ANALYSING DISCRETE SELF SUPERVISED SPEECH REPRESENTATION FOR SPOKEN LANGUAGE MODELING

Amitay Sicherman and Yossi Adi

School of Engineering and Computer Science
The Hebrew University of Jerusalem, Israel

ABSTRACT

This work profoundly analyzes discrete self-supervised speech representations (units) through the eyes of Generative Spoken Language Modeling (GSLM). Following the findings of such an analysis, we propose practical improvements to the discrete unit for the GSLM. First, we start comprehending these units by analyzing them in three axes: interpretation, visualization, and resynthesis. Our analysis finds a high correlation between the speech units to phonemes and phoneme families, while their correlation with speaker or gender is weaker. Additionally, we found redundancies in the extracted units and claim that one reason may be the units' context. Following this analysis, we propose a new, unsupervised metric to measure unit redundancies. Finally, we use this metric to develop new methods that improve the robustness of units' clustering and show significant improvement considering zero-resource speech metrics such as ABX. Code and analysis tools are available under the following link.

Index Terms— self supervised learning, generative spoken language modeling, textless NLP, speech LM

1. INTRODUCTION

Recently Self-Supervised Learning (SSL) methods for speech have shown great success on plenty of downs stream tasks [1]. From Automatic Speech Recognition [2, 3, 4] and speaker diarization [5], to phone segmentation [6], these models have shown remarkable results.

Specifically, these SSL models allow recent success in Generative Spoken Language Modeling (GSLM) [7, 8, 9]. In GSLM, we aim to learn a discrete representation of the speech signal. This is often done by applying the k-means algorithm over the continuous representation obtained from the SSL model. Then, we train a unit Language Model (uLM) over such representation, and lastly, we decode it back to a time domain signal using vocoder [10]. During inference time, we can sample from the uLM.

Although these models can generate meaningful and coherent speech utterances, little is known about the properties captured but these discrete representations. The authors in [11] examined the purity between phonetics elements and the discrete representations. The authors' proposed method analyzes the discrete SSL representation considering fine-grained linguistic properties, e.g., different articulatory classes or closure and release portions. The authors in [12] proposed a probing method to analyze the presence of phone classes, gender, and language information while comparing monolingual and bilingual models.

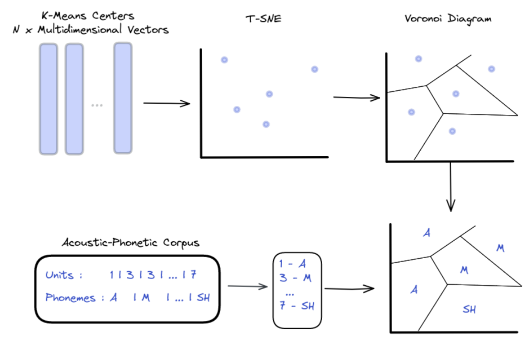

Fig. 1. Units visualization process.

This work analyzes quantitatively and visually discrete representations obtained by HuBERT and CPC models. Next, equipped with such an analysis, we provide a metric to identify redundancies in the discrete representations and propose a method to improve k-means clustering based on it.

We find a high correlation between the units and the phonemes, but with many redundancies in the units. We show that one reason may be the units' context. In Addition, we propose an unsupervised metric to measure these redundancies, and we use it to significant improvement the unit clustering.

2. BACKGROUND

The general GSLM pipeline is comprised of three main modules: (i) Speech-to-unit, (ii) Unit language model, and (iii) Unit-to-speech, where each of these modules is trained separately. Speech resynthesis can be achieved while ignoring the language model and directly feeding the quantized units into the unit-to-speech module [10]

