HAL
open science

A Light Transformer-Based Architecture for Handwritten Text Recognition

Killian Barrere, Yann Soullard, Aurélie Lemaitre, Bertrand Coüasnon

▶ **To cite this version:**

Killian Barrere, Yann Soullard, Aurélie Lemaitre, Bertrand Coüasnon. A Light Transformer-Based Architecture for Handwritten Text Recognition. Reconnaissance des Formes, Image, Apprentissage et Perception (RFIAP) 2022, Jul 2022, Vannes, France. hal-03857723

HAL Id: hal-03857723

https://hal.science/hal-03857723

Submitted on 17 Nov 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Light Transformer-Based Architecture for Handwritten Text Recognition

Killian Barrere Yann Soullard Aurélie Lemaitre Bertrand Coüasnon

RFIAP, Vannes
8th July 2022

IntuiDoc research team, Univ. Rennes, CNRS, IRISA, France

Handwritten Text Recognition

textline image

HTR Model → " The Little Key. "

output text

Transformer for Handwritten Text Recognition

Existing approaches [Kang et al. 2020, Singh et al. 2021]

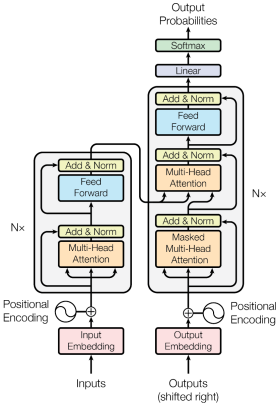
- Transformer layers to model the language
- **Big architectures** to obtain state-of-the-art results

Problem

- Require **a lot of data** to be trained
- Few annotated data in handwritten recognition (10k lines)
- ⇒ **Additional data** to perform well

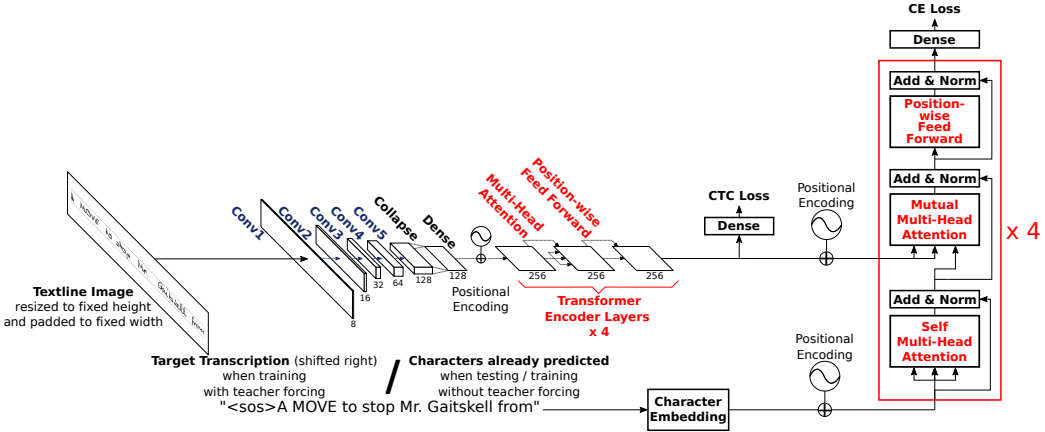
Our proposition

- **Light architecture to perform well with few data**
- Hybrid loss to ease the training

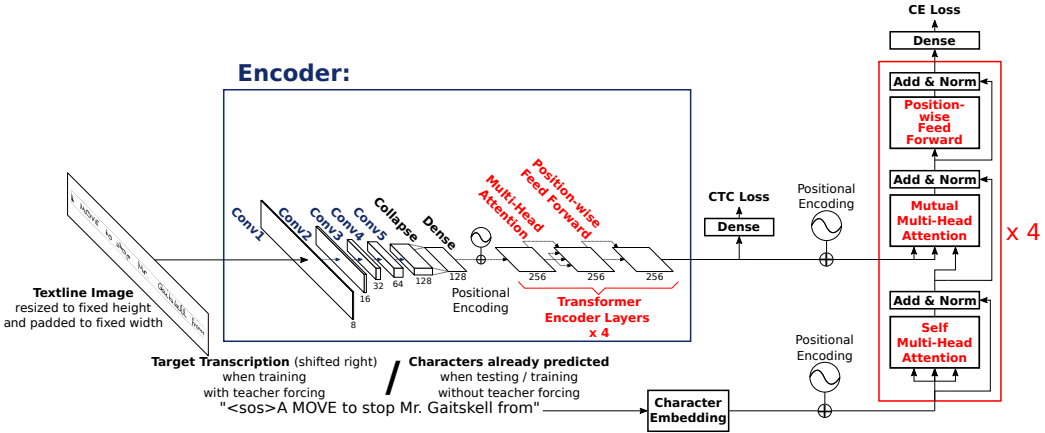


Original Transformer [Vaswani et al. 2017]

