Unsupervised Speech Recognition

Facebook AI & Google AI
(May, 2021)

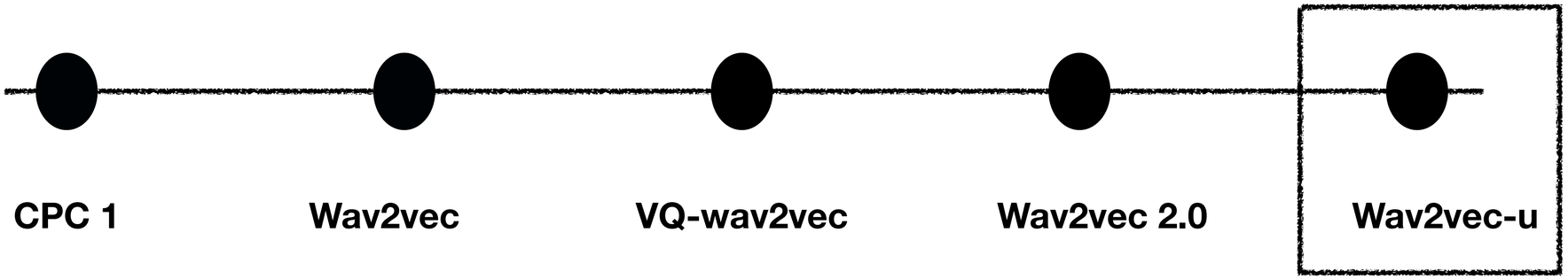
이준형

Overview

- Intro
- Wave2vec 2.0 Architecture
- Wave2vec-u Architecture
- Unsupervised learning
- Experiments
- Objective
- Conclusion

Intro

- CPC1: Contrastive Predictive Coding을 다양한 Task에 적용
- Wav2vec: 음성에 CPC를 적용하는 방법



Intro - Contrastive Predictive Coding(CPC)

