GENERATIVE SPOKEN LANGUAGE MODEL BASED ON CONTINUOUS WORD-SIZED AUDIO TOKENS

Anonymous authors
Paper under double-blind review

ABSTRACT

In NLP, text language models based on words or subwords are known to outperform their character-based counterparts. Yet, in the speech community, the standard input of spoken LMs are 20ms or 40ms-long discrete units (shorter than a phoneme). Taking inspiration from word-based LM, we introduce a Generative Spoken Language Model (GSLM) based on word-size continuous-valued audio tokens that can generate diverse and expressive language output. This is obtained by replacing lookup table for lexical types with a Lexical Embedding function, the cross entropy loss by a contrastive loss, and multinomial sampling by k-NN sampling. The resulting model is the first generative language model based on word-size continuous tokens. Its performance is on par with discrete unit GSLMs regarding generation quality and zero resource challenge metrics. Moreover, it is five times more memory efficient thanks to its large 200ms units. In addition, the embeddings before and after the Lexical Embedder are phonetically and semantically interpretable[1].

1 INTRODUCTION

Recent work has opened up the possibility of learning generative language models directly from the raw audio signals, without using either text or Automatic Speech Recognition (ASR) (Lakhotia et al., 2021; Kharitonov et al., 2021; Nguyen et al., 2022b; Borsos et al., 2022). The basic idea of these model is to rely on traditional text-based language models (LM), but replacing the text input with some other discrete tokens directly learned from audio in an unsupervised fashion. The advantage of learning units from speech instead of relying on ASR is that this procedure can capture non-verbal vocalizations (like laughter) or intonation and rhythm which are typically not transcribed, resulting in more expressive generations (Kreuk et al., 2021; Kharitonov et al., 2021). In addition, ASR may not be available in many languages that have insufficient textual resources and can make errors, which may then perturb the learning of the LM.

The problem of using self-discovered units, however, is that these units are typically very small, in fact, usually smaller than phonemes (Lakhotia et al., 2021; Borsos et al., 2022). We think that increasing the size of the units will favorably impact the semantic capabilities of a downstream spoken LM. This intuition comes from the NLP literature. Among others, Graves (2013); Mikolov et al. (2011); Bojanowski et al. (2015); Nguyen et al. (2022a) have shown a performance gap between character-based LM and word-based LM. The main reason is that at the level of characters, it is difficult for a text LM to extract long range syntactic and semantic relationships. This is one of the reason why recent state-of-the-art text-based LM (Radford et al., 2019) typically use a tokenizer representing word or subword units. Another advantage of large units is to save GPU memory at training time that enable to use both larger batch and longer sequences.

In speech, building the equivalent of a text-based tokenizer is hampered by two difficulties. First, the *boundary problem* is that, contrary to text in most orthographic systems, speech does not have spaces and punctuation to delimit between word units. Finding word boundaries from raw audio is itself a difficult challenge (Dunbar et al., 2022a). Second, the *clustering problem*, is that even if boundaries were available, speech is variable and the same word may surface in a variety of forms depending on speaker, accent, speech rate, etc. This problem may be even more difficult to solve than the first

[1] Audio examples are available at our anonymous website

one (Dunbar et al., 2022a) because of the highly skewed distribution of word frequencies (Algayres et al., 2022b). Here, we investigate the possibility to build a *continuous tokenizer* that sidesteps these two problems by using tokens that have neither perfect boundaries, nor require a clustering step.

Having a continuous tokenizer instead of a discrete one result in drastic changes from the point of view of the downstream LM. With a discrete tokenizer, one can define a finite list of tokens over which the LM can learn a lookup embedding table at the input of the model, and use a softmax layer at the output of the model. The softmax is used in training mode to compute the loss function through a cross entropy with the target token and at inference time to sample sentences. With continuous representations, the list of tokens is unbounded, making these computations intractable. We tackle this problem with a *Lexical Embedder*, a semi-learnable function that maps continuous tokens to a practically infinite list of embeddings.

The key question addressed in this paper is whether it is possible to generate speech using large (word sized) continuous units instead of short discrete ones. Our major technical contribution is to replace the three standard elements of a text-based LM (lookup table, cross-entropy loss function, multinomial sampling) with elements adapted to a virtually infinite list of continuous tokens. We show that with these changes, it is possible to generate speech of the same quality as discrete units models. This is interesting because our units are 200ms long which amounts to a 5 time memory reduction compared to regular discrete units (Lakhotia et al., 2021; Borsos et al., 2022), opening up the possibility to train spoken LMs on longer speech sequences. In addition, our model builds interpretable representations thanks to the Lexical Embedder which learns a mapping between an acoustic space, with phonetic properties, to a lexical space, with semantic and syntactic properties. We call the resulting model tGSLM (token-based GSLM).

2 Related work

Unsupervised speech representations like CPC, Wav2vec2.0 and HuBERT (van den Oord et al., 2018; Baevski et al., 2020; Hsu et al., 2021) are fixed-size representation (10 to 20ms long) that outperform traditional features, like mel-filterbanks and MFCCs, in many applications (Yang et al., 2021). In parallel to these works, there is a growing literature on variable-length acoustic encoding called speech sequence embeddings (SSE) (Algayres et al., 2022a; Jacobs et al., 2021; Kamper, 2018; Settle & Livescu, 2016). SSE models take a sequence of speech of any length and return a fixed-size vector. These models encode speech by maximizing phonetic information while minimizing speaker identity and recording conditions. SSEs are used for spoken term discovery (Thual et al., 2018), speech segmentation into phones or words (Kamper, 2022; Algayres et al., 2022b) but also as input to a BERT model (Algayres et al., 2022b) for spoken language modelling.

Speech generation is often performed with a neural vocoder conditioned on mel-filterbanks (van den Oord et al., 2016; Kumar et al., 2019; Kong et al., 2020; Prenger et al., 2018). In a text-to-speech pipeline, the mel-filterbanks are obtained with another neural network, which is conditioned on text (Ping et al., 2017; Shen et al., 2018). In the next step, the mel-filterbanks are decoded into natural sounding speech by a neural vocoder (van den Oord et al., 2016; Kumar et al., 2019; Kong et al., 2020; Prenger et al., 2018). For the Zerospeech Challenge 2019, Dunbar et al. (2019) proposed to remove text and replace it with unsupervised discrete units. This challenge has fueled a large body of works on learning low bitrate speech representations for speech compression, voice conversion and spoken language modelling (Chen & Hain, 2020; Liu et al., 2019; Feng et al., 2019; Baevski et al., 2019; Tjandra et al., 2019; Kharitonov et al., 2021; Lakhotia et al., 2021; Nguyen et al., 2020). For evaluation, the Zero-Resource challenge used bitrate and human evaluation.

Spoken Language Model are neural networks trained to predict missing parts of a spoken sentence with predictive or contrastive losses. GSLM (Lakhotia et al., 2021) is the first spoken LM able to generate expressive and consistent spoken sentences in a pure textless fashion. It uses a causal transformer LM trained with NLL loss on sequences of discrete units obtained with a k-means clustering (with k=100) of HuBERT frames. Once trained, GSLM can generate a sequence of discrete units by multinomial sampling that is decoded into speech with a separate vocoder. Specifically, the sampled HuBERT units are mapped to mel-filterbanks with Tacotron2.0 and decoded into speech with *WaveGlow* (Prenger et al., 2018), a neural vocoder. Lakhotia et al. (2021) also provide a way to evaluate their spoken LM using an ASR to transcribe their spoken generations and an external LM to compute the perplexity of the resulting transcriptions. In addition, the Zerospeech Challenge

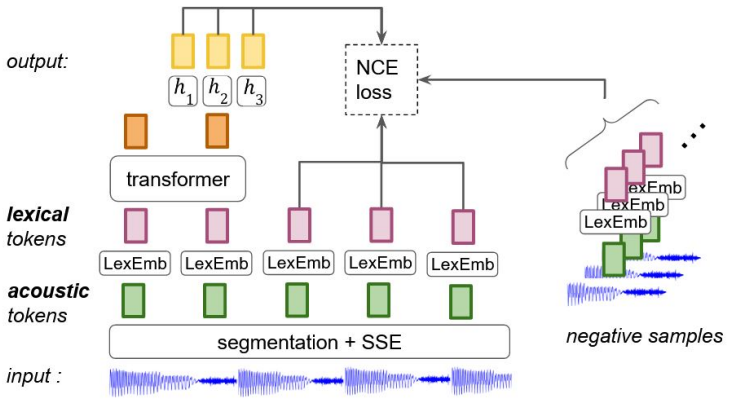

Figure 1: Speech is encoded into Wav2vec2.0 frames and segmented into chunks. These latter are converted into acoustic tokens with an SSE model, and turned into lexical tokens by applying the function *LexEmb*. Finally, lexical tokens are fed to a causal transformer LM which attempts to predict the first, second, and third following tokens using parallel output heads. The acoustic tokens are pre-extracted before training the learnable modules ($LexEmb$, the transformer and the final FCs) with the NCE loss. The negative samples are chosen randomly from other utterances of the same speaker.