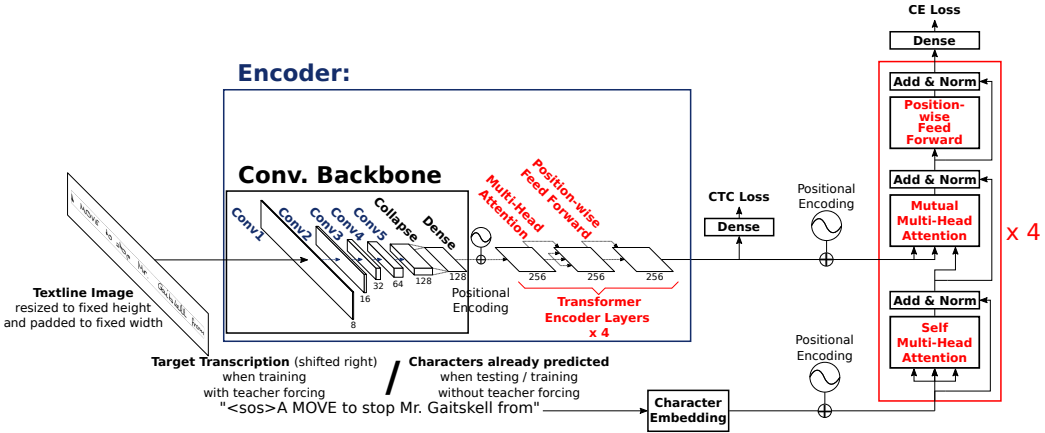
Our Light Transformer Architecture



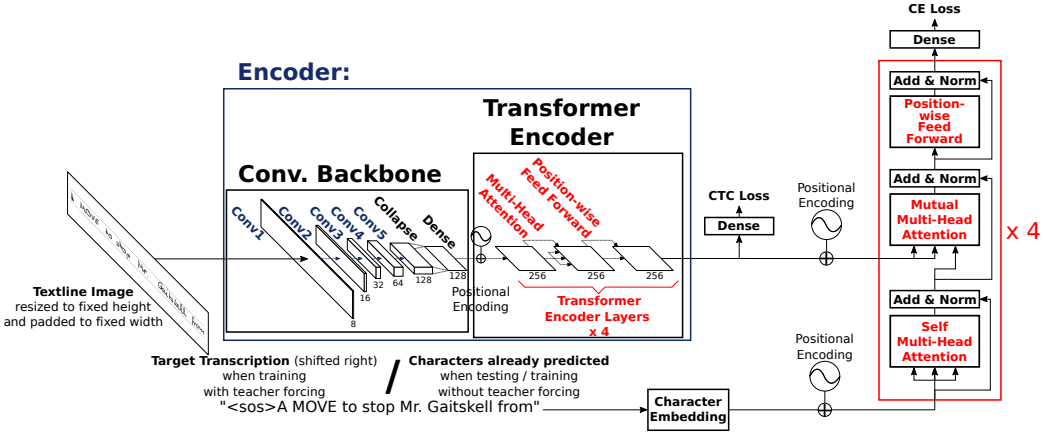
Our Light Transformer Architecture



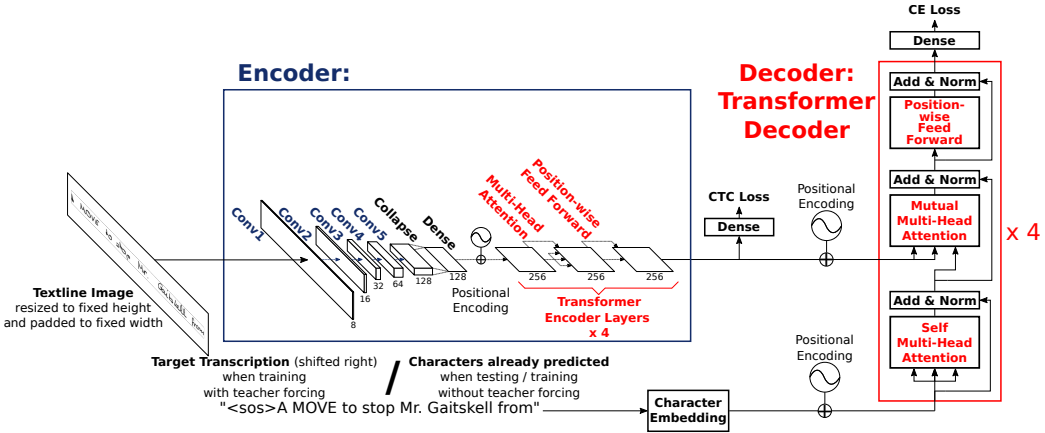
Our Light Transformer Architecture



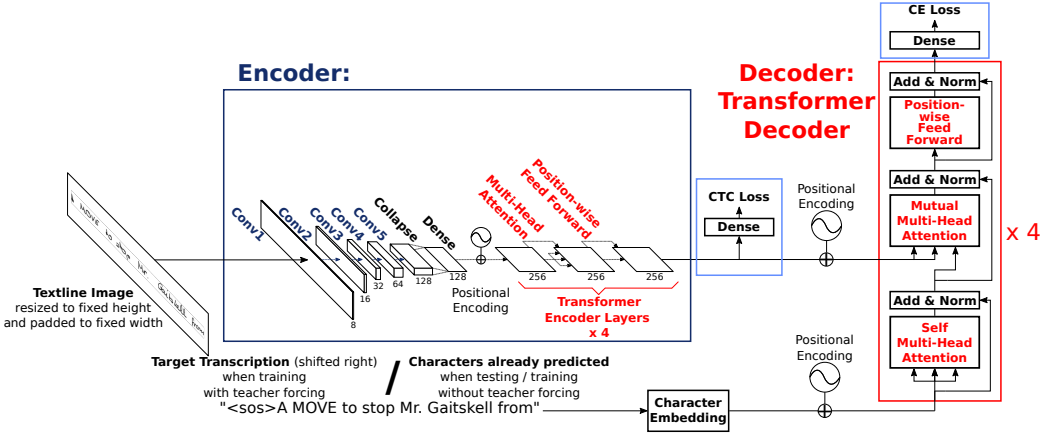
Our Light Transformer Architecture



Our Light Transformer Architecture



Our Light Transformer Architecture



A Light Architecture

How to make a smaller Transformer

Convolutional backbone

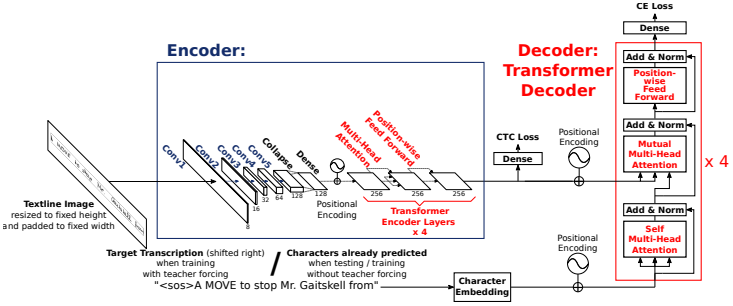
~~Big backbone (i.e. ResNet18)~~

⇒ Only 5 convolutional layers

Neurons in Transformer layers

~~Up to 1,024 neurons~~

⇒ Only 256 neurons



A Light Architecture

How to make a smaller Transformer

Convolutional backbone

~~Big backbone (i.e. ResNet18)~~

⇒ Only 5 convolutional layers

Neurons in Transformer layers

~~Up to 1,024 neurons~~

⇒ Only 256 neurons

In total

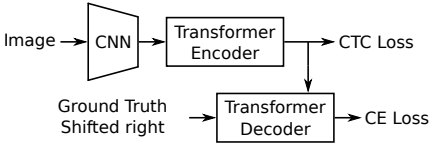
~~100M parameters~~

⇒ **6.9M parameters**

Potential benefits

- **Faster to train** compared to other Transformer-based architecture
- **Does not require additional data** to be trained efficiently

Hybrid Loss



$$\mathcal{L} = \lambda \cdot \mathcal{L}_{CTC} + (1 - \lambda) \cdot \mathcal{L}_{CE}$$

