NMT IN LINEAR TIME (BYTENET)

발표: 김동현

OVERVIEW

기존 모델 (seq2seq)
1. sequence에 따른 시간 super-linear 하게 증가 (RNN 병렬처리x)
2. fixed sequence에 결과를 담는 것이 긴 문장을 기억하기에 힘듦.

→ ByteNet 설계!

1. sequence에 따른 시간 linear 하게 증가 (CNN 병렬처리)
2. char level ENG-GER NMT SOTA

DESIDERATA

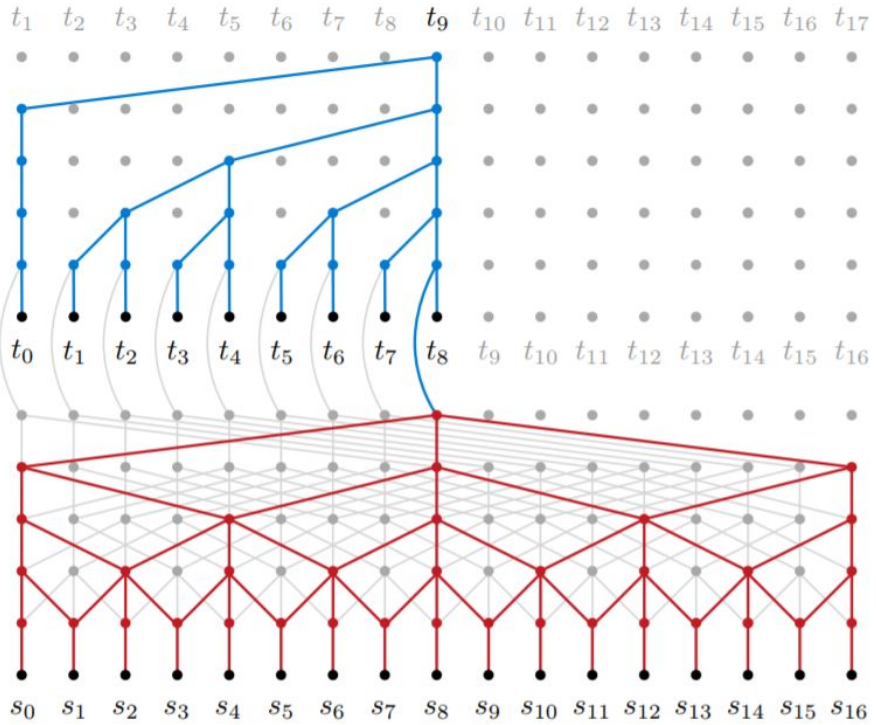
1. sequence 길이에 따른 실행시간이 linear 했으면 함
   → CNN 써서 해결!
2. input sequence가 길다면 source representation도 길어야 함
   → encoder-decoder stacking 활용!
3. 학습 경로가 짧도록 하면 long-range dependency 학습 잘함
   → dilation 활용!

BYTENET

decoder
encoder

Characteristics!

