Time series : noise denoising service

지하철 음성 잡음 제거 서비스

ToBig's 16th Conference

Team : 데시벨_수호자_byGPT

2023.07.15 (Sat)

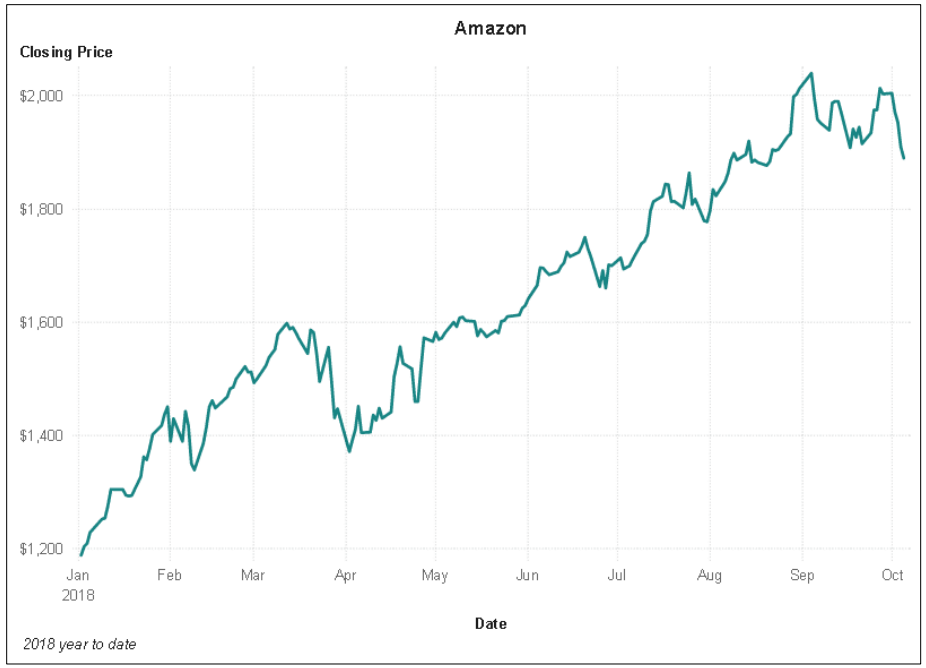
ToBig's 18th and 19th members

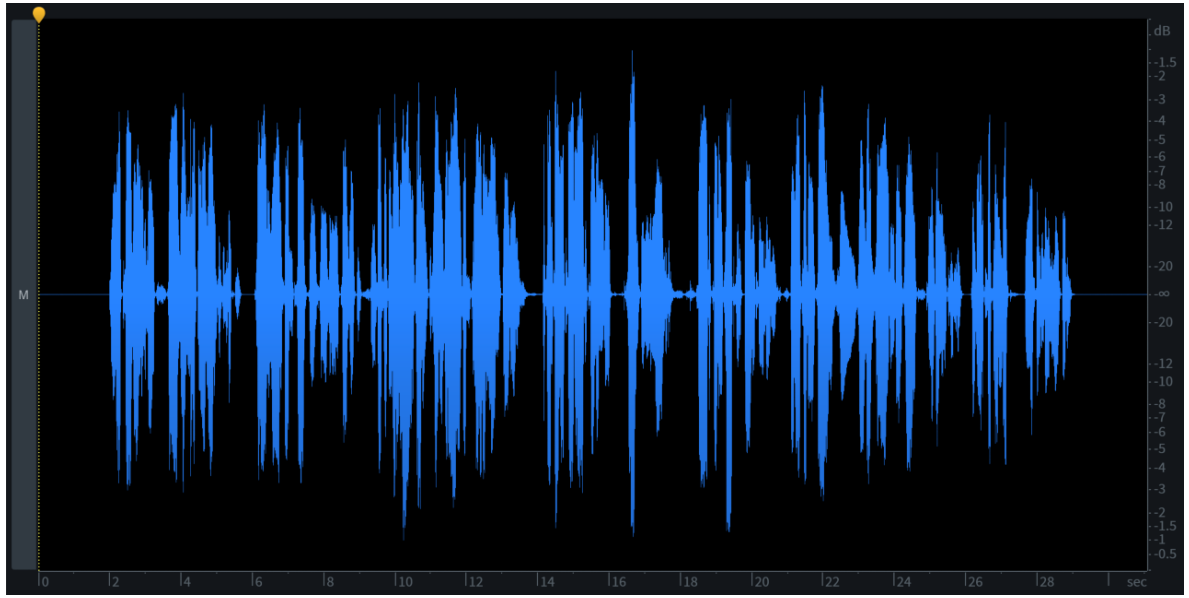
Contents

Part 1 Background

Background

시계열과 오디오?



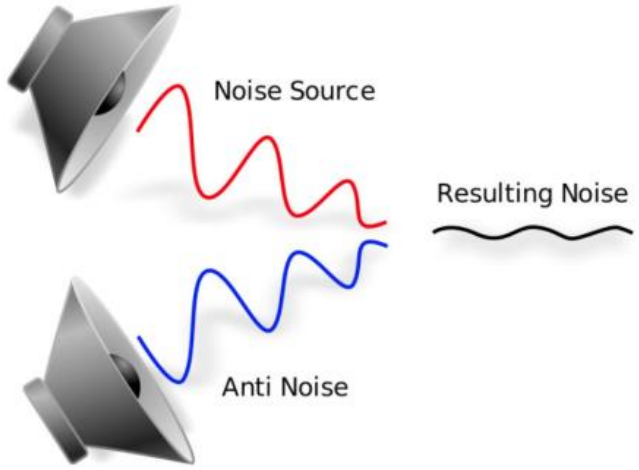


❖ 결론적으로, 오디오 데이터도 시계열에 포함

Part 2 Noise cancelling

1. 노이즈 캔슬링 정의

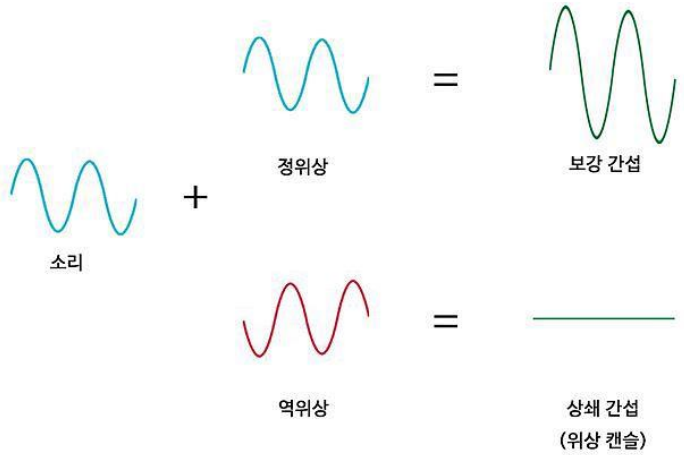
"음향기기를 통한 청취 행위시, 방해가 되는 외부소음을 차단 또는 상쇄하는 기술 "



