S2ST : Speech to Speech Translation

[Paper] TRANSPEECH: SPEECH-TO-SPEECH TRANSLATION WITH BILATERAL PERTURBATION

Contents

Instruction

Instruction

- **Speech - to - Speech 의 발전 (1) :**

1. **Cacaded 3 model (ASR + MT + TTS)**

   ① 음성인식 ASR(Auto Speech Recognition)을 통해 Source Speech로부터 Source 텍스트를 추출한다.

   ② 번역 모델 MT(Machine Translation)을 거쳐 Target 텍스트로 변경

   ③ 음성합성 TTS(Text to Speech) 로 음성을 합성

특징)

- 각각의 모듈을 개별적으로 나누어 학습, 수행하므로 실행하기 비교적 쉽다.
- 중간 결과물을 확인할 수 있다.

음성 : " **It is hot** "

ASR

텍스트 : **It is hot**

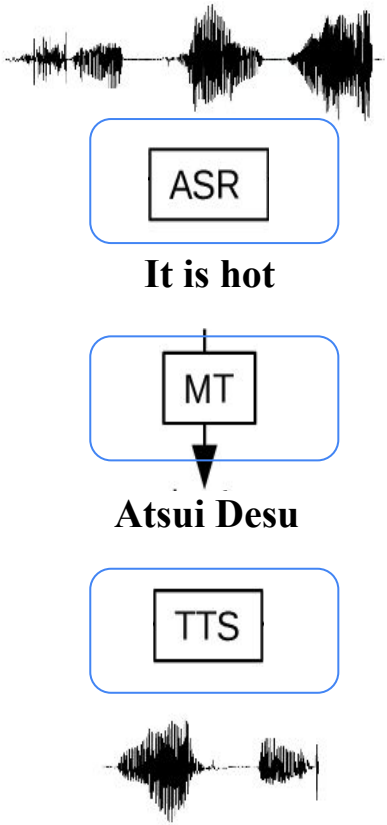
MT

텍스트 : **Atsui Desu**

TTS

음성 : "**Atsui Desu**"

Instruction



- **Speech - to - Speech 의 발전 (1) :**

1. **Cacaded 3 model (ASR + MT + TTS)**

① ASR(음성인식)

CPC, LAS, wav2vec, Pushing ASR , HuBERT , …

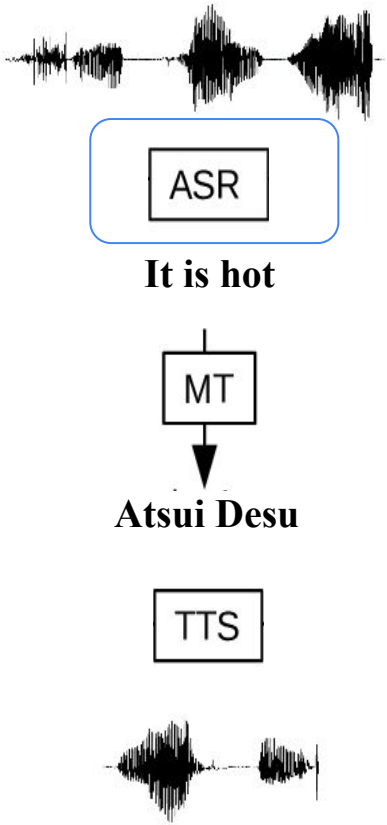
② MT(번역 모델)

Seq2Seq, Transformer…

③ TTS(음성합성)

Tacotron 1, Tacotron 2, …

Instruction



- **Speech - to - Speech 의 발전 (1) :**

1. **Cacaded 3 model (ASR + MT + TTS)**

① ASR(음성인식)
- speech와 text 데이터셋은 align해서 사용한다.
- ASR은 크게 2가지 과정으로 분류할 수 있다.
(1. Speech Feature Extraction + 2. Acoustic Model)
ex. MFCC, mel-spectrogram

[Self-Supervised Learning 방법론]
: Labeled 데이터 뿐만 아니라 Unlabeled 데이터를 함께 활용하여 모델의 성능을 향상=>좋은 representation을 추출하는 것

음성데이터만으로 음성의 특징을 잘 추출할 수 있는 모델 개발하고 ASR 개발 과정에서 음성의 특징벡터를 추출하는 과정을 대체

Instruction

ASR

It is hot

MT

Atsui Desu

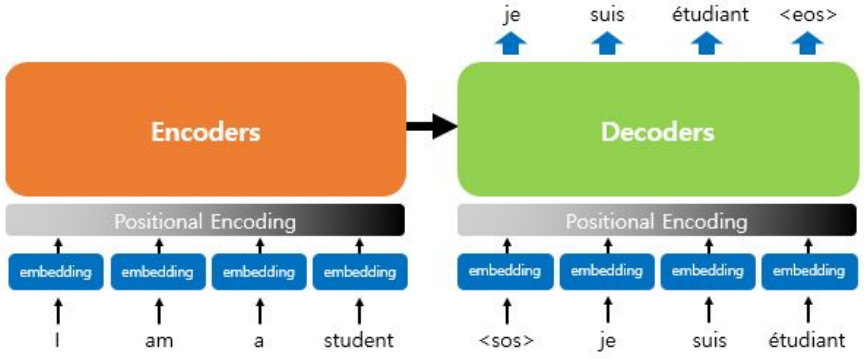
TTS

- **speech - to - speech 의 발전 (1) :**

1. **Cacaded 3 model (ASR + MT + TTS)**

② MT(번역 모델) ex. Transformer

- Encoders - Decoders 두 개의 모듈 구조로 구성.
- 각 단어의 위치정보를 더해서(Positional Encoding) 모델의 입력으로 사용 (Seq-to-Seq 모델과의 차이점)
- attention 구조 사용됨



<Transformer>

Instruction

ASR

It is hot

MT

Atsui Desu

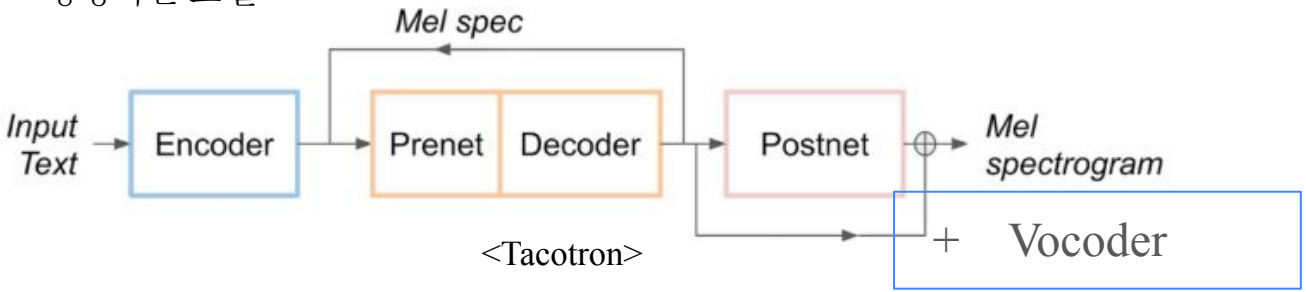
TTS

- **speech - to - speech 의 발전 (1) :**

1. **Cacaded 3 model (ASR + MT + TTS)**

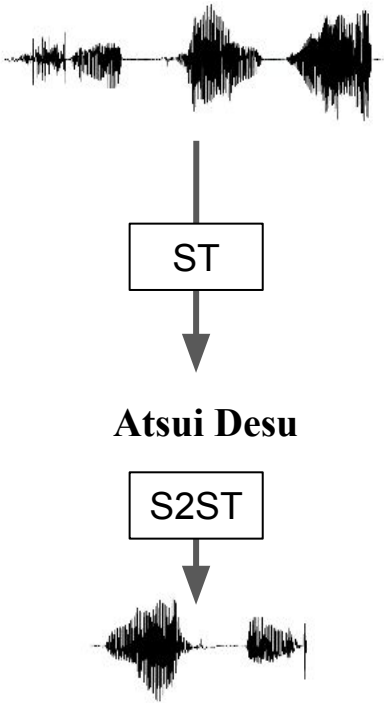
③ TTS(음성합성) ex. Tacotron

- Acoustic Model : 입력으로 텍스트(character)또는 음소(phoneme)을 받아 acoustic feature(mel spectrogram)으로 반환
- Vocoder : 입력으로 mel-spectrogram(및 유사한 스펙트로그램)을 받아서 실제 오디오를 생성하는 모델.
- Fully End-to-End TTS Model : 입력으로 텍스트 또는 음소를 받아 바로 오디오를 생성하는 모델



<Tacotron>

Instruction



- **speech - to - speech 의 발전 (2) :**

  Cacaded 2 model (S2T(ASR + MT) + TTS)

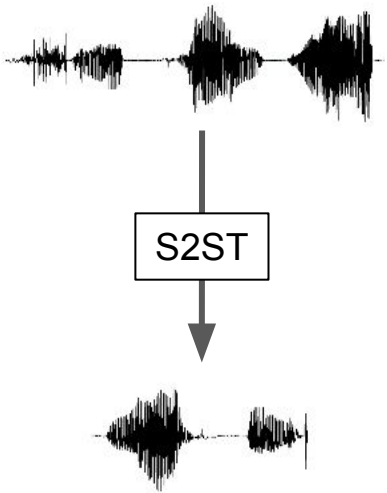
  ① end-to-end S2T(음성번역) : 음성번역(Speech Translation)을 적용하여 Target txt 를 추출

  ② TTS(음성합성) : Target txt에서 Target 음성 생성

  특징)

  - ST 하나의 모듈로 두 개의 Task를 동시에 수행하고 학습
  - 음성, 텍스트 번역 데이터셋이 많지 않음
  - ASR과 MT사이의 error propagation issue를 개선

Instruction



- **speech - to - speech 의 발전 (3) :**

  1 model (End-to-End model)

  ① S2ST : 음성을 직접 음성으로 번역하는 모델.

