 2024. URL http://jmlr.org/papers/v25/24-0991.html.

Thomas Schatz. *ABX-Discriminability Measures and Applications.* Theses, Université Paris 6 (UPMC), Sep 2016. URL https://hal.science/tel-01407461.

Thomas Schatz, Vijayaditya Peddinti, Francis Bach, Aren Jansen, Hynek Hermansky, and Emmanuel Dupoux. Evaluating speech features with the minimal-pair ABX task: Analysis of the classical MFC/PLP pipeline. In *Interspeech 2013*, pp. 1781–1785, 2013. doi:10.21437/Interspeech.2013-441.

Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli. wav2vec: Unsupervised pre-training for speech recognition. In *Interspeech 2019*, pp. 3465–3469, 2019. doi:10.21437/Interspeech.2019-1873.

Ashish Seth, Ramaneswaran Selvakumar, S Sakshi, Sonal Kumar, Sreyan Ghosh, and Dinesh Manocha. EH-MAM: Easy-to-Hard masked acoustic modeling for self-supervised speech representation learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 6386–6400, nov 2024. doi:10.18653/v1/2024.emnlp-main.366. URL https://aclanthology.org/2024.emnlp-main.366/.

Amitay Sicherman and Yossi Adi. Analysing discrete self supervised speech representation for spoken language modeling. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, 2023. doi:10.1109/ICASSP49357.2023.10097097.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation Learning with Contrastive Predictive Coding, jan 2019. URL https://arxiv.org/abs/1807.03748.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. URL http://jmlr.org/papers/v9/vandermaaten08a.html.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Nicol Visser and Herman Kamper. Spoken language modeling with duration-penalized self-supervised units, 2025. URL https://arxiv.org/abs/2505.23494.

Chengyi Wang, Yu Wu, Sanyuan Chen, Shujie Liu, Jinyu Li, Yao Qian, and Zhenglu Yang. Improving self-supervised learning for speech recognition with intermediate layer supervision. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7092–7096, 2022. doi:10.1109/ICASSP43922.2022.9747022.

Guanrou Yang, Ziyang Ma, Zhisheng Zheng, Yakun Song, Zhikang Niu, and Xie Chen. Fast-hubert: An efficient training framework for self-supervised speech representation learning. In *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 1–7, 2023. doi:10.1109/ASRU57964.2023.10389778.

Yifan Yang, Jianheng Zhuo, Zengrui Jin, Ziyang Ma, Xiaoyu Yang, Zengwei Yao, Liyong Guo, Wei Kang, Fangjun Kuang, Long Lin, Daniel Povey, and Xie Chen. k2SSL: A faster and better framework for self-supervised speech representation learning, 2025. URL https://arxiv.org/abs/2411.17100.

Zengwei Yao, Liyong Guo, Xiaoyu Yang, Wei Kang, Fangjun Kuang, Yifan Yang, Zengrui Jin, Long Lin, and Daniel Povey. Zipformer: A faster and better encoder for automatic speech recognition. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=9WD9KwssyT.

Sung-Lin Yeh and Hao Tang. Autoregressive co-training for learning discrete speech representation. In *Interspeech 2022*, pp. 5000–5004, 2022. doi:10.21437/Interspeech.2022-530.

Sung-Lin Yeh and Hao Tang. Estimating the completeness of discrete speech units. In *2024 IEEE Spoken Language Technology Workshop (SLT)*, pp. 415–422, 2024. doi:10.1109/SLT61566.2024.10832198.

S.J. Young, J.J. Odell, and P.C. Woodland. Tree-based state tying for high accuracy modelling. In *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*, 1994. URL https://aclanthology.org/H94-1062/.

Marcely Zanon Boito, Vivek Iyer, Nikolaos Lagos, Laurent Besacier, and Ioan Calapodescu. mHuBERT-147: A compact multilingual HuBERT model. In *Interspeech 2024*, pp. 3939–3943, 2024. doi:10.21437/Interspeech.2024-938.

Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. Soundstream: An end-to-end neural audio codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30: 495–507, 2022. doi:10.1109/TASLP.2021.3129994.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. OPT: Open pre-trained transformer language models, 2022. URL https://arxiv.org/abs/2205.01068.

Xin Zhang, Dong Zhang, Shimin Li, Yaqian Zhou, and Xipeng Qiu. Speechtokenizer: Unified speech tokenizer for speech language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=AF9Q8Vip84.

Table 5: SpidR pretraining hyperparameters. We trained with 16 A100 GPUs in our default setting.