2. 파동의 2가지 간섭효과

- 보강간섭 : 진폭과 주파수가 같은 파동이 만나 진폭이 더 커지는 현상

- 상쇄간섭 : 진폭과 주파수가 동일하지만 진행 방향이 반대인 파동을 만나 상쇄되는 효과



생활 속 노이즈 캔슬링

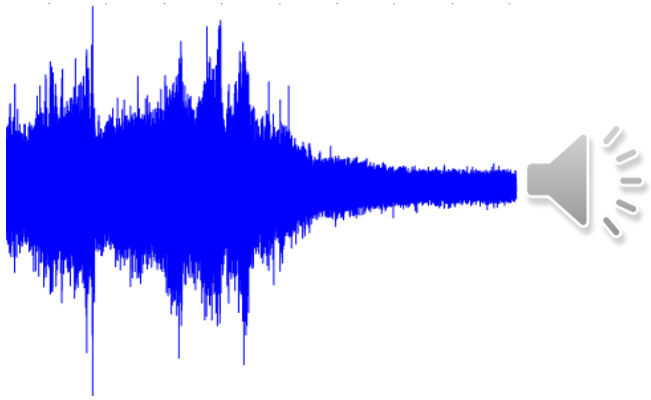

© I/O/E/SKB


외향 마이크가 외부 소리를 감지합니다.
AirPods Pro가 그에 상응하는
안티 노이즈를 발생시켜 외부 소리가 청력
기관에 닿기 전에 감쇠시킵니다.

Part 3 Methodology

학습 때 사용한 데이터셋 정리 및 기본적인 전처리 설명

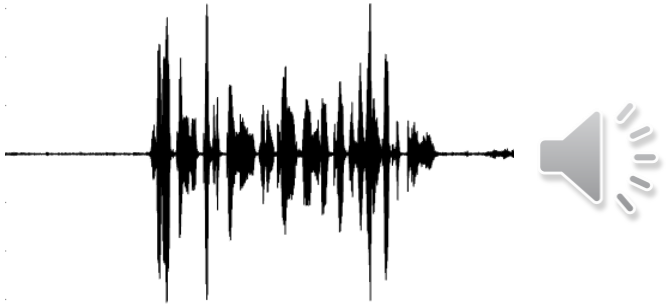
1)



URNBAN-SED dataset
뉴욕에서 들을 수 있는 30,000시간 가량의 데이터셋 중
기차 소음만 추출

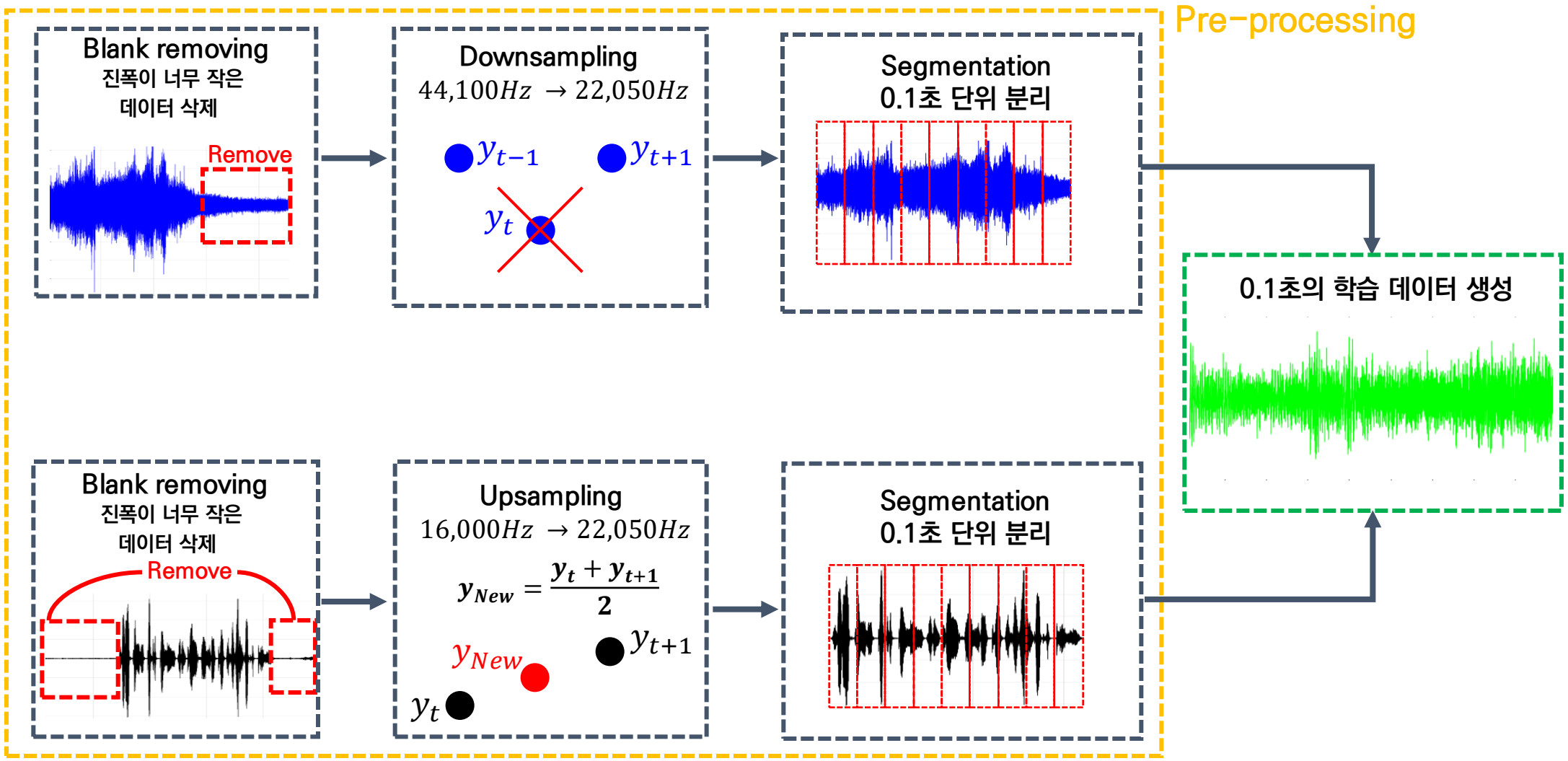
2)

