Speech-Based Language Modelling

李思瑞

2022.04.22

Baseline

ZeroSpeech 2021 is a challenge aimed at Spoken Language Modelling. This task consists in learning language models directly from raw audio in an unknown language, without any annotation or text.

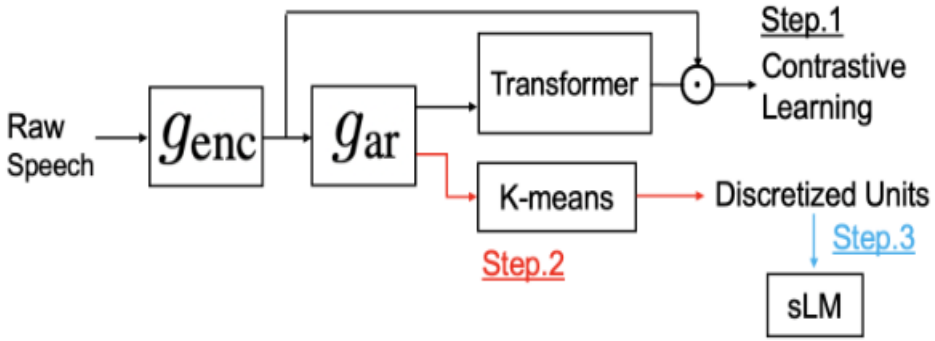

Figure 1: *Illustration of the baseline system. First, we train a Contrastive Predictive Coding (CPC) model which consists of g_{enc} and g_{ar} optimized by Eq. (4) (Step.1). Then, k-means clustering is performed to generate discretized units of audio data (Step.2). Finally, we train a spoken language model (sLM) using the discretized units as pseudo-labels (Step.3).*

Table 1: **Summary description of the four Zero Resouce Benchmark 2021 metrics**. The metrics in light blue use a pseudo-distance d between embeddings (d_h being from human judgments), the metrics in light orange use a pseudo-probability p computed over the entire input sequence.

Linguistic level	Metrics	Dataset	Task	Example
acoustic-phonetic	ABX	Libri-light	$d(a,x) < d(b,x)$? $a \in A, b \in B$, $x \neq a \in A$	within-speaker: $(\text{apa}_{s_1}, \text{aba}_{s_1}, \text{apa}_{s_1})$ across-speaker: $(\text{apa}_{s_1}, \text{aba}_{s_1}, \text{apa}_{s_2})$
lexicon	spot-the-word	sWUGGY	$p(a) > p(b)$?	(brick, blick) (squalled, squilled)
lexical semantics	similarity judgement	sSIMI	$d(a,b) \propto d_h(a,b)$?	(abduct, kidnap) : 8.63 (abduct, tap) : 0.5
syntax	acceptability judgment	sBLIMP	$p(a) > p(b)$?	(dogs eat meat, dogs eats meat) (the boy can't help himself, the boy can't help herself)

Phonetics: Libri-light ABX metrics. The ABX metric consists in computing, for a given contrast between two speech categories A and B (e.g., the contrast between triphones 'aba' and 'apa'), the probability that two sounds belonging to the same category are closer to one another than two sounds that belong to different categories. Formally, we compute an asymmetric score, with a and x, different tokens belonging to category A (of cardinality n_A) and b belonging to B (n_B), respectively:

$$\hat{e}(A,B) := \frac{1}{n_A(n_A-1)n_B} \sum_{\substack{a,x\in A \\ x\neq a}} \sum_{b\in B} \left[\mathbb{1}_{d(b,x)<d(a,x)} + \frac{1}{2}\mathbb{1}_{d(b,x)=d(a,x)}\right]$$

Lexical semantics: sSIMI similarity metrics

$$d_{SEM}(x,y) = sim\left(f_{pool}\left(h^{(i)}(q_1^x ... q_T^x)\right), f_{pool}\left(h^{(i)}(q_1^y ... q_S^y)\right)\right),$$

where f_{pool} is the pooling function and $h^{(i)}(\cdot)$ is the output of the i^{th} hidden layer of the LM.