Parameter	Value
Model	
Conv1d dimension	512
Conv1d [(kernel size, stride)]	$[(10,5)] + [(3,2)] \times 4 + [(2,2)] \times 2$
Conv1d bias	False
Conv1d normalization	LayerNorm
Projection dropout	0
Positional encoding layers	5
Positional encoding total kernel size	95
Positional encoding groups	16
Hidden dimension d	768
Number of Transformer layers L	12
Number of attention heads	12
Transformer dropout	0.1
Attention dropout	0.1
Feed-forward dimension	3072
Feed-forward dropout	0
Layer drop probability	5%
LayerNorm mode	After
Q, K, V projection biases	False
Number of codebooks K	8
Codebook decay τ	0.9
Codebook size V	256
Initial decay of teacher β_0	0.999
Decay timescale T	10 000
Decay of teacher at step t	$1 - (1 - \beta_0) \exp(-t/T)$

Parameter	Value
Optimizer	
Name	AdamW
Peak learning rate	5×10^{-4}
Betas	$(0.9, 0.95)$
Weight decay	0.01
Epsilon	1×10^{-6}
Warmup steps	12 000
Hold steps	188 000
Decay steps	200 000
Conv. freeze step	200 000
Data	
Min. sequence length	2000
Max. sequence length	320 000
Max. samples in batch	3 800 000
Number of buckets	1000
Padding	False
Random crop	True
Masking	
Start probability	8%
Span length	10
With overlap	True

A Implementation details

A.1 SpidR pretraining

Table 5 contains the full list of pretraining hyperparameters and figure 5 illustrates the two schedules that occur during training: the learning rate schedule and the EMA decay schedule of the teacher.

The positional encodings of DinoSR and SpidR are the same as those used by Baevski et al. (2022), and differ from Baevski et al. (2020); Hsu et al. (2021). Instead of only one convolutional layer with a large kernel size, they are made of 5 layers, each with a kernel size of $95/5 = 19$.

Batches are sampled using the following procedure. Audio files from LibriSpeech are first sorted and grouped into buckets by length, with only samples within the same bucket shuffled together. Batches are formed by selecting audio files from a given bucket until the target maximum number of samples in a batch is reached. If the target is not met, we continue filling the batch using files from the next bucket. No padding is applied, and audio samples longer than the maximum sequence length are randomly cropped.

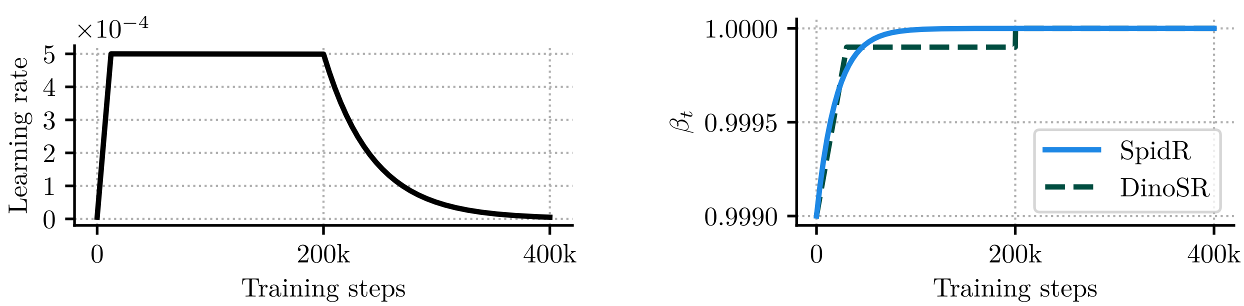


Figure 5: Learning rate schedule (left) and EMA decay schedule of the teacher for DinoSR and SpidR (right).

Table 6: ABX error rate of the codebook predictions (in %, chance level 50%, KL-symmetric distance). All models have codebooks of size 256. The best scores are in **bold** and second best are underlined.

Model	Layer	ABX within speaker ↓		ABX across speaker ↓	
		dev-clean	dev-other	dev-clean	dev-other
DinoSR	5	3.48	<u>3.89</u>	<u>3.80</u>	<u>4.89</u>
DinoSR†	5	<u>3.18</u>	3.92	3.48	4.99
HuBERT + Spin_{256}	-	3.85	5.20	4.32	6.36
WavLM + Spin_{256}	-	4.41	4.67	4.80	5.82
SpidR	6	**2.99**	**3.48**	**3.35**	**4.56**

† Our re-implementation.

A.2 Masking procedure

We follow the masking procedure of Baevski et al. (2020), with parameters of Liu et al. (2023), to sample the mask M, as shown in figure 6. We first extract features $\mathbf{x}$ of shape (n, d) with $d = 768$ from the audio signal using the downsampling module. The masking process works as follows: each frame $i \in \{1, ..., n\}$ has an 8% probability of starting a mask span of length 10. Mask spans can overlap, and the proportion of masked frames depends on the total number of frames n.

Using the parameters from table 5, the average sequence length in a LibriSpeech 960h batch is 216000, corresponding to 13.5 seconds of audio and to $n = 675$ frames. For a typical 13.5-second audio sample, approximately 43% of all time-steps are masked, with an average span length of 11.9 frames, corresponding to 238ms of audio, a median of 8 frames, and a maximum of about 50 frames. For reference, the average triphone duration in LibriSpeech `dev-clean` and `dev-other` is 237ms, based on the annotations from Nguyen et al. (2020).