AI Hub



한국어 자유대화 음성 (일반남여)
한국인 중 일반남녀가 대화하는 소리를 스튜디오에서 녹음한 데이터 약 4,000시간

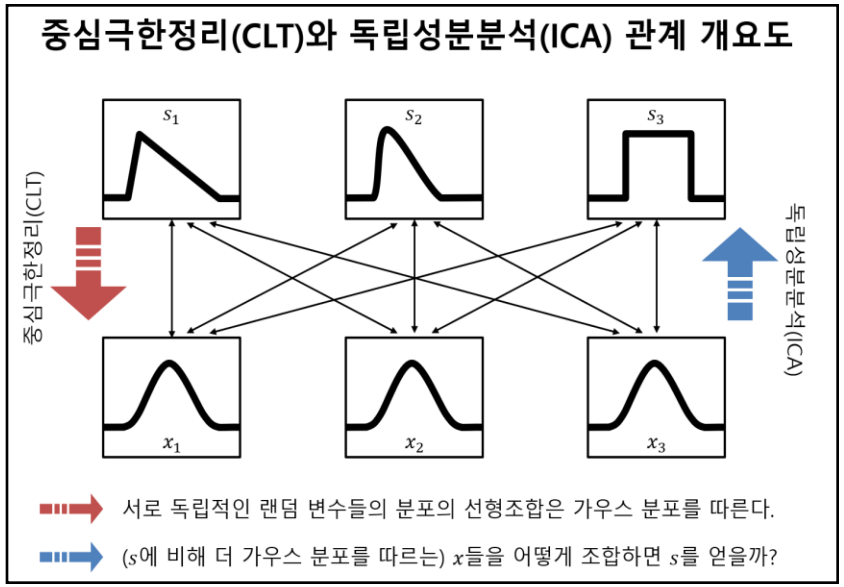
학습 때 사용한 데이터셋 정리 및 기본적인 전처리 설명



ICA (Independent Component Analysis)

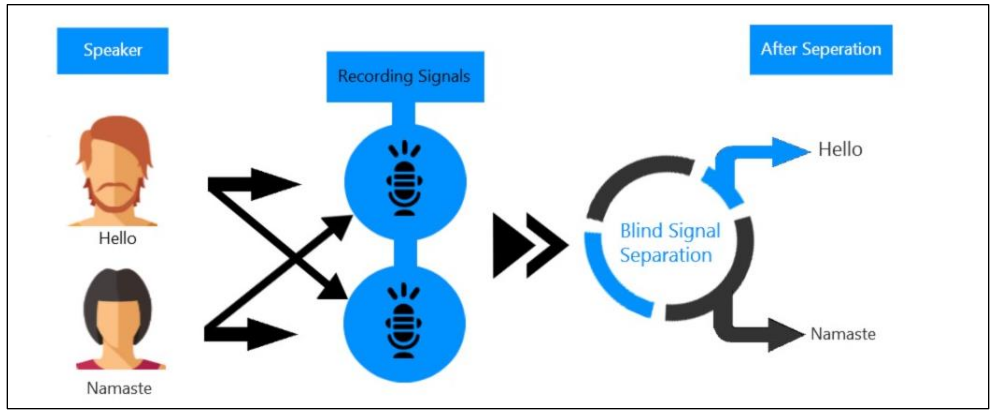
- PCA 와 비교 : PCA은 중요변수(각 변수들의 선형적 결합) 을 파악하는 것이고, ICA는 multisignal data 에서 여러 개의 독립적인 신호들로 분리하는 것.

-〉 ICA 선택 이유 : 소음과 음성은 독립일테니 분류할 수 있을 것이다. (중요변수와는 무관)



*** 독립성 체크**

$$p_{f_1,f_2}(f_1,f_2) = p_{f_1}(f1) \times p_{f_2}(f_2)$$

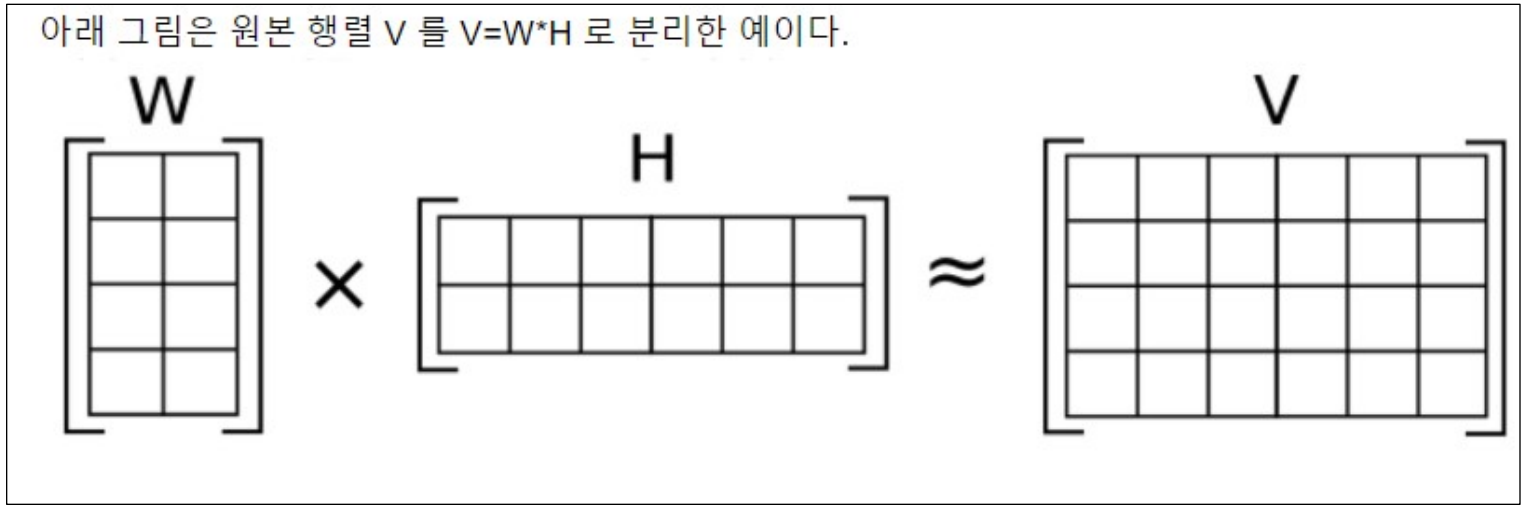


❖ **한계 :** ICA는 Noisy한 데이터만으로 노이즈와 음성신호를 분리하지 못하는 한계를 가짐.

NMF (Non-negative matrix factorization)

- NMF :차원축소 기법으로, PCA나 t-SNE와 다르게 직접적인 해석이 가능.
- PCA / t-SNE : 원본 데이터의 특징을 추출하여 새로운 특징 셋으로 표현 가능.
  BUT ! 새로운 특징셋이 원본 특징과 어떤 연관 관계를 가지는지는 해석이 불가능.

-> NMF 선택이유 : 새로운 특징셋이 어떻게 원본 데이터 들과 관계를 가지는지 확인이 가능.

