wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations

김 수

목차

- Introduction
- Model
- Training
- Results
- Conclusion

Introduction

- Self-supervised learning
  - ultra-low resource speech recognition
    - 라벨링된 10분 데이터로 학습 시 4.8(clear)/8.2(other) Word Error Rate(WER)
    - 라벨링된 960시간 데이터로 학습 시 1.8(clear)/3.3(other) WER

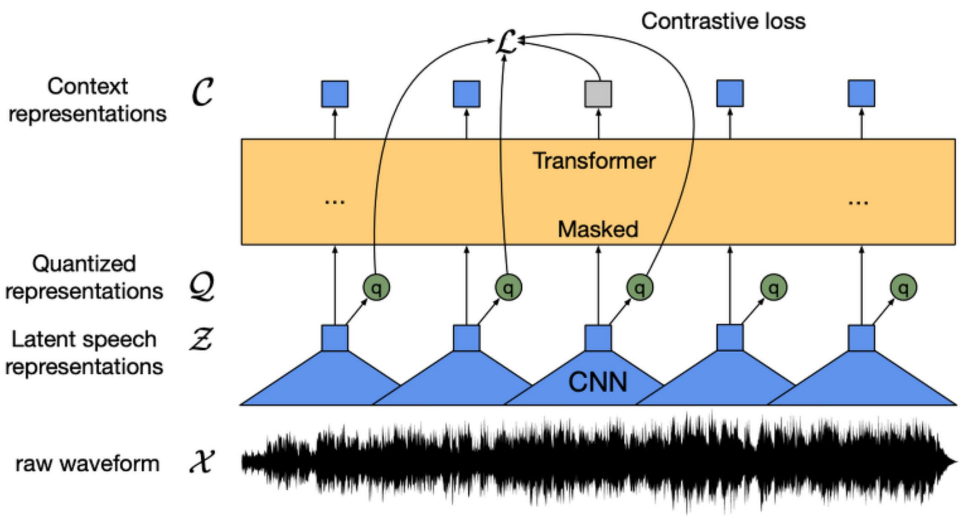
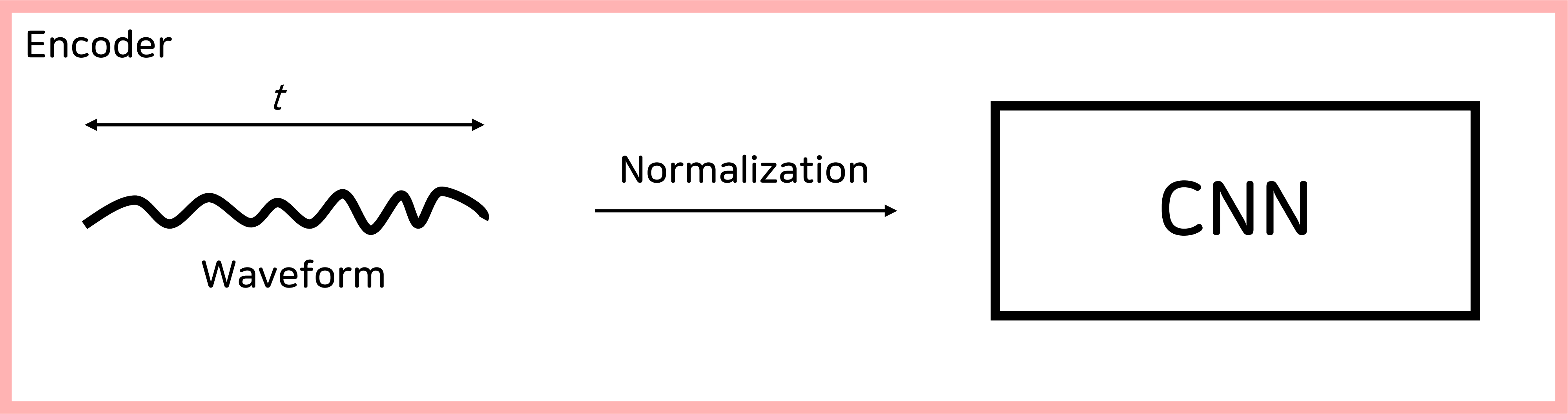


Figure 1: Illustration of our framework which jointly learns contextualized speech representations and an inventory of discretized speech units.

