Layer-normalized LSTM for Hybrid-HMM and End-to-End ASR

Mohammad Zeineldeen, Albert Zeyer, Ralf Schlüter, Hermann Ney

Human Language Technology and Pattern Recognition Group,
RWTH Aachen University, Aachen, Germany

AppTek GmbH, Aachen, Germany

ICASSP 2020, Barcelona, Spain
May 8, 2020

Introduction

Layer normalization is a critical component for training deep models

- Experiments showed that Transformer [Vaswani & Shazeer+ 17, Irie & Zeyer+ 19, Wang & Li+ 19] **does not converge** without layer normalization
- RNMT+ [Chen & Firat+ 18], deep encoder-decoder LSTM RNN model, also depends crucially on layer normalization for convergence.

Contribution of this work

- Investigation of layer normalization variants for LSTMs
- Improvement of the **overall performance** of ASR systems
- Improvement of the **stability** of training (deep) models
- Models become **more robust** to hyperparameter tuning
- Models can work well even without pretraining when using layer-normalized LSTMs

Introduction

Layer normalization (LN) [Ba & Kiros$^+$ 16] is defined as:

$$\mathrm{LN}(x;\gamma,\beta) = \gamma \odot \frac{x - \mathbb{E}[x]}{\sqrt{\mathrm{Var}[x] + \epsilon}} + \beta$$

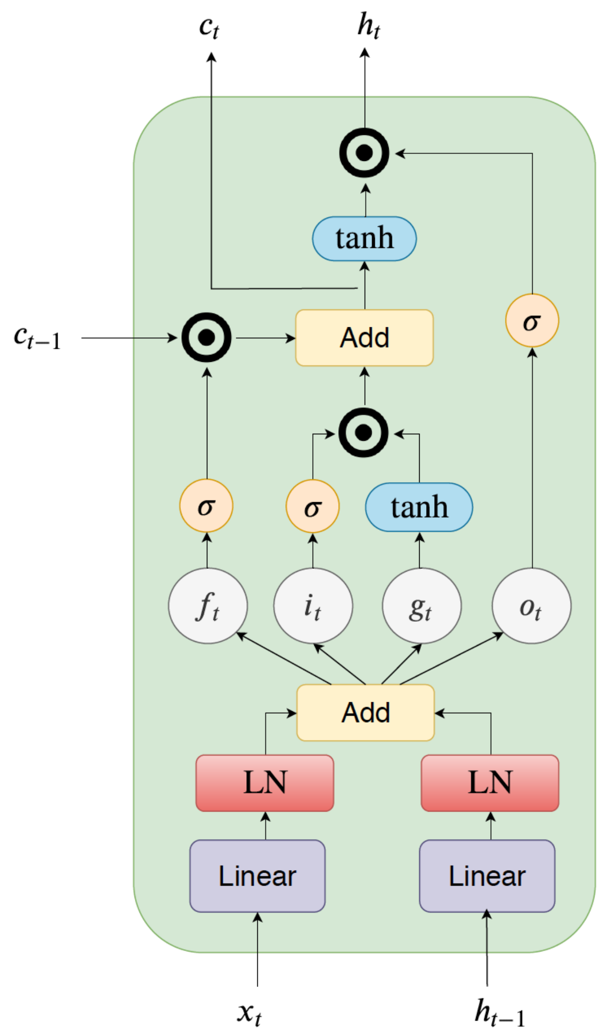
- $\mathbb{E}[x]/\mathrm{Var}[x]$ are mean/variance computed over the **feature dimension**
- $\gamma \in \mathbb{R}^D$ and $\beta \in \mathbb{R}^D$ are the gain and shift respectively (trainable parameters)
- $\odot$ is an element-wise multiplication operator
- ϵ is a small value used to avoid dividing by very small variance
- In the next slides, LN LSTM denotes layer-normalized LSTM

Layer-normalized LSTM Variants

Global Norm [Ba & Kiros+ 16]

$$\begin{pmatrix} f_t \\ i_t \\ o_t \\ g_t \end{pmatrix} = \mathrm{LN}(W_{hh}h_{t-1}) + \mathrm{LN}(W_{hx}x_t) + b$$