Speech To Unit (STU) The model first encodes the raw

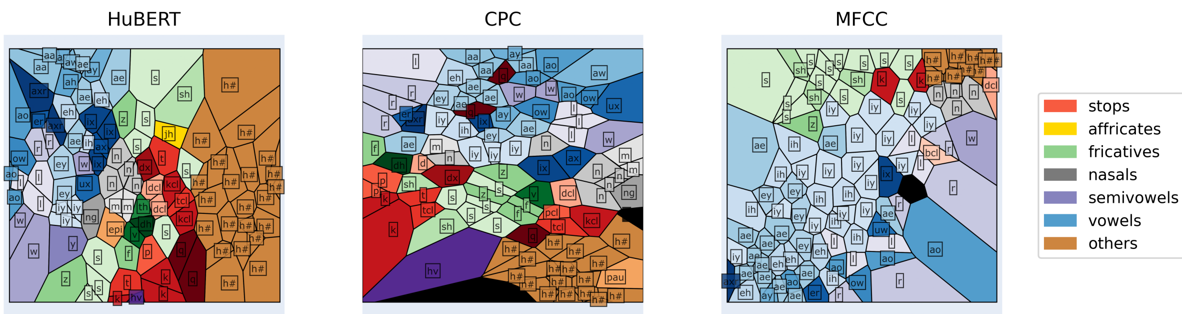

Fig. 2. Units visualization. Each bounded area represents a single unit and is colored by the unit's phoneme/phoneme family.

speech signal into a continuous representation and then quantizes the representation to a sequence of discrete units [7, 13, 14].

Formally, denote the domain of audio samples by $x \subset R$. Therefore, the representation for a raw signal is a sequence of samples $x = (x_1, \ldots, x_T)$, where $x_t \in x$ for all $1 \leq t \leq T$. Consider an encoder network, f, that gets as input the speech utterance and outputs a sequence of spectral representations sampled at a low frequency as follows $f(x) = (v_1, \ldots, v_{T'})$. Note that we do not assume anything about the structure of the encoder network f. Since the representations learned by such models are usually continuous, a k-means algorithm is applied over the models' outputs to generate discrete units, denoted as $z = (z_1, \ldots, z_{T'})$. Each element z_i in z is a positive integer, $z_i \in \{1, .., K\}$ for $1 \leq i \leq T'$, where K is the number of discrete units.

As the quantized representation, z, usually contain units repetitions that degrade the performance of the language modeling, a common approach is to collapse repetitions and generate a de-duplicated sequence while additionally storing the units' duration separately. For instance, the sequence `12,12,25,31,31,31` will be converted into `12,25,31` and the corresponding durations `2,1,3`.

Unit Language Model (ULM) is trained on the extracted and deduplicated discrete units, z. The language model can be used, for example, to generate speech conditionally or unconditionally.

Unit To Speech module converts the discrete speech representation, z, to a raw waveform. The authors in [7] used a Tacotron2.0 [15] based model followed by WaveGlow [16] vocoder. Later, [10] proposed a unit-based vocoder based on the HiFi-GAN architecture to convert units to speech directly. In this work, we focus on the latter setting.

3. METHOD

We analyze representations obtained by either HuBERT [2] or CPC [4] models considering a various number of clusters.

3.1. Analysis

Units Interpretation. We start by measuring the mutual information between the discrete representation and different speech properties (phonemes, speaker id, and gender) using the V-Measure score [17]. The v-Measure score is on a scale from 0 to 100, where 100 represents a perfect match. The v-measure can be considered a normalized mutual information score.

For this purpose, we align each utterance with its corresponding attribute. To get units-to-phonemes alignment, we iterate over the TIMIT corpus [18] (which has phoneme-to-signal alignment) and count all the units-to-phonemes pairs. Next, we assign a representative phoneme to each unit based on the majority in the unit - i.e., the phoneme label is the phoneme that appears with the unit the most. For speaker and gender analysis, we use the LibriSpeech corpus as it contains a large and diverse set of speakers.

Units Visualization. An additional point of view of the units' meaning is the spatial structure. For this purpose, we create a 2D spatial view containing information regarding the relationship between continuous representation, discrete units, and corresponding phonemes. Specifically, we apply the following two steps: (i) We project the high-dimensional speech representation into 2D using the T-SNE [19] with the perplexity of 30.0 and 1000 iterations. T-SNE is a nonlinear dimensionality reduction that intuitively preserves the nonlinear distance relations between neighbors in the high and low dimensions. Then, we use the Voronoi diagram [20] that converts the scatter plot into an area plot. Finally, we have left with a bounded area in the 2D space for each unit; (ii) In the second part, we create a single label to represent each cluster. First, we use the units-phonemes alignment from the TIMIT (similar to the process in the previous paragraph). Then, we assign for each cluster the most represented phoneme in it. Finally, we replace the unit id with their corresponding phonemes and color the area base on the phoneme and phoneme family. A visual description of the proposed method can be seen in Figure 1.

Units Resynthesis. Next, we analyze the units' information from the opposite direction - that is, through speech resyn-

Table 1. Units Interpretation results. For phonemes, higher is better. While for the speaker and gender, a lower score indicates that the units success 'ignores' this information.

