Synchronous Transformers for End-to-End Speech Recognition

Zhengkun Tian, *Jiangyan Yi, Ye Bai, Jianhua Tao, Shuai Zhang, Zhengqi Wen*
National Laboratory of Pattern Recognition,
Institute of Automation, CAS
Email: zhengkun.tian@nlpr.ia.ac.cn

Streaming End-to-End ASR

- In order to be truly useful, such end-to-end models must decode speech utterances in a streaming fashion. Streaming ASR can record and recognize almost **synchronously**.

Asynchronous Decoding

For most of attention-based sequence-to-sequence models, the inference process can be divided into two separated stages:

a. **Encoding**

b. **Decoding** (Beam Search)

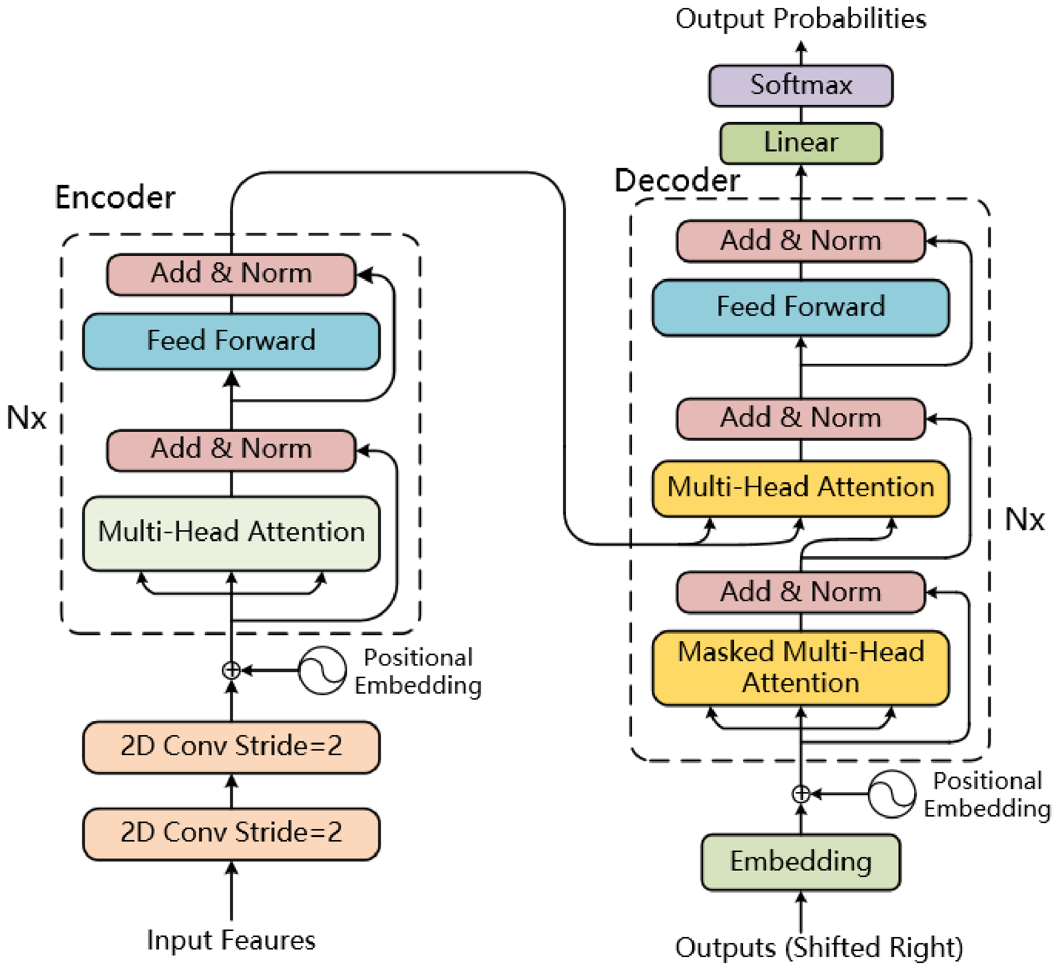
Highlights of Our Work

We proposed a synchronous transformer (Sync-Transformer) model.