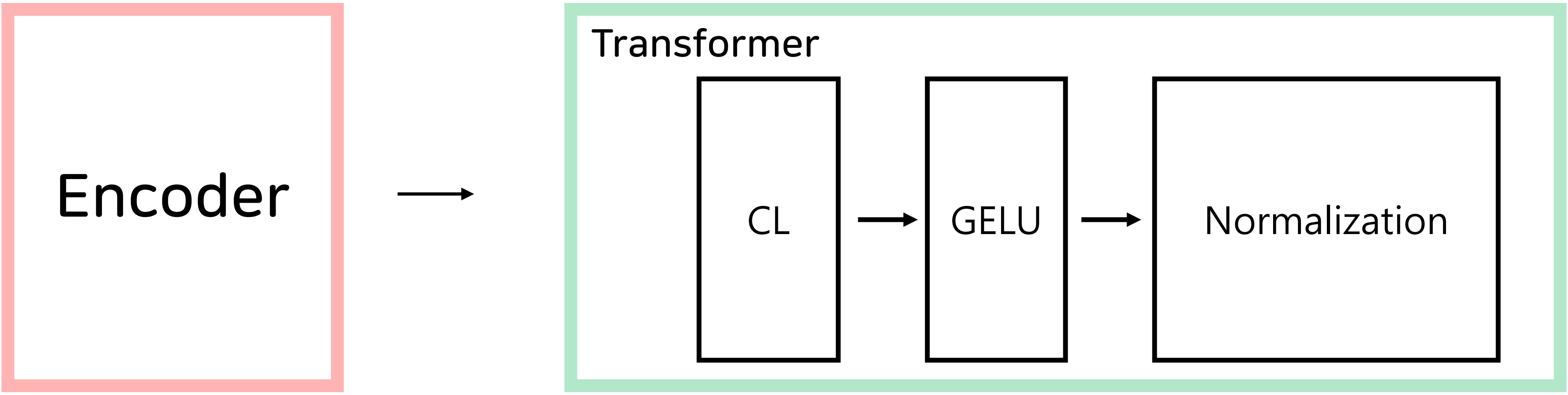
Model

- Feature encoder
  - Several blocks containing a temporal convolution followed by layer normalization
  - GELU activation function



Model

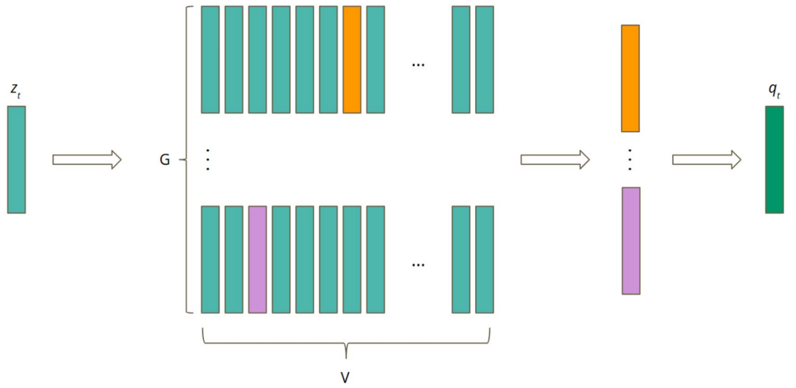
- Contextualized representations with Transformers
  - Relative positional embedding 작용하는 CNN 사용함