- CNN계열의 인코더와 Aggregate data로 구성된 Architecture
- Extract continuous vector

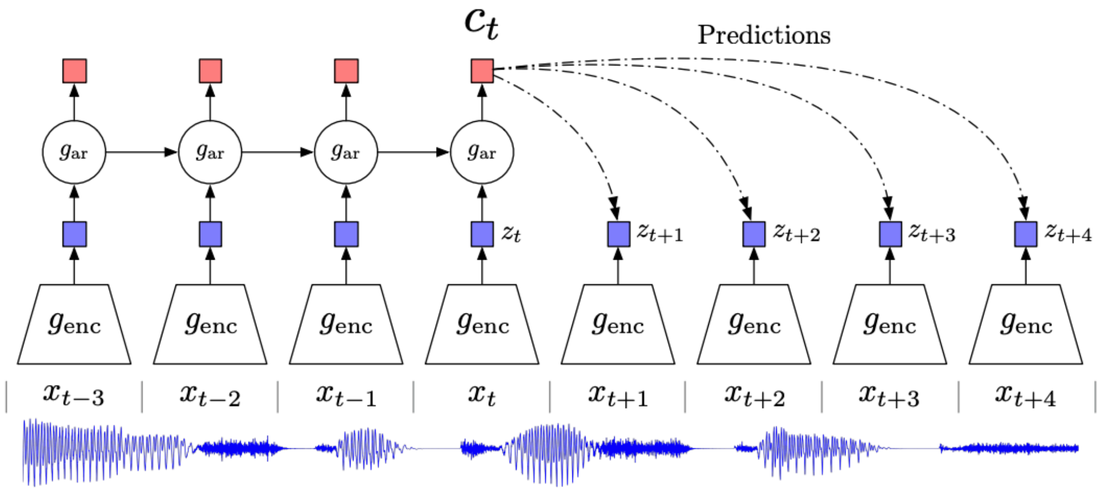
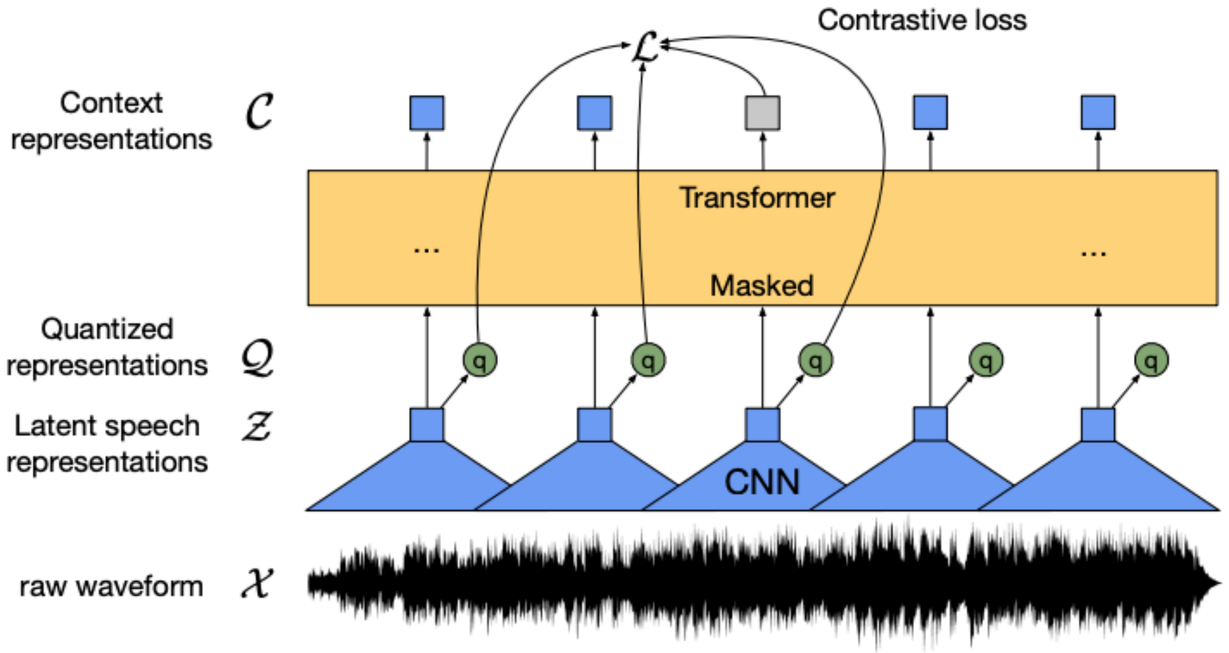


Figure 1: Overview of Contrastive Predictive Coding, the proposed representation learning approach. Although this figure shows audio as input, we use the same setup for images, text and reinforcement learning.