- Perform encoding and decoding synchronously.
- Combine the advantages of transformers and transducers in great depth.
- High accuracy and low latency

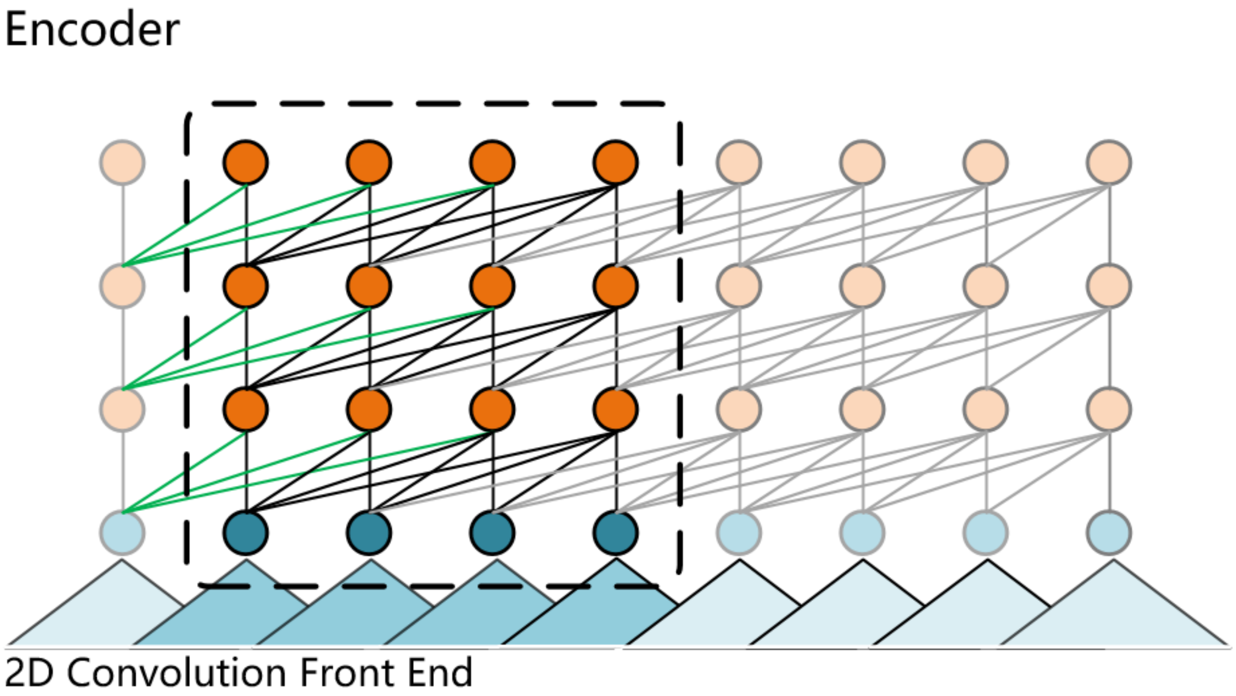
Model Architecture

- Encoder
  - 2 Conv layer with stride 2 (Sub-sampling)
  - 6 blocks
    - Feed Forward Net
    - Multi-Head Attention
    - Layer Norm And Residual Connection
- Decoder
  - 6 blocks
    - Feed Forward Net
    - Multi-Head Attention
    - Layer Norm And Residual Connection
  - Shared Embedding and output linear weights



