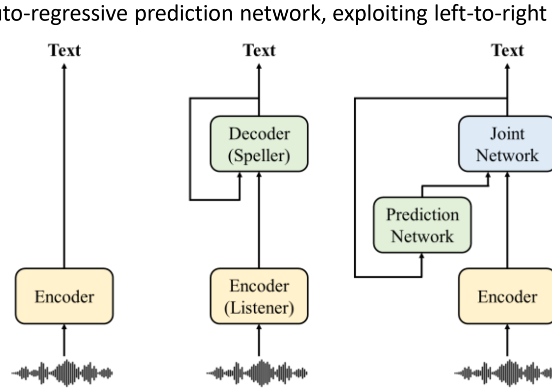
Recent Deep Neural Networks for Automatic Speech Recognition

NPEX 2022 Speech

CTC, LAS, RNNT

- ASR as sequence-to-sequence problem
  - Input length T and output length U are very different.
  - Three types of ASR systems have been introduced: CTC, LAS, RNNT (in order)
  - **CTC**: non-auto-regressive, fastest but needs more beam width
  - **LAS**: auto-regressive decoder, encoder-decoder architecture
  - **RNNT**: auto-regressive prediction network, exploiting left-to-right nature of speech

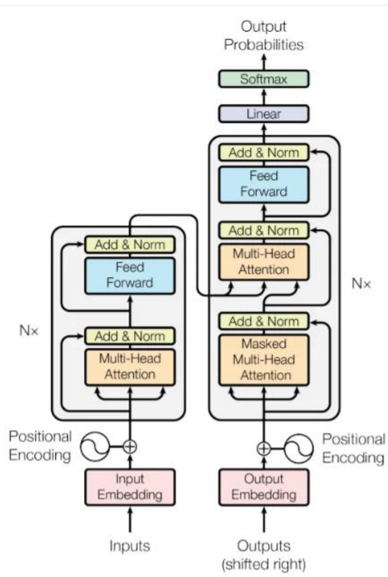


(a) CTC (b) LAS (c) RNNT

지금까지 다룬 Sequence Learning DNN Models

- LSTM RNN – 가장 많이 사용되는 RNN 구조
  - 장점: 비교적 훈련이 RNN 중에서는 잘 된다. 성능이 좋다.
  - 단점: 출력이 한 샘플 씩 계산이 된다. 병렬처리 환경에서 불리하다.
- QRNN – long term memory만 사용하고 short term feedback 을 사용 않는다. 대신에 입력을 추가 처리한다.
  - 장점: 출력을 여러 개 한번에 계산할 수 있다.
  - 단점: 성능 문제가 있다.
- CNN, gated convnet – 정해진 길이의 convolution 을 사용한다.
  - 장점: 출력을 여러 개 한번에 계산할 수 있다.
  - 단점: 좋은 성능을 위해서는 convolution 의 길이를 늘려야 한다. 이 문제를 time-depth-wise 1-D convolution 을 이용해 해결하기도 하였으나 주로 mobile 용으로 사용된다.
- Transformer model – 시간축의 연산을 모두 풀어놓고, 입력사이의 관계를 attention 을 이용해 파악.
  - 장점: 좋은 성능 (과거를 길게 본다), 출력을 여러 개 한번에 계산 (특히 훈련 시에 유리), 잘 훈련이 된다 (LSTM RNN 대비 2, 3배 빨리 훈련된다)
  - 단점: 길이가 길 경우 계산에 불리 (attention은 길이의 제곱 계산), 실시간 모델 만들기 어렵다 (내부 메모리 사용량 많다).