wav2vec 2.0 Architecture

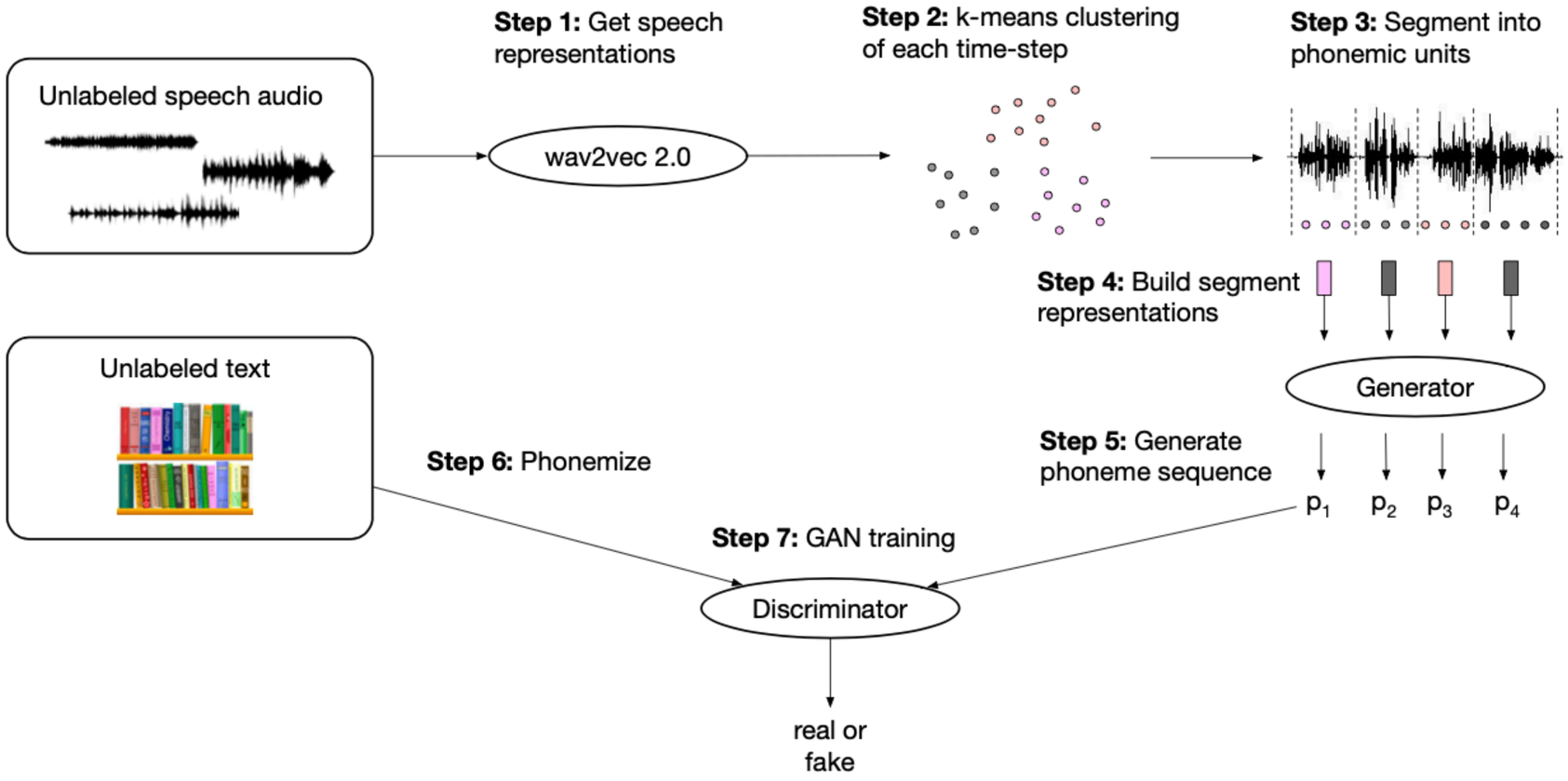
- Masks Spans of latent speech representations
- Self-supervised learning
- Train masked area with Contrastive loss



Intro - What's so special about wav2vec-u?

- Speech Recognition system that **don't require transcribed data**
- wav2vec-U (wav2vec Unsupervised) - Method to train speech recognition models **without any labeled data**.
- Leveraged **self-supervised** speech representations
- TIMIT dataset **phoneme error rate 26.1 -> 11.3**
- Librispeech dataset word **error rate of 5.9 on test-other**
- **Gain is significant for languages** that do not have specially prepared and labeled training data. (Amharic, Swahili, Kyrgyz, and Tatar)