Hybrid loss [Michael et al. 2019]

- Connectionist Temporal Classification (CTC) for the Encoder
- Cross Entropy (CE) for the Decoder

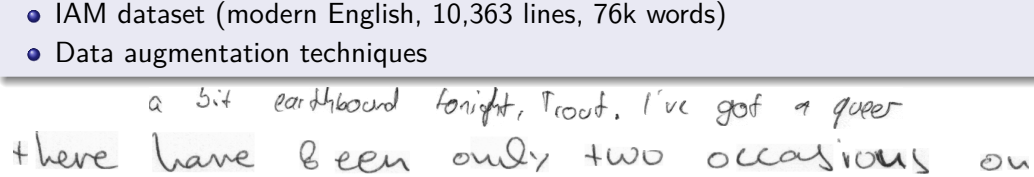
Potential benefits

- Help to train deep layers with gradients from both losses
- Faster convergence

Data used

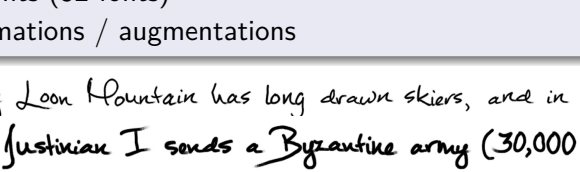
Real data, without additional data

- IAM dataset (modern English, 10,363 lines, 76k words)
- Data augmentation techniques



Our synthetic data (to compare with other transformers)

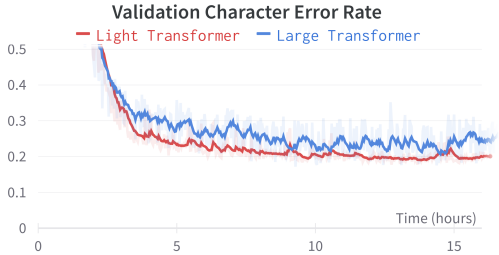
- Articles from Wikipedia (21,350 articles, 66M words)
- Handwritten fonts (32 fonts)
- Random deformations / augmentations



Benefits of Using a Light Architecture

Different sizes of our architecture

- **Light Transformer**: **6.9M params.**
- Large Transformer: 28M params.

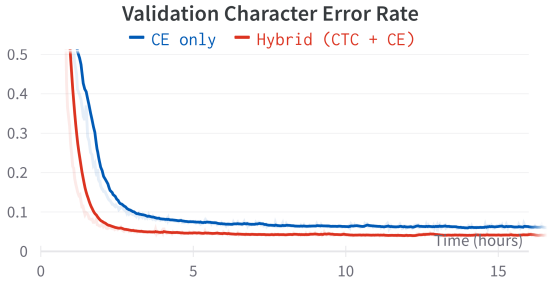


Architecture	IAM		IAM + Synth. Data	
	CER (%)	WER (%)	CER (%)	WER (%)
Our Light Transformer	**5.70**	**18.86**	**4.76**	**16.31**
Our Large Transformer	5.79	19.67	4.87	17.67

- Our light architecture is **competitive**
- Our light architecture might be **trained faster**

Interest of the Hybrid Loss

- **CE only**: Cross-Entropy loss after the decoder
- **Hybrid**: CTC after the encoder and CE after the decoder



Loss Function(s)	IAM		IAM + Synth. Data	
	CER (%)	WER (%)	CER (%)	WER (%)
CE only	10.29	26.36	6.76	19.62
Hybrid (CTC + CE)	**5.70**	**18.86**	**4.76**	**16.31**

- Faster convergence
- Crucial with few data
- Important with synthetic data

Comparison with the state of the art

Model Architecture	# params.	IAM CER (%)	IAM + Synth. Data CER (%)
CRNN + LSTM [Michael et al. 2019]		5.24	
FCN [Yousef et al. 2020]	3.4M	**4.9**	
VAN (line level) [Coquenet et al. 2022]	1.7M	4.95	
Transformer [Kang et al. 2020]	100M	7.62	**4.67**
FPHR Transformer [Singh et al. 2021]	28M		6.5
Forward Transformer [Wick et al. 2021]	13M	6.03	
Bidi. Transformer [Wick et al. 2021]	27M	5.67	
Our Light Transformer-based	6.9M	5.70	4.76