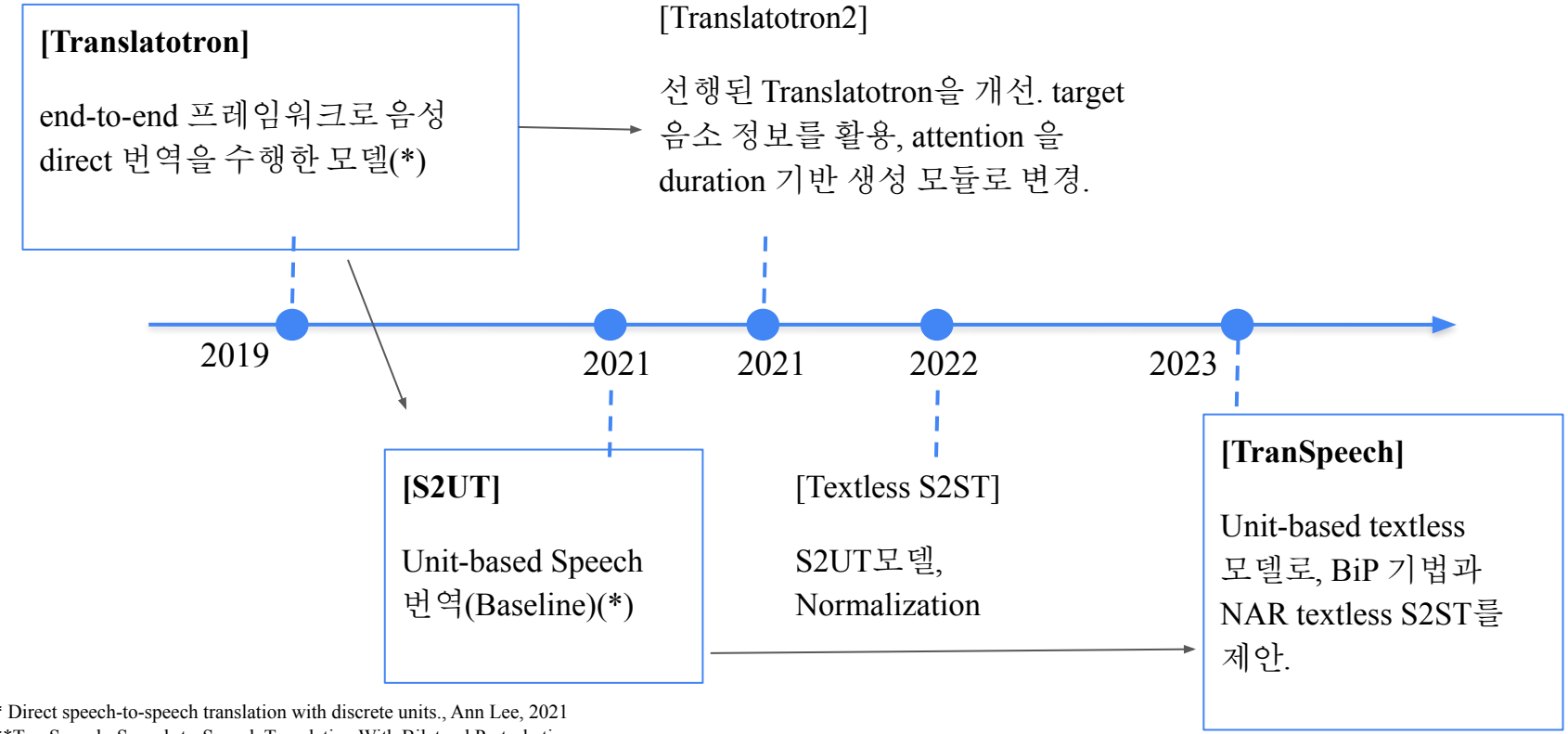
  => End-to-End 최초의 모델 : **“Translatotron”**

  특징)

  - 하나의 모델을 훈련하고 결과를 생성
  - Source Speech, Target Speech 데이터셋으로 훈련
  - 단어의 음성 특징을 보존할 수 있다.

Background

Background : History of Direct S2ST



* Direct speech-to-speech translation with discrete units., Ann Lee, 2021
**TranSpeech: Speech-to-Speech Translation With Bilateral Perturbation

Translatotron 1

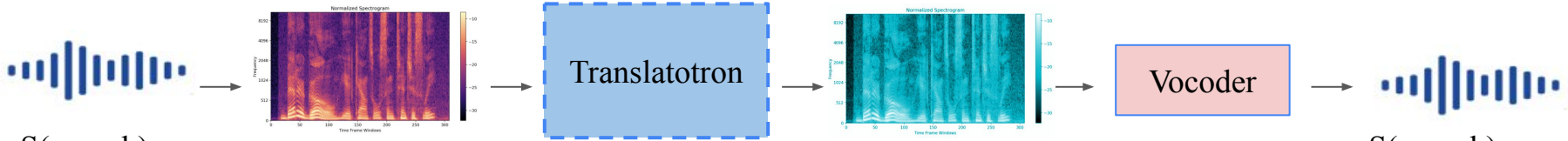
레퍼런스 모델_Translatotron

[Translatotron의 구조]

- Step 1) 음성 특징 추출
  신호처리 단계 : Input Speech => ((feature extraction)) =>log-mel spectrogram 벡터를 추출
  primary task : target speech에서 -> target spectrogram을 생성

- Step 2) 번역 수행
  spectrogram => ((Translatotron)) => Target spectrogram 생성

- Step 3) 음성합성
  Target spectrogram => ((Vocoder)) => Wave Form



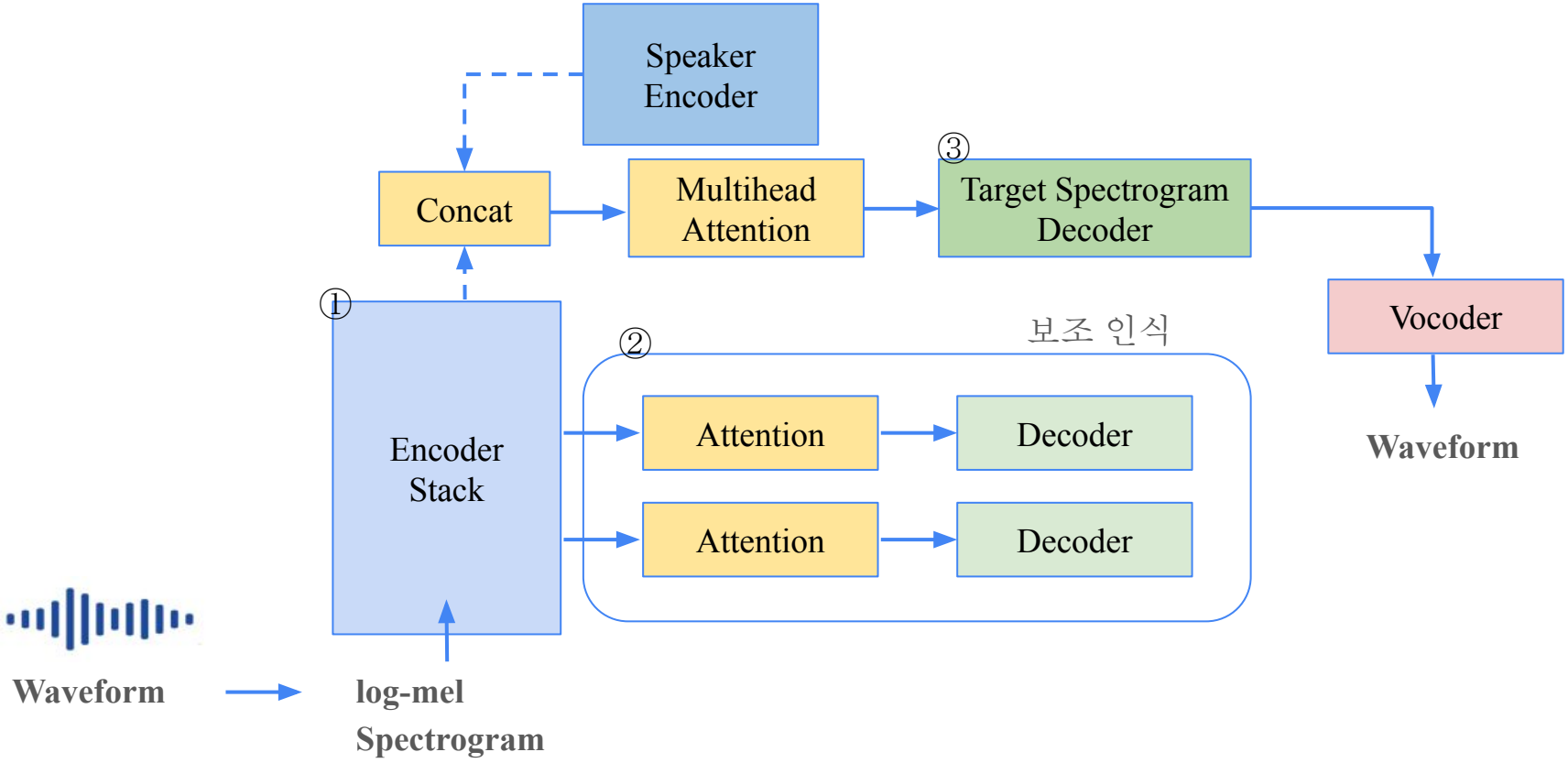
S(speech)

log-Mel Spectrogram

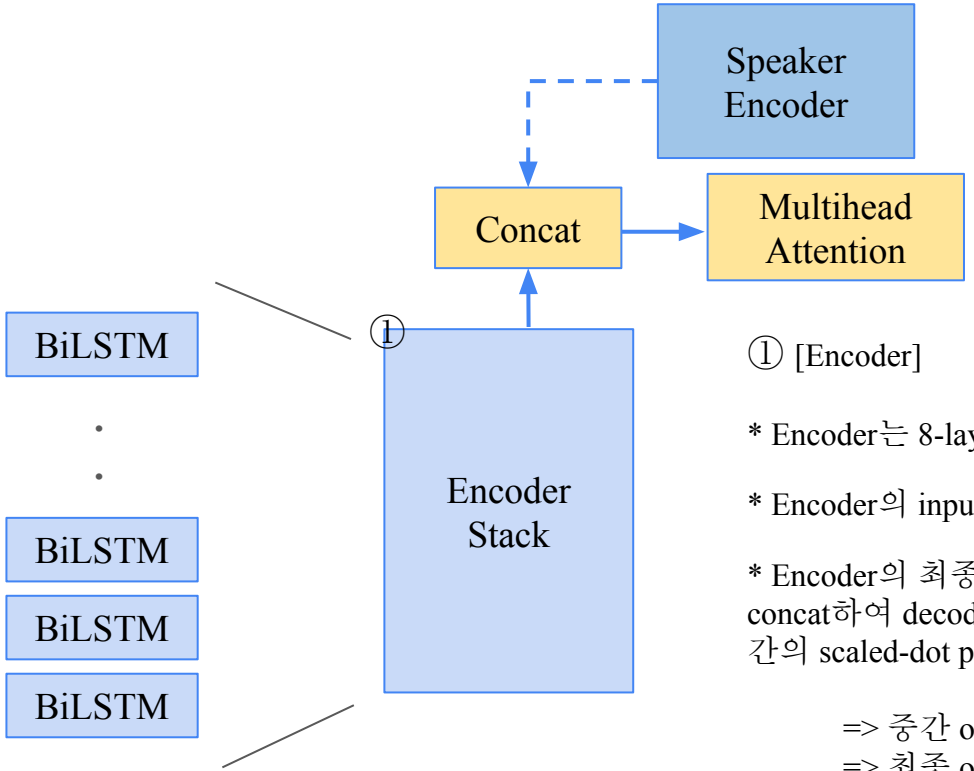
Target-Spectrogram

S(speech)

레퍼런스 모델_Translatotron Architecture



레퍼런스 모델_Translatotron



[Speaker Encoder]

* 화자의 정보를 추출, Target 화자의 MFCC를 입력으로 받아 화자의 특징 정보를 추출하여 Condition으로 제공 => speaker reference utterance
* LSTM으로 구성
* 사전학습된 Speaker Encoder를 사용

① [Encoder]