Model Architecture

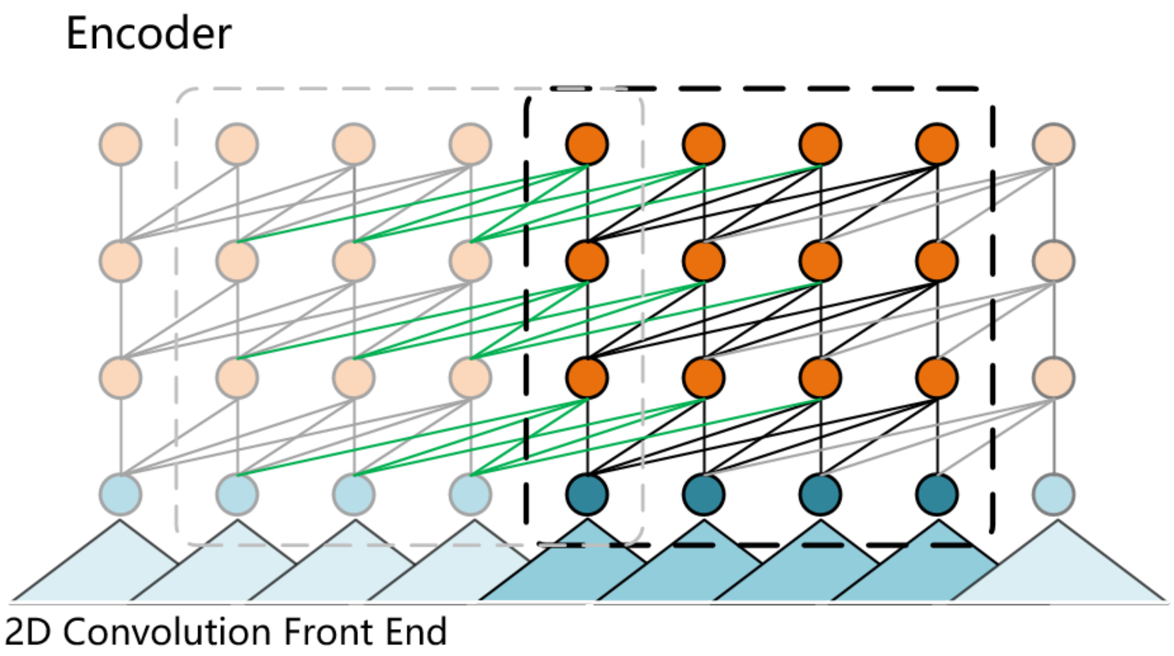
- Local Multi-Head Self-Attention in Encoder
- Every node in the encoder only focus on its left context and ignore its right contexts completely during calculating self-attention weights.



Model Architecture

- Local Multi-Head Self-Attention in Encoder

There is an **overlap** between two adjacent chunks to maintain a smooth transition of information between chunks.



