

K o S p e e c h

Open Source Project for Korean End-to-End Automatic Speech Recognition in Pytorch

kakaobrain

2020년 8월 3일

발표자 : 김수환

C CONTENTS

1. 발표자 소개

2. KoSpeech 소개

3. 2019 네이버 Speech 해커톤

- 대회 소개

- 시행착오

- 피드백

4. KoSpeech 개발 프로세스

- 데이터

- 모델

- 적용기법

5. To-Do List

6. Q & A

Presenter Introduction

KoSpeech: Open Source Project for Korean End-to-End Automatic Speech Recognition in Pytorch

| 발표자 소개



Soohwan **Kim**

Education

- 광운대학교 전자통신공학과 전공 *2014.03 ~ Present*
- 광운대학교 정보융합학부 데이터사이언스 부전공 *2018.09 ~ Present*

Experience

- 서강대학교 청각 지능 연구실 학부연구생 *2020.04 ~ Present*
- KoSpeech 오픈소스 프로젝트 *2020.01 ~ Present*
- 네이버 2019 AI 해커톤 Speech 12위 *2019.09 ~ 2019.10*

Interest

- Automatic Speech Recognition
- Natural Language Processing
- Software Development

Contact

- Tel: 010-4564-4668
- Blog: <https://blog.naver.com/sooftware>
- Email: sh951011@gmail.com
- Github: <https://github.com/sooftware>

KoSpeech Introduction

KoSpeech: Open Source Project for Korean End-to-End Automatic Speech Recognition in Pytorch

KoSpeech 소개



Open Source Project for Korean End-to-End (E2E) Automatic Speech Recognition (ASR) in Pytorch for Deep Learning Researchers

Python ☆ 72 🍷 23

- 20년 1월부터 시작한 End-to-End 방식의 한국어 음성 인식 오픈소스 프로젝트
- 기계번역 오픈소스인 OpenNMT에서 영감
- AI Hub에서 제공하는 1000h 한국어 음성 코퍼스에 대한 전처리와 학습 기능 제공
- 다양한 옵션을 통한 학습 가능
- 소스코드의 가독성과 확장성에 초점을 두고 진행
- IBM seq2seq, OpenNMT, Sean Naren DeepSpeech 등 여러 오픈소스 구조 참고
- 링크 : <https://github.com/sooftware/KoSpeech/>

```
usage: main.py [-h] [--mode MODE] [--sample_rate SAMPLE_RATE]
               [--frame_length FRAME_LENGTH] [--frame_shift FRAME_SHIFT]
               [--n_mels N_MELS] [--normalize] [--del_silence]
               [--input_reverse] [--feature_extract_by FEATURE_EXTRACT_BY]
               [--transform_method TRANSFORM_METHOD]
               [--time_mask_para TIME_MASK_PARA]
               [--freq_mask_para FREQ_MASK_PARA]
               [--time_mask_num TIME_MASK_NUM] [--freq_mask_num FREQ_MASK_NUM]
               [--architecture ARCHITECTURE] [--use_bidirectional]
               [--mask_conv] [--hidden_dim HIDDEN_DIM] [--dropout DROPOUT]
               [--num_heads NUM_HEADS] [--label_smoothing LABEL_SMOOTHING]
               [--num_encoder_layers NUM_ENCODER_LAYERS]
               [--num_decoder_layers NUM_DECODER_LAYERS] [--rnn_type RNN_TYPE]
               [--extractor EXTRACTOR] [--activation ACTIVATION]
               [--attn_mechanism ATTN_MECHANISM]
               [--teacher_forcing_ratio TEACHER_FORCING_RATIO]
               [--num_classes NUM_CLASSES] [--d_model D_MODEL]
               [--ffnet_style FFNET_STYLE] [--dataset_path DATASET_PATH]
               [--data_list_path DATA_LIST_PATH] [--label_path LABEL_PATH]
               [--spec_augment] [--noise_augment]
               [--noiseset_size NOISESET_SIZE] [--noise_level NOISE_LEVEL]
               [--use_cuda] [--batch_size BATCH_SIZE]
               [--num_workers NUM_WORKERS] [--num_epochs NUM_EPOCHS]
               [--init_lr INIT_LR] [--high_plateau_lr HIGH_PLATEAU_LR]
               [--low_plateau_lr LOW_PLATEAU_LR] [--valid_ratio VALID_RATIO]
               [--max_len MAX_LEN] [--max_grad_norm MAX_GRAD_NORM]
               [--rampup_period RAMPUP_PERIOD]
               [--decay_threshold DECAY_THRESHOLD]
               [--exp_decay_period EXP_DECAY_PERIOD]
               [--teacher_forcing_step TEACHER_FORCING_STEP]
               [--min_teacher_forcing_ratio MIN_TEACHER_FORCING_RATIO]
               [--seed SEED] [--save_result_every SAVE_RESULT_EVERY]
               [--checkpoint_every CHECKPOINT_EVERY]
               [--print_every PRINT_EVERY] [--resume]
```

KoSpeech 소개

- Documentation : <https://sooftware.github.io/KoSpeech/>
- Web Application : <http://www.kospeech.com/>

Attention

```
class kospeech.models.seq2seq.attention.LocationAwareAttention(d_model: int = 512, smoothing: bool = True) [source]
```

Applies a location-aware attention mechanism on the output features from the decoder. Location-aware attention proposed in "Attention-Based Models for Speech Recognition" paper. The location-aware attention mechanism is performing well in speech recognition tasks. We refer to implementation of ClovaCall Attention style.

Parameters:

- `d_model` (*int*) – dimension of model
- `smoothing` (*bool*) – flag indication whether to use smoothing or not.

Inputs: query, value, last_attn

- `query` (batch, q_len, hidden_dim): tensor containing the output features from the decoder.
- `value` (batch, v_len, hidden_dim): tensor containing features of the encoded input sequence.
- `last_attn` (batch_size * num_heads, v_len): tensor containing previous timestep's attention (alignment)

Returns: output, attn

- `output` (batch, output_len, dimensions): tensor containing the feature from encoder outputs
- `attn` (batch * num_heads, v_len): tensor containing the attention (alignment) from the encoder outputs.

Reference:

Documentation

KoSpeech

파일 선택 선택된 파일 없음

Predict

나 다음 주에 강릉 가는데 같이 가자 야 너네 학교 기숙사에서 자면 안 되냐?

Web Application

Naver AI Hackathon

KoSpeech: Open Source Project for Korean End-to-End Automatic Speech Recognition in Pytorch

네이버 AI 해커톤 - Speech

■ 대회 소개

네이버 Clova 팀에서 주최한 100시간의 한정된 데이터로 학습을 진행하여 CRR (Character Recognition Rate)를 기준으로 랭킹을 내는 컴피티션



1st

영화 평점과 지식in 문제로 한
첫번째 AI 해커톤!



2nd

2018년 겨울을 뜨겁게 달군 두번째 AI 해커톤!
AI Hackathon 2018 #Vision



3rd

2019년 여름
AI Hackathon 2019 #Speech

- https://campaign.naver.com/aihackathon_speech/

네이버 AI 해커톤 - Speech

- 데이터

네이버에서 제공한 100시간의 데이터를 사용. 20년 4월에 ClovaCall이라는 이름으로 데이터셋을 확장하여 공개함.

ClovaCall: Korean Goal-Oriented Dialog Speech Corpus for Automatic Speech Recognition of Contact Centers (Interspeech 2020)

ClovaCall: Korean Goal-Oriented Dialog Speech Corpus for Automatic Speech Recognition of Contact Centers

Jung-Woo Ha^{1*}, Kihyun Nam^{1,2*}, Jingu Kang¹, Sang-Woo Lee¹, Sohee Yang¹, Hyunhoon Jung¹, Eunmi Kim¹,

Hyeji Kim¹, Soojin Kim¹, Hyun Ah Kim¹, Kyoungtae Doh¹, Chan Kyu Lee¹, Nako Sung¹, Sunghun Kim^{1,3}

¹Clova AI, NAVER Corp. ²Hankuk University on Foreign Studies

³The Hong Kong University of Science and Technology

* Both authors equally contributed to this work.