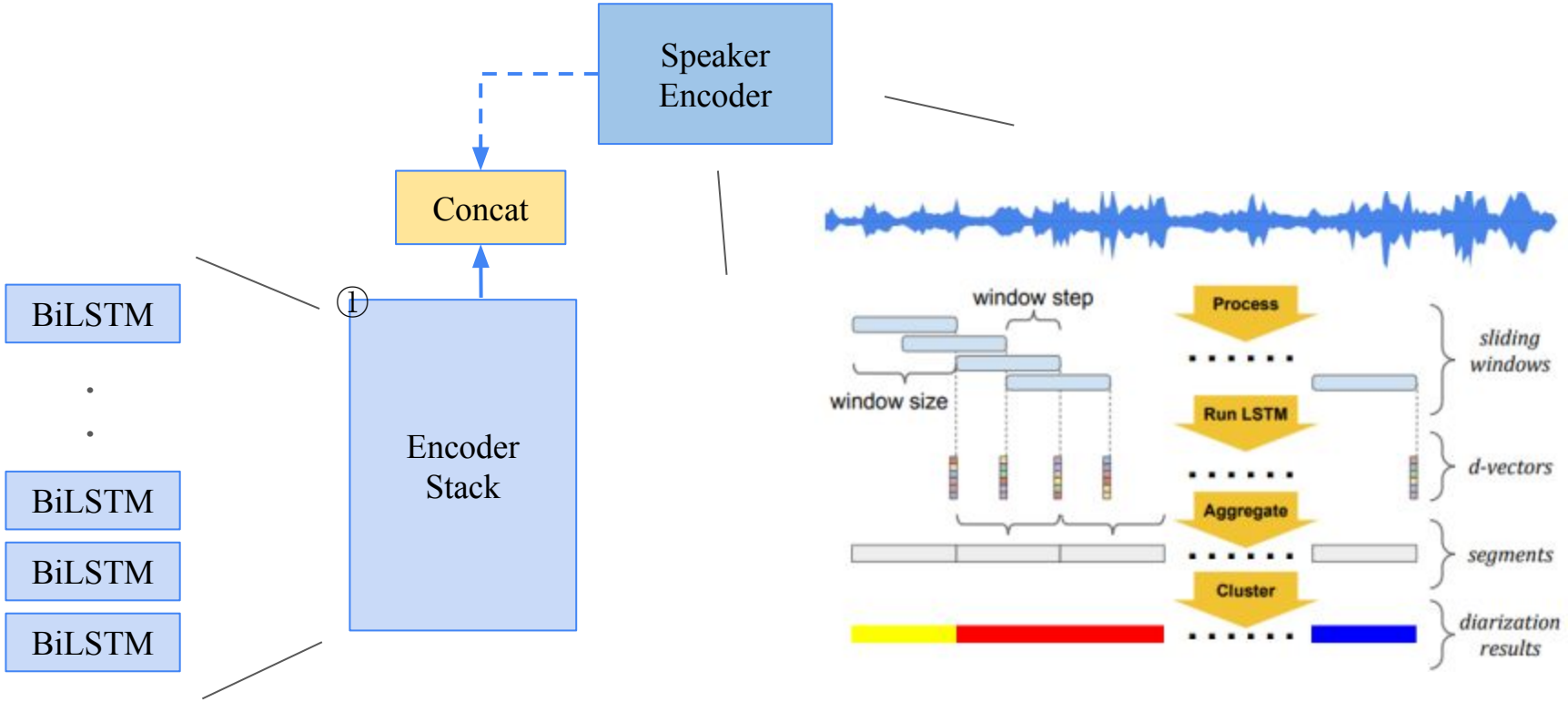
* Encoder는 8-layer stacked BiLSTM으로 구성.

* Encoder의 input은 80-log-mel-spectrogram 사용

* Encoder의 최종 output은 (speaker encoder를 사용했을 경우) vector와 concat하여 decoder output과 multi-head attention을 통해 encoder-decoder 간의 scaled-dot product attention이 이루어짐.

=> 중간 output(4-layer)은 보조 인식 작업에 활용
=> 최종 output은 번역 작업에 활용

레퍼런스 모델_Translatotron



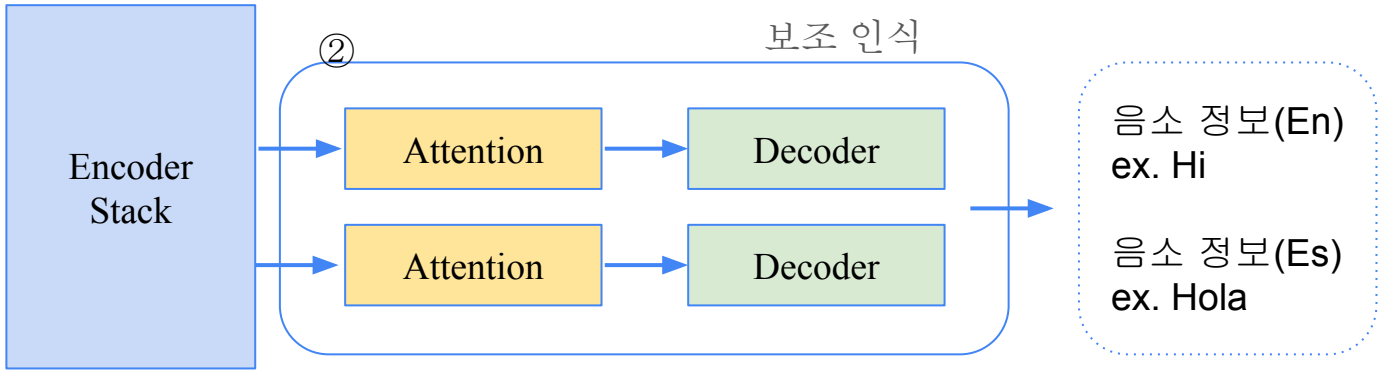
레퍼런스 모델_Translatotron

② [보조 인식 작업 - Auxiliary tasks]

*모델이 번역과 관련된 정보를 잘 학습할 수 있도록 보조적인 작업을 수행(학습 시 활용O, 추론 시 활용X)

* 인코더에서 받은 정보를 활용하여 음성에 해당하는 음소 문장을 생성하는 기능

* Low Level : Source에 해당하는 음소 문장 생성
* High Level : Target에 해당하는 음소 문장 생성

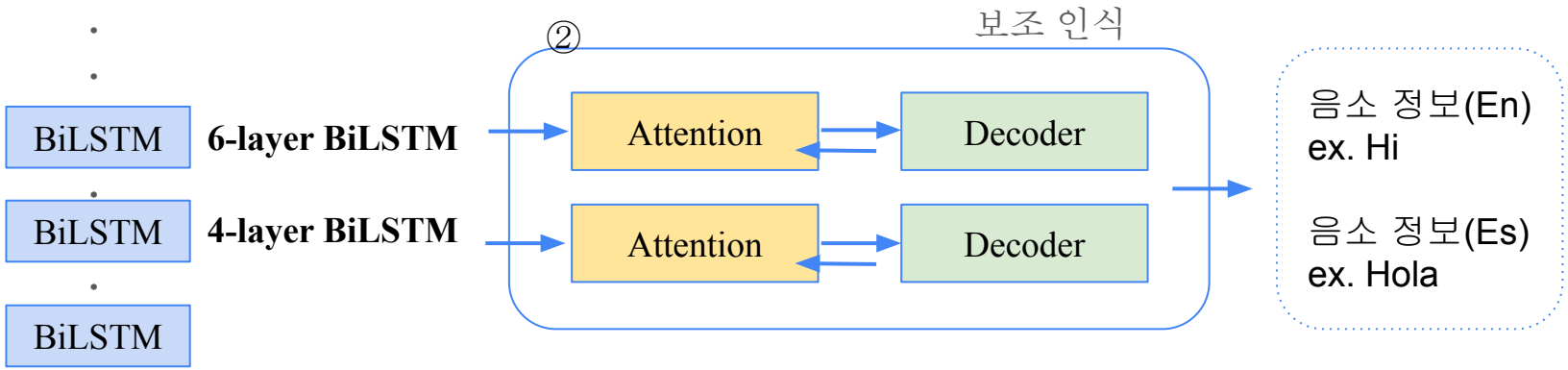


레퍼런스 모델_Translatotron

② [보조 인식 작업 - Auxiliary tasks]

* Lower Decoder : 2개의 LSTM Stack으로 구성. Encoder 4-layer의 Lower 결과물을 활용하여 Source 음운 출력
  => Encoder에서 Source에 대한 정보를 학습 및 추출할 때 보조하는 역할

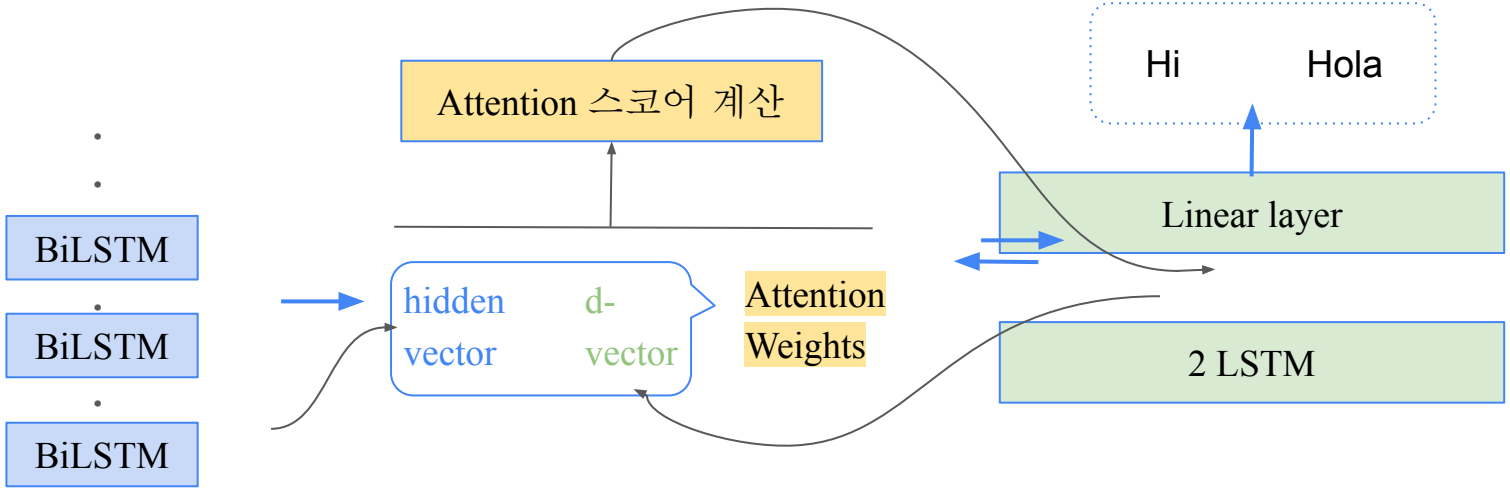
* Higher Decoder : 2개의 LSTM Stack으로 구성. Encoder 6-layer의 Higher 결과물을 활용하여 Target 음운 출력
  => Encoder에서 Target에 대한 정보를 학습 및 추출할 때 보조하는 역할



레퍼런스 모델_Translatotron

② [보조 인식 작업 - Auxiliary tasks]