1. FCN (1d CNN)
2. Enc-Dec Stacking
3. Dynamic Unfolding
4. Masking

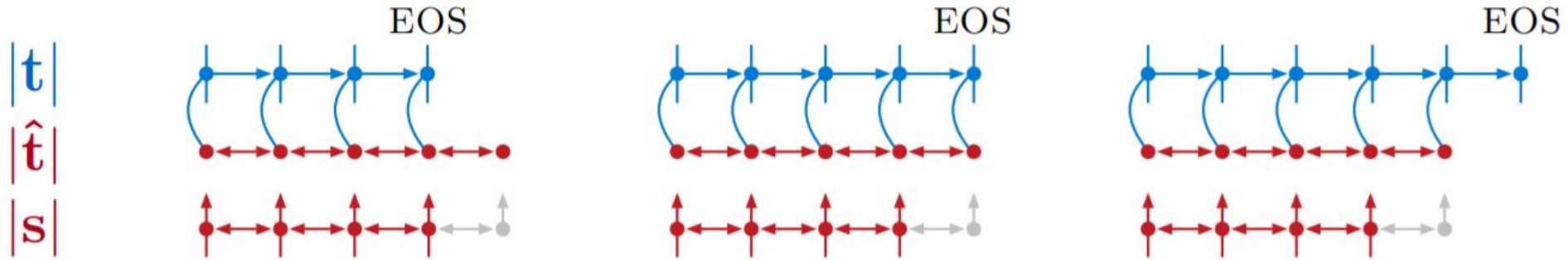


DYNAMIC UNFOLDING

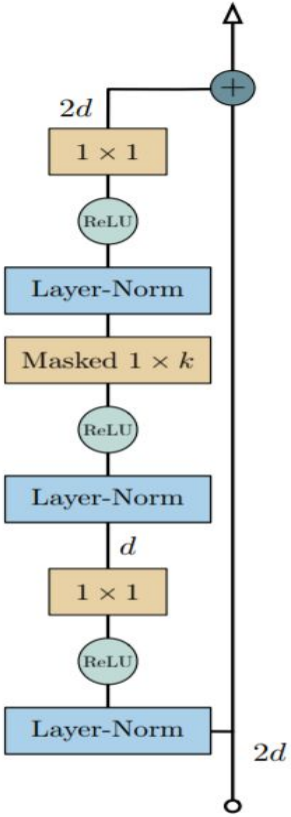
1. |s| → input seq len
2. |t^| → enc out len
3. t → final out len

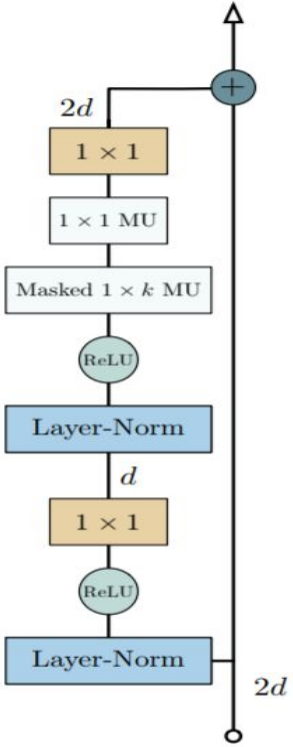
$$|\hat{\mathbf{t}}| = a|\mathbf{s}| + b$$

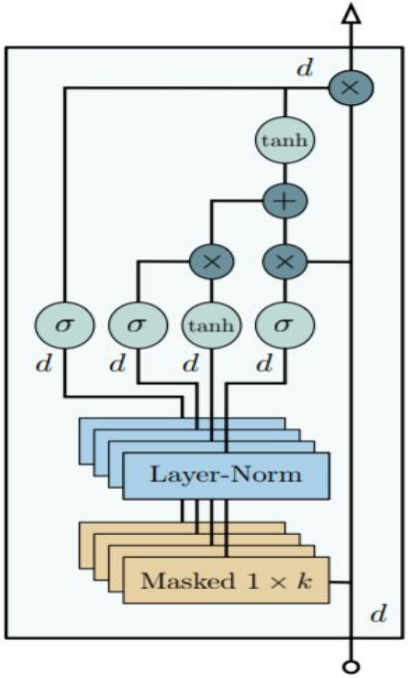
a = 1.2, b = 0 (ENG-GER)