- Jung-Woo Ha et al 「ClovaCall: Korean Goal-Oriented Dialog Speech Corpus for Automatic Speech Recognition of Contact Centers」 Interspeech 2020

네이버 AI 해커톤 - Speech

- Feature

40차 MFCC (Mel Frequency Cepstral Coefficient) 사용

Frame_length : 20ms, Frame_shift : 10ms

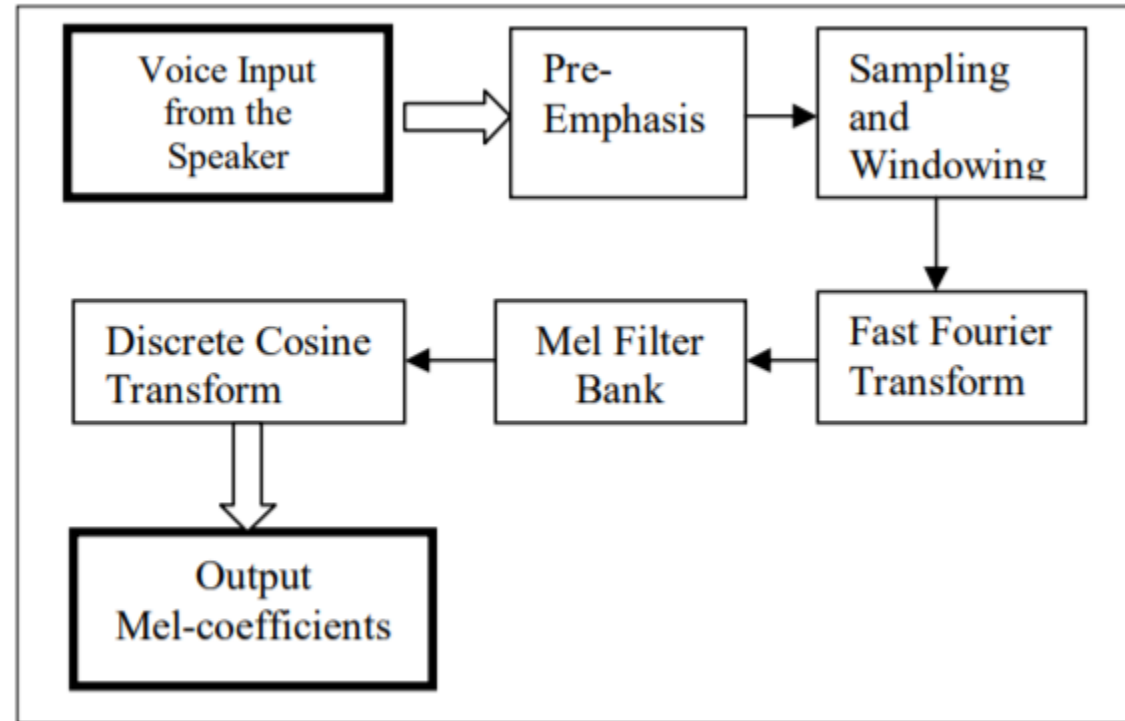


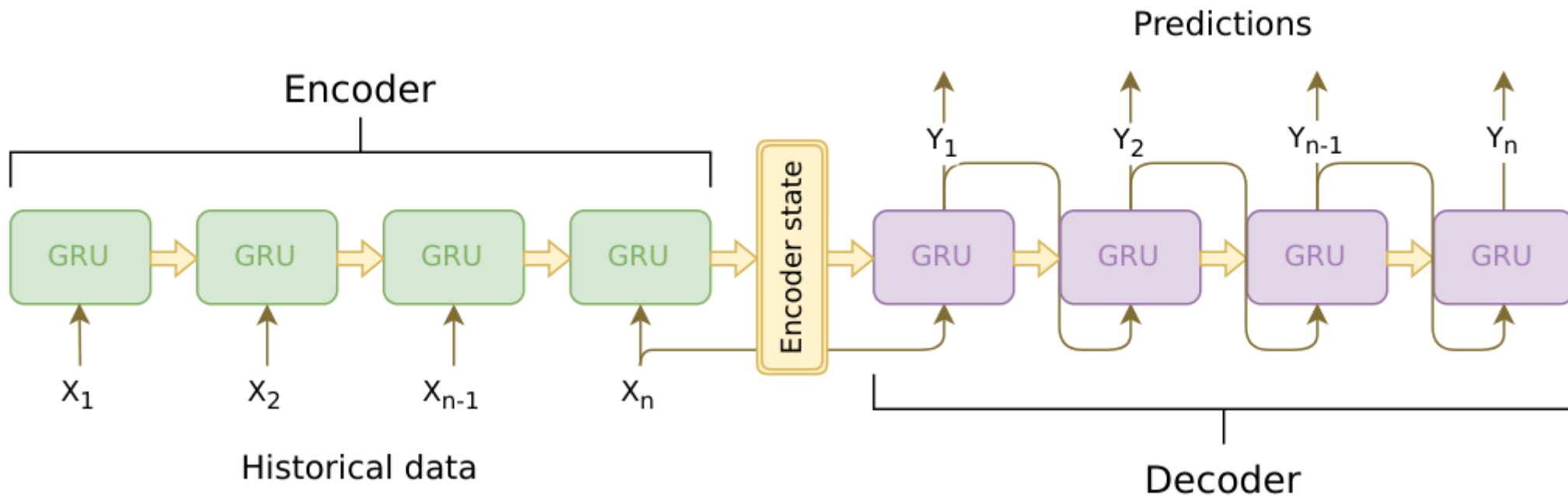
Figure : 「Voice Recognition Using MFCC Algorithm」 IJIRAE 2014

네이버 AI 해커톤 - Speech

■ 모델

어텐션 기반 Sequence-to-Sequence 모델 사용 (RNN type : GRU)

아웃풋으로 문자 단위 (Character) 사용 : 820개의 문자 레이블 사용



• <https://github.com/IBM/pytorch-seq2seq>

네이버 AI 해커톤 - Speech

- Convolution extractor

인코더에 VGG Net을 참고한 컨볼루션 레이어 도입

DeepSpeech2에서 제안한 Convolutional Extractor와 비교하여 인식률 상승

Activation function으로는 DeepSpeech2에서 제안한 $\text{hardtanh}(\text{min}=0, \text{max}=20)$ 사용

Convolution2D (# in = 1, # out = 32, filter = 41×11 , padding = 20×5)

Convolution2D (# in = 32, # out = 32, filter = 21×11 , padding = 10×5)



Convolution2D (# in = 1, # out = 16, filter = 3×3 , padding = 1×1)

Convolution2D (# in = 16, # out = 32, filter = 3×3 , padding = 1×1)

Convolution2D (# in = 32, # out = 64, filter = 3×3 , padding = 1×1)

Maxpool2D (patch = 3×3 , stride = 2×2)

Convolution2D (# in = 64, # out = 128, filter = 3×3 , padding = 1×1)

Convolution2D (# in = 128, # out = 256, filter = 3×3 , padding = 1×1)

Maxpool2D (patch = 3×3 , stride = 2×2)

Dario Amodei et al. Deep Speech2 End-to-End Speech Recognition in English and Mandarin

Referred VGG Net

네이버 AI 해커톤 - Speech

- Hyperparameter Tuning

Seq2seq 구조에서 하이퍼파라미터가 성능에 많은 영향을 미치는 점에 착안하여 하이퍼파라미터 튜닝에 집중







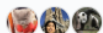



































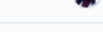





Hyperparameter	Use
attention	dot
num_encoder_layers	4
num_decoder_layers	4
rnn_type	gru
hidden_dim	256
batch_size	32
dropout	0.3
teacher_forcing	0.99
lr	Multi-step
epochs	30

네이버 AI 해커톤 - Speech

■ 최종 성적

75.33% CRR로 100팀 중 12위

학부팀 중 2위

Rank	Name		Score	Recorded	Count	Share
1 -	Morae		85.3494	a day ago	8	 
2 -	blah		82.7253	18 days ago	36	 
3 -	마더판다스		82.1625	a day ago	38	 
4 -	Absolutely		80.7398	2 days ago	25	 
5 -	THE보이스		79.7218	15 hours ago	31	 
6 -	딤마을이장님		79.1653	a day ago	27	 
7 -	하지만어림도없지		78.5267	a day ago	57	 
8 -	SPEECHLESS 4		77.5403	a day ago	52	 
9 -	행복코딩		77.0155	a day ago	60	 
10 -	제안서쓰기싫어		76.9902	19 days ago	44	 
11 -	Oracle		75.9216	a day ago	35	 
12 -	Kai.Lib		75.3272	a day ago	88	 
13 -	IIIIIIII		74.2966	19 days ago	20	 
14 -	매콤쌈무		72.9371	a day ago	10	 
15 -	모두의스피치		72.9244	a day ago	27	 
16 -	52G		71.5144	20 days ago	7	 

네이버 AI 해커톤 - Speech

■ 피드백

대회 상위권 팀로부터 받은 피드백

Clova Speech 헤드엔지니어님 피드백

1. 트랜스포머는 데이터가 적을 시 성능이 좋지 않음
2. 음향모델에 언어모델을 추가적으로 적용해주면 성능이 향상될 것

1등팀 Voithru 스타트업- (85.35%)

1. 빔서치 사용
2. Seq2seq 구조 사용
3. Spec Augmentation 사용
4. 80차 Log Mel-Spectrogram 사용
5. 인코더에 VGGNet을 참고한 컨볼루션 레이어 적용
6. 오버피팅을 방지하기 위해 배치시 서로 다른 주제 및 화자로 배치
7. 레이어 크기를 인코더는 더 깊게 디코더는 더 얇게 적용

2등팀 부산대학교 음성신호처리 연구실 - (82.73%)

1. 빔서치 사용
2. Seq2seq 구조 사용
3. Additive 어텐션 사용
4. Spectrogram 피쳐 사용

3등팀 인하대학교 바이오 IT 연구실 - (82.16%)

1. 앙상블로 인식률 상승
2. Seq2seq 구조 사용
3. 빔서치 사용 후 상당한 인식률 개선
4. Multi-Head 어텐션 사용

KoSpeech Development Process

KoSpeech: Open Source Project for Korean End-to-End Automatic Speech Recognition in Pytorch

데이터

■ 데이터셋

AI Hub에서 공개한 1,000시간의 일상대화 데이터셋 사용 (KsponSpeech)

■ 전처리

- Raw Data

"b/ 아/ 모+ 몬 소리야 (70%)/(칠 십 퍼센트) 확률이라니 n/"

- b/, n/, / .. 등의 잡음 레이블 삭제

"아/ 모+ 몬 소리야 (70%)/(칠 십 퍼센트) 확률이라니"