Lexicon: sWUGGY spot-the-word metrics

Syntax: sBLIMP acceptability metrics

$$\text{span-PP}_{M_d,\Delta t}(q_1..q_T) = \prod_{\substack{i=1+j\Delta t \\ \lfloor (T-1)/\Delta t \rfloor \geq j \geq 0}} P(q_i..q_{i+M_d}|q_1..q_{i-1}q_{i+M_d+1}..q_T),$$

where M_d is a chosen decoding span size, and Δt is a temporal sliding size. For the LSTM model, we computed the probability of the discretized sequence with the classic left-to-right scoring style obtained by the chain rule: $P(q_1..q_T) = \prod_{i=1}^{T} P(q_i|q_1..q_{i-1})$.

Table 2: **Characteristics of the baseline acoustic CPC models**. We took the last LSTM layer of CPC-small and the second LSTM hidden layer of CPC-big as inputs to the clustering as they give the best ABX scores (Supplementary Table S1).

Model	CPC configuration		Training data	Input to kmeans
	autoregressive	hidden units		
CPC-small	2-layer LSTM	256	LibriSpeech clean-100	LSTM level 2
CPC-big	4-layer LSTM	512	Libri-light clean-6k	LSTM level 2

Table 3: **Characteristics of the baseline LMs.** L refers to the number of hidden layers; ED, HD and FFD refer to the dimension of the embedding layer, hidden layer, and feed-forward output layer respectively; H refers to the number of attention heads in the BERT case.

Model	Architecture					nb parameters	Train data	Compute Budget
	L	ED	HD	FFD	H			
BERT	12	768	768	3072	12	90M	LS960	48h - 32 GPUs
BERT-small	8	512	512	2048	8	28M	LS960	60h - 1GPU
LSTM	3	200	1024	200	-	22M	LS960	60h- 1GPU

Speech Representation Learning Combining Conformer CPC with Deep Cluster for the ZeroSpeech Challenge 2021

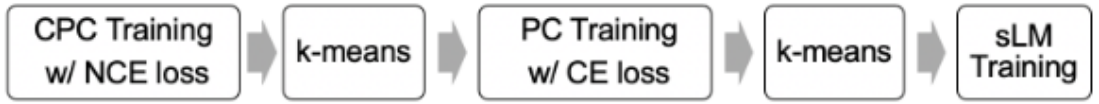

Figure 2: *Illustration of our proposed system (CPC with deep cluster). First, we train a CPC model which consists of g_{enc} and g_{ar} optimized by Eq. (4). Then, k-means clustering is performed to generate discretized units of audio data. Next, another CPC network is trained for phoneme classification (PC) using the discretized units as pseudo-labels. After that, we obtain more linguistically discriminative representation by second-round clustering. Finally, we train a sLM based on pseudo-labels.*

$$\mathcal{L}_t = -\frac{1}{K}\sum_{k=1}^{K}\log\left[\frac{\exp(z_{t+k}^T h_k(c_t))}{\sum_{\tilde{z}\in\mathcal{N}_t}\exp(\tilde{z}^T h_k(c_t))}\right], \quad (1)$$

$$\tilde{c} = c + \frac{1}{2}\mathrm{FFN}(c), \quad (3)$$

$$c' = \tilde{c} + \mathrm{MHSA}(\tilde{c}), \quad (4)$$

$$c'' = c' + \mathrm{Conv}(c'), \quad (5)$$

$$y = \mathrm{Layernorm}(c'' + \frac{1}{2}\mathrm{FFN}(c'')) \quad (6)$$

Table 2: *Within (all stimuli a, b and x in Eq. (2) are uttered by the same speaker) and Across (a and b are from the same speaker, and x from a different speaker) Speaker ABX metric (lower is better) on Libri-light dev-clean and dev-other. All embeddings are extracted from the final layer of the autoregressive network before clustering. "DC" stands for the deep cluster. "1st" of Training Data means a data set for contrastive learning and "2nd" of that means a data set for phoneme classification. Each model is trained on LibriSpeech (LS) or Libri-light (LL).*