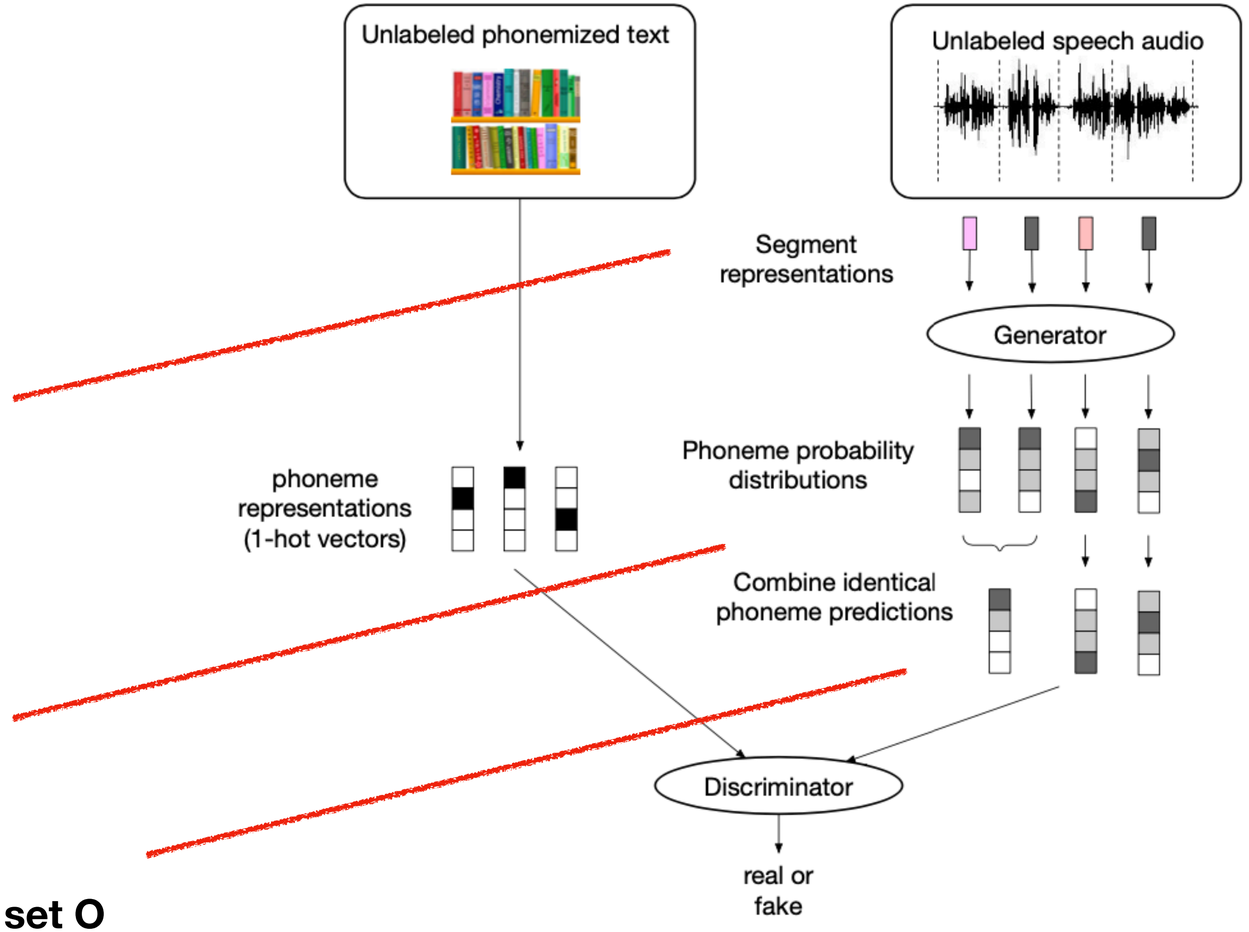
wav2vec-u Architecture



wav2vec-u Architecture

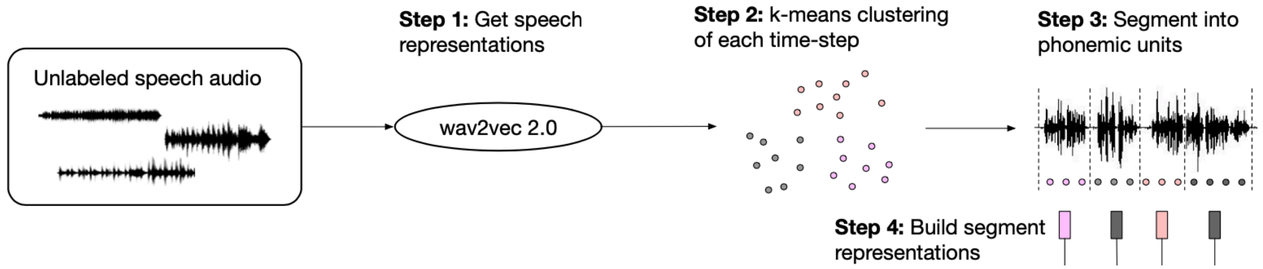
sequence of T segment representations S=[s1,...,sT]

mapped to a sequence of M phonemes P = [p1,...,pM]