Attention : 1개의 Head를 갖고 있는 Additive Attention 를 활용
Decoder는 Tacotron 2 TTS model과 비슷한 구조를 가짐



레퍼런스 모델_Translatotron

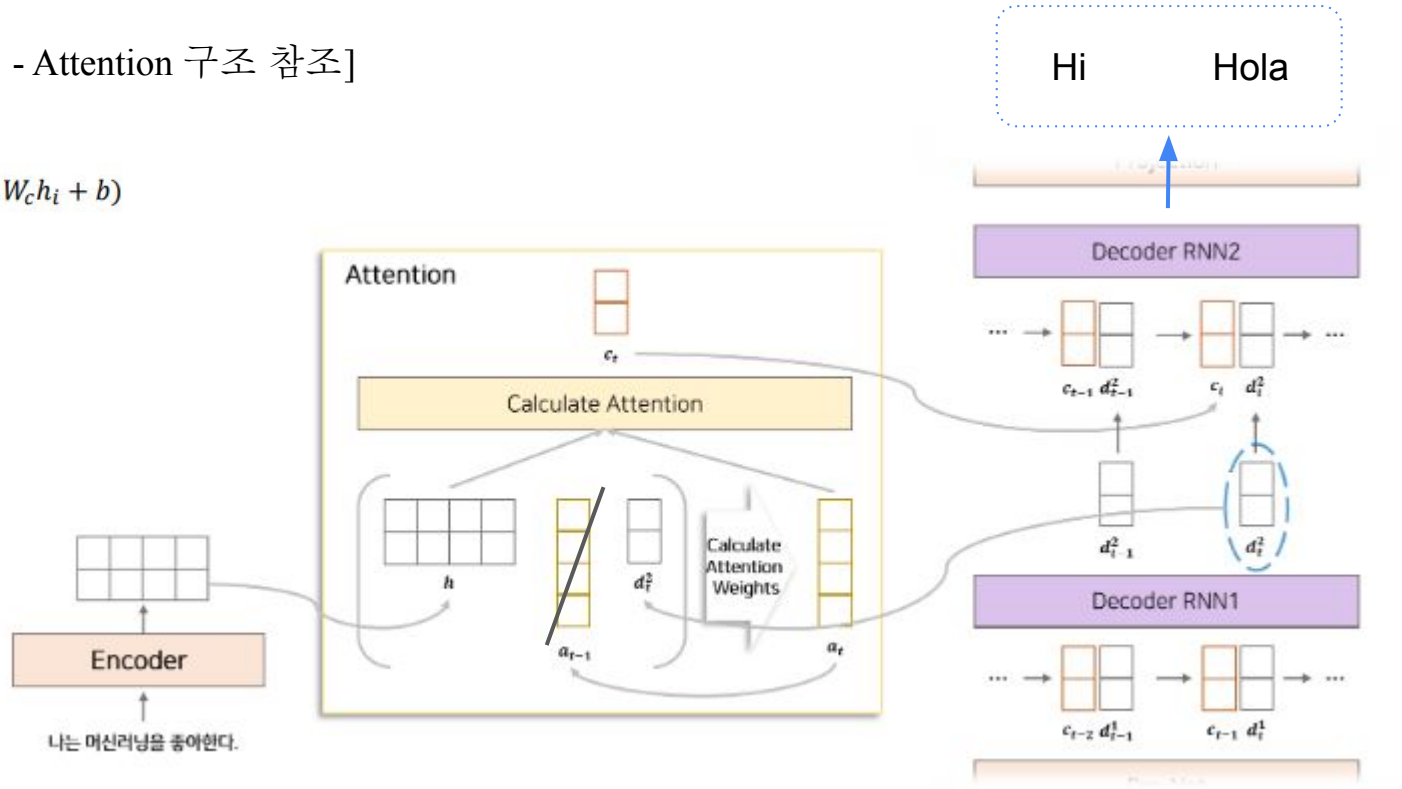
[Tacotron 2 TTS 모델의 - Attention 구조 참조]

$$s_{t,i} = w_a^T \tanh(W_b d_{t-1} + W_c h_i + b)$$

$$a_{t,i} = \frac{\exp(s_{t,i})}{\sum_{i=1}^{n} \exp(s_{t,i})}$$

$$a_t = [a_{t,1}, a_{t,2}, \dots, a_{t,n}]$$

$$c_t = \sum a_{t,i} h_i = a_t h$$



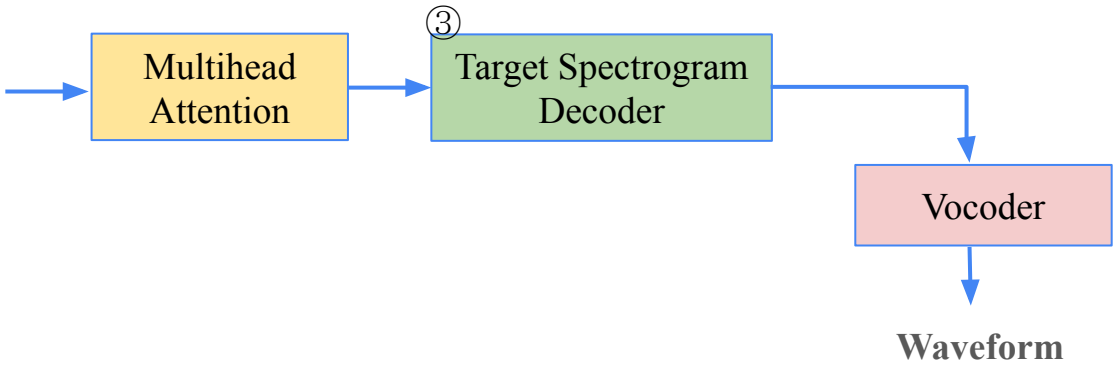
레퍼런스 모델_Translatotron

③ [번역된 음성 생성]

* 인코딩 된 정보를 바탕으로 번역된 음성의 Spectrogram을 생성

* Encoder에서 받은 Source 정보와 화자 정보(Speaker Enc.)를 바탕으로 Target 언어와 Target 화자의 음성 (Spectrogram)을 생성

* 화자 정보에 따라 다양한 Target 화자의 음성을 생성할 수 있음



레퍼런스 모델_Translatotron

③ [번역된 음성 생성]

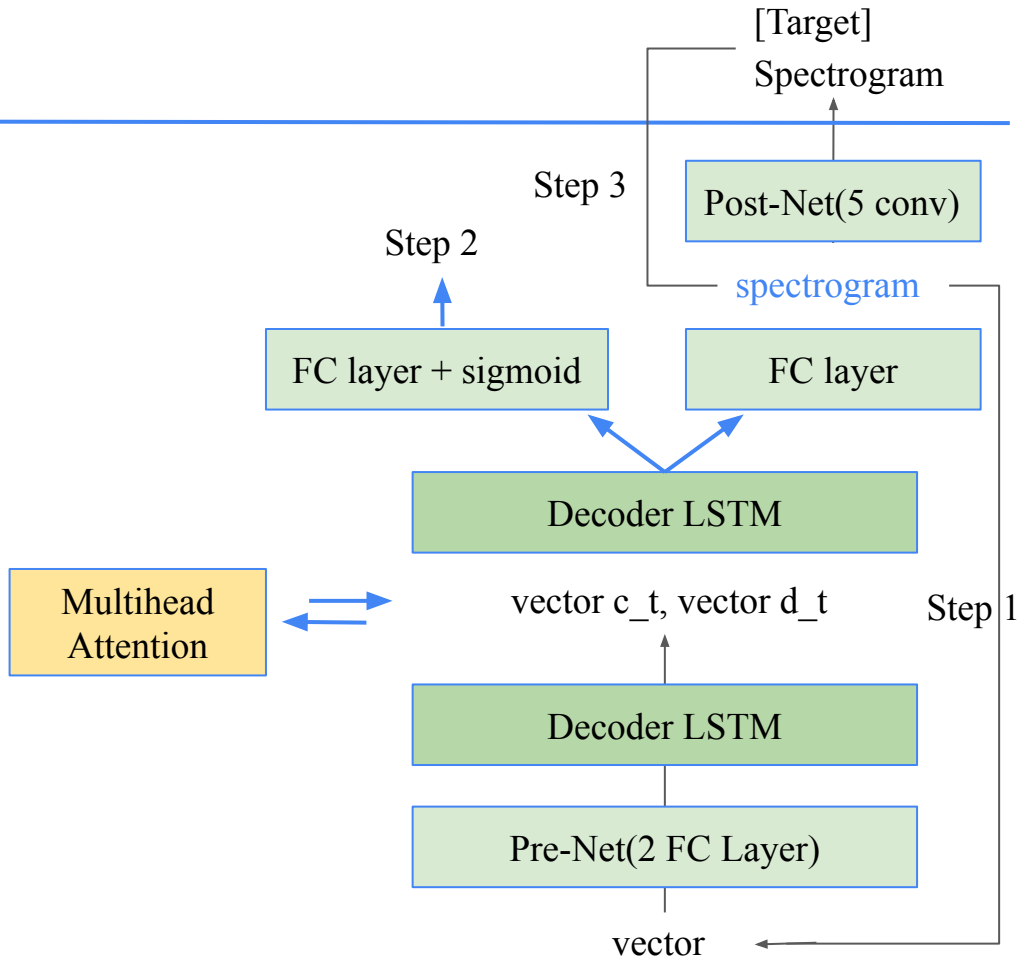
Step 1) 현재 시점의 Spectrogram을 생성
Step 2) 현재 시점의 종료확률을 계산 (0 or 1)
Step 3) mel-spectrogram의 품질을 향상

* Decoder가 생성한 Mel-Spectrogram Frame과 실제 Label의 Mel-Spectrogram Frame간의 MSE로 Loss를 계산한다.

* Decoder에서 Mel-Spectrogram까지만 생성하고, Vocoder를 사용해 음성 신호로 변환

* RNN 기반이라 Stack을 쌓는 병렬처리가 불가.