Embedding	**Training Data**		**within (↓)**		**across (↓)**	
	1st	2nd	dev-clean	dev-other	dev-clean	dev-other
Baseline : CPC-small	LS-100h	/	6.24	8.48	8.17	13.55
Baseline : CPC-small	LS-460h	/	6.19	7.34	8.71	13.02
Proposed: Conformer CPC-small	LS-100h	/	5.78	7.83	8.23	13.59
Proposed: Conformer CPC-small	LS-460h	/	5.40	7.17	7.55	12.19
Proposed: CPC-small+DC	LS-100h	LS-100h	4.78	6.78	7.01	12.34
Proposed: CPC-small+DC	LS-460h	LS-460h	**3.93**	**5.18**	**5.99**	**10.00**
Proposed: Conformer CPC-small+DC	LS-460h	LS-460h	4.05	5.38	6.12	10.60
Baseline : CPC-big	LL-6kh	/	3.41	4.18	**4.85**	**7.64**
Proposed: CPC-big+DC	LL-6kh	LS-960h	3.28	4.14	4.96	8.28
Proposed: CPC-big+DC (1024units)	LL-6kh	LS-960h	**3.11**	**3.98**	4.96	7.92

Table 3: *Overall performance (higher is better) of the baseline and the proposed models on dev sets on three zero-shot metrics. For all models, the k-means clustering (k=50) was performed on LibriSpeech clean-100h, and the BERT-small models were trained on discretized units of LibriSpeech 960h.*

System	**Training Data**		**sWUGGY (↑)**	**sBLIMP (↑)**	**sSIMI (↑)**	
	1st	2nd			synth.	libri.
Baseline : CPC-small	LS-100h	/	65.79	52.88	-0.09	**9.23**
Baseline : CPC-small	LS-460h	/	66.21	52.79	-0.67	4.92
Proposed: Conformer CPC-small	LS-100h	/	62.22	52.96	**0.90**	7.22
Proposed: Conformer CPC-small	LS-460h	/	66.10	**53.39**	-1.84	5.17
Proposed: CPC-small+DC	LS-100h	LS-100h	65.42	52.86	-1.10	8.14
Proposed: CPC-small+DC	LS-460h	LS-460h	64.89	52.75	-2.11	8.89
Proposed: Conformer CPC-small+DC	LS-460h	LS-460h	**67.21**	53.38	-0.17	7.07
Baseline : CPC-big	LL-6kh	/	65.81	52.91	**3.88**	**5.56**
Proposed: CPC-big+DC	LL-6kh	LS-960h	**66.01**	**54.15**	-0.81	5.45
Proposed: CPC-big+DC (1024units)	LL-6kh	LS-960h	62.64	54.06	-1.65	4.81

Information Retrieval for ZeroSpeech 2021: The Submission by University of Wroclaw

- Factoring Out Speaker Identities

Factoring Out Speaker Identities The embeddings produced by CPC contain information about both the phonetic content and speaker identity. In case of ABX, which is a phoneme recognition metric, the latter is irrelevant. We therefore project the embeddings of the baseline model (CPC-big [2]) into the nullspace of a linear speaker classification model to render the embeddings less speaker-sensitive. We perform speaker classification on baseline CPC embeddings with a projection factorized into matrices A and B, where $A \in \mathbb{R}^{D_{inb} \times D_{emb}}$, $B \in \mathbb{R}^{D_{spk} \times D_{inb}}$, D_{emb} is the dimensionality of embeddings and D_{inb} is the linear bottleneck dimensionality. In order to compute ABX, we multiply the CPC-derived embeddings by $A' \in \mathbb{R}^{(D_{emb} - D_{inb}) \times D_{emb}}$, the nullspace matrix of A.