- 제공된 (철자전사)/(발음전사) 중 발음전사 사용

"아/ 모+ 몬 소리야 칠 십 퍼센트 확률이라니"

- 간투어 표현 등을 위해 사용된 '/', '*', '+' 등의 레이블 삭제

"아 모 몬 소리야 칠 십 퍼센트 확률이라니"

- <https://github.com/sooftware/KsponSpeech-preprocess>

id	char	freq
0		5777462
1	.	640924
21	,	152938

Case	Ground Truth	Prediction
1	아 몬 소리야, 그건 또,	아 원소리야 그건 또
2	아, 아, 아, 그렇지	아 아 아 그렇지
3	그럼, 그럼, 그렇구말구,	그럼 그럼 그렇고말고

데이터

데이터셋 분석

AI Hub 데이터셋에서 등장하는 문자 및 등장 횟수 확인 후 1번만 등장하는 문자가 포함된 약 300개 데이터 제외
2,333개의 문자 → 2,036개의 문자

id	char	freq
0		5774462
1	.	640924
2	그	556373
3	이	509291
4	는	374559
.	.	.
2329	갑	1
2330	감	1
2331	각	1
2332	갓	1



id	char	freq
0		5774462
1	.	640924
2	그	556373
3	이	509291
4	는	374559
.	.	.
2032	꼐	2
2033	겪	2
2034	꺠	2
2035	간	2

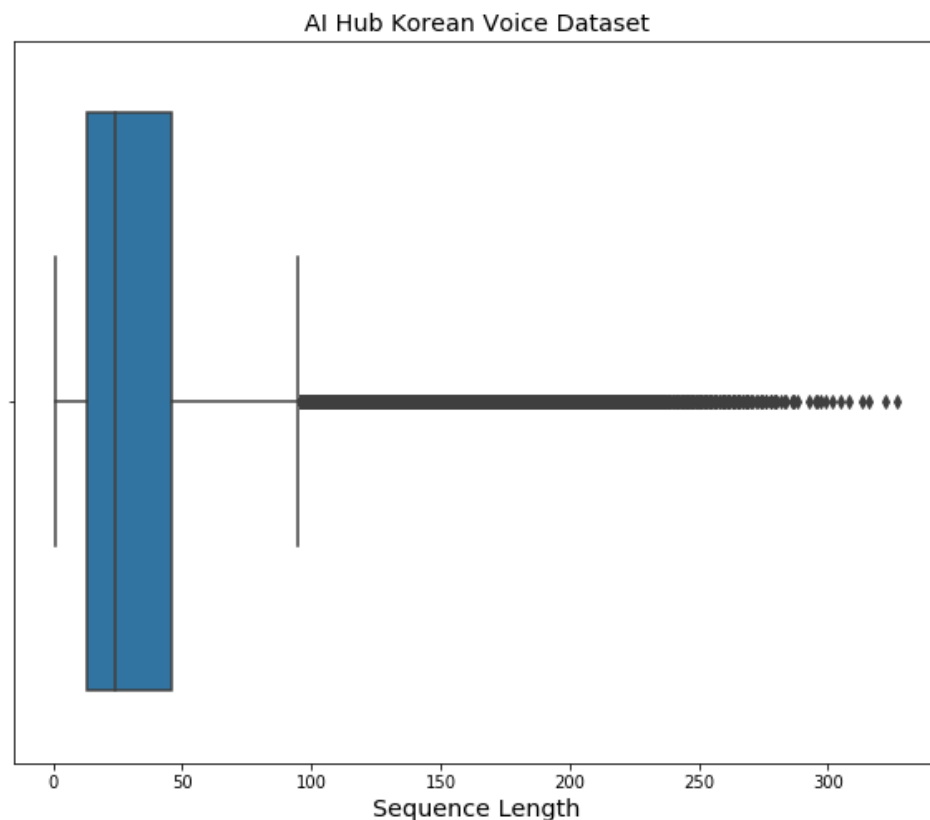
데이터

■ 데이터셋 분석

AI Hub 데이터셋 발화 문자 길이를 Box-Plot을 이용하여 분석 (약 시퀀스 길이 100을 기준으로 아웃라이어 검출)

시퀀스 길이 100 이상 : 28,607개 (전체 데이터의 4.68%)

한정된 GPU 자원에서 Out-Of-Memory를 방지하기 위해 100 이상의 시퀀스 길이를 가진 데이터는 제외하여 학습 및 테스트 데이터 구성



데이터

- Training data

569,938개의 pcm – txt 파일로 구성 (780h)

	audio	label
0	KsponSpeech_268389.pcm	KsponScript_268389.txt
1	KsponSpeech_181280.pcm	KsponScript_181280.txt
2	KsponSpeech_440942.pcm	KsponScript_440942.txt
3	KsponSpeech_360927.pcm	KsponScript_360927.txt
4	KsponSpeech_296731.pcm	KsponScript_296731.txt

- Test data

12,000개의 pcm – txt 파일로 구성 (16h)

	audio	label
0	KsponSpeech_058176.pcm	KsponScript_058176.txt
1	KsponSpeech_056513.pcm	KsponScript_056513.txt
2	KsponSpeech_060597.pcm	KsponScript_060597.txt
3	KsponSpeech_512919.pcm	KsponScript_512919.txt
4	KsponSpeech_068657.pcm	KsponScript_068657.txt

데이터

▪ 피쳐 추출

학습시 옵션을 통해 다양한 방식으로 피쳐 추출 가능

현재까지 성능으로는 Spectrogram이 가장 좋은 성능을 보임 (frame_length 20ms, frame_shift 10ms)

성능 기준 : Spectrogram ≍ Filter Bank > Mel-Spectrogram > MFCC

피쳐 사이즈 :	161	80	80	40
----------	-----	----	----	----

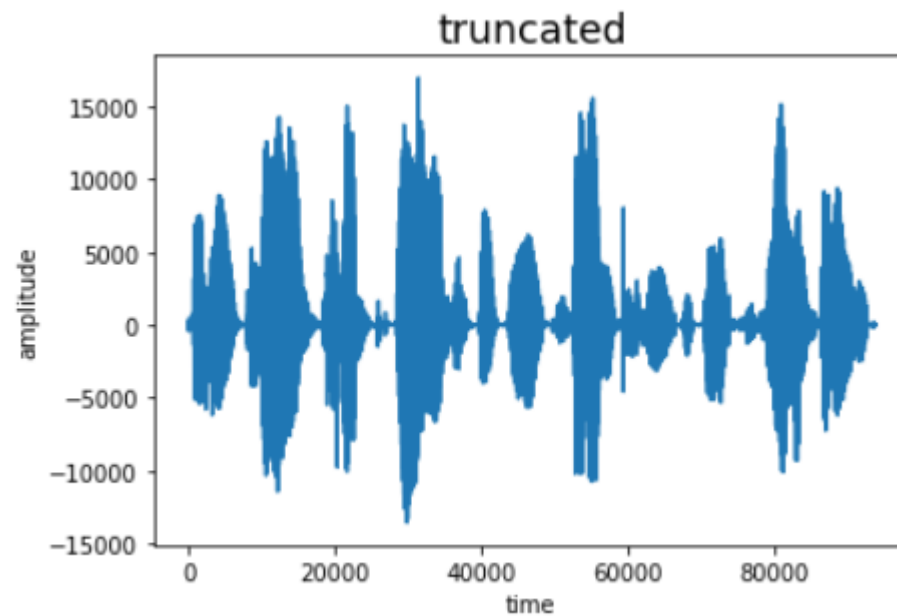
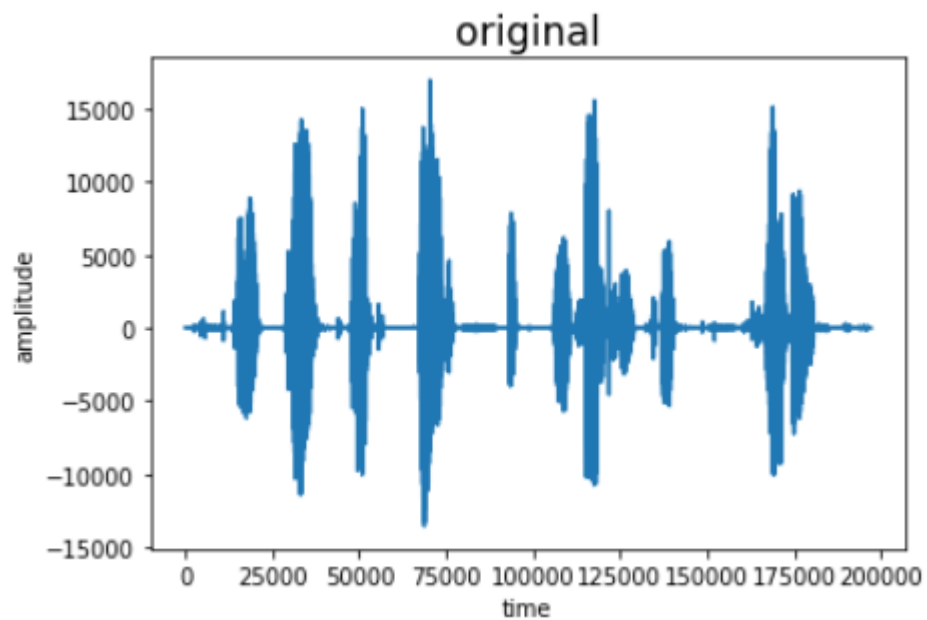
* Options