set O

Segmenting Audio Signal

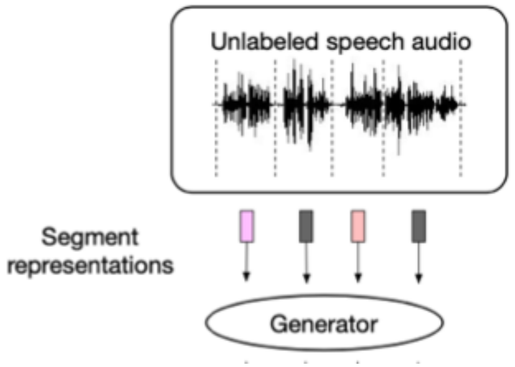


- Clustering the wav2vec 2.0 speech representations(c1,…,cT)
- Unlabeled speech data -> K-means clustering to identify K = 128clusters. (FAISS library)
- Label the data with cluster ID. c_t is labeled with the corresponding cluster ID and introduce speech segment boundaries whenever the cluster ID changes

$$\text{cluster ID } i_t \in \{1, \ldots, K\}$$

- Once speech audio representations are segmented, we compute a 512-dimensional PCA over all speech representations output by wav2vec 2.0 for the training set

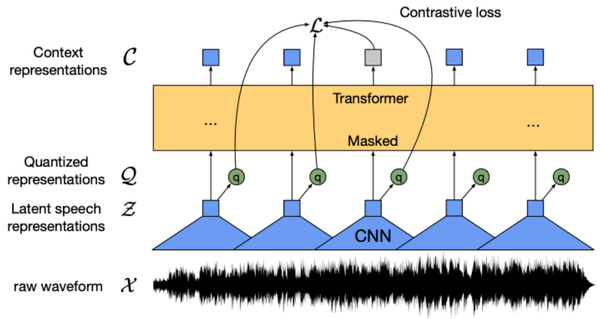
Segmenting Audio Signal

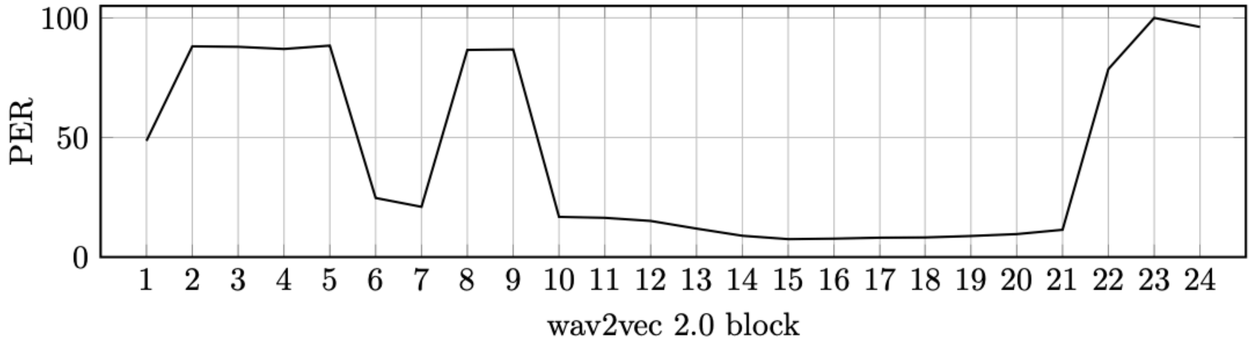


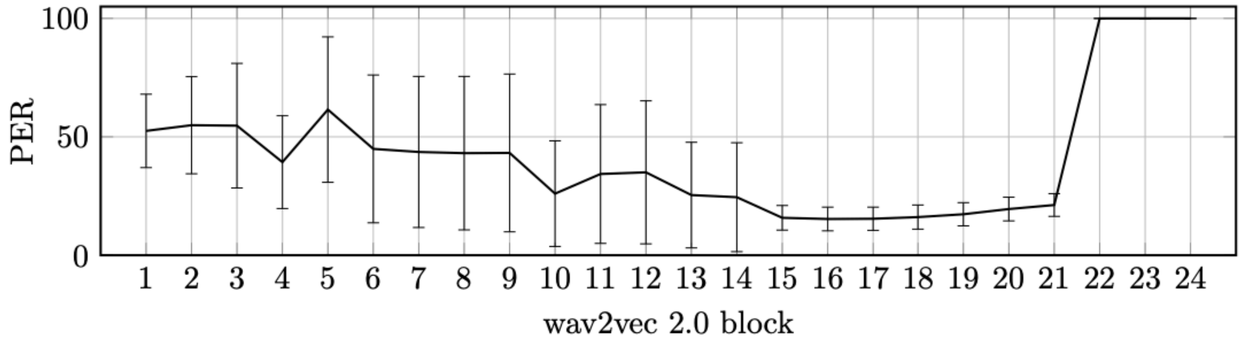
- Mean-pool the PCA representations for a particular segment to obtain an average representations of the segment
- PCA retains only the most important features and we found this to be effective. Segment boundaries are noisy due to lack of supervision so also mean-pool pairs of adjacent segment representations to increase robustness
- S = s1,…sT for a given utterance