Model	Size	Speaker	Gender	Phoneme
CPC	50	1.35	0.66	47.30
	100	2.35	0.54	**48.45**
	200	3.70	1.62	47.74
	2000	10.39	4.14	44.06
HuBERT	50	0.73	0.03	42.49
	100	1.41	0.17	45.48
	200	1.95	0.21	46.64
	2000	5.15	0.65	43.32
MFCC	50	9.11	2.90	8.57
	100	11.54	3.97	8.73
	200	13.81	4.59	8.96

thesis. We decode the units back to speech using a look-up-table of the corresponding 20ms speech segments, then we transcribe the generated audio and measure the transcription error. Intuitively, in case of a strong correlation between the units and the phonemes - we can take a single "sound" to represent each unit - and apply the UTS step using the concatenation of these sound pieces. Notice that this approach differs from the one in [10] as there is no neural vocoder.

Formally, let u, l be sequences of deduplicated units and their length obtained by applying STU on the input audio x. and let x_i be part in x that is matched to the deduped unit, u_i. Notice x_i can be of arbitrary length.
Lookup Vocoder defines as :

$$LV(u,l) = \texttt{concat}(F(u_1,l_1),\dots,F(u_n,l_n)),$$
$$F(u_i,l_i) = \begin{cases} T[Key(u_i,l_i)], & \text{if } Key(u_i,l_i) \text{ in T} \\ x_i, & \text{else} \end{cases}, \qquad (1)$$

where T is a Look-up-table that stores for each key the corresponding x_i of the first appearance of this key, and Key maps unit and length into key. Note that the lookup table is filled along a random order iteration over the dataset.

We also measure the "memorization" of the process as the mean percentage of unseen keys in the lookup table.

We explore four different types of Key : (i) **Local-Single**- $Key(u_i) = (u_i)$; (ii) **Local-Full**- $Key(u_i) = (u_i, l_i)$; (iii) **Context-Single**- $Key(u_i) = (u_{i-1}, u_i, u_{i+1})$; (iv) **Context-Full**- $Key(u_i) = (u_{i-1}, u_i, u_{i+1}, l_i)$.

We analyze representations obtained by either HuBERT [2] or CPC [4] models considering a various number of clusters.

3.2. Circular Resynthesis

We introduce the Circular Resynthesis (CR) method, an utterly unsupervised evaluation metric that aims to measure the redundancies in the discrete units. We first perform a complete resynthesis procedure, in which we encode and decode the speech signal. Then, we apply an additional STU stage

Table 2. Units resynthesis character error rate (CER) and mean percentage of unseen keys (Memorization). The table contains results for different lookup key types: Local-Single (**L-S**), Local-Full (**L-F**), Context-Single **(C-S)** and Context-Full (**C-F**).

CER						
Model	**Size**	**Hifi-GEN**	**C-F**	**C-S**	**L-F**	**L-S**
CPC	50	5.95	9.12	25.36	39.57	60.98
	100	5.67	6.52	15.21	22.51	53.59
	200	5.37	5.12	10.16	15.18	40.65
HuBERT	50	7.31	10.31	14.96	47.24	58.42
	100	4.39	5.24	6.26	26.55	57.49
	200	4.10	4.25	4.69	15.56	19.88
MFCC	50	50.47	33.85	57.60	71.43	69.22
	100	44.68	15.79	46.55	67.54	66.13
	200	41.67	6.22	30.47	61.46	61.31
Memorization						
Model	**Size**	**Hifi-GEN**	**C-F**	**C-S**	**L-F**	**L-S**
CPC	50	-	2.66	0.17	0.043	0.006
	100	-	3.74	0.35	0.062	0.007
	200	-	5.52	0.72	0.085	0.010
HuBERT	50	-	2.35	0.33	0.028	0.006
	100	-	3.46	0.75	0.039	0.008
	200	-	4.91	1.49	0.056	0.012
MFCC	50	-	3.45	0.29	0.038	0.005
	100	-	7.09	1.01	0.052	0.006
	200	-	12.56	2.50	0.073	0.009

and measure the Unit-Edit-Distance (UED) between the first and the second units representing the speech. This metric was recently proposed by [14] to evaluate the robustness of discrete speech representation against signal variations. Intuitively, a high UED indicates redundancies in the discrete units. To reach the final CR metric, for each pair of units, we calculate the percentage of swaps between them over all the dataset's transcriptions.

