

# Progress of Neural Machine Translation with Memory Network

Yang FENG

Jan 3, 2016

## 1 Baseline

**Improve the baseline** by

1. rnn to bidirectional\_rnn;
2. SGD optimizer to Adadelta optimization;
3. initializing the initial attention state;
4. seperating hidden\_edim and hidden\_units (in the baseline, they share a common parameter);
5. changing the frequency of saving models.

**To do:**

change one-best greedy search to **beam search**.

**Cannot get reasonable results of baseline**, shown in Table 1. Investigated in terms of

1. output order chaos;
2. file format.

**Fix this problem** by using greater learning rate of 0.5, and results are shown in Table 2.

## 2 Adding attended lexical memory

### 2.1 Investigate (2017-1-3)

1. learning rate, shown in Table 2.1

2. the placeholder for the target translation of the source words whose target translation is less than trans-limit, shown in Table 2.1  
v0: all use PAD\_ID  
v1: use NULL\_ID for words, and PAD\_ID for PAD  
v2: all use NULL\_ID
3. size of translations (trans-limit), shown in Table 2.1  
For each source word, the number of target translation words
4. training algorithm shown in Table 2.1  
Also tried Adam, but got very bad results.

System	Speed (per epoch)	Dev (BLEU4 )	Test (BLEU4)
nmt	84s	10.3	11.1 (49w epch)
nmt <sup>+</sup>	108s	13.0	15.4 (26w epch)
nmt <sup>+</sup> +mn	180s	11.1	12.5 (30w epch)

Table 1: configuration: learning\_rate=0.001, hidden\_edim=310, hidden\_units=310, batch\_size=80, decay\_learning\_rate=0.99, SGD

System	Speed (per epoch)	Dev (BLEU4 )	Test (BLEU4)
nmt (sgd)	84s	29.6	31.4 (28800 step)
nmt <sup>+</sup> (sgd)	108s	33.0	38.1 (34200 step)
nmt <sup>+</sup> +mn (ada) _v0	180s	30.6	36.0 (71400 step)

Table 2: configuration: learning\_rate=0.5, hidden\_edim=310, hidden\_units=500, batch\_size=80, decay\_learning\_rate=0.99

system	rate=0.001		rate=0.005	
	dev	test	dev	test
nmt <sup>+</sup>	13.0	15.4 (26w epch)	33.0	38.1 (34200 step)
nmt <sup>+</sup> +mn	11.1	12.5 (30w epch)	30.6	36.0 (71400 step)

Table 3: results of different learning rate

translimit	1	2	10
trainer	Adadelta	Adadelta	SGD
nmt <sup>+</sup> +mn-v0	30.6-36.0	-	-
nmt <sup>+</sup> +mn-v1	30.0-35.3	29.8-36.6	29.9-37.3
nmt <sup>+</sup> +mn-v2	-	30.0-38.1	30.5-37.8

Table 4: comparison of different configurations of **palcehlder**, **trans-limit**, **training algorithm**.

nmt+  
 BLEU = 38.1, 74.8/47.5/31.6/20.5 (BP=0.978, ratio=0.979, hyp\_len=3715,  
 ref\_len=3796) (34200)

v0-1-Ada  
 BLEU = 36.0, 73.3/45.4/29.3/18.9 (BP=0.976, ratio=0.976, hyp\_len=3706,  
 ref\_len=3797) (71400)

v1-1-Ada  
 BLEU = 35.3, 72.1/43.5/28.0/18.7 (BP=0.986, ratio=0.986, hyp\_len=3786,  
 ref\_len=3838) (34600)

v1-2-Ada  
 BLEU = 36.6, 72.8/45.1/29.5/20.0 (BP=0.982, ratio=0.982, hyp\_len=3742,  
 ref\_len=3810) (103800)

v1-2-sgd  
 BLEU = 37.3, 73.8/45.8/30.4/20.3 (BP=0.981, ratio=0.981, hyp\_len=3708,  
 ref\_len=3781) (48500)