Experiments

- 1. Removing Silences(rVAD)
- 2. Choosing Speech Representations

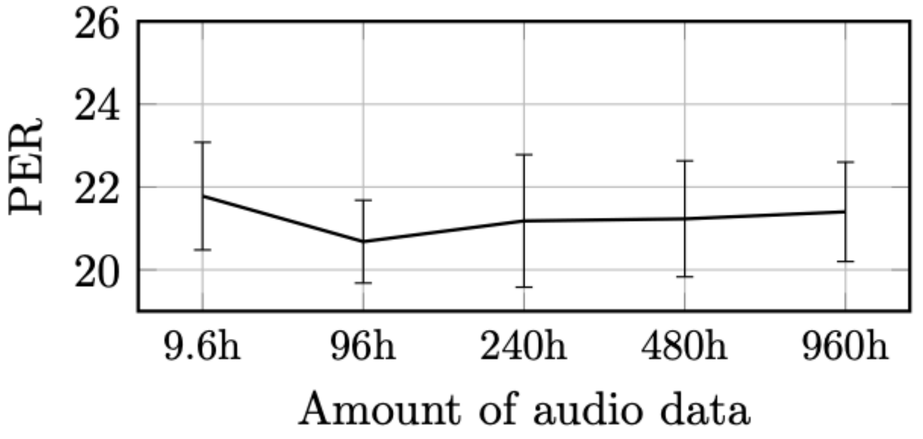


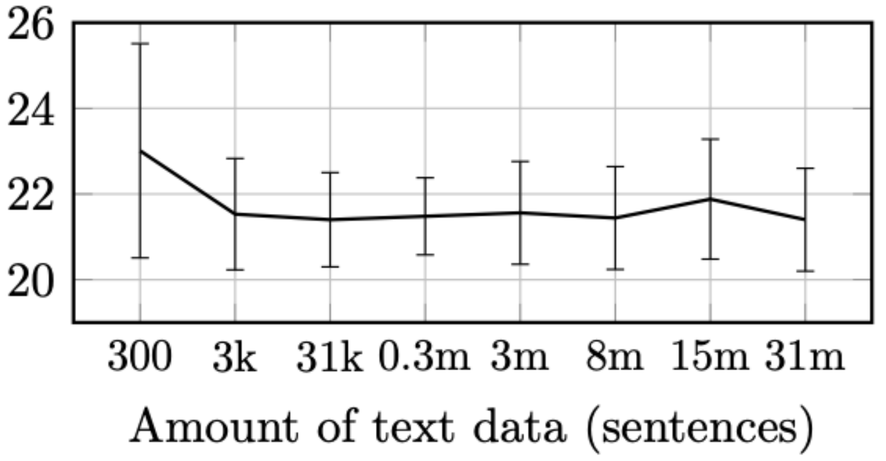




Experiments

- How much data is sufficient to achieve good performance?
- 9.6h of speech audio
- 3,000 sentences of text data
- Both were sufficient to achieve excellent performance





Objective

$$\min_{\mathcal{G}} \max_{\mathcal{C}} \quad \mathop{\mathbb{E}}_{P^r \sim \mathcal{P}^r} [\log \mathcal{C}(P^r)] - \mathop{\mathbb{E}}_{S \sim \mathcal{S}} [\log (1 - \mathcal{C}(\mathcal{G}(S)))] - \lambda \mathcal{L}_{gp} + \gamma \mathcal{L}_{sp} + \eta \mathcal{L}_{pd}$$