* Pre-Net : 이전시점의 정보를 압축하는 Bottle Neck 역할



레퍼런스 모델_Translatotron

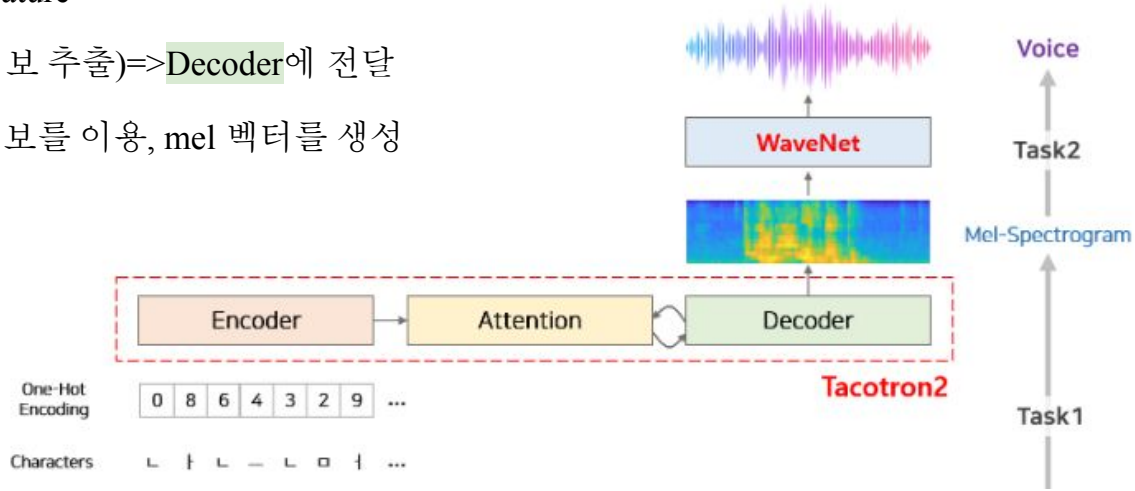
[Tacotron 2]

* Tacotron 2 = pre-net + (autoregressive LSTM stack) + post-net

* character입력=>(encoder)=>hidden feature

* hidden feature=>(시간순서에 맞게 정보 추출)=>Decoder에 전달

* Decoder는 attention으로부터 얻은 정보를 이용, mel 벡터를 생성



Background_Translatotron Experiment

[Tacotron 2]

- 데이터셋
  - 스페인어-영어(문장 및 음성) 데이터셋 사용
- 훈련방법
  - TTS 모듈을 활용하여 Target 스크립트를 기반으로 음성 생성
  - 생성된 음성을 Target 음성으로 활용하여 Translatotron 학습
- 평가방법
  - BLEU (번역평가) - 위의 표
  - MOS(음질 평가) - 아래의 표

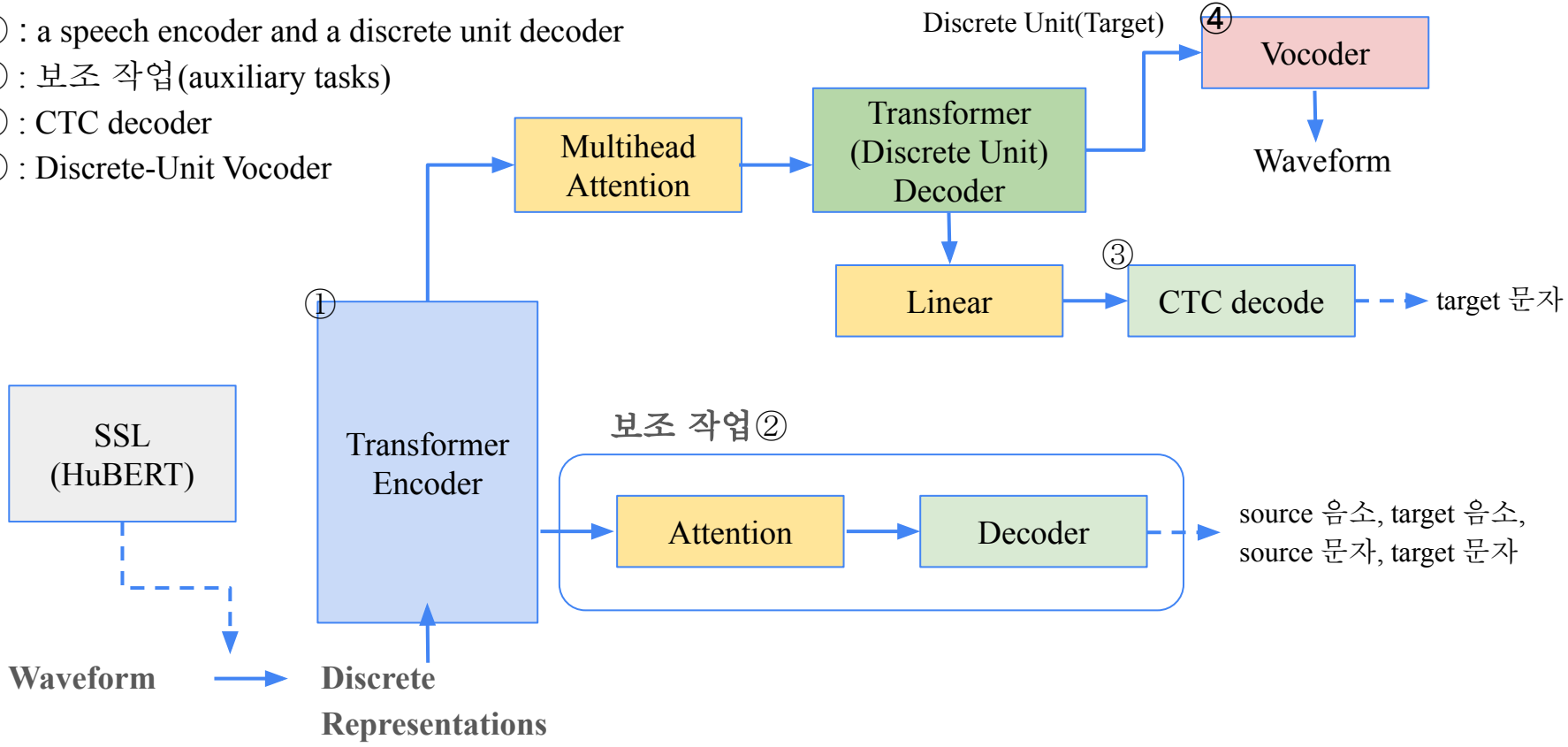
Auxiliary loss	dev1	dev2	test
None	0.4	0.6	0.6
Source	7.4	8.0	7.2
Target	20.2	21.4	20.8
Source + Target	24.8	26.5	25.6
Source + Target (1-head attention)	23.0	24.2	23.4
Source + Target (encoder pre-training)	30.1	31.5	31.1
ST [19] → TTS cascade	39.4	41.2	41.4
Ground truth	82.8	83.8	85.3

Model	Vocoder	Conversational	Fisher-test
Translatotron	WaveRNN	4.08 ± 0.06	3.69 ± 0.07
	Griffin-Lim	3.20 ± 0.06	3.05 ± 0.08
ST→TTS	WaveRNN	4.32 ± 0.05	4.09 ± 0.06
	Griffin-Lim	3.46 ± 0.07	3.24 ± 0.07

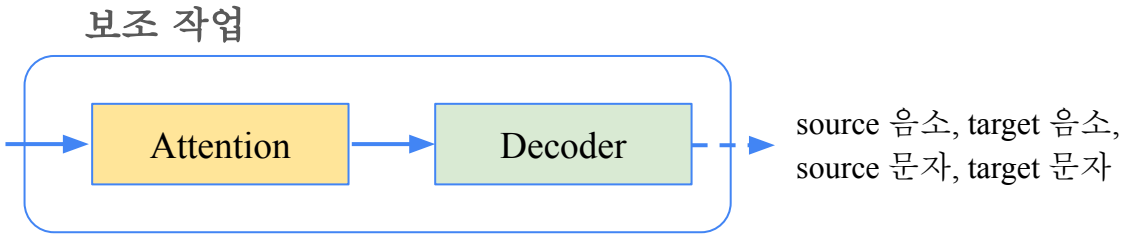
S2UT

레퍼런스 모델_S2UT

① : a speech encoder and a discrete unit decoder
② : 보조 작업(auxiliary tasks)
③ : CTC decoder
④ : Discrete-Unit Vocoder



레퍼런스 모델_S2UT



Dircrete S2ST의 장점

: 적은 컴퓨팅 비용, 적은 디코딩 단계

- self-supervised learning => discretized speech units

- 이 연구에서는 멜 스펙트로그램 기능 대신 대상 음성의 자체 감독 이산 표현을 예측하여 대상 음성을 간접적으로 모델링하는 문제를 해결

- 이산 단위(Discrete Unit)는 음성에서 음성 또는 운율 정보를 식별

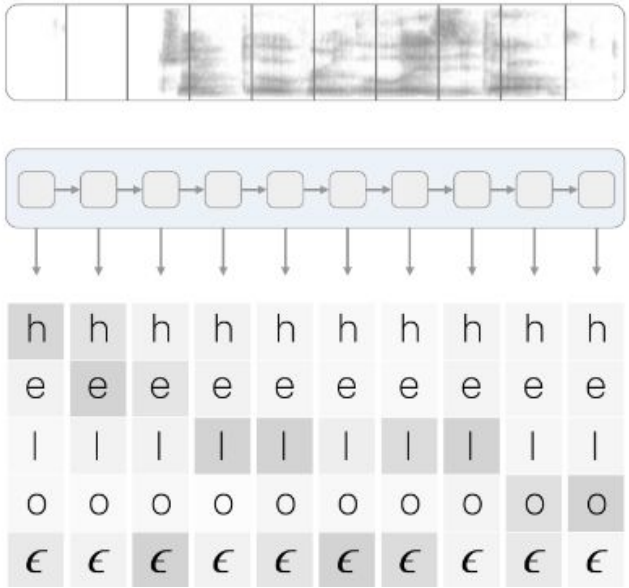
- Transcript 없어도 가능.
CTC 디코딩으로 음성 및 텍스트 출력 간의 길이 불일치 문제 개선

레퍼런스 모델_S2UT

[CTC Loss]

Motiv.

음성 인식 모델(ASR)을 학습하려면 음성(피처) 프레임 각각에 음소에 대한 레이블 정보가 있어야 함.(aligned) 단, MFCC와 같은 feature는 크기가 작아 레이블링이 많이 필요하고 정확도가 떨어짐.



How to use.

어떤 단어의 character가 audio와 alignment에 맞는지, 주어진 input과 output 사이의 가능한 모든 alignment의 가능성을 합해 loss를 계산한다.
모델을 쌓고 CTC loss를 적용하면 음성인식이 가능하다.

$$p(Y \mid X) = \sum_{A \in \mathcal{A}_{X,Y}} \prod_{t=1}^{T} p_t(a_t \mid X)$$

The CTC conditional **probability** | **marginalizes** over the set of valid alignments | computing the **probability** for a single alignment step-by-step.