Training

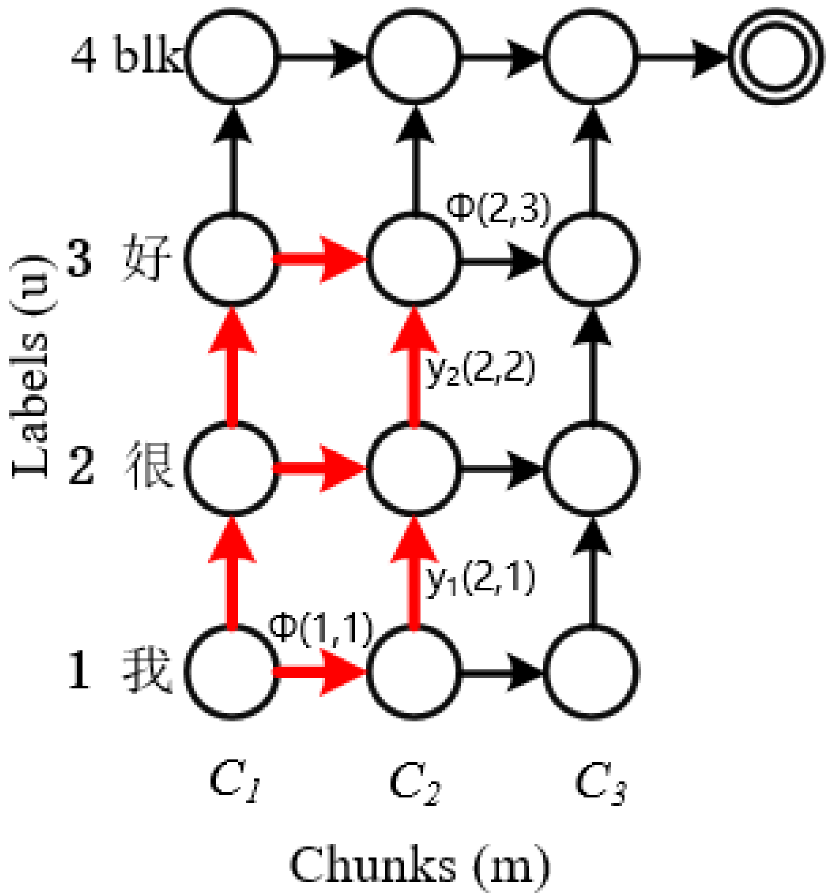
Forward Variables $\alpha(m,u)$

m – the m-th of chunk

u – the u-th of labels

$$\alpha(m,u)=\alpha(m-1,u)\phi(m,u) + \alpha(m,u-1)y_u(m,u-1)$$

$$p(y_{1:U}|x_{1,T})=\alpha(M,U+1)\phi(M,U+1)$$



Training

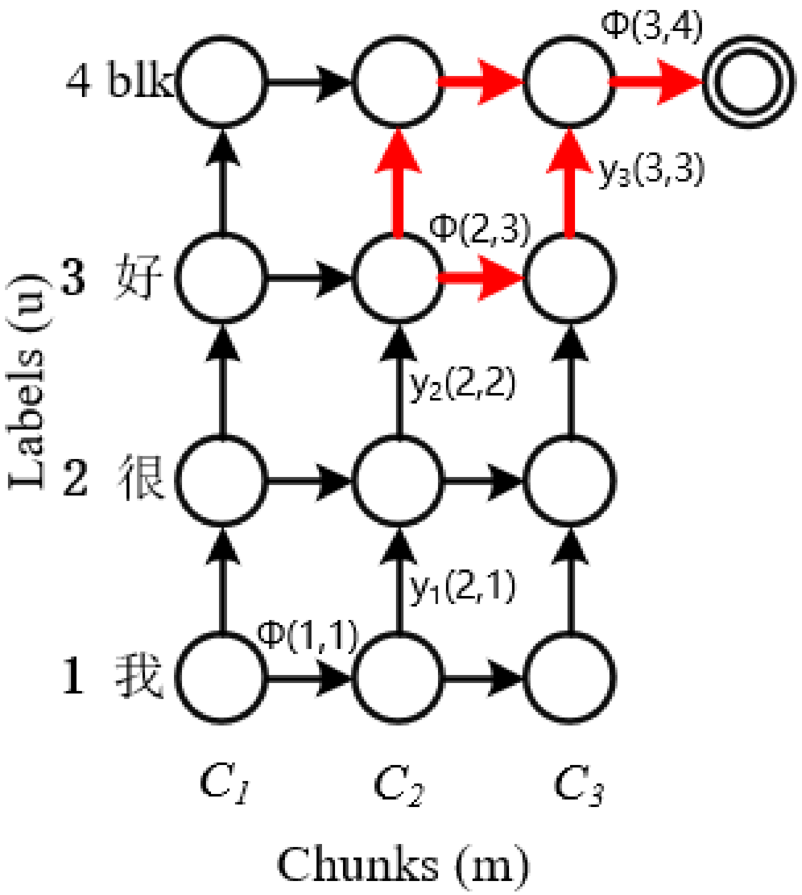
Backward Variables $\beta(m, u)$

m – the m-th of chunk

u – the u-th of labels

$$\beta(m,u) = \beta(m+1,u)\phi(m,u) + \beta(m,u+1)y_{u+1}(m,u)$$

$$\beta(M,U+1) = \phi(M,U+1)$$



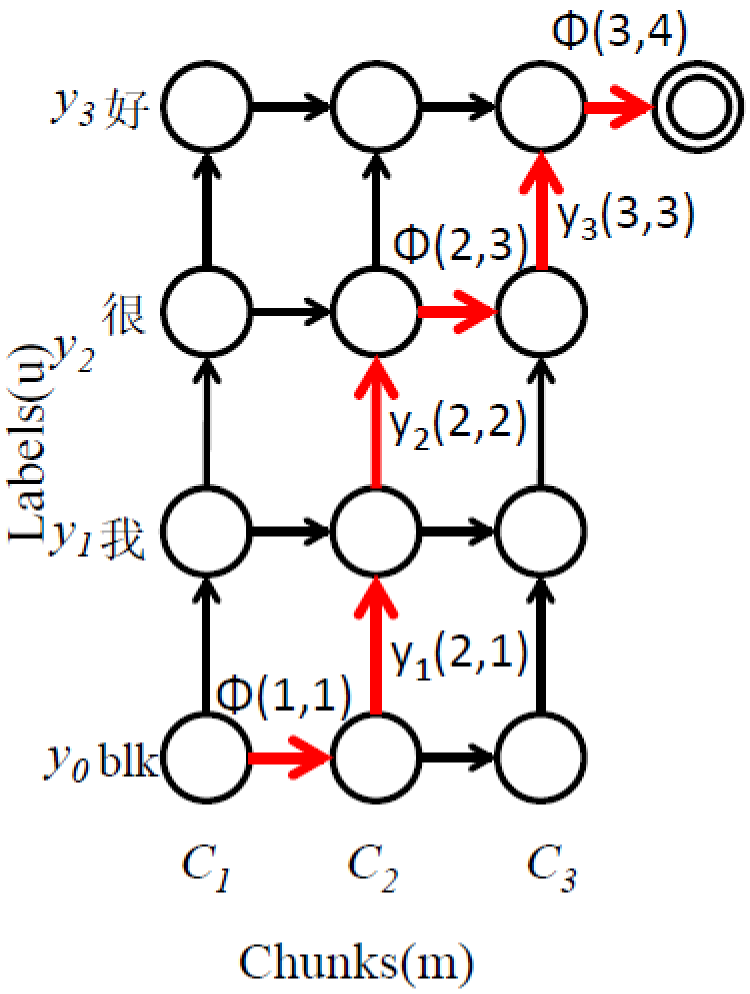
Training

Sum over the probabilities of all alignment paths