- 1. Gradient Penalty
- 2. Segment Smoothness Penalty
- 3. Phoneme diversity loss

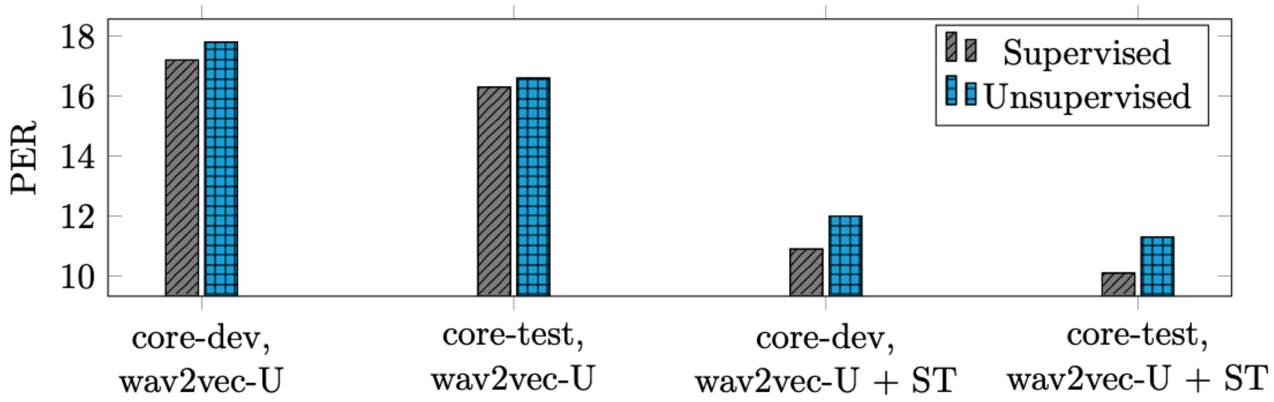
$$\mathcal{L}_{gp} = \mathop{\mathbb{E}}_{\tilde{P} \sim \tilde{\mathcal{P}}} \left[\left(\|\nabla \mathcal{C}(\tilde{P})\| - 1 \right)^2 \right]$$

$$\mathcal{L}_{sp} = \sum_{(p_t, p_{t+1}) \in \mathcal{G}(S)} \|p_t - p_{t+1}\|^2$$

$$\mathcal{L}_{pd} = \frac{1}{|B|} \sum_{S \in B} -H_{\mathcal{G}}(\mathcal{G}(S))$$

Unsupervised Cross-Validation Metric

- 1. language model entropy
- 2. Vocabulary usage



Self Training

- For self-training, perform TWO iterations
- First, we pseudo-label the training data with the Unsupervised GAN model and train an HMM on the pseudo-labels.
- Second, we relabel the training data with the HMM and then fine-tune the original wav2vec 2.0 model using the HMM pseudo-labels with a CTC loss.

Results

Model	Unlabeled data	LM	dev		test	
			clean	other	clean	other
960h - Supervised learning						
DeepSpeech 2 (Amodei et al., 2016)	-	5-gram	-	-	5.33	13.25
Fully Conv (Zeghidour et al., 2018)	-	ConvLM	3.08	9.94	3.26	10.47
TDNN+Kaldi (Xu et al., 2018)	-	4-gram	2.71	7.37	3.12	7.63
SpecAugment (Park et al., 2019)	-	-	-	-	2.8	6.8
SpecAugment (Park et al., 2019)	-	RNN	-	-	2.5	5.8
ContextNet (Han et al., 2020)	-	LSTM	1.9	3.9	1.9	4.1
Conformer (Gulati et al., 2020)	-	LSTM	2.1	4.3	1.9	3.9
960h - Self and semi-supervised learning						
Transf. + PL (Synnaeve et al., 2020)	LL-60k	CLM+Transf.	2.00	3.65	2.09	4.11
IPL (Xu et al., 2020b)	LL-60k	4-gram+Transf.	1.85	3.26	2.10	4.01
NST (Park et al., 2020)	LL-60k	LSTM	1.6	3.4	1.7	3.4
wav2vec 2.0 (Baevski et al., 2020c)	LL-60k	Transf.	1.6	3.0	1.8	3.3
wav2vec 2.0 + NST (Zhang et al., 2020b)	LL-60k	LSTM	1.3	2.6	1.4	2.6
Unsupervised learning						
wav2vec-U LARGE	LL-60k	4-gram	13.3	15.1	13.8	18.0
wav2vec-U LARGE + ST	LL-60k	4-gram	3.4	6.0	3.8	6.5
	LL-60k	Transf.	3.2	5.5	3.4	5.9