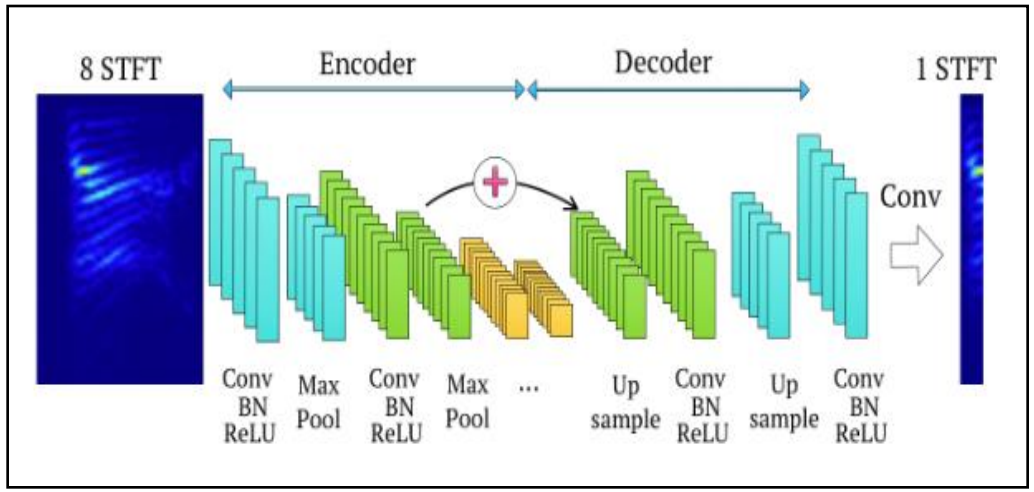


❖ **한계 :** 특징추출을 위한 소요시간이 기존의 특징 파라미터인 mfcc 보다 큼.

채택!

CED

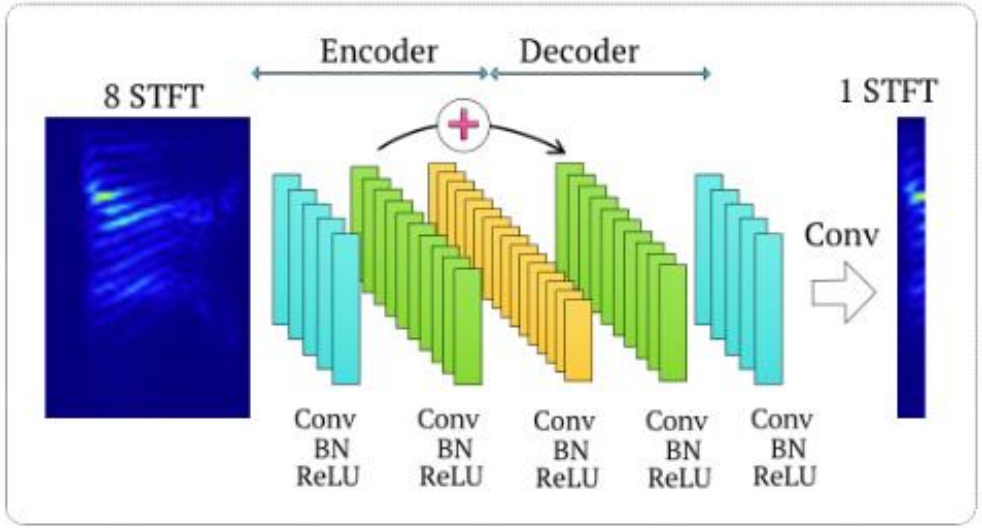
Convolutional Encoder Decoder network



- CED : 대칭 인코더딩 레이어 & 디코딩 레이어
- Encoder : convolution, batch-norm, max-pooling, ReLu
- activation 반복 -> 인코더를 따라 feature representation 학습
- decoder를 따라 representation -> reconstruction

R-CED

Redundant - Convolutional Encoder Decoder network

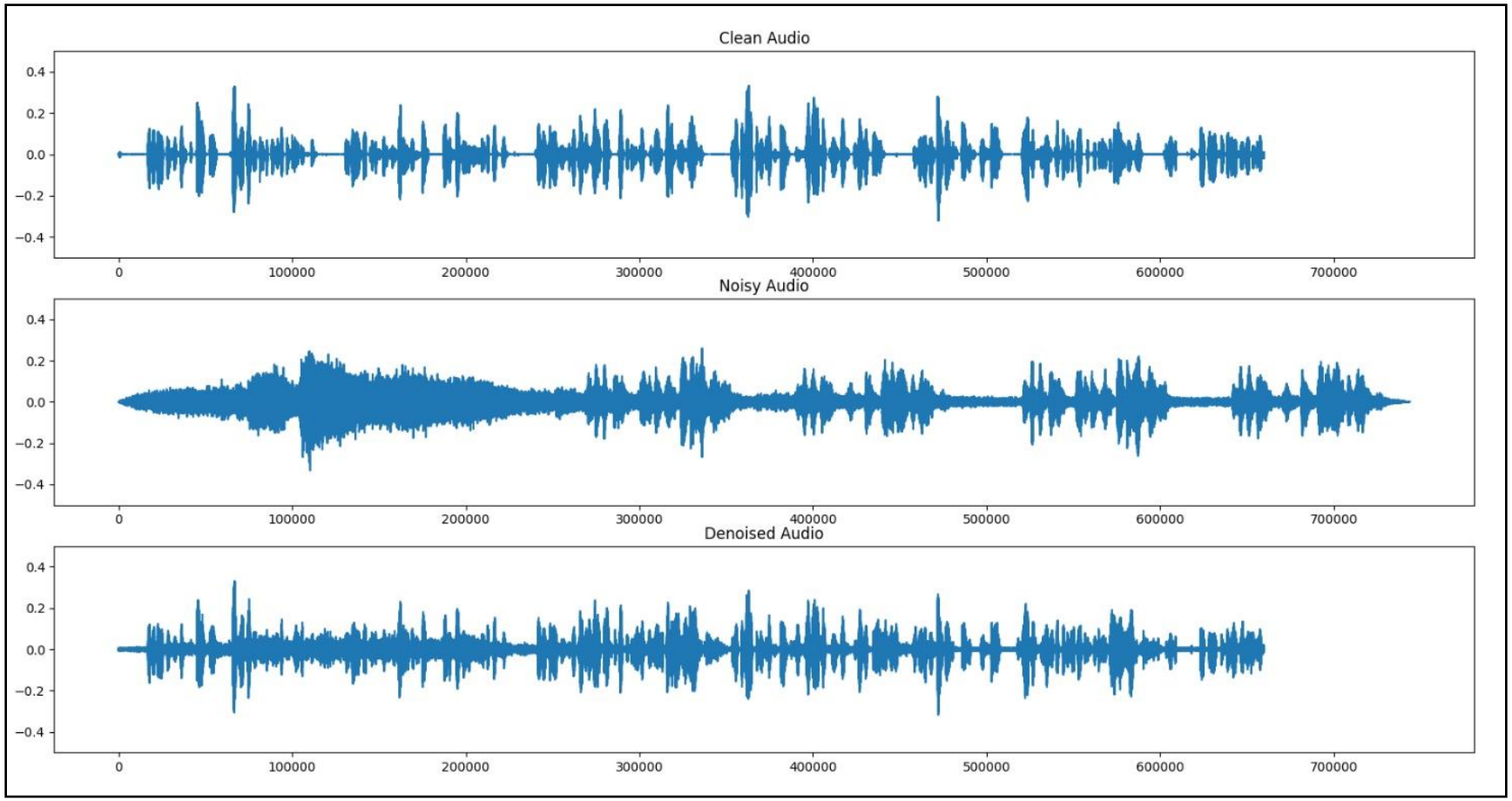


- Decoder 를 따라 representation -> reconstruction
- R-CED 풀링 레이어가 없음 -> 업샘플링 레이어도 필요 없음.
- Ced 와 반대로 인코더를 따라 피처를 더 높은 차원으로 인코딩 하고 디코더를 따라 압축하는 형태