- **transform_method** : 피쳐 추출 방식 설정 { Spectrogram, Mel-Spectrogram, MFCC, Filter Bank }
- **feature_extract_by** : 피쳐 추출 제공 라이브러리 설정 { librosa, torchaudio, kald, torch }
- **frame_length** : Spectrogram 추출시 프레임 사이즈 설정
- **frame_shift** : Spectrogram 추출시 프레임 쉬프트 사이즈 설정
- **n_mels** : Mel-Spectrogram, MFCC, Filter Bank로 피쳐 추출 시 n차 mel로 뽑을지 설정
- **del_silence** : 오디오 파일에서 silence 구간을 삭제할지 여부를 설정
- **input_reverse** : 들어온 인풋 오디오를 reverse 할지 여부를 설정
- **normalize** : 피쳐의 normalize 여부 설정

데이터

- 침묵 구간 삭제

30dB을 기준으로 침묵 제거 → 인식률 및 학습 속도 향상 (30dB는 실험을 통해 얻은 값)



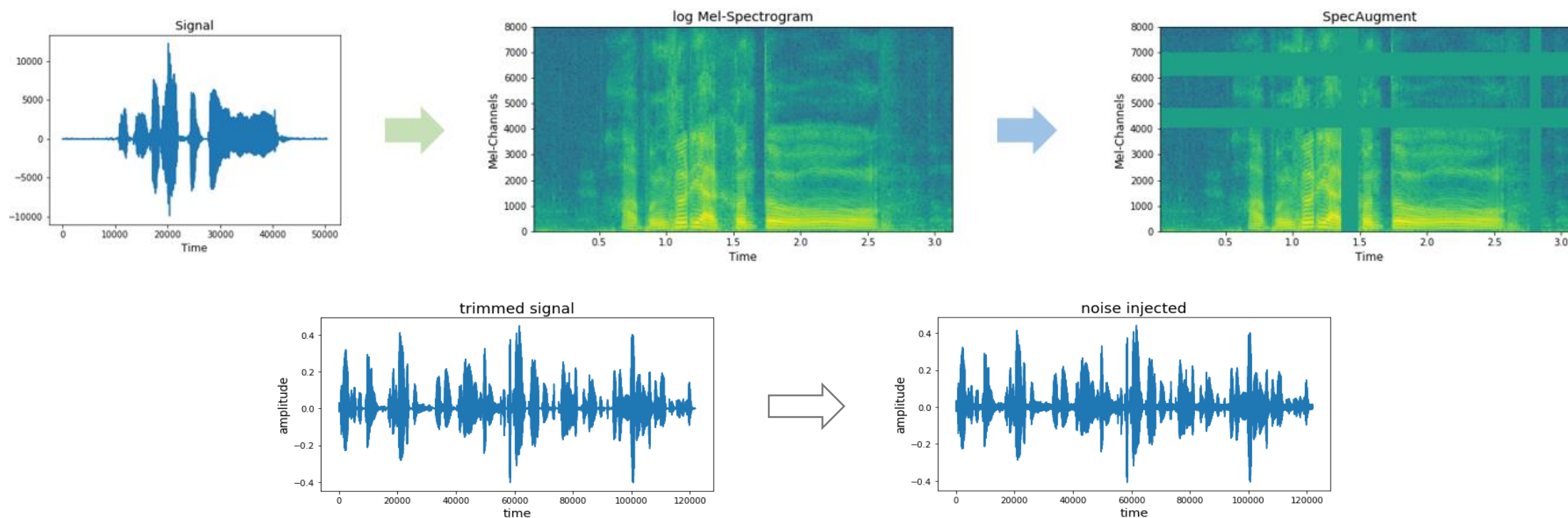
데이터

■ Data Augmentation

SpecAugment & Noise Augment 적용

SpecAugment에서 연산량에 비해 효과가 작은 Time-warping 기법을 제외한 Time-masking, Frequency-Masking 기법 적용

SpecAugment를 실시간으로 적용하여 매 에폭 다른 데이터가 들어감으로써 언더피팅 유발

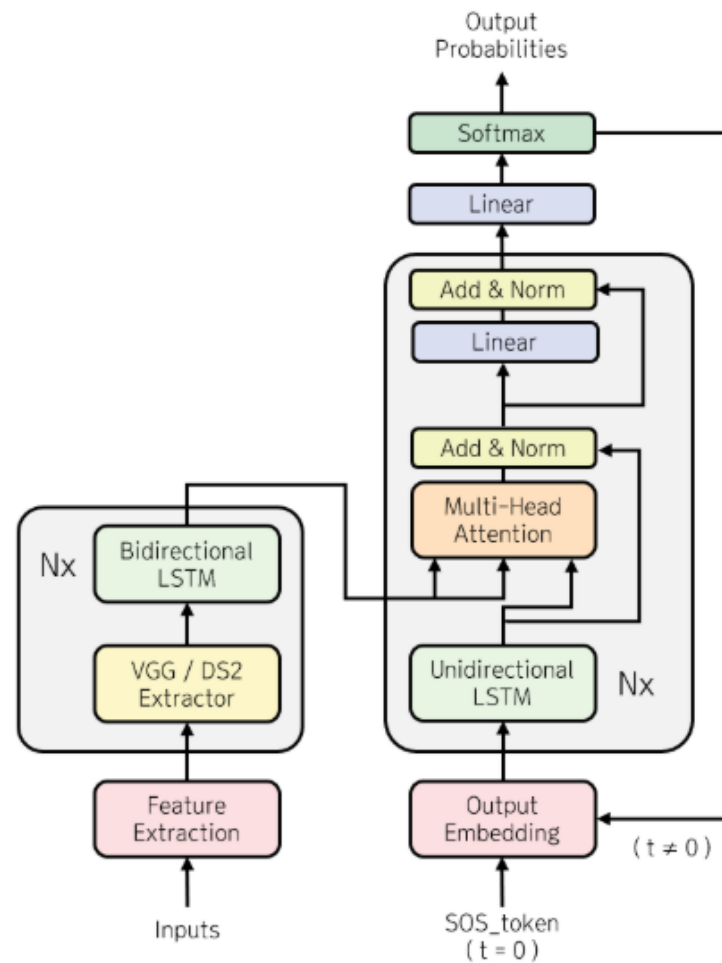


모델

▪ Acoustic model

- Sequence-to-Sequence 구조 기반
- Bidirectional encoder
- CNN Extractor
{ VGG, DeepSpeech2 }
- 어텐션
{ Multi-Head, Location-Aware, Additive, Scaled-dot }
- Skip Connection
- 디코딩 : Greedy / Beam Search

- Willian Chan et al, 「Listen, Attend and Spell」 ICASSP 2016
- Ashish Vaswani et al 「Attention Is All You Need」 NIPS 2017
- Chiu et al 「State-Of-The-Art Speech Recognition with Sequence-to-Sequence Models」 ICASSP 2018
- Daniel S. Park et al 「SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition」 Interspeech 2019



모델

▪ CNN Extractor

`--extractor` 옵션으로 Deep Speech2, VGG Extractor 중 선택 가능

VGG Extractor가 인식을 면에서 더 나은 성능을 보임 (네이버 대회 당시 사용했던 구조보다 더 적은 레이어로 더 높은 성능을 보임)

Activation function으로 Hardtanh(min=0, max=20), ReLU, ELU, GELU, Leaky ReLU 등을 실험

→ Hardtanh(min=0, max=20)이 가장 좋은 성능을 보임

Convolution2D (# in = 1, # out = 32, filter = 41×11 , padding = 20×5)

Convolution2D (# in = 32, # out = 32, filter = 21×11 , padding = 10×5)

Convolution2D (# in = 1, # out = 64, filter = 3×3 , padding = 1×1)

Convolution2D (# in = 64, # out = 64, filter = 3×3 , padding = 1×1)

Maxpool2D (patch = 3×3 , stride = 2×2)

Convolution2D (# in = 64, # out = 128, filter = 3×3 , padding = 1×1)

Convolution2D (# in = 128, # out = 128, filter = 3×3 , padding = 1×1)

Maxpool2D (patch = 3×3 , stride = 2×2)

Dario Amodei et al. Deep Speech2: End-to-End Speech Recognition in English and Mandarin

Takaaki Hori et al. Advances in Joint CTC-Attention based E2E Automatic Speech Recognition with a Deep CNN Encoder and RNN-LM

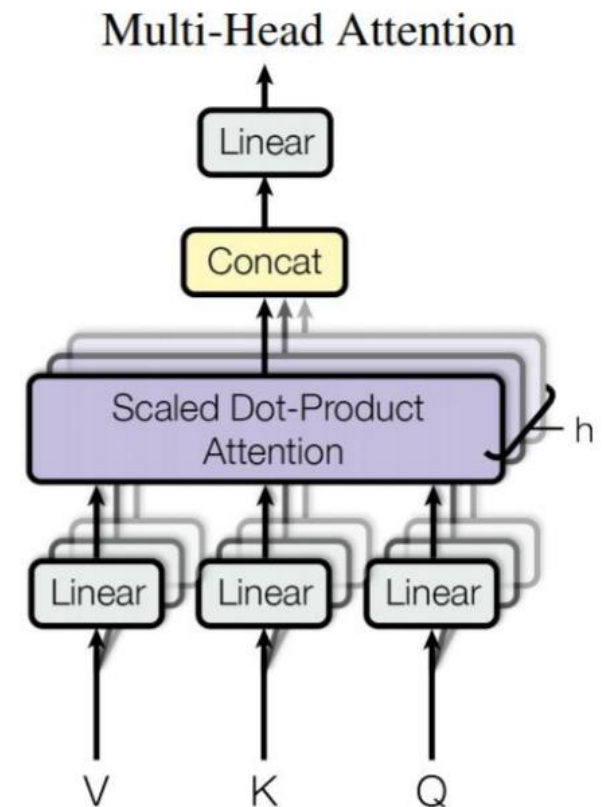
모델

▪ Attention

`--attn_mechanism` 옵션으로 Multi-Head, Location-Aware, Additive, Scaled-dot 어텐션 중 선택 가능
Multi-Head Attention이 가장 좋은 성능을 보임 (# head 4, 8, 16 실험 결과 4가 가장 좋은 성능을 보임)