2021 (Nguyen et al., 2020) designed a set of zero-shot metrics to probe what spoken LMs learn. A recent paper (Borsos et al., 2022), audioLM, came to our attention, which we did not have the time to include in our experiments. AudioLM works similarly to GSLM yet with the ability to generate speech that preserves the identity of the speaker. In another line of work, Algayres et al. (2022b) trained a BERT model with a contrastive loss function on sentences represented as series of SSEs. They showed the resulting BERT is able to model semantics and syntax. This work suggests that discrete tokenizer and the NLL loss are not necessary to tackle language modelling on speech. We take inspiration on their work to design our approach.

3 Approach

3.1 tGSLM: training

The general structure of tGSLM is presented in Figure 1. It is composed of an **encoder** which segments the input speech into sequences of possibly varying size, and compute a fixed sized Speech Sequence Embedding (SSE), which we call acoustic tokens (Section 3.1.1). These tokens are turned into lexical tokens through a learnable **Lexical Embedder** (Section 3.1.2), and fed into a causal **Language Model** that has been modified to deal with continuous inputs (Section 3.1.3).

3.1.1 Acoustic tokens

In Figure 1, a speech sequence, S, is turned into a n acoustic tokens, $(a_0, ..., a_n)$, after applying speech segmentation and an SSE model.

Speech segmentation consists in finding word boundaries in a speech sentence (Algayres et al., 2022b; Kamper, 2022; Kreuk et al., 2020). In this work, we rely on a naive method by placing a boundary every 200 ms, regardless of the content of the speech signal. In the results section, we show that this method leads to similar results than recent, more complex speech segmentation systems.

The acoustic tokens $(a_i)_{i \leq n}$ are built by first encoding the speech sentence S into a series of n' frames $(f_i)_{i \leq n'}$ with the 8th layer of Wav2vec2.0 Base from Baevski et al. (2020). For any two boundaries (k, l), $a_i = SSE([f_k, ..., f_l])$ where SSE is a self-supervised system from Algayres et al. (2022a) trained with contrastive learning. This model has state-of-the-art performances on phonetic representation of pre-segmented words as measured by the Mean-Average-Precision met-

ric. The acoustic tokens are extracted in a preprocessing step and stored before the training of the subsequent LM.

3.1.2 LEXICAL TOKENS

In a text-based transformer LM, there is often a linear FC layer before the transformer, with the size of the vocabulary, that maps discrete word tokens to lexical tokens (Vaswani et al., 2017). These lexical tokens, also known as word embeddings (Mikolov et al., 2013), learn during training semantic and syntactic properties that have been measured extensively in the NLP literature. In our case, the situation is different. First, instead of discrete word tokens, our LM takes as input continuous acoustic tokens which latent vocabulary size is unknown. Second, the mapping between acoustic and lexical space cannot be linear, as two speech segments may sound the same, i.e. be close in the acoustic space, while being semantically/syntaxically different, i.e. far in the lexical space. This highly non-linear function between acoustic and lexical space is learned by our lexical embedder: $LexEmb = L \circ q$ function. L is a stack of non-linear FC layers learned jointly with the LM. q is an information bottleneck quantization function that we had to introduce to minimize the presence of low-level non-linguistic acoustic information. For a speech sequence S composed of n tokens, we note the sequence of lexical tokens $(l_i)_{i \leq n}$ such as $\forall i \leq n, l_i = LexEmb(a_i)$.

To understand why we need q, we have to go back to the LexEmb function input: the acoustic tokens. The acoustic tokens are derived from Wav2vec2.0, which is a transformer architecture whose attention mechanism covers the whole sentence. Each wav2vec2 frame therefore contain potential information about relative positions (through the transformer's positional embeddings), adjacent acoustic materials (through self attention) or global properties like speaker. What we've found in preliminary experiments is that this information may leak into the acoustic tokens and be amplified by the prediction or contrastive loss of the downstream causal LM. Fortunately, it turns out that this information has low variance and can be partially removed by slightly degrading the quality of the acoustic tokens. The degradation of the acoustic tokens is the role of the function q. q is composed of a PCA reduction and a quantization step that we call *d-k-means*, that stands for per-dimension k-means. Specifically, given a speech database that has been segmented and encoded into N acoustic tokens, $(a_i)_{i \leq N}$, we reduce their dimensions to d with a PCA. Then, we train d different k-means, one for each dimension of the PCA. In other words, for each $j \leq d$, we train a k-means on $(PCA(a_i)[j])_{i \leq N}$. We chose the number of centroids per k-means to be proportional to the explained variance of each of the PCA dimensions. Once the k-means are trained, each dimension of each acoustic tokens is mapped to its cluster id. Finally, the cluster ids are turned into onehot vectors and concatenated into one vector (see Appendix A.1 for more detailed explanations). d-k-means is inspired from multi-stage vector quantizer (VQ) (Vasuki & Vanathi, 2006) where several VQ codebooks are learned in parallel as in Baevski et al. (2020); Zeghidour et al. (2021). The PCA and the d-k-means are trained over the whole training set as a preprocessing step, before the transformer LM. We ablate the use of q in Appendix A.1 and show that it is necessary for the LM to generate sentences[2].

3.1.3 CAUSAL LANGUAGE MODEL

The LM is a standard causal transformer with two modifications: the loss function and the prediction heads. First, in a standard LM, the number of possible types is fixed beforehand and remains tractable even for a very large corpus (10k to 100k). Here, because the number of different lexical tokens is virtually infinite, we cannot use a standard softmax and cross entropy loss. Instead, we use a contrastive loss: the NCE loss[3]. This loss works by maximizing the similarity between a pair of positive samples while minimizing the similarity between the positive samples and various negative samples. However, even though the SSE model from Algayres et al. (2022a) has learned to be speaker invariant, there is still a lot of speaker-related information encoded into the acoustic tokens. This is a problem already encountered in Algayres et al. (2022a); van den Oord et al. (2018) that is dealt with by sampling the negative tokens from the same speaker as the positive tokens.

[2]Due to this quantization step, the resulting vectors (PCA+ d-k-means) could in principle be mapped to a finite dictionary of tokens, but, in practice, there is little or no collision and the number of classes remains identical to the number of tokens, i.e., way too high to apply a softmax.

[3]Using L2 reconstruction with an additional decoder instead of contrastive learning did not work for us

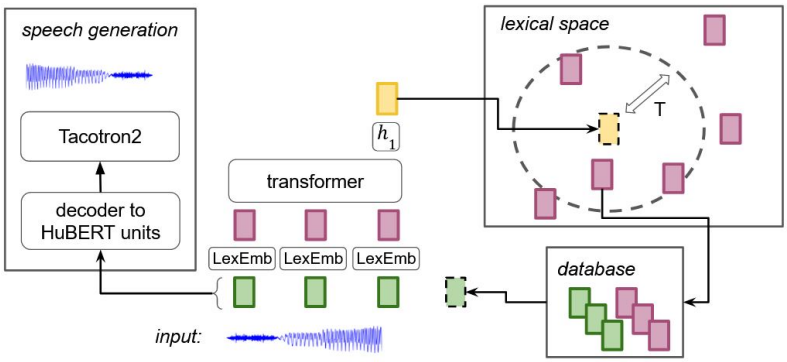

Figure 2: Our sampling procedure. Given a list of audio files unseen during training, N random speech segments are stored in their acoustic and lexical forms: $(a_i, l_i)_{i \leq N}$. In addition, a *lexical space* is created by indexing $(l_i)_{i \leq N}$ into a k-NN graph. Given a speech prompt, segmented and encoded into $(a_0, ..., a_t)$, we do a forward pass in tGSLM and search for the nearest neighbors of h_1 output in the lexical space. l_{t+1} is sampled and its corresponding a_{t+1} is appended to $(a_0, ..., a_t)$. When a final a_T token is sampled, $(a_0, ..., a_T)$ is decoded into HuBERT units and speech is generated with Tacotron2.

Second, in a standard LM, the output head typically predicts the next word. However, in the case of speech, the boundary between individual phonemes is blurred by coarticulation. It is therefore easy to predict the next word by just attending to very local acoustic information at the end of the last word (something impossible to do with characters which are sequentially disentangled). We therefore introduce three prediction heads (three linear FC layers: h_1,h_2,h_3) which do not only predict the first next token, but also the second and third as they cannot be co-articulated with the last token encoded by the LM. These prediction layers are trained jointly with the LM. We justify the choice of three prediction heads with a gridsearch available in appendix at Table 6.