R-CED (Redundant - Convolutional Encoder Decoder network)

	Layer Configuration	Number of Filters	Filter Width
CED (11 Conv)	Encoder: (Conv, BN, ReLU, Pool) × 5 Decoder: (Conv, BN, ReLU, Upsample) × 5	12-16-20-24-32-24-20-16-12-8-1	13-11-9-7-5-7-9-11-13-8-129
R-CED (10 Conv)	(Conv, ReLU, BN) × 9, Conv.	12-16-20-24-32-24-20-16-12-1	13-11-9-7-7-7-9-11-13-129
R-CED (16 Conv)	(Conv, ReLU, BN) × 15, Conv.	10-12-14-15-19-21-23-25-23-21-19-15-14-12-10-1	11-7-5-5-5-5-7-11-7-5-5-5-5-7-11-129

R-CED (Redundant - Convolutional Encoder Decoder network)



Part 4 Our contributions

자세한 개념

1. 푸리에 변환

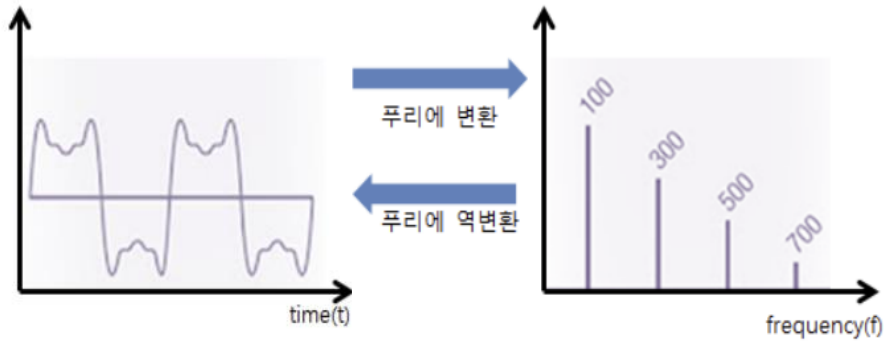
- 음성 신호에 푸리에 변환을 적용하면 각 진동수 성분이 그 음성에 얼마나 들어있는지 알 수 있음

➢ 쉽게 설명하면 음성 신호에 저음이 얼마나 있고, 고음이 얼마나 있는지를 정량적으로 구할 수 있음

2. STFT(Short Time Fourier Transform)

- 음성을 작게(0.01초 수준) 잘라서 각 작은 조각에 푸리에 변환을 적용

➢ 이것을 STFT라고 부르고 일반적으로 이 결과의 L2 norm을 Spectrogram 이라고 부름



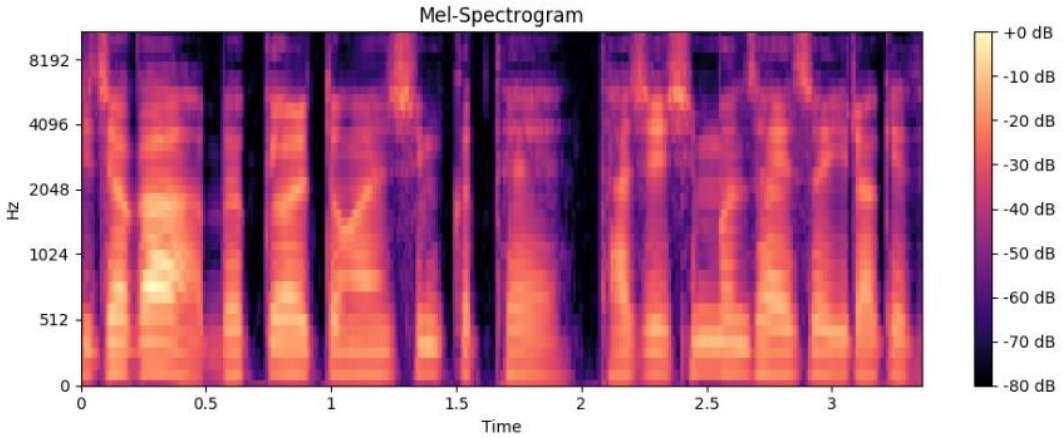
3. Melspectrogram

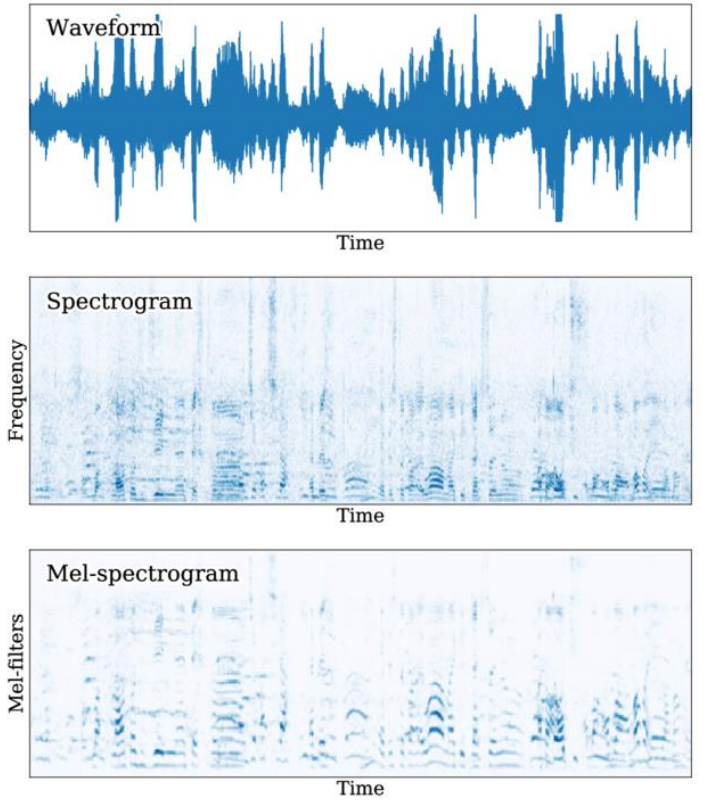
- Melspectrogram은 Spectrogram에 mel- filter라는 필터를 적용해서 얻어짐.
- 이는 사람의 청각 기관이 저음에서 주파 수 변화에 민감하고 고음에서는 주파수의 변화에 덜 민감한 특징을 반영함
- 딥러닝과 사람의 청각 반응은 관련 없어 보일 수 있으나 음성 처리나 자연에 처리 분야에서도 Melspectrogram은 널리 사용되고 있으며 좋은 성능을 보여줌.
- 또한, Melspectrogram은 spectrogram보다 크기가 작아서 학습 속도 등에서 유리함