성능 기준 : Multi-Head > Location-Aware > Additive > Scaled-dot

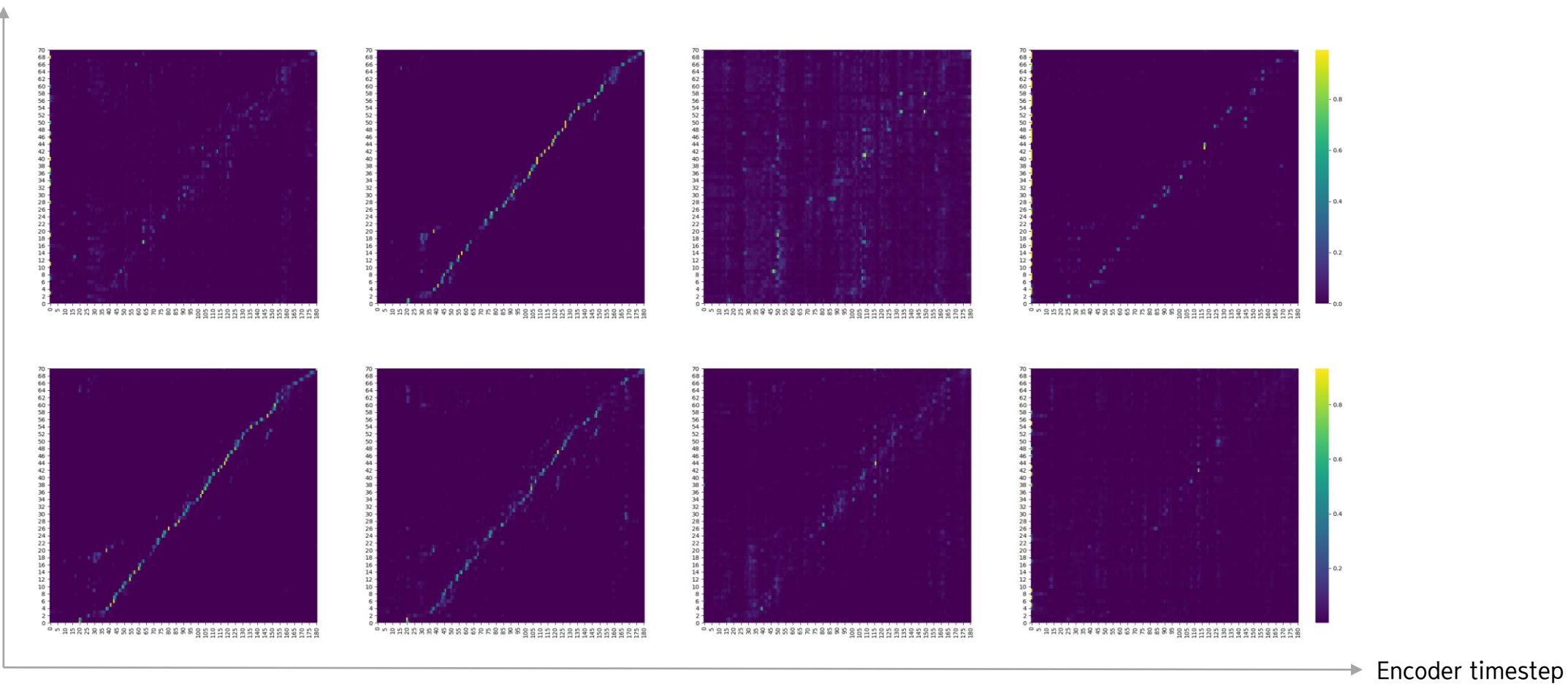
- Bahdanau et al 「Neural Machine Translation by Jointly Learning to Align and Translate」 ICLR 2015
- J Chorowski et al. 「Attention-Based Models for Speech Recognition」 NIPS 2015
- Ashish Vaswani et al 「Attention Is All You Need」 NIPS 2017
- Chiu et al 「State-Of-The-Art Speech Recognition with Sequence-to-Sequence Models」 ICASSP 2018



모델

▪ Attention alignment

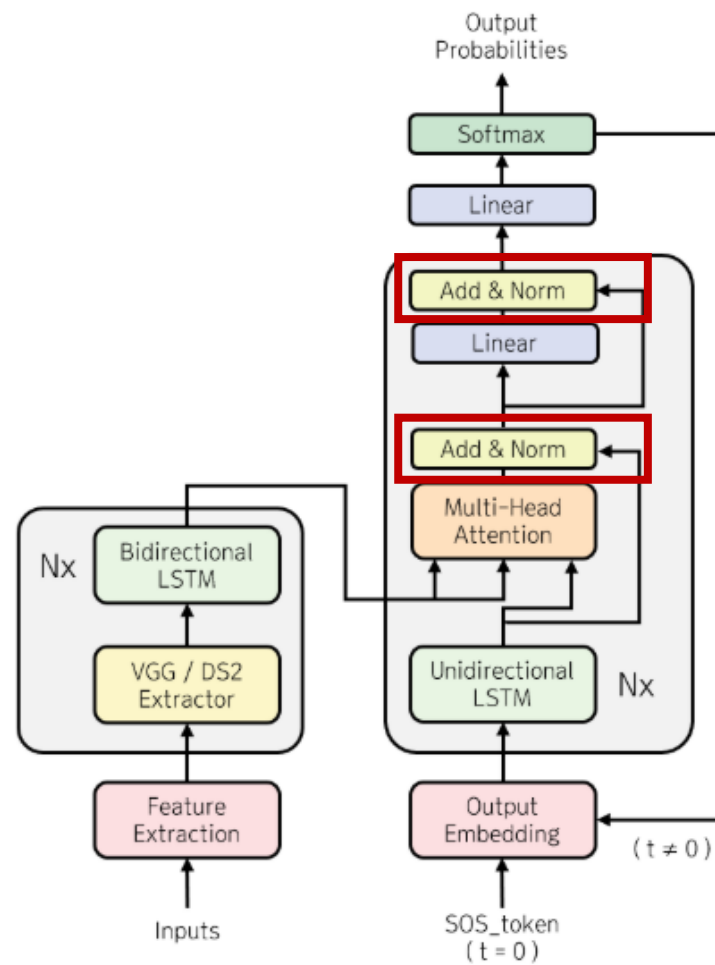
Decoder
timestep



모델

▪ Skip Connection

트랜스포머 구조가 좋은 성능을 보이는 점에 착안하여 적용
적용 전과 비교하여 약간의 높은 성능을 보임



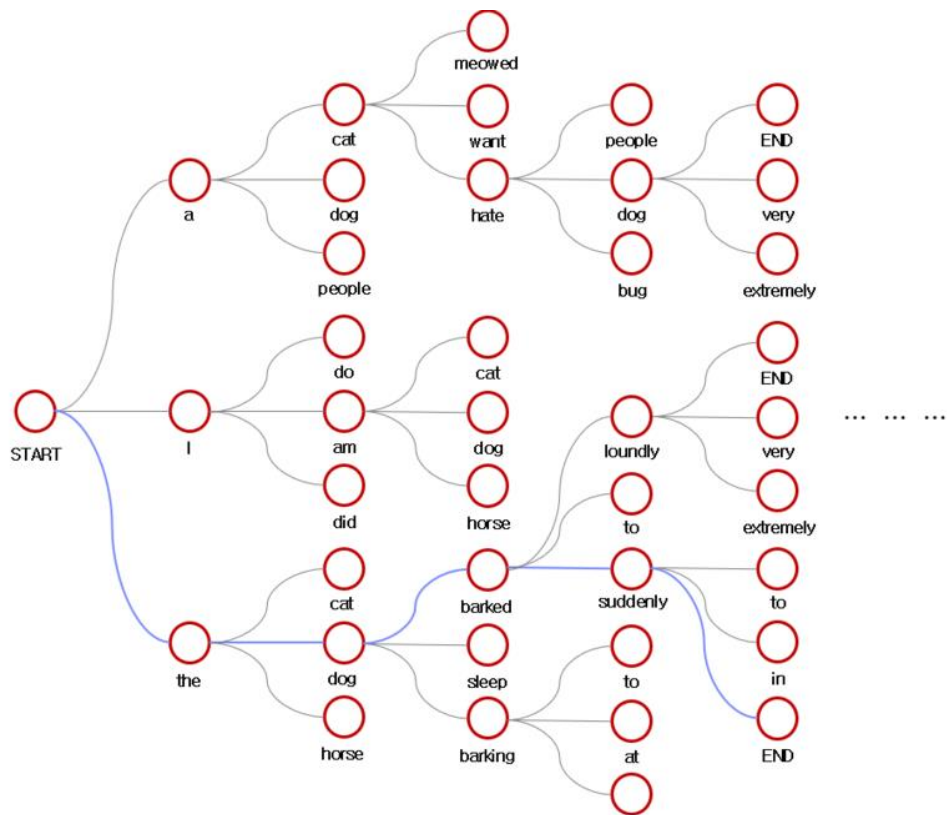
- Ashish Vaswani et al 'Attention Is All You Need' NIPS 2017