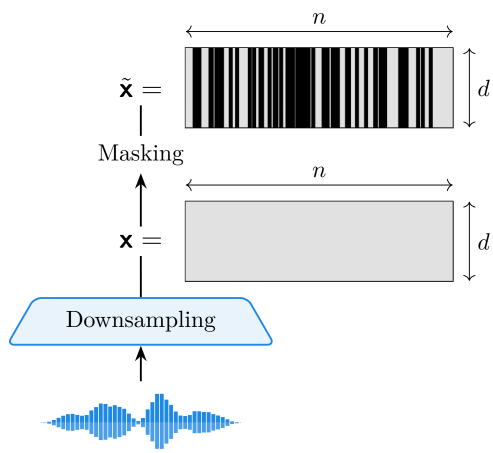


Figure 6: Masking procedure. The masked frames are in black, and the unmasked ones in gray.

B Additional results

B.1 Discriminability of continuous embeddings

In table 6, we compute ABX discriminability on the softmax outputs from either the prediction heads or the Spin codebooks. Instead of using the standard angular distance, we use the symmetrized KL divergence. This metric was used in Liu et al. (2023) to evaluate DinoSR. We select the Spin checkpoints from Chang et al. (2023) with the same codebook size as DinoSR and SpidR.

We evaluate the phoneme and word discriminability of continuous embeddings from a wide range of monolingual English speech models in table 7. All Base size models were trained on LibriSpeech and all Large models on Libri-Light. For each model, we select the best-performing layer in terms of average ABX score. We distinguish between standard self-supervised models (including SpidR), self-supervised models with additional robustness losses such as WavLM (Chen et al., 2022), and supervised models. We conducted a preliminary experiment where we fine-tuned SpidR with our own implementation of Spin.

Overall, masked prediction using discrete targets produces representations with salient phonetic information, and additional losses promoting invariance to acoustic and speaker conditions further improve performance. Supervision does not necessarily help—Whisper exhibits poor ABX scores, likely because it learns from multiple tasks simultaneously, making phonetic information less salient in its encoder representations.

Table 7: Evaluation of representations from monolingual English speech models. All operate at a 50Hz framerate, except Conformer which is at 25Hz. Apart from Whisper, all Base size models were trained on LibriSpeech and all Large ones on Libri-Light. The best scores are in **bold** and second best are underlined.

Model	Layer	ABX within speaker ↓		ABX across speaker ↓		MAP words ↑	
		dev-clean	dev-other	dev-clean	dev-other	dev-clean	dev-other
Self-supervised models							
wav2vec 2.0 (Baevski et al., 2020)	6	4.47	5.63	5.25	7.82	44.81	31.92
wav2vec 2.0 Large (Baevski et al., 2020)	16	4.35	5.14	5.20	7.29	46.10	34.21
HuBERT (Hsu et al., 2021)	11	3.38	4.26	4.01	6.49	46.07	33.37
HuBERT Large (Hsu et al., 2021)	24	3.90	4.30	4.49	5.92	**68.99**	**59.11**
HuBERT Extra Large (Hsu et al., 2021)	48	4.04	4.38	4.76	6.21	64.64	54.64
data2vec (Baevski et al., 2022)	4	4.41	5.49	5.07	7.40	39.34	27.90
data2vec Large (Baevski et al., 2022)	7	4.51	5.46	5.20	7.15	38.56	28.70
data2vec 2.0 (Baevski et al., 2023)	1	5.13	5.77	5.72	7.53	69.38	53.49
data2vec 2.0 Large (Baevski et al., 2023)	2	6.68	6.55	7.41	8.43	66.85	55.42
Eh-MAM (Seth et al., 2024)	1	4.31	5.36	4.96	7.58	60.63	42.80
DinoSR (Liu et al., 2023)	5	4.05	5.11	4.72	7.29	63.02	45.86
DinoSR† (Liu et al., 2023)	5	4.29	5.56	5.22	8.56	51.35	33.74
SpidR	6	**3.32**	**3.74**	**3.66**	**4.95**	66.50	55.26
With self-supervised robustness loss							
WavLM Base (Chen et al., 2022)	11	3.03	3.71	3.50	5.21	71.57	58.09
WavLM Base+ (Chen et al., 2022)	12	3.54	4.08	4.08	5.82	62.77	51.63
WavLM Large (Chen et al., 2022)	24	3.94	4.32	4.62	5.99	67.56	57.49
ContentVec$_{100}$ (Qian et al., 2022)	12	3.29	4.04	3.83	5.49	63.85	52.17
HuBERT + Spin$_{2048}$ (Chang et al., 2023)	12	**2.70**	**3.23**	**3.05**	**4.08**	68.70	61.41
WavLM + Spin$_{2048}$ (Chang et al., 2023)	12	3.05	3.51	3.52	4.44	**75.20**	**67.13**
SpidR + Spin$_{2048}$	12	2.73	3.51	3.11	4.47	61.33	50.99
With supervision							
Whisper small.en (Radford et al., 2023)	8	7.03	8.21	8.27	11.81	16.35	11.03
Conformer ASR‡ (Gulati et al., 2020)	8	2.79	3.76	3.21	5.22	63.03	45.74
wav2vec 2.0 ASR (Baevski et al., 2020)	6	3.94	4.90	4.50	6.52	56.98	43.52
wav2vec 2.0 Large ASR (Baevski et al., 2020)	10	3.52	4.58	4.09	6.20	54.04	40.06
HuBERT Large ASR (Hsu et al., 2021)	14	4.69	5.38	5.48	7.27	43.25	33.09
HuBERT + phoneme classif. (Poli et al., 2024)	12	**0.82**	**1.54**	**0.97**	**2.35**	**68.48**	**57.04**