3.3. Robust Clustering

Equipped with the CR metric, we explore three simple methods to improve the k-means clustering quality. In all three methods, we start from the standard k-means with $k = 2000$ and iteratively merge the clusters to reach the target number of clusters. The first method, named **Double K-means (K-K)**. In which we apply an additional k-means over the cluster centroids from the first k-means step. In the second method, denoted as **K-means with Hierarchical Clustering (K-H)**, we apply an agglomerative clustering over the cluster centroids from the first k-means step. The last method, named **K-means with Weighed Hierarchical Clustering (K-WH)**, we use an agglomerative clustering using a modified version of the euclidean distance, weighted by the CR metric. Formally, the distance metric is defined as follows:

$$D(i,j) = L2(c_i, c_j) \cdot [1 - \frac{CR(u_i,u_j) + CR(u_j,u_i)}{2}] \qquad (2)$$

Table 3. Comparing the different clustering methods using ABX and speaker information. For both metrics, lower is better.

Model	Size	ABX within				ABX across				Speaker probing			
		k-means	K-K	K-H	K-WH	k-means	K-K	K-H	K-WH	k-means	K-K	K-H	K-WH
CPC	50	5.66	**5.38**	9.62	8.80	7.83	**6.77**	11.46	10.56	42.22	32.96	19.26	**18.15**
	100	**5.42**	5.44	6.66	6.04	**7.07**	7.13	8.26	7.49	52.96	45.19	20.37	**15.56**
	200	5.53	**5.27**	5.61	5.68	7.35	**7.10**	7.28	7.13	63.70	49.63	26.30	**22.59**
HuBERT	50	7.23	5.67	**5.94**	6.12	8.93	**6.83**	7.43	7.67	**30.37**	36.30	36.67	31.85
	100	5.82	**5.01**	5.30	5.29	7.47	6.50	6.54	**6.32**	48.15	48.89	48.15	**46.67**
	200	5.79	5.24	5.18	**5.05**	7.49	6.42	6.46	**6.07**	65.19	61.11	**54.81**	62.96

while c_i, c_j are the ith and jth cluster continuous centroids, and u_i, u_j are the ith and jth discrete unit.

The intuition behind the last method is that clusters with high CR are more likely to be merged. Therefore, based on the understanding that CR measures redundancies (or similar acoustic concepts), these methods reduce redundancies and combine clustering representing similar acoustic concepts.

4. RESULTS

4.1. Datasets

We use the Librispeech[21] corpus to learn the k-means clustering (`train-clean-100`), and the `test-clean` to evaluate both the clustering methods and the lookup vocoder. We also use the Librispeech corpus to calculate the V-Measure for speaker and gender. For computing the V-Measure over phonemes, we use the TIMIT benchmark.

4.2. Analysis

Units Interpretation. Table 1 presents the V-Measure results regarding three different attributes - speaker, gender, and phoneme. The V-Measure for the speaker and gender scores is lower than the score of the phonemes, indicating a high correlation to the phonemes and a low correlation to the speaker or gender. In addition, when we check the effect of the number of the units- while for the speaker/gender, more units lead to a higher score, in the phoneme score, there is a max point both for the HuBERT and CPC configurations. Therefore, we claim that redundancies cause this trend in the units. Finally, we can see that CPC has a higher score for the phonemes- but also a higher score for speaker and gender.

Units Visualization. Figure 2 shows the spatial structure of the units. One can see that there is a very consistent structure- first, units that represent the same phoneme are usually close to each other. Moreover, phonemes from the same family (affricates, fricatives, Etc.') tend also to be close to each other. In addition, we can see that while for HuBERT and CPC, the space divide between the different phonemes families is generally equal, in the MFCC model, almost all the space uses for vowels. Notice redundancies in the clusters can also be observed from such figures.

Units Resynthesis. In Table 2, we show the results for the units' resynthesis. We can see that for some configurations, there is a slight difference between the HiFi-GAN and the lookup scores- this strengthens our understanding that units express fixed sounds and are mainly correlative to phonemes. We can see that the context of the units critically affects the results, while the unit's length has a lower effect. Finally, this understanding may help understand units' redundancies, i.e., the same phoneme in a different context will represent different units. In addition, we can see a correlation between CER and memorization. However, it is essential to note that the model architecture and the lookup method also significantly impact the overall performance.