레퍼런스 모델_S2UT

[Experiment]

- SSL 모델은 HuBERT 사용

- TTS가 실행 시간의 가장 큰 비율을 차지한다. (S2T+TTS의 경우 >89%, SR+MT+TTS의 경우 >81%) S2UT 시스템은 S2T+TTS에 비해 1.5배 더 빠르게 실행, 최대 55% 메모리를 감소

- 평가(BLEU)를 위해 오픈 소스 ASR 모델을 사용.
(ASR : Wav2Vec 2.0 Large (LV-60) + Self Training / 960 hours / Libri-Light + Librispeech dataset)

- vocoder는 unit-based Hifi-GAN 모델 사용

레퍼런스 모델_S2UT

ID		BLEU dev speech	BLEU dev text	BLEU dev2 speech	BLEU dev2 text	BLEU test speech	BLEU test text	MOS test
1	Synthetic target	88.5	100.0	89.4	100.0	90.5	100.0	3.49 ± 0.14
	Cascaded systems:							
2	ASR (beam=10) + MT (beam=5) + TTS	42.1	45.1	43.5	46.1	43.9	46.3	3.37 ± 0.15
3	S2T (beam=10) + TTS	38.5	41.1	39.9	42.4	40.2	42.1	3.43 ± 0.14
	Direct systems:							
4	Transformer *Translatotron* ($r = 5$, w/ *sp, tp*)	25.0	-	26.3	-	26.2	-	-
5	Transformer *Translatotron* ($r = 5$, w/ *sc, tc*)	32.9	-	34.1	-	33.2	-	3.31 ± 0.11
6	S2UT, no reduction ($r = 1$, w/ *sc, tc*)	33.4	-	34.6	-	34.1	-	3.35 ± 0.14
7	S2UT *stacked* ($r = 5$, w/ *sc, tc*)	34.0	-	34.5	-	34.4	-	-
	Direct systems with dual modality output:							
8	S2UT *stacked* + CTC ($r = 5$, w/ *sc, tc*)	34.4	36.4	36.4	37.9	34.4	35.8	3.32 ± 0.14
9	S2UT *reduced* + CTC (w/ *sc, tc*), beam=1	36.8	40.0	38.4	41.5	38.5	40.7	-
10	S2UT *reduced* + CTC (w/ *sc, tc*), beam=10	**38.2**	**41.3**	**39.5**	**42.2**	**39.9**	**41.9**	3.41 ± 0.14
	From the literature*:							
11	*Translatotron* (Jia et al., 2019b)	24.8	-	26.5	-	25.6	-	3.69 ± 0.07
12	+ pre-trained encoder (Jia et al., 2019b)	30.1	-	31.5	-	31.1	-	-
13	*Translatotron 2* (Jia et al., 2021)	-	-	-	-	37.0	-	3.98 ± 0.08
14	+ data augmentation (Jia et al., 2021)	-	-	-	-	40.3	-	3.79 ± 0.09

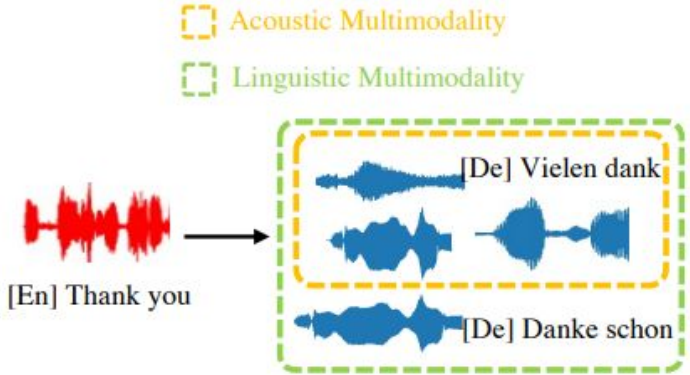
[Paper] Experimental Method

Instruction : Objective

[direct S2ST의 목표]

1) 성능 개선(**high quality**) : direct S2ST 에서 가장 큰 목적이 되는 것. (without using the transcription.)

2) 지연시간 개선(**low latency**) : 실시간 적용도 고려했을 때, 추론을 더 빠르게 할 수 있도록 개선하는 것.



[해결해야 할 과제 구체화]

1) "**acoustic multimodality**" 을 해결하여 번역 성능 개선

2) 병렬 모델(parallel model)을 만들 때 불확실성 (**indeterminacy**)이 커지는 문제 개선

Instruction : Proposed

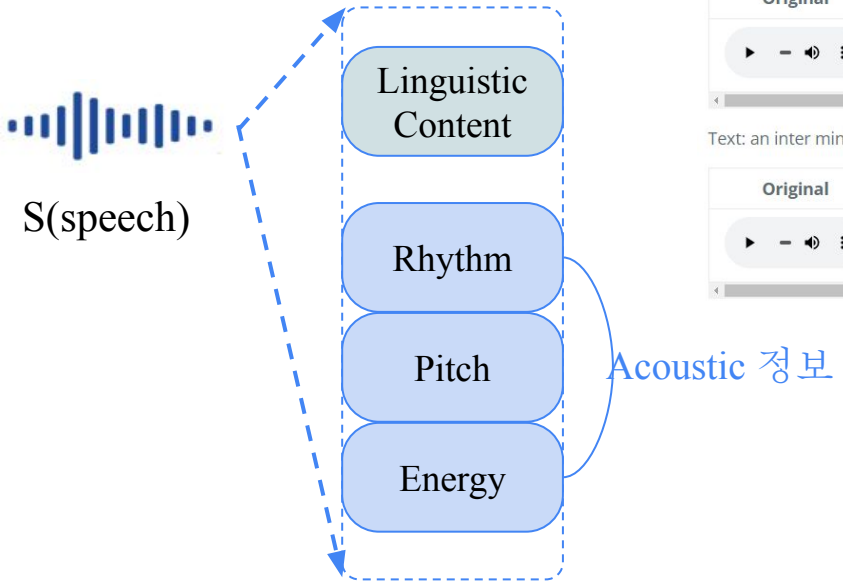
모델의 파이프라인 = S2UT 모델 사용

[Method]

1. Speech Analysis 단계에서 양방향 Bilateral Perturbation(BiP) 방법론을 제시
2. TTS 단계에서 AR => NAR 모델을 제안
3. Encoding/Decoding 구조에서 병렬적 구조를 제안
    a. Conformer 제안
    b. 지식증류 제안
    c. Mask-Algorithm 제안
    d. NPD 제안

[Result]
번역 성능을 높이고, 추론 시간을 단축하기 위함

Speech Analysis : Acoustic Multi-modality



Text: really interesting work will finally be undertaken on that topic.

Original	Pitch Norm	Energy Norm	F = fs(pr(peq(x)))	F' = peq(fs(pr(x)))	RR
					0:00

Text: an inter ministerial committee on disability was held a few weeks back.

Original	Pitch Norm	Energy Norm	F = fs(pr(peq(x)))	F' = peq(fs(pr(x)))	RR
		0:00	0:00		

Speech Analysis : BiP(Bilateral Perturbation)

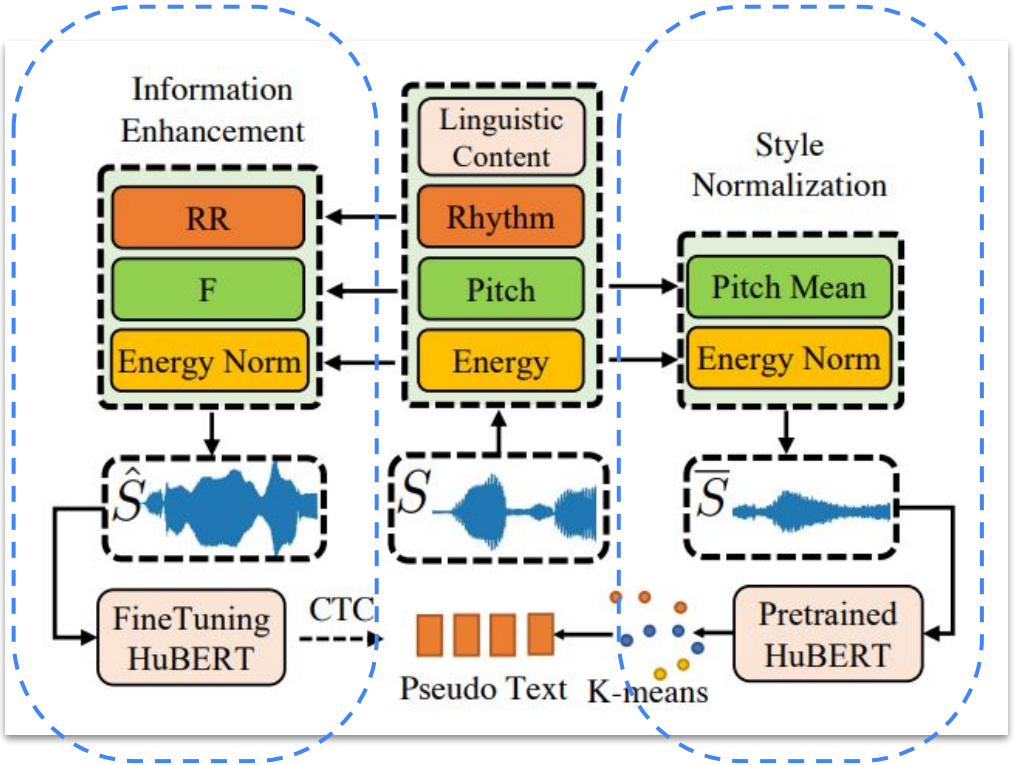
① format shifting **fs**
– Unif(1,1.4)

② pitch randomization **pr**
– Unif(1,2) ~ Unif(1,5)

③ random resampling **RR**
– length 19 frames to 32 frames

④ random frequency shaping **peq**

$$F = fs(pr(peq(S)))$$



Speech Analysis : BiP(Bilateral Perturbation)

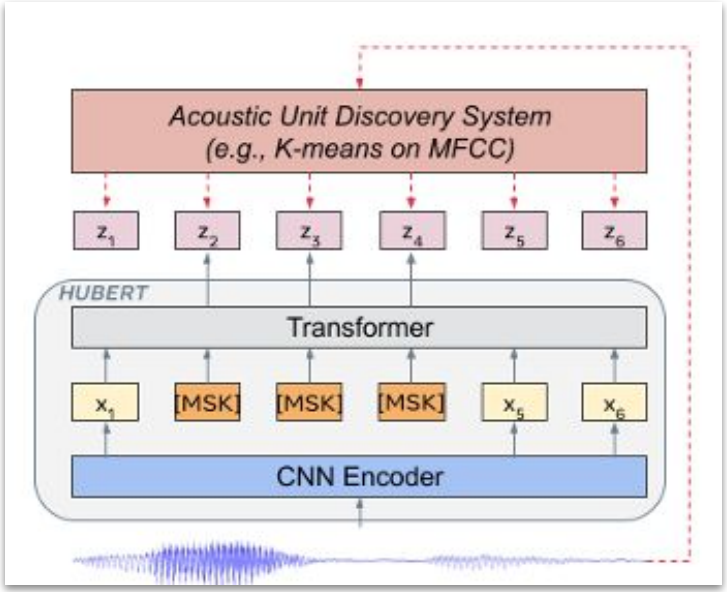
[HuBERT]

* Self-Supervised(자기지도 학습)을 적용한 음성 표현 학습

[자기지도 학습의 필요성]

- 다양한 측면의 비언어 잡음(웃음, 말 끌기 등)을 구체적으로 모델링
- 음향 및 언어 모델을 학습하도록 함
- 음성 데이터만으로 학습

cf) 전통적인 음향 모델은 텍스트와 음성 쌍을 강제 정렬해서 음성-의사 레이블을 제공



Speech Analysis : BiP(Bilateral Perturbation)

BiP 적용 전후 음성 샘플 비교

단위 오류율**(UER)** :

음향 변화의 불확정성과 멀티모달리티를
측정하기 위한 평가 지표

Pretrained SSL: 최대 22.7% UER (리듬)
=> 불확정적(indeterministic)