3.2 TGSLM: GENERATION

Once tGSLM training is over, we use it to generate full spoken sentences. We do that in two steps: we generate a sequence of acoustic tokens (Section 3.2.1) and then decode this sequence into speech (Section 3.2.2.

3.2.1 SAMPLING

To generate a spoken sentence, we take inspiration of the popular top-k sampling method used in NLP to generate text sentences. This method requires to sample series of word tokens by sampling among the most probable word types. In our case, we do not have access to types so we are going to sample among the most probable lexical tokens. Our sampling method is summarized at Figure 2. We start by collecting a few dozens of hours of speech that have not been seen during tGSLM training. The utterances are segmented and encoded into N speech segments and stored into their acoustic and lexical forms: $(a_i, l_i)_{i \leq N}$. We index $(l_i)_{i \leq N}$ into a k-NN graph called the lexical space. Given a prompt of t acoustic tokens $(a_0, ..., a_t)$, we do a forward pass into tGSLM. Then, we compute the cosine similarity of h_1 output and its k closest neighbors in the lexical space. We apply a softmax on the vector of cosine similarities and treat it as a multinomial distribution to sample one element: l_{t+1}. The softmax function contains a temperature parameter that controls the range of the sampling area. The acoustic tokens a_{t+1} that correspond l_{t+1} is retrieved from the stored database and appended to $(a_0, ..., a_t)$. Once the desired length is reached, the sequence of acoustic tokens is decoded into a spoken sentence as explained in the next section.

3.2.2 SPEECH GENERATION

Lakhotia et al. (2021); Kharitonov et al. (2022) trained a Tacotron2.0 decoder (Shen et al., 2018) to map deduplicated HuBERT units into mel filterbanks. Then, speech is generated from the mel

filterbanks by a *WaveGlow* vocoder (Prenger et al., 2018). In order to make use of this pretrained Tacotron2.0 decoder, we trained an encoder-decoder transformer model to map series of acoustic tokens to series of HuBERT units. During training, the encoder computes an attention over a series of acoustic tokens while the decoder predicts HuBERT units auto-regressively. At inference, given a series of acoustic tokens, a corresponding sequence of HuBERT units is obtained by taking the argmax of the decoder softmax function. Finally, the HuBERT units are given as input to the pretrained Tacotron2.0 to be decoded into spoken utterances.

4 METHODS

4.1 DATASETS

LJ Speech (LJ), LibriSpeech (LS), Libri-light 6k clean (LL6k-clean) are three corpora of studio recordings of read English of respectively 24, 1k and 6k hours (Ito & Johnson, 2017; Panayotov et al., 2015; Rivière & Dupoux, 2021). These corpora are used to train the different parts of the pipeline. The training details and specific model architectures can be found in Appendix Section A.2.

4.2 GENERATION TASK

To evaluate the overall quality of generated spoken sentences, Lakhotia et al. (2021) use text-based metrics by transcribing the generations with an external ASR system[4]. Two scores, called PPX/VERT, are computed on the batch of transcribed speech. The perplexity score (PPX) is obtained with an external transformer LM [5] trained on the English NewsCrawl dataset. The diversity (VERT) score is the average of self-BLEU (Zhu et al., 2018) and auto-BLEU (Lakhotia et al., 2021) scores. As sentence generation is conditioned on a temperature parameter, there is not one single PPX/VERT score for a spoken LM. Typically, low temperatures produce high VERT and low PPX, whereas high temperatures produce low VERT and high PPX. Lakhotia et al. (2021) chose to compare the performance of their spoken LM with the PPX/VERT obtained on a batch of text from the LJ corpus. We propose to add a second harder comparison point with a batch of text from the LibriSpeech, a lexically richer corpus. Also, VERT scores are dependent on number of words present in the batch of generated sentences, a parameter that Lakhotia et al. (2021) did not control. We propose to use batches of 100 sentences of 30 words each (a compromise between acceptable generation time and low variance). For that batch size, the PPX/VERT on LJ, written LJ-VERT, is $140/0.189$ while PPX/VERT on LibriSpeech, LS-VERT, is $182/0.113$.

4.3 ZERO-SHOT TASKS

$sWUGGY$ and $sBLIMP$ are zero-shot tasks to evaluate spoken language models introduced in the Zerospeech Challenge 2021 (Nguyen et al., 2020):. These metrics are inspired by psycholinguistics and are used for interpreting what spoken LM learns. $sWUGGY$ is a list of pairs of word/non-word synthesized with the Google TTS API and filtered for the word that are in the LibriSpeech training set. $sBLIMP$ is list of pairs of syntaxically correct/incorrect synthesized sentences. Both $sWUGGY$ and $sBLIMP$ require the spoken LM to attribute a highest probability to the correct element in each pair. Probabilities are computed by applying the spoken LM training loss directly on the test items.

ABX_{sem} and ABX_{POS} are additional zero-shot tasks introduced in Algayres et al. (2022b) to evaluate semantic encoding and Part-Of-Speech (POS) tagging, this time not based on probabilities but on distances between embeddings. An ABX task is a list of triplets A,B and X where A and B belong to the same category and X is a distractor. The task is to encode the triplet with a distance d and show that $d(A, B) < d(A, X)$. In this case, A,B and X are spoken words given in the context of a sentence. For ABX_{sem}, A and B are close semantically and X is random. For ABX_{POS} A and B share the same POS tag and X has different POS tag.

Normalised Edit Distance (NED) introduced in Versteegh et al. (2016) is a term discovery task

[4]ASR transcripts are obtained with a pretrained large Wav2Vec 2.0 model, trained on LibriSpeech-960h combined with a standard KenLM 4-gram LM

[5]https://github.com/facebookresearch/fairseq/tree/main/examples/language_model

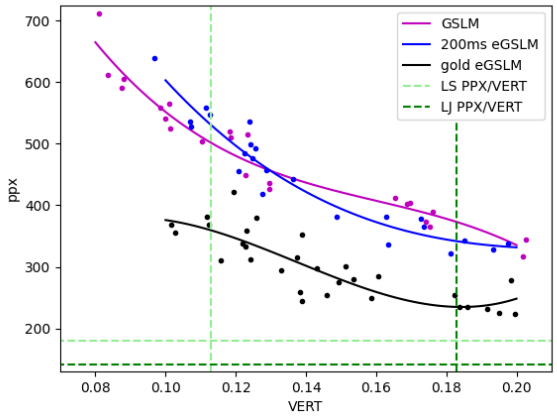

Figure 3: PPX and VERT scores for GSLM, 200ms-tGSLM and gold-tGSLM. Each dot is obtained by generating sentences with a fixed temperature parameter. The curves are 3rd-degree polynomial interpolation of the dots. The green dashed lines are the oracle PPX/VERT obtained on the LibriSpeech and LJ corpus.

	WUGGY↑	**SBLIMP**↑	ABX_{sem} ↑	ABX_{POS} ↑	**PPX@LS-VERT**↓	**PPX@LJ-VERT**↓
GSLM	**70.36**	**56.31**	55.85	59.03	**503.25**	387.45
200ms-tGSLM	68.53	55.31	**55.89**	**60.3**	532.87	**356.24**
gold-tGSLM	*86.37*	-[6]	*65.6*	*75.59*	*361.84*	*255.32*

Table 1: Results on zero-shots and generation tasks for 200ms-tGSLM and GSLM, trained on LL6k-clean, and gold-tGSLM, trained on LibriSpeech. ABX is computed on tGSLM lexical tokens and on GSLM 9th layer

that consists in finding clusters or pairs of speech segments, from unsegmented audio, that have the same phonetic transcription. For each discovered pair, the NED is computed as the edit distance normalized by the length of the longest item in the pair. As for ABX tasks, the NED is also based on distance between embeddings. To compute a NED score, we take inspiration of the procedure introduced in Thual et al. (2018). Given a segmentation of the LibriSpeech dev-clean subset, all speech segments are embedded into fixed-size vectors. With a k-NN, we search for the pairs of closest embeddings and sort them by cosine similarity. Starting from the higher similarities, we retrieve as much pair as necessary to cover the whole dev-clean set. With the phoneme-level transcription of the dev-clean set, all pairs can be transcribed into series of phonemes. The final NED score is obtained by averaging the NED over all pairs of transcriptions. NED and ABX tasks both rely on embeddings that can be extracted at any level of a multi-layer neural model.