v1-10-Ada  
 BLEU = 36.7, 72.7/45.8/29.6/19.0 (BP=0.993, ratio=0.993, hyp\_len=3814,  
 ref\_len=3841) (15000)

v2-2-Ada  
 BLEU = 38.1, 74.0/46.5/31.0/21.1 (BP=0.984, ratio=0.984, hyp\_len=3770,  
 ref\_len=3832) (73000)

v2-2-sgd  
 BLEU = 37.8, 73.2/45.5/30.3/21.3 (BP=0.987, ratio=0.987, hyp\_len=3753,  
 ref\_len=3802) (50000)

v2-10-Ada  
 BLEU = 35.8, 73.3/45.0/28.9/18.2 (BP=0.987, ratio=0.987, hyp\_len=3723,  
 ref\_len=3772) (25000)

## 2.2 Impact of memory (2017-1-10)

### 2.2.1 Result Analysis

Results shown in Table 5. I also investigated the translations and got the following conclusions:

1. The nmt model usually generate some translation off the subject;
2. nmt+mn model can alleviate the above issue, but it overemphasize the memory word, leading to duplicate memory words;
3. Do have some right words in memory, but not selected.

system	BLEU	hyp-len	mem/hyp	memory				rnn			
				precision	mem/ref	mem	ref	precision	rnn/ref	rnn	ref
baseline	38.1	3715	0.54	89.2	0.65	2010	3115	57.8	0.56	1705	3094
v2-2-ada	38.1	3770	0.53	89.1	0.65	2005	3096	56.8	0.57	1765	3102
v2-2-sgd	37.8	3753	0.53	89.0	0.64	1998	3108	55.2	0.57	1755	3094
v2-10-ada	35.8	3723	0.77	84.1	0.86	2873	3326	36.6	0.28	850	3065

Table 5: Results Analysis. hyp-len: the total words of translation; mem/hyp: the ratio of the translation words hitting the memory against the total translation words; precision: the unigram precision that memory words against reference words; mem/ref: the ratios of the translation words hitting memory against the total selected reference words; mem: the number of translation words that hit memory; ref: the number of selected ref words

### 2.2.2 Memory Tailor

choose 3 entries in memory:

葡萄酒-wine 电视机-television 白兰地-brandy

system	test	details
Moses-o1	40.3	80.3/50.0/33.3/22.3, ratio=0.971, hyp_len=3720, ref_len=3833
Moses-o2	49.1	83.1/58.4/42.3/30.6, ratio=0.981, hyp_len=3664, ref_len=3736
Moses-o3	53.0	83.9/61.6/46.4/34.5, ratio=0.987, hyp_len=3756, ref_len=3807
nmt <sup>+</sup>	38.1	74.8/47.5/31.6/20.5, ratio=0.979, hyp_len=3715, ref_len=3796
v2-2-full	38.1	74.0/46.5/31.0/21.1, ratio=0.984, hyp_len=3770, ref_len=3832
v2-2-3entry	26.45	67.4/35.1/20.2/10.8, ratio=0.985, hyp_len=3707, ref_len=3762
v2-2-3entry-retrain	36.2	72.6/44.7/29.3/19.0, ratio=0.987, hyp_len=3773, ref_len=3824

### 2.2.3 Problem of Duplicate Words

To solve the problem of duplicate words and improve the fluency by

1. add  $y_{i-1}$  to the attention calculation for memory;
2. add  $c_i$  to the attention calculation for memory;

The results are shown in Table

system	test	details
v2+ $c_i$	36.0	72.8/44.8/28.9/19.4, ratio=0.978, hyp_len=3733, ref_len=3817
v2+ $y_{i-1}$	38.0	74.2/47.1/30.8/20.4, ratio=0.986, hyp_len=3777, ref_len=3831

Table 6: Results of adding  $c_i$  and  $y_{i-1}$  to the attention of memory