Mel spectrogram

Mel Spectrogram

: 소리의 파형을 인간이 들을 수 있는 범위로 줄인 Mel scale로 다운 스케일한 이후 그 파형을 그림으로 그린 모양





음성데이터 분석방법 (MEL-Spectrogram 으로 추출)

❖ Mel – scale

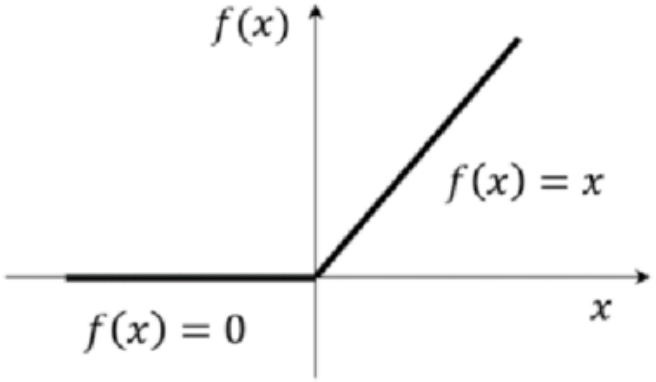
- 달팽이관은 주파수가 낮은 대역에서는 변화하는 주파수를 잘 감지하는데, 주파수가 높은 대역에서는 주파수 감지를 잘 하지 못함.
- 이를 고려하여 scaling해줄 수 있는데, 이때 이 기준을 Mel-Scale 이라고 함.

❖ Spectrogram

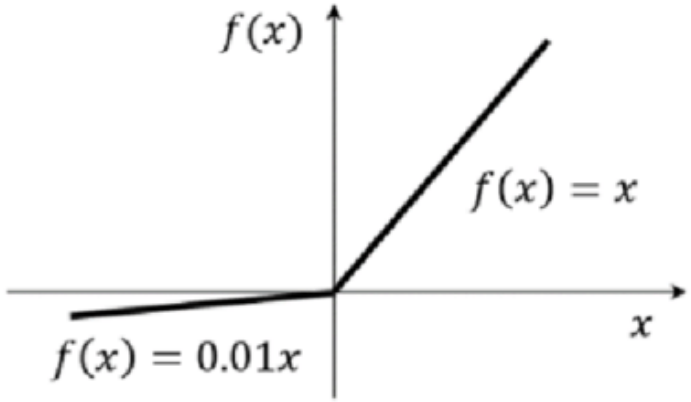
- 소리나 파형을 시각화한 도구
- 일반적으로, 가로축이 Time, 세로축이 Frequency, 색깔이 amplitude의 크기를 의미하며 colorbar 형태로 안내되어 있음.
- Mel- Spetrogram은 이 중 주파수를 mel-scale로 변환한 형태.

활성화 함수 변경

- 기존 보편적인 Relu ⟹ Leaky-Relu
- Y ? contrastive learning 진행하는 r-CED 특성상, 상대 분포에 대한 정보를 아예 없애는 것보다 조금 남기는 것이 유리

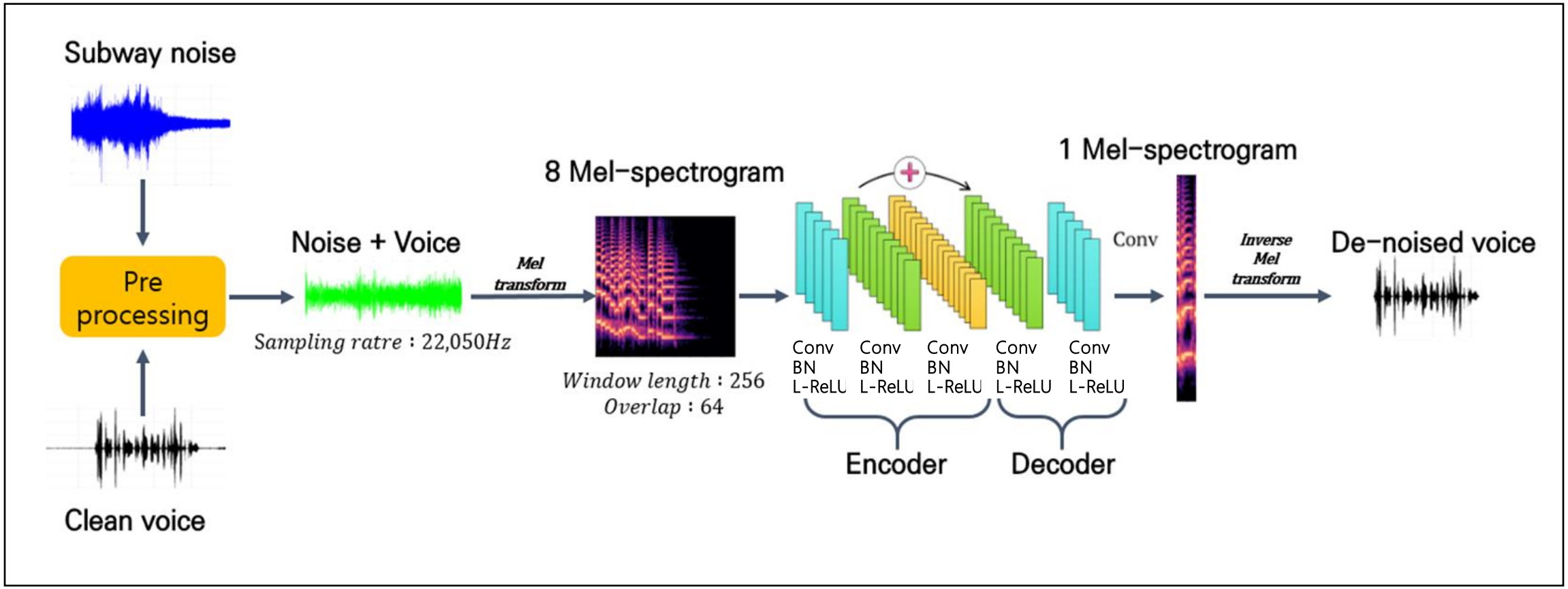


Relu activation function



Leaky-Relu activation function

우리가 최종으로 선택한 모델, 구조 및 설명



Part 5 Results

음성 품질 평가 지표 : PESQ, STOI, SNR

음성 품질 평가에는 두 가지가 존재 - 1) 주관적 방법 2) 객관적 방법

* 주관적 음질 평가는 시간과 비용이 소모된다는 한계가 존재하기 때문에 주관적 음성 품질 평가 방법과 비슷한 결과 값을 내는 객관적 음성 품질 평가 방법 사용.