5 Results

5.1 Performances on automatic metrics

Figure 3 provides a comparison of the original discrete unit-based GSLM with two version of our continuous unit model: 200ms-tGSLM, trained on speech segmented every 200ms and gold-tGSLM, trained on speech segmented on the true word boundaries. GSLM and 200ms-tGSLM are trained on LL6k-clean[7] while the topline, gold-tGSLM, is trained only on LibriSpeech corpus (word boundaries cannot be computed for LL6k-clean because sentence-level speech and text alignments are missing). The dots in Figure 3 represent batches of generated sentences conditioned on different temperatures. Color curves are the 3rd degree polynomial interpolation of the dots. In green dashed

[6]Nguyen et al. (2020) did not provide true word boundaries for sBLIMP

[7]Training 200ms-tGSLM on Libri-light 60k (Kahn et al., 2019), a larger but noisier corpus, slightly undermined the performance.

	WUGGY↑	SBLIMP↑	ABX_{sem} ↑	ABX_{POS} ↑	PPX@LS-VERT↓	PPX@LJ-VERT↓
GSLM	**65.85**	**54.35**	55.18	**61.61**	664.23	497.65
sylseg-tGSLM	64.39	53.21	54.64	60.01	634.34	505.87
dpparse-tGSLM	65.54	53.82	**55.6**	58.65	634.34	505.87
200ms-tGSLM	63.15	53.34	55.08	60.24	**610.32**	**490.32**

Table 2: Results on zero-shot and generation tasks for GSLM and for tGSLM on three different speech segmentation methods. Models are all trained on LibriSpeech. ABX is computed on tGSLM lexical tokens and on GSLM 9th layer

lines appear the anchor points computed on batches of text from LibriSpeech and LJ. Regarding performances, 200ms-tGSLM is on par with GSLM: slightly better at LJ-VERT and slightly worse at LS-VERT. The intersection of the dashed lines and the curves gives a score that is reported as PPX@LS-VERT and PPX@LJ-VERT in Table 1. Our models and GSLM can also be compared with transcripts of speech generations available in appendix at Tables 7,8 and 9. The topline gold-tGSLM produces much better generations than the two other models. We wanted the topline to be unaffected by errors in speech decoding. Therefore, when gold-tGSLM has generated a series of lexical tokens, it bypasses the speech decoder and retrieves the true transcriptions directly from the corpus word-level alignment.

Zero-shot tasks are also available in Table 1. GSLM and 200ms-tGSLM score similarly on all zero-shot metrics, with an advantage for GSLM on $sWUGGY$ and $sBLIMP$ and an advantage for 200ms-tGSLM on ABX_{sem} and ABX_{POS}. The topline gold-tGSLM, once again gets much stronger results. ABX scores are obtained, for GSLM at the 9th layer of the transformer and for tGSLM with the lexical tokens.

We think that small differences of performances on zero-shot tasks across models are not significant as these metrics are known to have some unexplained variances across self-supervised models Algayres et al. (2020). Similarly, small difference on PPX/VERT across models are due to the task intrinsic variance due to its reliance on sampling.

5.2 Speech segmentation

To study the impact of speech segmentation on tGSLM, we trained this model on LibriSpeech with two extra segmentation methods: SylSeg (Räsänen et al., 2018), and DP-Parse (Algayres et al., 2022b)[8]. Sylseg segments speech into syllable-like units, using damped oscillators that exploit rhythmic cues of syllabic structure in speech. DP-Parse (Algayres et al., 2022b) segments speech into word-like units with state-of-the-art performances. This model adapts a non-parametric Bayesian model for text segmentation (Goldwater et al., 2009) to speech. Table 2 shows generation and zero-shot scores. Overall, regarding speech generation, 200ms-tGSLM outperform sylseg-tGSLM, dpparse-tGSLM and also GSLM. For zero-shot tasks, once again, all models score similarly. ABX scores are again obtained for GSLM with embeddings extracted from the 9th layer of the transformer and for tGSLM from the lexical tokens.

Even though true word boundaries strongly benefit tGSLM, using unsupervised speech segmentation methods did not prove beneficial. We think this is due to the low performances of state-of-the-art speech segmentation systems. These latter are only marginally better than random segmentations and lag largely behind text segmentation performances Dunbar et al. (2022b); Algayres et al. (2022b). This result suggests that progress is needed in unsupervised speech segmentation to be able to combine segmented units into intelligible speech. After all, the best segmentation method that we works for us is the 200ms method. We have also experimented with other durations as 120ms,280ms and 360ms. We chose to go on with 200ms based on a compromise between maximal duration and maximal zero-shot tasks performances. These scores that can be found in appendix at Table 5.

5.3 Interpretability

So far, ABX and NED have been measured at the level of the lexical tokens. In order to analyze what is learned by $LexEmb$ we measure the ABX and NED with lexical tokens (like in the last

[8]We did not train those models on LL6k-clean because DP-Parse is hard to scale to large datasets

section) but also with acoustic tokens. In Table 3, the ABX scores show that the acoustic tokens are at chance level on semantic and syntactic encoding. After the $LexEmb$ function, the lexical tokens loose a bit of their phonetic encoding (NED increases) but gain the ability to represent semantics and syntax. However, the NED is not at chance level, meaning that a bit of acoustic information has leaked into the lexical tokens. To visualize the difference between acoustic and lexical spaces, we provide t-SNE maps in Appendix Section A.4.

models	tokens	**NED** $\downarrow$	ABX_{sem} $\uparrow$	ABX_{POS} $\uparrow$
200ms-tGSLM	acoustic	**34.51**	50.14	49.87
	lexical	47.98	**55.08**	**60.24**
gold-tGSLM	acoustic	**16.15**	50.20	50.12
	lexical	22.70	**65.60**	**75.59**

Table 3: NED and ABX scores on acoustic and lexical tokens for 200ms-tGSLM and gold-tGSLM both trained on LibriSpeech. ABX and NED are computed on tGSLM lexical tokens

5.4 Memory consumption

GSLM model Lakhotia et al. (2021) and 200ms-tGSLM use the same transformer LM but with different type of inputs. Compared to the 200ms-long units of our model, GSLM is trained on discrete units that are 40ms long in average (when contiguous duplicates are removed). Therefore, we expected from our model to be 5 times more memory efficient than GSLM[9] which can be observed by the maximal batch size that both models can handle. Indeed, on the one hand, we managed to train GSLM with 34 60-seconds-long sentences on a 32G V100 GPU whitout OOM error. On the other hand, 200ms-tGSLM can fit as much as 162 sentences, which shows almost a 5 time reduction (≈ 4.76) of memory use. Increasing the batch size is not necessary in our setting as best performances are obtained with a batch size of 32 sentences for 200ms-tGSLM. However, in order to train a spoken LM on corpora with much longer audio sequences, for instances Spotify or Youtube interviews, memory consumption will become a bottleneck and reducing the number of tokens in input could become crucial. To complete our analysis, we provide in Appendix A.5 a theoretical analysis of memory reduction.

6 Conclusion

We introduced a generative spoken language model based on continuous word-sized acoustic tokens. To guarantee reproducibility, a link to the complete source code will be made available upon acceptance. This model is able to generate speech with the same level of diversity and accuracy as a model based on discrete units. This shows that building a lexicon of types is not necessary for spoken language modeling, which is a encouraging considering the difficulty of clustering large segments of speech without degrading the representation. In addition, this performance was obtained with segments that were not very well aligned with word boundaries: 200ms segments or unsupervised segments obtained by DP-Parse. The good result obtained with gold word boundaries indicate that there is room for improvement by using segments better aligned with word boundaries. Further work is also needed to better limit the leakage of low level acoustic information into the LM through continuous units, which our analysis has shown is detrimental to the performance of the generative model (see also Nguyen et al. (2022c)). Finally, the fact that the units are about 5 times larger than standard GSLM units aligns with the NLP literature that is in favor of word-based LMs. It opens the possibility to fit larger spans of audio in GPUs and capture longer distance relationships.

[9] As a reminder, the acoustic tokens that are the input of 200ms-tGSLM are extracted as a preprocessing step. They do not impact memory usage at training time.

REFERENCES

Robin Algayres, Mohamed Salah Zaiem, Benoit Sagot, and Emmanuel Dupoux. Evaluating the reliability of acoustic speech embeddings, 2020. URL https://arxiv.org/abs/2007.13542.

Robin Algayres, Adel Nabli, Benoît Sagot, and Emmanuel Dupoux. Speech sequence embeddings using nearest neighbors contrastive learning, 2022a. URL https://arxiv.org/abs/2204.05148.

Robin Algayres, Tristan Ricoul, Julien Karadayi, Hugo Laurençon, Salah Zaiem, Abdelrahman Mohamed, Benoît Sagot, and Emmanuel Dupoux. Dp-parse: Finding word boundaries from raw speech with an instance lexicon, 2022b. URL https://arxiv.org/abs/2206.11332.

Alexei Baevski, Steffen Schneider, and Michael Auli. vq-wav2vec: Self-supervised learning of discrete speech representations. *CoRR*, abs/1910.05453, 2019. URL http://arxiv.org/abs/1910.05453.

Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *CoRR*, abs/2006.11477, 2020. URL https://arxiv.org/abs/2006.11477.

Piotr Bojanowski, Armand Joulin, and Tomás Mikolov. Alternative structures for character-level rnns. *CoRR*, abs/1511.06303, 2015. URL http://arxiv.org/abs/1511.06303.

Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Olivier Teboul, David Grangier, Marco Tagliasacchi, and Neil Zeghidour. Audiolm: a language modeling approach to audio generation. *arXiv preprint arXiv:2209.03143*, 2022.

Mingjie Chen and Thomas Hain. Unsupervised Acoustic Unit Representation Learning for Voice Conversion Using WaveNet Auto-Encoders. In *Proc. Interspeech 2020*, pp. 4866–4870, 2020. doi: 10.21437/Interspeech.2020-1785.

Ewan Dunbar, Robin Algayres, Julien Karadayi, Mathieu Bernard, Juan Benjumea, Xuan-Nga Cao, Lucie Miskic, Charlotte Dugrain, Lucas Ondel, Alan W. Black, Laurent Besacier, Sakriani Sakti, and Emmanuel Dupoux. The zero resource speech challenge 2019: TTS without T. *CoRR*, abs/1904.11469, 2019. URL http://arxiv.org/abs/1904.11469.

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*, 2022a.

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 2022b. doi: 10.1109/jstsp.2022.3206084. URL https://doi.org/10.1109%2Fjstsp.2022.3206084.

Siyuan Feng, Tan Lee, and Zhiyuan Peng. Combining Adversarial Training and Disentangled Speech Representation for Robust Zero-Resource Subword Modeling. In *Proc. Interspeech 2019*, pp. 1093–1097, 2019. doi: 10.21437/Interspeech.2019-1337.

Sharon Goldwater, Thomas Griffiths, and Mark Johnson. A bayesian framework for word segmentation: Exploring the effects of context. *Cognition*, 112:21–54, 04 2009. doi: 10.1016/j.cognition.2009.03.008.

Alex Graves. Generating sequences with recurrent neural networks. *CoRR*, abs/1308.0850, 2013. URL http://arxiv.org/abs/1308.0850.

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. *CoRR*, abs/2106.07447, 2021. URL https://arxiv.org/abs/2106.07447.

Keith Ito and Linda Johnson. The lj speech dataset. 2017. URL https://keithito.com/LJ-Speech-Dataset/.

Christiaan Jacobs, Yevgen Matusevych, and Herman Kamper. Acoustic word embeddings for zero-resource languages using self-supervised contrastive learning and multilingual adaptation. In *2021 IEEE Spoken Language Technology Workshop (SLT)*, pp. 919–926, 2021. doi: 10.1109/SLT48900.2021.9383594.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *CoRR*, abs/1702.08734, 2017. URL http://arxiv.org/abs/1702.08734.

Jacob Kahn, Morgane Rivière, Weiyi Zheng, Evgeny Kharitonov, Qiantong Xu, Pierre-Emmanuel Mazaré, Julien Karadayi, Vitaliy Liptchinsky, Ronan Collobert, Christian Fuegen, Tatiana Likhomanenko, Gabriel Synnaeve, Armand Joulin, Abdelrahman Mohamed, and Emmanuel Dupoux. Libri-light: A benchmark for ASR with limited or no supervision. *CoRR*, abs/1912.07875, 2019. URL http://arxiv.org/abs/1912.07875.

Herman Kamper. Truly unsupervised acoustic word embeddings using weak top-down constraints in encoder-decoder models. *CoRR*, abs/1811.00403, 2018. URL http://arxiv.org/abs/1811.00403.

Herman Kamper. Word segmentation on discovered phone units with dynamic programming and self-supervised scoring, 2022. URL https://arxiv.org/abs/2202.11929.

Herman Kamper, Aren Jansen, Simon King, and Sharon Goldwater. Unsupervised lexical clustering of speech segments using fixed-dimensional acoustic embeddings. In *2014 IEEE Spoken Language Technology Workshop (SLT)*, pp. 100–105, 2014. doi: 10.1109/SLT.2014.7078557.

Eugene Kharitonov, Ann Lee, Adam Polyak, Yossi Adi, Jade Copet, Kushal Lakhotia, Tu-Anh Nguyen, Morgane Rivière, Abdelrahman Mohamed, Emmanuel Dupoux, et al. Text-free prosody-aware generative spoken language modeling. *arXiv preprint arXiv:2109.03264*, 2021.

Eugene Kharitonov, Jade Copet, Kushal Lakhotia, Tu Anh Nguyen, Paden Tomasello, Ann Lee, Ali Elkahky, Wei-Ning Hsu, Abdelrahman Mohamed, Emmanuel Dupoux, and Yossi Adi. textless-lib: a library for textless spoken language processing, 2022. URL https://arxiv.org/abs/2202.07359.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. *CoRR*, abs/2010.05646, 2020. URL https://arxiv.org/abs/2010.05646.

Felix Kreuk, Joseph Keshet, and Yossi Adi. Self-supervised contrastive learning for unsupervised phoneme segmentation. *arXiv preprint arXiv:2007.13465*, 2020.

Felix Kreuk, Adam Polyak, Jade Copet, Eugene Kharitonov, Tu-Anh Nguyen, Morgane Rivière, Wei-Ning Hsu, Abdelrahman Mohamed, Emmanuel Dupoux, and Yossi Adi. Textless speech emotion conversion using decomposed and discrete representations. *arXiv preprint arXiv:2111.07402*, 2021.

Kundan Kumar, Rithesh Kumar, Thibault de Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre de Brebisson, Yoshua Bengio, and Aaron Courville. Melgan: Generative adversarial networks for conditional waveform synthesis, 2019. URL https://arxiv.org/abs/1910.06711.

Kushal Lakhotia, Evgeny Kharitonov, Wei-Ning Hsu, Yossi Adi, Adam Polyak, Benjamin Bolte, Tu Anh Nguyen, Jade Copet, Alexei Baevski, Adelrahman Mohamed, and Emmanuel Dupoux. Generative spoken language modeling from raw audio. *CoRR*, abs/2102.01192, 2021. URL https://arxiv.org/abs/2102.01192.

Andy T. Liu, Po chun Hsu, and Hung-Yi Lee. Unsupervised End-to-End Learning of Discrete Linguistic Units for Voice Conversion. In *Proc. Interspeech 2019*, pp. 1108–1112, 2019. doi: 10.21437/Interspeech.2019-2048.

Tomas Mikolov, Ilya Sutskever, Anoop Deoras, Hai Son Le, Stefan Kombrink, and Jan Honzaernocky. Subword language modeling with neural networks. 2011.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013. URL https://arxiv.org/abs/1301.3781.

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. *CoRR*, abs/2011.11588, 2020. URL https://arxiv.org/abs/2011.11588.

Tu Anh Nguyen, Maureen de Seyssel, Robin Algayres, Patricia Roze, Ewan Dunbar, and Emmanuel Dupoux. Are word boundaries useful for unsupervised language learning?, 2022a. URL https://arxiv.org/abs/2210.02956.

Tu Anh Nguyen, Eugene Kharitonov, Jade Copet, Yossi Adi, Wei-Ning Hsu, Ali Elkahky, Paden Tomasello, Robin Algayres, Benoit Sagot, Abdelrahman Mohamed, et al. Generative spoken dialogue language modeling. *arXiv preprint arXiv:2203.16502*, 2022b.

Tu Anh Nguyen, Benoit Sagot, and Emmanuel Dupoux. Are discrete units necessary for spoken language modeling? *IEEE Journal of Selected Topics in Signal Processing*, pp. 1–9, 2022c. doi: 10.1109/JSTSP.2022.3200909.

Zizheng Pan, Bohan Zhuang, Jing Liu, Haoyu He, and Jianfei Cai. Scalable vision transformers with hierarchical pooling, 2021. URL https://arxiv.org/abs/2103.10619.

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.

Wei Ping, Kainan Peng, Andrew Gibiansky, Sercan Arik, Ajay Kannan, Sharan Narang, Jonathan Raiman, and John Miller. Deep voice 3: 2000-speaker neural text-to-speech. 10 2017.

Ryan Prenger, Rafael Valle, and Bryan Catanzaro. Waveglow: A flow-based generative network for speech synthesis. *CoRR*, abs/1811.00002, 2018. URL http://arxiv.org/abs/1811.00002.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. Technical report, Tech. Rep., OpenAI, 2022.

Okko Johannes Räsänen, Gabriel Doyle, and Michael C. Frank. Pre-linguistic segmentation of speech into syllable-like units. *Cognition*, 171:130–150, 2018.

Morgane Rivière and Emmanuel Dupoux. Towards unsupervised learning of speech features in the wild. In *2021 IEEE Spoken Language Technology Workshop (SLT)*, pp. 156–163, 2021. doi: 10.1109/SLT48900.2021.9383461.