Comparison with the state of the art

Model Encoder	# params.	IAM CER (%)	IAM + Synth. Data CER (%)
CRNN + LSTM [Michael et al. 2019]		5.24	
[illegible]	3.4M	**4.9**	
[illegible] 2022]	1.7M	4.95	
Transformer [Kang et al. 2020]	100M	7.62	**4.67**
FPHR Transformer [Singh et al. 2021]	28M		6.5
Forward Transformer [Wick et al. 2021]	13M	6.03	
Bidi. Transformer [Wick et al. 2021]	27M	5.67	
Our Light Transformer-based	6.9M	5.70	4.76

Compared with other Transformers:

Comparison with the state of the art

Model Encoder	# params.	IAM CER (%)	IAM + Synth. Data CER (%)
CRNN + LSTM [Michael et al. 2019]		5.24	
FCN [Yousef et al. 2020]	3		
VAN (line level) [Coquenet et al. 2022]	1.1		
Transformer [Kang et al. 2020]	100M	7.62	**4.67**
FPHR Transformer [Singh et al. 2021]	28M		6.5
Forward Transformer [Wick et al. 2021]	13M	6.03	
Bidi. Transformer [Wick et al. 2021]	27M	5.67	
Our Light Transformer-based	6.9M	5.70	4.76

Low error rates without additional data

Comparison with the state of the art

Model Encoder	# params.	IAM CER (%)	IAM + Synth. Data CER (%)
CRNN + LSTM [Michael et al. 2019]		5.24	
FCN [Yousef et al. 2020]	3.4M	**4.9**	
VAN (line level) [Coquenet et al. 2022]	1.7M	4.95	
Transformer [Kang et al. 2020]	100M	7.62	**4.67**
FPHR Transformer [Singh et al. 2021]	28M		6.5
Forward Transformer [Wick et al. 2021]	13M	6.03	
Bidi. Transformer [Wick et al. 2021]	27M	5.67	
Our Light Transformer-based	6.9M	5.70	4.76

State-of-the-art results with synthetic data

Comparison with the state of the art

Model Encoder	# params.	IAM CER (%)	IAM + Synth. Data CER (%)
CRNN + LSTM [Michael et al. 2019]		5.24	
FCN [Yousef et al. 2020]		**4.9**	
VAN (line level) [Coquenet et al. 202		4.95	
Transformer [Kang et al. 2020]	100M	7.62	**4.67**
FPHR Transformer [Singh et al. 2021]	28M		6.5
Forward Transformer [Wick et al. 2021]	13M	6.03	
Bidi. Transformer [Wick et al. 2021]	27M	5.67	
Our Light Transformer-based	6.9M	5.70	4.76

While being
a light Transformer

Comparison with the state of the art

Model Encoder	# params.	IAM CER (%)	IAM + Synth. Data CER (%)
CRNN + LSTM [Michael et al. 2019]		5.24	
FCN [Yousef et al. 2020]	3.4M	**4.9**	
VAN (line level) [Coquenet et al. 2022]	1.7M	4.95	
Transformer [Kang et al. 2020]	100M	7.62	**4.67**
FPHR Transformer [Singh et al. 2021]	28M		6.5
Forward Transformer [Wick et al. 2021]	13M	6.03	
Bidi. Transformer [Wick et al. 2021]	27M	5.67	
Our Light Transformer-based	6.9M	5.70	4.76

Conclusion

Our Contribution

A **light Transformer architecture**, trained with a **hybrid loss**

- **Faster to train** than other Transformers
- **Good results without additional data**
- **State-of-the-art results** with synthetic data

Future Works: Historical Documents

- Ability of Transformers to **model the language** is crucial
- **Very few annotated data** $\Rightarrow$ our light Transformer architecture

Find the paper in

- DAS 2022 (A Light Transformer-Based Architecture for Handwritten Text Recognition)
- HAL: `https://hal.archives-ouvertes.fr/hal-03685976v1`

State of the Art: From RNN to Transformer

Usual approaches: Convolutional Recurrent Neural Networks (CRNN)

- Convolutional layers + recurrent layers

⇒ Lack of parallelism / **slow training speed**

Fully Convolutional Networks [Ingle et al. 2019, Yousef et al. 2020, Coquenet et al. 2021]

- Composed of convolutional layers, no recurrent layers

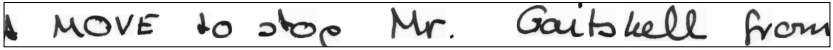
⇒ Faster training speed, but might be **hard to learn long-range contexts**