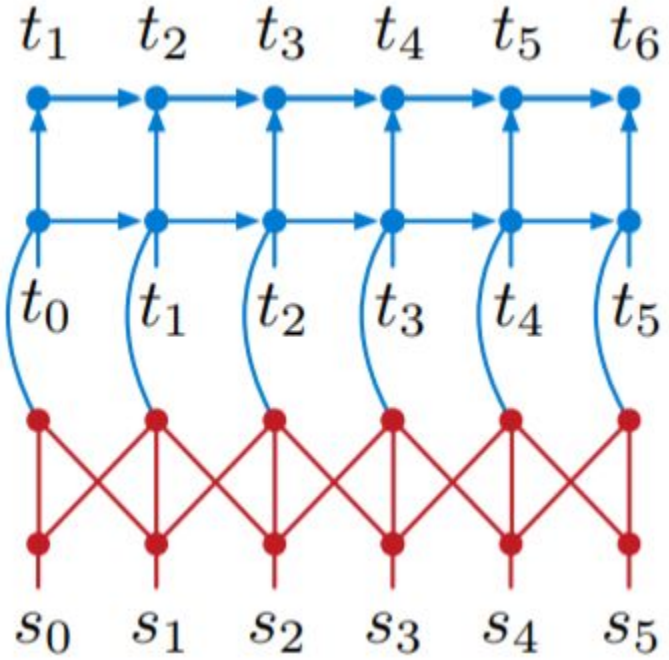
RESIDUAL BLOCKS

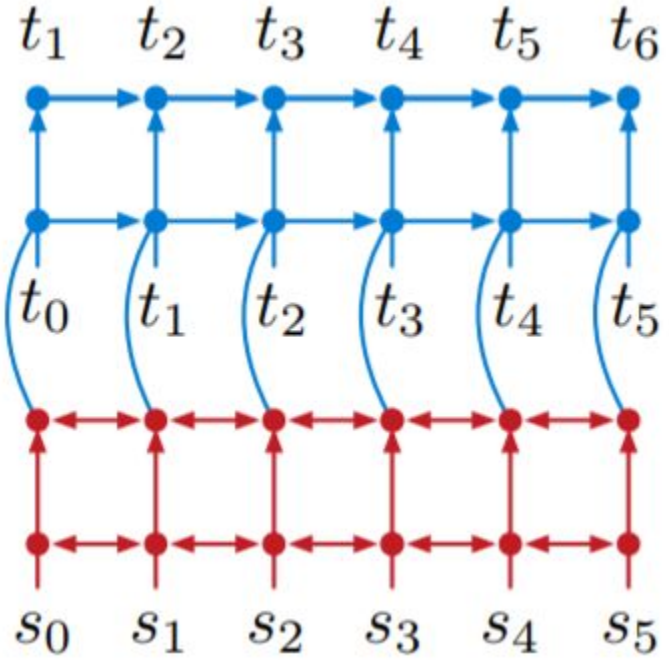






VARIANTS


t_1 t_2 t_3 t_4 t_5 t_6
t_0 t_1 t_2 t_3 t_4 t_5
s_0 s_1 s_2 s_3 s_4 s_5


t_1 t_2 t_3 t_4 t_5 t_6
t_0 t_1 t_2 t_3 t_4 t_5
s_0 s_1 s_2 s_3 s_4 s_5

RESULT (MODEL PROPERTIES)

Model	Net_S	Net_T	Time	RP	Path_S	Path_T
RCTM 1	CNN	RNN	$\|S\|\|S\| + \|T\|$	no	$\|S\|$	$\|T\|$
RCTM 2	CNN	RNN	$\|S\|\|S\| + \|T\|$	yes	$\|S\|$	$\|T\|$
RNN Enc-Dec	RNN	RNN	$\|S\| + \|T\|$	no	$\|S\| + \|T\|$	$\|T\|$
RNN Enc-Dec Att	RNN	RNN	$\|S\|\|T\|$	yes	1	$\|T\|$
Grid LSTM	RNN	RNN	$\|S\|\|T\|$	yes	$\|S\| + \|T\|$	$\|S\| + \|T\|$
Extended Neural GPU	cRNN	cRNN	$\|S\|\|S\| + \|S\|\|T\|$	yes	$\|S\|$	$\|T\|$
Recurrent ByteNet	RNN	RNN	$\|S\| + \|T\|$	yes	$\max(\|S\|, \|T\|)$	$\|T\|$
Recurrent ByteNet	CNN	RNN	$c\|S\| + \|T\|$	yes	c	$\|T\|$
ByteNet	CNN	CNN	$c\|S\| + c\|T\|$	yes	c	c

Table 1. Properties of various neural translation models.

RESULT (BLEU SCORE)

Model	Inputs	Outputs	WMT Test '14	WMT Test '15
Phrase Based MT (Freitag et al., 2014; Williams et al., 2015)	phrases	phrases	20.7	24.0
RNN Enc-Dec (Luong et al., 2015)	words	words	11.3	
Reverse RNN Enc-Dec (Luong et al., 2015)	words	words	14.0	
RNN Enc-Dec Att (Zhou et al., 2016)	words	words	20.6	
RNN Enc-Dec Att (Luong et al., 2015)	words	words	20.9	
GNMT (RNN Enc-Dec Att) (Wu et al., 2016a)	word-pieces	word-pieces	**24.61**	
RNN Enc-Dec Att (Chung et al., 2016b)	BPE	BPE	19.98	21.72
RNN Enc-Dec Att (Chung et al., 2016b)	BPE	char	21.33	23.45
GNMT (RNN Enc-Dec Att) (Wu et al., 2016a)	char	char	22.62	
ByteNet	char	char	**23.75**	**26.26**

Table 2. BLEU scores on En-De WMT NewsTest 2014 and 2015 test sets.