Model

- Quantization model
  - Codebook : 신호나 데이터의 표현
  - Gumbel softmax

Encoder → Quantization model

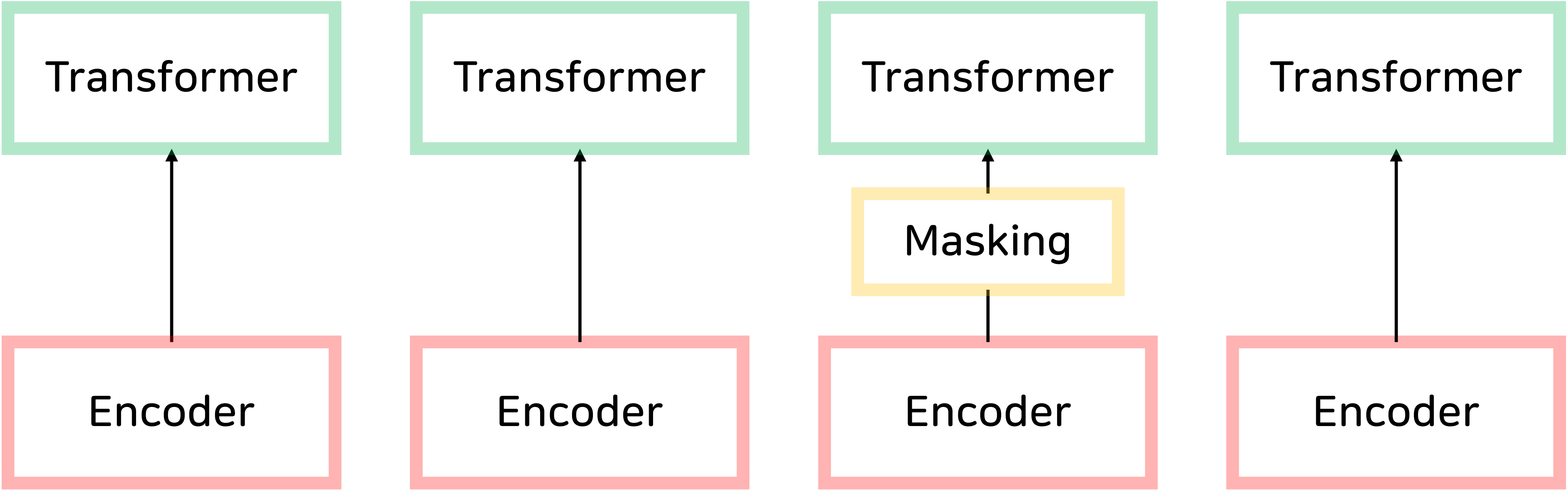


[그림03] G개의 codebook, V개의 code words를 활용한 Wav2Vec 2.0의 quantization 모듈

Self-supervised learning을 위함

Training

- Masking
  - 일정 비율 p만큼 trained feature vector로 masking



Training

- Objective

$$\mathcal{L} = \underset{\text{Contrastive Loss}}{\mathcal{L}_m} + \underset{\text{Diversity Loss}}{\alpha \mathcal{L}_d}$$

  - Contrastive Loss

Cosine Similarity

$$\mathcal{L}_m = -\log \frac{\exp(\boxed{sim(\mathbf{c}_t, \mathbf{q}_t)}/\kappa)}{\sum_{\tilde{\mathbf{q}} \sim \mathbf{Q}_t} \exp(sim(\mathbf{c}_t, \tilde{\mathbf{q}})/\kappa)}$$

Training

- Objective

$$\mathcal{L} = \underbrace{\mathcal{L}_m}_{\text{Contrastive Loss}} + \underbrace{\alpha \mathcal{L}_d}_{\text{Diversity Loss}}$$

- Diversity Loss
  - To increase Quantized codebook representation
  - V entries 와 각각의 G codebooks의 조합 확률이 균등하여 모든 code word를 균등하게 고려할 수 있도록 설계함

$$\mathcal{L}_d = \frac{1}{GV} \sum_{g=1}^{G} -H(\bar{p}_g) = \frac{1}{GV} \sum_{g=1}^{G} \sum_{v=1}^{V} \bar{p}_{g,v} \log \bar{p}_{g,v}$$

Training

- Fine-tuning
  - Minimizing a CTC loss
  - Modified version of SpecAugment

Encoder → Transformer → 무작위로 초기화된 linear projection

Experimental Setup

- Datasets
  - Librispeech corpus
    - without LS-960 or LV-60k
  - Fine-tuning
    - TIMIT dataset

Experimental Setup

- Pre-training

모델 구성	Transformer 블록 수	모델 차원	내부 차원 (FFN)	어텐션 헤드 수	크롭 크기 (샘플 수)	GPU 수	학습 기간 (일)	총 배치 크기 (시간)
BASE	12	768	3,072	8	250,000	64	1.6	1.6
LARGE	24	1,024	4,096	16	320,000	128	2.3 (Librispeech), 5.2 (LibriVox)	2.7

항목	BASE 모델	LARGE 모델
옵티마이저	Adam	Adam
학습률 초기화	8%의 업데이트 후 최대 5 x 10^-4	8%의 업데이트 후 최대 3 x 10^-4
학습 업데이트 수	400k	250k
LV-60k 학습 업데이트 수	600k	-
다양성 손실 가중치	0.1	0.1
양자화 모듈 G	2	2
양자화 모듈 V	320	320
코드워드 최대 이론값	102.4k	102.4k
엔트리 크기 (BASE)	128	384
Gumbel softmax 온도 (BASE)	2에서 최소 0.5로	2에서 최소 0.1로
대조적 손실 온도	0.1	0.1
L2 패널티	적용	적용
그래디언트 스케일 다운	10배	10배
레이어 정규화	사용 안 함	사용 안 함
대조적 손실 디스트랙터	100	100
훈련 체크포인트 선택	최소 Lm 값으로 선택	최소 Lm 값으로 선택

Experimental Setup

- Fine-tuning

Model	Labeled Datasets	Learning rate	Batch size (sample)	GPU	time
BASE	960시간	1e-4	3.2백만	8	1,600초
LARGE	960시간	1e-4	1.28백만	24	1,920초

Experimental Setup

- Language Models and Decoding

언어 모델 유형	블록 수	모델 차원	내부 차원	어텐션 헤드 수	빔 크기 (dev)	빔 크기 (test)
4-gram 모델	-	-	-	-	500	1500
Transformer 모델	20	1280	6144	16	50	500