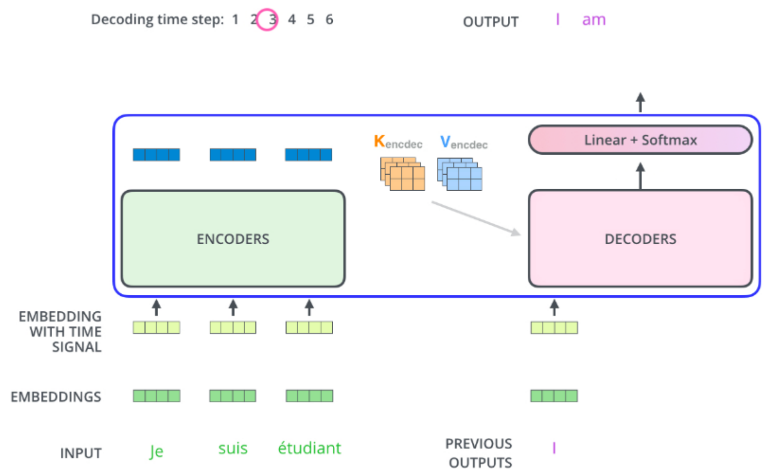
Transformer for Natural Language Processing



02

Attention and Transformers

How it works



But what about Self-attention?

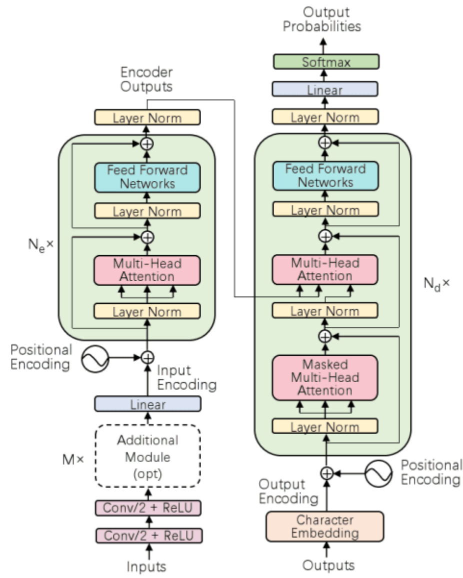
Attention and Transformers

Transformer Results

Table 2: The Transformer achieves better BLEU scores than previous state-of-the-art models on the English-to-German and English-to-French newstest2014 tests at a fraction of the training cost.

Model	BLEU		Training Cost (FLOPs)	
	EN-DE	EN-FR	EN-DE	EN-FR
ByteNet [18]	23.75			
Deep-Att + PosUnk [39]		39.2		$1.0 \cdot 10^{20}$
GNMT + RL [38]	24.6	39.92	$2.3 \cdot 10^{19}$	$1.4 \cdot 10^{20}$
ConvS2S [9]	25.16	40.46	$9.6 \cdot 10^{18}$	$1.5 \cdot 10^{20}$
MoE [32]	26.03	40.56	$2.0 \cdot 10^{19}$	$1.2 \cdot 10^{20}$
Deep-Att + PosUnk Ensemble [39]		40.4		$8.0 \cdot 10^{20}$
GNMT + RL Ensemble [38]	26.30	41.16	$1.8 \cdot 10^{20}$	$1.1 \cdot 10^{21}$
ConvS2S Ensemble [9]	26.36	**41.29**	$7.7 \cdot 10^{19}$	$1.2 \cdot 10^{21}$
Transformer (base model)	27.3	38.1	$\mathbf{3.3 \cdot 10^{18}}$	
Transformer (big)	**28.4**	**41.8**	$2.3 \cdot 10^{19}$	

Transformer-based ASR



Transformer-Transducer

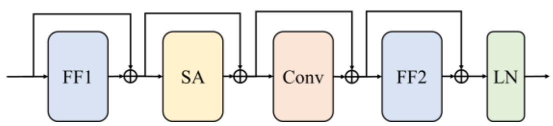
- Transformer encoder for ASR
  - Mostly used with RNNT
  - Much faster training compared to LSTM (Transformer is fully parallelizable)
  - Much slower inference because of $O(T^2)$ computation complexity
- **Transformer** encoder + Transformer transducer
  - 18-layer Transformer encoder + 2-layer Transformer prediction network
  - Relative positioning encoding (RPE) to embed the distance between frames
  - Good for streaming purpose when used with limited left & right context

Table 2. Comparison of WERs for Hybrid (streamable), LAS (e2e), RNN-T (e2e & streamable) and Transformer Transducer models (e2e & streamable) on LibriSpeech test sets.