Shane Settle and Karen Livescu. Discriminative acoustic word embeddings: Recurrent neural network-based approaches. *CoRR*, abs/1611.02550, 2016. URL http://arxiv.org/abs/1611.02550.

Jonathan Shen, Ruoming Pang, Ron J. Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, Rj Skerrv-Ryan, Rif A. Saurous, Yannis Agiomvrgiannakis, and Yonghui Wu. Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4779–4783, 2018. doi: 10.1109/ICASSP.2018.8461368.

Alexis Thual, Corentin Dancette, Julien Karadayi, Juan Benjumea, and Emmanuel Dupoux. A K-nearest neighbours approach to unsupervised spoken term discovery. In *IEEE Spoken Language Technology SLT-2018*, Proceedings of SLT 2018, Athènes, Greece, December 2018. URL https://hal.archives-ouvertes.fr/hal-01947953.

Andros Tjandra, Berrak Sisman, Mingyang Zhang, Sakriani Sakti, Haizhou Li, and Satoshi Nakamura. VQVAE Unsupervised Unit Discovery and Multi-Scale Code2Spec Inverter for Zerospeech Challenge 2019. In *Proc. Interspeech 2019*, pp. 1118–1122, 2019. doi: 10.21437/Interspeech.2019-3232.

Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew W. Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *CoRR*, abs/1609.03499, 2016. URL http://arxiv.org/abs/1609.03499.

Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *CoRR*, abs/1807.03748, 2018. URL http://arxiv.org/abs/1807.03748.

Lisa Van Staden and Herman Kamper. A comparison of self-supervised speech representations as input features for unsupervised acoustic word embeddings, 2020. URL https://arxiv.org/abs/2012.07387.

A. Vasuki and P.T. Vanathi. A review of vector quantization techniques. *IEEE Potentials*, 25(4): 39–47, 2006. doi: 10.1109/MP.2006.1664069.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017. URL http://arxiv.org/abs/1706.03762.

Maarten Versteegh, Xavier Anguera, Aren Jansen, and Emmanuel Dupoux. The zero resource speech challenge 2015: Proposed approaches and results. *Procedia Computer Science*, 81:67–72, 2016. ISSN 1877-0509. doi: https://doi.org/10.1016/j.procs.2016.04.031. URL https://www.sciencedirect.com/science/article/pii/S187705091630045X. SLTU-2016 5th Workshop on Spoken Language Technologies for Under-resourced languages 09-12 May 2016 Yogyakarta, Indonesia.

Shuwen Yang, Po-Han Chi, Yung-Sung Chuang, Cheng-I Jeff Lai, Kushal Lakhotia, Yist Y. Lin, Andy T. Liu, Jiatong Shi, Xuankai Chang, Guan-Ting Lin, Tzu-Hsien Huang, Wei-Cheng Tseng, Ko tik Lee, Da-Rong Liu, Zili Huang, Shuyan Dong, Shang-Wen Li, Shinji Watanabe, Abdelrahman Mohamed, and Hung yi Lee. SUPERB: Speech Processing Universal PERformance Benchmark. In *Proc. Interspeech 2021*, pp. 1194–1198, 2021. doi: 10.21437/Interspeech.2021-1775.

Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. Soundstream: An end-to-end neural audio codec. *CoRR*, abs/2107.03312, 2021. URL https://arxiv.org/abs/2107.03312.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. Texygen: A benchmarking platform for text generation models. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, pp. 1097–1100, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450356572. doi: 10.1145/3209978.3210080. URL https://doi.org/10.1145/3209978.3210080.

George Kingsley Zipf. Human behavior and the principle of least effort. *Addison-Wesley Press*, 1949.

A Appendix

A.1 Discussion on q

A.1.1 Mathematical details on q

Let us now derive q computation. Given a training corpus, that is segmented and encoded into a collection of acoustic tokens $(a_i)_{i\leq N}$. A PCA is trained on $(a_i)_{i\leq N}$ and the d first dimensions

are kept, let us write $(a'_i)_{i \leq N}$ the resulting vectors and $(v_0, ..., v_{d'})$ the explained variance of each PCA dimensions. Then, we train d separate k-means on each dimension of the PCA. The number of cluster per k-means is computed as $(\left\lceil K \frac{v_0}{v_0} \right\rceil, , ..., \left\lceil K \frac{v_{d'}}{v_0} \right\rceil)$. The values of d and K were set to maximize the scores at the zero-shot tasks. Once the k-means are trained, the centroids are stored in d dictionnaries $(k_0, ..., k_d)$. For any $i \leq N$, we compute $q(a_i)$ by assigning $\forall j \leq d$, $q(a_i)[j]$ to its closest centroids in k_j. Finally, cluster ids are turned into onehot vectors and concatenated into a single vector. The following operations sum up the process.

$$\forall i \leq n, q(a_i) \leftarrow \begin{pmatrix} \underset{j \leq K}{\text{argmax}}(a_i[0] - k_0[j]) \\ \underset{j \leq \left\lceil K \frac{v_1}{v_0} \right\rceil}{\text{argmax}} (a_i[1] - k_1[j]) \\ \vdots \\ \underset{j \leq \left\lceil K \frac{v_d}{v_0} \right\rceil}{\text{argmax}} (a_i[d] - k_d[j]) \end{pmatrix}$$

$$q(a_i) \leftarrow \begin{pmatrix} onehot(q(a_i[0])) \\ onehot(q(a_i[1])) \\ \vdots \\ onehot(q(a_i[d])) \end{pmatrix}$$

$$q(a_i) \leftarrow concatenate(q(a_i[0]), ..., q(a_i[d]))$$

A.1.2 Ablation and discussion on q

The function q introduced in Section 3.1.2, composed of a PCA and our d-k-means method, is ablated in Table 4. In all configurations, the embeddings right after the $LexEmb$ function are used to compute the ABX and NED scores. On the one hand, q degrades the phonetic information in the lexical tokens (NED increases) and makes training harder (validation loss increases). On the other hand, q maximize semantic and syntactic information (ABX increases) as well as generation quality (PPX decreases). A $null$ value in Table 4 means that the model is not able to produce intelligible sentences with this setup. First, these experiments show the necessity of q for the 200ms-tGSLM to generate spoken sentences. Second, the combination of these results reveal that q prevents the model from converging quickly to a bad local minimum that hinders generalization.

It follows our intuition from Section 3.1.2: there seems to be a low-variance signal encoded in the acoustic tokens that interfere with the semantic and syntactic modelling. In our opinion, this signal gives away both local information, direct right and left context due to coarticulation, and global sentence-level information (relative token position and speaker identity).

	PCA	d-k-means	**Valid loss**↓	**NED**↓	ABX_{sem} ↑	ABX_{POS} ↑	**PPX@LS-VERT**↓	**PPX@LJ-VERT**↓
200ms-tGSLM			**2.51**	**35.21**	53.87	58.40	null	null
200ms-tGSLM	✓		4.33	41.50	54.16	57.99	840.65	null
200ms-tGSLM	✓	✓	6.21	44.32	**55.08**	**60.24**	**610.32**	**490.32**
gold-tGSLM			**3.99**	**17.21**	55.13	63.54	608.24	475.65
gold-tGSLM	✓		6.20	21.87	58.59	67.71	432.78	384.57
gold-tGSLM	✓	✓	7.15	22.70	**65.60**	**75.59**	**361.84**	**255.32**

Table 4: Results on zero-shot and generation tasks for ablations of the PCA and d-k-means components of the $LexEmb$ function. Models are trained on LibriSpeech. ABX and NED are computed on tGSLM lexical tokens. $null$ means that no intelligible speech can be generated in this setting.

One may say that if q is used to mitigate the downsides of the attention mechanism of Wav2vec2.0, why not using more local features like MFCC or Mel-filterbanks? We argue, that even though these latter features are still good for supervised tasks as ASR Radford et al. (2022), they are substantially outperformed by recent self-supervised speech models (Wav2vec2.0, CPC, HuBERT,...) at the tasks of zero-shot word discrimination Algayres et al. (2022a); Van Staden & Kamper (2020) and keyword spotting Yang et al. (2021). Therefore, we think Mel-filterbanks and MFCCs are ill-suited to be the input of acoustic tokens.

A.2 HYPERPARAMETERS

Wav2vec2.0 and SSE are trained on the LibriSpeech corpus respectively by Baevski et al. (2020) and Algayres et al. (2022a). Wav2vec2.0 Base is a stack of 7 convolution layers and 12 transformer layers. The SSE is composed of a one GLU convolution layer (kernel size: 4, number of channels: 512, stride: 1), a transformer layer (attention heads: 4, size of attention matrices: 512 neurons, and FFN: 2048 neurons) and a final max-pooling layer along the time axis.