† Our re-implementation.

‡ Using `speechbrain/asr-conformer-transformerlm-librispeech` (Ravanelli et al., 2024).

We compare in figure 7 the ABX and MAP on continuous embeddings by layer for HuBERT, DinoSR (both the original checkpoint and our replication) and SpidR. The ABX scores are averaged across subsets and speaker conditions, and MAP across the two subsets.

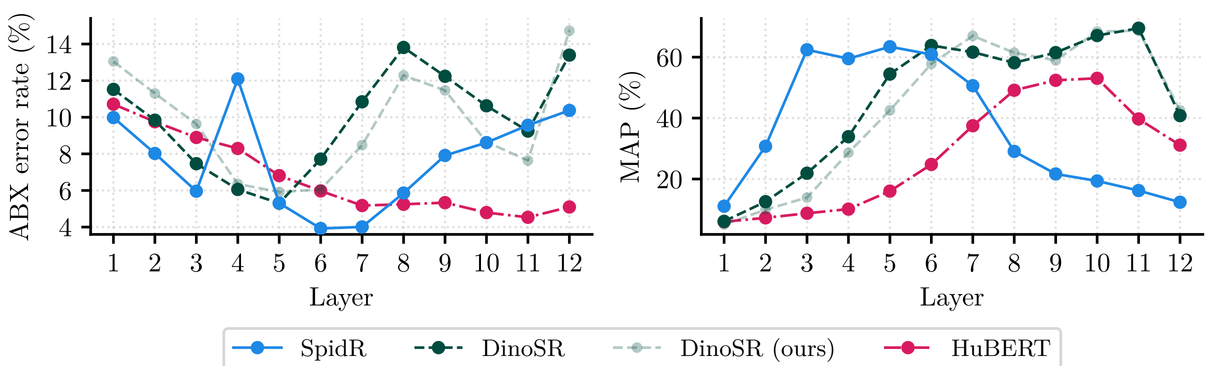


Figure 7: ABX and MAP (in %, chance level 50% for ABX) by layer for SpidR, DinoSR and HuBERT.