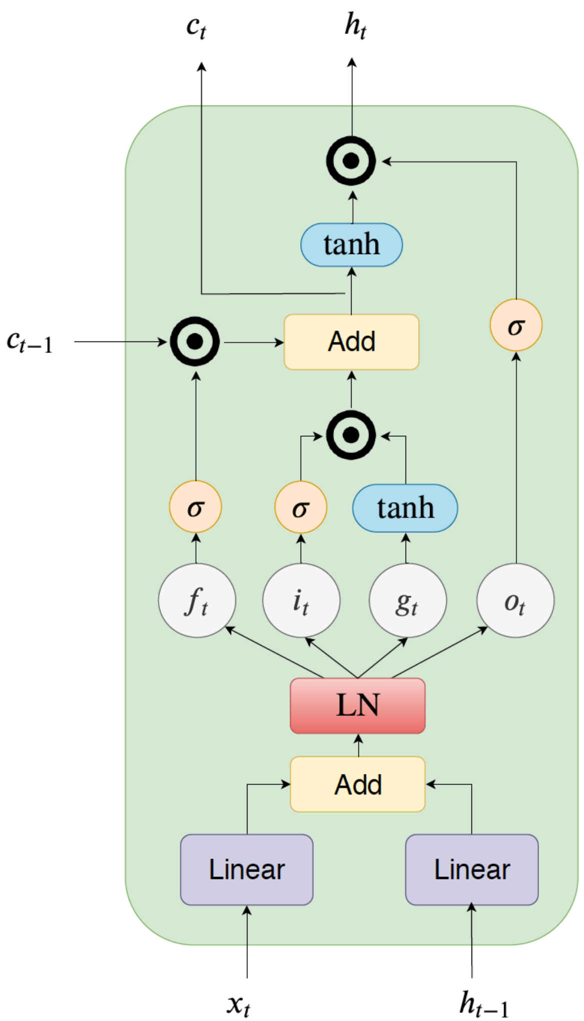
- LN is applied **separately** to each of the forward and recurrent inputs
- Gives the model the flexibility of learning two relative normalized distributions



Global Joined Norm

$$\begin{pmatrix} f_t \\ i_t \\ o_t \\ g_t \end{pmatrix} = \mathrm{LN}(W_{hx}x_t + W_{hh}h_{t-1})$$

- To our best knowledge, this variant was not used in any work
- LN is applied **jointly** to the forward and recurrent inputs after adding them together
- There is a single globally normalized distribution

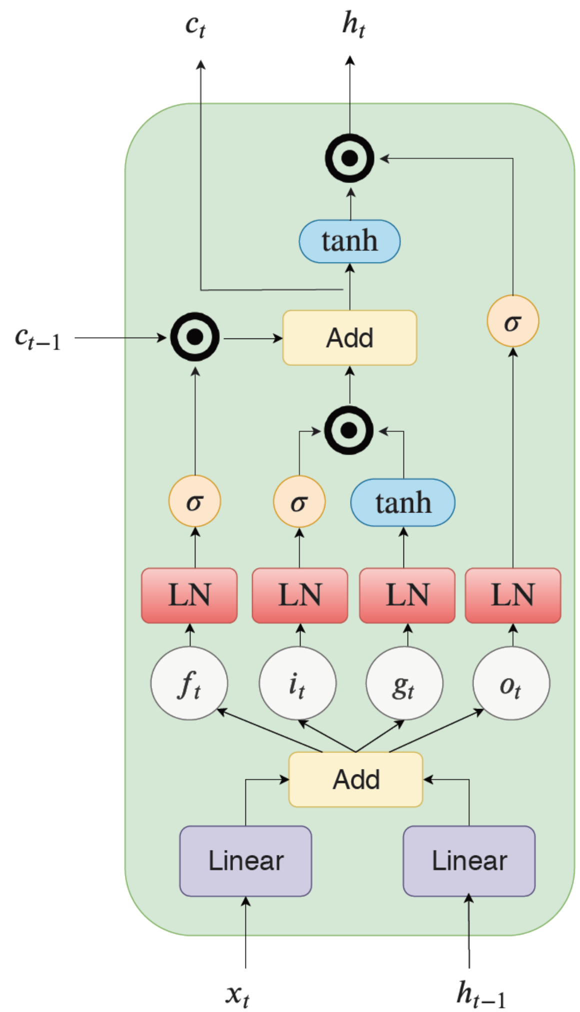


Layer-normalized LSTM Variants

Per Gate Norm [Chen & Firat⁺ 18]

$$\begin{pmatrix} f_t \\ i_t \\ o_t \\ g_t \end{pmatrix} = \begin{pmatrix} \mathrm{LN}(f_t) \\ \mathrm{LN}(i_t) \\ \mathrm{LN}(o_t) \\ \mathrm{LN}(g_t) \end{pmatrix}$$

- LN is applied separatly to each LSTM gate
- There are learned distributions for each gate