1.PESQ
(Perceptual Evaluation of Speech Quality)

- 음질을 나타내는 척도

1. 입력 음성신호와 시스템을 통과한 출력 음성 신호를 정렬하고 필터링하여 수화기의 특성을 고려
2. 최대 4.5로, 높을수록 시스템 처리 후 오디오가 처리 전 오디와 일치하다는 것

2.STOI
(Short-Time Objective Intelligibility)

- 음성의 명료도를 나타내는 척도

1. 각 프레임의 전력을 계산하여 일정전력보다 낮은 프레임들을 제거
2. 추출된 주파수대역의 성분을 사용하여 각 프레임의 옥타브대역 correlation의 평균을 계산하여 0에서 1 사이의 STOI 값을 계산

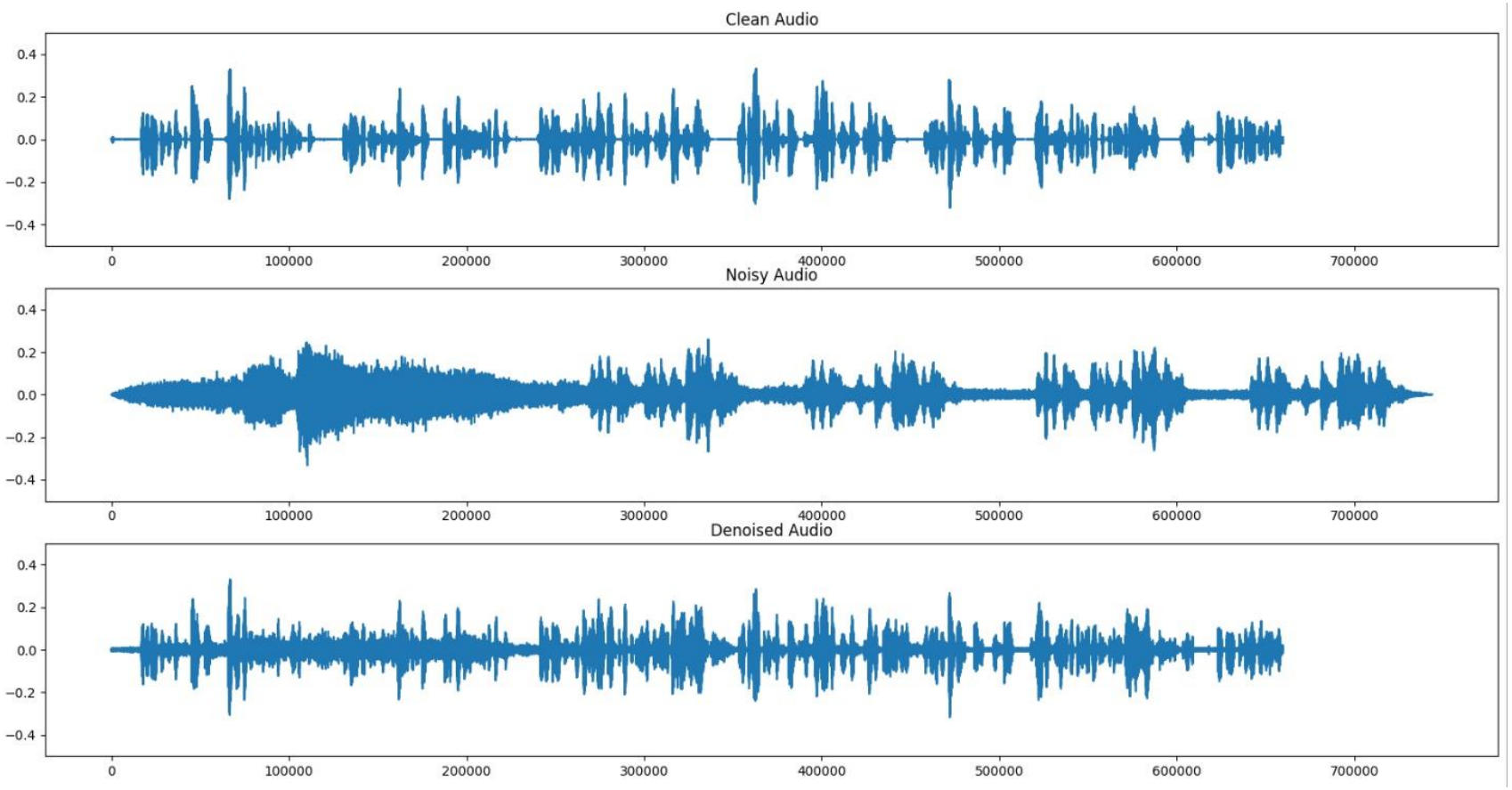
3.SNR
(Signal-to-Noise Ratio)

- 음성과 잡음의 정도

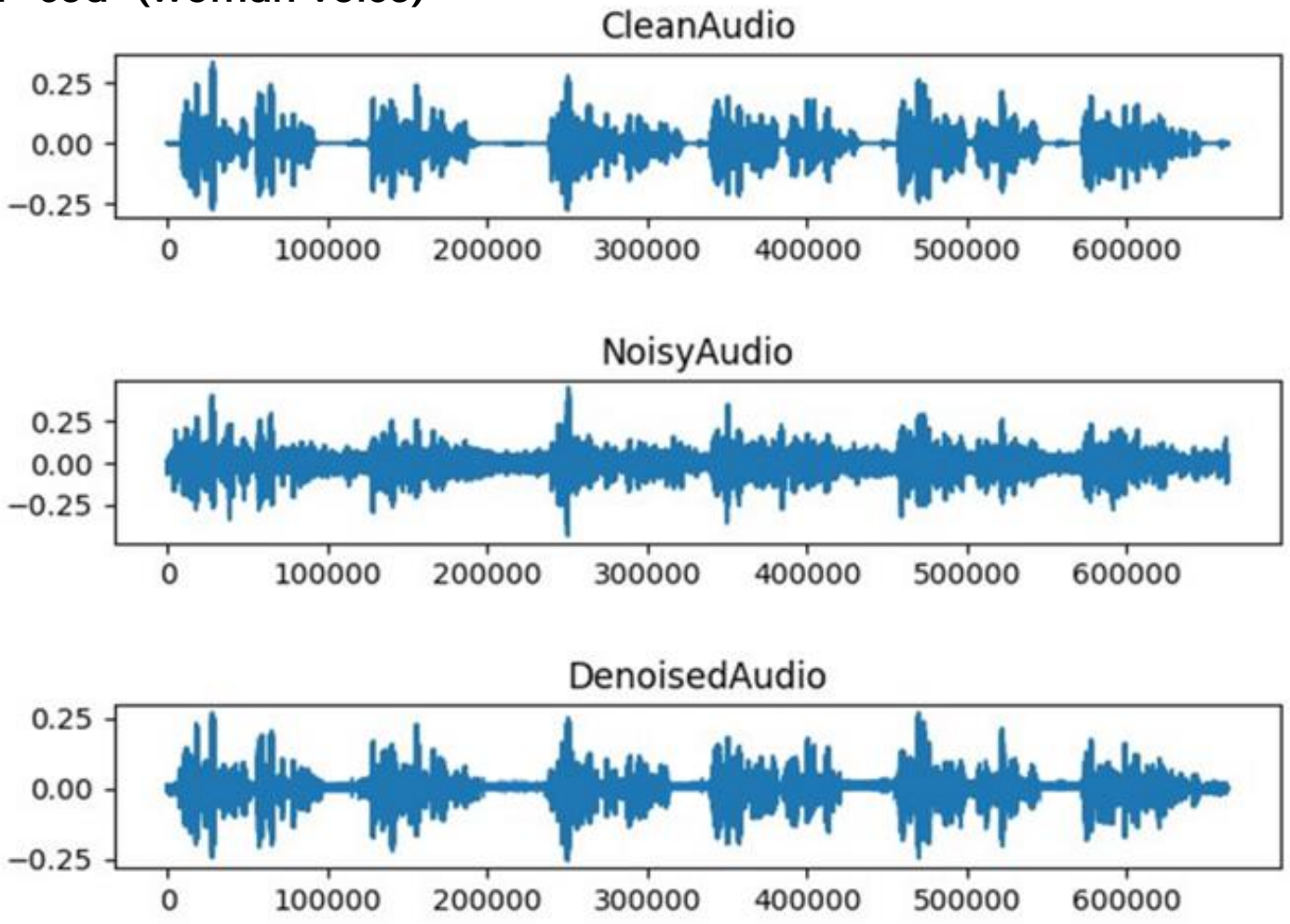
1. Decibel을 통해 음성과 잡음의 정도를 나타낸 수치
2. SNR 값이 높을수록, noise power 값이 낮을수록 더 구분된 소리임을 표현