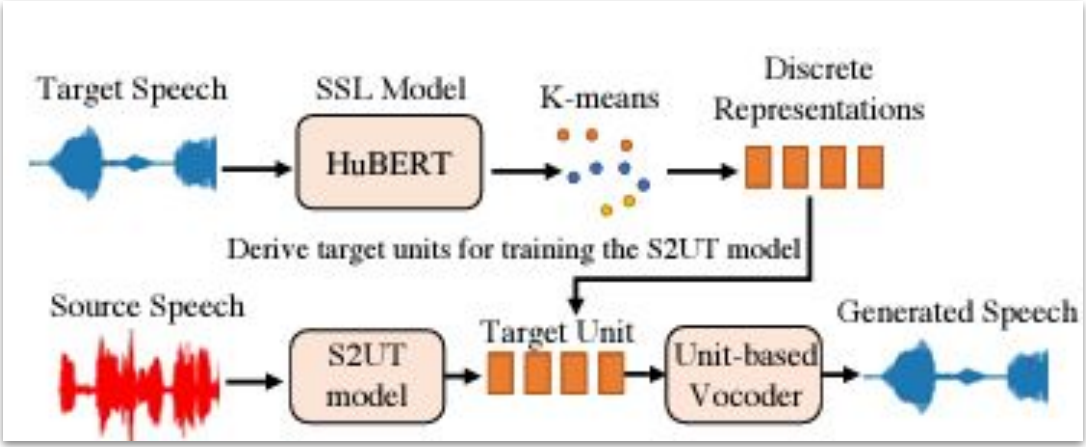
=> BiP의 효율성 입증

Acoustic	Pretrained	BiP-Tuned
Reference	0.0	0.0
Rhythm $\hat{S}_r$	22.7	10.2
Pitch $\hat{S}_p$	16.3	4.3
Energy $\hat{S}_e$	10.5	1.8

TranSpeech : 1. Architecture

TranSpeech-S2ST pipeline 에서 공통적으로 갖는 구조:

1. SSL(자기 지도학습) 모델은 HuBERT(tuned by BiP) 사용
2. Sequence-to-Sequence 모델은 S2UT 사용
3. Vocoder(음성 합성)는 Unit-based Vocoder 사용



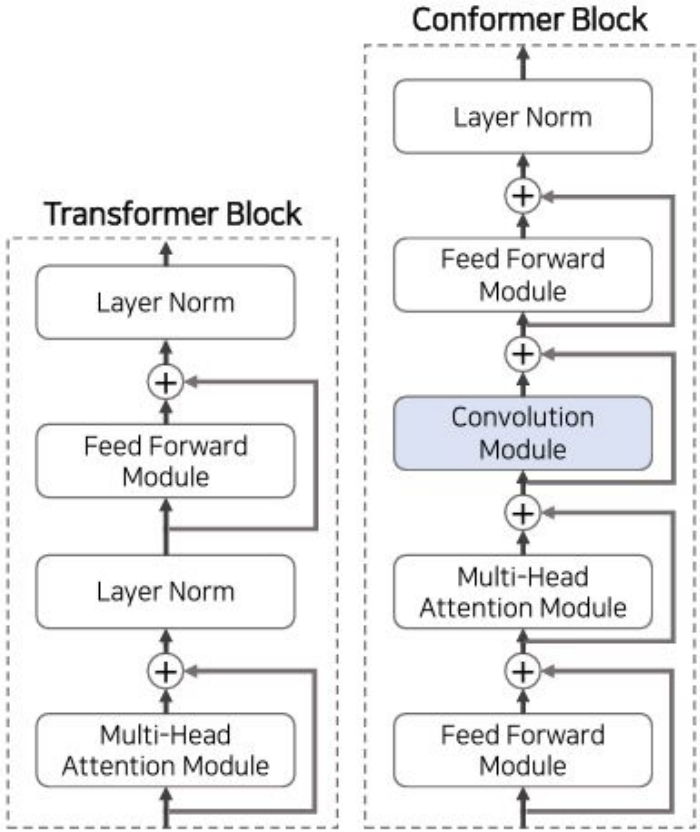
TranSpeech : 1. Conformer Encoder

1. Conformer Encoder

* 구조 특징
  - Transformer Block 대신Conformer Block을활용

* Conformer Block
: Conformer 논문에서Transformer 대신 사용된 일련의 모듈 조합
  - Transformer Block에 Convolution이 추가된 형태

* Conformer Block의 장점
  - Self Attention 모듈은 Global 정보를 취합
  - Convolution 모듈은 Local 정보를 취합
  - Global, Local 정보를 함께 다루는 음성task에서 효과.



TranSpeech : 1. Conformer Encoder

1. Conformer Encoder

- 음성 인식 도메인에서 CNN과 Transformer Encoder의 Self Attention을 결합한 네트워크를 기반으로 함.
- CNN과 Transformer의 이점을 잘 활용
- 다양한 downstream task에서 좋은 결과를 냄.

Hyperparameter		TranSpeech
Conformer Encoder	Conv1d Layers	2
	Conv1d Kernel	(5, 5)
	Encoder Block	6
	Encoder Hidden	512
	Encoder Attention Heads	8
	Encoder Dropout	0.1
Length Predictor	Projection Dim	512
Unit Decoder	Unit Dictionary	1000
	Decoder Block	6
	Decoder Hidden	512
	Decoder Attention Headers	8
	Decoder Dropout	0.1

Table 4: Hyperparameters of TranSpeech.

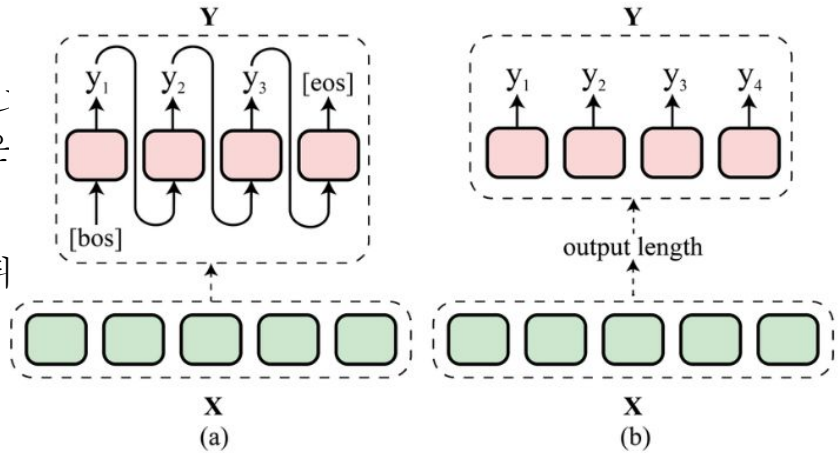
TranSpeech : 1. NAR Unit Decoder

1-2. Non-autoregressive Unit Decoder(NAR Unit Decoder)

ARvsNAR

- Autoregressive 모델들은 이전 샘플을 통해 다음 샘플을 하나씩 만드는 방식을 사용하여 생성 속도가 매우 느림. 비교적 높은 성능.

- Non-autoregressive 모델을은 앞의 샘플을 보지 않고도 그 뒤 샘플을 생성할 수 있어 보통 parallel 라 표현. 빠른 추론 속도



(a) Autoregressive; (b) Non-Autoregressive

특징)

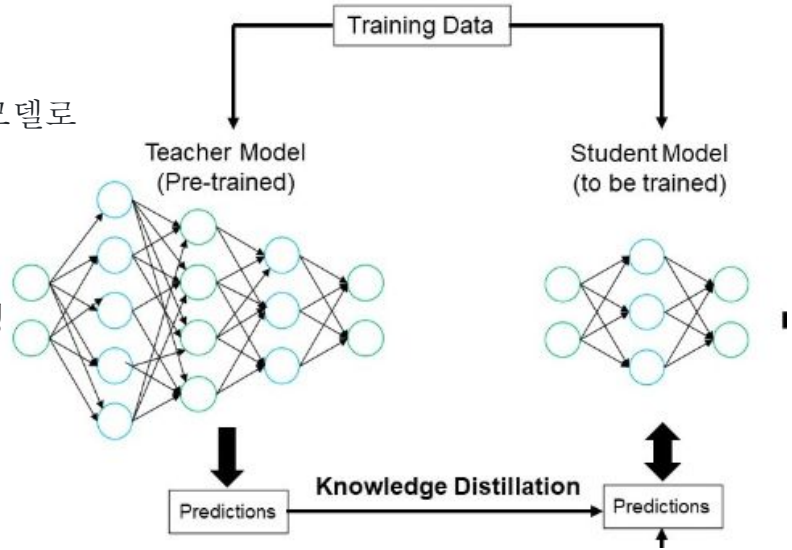
- 병렬 디코딩이 출력 토큰 간의 조건부 독립성을 가정한다.
  ⇔ 이전까지의 토큰 결과를 반영하지 않음

* 출처 : Non-Autoregressive Neural Machine Translation, ICLR 2018

TranSpeech : 2. Knowledge distillation

2. Knowledge distillation

: 더 정확도가 높은 모델(AR 모델)로부터 증류한 지식을 NAR 모델로 transfer



특징)

- 정확도 개선. 속도와 정확도 간의 합리적인 trade-off 달성
- AR 모델이 덜 noisy한 점을 이용
- Linguistic multimodality 개선

* 출처 : Non-Autoregressive Neural Machine Translation, ICLR 2018

TranSpeech : 3. Mask-Predict Algorithm

3. Mask-Predict 디코딩 Algorithm
-> NAR 디코딩에 적용

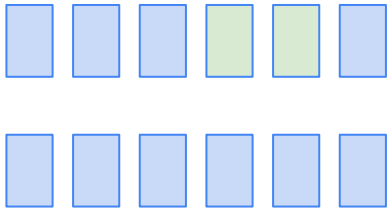
[Problem]
- 기존의 NAR 방법들은 각 토큰을 독립적 조건부 확률로 보았기 때문에 불안정한 결과 도출 문제가 있었음.
  ex. Hi hello, Thanks thank, you thank
- 마스크 위치는 랜덤하게 정함
- Cross Entropy loss를 업데이트

[수행]
- target sequence의 길이 N을 예측 & Mask all units
- After Mask, masked units Y(yi)와 unmasked units Y_obs를 조건부 확률로 예측

[Result]
- 더 나아가 기존의 AR 모델이 생성한 결과를 이용해 학습하는 distillation 방법 제안
- Target Sequence의 길이(N)를 구하는 과정 필요

$$y_i^t = \arg\max_w P(y_i = w \mid X, Y_{obs}^t; \theta)$$

$$p_i^t = \max_w P(y_i = w \mid X, Y_{obs}^t; \theta)$$



* 출처 : Non-Autoregressive Neural Machine Translation, ICLR 2018

TranSpeech : 4. Advanced Decoding

Target Length Beam

: 디코딩 과정에서 기억해야 하는 후보 수를 K개로 제한하여 계산의 효율성을 높인 방식. K는 빔 너비 (Beam width) 또는 빔 크기 (Beam size)를 의미

=> 가장 높은 확률을 가진 상위 K 길이 후보를 선택(5-15 정도의 값)

=>길이가 다른 동일한 예제를 병렬로 디코딩

=> 다음에 가장 높은 확률의 시퀀스를 선택

Noisy Parallel Decoding(NPD)