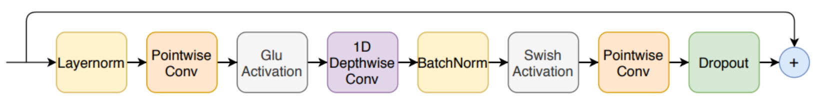
Model	Param size	No LM (%)		With LM (%)	
		clean	other	clean	other
Hybrid [22]	-	-	-	2.26	4.85
LAS[23]	361M	2.8	6.8	2.5	5.8
BiLSTM RNN-T	130M	3.2	7.8	-	-
FullAttn T-T (Ours)	139M	2.4	5.6	**2.0**	**4.6**

Conformer (RNNT)

- Conformer-Transducer
  - A variant of Transformer with additional convolution module
  - Conv module supplements the **locality-aware** characteristics which is often weak in Transformer-based models
  - Time-domain 1D convolution
  - Fewer layers than CNN-based models
  - Fewer parameters than LSTM-based models
  - Used for many state-of-the-art ASR models



Conformer layer



Conv module inside Conformer layer

Conformer (CTC, LAS)

- Conformer also shows superior performance with CTC and LAS frameworks

Table 1. CER/WER results on various open source ASR corpora. Both Transformer and Conformer models are implemented based on ESPnet toolkit. ∗ marks ESPnet2 results. † and ‡ indicate only w/ speed or only w/ SpecAugment, respectively. § denotes w/o any data augmentation.

Dataset	Vocab	Metric	Evaluation Sets	Transformer	Conformer
AIDATATANG	Char	CER	dev / test	(†) 5.9 / 6.7	**4.3 / 5.0**
AISHELL-1	Char	CER	dev / test	(†) 6.0 / 6.7	(*) **4.4 / 4.7**
AISHELL-2	Char	CER	android / ios / mic	(†) 8.9 / 7.5 / 8.6	**7.6 / 6.8 / 7.4**
AURORA4	Char	WER	dev_0330 (A / B / C / D)	**3.3 / 6.0 / 4.5** / 10.6	4.3 / 6.0 / 5.4 / **9.3**
CSJ	Char	CER	eval{1, 2, 3}	(∗) 4.7 / 3.7 / 3.9	(∗) **4.5 / 3.3 / 3.6**
CHiME4	Char	WER	{dt05, et05}_{simu, real}	(†) 9.6 / 8.2 / 15.7 / 14.5	**9.1 / 7.9 / 14.2 / 13.4**
Fisher-CallHome	BPE	WER	dev / dev2 / test / devtest / evltest	22.1 / 21.5 / 19.9 / 38.1 / 38.2	**21.5 / 21.1 / 19.4 / 37.4 / 37.5**
HKUST	Char	CER	dev	(†) 23.5	(†) **22.2**
JSUT	Char	CER	our split	(†) 18.7	**14.5**
LibriSpeech	BPE	WER	{dev, test}_{clean, other}	2.1 / 5.3 / 2.5 / 5.5	**1.9 / 4.9 / 2.1 / 4.9**
REVERB	Char	WER	et_{near, far}	(†) 13.1 / 15.4	(†) **10.5 / 13.9**
Switchboard	BPE	WER	eval2000 (callhm / swbd)	17.2 / 8.2	**14.0 / 6.8**
TEDLIUM2	BPE	WER	dev / test	9.3 / 8.1	**8.6 / 7.2**
TEDLIUM3	BPE	WER	dev / test	10.8 / 8.4	**9.6 / 7.6**
VoxForge	Char	CER	our split	(§) 9.4 / 9.1	(§) **8.7 / 8.2**
WSJ	BPE	WER	dev93/ eval92	(‡) **7.4 / 4.9**	(‡) 7.7 / 5.3
WSJ-2mix	Char	WER	tt	(§) 12.6	(§) **11.7**

Conformer with LAS (=encoder + decoder)

Dataset	Transformer-CTC	Conformer-CTC
CSJ	6.0 / 4.2 / 4.8	**4.8 / 3.7 / 3.8**
TEDLIUM2	16.7 / 16.6	**9.3 / 8.7**
VoxForge	14.0 / 14.1	**9.2 / 8.4**
WSJ	19.4 / 15.5	**12.9 / 10.9**