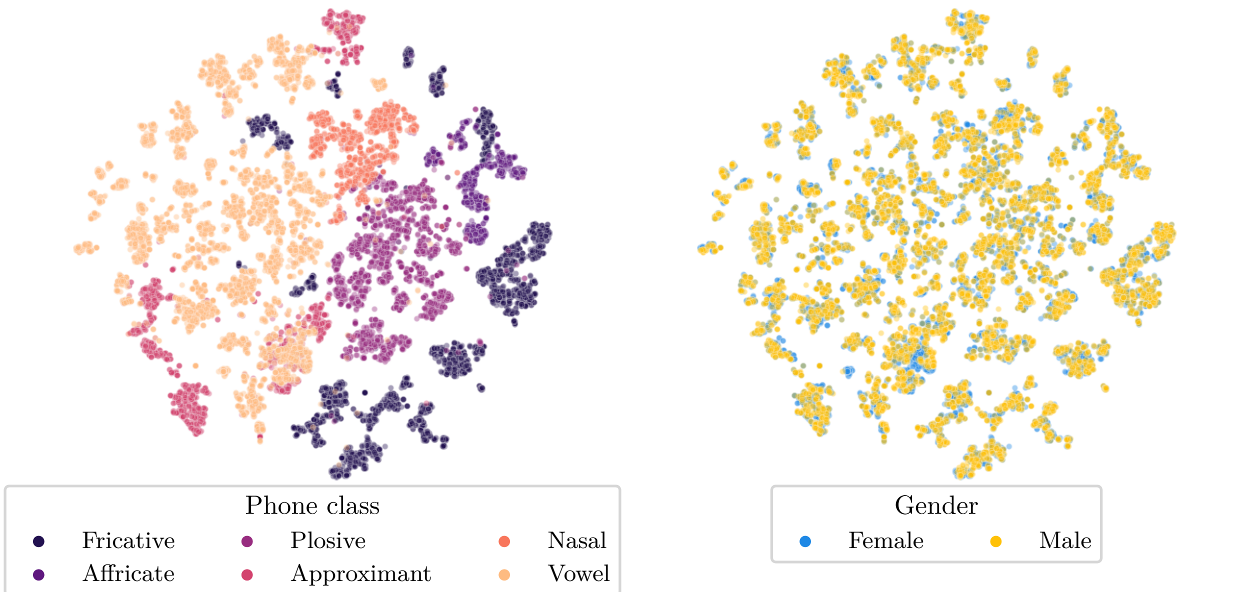


Figure 8: t-SNE visualization of phone embeddings from SpidR layer 6 on LibriSpeech `dev-clean`. Embeddings are colored by phone class (left) and by speaker gender (right).

B.2 Embeddings visualization

We visualize the embedding space of SpidR in two dimensions using t-SNE (van der Maaten & Hinton, 2008), following de Seyssel et al. (2022). We train t-SNE on phone embeddings of LibriSpeech `dev-clean` from layer 6 of SpidR. For each speaker, we sample 10 instances per phone and average each embedding along the time dimension, resulting in approximately 15 000 samples.

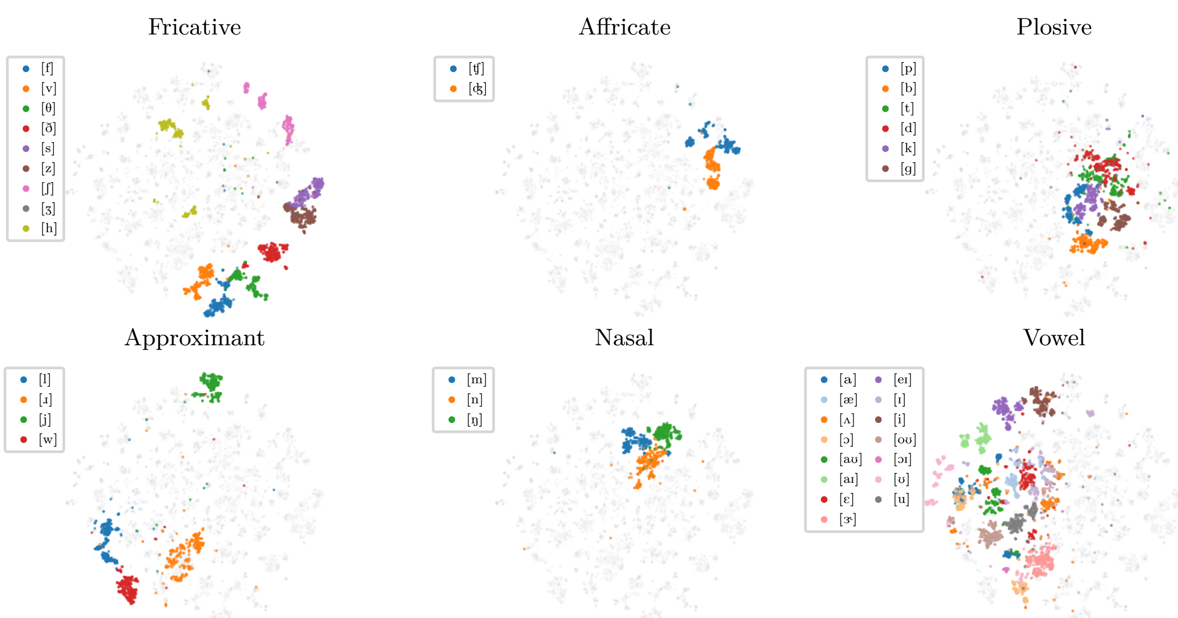
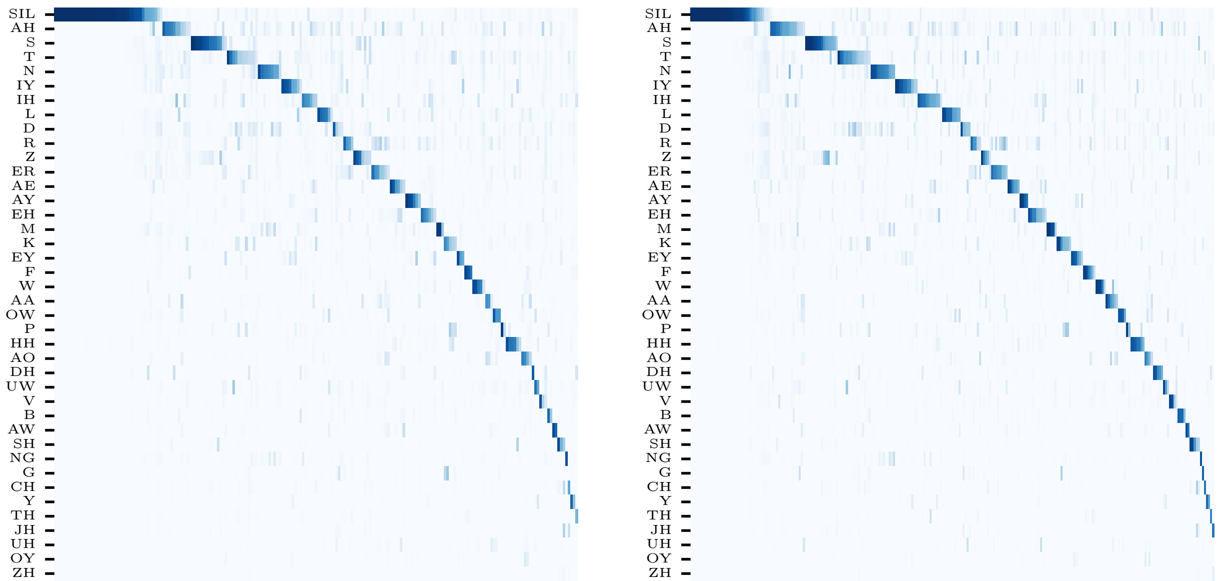