$$p(y_{1:U}|x_{1,T}) = \sum_{(m,u):m+u=n} \alpha(m,u)\beta(m,u)$$

Minimize the negative log-loss function

$$\mathcal{L} = -\ln p(y_{1:U}|x_{1,T})$$



Training

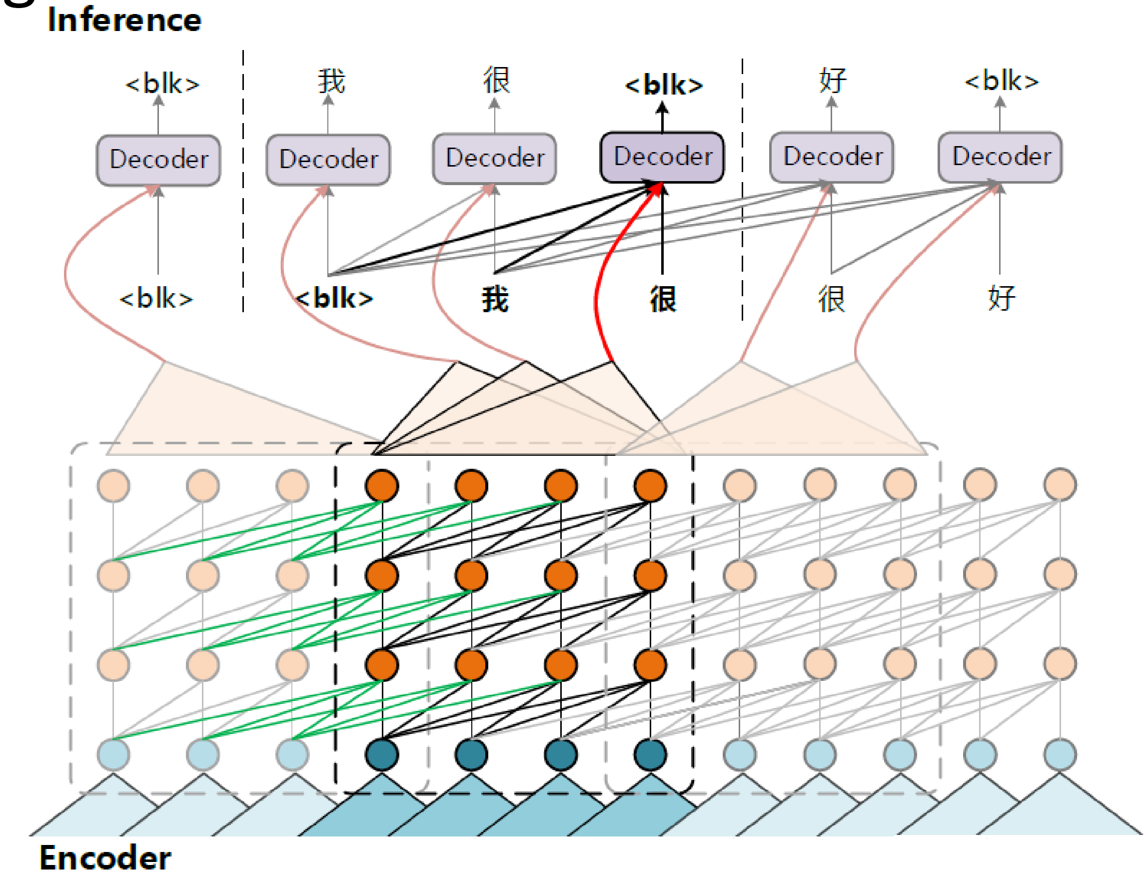
The training process is divided into two steps.

◎ Utilize a trained transformer model to **initialize** the parameters of Sync-Transformer.

◎ Then apply the **forward-backward** algorithm to train a Sync-Transformer.

Inference

- Sync-Transformer decoder an utterance chunk by chunk.
- Once a *<blk>* is predicted, It will switch to the next chunk and continue decoding.



Dataset

- A public Mandarin speech corpus **AISHELL-1**
- Training Set 150 hours / 120098 utterances
- Development 20 hours / 14326 utterances
- Test set 10 hours / 7176 utterances

Experiments Setup

- Encoder
  - 2 layer conv layer front end (stride 2, channels 256 and kernel size 3)
  - 6 blocks /d_{model} 256 / d_{ff} 1024
  - Left context length 20 and right context length 0
- Decoder
  - 6 blocks /d_{model} 256 / d_{ff} 1024
  - Share the weights of embedding and output linear layer
  - 4232 characters as model units (including a *<blk>* and a *<unk>*)
- Training And Inference
  - First stage: 60 epochs   Second stage: 10 epochs
  - Beam Width: 5