LexEmb is composed of two functions $L \circ q$. L is a stack of five three-layers blocks each formed by a 1024-neurons FC layer, a layer norm and a ReLU activation. q is of a PCA and a collection of k-means that are trained on LL6k-clean. The PCA has $d = 24$ dimensions and the number of centroids for the first k-means is $K = 10$.

Transformer is identical to the one used in the original GSLM paper (Lakhotia et al., 2021). It contains 12 transformer layers with 16 heads, 1024-neurons attention matrices, 4096-neurons FFN. On top of the transformer, the three parallel $h1$,$h2$,$h3$ functions are 1024-neurons FCs. L,$h1$,$h2$,$h3$ and the transformer are trained on 32 GPUs, for 200k iterations on either the LibriSpeech or LL6k-clean. Each batch is composed of 64 audio sentences that are composed of 64 tokens. The learning rate is set at 5^{-4} with warm-up of 5000 updates and polynomial decay. We use Adam optimizer with a weight decay of 0.1. A dropout of 0.1 is applied during training. The loss function is the NCE loss with a temperature of 0.1 and 500 negative samples.

Sampling is performed in a FAISS k-NN (Johnson et al., 2017) that contains all the lexical tokens segmented in the dev-clean and test-clean from the LibriSpeech (roughly 10 hours of speech). The number of nearest neighbors from which the next token is sampled is set to 1000.

Speech generation model is an encoder and a decoder that shares the same architecture: 4 transformer layers with 8 heads, 512-neurons attention matrices, 3072-neurons FFN. It is trained on 32 GPUs, for 30k iterations on the LibriSpeech. Each batch is composed of four audio sentences that are at maximum 20 seconds long. The learning rate is set at 5^{-5} with warm-up of 10^3 updates and polynomial decay. We use a dropout probability of 0.1 and Adam optimizer with a weight decay of 0.1. The Tacotron2.0 from Lakhotia et al. (2021); Kharitonov et al. (2022) was trained on LJ.

A.3 CLUSTERING LARGE UNITS

The core element of tGSLM is the transformer that is trained as a LM to predict the future. To train the LM, our first idea was to cluster acoustic tokens into discrete tokens so that the LM could be trained with the classical NLL loss. Nevertheless, clustering large speech tokens is a hard task. First because the latent number of classes rises exponentially with the average duration of the tokens. Second because of word tokens are distributed along a highly skewed distribution known as the Zipf Law (Zipf, 1949). Kamper et al. (2014) have tackled the problem and have shown that a regular k-means is one of the best clustering method for that problem. Yet, all our attempt to use k-means on the acoustic tokens failed: the validation loss was barely decreasing, and the resulting models were unable to generate any understandable sentence. We decided not to use clustering and adapt our pipeline to continuous input tokens.

A.4 PROBING ACOUSTIC AND LEXICAL SPACES

Figure 4 is a visualization of the acoustic and lexical representation learned by gold-tGSLM. All speech segments corresponding to real words in the LibriSpeech dev-clean set are indexed in k-NN graphs on their acoustic or lexical form. Each embedding is labelled with its true transcription. By searching for the nearest neighbors of a center word (in red in the figure), we highlight in green the neighbors that we judged semantically related to the center word. Figure 4 shows that an acoustic token has usually no semantically related neighbor other than ones with the same transcription. By contrast, lexical tokens have semantic and syntactic properties: 'London' is close to other cities and countries, 'blue' is close to color names, beautiful is close to other positive adjectives, and 'chair' is close to 'desk' and 'table'. Nonetheless, it appears acoustic information has leaked from the acoustic tokens into the lexical tokens. For instance, the lexical neighbors of 'blue' are colors or shades that start with a 'b' and 'chest' appears in the neighborhood of 'chair'.

A.5 ESTIMATION OF MEMORY CONSUMPTION

To estimate the memory consumption of a transformer LM with L layers, let us write $x \in \mathbb{R}^{n \times d}$ a sentence of n tokens represented by embeddings of size d. Using the formula expressed in the supplementary material of Pan et al. (2021) (which is straightforward to derive), the number of activations to store in memory during backpropagation is approximately $\phi(n, d) = (12nd^2 + 2n^2d)L$. In our case, for both GSLM and 200ms-tGSLM, $d = 1024$. In the LL6k-clean corpus sentences are 60s-long in average with make $n = 1500$ for GSLM and $n = 300$ for 200ms-tGSLM. 200ms-tGSLM should expect a memory reduction by a factor of $\frac{\phi(300,1024)}{\phi(1500,1024)} \approx 5.93$ compared to GSLM. In practice, we observe a lower memory reduction (≈ 4.76) which can be explained by the additional parameters that are present in 200ms-tGSLM and not in GSLM, namely the $LexEmb$ function and three prediction heads).

models	$sWUGGY \uparrow$	$sBLIMP \uparrow$	$ABX_{sem} \uparrow$	$ABX_{POS} \uparrow$	*average* $\uparrow$
120ms-tGSLM	63.55	53.86	55.74	60.12	58.32
200ms-tGSLM	63.15	53.34	55.08	60.24	57.95
280ms-tGSLM	61.89	51.64	52.8	56.28	55.65
360ms-tGSLM	60.18	51.29	52.18	55.45	54.75

Table 5: Zero-shot tasks computed on tGSLM trained on LibriSpeech for different unit durations

models	$sWUGGY \uparrow$	$sBLIMP \uparrow$	$ABX_{sem} \uparrow$	$ABX_{POS} \uparrow$	*average* $\uparrow$
next word	61.57	52.08	51.48	53.84	54.75
next two words	63.02	**53.48**	54.79	58.01	57.35
next three words	**63.15**	53.34	**55.08**	**60.24**	**57.95**
next four words	62.25	53.1	54.43	58.81	57.14

Table 6: Zero-shot tasks for 200ms-tGSLM trained on LibriSpeech to predict the next one, two, three, or four words

200ms-tGSLM examples
Generation at LJ-VERT
What is it ask her mother i want to see you said mrs tumbled i want to tell you what you you said mr cockry you are no more chance than you know.
We have no desire to prevent to the astonishment of that person from the government who is not so far for receiving any property or relation to the world.
Her father in her son were under growth her father was just like a treasure man who was a devil and hazards beyond his words she was a very clearly.
We also see that it will be obliged to invite us to applyge them to observe such a thing is a base we must not set down that the
And although he was not equally successful to him he sought the pririate regularly observed his friends invent to him and presented him their own secret he had did
Because they were rested and although they could not expect to be obliged to regarded as a men of a gold and power they were not really unbusy
You see he is if i miss thing i think he is dead it said mrs carpenter rather smallly for anything if he is a total let she said
He remembered that great city which he tried to entertain in its pointedof view but he was very pleasant to him and could not bring whom away besides this
Having required a measure for a month before their distance of sixty years he appeared to be affected by any conditions of one state and have in no battle
Now the king's brothers came to him and brought him up and said i'll poor woman i woke it of you anything but i am brother and borrows my
Generation at LS-VERT
He turned his hands on the sale exposition and gave him to acy of old meal which wealth had never bore be a foreseenly large.
While waiting to him he wished that he would wait for himself into his mother's house and held light he was that that she might be able to look.
And perhaps i have nothing to say about what would you want to know i did i don't know i suppose you want to know what could call the
That's all i can't do insaid woman looking out of the croad toward him while i don't know such any end enging his hand you seem to see your
And having been described as the great activity of that which he was attempting all that if he now remar it for his purpose was intended with principles as
It was the time had been prepared for for that such was the place that when he was saing to his land she'd made up all though blood that he
It was just as willing to oppose the person who had been told of his chion he was now about to go to bank in the family to a
Having been in a moment' officially desirable to acquaint him with his reference with the glorious presence of his master's cabinet he did not return to a subject of
Yes i was said he but was a general service she began i could not file forhard seek i want to take my as andt understand the chance of
But afterwards they had gone to top what waking into the stone doors the weathering tight their habitation and the north were histor carryance and mr carb's face and

Table 7: Example generations of 200ms-tGSLM trained on LL6k-clean. These generations are selected from batches of sentences that have a VERT equal 0.113 (LS-VERT) or 0.189 (LJ-VERT).

