An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling

Radek Bartyzal

Let's talk ML in Prague

5. 3. 2019

Sequence Modeling

Sequence Modeling:

- sequence to sequence of same length
- predict after each time step: $x_t \rightarrow y_t$
- predictions based only on the previous elements in the sequence

$\Longrightarrow$ Not suitable for e.g. translation where

- output sequence has different length
- each element of output sequence depends on the whole input sequence = we compress the whole input sequence and then reconstruct it

Temporal Convolutional Networks (TCN)

Family of architectures:

- causal convolution = only look at the past
- sequence to sequence of the same length = 1D FCN with zero padding to keep same size for the next layer

Able to have long effective history by:

- deep nets with residual connections = learn modifications to the identity mapping rather than the entire transformation
- dilated convolutions = exponentially increased receptive field with subsequent layers

Dilated Causal Convolution

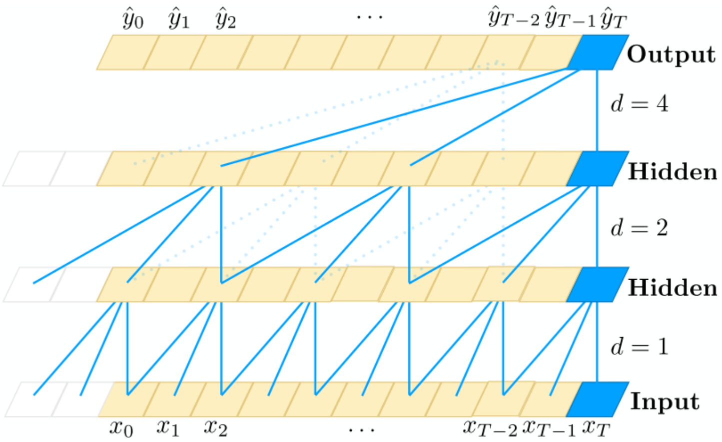

Figure: Dilated causal convolution with $k = 3$, $d = [2^0, 2^1, 2^2]$. The receptive field is able to cover all values from the input sequence.

TCN Residual Block

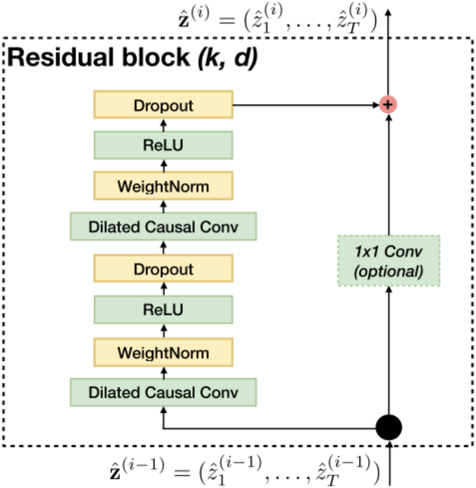

Figure: TCN residual block. An 1x1 convolution is added when residual input and output have different dimensions.

Generic networks results

Sequence Modeling Task	Model Size (≈)	Models			
		LSTM	GRU	RNN	**TCN**
Seq. MNIST (accuracy^h)	70K	87.2	96.2	21.5	**99.0**
Permuted MNIST (accuracy)	70K	85.7	87.3	25.3	**97.2**
Adding problem T=600 (loss$^\ell$)	70K	0.164	**5.3e-5**	0.177	**5.8e-5**
Copy memory T=1000 (loss)	16K	0.0204	0.0197	0.0202	**3.5e-5**
Music JSB Chorales (loss)	300K	8.45	8.43	8.91	**8.10**
Music Nottingham (loss)	1M	3.29	3.46	4.05	**3.07**
Word-level PTB (perplexity$^\ell$)	13M	**78.93**	92.48	114.50	88.68
Word-level Wiki-103 (perplexity)	-	48.4	-	-	**45.19**
Word-level LAMBADA (perplexity)	-	4186	-	14725	**1279**
Char-level PTB (bpc$^\ell$)	3M	1.36	1.37	1.48	**1.31**
Char-level text8 (bpc)	5M	1.50	1.53	1.69	**1.45**

Figure: The generic TCN architecture outperforms canonical recurrent networks across a comprehensive suite of tasks and datasets.

State of the art results

TCN vs. SoTA Results					
Task	**TCN Result**	**Size**	**SoTA**	**Size**	**Model**
Seq. MNIST (acc.)	99.0	21K	99.0	21K	Dilated GRU (Chang et al., 2017)
P-MNIST (acc.)	97.2	42K	95.9	42K	Zoneout (Krueger et al., 2017)
Adding Prob. 600 (loss)	5.8e-5	70K	5.3e-5	70K	Regularized GRU
Copy Memory 1000 (loss)	3.5e-5	70K	0.011	70K	EURNN (Jing et al., 2017)
JSB Chorales (loss)	8.10	300K	3.47	-	DBN+LSTM (Vohra et al., 2015)
Nottingham (loss)	3.07	1M	1.32	-	DBN+LSTM (Vohra et al., 2015)
Word PTB (ppl)	88.68	13M	47.7	22M	AWD-LSTM-MoS + Dynamic Eval. (Yang et al., 2018)
Word Wiki-103 (ppl)	45.19	148M	40.4	>300M	Neural Cache Model (Large) (Grave et al., 2017)
Word LAMBADA (ppl)	1279	56M	138	>100M	Neural Cache Model (Large) (Grave et al., 2017)
Char PTB (bpc)	1.31	3M	1.22	14M	2-LayerNorm HyperLSTM (Ha et al., 2017)
Char text8 (bpc)	1.45	4.6M	1.29	>12M	HM-LSTM (Chung et al., 2016)

Figure: State of the art (SOTA) results.

Sources

1. Bai, Shaojie, J. Zico Kolter, and Vladlen Koltun. "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling." arXiv preprint arXiv:1803.01271 (2018).
https://arxiv.org/abs/1803.01271
Code: https://github.com/locuslab/TCN