* 비가청음 : 인간의 청각 범위를 벗어난 주파수 영역에 위치하는 음

기존 r-ced (woman voice)



New r-ced (woman voice)



PESQ and STOI scores

	PESQ (original with denoised)		STOI (original with denoised)	
	R-CED	New R-CED	R-CED	New R-CED
Woman (online data)	1.82	**1.90**	0.78	**0.84**
Man (KSW recording)	**2.46**	2.45	0.91	**0.92**

```
[35] print(pesq_func(woman_voice,woman_voice,sr))
     print(pesq_func(woman_voice,woman_union,sr))
     print(pesq_func(woman_voice,woman_denoised,sr))
     print(stoi_func(woman_voice,woman_voice,sr))
     print(stoi_func(woman_voice,woman_union,sr))
     print(stoi_func(woman_voice,woman_denoised,sr))
```

```
4.5
2.1864326000213623
1.90566086769104
1.0
0.8290407722152972
0.845404309308398
```

```
[36] print(pesq_func(man_voice,man_voice,sr))
     print(pesq_func(man_voice,man_union,sr))
     print(pesq_func(man_voice,man_denoised,sr))
     print(stoi_func(man_voice,man_voice,sr))
     print(stoi_func(man_voice,man_union,sr))
     print(stoi_func(man_voice,man_denoised,sr))
```

```
4.5
2.221285820007324
2.453169584274292
0.9999999999999994
0.7695079844902788
0.9271441729537642
```

SNR scores

	Original audio		Denoised audio	
	SNR	Noise power	SNR	Noise power
Woman (online data)	−23.469	−4.507	−15.122	−11.443
Man (KSW recording)	−23.699	−7.237	−22.533	−7.025

```
# print('SNR by definition, not computation: {} dB'.format(defSNR))
print('measured SNR: {} dB'.format(r))
print('Noise Power: {} dB'.format(10*np.log10(noisePower)))

 measured SNR: -23.469273343189553 dB
 Noise Power: -4.507364348480401 dB

print('measured SNR: {} dB'.format(r))
print('Noise Power: {} dB'.format(10*np.log10(noisePower)))

 measured SNR: -15.122498809546194 dB
 Noise Power: -11.443120314814559 dB
```

```
# print('SNR by definition, not computation: {} dB'.format(defSNR))
print('measured SNR: {} dB'.format(r))
print('Noise Power: {} dB'.format(10*np.log10(noisePower)))

 measured SNR: -23.699076207711123 dB
 Noise Power: -7.287318187516952 dB

print('measured SNR: {} dB'.format(r))
print('Noise Power: {} dB'.format(10*np.log10(noisePower)))

 measured SNR: -22.533015639588715 dB
 Noise Power: -7.025428928326905 dB
```

Denoising 구현
Woman voice

지하철 소음 제거 전

지하철 소음 제거 후

- 소음 제거 전에는 지하철 소리와 주변 사람들의 소리가 크게 들리는 데 비해
  소음 제거 후에는 지하철 소리가 거의 들리지 않는 것을 확인

Denoising 구현
Man voice

지하철 소음 제거 전

지하철 소음 제거 후

- 직접 녹음 한 경우 또한 소음 제거 후 지하철 소리는 줄어들었지만 말하는 음성에도 손실이 존재

Part 6 Conclusion

Results conclusion

✓ **Pinpointed R-ced model to audio denoising**

- Melspectrogram으로 변경 후 인간 소리에 더 맞게 변형
- Pretrain을 한국 지하철 소리도 추가
- 활성화 함수를 contrastive learning에 좀 더 어울리는 것으로 변형

✓ **Results**

- 우리의 Denoise audio 가 r-ced에 비해 정량적으로 더 나은 수치를 보임
- 한국 지하철에 좀 더 맞게 개조

Future research

- 소리 분류의 정확도는 주변 상황에 따라 달라질 수 있기에 노이즈 관련 연구가 추가로 진행이 필요.
- 소리를 탐지하는 마이크의 성능도 중요하게 작용.
- 감지 해야 할 소리가 동시다발적으로 발생하면, 모든 소리를 인식하기 힘들다는 한계가 존재. -〉 동시다발적으로 발생하는 소리에 대한 처리를 위한 모델 성능 개선이 필요.
- 소리의 종류를 늘려서 제조, 공정, 디지털 헬스케어, 금융, 게임 등 다양한 산업에서 사용 가능하도록 연구 필요.
- 실시간으로 가능하게 구현하는 서비스 개발.

활용 방안

✓ Denoising 활용

- 오디오 및 비디오에서 배경 잡음 소리 제거

➢ ex) 마이크를 사용하여 길거리 인터뷰를 녹화할 때, 결과 영상에 숨소리, 누군가의 목소리, 교통 소음 및 기타 주변 소리나 마이크 결함으로 인한 소음 포함되는 경우 해결 가능

- 줌/ 화상 회의에서 주변 실시간 잡음 제거 가능

✓ Denoising 에서 나아가 음성분류

- 청각 장애인을 위한 소리 분류 시스템

➢ 위험감지 어려움 :차 경적소리, 사이렌 소리 등 위험 감지 요소가 될 수 있는 소리를 파악하지 못하여 위험 노출 가능성이 높다.

➢ 여러 소리를 분류하여 위험한 소리나 알아야할 소리의 특정 데시벨 이상 -〉 진동으로 인식

Appendix

- 소스코드

링크 : https://github.com/DONG-JOON-LEE/Tobigs_Conference_2023-1

- R-CED (논문 제목)

소스코드 : https://github.com/SreemukhMantripragada/SpeechEnhancement/blob/main/Using%20the%20model.ipynb
논문원본 : https://arxiv.org/abs/1609.07132

데시벨_수호자_byGPT

Q & A

18기 김정우

19기 하주찬

18기 남주연

18기 김성우

19기 이동준

18기 이다인