- Averaging with Centroids

Specifically, we take a weighted average of every dense CPC-derived embedding e in the embedding space with its cluster centroid c_e:

$$\hat{e} = \alpha\, c_e + (1 - \alpha)\, e. \qquad (1)$$

Table 1: *ABX error rates (%, cosine distance) for multiple sizes of nullspaces of speaker classification models. The nullspace dimension complements the bottleneck dimension used to train the speaker recognizer.*

Evaluation			Nullspace dimensionality: None	464	448	416	320	256
dev	clean	within	3.38	3.28	**3.25**	3.29	3.26	3.31
	clean	across	4.17	3.98	**3.94**	3.92	3.98	3.99
	other	within	4.81	4.63	**4.60**	4.61	4.62	4.67
	other	across	7.53	7.34	**7.24**	7.24	7.21	7.26

Table 3: *ABX error rates (%, cosine distance) for weighted averaging of CPC embeddings with centroids. The bottom half shows results combined with the best 448-dimensional nullspace setup. The nullspace dimension is equal to the difference of dimensions between the embeddings and the bottleneck used to train the speaker classifier. Phoneme classification results in Table 4*

Evaluation			Centroid weight: None	0.2	0.3	0.4	0.5	0.6
dev	clean	within	3.43	3.23	3.16	**3.09**	3.07	3.22
	clean	across	4.20	3.96	3.84	**3.76**	3.77	3.97
	other	within	4.84	4.64	4.62	**4.61**	4.75	5.19
	other	across	7.63	7.38	7.28	**7.32**	7.53	7.93
dev + nullspace	clean	within	3.25	3.03	2.97	**2.94**	2.93	3.10
	clean	across	3.94	3.66	3.60	**3.58**	3.57	3.75
	other	within	4.60	4.49	4.47	**4.47**	4.66	4.98
	other	across	7.24	7.05	6.94	**7.02**	7.21	7.67

Table 5: *Scores for DTW lookup on different quantizations, and with linear and optimal distance matrices. The results were computed for the base dev set of the lexical task (no OOV subset). We used the train-full-960 subset of the LibriSpeech as dictionary.*

Quantization	Distance matrix	Classification accuracy no norm.	norm.
Baseline	none (constant)	68.47%	69.33%
Baseline	Euclidean	68.94%	70.98%
Baseline	Euclidean^2	71.00%	71.64%
Cosine	cosine	72.61%	73.36%
Cosine	cosine$^{1.6}$	73.12%	73.92%

Table 7: *Correlation between human judgments and system responses (× 100). For other contestants the best submission on the test part of the data is presented.*

	synth.		libri.	
Method	**dev**	**test**	**dev**	**test**
LSTM Baseline	4.42	7.35	7.07	2.38
BERT Baseline	6.25	5.17	4.35	2.48
Ours	5.90	2.42	10.20	9.02
van Niekerk et al.	4.29	9.23	7.69	-1.14
Liu et al.	3.16	7.30	1.79	-4.33
Maekaku et al.	-2.10	6.74	8.89	2.03

LSTM language model trained on quantized nullspace features from LibriSpeech dev subset. In the competition, it had 53% accuracy both on dev and test sets, slightly outperforming the baseline, and being close to 54% of the best submission.

Analyzing Speaker Information in Self-Supervised Models to Improve Zero-Resource Speech Processing

representation structures this information. We hypothesize that the per-utterance mean of the features captures a large degree of the speaker information. This is reasonable under the assumption that speaker identity remains constant over an utterance with phonetic content varying over shorter time scales [20].

utterance (or set of utterances) from a single speaker, we remove speaker information from the CPC features by subtracting the mean and scaling to unit variance. In the remainder of the paper,

Figure 1: *UMAP visualizations of CPC features. (a) The per-utterance means of CPC features for six speakers. (b) Per-frame CPC features for the blue and purple speakersin (a). (c) Per-frame CPC features (standardized per utterance) for the same speakers.*