Figure 9: t-SNE visualization of phone embeddings from SpidR layer 6 on LibriSpeech `dev-clean`, colored by individual phones within each phone class. Embeddings from other classes are shown in gray.



(a) With codebook predictions (203 active codes). (b) With K-means quantization (256 active codes).

Figure 10: $\mathbb{P}(\text{phone} \mid \text{code})$ visualization for SpidR layer 6 using either codebook predictions (left) or K-means quantization (right), on LibriSpeech `dev-clean` and `dev-other`.

In figure 8, we color the embeddings by either the underlying phone class or by the speaker gender. For more fine-grained visualization, we color by individual phones within each phone class in figure 9. Overall, the embedding space is well clustered by phone class, and even by individual phone, whereas the speaker information is not directly extractable from the embeddings.

B.3 Layer-wise analysis

In addition to the discrete units analysis in table 2, we compute in figure 11 the ABX discriminability and PNMI for other intermediate layers of SpidR and HuBERT, with units derived from codebook predictions or K-means quantization. As in section 4.1, the K-means are trained on LibriSpeech `train-clean-100`. Figure 10 shows the $\mathbb{P}(\text{phone} \mid \text{code})$, with codes from SpidR layer 6 on LibriSpeech `dev-clean` and `dev-other`. The vertical axes are sorted by phone frequency in the annotated data.

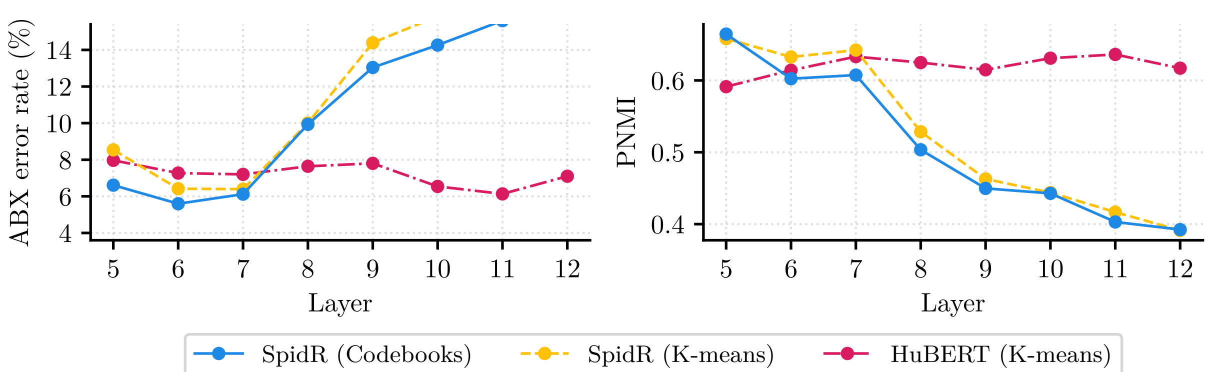


Figure 11: ABX (in %, chance level 50%) and PNMI by layer on discrete units from SpidR using codebook predictions or K-means, and from HuBERT using K-means, with $V = 256$ units. ABX scores averaged across subsets and speaker conditions, and PNMI computed on LibriSpeech `dev-clean` and `dev-other`.