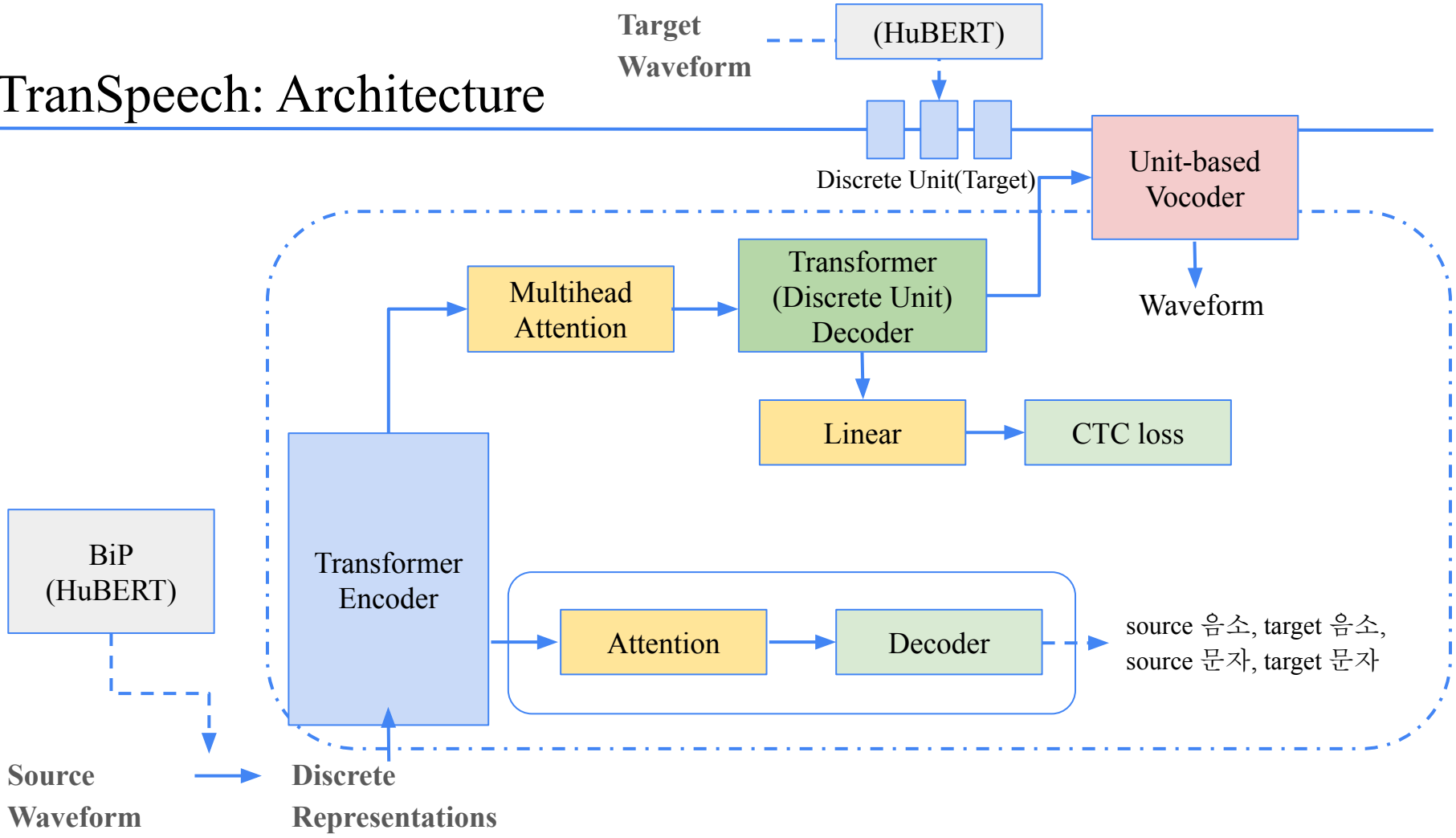
: AR 모델에 있는 디코딩 절차가 없음

=> 따라서 생성된 토큰에 대해 AR 모델에서의 확률을 다시 구한다.

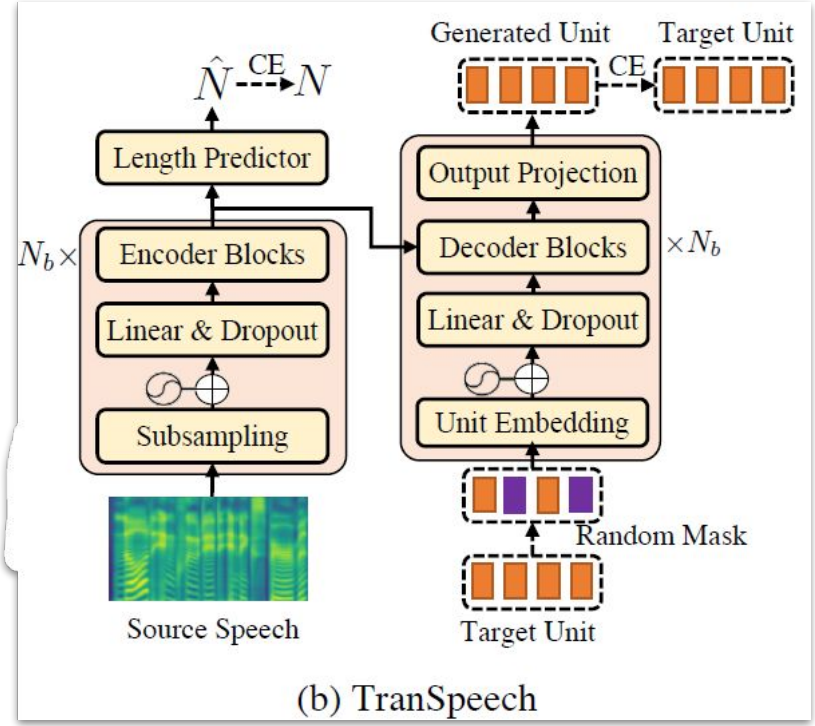
=>AR teacher 사용, 최적의 번역을 계산

=> 최대가 되는 Fertility(목표) 분포를 기반으로 디코딩하기

TranSpeech: Architecture



TranSpeech: Architecture



* 출처 : Non-Autoregressive Neural Machine Translation, ICLR 2018

Experiment

Experiment

Dataset : CVSS-C dataset (En-Fr, Fr-En, En-Es) (from CoVoST 2)

*CVSS는 약 21개 언어와 영어의 번역 pair 중 3쌍으로(En-Fr/Fr-En/En-Es), S2ST의 훈련 코퍼스
*단일 화자의 발화 데이터
*데이터 길이의 info) mean: 9.63, med: 10, max: 27, min: 1[단위:word]

Unchanged:

- SSL : (m)HuBERT 를 사용.
- Vocoder : unit-based HiFi-GAN vocoder를 사용

평가지표 :

- 번역 성능 : BLEU 평가지표
- 음질 : MOS 평가지표

+ [MOS(Mean Opinion Score) 평가지표]

: 피실험자에게 음성을 들려주고 1점-5점까지 직접 평가

Experiment : BLEU

[BLEU (Bilingual Evaluation Understudy) 평가지표]

: 기계 번역 결과와 사람이 직접 번역한 결과가 얼마나 유사한지 비교하여 번역에 대한 성능을 측정하는 방법 ⇔ 생성된 speech를 text로 변환(ASR 모델 사용)한 결과와 reference text를 비교 계산한 값.

ex. 모델이 추론한 문장이 candidate , 사람이 해석한 문장을 Reference 라고 할 때,

Candidate1 : the the the the the the the
Candidate2 : the cat the cat on the mat

Reference1 : the cat is on the mat
Reference2 : there is a cat on the mat

바이그램	the cat	cat the	cat on	on the	the mat	SUM
$Count$	2	1	1	1	1	6
$Count_{clip}$	1	0	1	1	1	4

$$p_n = \frac{\sum_{n\text{-}gram \in Candidate} Count_{clip}(n\text{-}gram)}{\sum_{n\text{-}gram \in Candidate} Count(n\text{-}gram)}$$

Experiment : BLEU

$$p_n = \frac{\sum_{n\text{-}gram \in Candidate} Count_{clip}(n\text{-}gram)}{\sum_{n\text{-}gram \in Candidate} Count(n\text{-}gram)}$$

$$BLEU = BP \times exp(\sum_{n=1}^{N} w_n \log p_n)$$

$$BP = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \leq r \end{cases}$$

c : candidate 의 길이
r : candidate 와 가장 길이 차이가 작은 reference의 길이

Experiment : Table

ID	Model	BiP	Fr-En	En-Fr	En-Es	Speed	Speedup
Autoregressive models							
1	Basic Transformer (Lee et al., 2021a)†	✗	15.44	15.28	10.07	870	1.00×
2	Basic Norm Transformer (Lee et al., 2021b)†	✗	15.81	15.93	12.98		
3	Basic Conformer	✗	18.02	17.07	13.75	895	1.02×
4	Basic Conformer	✓	**22.39**	**19.65**	**14.94**		
Non-autoregressive models with naive decoding							
5	TranSpeech - Distill	✗	14.86	14.12	10.27	**9610**	**11.04**×
6	Transpeech - Distill	✓	16.23	15.9	10.94		
7	TranSpeech	✓	**17.24**	**16.3**	**11.79**		
Non-autoregressive models with advanced decoding							
8	TranSpeech (iter=15)	✓	18.03	16.97	12.62	4651	5.34×
9	TranSpeech (iter=15 + b=15)	✓	18.10	17.05	12.70	2394	2.75×
10	TranSpeech (iter=15 + b=15 + NPD)	✓	**18.39**	**17.50**	**12.77**	2208	2.53×
Cascaded systems							
11	S2T + TTS	/	27.17	34.85	32.86	/	/
12	Direct ASR	/	71.61	50.92	68.75	/	/
13	Direct TTS	/	82.41	76.87	83.69	/	/

Experiment : Case Study

Table 2: **Two examples comparing translations produced by TranSpeech and baseline models.** We use the bond fonts to indicate the the issue of **noisy and incomplete translation.**

Source:	l'origine de la rue est liée à la construction de la place rihour.
Target:	the origin of the street is linked to the construction of rihour square.
Basic Conformer:	the origin of the street is linked to the construction of **the.**
TranSpeech:	th origin of the **seti** is linked to the construction of the **rear**.
TranSpeech+BiP:	the origin of the street is linked to the construction of the **ark.**
TranSpeech+BiP+Advanced:	the origin of the street is linked to the construction of the work.
Source:	il participe aux activités du patronage laïque et des pionniers de saint-ouen.
Target:	he participates in the secular patronage and pioneer activities of saint ouen.
Basic Conformer:	he participated in the activities of the late patronage a **d** see.
TranSpeech:	he **takes in the patronage activities in of saint.**
TranSpeech+BiP:	he participated in the activities of the lake patronage and **say pointing**
TranSpeech+BiP+Advanced:	he participated in the activities of the wake patronage and saint pioneers

Conclusion

Conclusion

for Future unit - textless S2ST studies,

- BiP 개념을 포함한 TranSpeech를 제시한 것
  (음향 멀티모달리티 크게 줄임)
- NAR Decoder를 적용한 S2ST 기술을 최초로 확립 -> 속도 개선
- TranSpeech에서 병렬성(parallel)을 최대한 활용
- 지식 증류로 언어적 멀티모달리티 줄임