Experiments

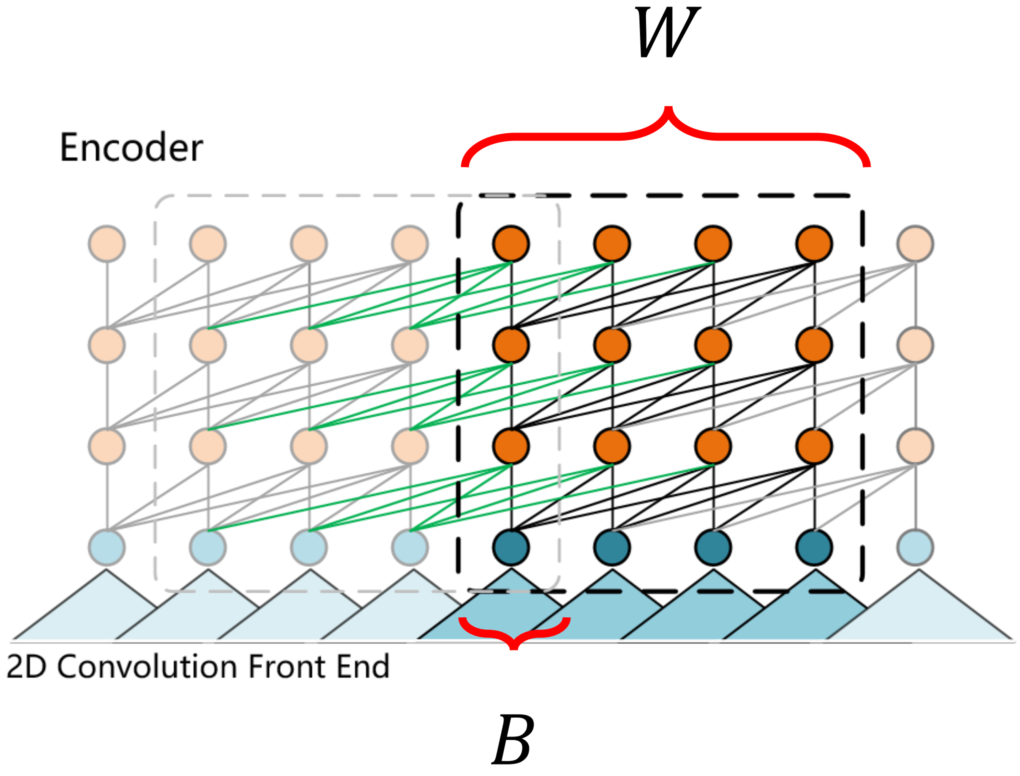
- Comparison of different window lengths and overlap lengths

Table 1. Comparison of different window lengths (CER %).

W	**5**	**10**	**15**	**20**	**25**
Dev	8.64	**7.99**	8.57	8.68	11.04
Test	9.73	**9.06**	9.51	9.76	11.71

Table 2. Comparison of different overlap lengths (CER %).

B	**4**	**3**	**2**	**1**	**0**
Dev	8.60	**7.91**	7.99	9.53	9.61
Test	9.56	**8.91**	9.06	10.39	10.47



Experiments

- Comparison with other end-to-end models

Table 3. Comparisons with other models (CER %).

Model	Online	Steps	Dev	Test
LAS [20]	No	U	-	10.56
Transformer	No	U	7.80	8.64
RNN-T [10]	No	T+U	10.13	11.82
SA-T [10]	No	T+U	8.30	9.30
Chunk-Flow SA-T [10]	Yes	T+U	8.58	9.80
Sync-Transformer	Yes	U+M	7.91	8.91

U is the length of the target sequence.
T is the number of frames.
M is the number of chunks.

$$U < U + M \ll T < T + U$$

Conclusions

- We proposed a streaming model named synchronous transformer, which combines the advantages of transformers and transducers model in great depth.

- Sync-Transformer can encode and decode synchronously like transducer.

- Sync-Transformer can achieve high accuracy like transformer and low latency.

Thanks

Email: zhengkun.tian@nlpr.ia.ac.cn

WeChat