모델

▪ Beam Search

일반적으로 그리디 서치보다 높은 인식률을 보이는 것으로 알려진 빔서치 적용

➔ 적용 결과 약 2 – 3% 정도의 성능 저하를 보임



성능 저하 이유 추측

➔ 해외 논문 및 네이버 대회 당시와 비교했을 때 원인은
2,000개가 넘는 클래스 (Character) 수 때문이라고 추측

TO DO

➔ 예측 단위를 Grapheme (자소) 로 하여 클래스 수를 줄여서 빔서치 적용

모델

Greedy Search vs Beam Search

그리디 서치가 약 2-3% 정도 높은 인식률을 보임. 두 방식 모두 발음상 유사하게 예측된 것을 확인할 수 있음.

	original	greedy search	beam search
0	<s>엑스레이 찍고	에 그래 찍고	에 그래 고고
1	<s>이게 끊기는 게 어떻게 끊기냐면은 일 이 초 차이로 소리가 안 나오다가 나오거든	이게 끊기는 게 어떻게 큰 기념이냐면은 일 주차이로 소리 같나 나오거든	이게 끊기는 게 어떻게 끊기냐면은 일 위 초 차로 소리가갔 오나다 나오거든
2	<s>어 약간 너 무삭제 이런 거 본 거야?	어 약간 근데 무삭제? 이런 거 본 거야?	어 약간 근데 무삭제 이런 거 뭐가
3	<s>놀 만큼 논 거고 놀고 한 거지	놀만큼 농구가 또 놀고 한 거지	놀만큼 농구거 같애 놀고
4	<s>진짜 아니더라 그건	아 진짜 아니더라 그건	진 진짜 아니더라 그
5	<s>아빠 내가 다시 전화할게	아빠 내가 다시 전화할게	아빠 내가 다시 전화 게게
6	<s>수시면은 옛날 고등학교 성적으로 하는 거야?	수시면은 옛날 고등학교 성적으로 하는 거야?	수시면 옛날 고고학교 성적으로 하는 거야?
7	<s>으럼 차에서 역소 노래 나오면 어? 그면서 그러고 너 나 블랙핑크는 나도 군대...	응 차에서 역소놀에다 뭐 그런 거 보면서 그러거든 나도 바로 군대에서 그렇게 안 들...	응 차군서 역소 클래나매아그거는진진짜 러러거 나 나 발리너 그래 나도 대에서...
8	<s>나 진짜 어떻게 고등학교 삼 년 내내 같은 반인데 진짜 그렇게 안 친할 수가 ...	나 진짜 어차피 고등학교 삼 년 내내 같은 반인데 진짜 그렇게 안 친했어 그랬지 나...	나 진짜 어떻게 고 하도 삼 년 내내 같은 반인데 진짜 그렇게 안 친날 수 있고면전...
9	<s>괜찮을지 모르겠어	괜찮겠지 모르겠어	괜찮았지 모르겠어
10	<s>거기랑 나는 대신증권도 쓸 거임	거기랑 나는 대신 준 건 더 절건	거기랑 나는 대신 증권도 찢 건는
11	<s>지금 청소기 집에 있는 게 너무 선 있는 거라 되게 무겁고 불편한데	지금 형 성석이 집에 있는 게 너무 선 있는 거라 되게 무겁고 불편한데	지금 상석이 집에 있는 게 너무 선 있는 거 되게 무겁고 불편한데
12	<s>왜 좋아?	왜 좋아?	왜주 와?
13	<s>야간 스키가 진짜 재밌더라	야간 스케 진짜 재밌더라	야간 스페야진짜짜중재밌더라
14	<s>그 미국 브랜드니까	그 미국 부래 되니까	그 미국 다래는 그간
15	<s>거 원래 이따만한	거 원래 있다만	거 원래 있다만
16	<s>어 그냥 꿀 빨았지 뭐	어 그냥 꿀바랐지 뭐	어 그냥 꿀 빨았지 뭐
17	<s>음 만두 좋지	음 만들 좋지	음 만들 좋 취
18	<s>막 다른 사람들이 물어봤을 때 한 휴 군대 한 이 주 남았다고	막 아는 사람들이 물어봤을 때 한 주 구 년 일 나왔다고	막 하른 사람들이 물어봤을 때 아 나 일 왔다다

84% 정도의 인식률을 기록한 모델

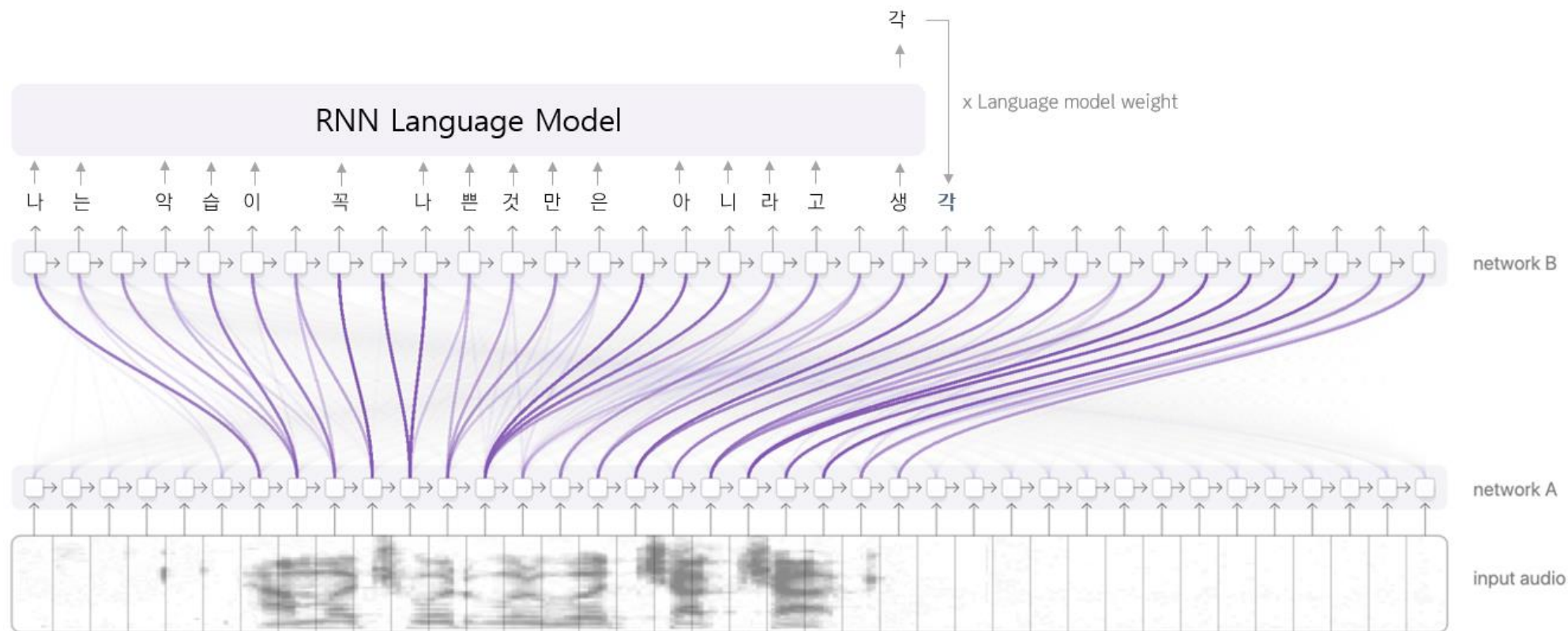
모델

Language Model

Character 단위의 RNN Language Model과 Fusion

Character 단위라는 점을 고려하여 매 타임스텝 모든 이전 예측 값을 Language Model의 입력으로 사용

Fusion 결과 약 2 - 3% 정도의 성능 저하 → KsponSpeech는 구어체인 반면, 언어모델은 문어체로 학습하여 성능 저하를 일으켰다고 판단



모델

- 하이퍼파라미터

현재까지 실험으로 얻은 하이퍼파라미터

Hyperparameter	Use
attention	Multi-Head
encoder_bidirectional	True
rnn_type	lstm
num_encoder_layers	3
num_decoder_layers	2
encoder_hidden_dim	512
decoder_hidden_dim	1024
num_heads	4
cnn_extractor	vgg
cnn_activation	hardtanh(0, 20)
teacher_forcing_ratio	(1.0, 0.8, 0.02)
dropout_p	0.3