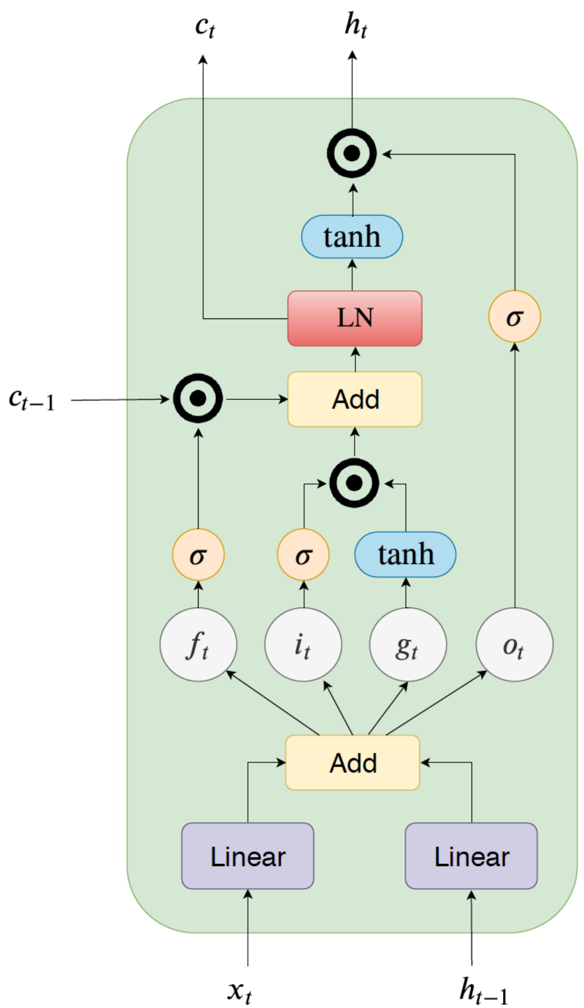


Layer-normalized LSTM Variants

Cell Norm [Ba & Kiros+ 16]

$$c_t = \mathrm{LN}(\sigma(f_t) \odot c_{t-1} + \sigma(i_t) \odot \tanh(g_t))$$

- LN is applied to the LSTM cell output



Experimental Setups

Data

- Switchboard 300h (English telephone speech)
- For testing, Hub5'00 (Switchboard + CallHome) and Hub5'01 are used

Hybrid baseline

- For NN training, alignments from a triphone CART-based GMM are used as ground truth labels
- The NN acoustic model consists of L bidirectional LSTM RNN layers
- The number of units in each direction is 500
- A 4-gram count-based language model is used for recognition

End-to-end baseline

- Attention based end-to-end baseline [Zeyer & Irie+ 18, Chan & Jaitly+ 16]
- 6 bidirectional LSTM RNN layers encoder with 1024 units for each direction
- 1 unidirectional LSTM RNN layer decoder with 1024 units
- Multi-layer perceptron attention is used
- Uses byte-pair-encoding as subword units with an alphabet size of 1k
- No utilization of a language model or any data augmentation methods

LN-LSTM for Hybrid-HMM ASR

L	Layer Norm Variant	Layer Norm Cell	WER [%] Hub5'00 $\sum$	Hub5'00 SW	Hub5'00 CH	Hub5'01 $\sum$	Epoch
6	-	-	14.3	9.6	19.0	14.5	12.8
	Joined	Yes	**14.1**	9.5	**18.8**	**14.1**	12.8
	Global	Yes	**14.1**	**9.3**	18.9	14.2	12.6
	Per Gate	Yes	14.5	9.8	19.2	14.6	12.8
	Joined	No	14.4	9.7	19.1	14.5	13.2
	Global	No	14.2	9.5	18.9	**14.1**	12.8
	Per Gate	No	14.7	10.0	19.4	14.6	12.8
8	-	-	14.4	9.8	19.1	14.3	12.6
	Joined	Yes	14.4	9.6	19.2	14.4	12.8
	Global	Yes	**14.0**	9.6	**18.5**	**14.1**	12.8
	Per Gate	Yes	14.2	9.5	18.9	14.3	12.8
	Joined	No	14.5	9.9	19.1	14.7	11.0
	Global	No	**14.0**	**9.4**	18.6	14.4	12.8
	Per Gate	No	14.5	9.8	19.2	14.8	10.8

- L: number of layers
- Training is often stable so we do not expect significant improvement
- Small improvement with deeper models
- **Global Norm** reports the best results

Experiments

LN-LSTM for end-to-end ASR[1]