Conformer with CTC

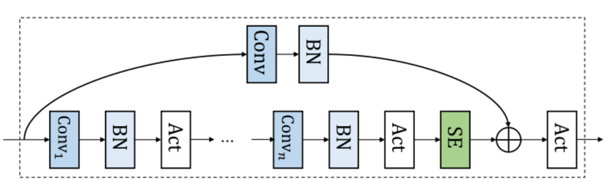
Recent DNN Models for ASR

- DNN for sequence processing
  - Convolutional Neural Networks (CNN)
  - Long Short-Term Memory (LSTM)
  - Transformer / Conformer

Method	#Params (M)	WER Without LM		WER With LM	
		testclean	testother	testclean	testother
Hybrid					
Transformer [33]	-	-	-	2.26	4.85
CTC					
QuartzNet [9]	19	3.90	11.28	2.69	7.25
LAS					
Transformer [34]	270	2.89	6.98	2.33	5.17
Transformer [19]	-	2.2	5.6	2.6	5.7
LSTM	360	2.6	6.0	2.2	5.2
Transducer					
Transformer [7]	139	2.4	5.6	2.0	4.6
ContextNet(S) [10]	10.8	2.9	7.0	2.3	5.5
ContextNet(M) [10]	31.4	2.4	5.4	**2.0**	4.5
ContextNet(L) [10]	112.7	**2.1**	4.6	**1.9**	4.1
Conformer (Ours)					
Conformer(S)	10.3	**2.7**	**6.3**	**2.1**	**5.0**
Conformer(M)	30.7	**2.3**	**5.0**	**2.0**	**4.3**
Conformer(L)	118.8	**2.1**	**4.3**	**1.9**	**3.9**

ContextNet (RNNT)

- CNN encoder for ASR
  - Needs lots of #layers
  - Much fewer #parameters than LSTM
- ContextNet = **CNN** encoder + RNN transducer
  - 107 convolution layers with kernel size = 5
  - STFT with 10ms window → stride 3 times: 80ms per frame
- SE (Squeeze-and-Excite)
  - Adaptively re-weight each feature dimension based on the input
  - Provide global perspective of the entire sequence

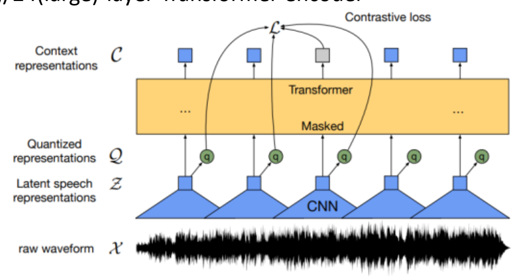


C_i block

Block ID	#Conv layers	#Output channels	Kernel size	Other
C_0	1	$256 \times \alpha$	5	No residual
C_1-C_2	5	$256 \times \alpha$	5	
C_3	5	$256 \times \alpha$	5	stride is 2
C_4-C_6	5	$256 \times \alpha$	5	
C_7	5	$256 \times \alpha$	5	stride is 2
C_8-C_{10}	5	$256 \times \alpha$	5	
C_{11}-C_{13}	5	$512 \times \alpha$	5	
C_{14}	5	$512 \times \alpha$	5	stride is 2
C_{15}-C_{21}	5	$512 \times \alpha$	5	
C_{22}	1	$640 \times \alpha$	5	No residual

Wav2Vec 2.0

- Self-supervised learning
  - Learn useful representations without label
  - Mask parts of waveform and train the model to predict the original values
  - 12(base)/24(large)-layer Transformer encoder



- Fine-tuning for ASR
  - Only 1-hour labeled data shows better result than previous 100-hour case
  - When using full 960-hour (LibriSpeech) data, Wav2Vec pre-training shows much higher recognition performance than training the model from scratch.