Results

- Low-Resource Labeled Data Evaluation

Model	Unlabeled data	LM	dev clean	dev other	test clean	test other
10 min labeled						
Discrete BERT [4]	LS-960	4-gram	15.7	24.1	16.3	25.2
BASE	LS-960	4-gram	8.9	15.7	9.1	15.6
		Transf.	6.6	13.2	6.9	12.9
LARGE	LS-960	Transf.	6.6	10.6	6.8	10.8
	LV-60k	Transf.	4.6	7.9	4.8	8.2
1h labeled						
Discrete BERT [4]	LS-960	4-gram	8.5	16.4	9.0	17.6
BASE	LS-960	4-gram	5.0	10.8	5.5	11.3
		Transf.	3.8	9.0	4.0	9.3
LARGE	LS-960	Transf.	3.8	7.1	3.9	7.6
	LV-60k	Transf.	2.9	5.4	2.9	5.8
10h labeled						
Discrete BERT [4]	LS-960	4-gram	5.3	13.2	5.9	14.1
Iter. pseudo-labeling [58]	LS-960	4-gram+Transf.	23.51	25.48	24.37	26.02
	LV-60k	4-gram+Transf.	17.00	19.34	18.03	19.92
BASE	LS-960	4-gram	3.8	9.1	4.3	9.5
		Transf.	2.9	7.4	3.2	7.8
LARGE	LS-960	Transf.	2.9	5.7	3.2	6.1
	LV-60k	Transf.	2.4	4.8	2.6	4.9
100h labeled						
Hybrid DNN/HMM [34]	-	4-gram	5.0	19.5	5.8	18.6
TTS data augm. [30]	-	LSTM			4.3	13.5
Discrete BERT [4]	LS-960	4-gram	4.0	10.9	4.5	12.1
Iter. pseudo-labeling [58]	LS-860	4-gram+Transf.	4.98	7.97	5.59	8.95
	LV-60k	4-gram+Transf.	3.19	6.14	3.72	7.11
Noisy student [42]	LS-860	LSTM	3.9	8.8	4.2	8.6
BASE	LS-960	4-gram	2.7	7.9	3.4	8.0
		Transf.	2.2	6.3	2.6	6.3
LARGE	LS-960	Transf.	2.1	4.8	2.3	5.0
	LV-60k	Transf.	1.9	4.0	2.0	4.0

Results

- High-Resource Labeled Data Evaluation on Librispeech

Table 2: WER on Librispeech when using all 960 hours of labeled data (cf. Table 1).

Model	Unlabeled data	LM	dev		test	
			clean	other	clean	other
Supervised						
CTC Transf [51]	-	CLM+Transf.	2.20	4.94	2.47	5.45
S2S Transf. [51]	-	CLM+Transf.	2.10	4.79	2.33	5.17
Transf. Transducer [60]	-	Transf.	-	-	2.0	4.6
ContextNet [17]	-	LSTM	1.9	3.9	1.9	4.1
Conformer [15]	-	LSTM	2.1	4.3	1.9	3.9
Semi-supervised						
CTC Transf. + PL [51]	LV-60k	CLM+Transf.	2.10	4.79	2.33	4.54
S2S Transf. + PL [51]	LV-60k	CLM+Transf.	2.00	3.65	2.09	4.11
Iter. pseudo-labeling [58]	LV-60k	4-gram+Transf.	1.85	3.26	2.10	4.01
Noisy student [42]	LV-60k	LSTM	1.6	3.4	1.7	3.4
This work						
LARGE - from scratch	-	Transf.	1.7	4.3	2.1	4.6
BASE	LS-960	Transf.	1.8	4.7	2.1	4.8
LARGE	LS-960	Transf.	1.7	3.9	2.0	4.1
	LV-60k	Transf.	1.6	3.0	1.8	3.3

Results

- Phoneme Recognition on TIMIT
  - 음소(Phoneme, 音素)
    - 말의 뜻을 구별하는 최소의 언어단위

Table 3: TIMIT phoneme recognition accuracy in terms of phoneme error rate (PER).

	dev PER	test PER
CNN + TD-filterbanks [59]	15.6	18.0
PASE+ [47]	-	17.2
Li-GRU + fMLLR [46]	–	14.9
wav2vec [49]	12.9	14.7
vq-wav2vec [5]	9.6	11.6
This work (no LM)		
LARGE (LS-960)	7.4	8.3

Results

- Ablations

Table 4: Average WER and standard deviation on combined dev-clean/other of Librispeech for three training seeds. We ablate quantizing the context network input and the targets in the contrastive loss.

	avg. WER	std.
Continuous inputs, quantized targets (Baseline)	7.97	0.02
Quantized inputs, quantized targets	12.18	0.41
Quantized inputs, continuous targets	11.18	0.16
Continuous inputs, continuous targets	8.58	0.08

Link

- GitHub
- https://zerojsh00.github.io/posts/Wav2Vec2/