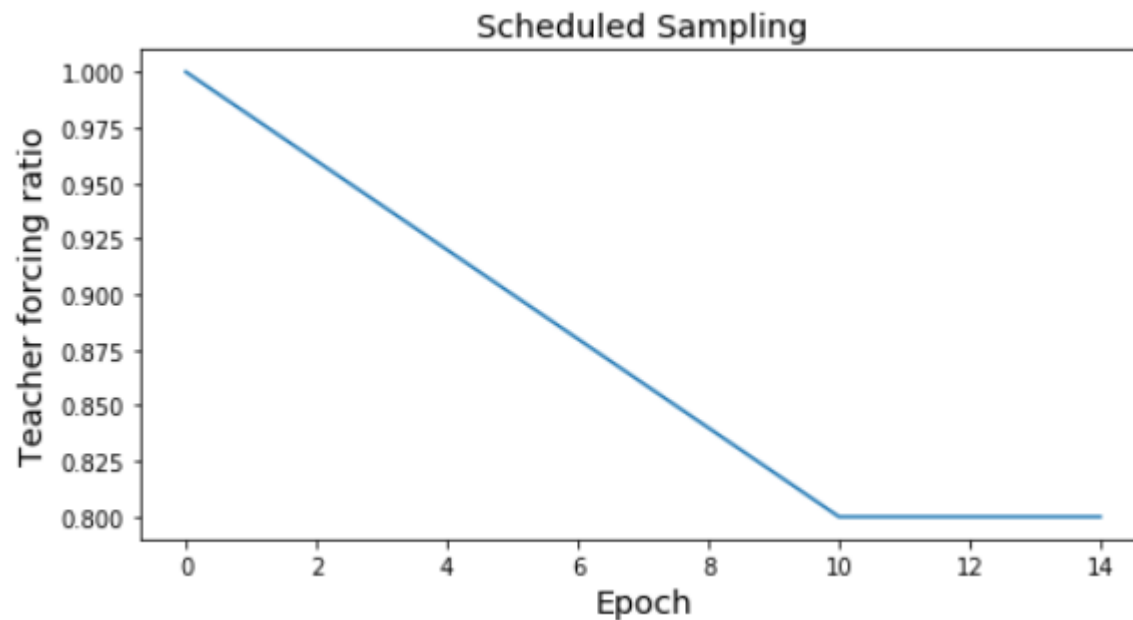
CER : 10.31% (Greedy)

적용 기법

- Scheduled Sampling (Teacher forcing scheduling)

추론 (Inference) 과정에서는 ground truth를 제공받을 수 없으므로, 전 타임스텝의 예측값을 입력으로 예측을 이어가야함
이러한 학습과 추론 단계에서의 차이가 모델의 안정성을 떨어뜨릴 수 있음 (Exposure Bias Problem)

$\text{teacher_forcing_ratio} = \max(1.0 - 0.02 \times \text{epoch}, 0.8)$

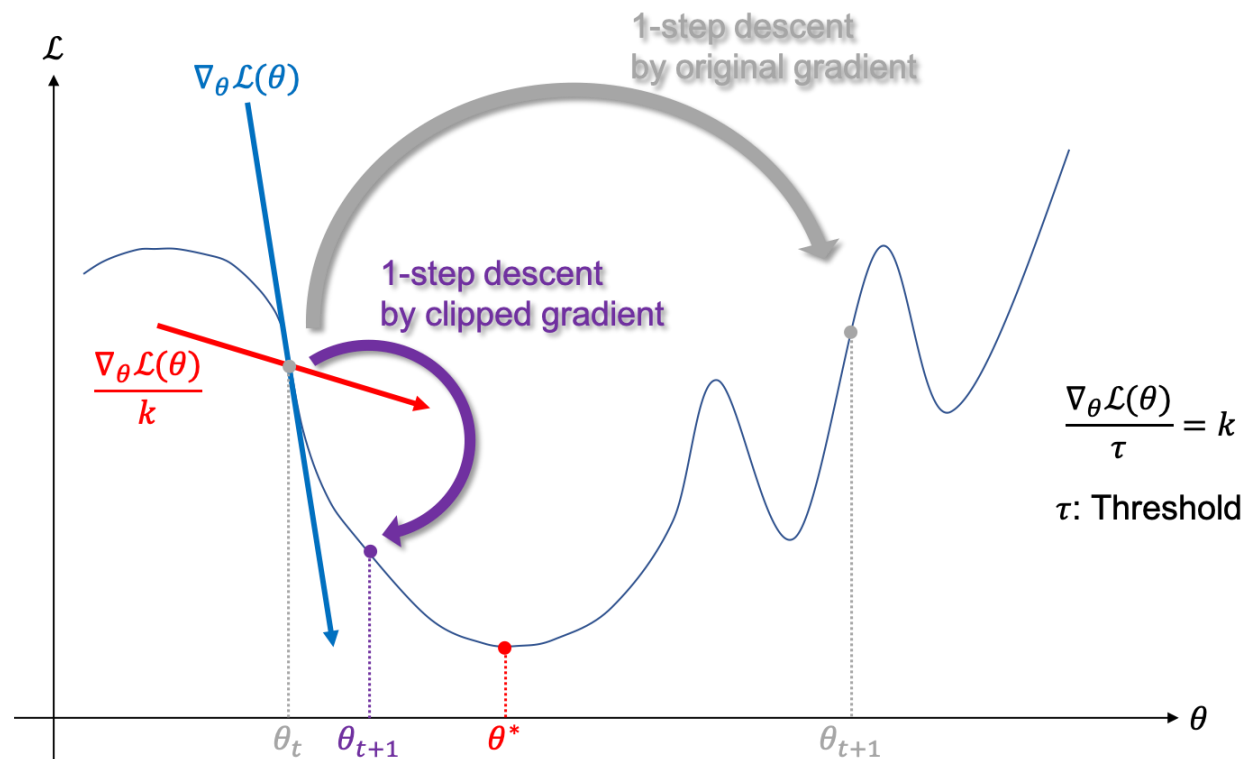


- Chiu et al 'State-Of-The-Art Speech Recognition with Sequence-to-Sequence Models' ICASSP 2018

적용 기법

▪ Gradient Clipping

안정적인 학습을 위해 gradient clipping 기법 적용 ($\tau = 400$)

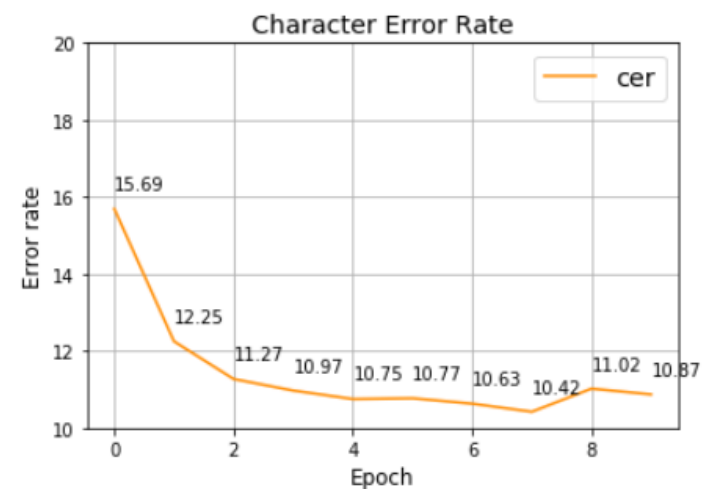
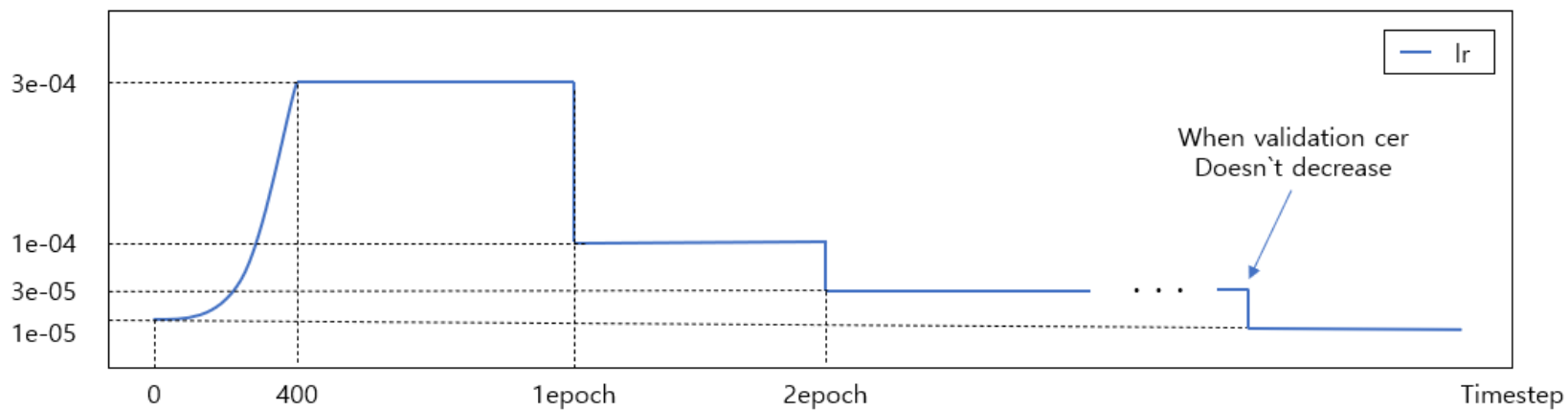


적용 기법

Label Smoothing

Ground truth에 대한 모델의 Over confidence를 방지하기 위해 음성인식의 여러 State-Of-The-Art 모델에 사용된 label-smoothing 기법 적용

Learning rate Scheduling



- Chiu et al. 'State-Of-The-Art Speech Recognition with Sequence-to-Sequence Models' ICASSP 2018
- Daniel S. Park et al. 'SpecAugment : A Simple Data Augmentation Method for Automatic Speech Recognition' Interspeech 2019

To-Do List

KoSpeech: Open Source Project for Korean End-to-End Automatic Speech Recognition in Pytorch

데이터 To-do

- 모든 데이터 사용

한정된 GPU 자원 때문에 사용 못한 시퀀스 길이 100 이상의 데이터를 모두 사용

- 데이터 전처리

일반적으로 잘 사용되지 않는 문자들을 발음이 유사한 빈도수가 높은 문자로 대체하는 방법

- 철자전사 방식 사용

기존에는 발음전사 방식을 사용. 철자전사 방식을 사용하여 실험 후 비교 → 현재 진행중인 음성 번역 프로젝트에 적용 예정

- Raw data

```
"b/ 아/ 모+ 몬 소리아 (70%)/(칠 십 퍼센트) 확률이라니 n/"
```

- Option1 : phonetic

```
"아/ 모+ 몬 소리아 칠 십 퍼센트 확률이라니"
```

- Option2 : numeric

```
"아/ 모+ 몬 소리아 70% 확률이라니"
```

아웃풋 단위 To-do

- 아웃풋 단위로 Grapheme 사용

Grapheme (자소) 으로 예측 단위를 변경할 시, 2,000개가 넘는 클래스에서 49개의 클래스로 줄일 수 있음.