Wav2Vec XLSR

- Extend Wav2Vec to learn general speech representations
  - Pre-trained encoder is used for multiple language ASR
  - Pre-trained with 53 languages, achieve SOTA for many low-resource languages

#pre-training languages

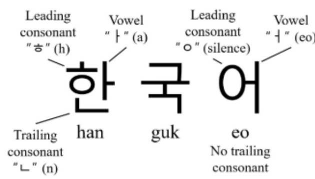
Low-resource languages

Model	D	#pt	#ft	es	fr	it	ky	nl	ru	sv	tr	tt	zh	Avg
Number of pretraining hours per language				168h	353h	90h	17h	29h	55h	3h	11h	17h	50h	793h
Number of fine-tuning hours per language				1h	1h	1h	1h	1h	1h	1h	1h	1h	1h	10h
Baselines from previous work														
m-CPC† (Rivière et al., 2020)	LS_{100h}	10	1	38.7	49.3	42.1	40.7	44.4	45.2	48.8	49.7	44.0	55.5	45.8
m-CPC† (Rivière et al., 2020)	LS_{360h}	10	1	38.0	47.1	40.5	41.2	42.5	43.7	47.5	47.3	42.0	55.0	44.5
Fer et al.† (Fer et al., 2017)	BBL_{all}	10	1	36.6	48.3	39.0	38.7	47.9	45.2	52.6	43.4	42.5	54.3	44.9
Our monolingual models														
XLSR-English	CV_{en}	1	1	13.7	20.0	19.1	13.2	19.4	18.6	21.1	15.5	11.5	27.1	17.9
XLSR-Monolingual	CV_{mo}	1	1	6.8	10.4	10.9	29.6	37.4	11.6	63.6	44.0	21.4	31.4	26.7
Our multilingual models														
XLSR-10 (unbalanced)	CV_{all}	10	1	9.7	13.6	15.2	11.1	18.1	13.7	21.4	14.2	9.7	25.8	15.3
XLSR-10	CV_{all}	10	1	9.4	14.2	14.1	8.4	16.1	11.0	20.7	11.2	7.6	24.0	13.6
XLSR-10 (separate vocab)	CV_{all}	10	10	10.0	13.8	14.0	8.8	16.5	11.6	21.4	12.0	8.7	24.5	14.1
XLSR-10 (shared vocab)	CV_{all}	10	10	9.4	13.4	13.8	8.6	16.3	11.2	21.0	11.7	8.3	24.5	13.8
Our multilingual models (Large)														
XLSR-10	CV_{all}	10	1	7.9	12.6	11.7	7.0	14.0	9.3	20.6	9.7	7.2	22.8	12.3
XLSR-10 (separate vocab)	CV_{all}	10	10	8.1	12.1	11.9	7.1	13.9	9.8	21.0	10.4	7.6	22.3	12.4
XLSR-10 (shared vocab)	CV_{all}	10	10	7.7	12.2	11.6	7.0	13.8	9.3	20.8	10.1	7.3	22.3	12.2
Our Large XLSR-53 model pretrained on 56k hours														
XLSR-53	D_{53}	53	1	**2.9**	**5.0**	**5.7**	**6.1**	**5.8**	**8.1**	**12.2**	**7.1**	**5.1**	**18.3**	**7.6**

State-of-the-art WER

Korean Speech Recognition (Ours)

- Korean ASR
  - Different tokenization (consonant + vowel)
  - Korean is now not a low-resource language; there is a lot of labeled data
  - Ex) Korean Free Conversation (한국어 자유 음성 대화) – 7,600 hours



	#Utterances	#Syllables	Hours(h)
Train	4,000,000	74,797,987	6,089.2
Valid	421,770	7,878,410	640.9
Test	557,564	10,116,858	871.4

  - We achieved WER(word error rate) 6.6%, SER(syllable error rate) 1.46% using 16-layer Conformer encoder with CTC

(a) *Word error rate (WER)*

	Valid (%)	Test (%)
No LM decoding	6.72	9.27
4-gram LM	5.84	8.57
6-gram LM	4.39	7.87
LSTM-LM (w/o SkipTC)	3.94	6.73
LSTM-LM (w/ SkipTC)	**3.75**	**6.59**

(b) *Syllable error rate (SER)*

	Valid (%)	Test (%)
No LM decoding	1.65	2.19
4-gram LM	1.34	1.96
6-gram LM	0.96	1.80
LSTM-LM (w/o SkipTC)	0.87	1.50
LSTM-LM (w/ SkipTC)	**0.80**	**1.46**