gold-tGSLM examples
Generation at LJ-VERT
But you have been wanting to teach me all her life in the world of her own healthy health and she has her fathers abilities an your pride.
The old woman or that he would have learned all her life amongst the gods and teaching them in their father's studies and having been up the days.
It goes on until i know what i am doing while i am going away from my camp in the neighbourhood to morrow we you from the whole on my.
An elegant geographical character would you think it a deed or an excellent thing to do with the hold in the future won't you pick up a bit of an
The evening of the twenty fifth of november eighteen united eight he returned to his royal house an the hold of the hospital an the next evening he you
Guiding them in some ordinary way or buying them into cold or buying them with a copal spoon which should be thrown out of the souls of the bulkhead
Secies and germany each of them had undergone more than three thousand roubles and hour to saved a bit of jewelry from s odin share and the hardness and
Of the kavin and when to the door where she stood a few minutes later to reach the bottom of the harboured near the labyrinth where she reached her
It says the king listening the light of his bushy fingers an holding his pipe in his arms do not bother me any more about it you know more than
The investigation and on his returned to her fathers room he set down his gun at a hundred yards and the middle of the hall to learn the hut
Generation at LS-VERT
I can tell her that only one of my friends and loves do you think i would read her about this uttered it all the wicked said missus williams
She reached her big house and stood by the dora in turning to the king he said to her you will not marion me any more have you hear
To his voices and his broken heart screen with delight to henry smokeless who had entered into his dining room to limp him a mystic playful of his faintest
I shall not go without thee said heat pausing to her part a of good direction and fixing her ices upon her eyes with a distant cheeks to her.
Than time of missus esplanade visit her own house and china herself alone of theirlocal service and the frere settlements where built for thousand of the happiest teachers
Father and mother were all seated at boston waiting for the empty school at ostrog at nine o'clock a the knight of july evening a ninth jeanne annie eighteen
Minutes later he heard a bill calling against the young man who had denounce him his face became a melancholy shake in his astonishment what is it said george
A poor boy in has a good power for somebody's harm to be at heaven what could you to givewhat this seemed to him a hard proposition
Paper it was needless to be summoned to it by the princess and the girl became very much surprise and said about recovering the bicycle with her finger to
To touch it he s a gentle young ma'am and does not see any other foreign of mortality unconnected with her father who is afraid of his flesh to

Table 8: Example generations of gold-tGSLM, trained on LibriSpeech. These generations are selected from two batches of sentences that have a VERT equal 0.113 (LS-VERT) or 0.189 (LJ-VERT).

GSLM examples
Generation at LJ-VERT
They did time in the desert two or three hundred years afterward among them the castle was not my father and they were found in palestine by fire and they
Another excellent is descended a breath let the corp of a prisoner and a blow was begun the bell rang the gunsprang from the captain's paul and dropped into the
And then the passing future would have been too much but to waittill the end of the week and after a little time she had gone down to the palace
But he passed along entirely untouched and was still together so frightened in the morning he went to look out for some place with a barian laine and then he
The brast of the bravest of the entire youth and of many of the slaves of the counillllors or of every fine breed and of the princess of france has
He had not in the least delicate way of helping her but had helld her into a pretty soft and a passionate graceful manner he told her every day because
But that man did not fight in the second place no ne did pay the attention to poverty it is not tpossible to suspect that the man of the previous
But all the people had come to see me and had not seen me again and they felt as if i were again coming to see me and so little
And people stared at him for a moment as if they were dead but he had not told them of his destiny that he would do so and they had
And a cow calling up his pipe said that no sign of the procession was ever heard and that no punishment was made or judgment was made nor any other
Generation at LS-VERT
His proposals that being so poing doubful i should very much regard and alia in boa's addition to cloak the great morning had given me a plague off waivering she
Someone to found a brown line and dance spent a moment over the vessel all saw the fair young chinese yard and dry he would waved dances in bubbles from
All cathics that are not due to f co notion or naturalist that is intentionble but if there is a personality of faith in them who was intentupon for seeing
The reef made the partets at the corner of a platform with them rose and ground on the floor of the lobby and of chapter fourteen two thousand se of
As if sudden impulse were convinced of their usual impulses and a strong exercise upon them or rather in their progress to bring their education to the reduction of manly
And rushing off from the cold winds in the west in the silence of the rock the cherry wavering soft quietness of people makes breath so cheap in a course
He had been burden with visitor and had petched his old preserance for death and mary the intamminable enterprising scenes caused by constantiis this trumpetts und drrawn courage and he
Great worked done an artificial lines of bounding is below the had an arm as it were it lookedfted itself and everything was so exquisite that the site was hard
To jew knew that they had been driven a doctor adreadful mattering to you the young girl whom eyed by a relatives ever since daily matters while a week before
Evenings in a ball volume whose close ways were rotted in whther cuts that fiddler devilalonsome his wife soldiers were harassing a women with deafferenren american last grading under fair

Table 9: Example generations of GSLM, trained on LL6k. These generations are selected from two batches of sentences that have a VERT equal 0.113 (LS-VERT) or 0.189 (LJ-VERT).

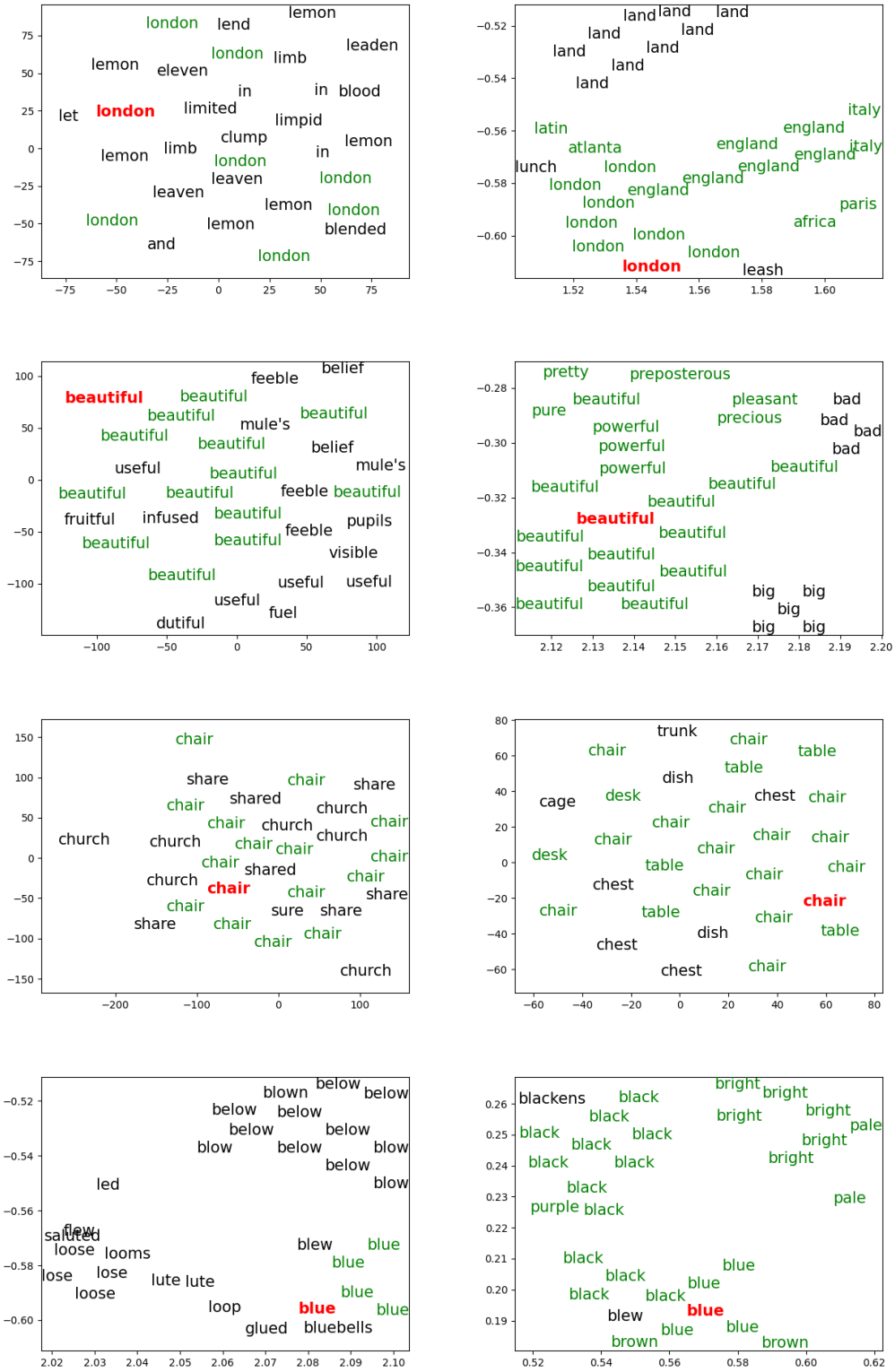

Figure 4: t-SNE representations of acoustic (left side) and lexical (right side) tokens. After training gold-tGSLM, all speech segments corresponding to word tokens in the LibriSpeech dev-clean subset are indexed into their acoustic or lexical form. By probing an acoustic or lexical token (appearing in red), we can have a look at their acoustic and lexical nearest neighbors. The neighbors that appear in green are those deemed as semantically related to the probe.