Table 2: WER on the Librispeech dev/test sets when using 960 hours of unlabeled audio data from Librispeech (LS-960) or 53.2k hours from Libri-Light (LL-60k) using representations from wav2vec 2.0 LARGE. Librispeech provides clean dev/test sets which are less challenging than the other sets. We report results for GAN training only (wav2vec-U) and with subsequent self-training (wav2vec-U + ST).

Model	LM	core-dev	core-test	all-test
Supervised learning				
LiGRU (Ravanelli et al., 2018)	-	-	14.9	-
LiGRU (Ravanelli et al., 2019)	-	-	14.2	-
Self and semi-supervised learning				
vq-wav2vec (Baevski et al., 2020b)	-	9.6	11.6	-
wav2vec 2.0 (Baevski et al., 2020c)	-	7.4	8.3	-
Unsupervised learning - matched setup				
EODM (Yeh et al., 2019)	5-gram	-	36.5	-
GAN* (Chen et al., 2019)	9-gram	-	-	48.6
GAN + HMM* (Chen et al., 2019)	9-gram	-	-	26.1
wav2vec-U	4-gram	17.0	17.8	16.6
wav2vec-U + ST	4-gram	11.3	12.0	11.3
Unsupervised learning - unmatched setup				
EODM (Yeh et al., 2019)	5-gram	-	41.6	-
GAN* (Chen et al., 2019)	9-gram	-	-	50.0
GAN + HMM* (Chen et al., 2019)	9-gram	-	-	33.1
wav2vec-U*	4-gram	21.3	22.3	24.4
wav2vec-U + ST*	4-gram	13.8	15.0	18.6

Table 3: TIMIT Phoneme Error Rate (PER) in comparison to previous work for the matched and unmatched training data setups (§ 5.1). PER is measured on the standard Kaldi dev and test sets (core-dev/core-test) as well as a slightly larger version of the test set (all-test) as used by some of the prior work. (*) indicates experiments that do not use the standard split excluding SA utterances.

Results

Model	Labeled data used	LM	de	nl	fr	es	it	pt	Avg
Labeled training hours (full)			2k	1.6k	1.1k	918	247	161	
Supervised learning									
Pratap et al. (2020)	full	5-gram	6.49	12.02	5.58	6.07	10.54	19.49	10.0
Unsupervised learning									
wav2vec-U	0h	4-gram	32.5	40.2	39.8	33.3	58.1	59.8	43.9
wav2vec-U + ST	0h	4-gram	11.7	21.4	14.6	10.9	24.1	28.7	18.6

Table 4: WER on the Multilingual Librispeech (MLS) dataset using representations from the wav2vec 2.0 XLSR-53 model. We consider German (de), Dutch (nl), French (fr), Spanish (es), Italian (it), Portuguese (pt).

Model	tt	ky
Supervised learning		
Fer et al. (2017)	42.5	38.7
m-CPC (Rivière et al., 2020)	42.0	41.2
XLSR-53 (Conneau et al., 2020)	5.1	6.1
Unsupervised learning		
wav2vec-U	25.7	24.1
wav2vec-U + HMM	13.7	14.9

Table 5: PER for low-resource languages, Tatar (tt) and Kyrgyz (ky).

Model	sw
Supervised learning	
Besacier et al. (2015)	27.36
Unsupervised learning	
wav2vec-U	52.6
wav2vec-U + ST	31.0

Table 6: WER for Swahili from the ALFFA corpus. We compare to the supervised baseline of the ALFFA project.

Future Trend

- More unsupervised learning
- Aim to make it work for many other languages
- How can we implement korean language to wave2vec architecture?