문자 단위와의 성능 비교, 빔서치 적용시 성능 향상 여부

- Character

아 모 문 소리아 진짜

- Grapheme

ㅇㅏ ㅁㅓ ㅁㅓㄴ ㅓㅓㄹ | ㅇㅏ ㅈ | ㄴㅈㅏ

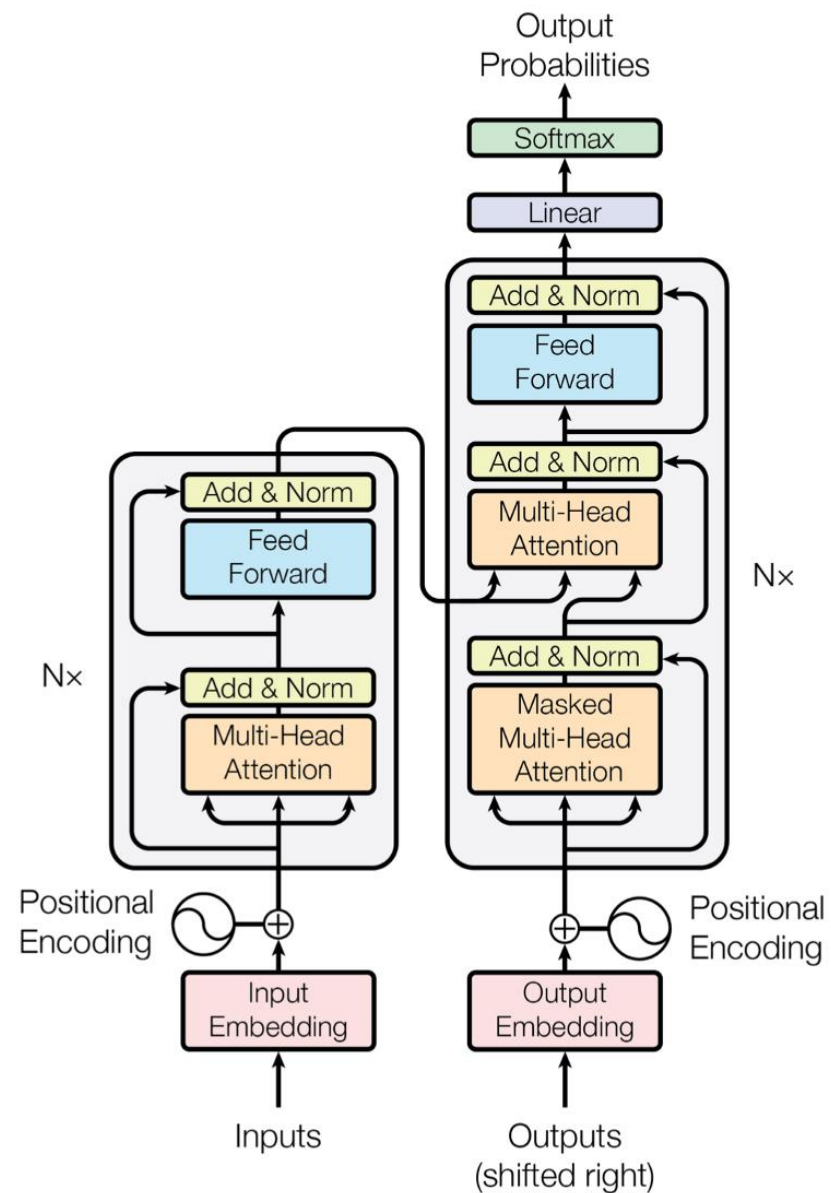
- Hosung Park et al 「Korean Grapheme Unit-based Speech Recognition Using Attention-CTC Ensemble Network」 IEEE 2019

모델 To-do

Transformer

최근 모델 구현 후 적용하였으나, 첫 backpropagation 이후
loss가 nan이 되는 버그가 있어 디버깅 중

➔ 해결 이후, 인코더에 CNN Extractor 추가 후 Seq2seq와 성능 비교



모델 To-do

- Jasper

Convolution Layer + CTC로 WER 2.95%를 기록한 모델

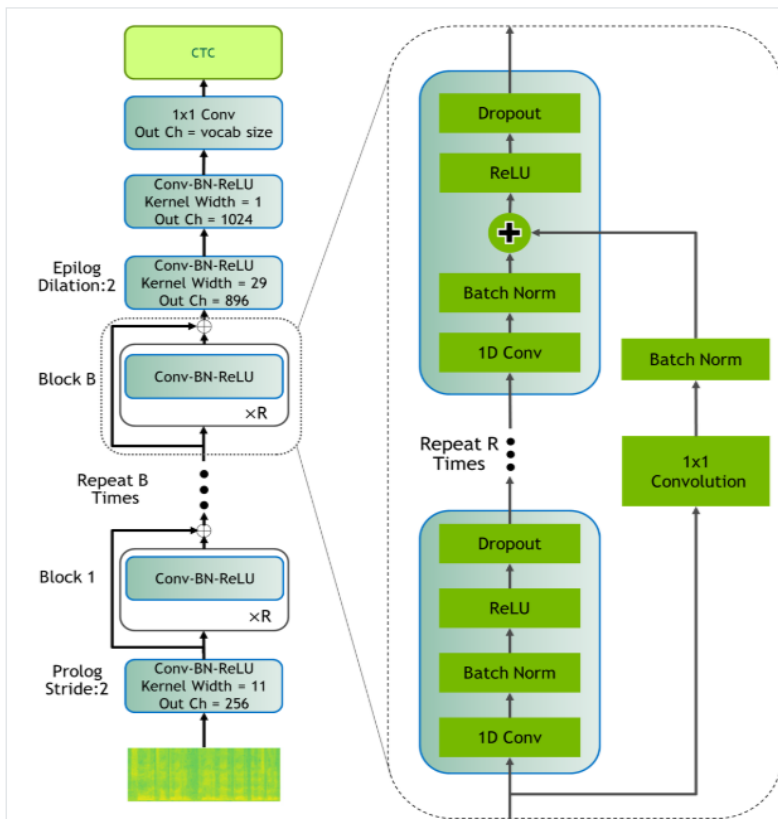


Figure 1: Jasper BxR model: B- number of blocks, R- number of sub-blocks

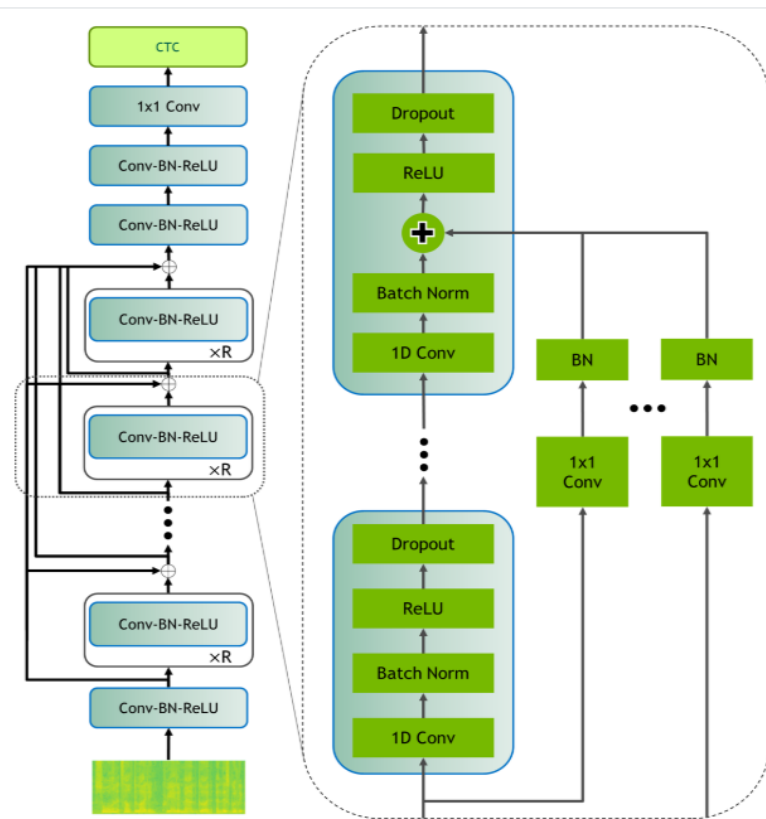


Figure 2: Jasper Dense Residual

Q & A

Thank You