4.3. Robust Clustering

We evaluate the proposed approach in two axes: **ABX**- We use the discrete representation and the cosine similarity metric to measure ABX within and across [22]; **speaker information**- We measure the accuracy of a simple DNN to predict the speaker given the discrete representation, similarly to [13][1].

Table 3 summarizes the results. We can see that the proposed methods improve both metrics outcomes for most of the configurations. Furthermore, the best results for ABX-across were obtained using CR- this strengthens our claim regarding the unit's redundancies.

5. CONCLUSION

In this work, we analyzed the GSLM discrete unit from three different and complementary points of view: interpretation, visualization, and resynthesis. The analysis showed a strong correlation between the units and the phonemes. In addition, we found redundancies in the units, which the units' context can explain. Finally, based on these understandings, we proposed methods that improve the unit's clustering.

Acknowledgements

We would like to acknowledge support for this research from the Israeli Science Foundation (ISF grant 2049/22).

[1]The model architecture is an encoding layer (dim 32) followed by a transformer (4 heads, hidden size 128, dropout 0.1) and a linear layer. Librispeech 'dev-clean' was used for training and 'test-clean' for evaluation. We applied STU on each utterance and used the units as input. We trained the model for 100 epochs using NLL loss and Adam optimizer (lr=0.001).

6. REFERENCES

[1] Shu-wen Yang et al., "Superb: Speech processing universal performance benchmark," *arXiv preprint arXiv:2105.01051*, 2021.

[2] 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*, vol. 29, pp. 3451–3460, 2021.

[3] Alexei Baevski et al., "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in Neural Information Processing Systems*, vol. 33, pp. 12449–12460, 2020.

[4] Morgane Riviere et al., "Unsupervised pretraining transfers well across languages," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7414–7418.

[5] Yehoshua Dissen, Felix Kreuk, and Joseph Keshet, "Self-supervised speaker diarization," *arXiv preprint arXiv:2204.04166*, 2022.

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

[7] Kushal Lakhotia, Eugene Kharitonov, Wei-Ning Hsu, Yossi Adi, Adam Polyak, Benjamin Bolte, Tu-Anh Nguyen, Jade Copet, Alexei Baevski, Abdelrahman Mohamed, et al., "On generative spoken language modeling from raw audio," *Transactions of the Association for Computational Linguistics*, vol. 9, pp. 1336–1354, 2021.

[8] Tu Anh Nguyen et al., "Generative spoken dialogue language modeling," *arXiv preprint arXiv:2203.16502*, 2022.

[9] 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.

[10] Adam Polyak et al., "Speech resynthesis from discrete disentangled self-supervised representations," *arXiv preprint arXiv:2104.00355*, 2021.

[11] Dan Wells, Hao Tang, and Korin Richmond, "Phonetic analysis of self-supervised representations of english speech," *Proc. Interspeech 2022*, 2022.

[12] Maureen de Seyssel, Marvin Lavechin, Yossi Adi, Emmanuel Dupoux, and Guillaume Wisniewski, "Probing phoneme, language and speaker information in unsupervised speech representations," *arXiv preprint arXiv:2203.16193*, 2022.

[13] Eugene Kharitonov et al., "textless-lib: a library for textless spoken language processing," *arXiv preprint arXiv:2202.07359*, 2022.

[14] Itai Gat, Felix Kreuk, Ann Lee, Jade Copet, Gabriel Synnaeve, Emmanuel Dupoux, and Yossi Adi, "On the robustness of self-supervised representations for spoken language modeling," *arXiv preprint arXiv:2209.15483*, 2022.

[15] Jonathan Shen et al., "Natural tts synthesis by conditioning wavenet on mel spectrogram predictions," in *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2018, pp. 4779–4783.

[16] Ryan Prenger et al., "Waveglow: A flow-based generative network for speech synthesis," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 3617–3621.

[17] Andrew Rosenberg and Julia Hirschberg, "V-measure: A conditional entropy-based external cluster evaluation measure," in *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*, 2007, pp. 410–420.

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

[19] Laurens van der Maaten and Geoffrey Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. 86, pp. 2579–2605, 2008.

[20] Franz Aurenhammer, "Voronoi diagrams—a survey of a fundamental geometric data structure," *ACM Computing Surveys (CSUR)*, vol. 23, no. 3, pp. 345–405, 1991.

[21] Vassil Panayotov et al., "Librispeech: an asr corpus based on public domain audio books," in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[22] Jacob Kahn et al., "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)*. IEEE, 2020, pp. 7669–7673.