Table 1: *Speaker verification results for the supervised topline and the CPC- and MFCC-based systems.*

	EER (%)	Accuracy (%)
Topline: GE2E	1.6	98.8
Proposed: Mean of CPC	6.7	95.8
Baseline: Mean of MFCCs	19.8	59.8

Table 2: *Probing experiments where phone, speaker and gender classifiers are trained on CPC features. Clustering is performed on the CPC features using K-means with 50 clusters.*

		Accuracy (%)		
Standardized	Clustered	Phone	Speaker	Gender
Linear classifiers:				
✗	✗	75.7	93.4	96.7
✓	✗	77.0	14.8	55.3
✗	✓	46.6	3.4	53.5
✓	✓	48.5	3.1	50.9
Non-linear classifiers:				
✗	✗	80.1	99.5	99.8
✓	✗	79.7	89.0	98.1

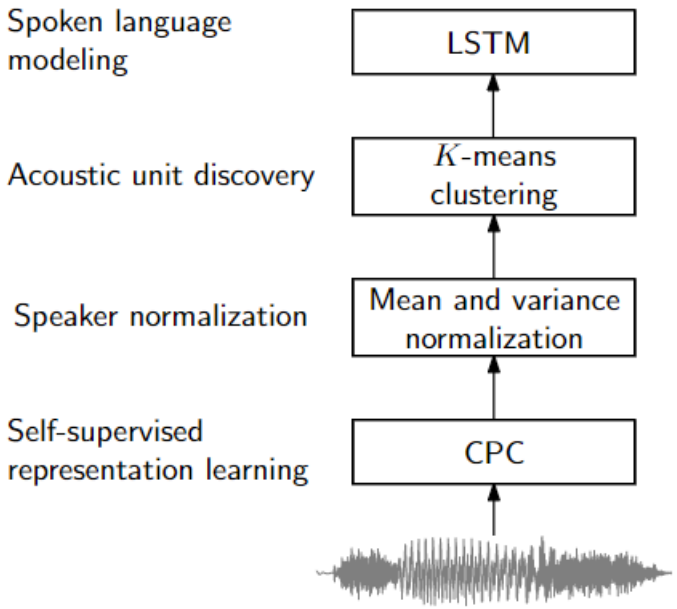

Figure 2: *We propose a speaker normalization method for CPC features. We incorporate speaker normalization into an acoustic unit discovery system (based on K-means clustering) and spoken*

Table 3: *ABX error rates for CPC features and MFCCs.*

		Within (%)		Across (%)	
Standardized	Clustered	clean	other	clean	other
CPC features:					
✗	✗	3.41	4.85	4.18	7.64
✓	✗	3.41	4.81	4.12	7.49
✗	✓	6.38	10.22	8.26	14.86
✓	✓	5.38	8.80	6.56	12.79
Baseline: MFCCs		10.95	13.55	20.94	29.4

Table 6: *ABX results after pruning the CPC dimensions that are least informative for predicting speaker.*

# features	64	128	192	256	320	384	512
Within	8.88	7.74	7.05	6.88	**6.79**	7.04	7.09
Across	11.90	10.62	9.97	9.64	**9.44**	9.49	9.68

Table 7: *Results on the lexical, syntactic, and semantic spoken language modeling tasks.*

			Semantic	
	Lexical	Syntactic	Synth.	Libri.
Topline:				
Forced Align	92	63	8.5	2.4
Phone	**98**	67	12.2	20.2
RoBERTa	96	**82**	**33.2**	**27.8**
High budget:				
BERT baseline	68	56	6.3	2.5
Low budget:				
LSTM baseline	61	53	7.4	**2.4**
LSTM speaker-norm	**65**	**54**	**9.2**	-1.1
Chorowski et al. [28]	64	53	5.2	-0.9
Maekaku et al. [29]	61	**54**	7.0	-1.2