Multi-Head Attention (Transformer layers) [Vaswani et al. 2017]

- Able to **learn long-range context**
- Strong **parallelism**

⇒ **Good alternative** but **require a lot of training data**

A l'entraînement : Decoding en parallèle et Teacher Forcing



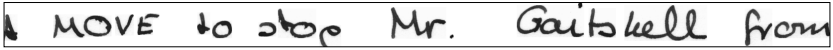
<sos>A MOVE to stop Mr. Gaitskell from

En paralèlle grâce à du Masking

<sos> →	A
<sos>A →	A_
<sos>A_ →	A_N
<sos>A_M →	A_MO
<sos>A_MO →	A_MOV
⋮	⋮
<sos>A_MOVE_to_stop_Mr._Gaitskell_from →	A_MOVE_to_stop_Mr_Gaitskell_from<eos>

<sos>A_MOVE_to_stop_Mr._Gaitskell_from → A_NOVE_to_stop_Mr_Gaitskell_from<eos>

A l'inférence : Decoding séquentiel



Decoding Séquentiel

<sos>	→	A
<sos>A	→	A_
<sos>A_	→	A_N
<sos>A_N	→	A_NO
<sos>A_NO	→	A_NOV
⋮	⋮	⋮
<sos>A_NOVE_to_stop_ Mr._Gaitswell_from	→	A_NOVE_to_stop_Mr_ Gaitswell_from<eos>

→ A_NOVE_to_stop_Mr_ Gaitswell_from<eos>

CER and WER

Definition

- CER: Character Error Rate
- WER: Word Error Rate
- Edit distance between prediction and groundtruth

$$CER = \frac{I+D+S}{N}$$

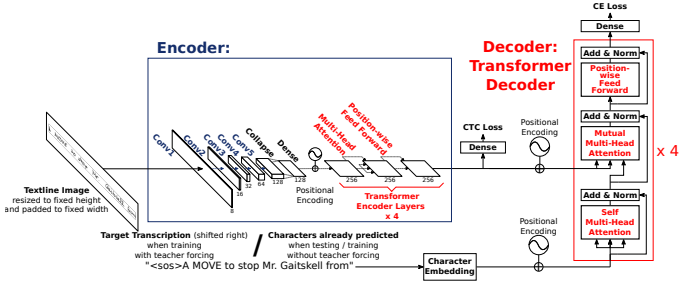
I Number of Insertions

S Number of Substitutions

D Number of Deletions

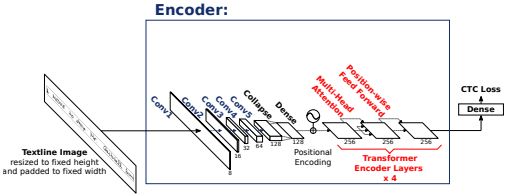
N Length of the ground-truth

Ablation Study: Transformer Layers instead of Recurrent Layers



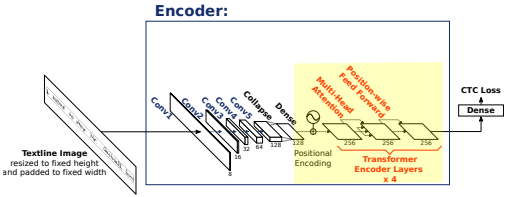
Without the decoder

Ablation Study: Transformer Layers instead of Recurrent Layers



Without the decoder

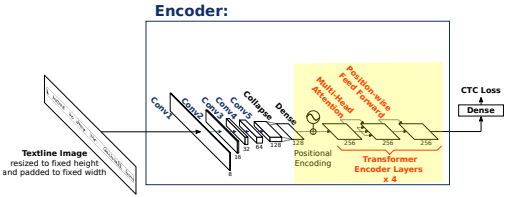
Ablation Study: Transformer Layers instead of Recurrent Layers



Without the decoder

- CRNN with Transformer instead of recurrent layers

Ablation Study: Transformer Layers instead of Recurrent Layers



Architecture	# params.	IAM		IAM + Synth. Data	
		CER (%)	WER (%)	CER (%)	WER (%)
CRNN (Baseline)	1.7M	6.14	23.26		
Our Encoder only	3.2M	5.93	22.82		