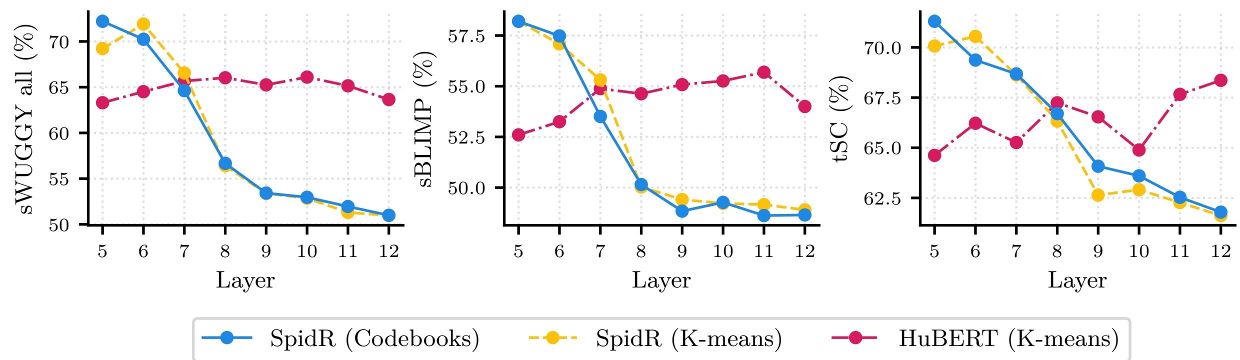


Figure 12: Zero-shot spoken language modeling from each layer of HuBERT and SpidR (in %, chance level 50%), with units from codebook predictions or from K-means quantization, with $V = 256$ units.

We also trained spoken language models from the units obtained from each intermediate layer in the same conditions as section 4.1. Figure 12 shows the accuracies on zero-shot spoken language modeling for the three encoders. Finally, to assess how well the zero-shot metrics serve as proxy tasks, we compare spoken language modeling scores against phonetic- and word-level metrics in figure 13 (continuous embeddings) and figure 14 (discrete units). We distinguish between SpidR using K-means units, where ABX is computed on standard embeddings, and SpidR using codebook predictions, where ABX is computed on codebook predictions with symmetric KL divergence. We compute Pearson correlation coefficients between each proxy metric and downstream evaluation score. Note that this analysis does not capture inter-model differences well, and that correlations are influenced by the fact that SpidR's final layers perform poorly across most metrics.

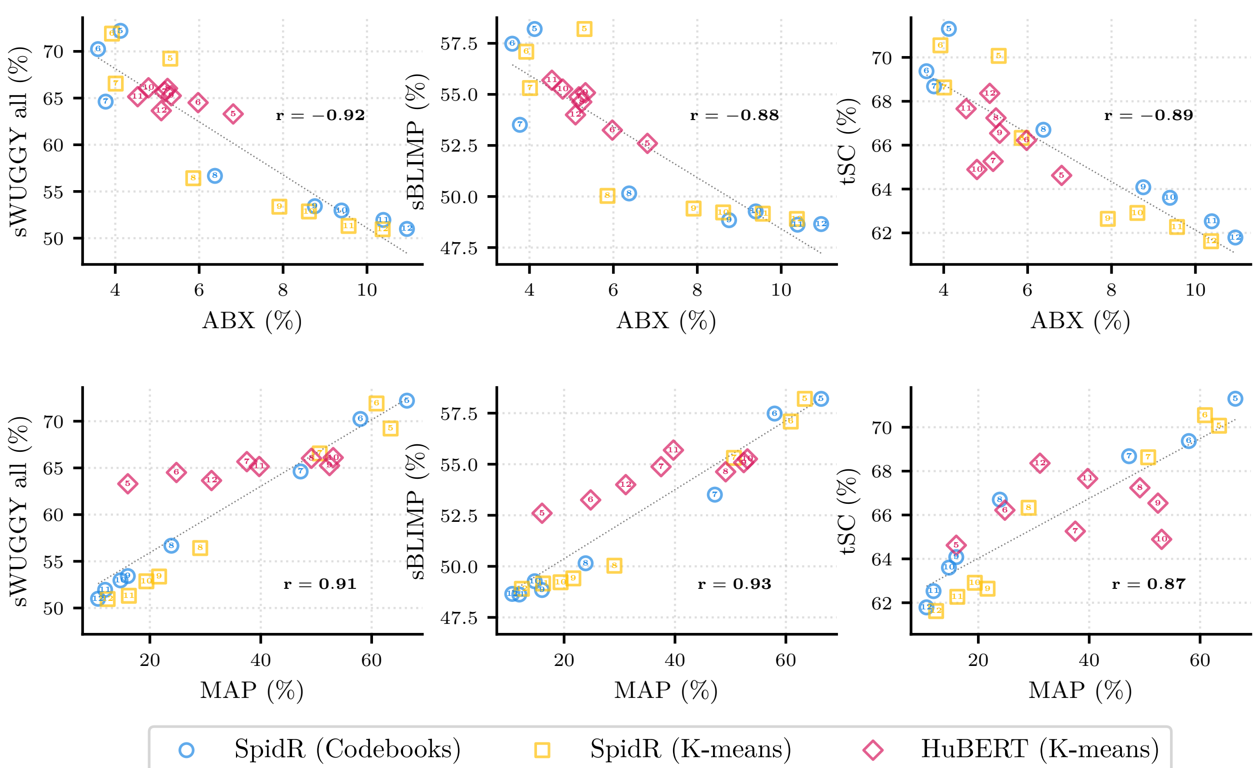


Figure 13: Spoken language modeling against discriminability of the continuous representations. Dots are labeled by intermediate layer index. ABX for SpidR (Codebooks) is computed over codebook predictions.