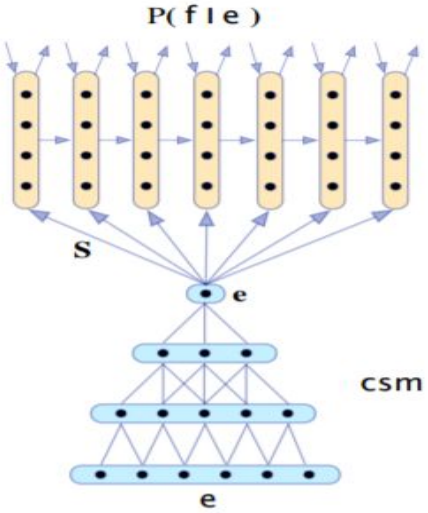
	WMT Test '14	WMT Test '15
Bits/character	0.521	0.532
BLEU	23.75	26.26

RESULTS (BPC)

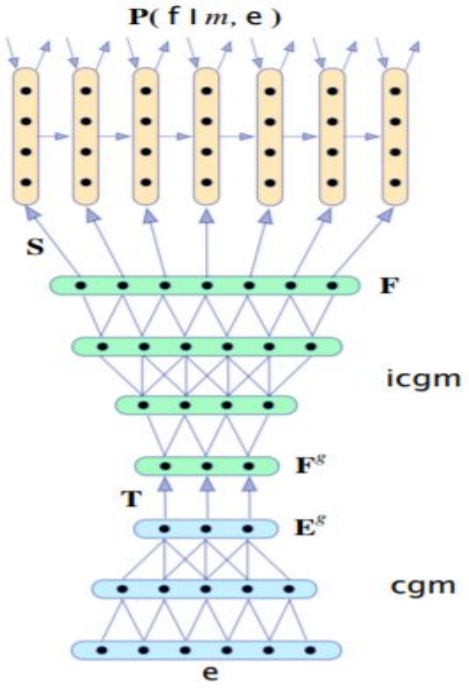
Model	Test
Stacked LSTM (Graves, 2013)	1.67
GF-LSTM (Chung et al., 2015)	1.58
Grid-LSTM (Kalchbrenner et al., 2016a)	1.47
Layer-normalized LSTM (Chung et al., 2016a)	1.46
MI-LSTM (Wu et al., 2016b)	1.44
Recurrent Memory Array Structures (Rocki, 2016)	1.40
HM-LSTM (Chung et al., 2016a)	1.40
Layer Norm HyperLSTM (Ha et al., 2016)	1.38
Large Layer Norm HyperLSTM (Ha et al., 2016)	1.34
Recurrent Highway Networks (Srivastava et al., 2015)	1.32
ByteNet Decoder	**1.31**

Table 3. Negative log-likelihood results in bits/byte on the Hutter Prize Wikipedia benchmark.