Pre-train	Layer Norm		WER [%]				Epoch
	Variant	Cell	Hub5'00			Hub5'01	
			$\sum$	SW	CH	$\sum$	
Y	-	-	19.1	12.9	25.2	18.8	13.0
	Joined	Yes	18.3	12.1	24.5	17.8	10.8
	Global		22.2	14.9	29.4	20.7	20.0
	Per Gate		18.1	**11.7**	24.4	17.8	13.0
	Joined	No	**17.9**	11.8	**23.9**	**17.6**	11.8
	Global		19.1	12.8	25.5	18.5	12.3
	Per Gate		18.4	12.0	24.8	18.1	13.3
N	-	-	19.2	12.9	25.5	18.6	<u>20.0</u>
	Joined	Yes	*	*	*	*	
	Global		19.0	12.5	25.4	18.4	11.0
	Per Gate		*	*	*	*	
	Joined	No	**17.2**	**11.1**	**23.2**	**16.7**	<u>13.3</u>
	Global		18.9	12.2	25.4	18.1	16.0
	Per Gate		18.4	12.0	24.8	18.1	13.3

[1] LN is applied to both encoder and decoder

- 10% relative improvement in terms of WER
- **Global Joined Norm** reports the best results and even without pretraining
- Baseline without pretraining requires **heavy hyperparameter tuning**
- LN LSTM models require **less hyperparameter** tuning to converge and often from the first run
- **Faster convergence** is observed with LN LSTM
- *: model broken

Training variance

- Run same model with multiple random seeds
- Run multiple times same model with same random seed

Layer Norm	Variant	WER [%] (min-max, μ, σ) Hub5'00	Hub5'01
No	5 seeds	19.4-20.7, 20.2, **0.19**	19.1-20.2, 19.7, 0.18
Yes	5 seeds	17.1-17.6, 17.3, **0.08**	16.7-16.9, 16.8, 0.03
No	5 runs	19.2-19.7, 19.4, 0.08	18.6-19.4, 19.0, 0.14
Yes	5 runs	17.2-17.4, 17.3, 0.03	16.7-17.0, 16.8, 0.04

- Applied for the attention-based end-to-end model
- For LN LSTM, Global Joined Norm is used
- No pretraining is applied
- LN LSTM model is **robust to parameter initialization**

Experiments

Deeper encoder

Layer Norm	encN	WER [%] Hub5'00 $\sum$	SW	CH	Hub5'01 $\sum$
No	6	19.2	12.9	25.5	18.6
Yes		**17.2**	11.1	23.2	16.7
No	7	∞	∞	∞	∞
Yes		17.4	11.4	23.4	16.8
No	8	∞	∞	∞	∞
Yes		17.5	11.3	23.7	16.9

- Applied for the attention-based end-to-end model
- encN: number of encoder layers
- Global Joined Norm is used and no pretraining is applied
- ∞: no convergence
- Worse results due to **overfitting**
- LN LSTM allows training **deeper** models **without pretraining**

Conclusion & Outlook

Summary

- Investigated different variants of LN LSTM
- Successful training with **better stability**, and **better overall system performance** for ASR using LN LSTM
- Experiments show that LN LSTM models require **less hyperparameter** tuning, in addition to being **robust** to training variance
- Showed that in some cases there is no need for pretraining with LN LSTMs
- LN LSTM allows for training deeper models

Future work

- How much layer normalization do we need?
- Implementing an optimized LN-LSTM kernel for speed-up
- Applying SpecAugment [Park & Chan^{+} 19] for data augmentation

Thank you for your attention

Apendix

[Ba & Kiros+ 16] J. Ba, J. R. Kiros, G. E. Hinton.
Layer normalization.
ArXiv, Vol. abs/1607.06450, 2016.

[Chan & Jaitly+ 16] W. Chan, N. Jaitly, Q. Le, O. Vinyals.
Listen, attend and spell: A neural network for large vocabulary conversational speech recognition.
In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4960–4964, March 2016.

[Chen & Firat+ 18] M. X. Chen, O. Firat, A. Bapna, M. Johnson, W. Macherey, G. Foster, L. Jones, N. Parmar, M. Schuster, Z. Chen, Y. Wu, M. Hughes.
The best of both worlds: Combining recent advances in neural machine translation.
CoRR, Vol. abs/1804.09849, 2018.

[Irie & Zeyer+ 19] K. Irie, A. Zeyer, R. Schlüter, H. Ney.
Language modeling with deep transformers.
ArXiv, Vol. abs/1905.04226, 2019.

[Park & Chan+ 19] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, Q. V. Le.
Specaugment: A simple data augmentation method for automatic speech recognition.
ArXiv, Vol. abs/1904.08779, 2019.

[Vaswani & Shazeer+ 17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin.
Attention is all you need.
In *NIPS*, 2017.

[Wang & Li+ 19] Q. Wang, B. Li, X. Tong, J. Zhu, C. Li, D. F. Wong, L. S. Chao.
Learning deep transformer models for machine translation.
In *ACL*, 2019.

[Zeyer & Irie+ 18] A. Zeyer, K. Irie, R. Schlüter, H. Ney.
Improved training of end-to-end attention models for speech recognition.
In *Interspeech*, Hyderabad, India, Sept. 2018.