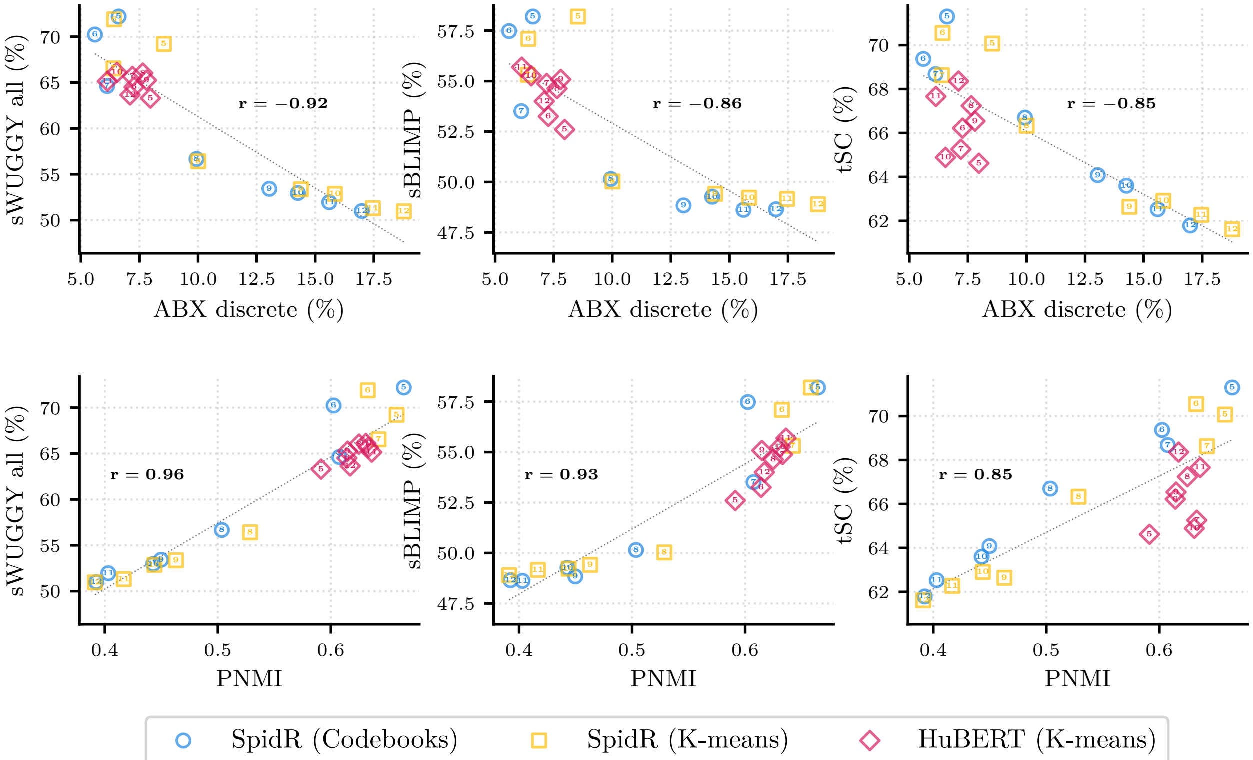


Figure 14: Spoken language modeling against phonetic evaluation of the discrete units, with $V = 256$ units. Dots are labeled by intermediate layer index.

C Ablation study

Table 8: Ablation study from DinoSR to SpidR (in % except for PNMI, chance level 50% for ABX). The ABX scores are averaged across subsets and speaker conditions, MAP across the two subsets, and PNMI is computed on LibriSpeech `dev-clean` and `dev-other`.

Model	Layer	Continuous embeddings		Discrete units quality	
		ABX ↓	MAP ↑	ABX ↓	PNMI ↑
DinoSR[†]	5	5.91	42.55	7.70	0.614
DinoSR + Heads	7	4.22	63.39	7.87	0.610
DinoSR + Exp. EMA	6	5.86	51.85	8.77	0.609
DinoSR + Heads + Exp. EMA = SpidR	6	3.92	60.88	6.32	0.602

[†] Our re-implementation.

We developed SpidR by making two key changes from DinoSR. First, we modified the learning objective by adding prediction heads to the student's intermediate layers instead of using only the final layer. This showed promising results, but we noticed that the training loss would slightly increase mid-training, suggesting that the student was struggling to keep up with the teacher. To solve this problem, we modified the teacher's EMA decay schedule to follow a smoother trajectory that approaches 1 faster without plateauing at 0.9999.

In table 8 we ablate these two changes: "Heads" refers to the new learning objective, and "Exp. EMA" refers to the new shape of the EMA decay schedule. We evaluate both in terms of continuous embedding discriminability and discrete units quality.