APPENDIX A. RECURRENT CONTINUOUS MODELS

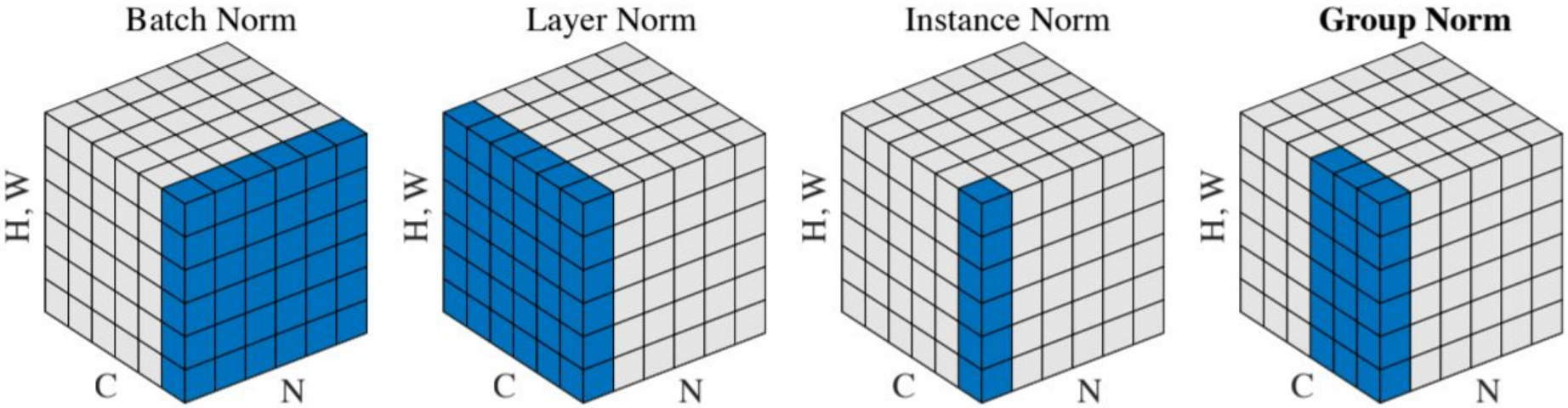


RCTM I

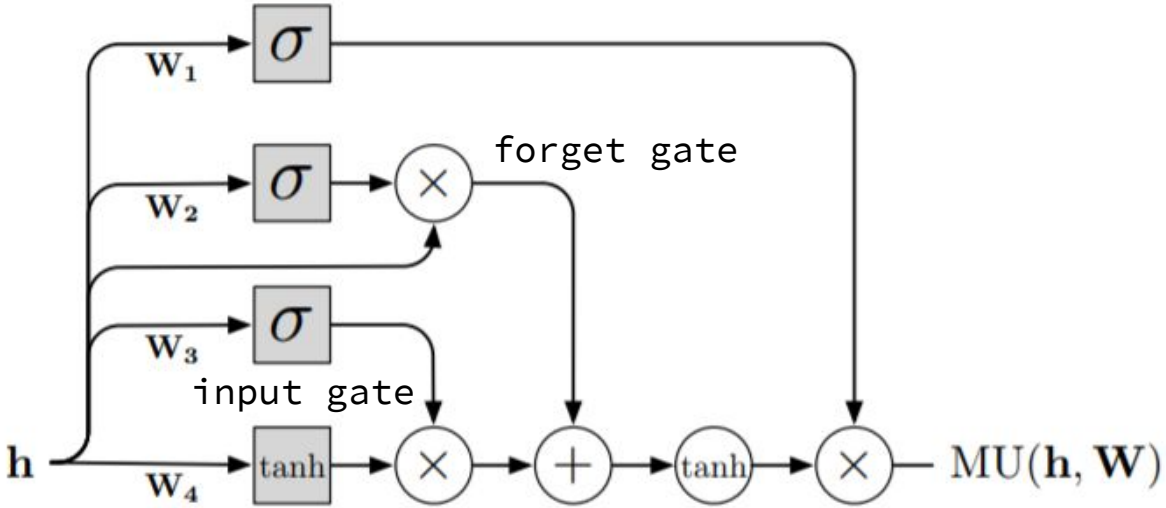


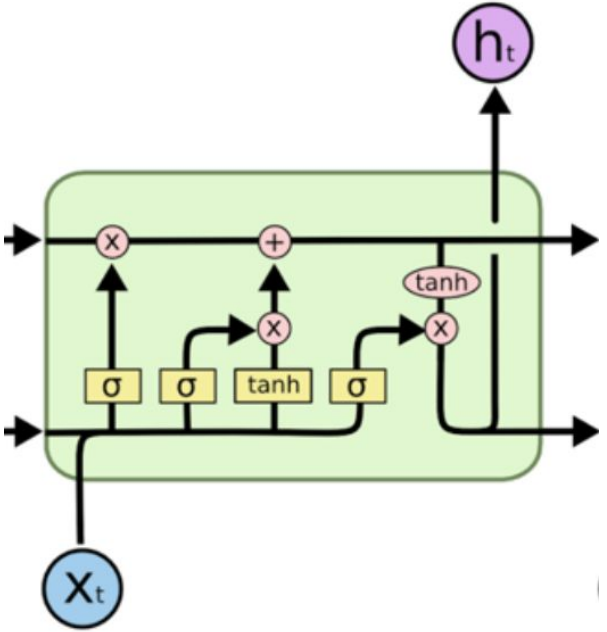
RCTM II

APPENDIX B. LAYER NORM



APPENDIX C. RESIDUAL MULTIPLICATIVE BLOCK





APPENDIX D. BIT PER CHARACTER

$$bpc(string) = \frac{1}{T}\sum_{t=1}^{T} H(P_t, \hat{P}_t) = -\frac{1}{T}\sum_{t=1}^{T}\sum_{c=1}^{n} P_t(c)\log_2 \hat{P}_t(c),$$
$$= -\frac{1}{T}\sum_{t=1}^{T} \log_2 \hat{P}_t(x_t).$$