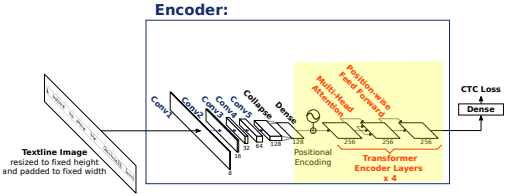
Without the decoder

- CRNN with Transformer instead of recurrent layers

Recurrent $\Rightarrow$ Transformer

- More parameters
- Lower error rates
  - Better context

Ablation Study: Transformer Layers instead of Recurrent Layers



Architecture	# params.	IAM		IAM + Synth. Data	
		CER (%)	WER (%)	CER (%)	WER (%)
CRNN (Baseline)	1.7M	6.14	23.26	5.66	21.62
Our Encoder only	3.2M	5.93	22.82	6.15	24.02

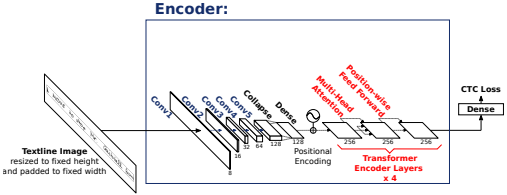
Without the decoder

- CRNN with Transformer instead of recurrent layers

Recurrent $\Rightarrow$ Transformer

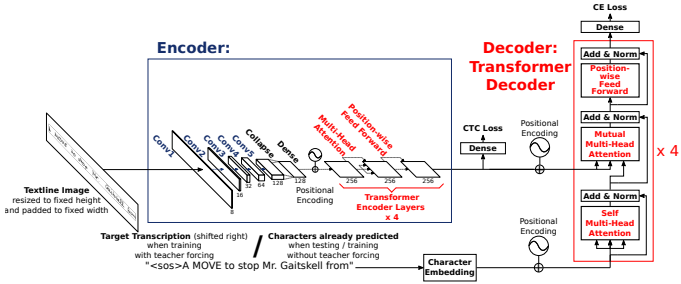
- More parameters
- Lower error rates
  - Better context
- Worse with synthetic data (may not generalize well)

Ablation Study: Decoder



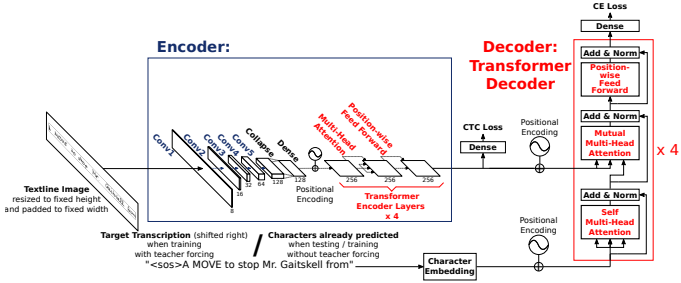
With the decoder

Ablation Study: Decoder



With the decoder

Ablation Study: Decoder



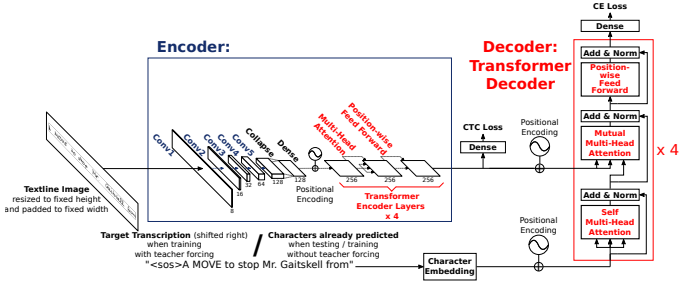
Architecture	# params.	IAM		IAM + Synth. Data	
		CER (%)	WER (%)	CER (%)	WER (%)
CRNN (Baseline)	1.7M	6.14	23.26		
Our Encoder only	3.2M	5.93	22.82		
Our Light Transformer	6.9M	**5.70**	**18.86**		

With the decoder

Ability to model the language

- Lower error rates
- Stronger impact on the WER

Ablation Study: Decoder



Architecture	# params.	IAM		IAM + Synth. Data	
		CER (%)	WER (%)	CER (%)	WER (%)
CRNN (Baseline)	1.7M	6.14	23.26	5.66	21.62
Our Encoder only	3.2M	5.93	22.82	6.15	24.02
Our Light Transformer	6.9M	**5.70**	**18.86**	**4.76**	**16.31**

With the decoder

Ability to model the language

- Lower error rates
- Stronger impact on the WER

Benefits more from synthetic data

- More data to learn the language