Converting Continuous-Space Language Models into *N*-gram Language Models for Statistical Machine Translation

Rui Wang †,‡, Masao Utiyama‡, Isao Goto ‡, Eiichro Sumita ‡, Hai Zhao †, Bao-Liang Lu †

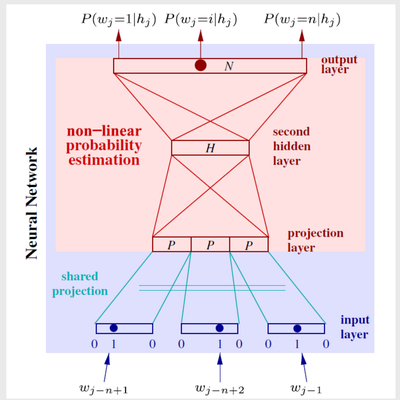
† Shanghai Jiao Tong University

‡ National Institute of Information and Communications Technology

Introduction

Neural network language models, or continuous-space language models (CSLMs), have been shown to improve the performance of statistical machine translation (SMT) when they are used for reranking n-best translations. However, CSLMs have not been used in the first pass decoding of SMT, because using CSLMs in decoding takes a lot of time. In contrast, we propose a method for converting CSLMs into back-off n-gram language models (BNLMs) so that we can use converted CSLMs in decoding. We show that they outperform the original BNLMs and are comparable with the traditional use of CSLMs in reranking with higher decoding efficiency.

CSLM



computational complexity of calculating the probabilities of all words is quite high

short-list, which consists of the most frequent words in the vocabulary (Schwenk, 2007;Schwenk, 2010)

$$P(w_i|h_i) = \begin{cases} \frac{P_c(w_i|h_i)}{1-P_c(o|h_i)} P_s(h_i) & if\ w_i \in short-list \\ P_b(w_i|h_i) & otherwise \end{cases}$$

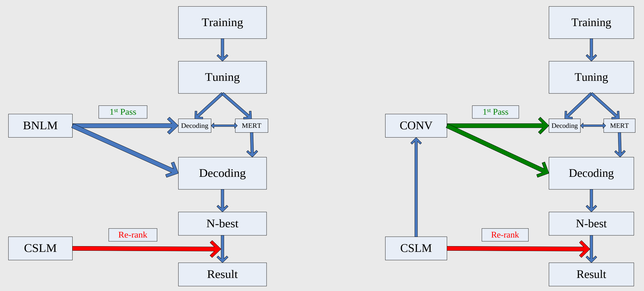
$$P_s(h_i) = \sum_{v \in short-list} P_b(v|h_i)$$

Why not CSLM in SMT decoding?

- 2000 NTCIR-9 English Sentences as test data.
- 5-gram CSLM (4 context words) and BNLM trained from the same 1 million NTCIR-9 English sentences.
- Evaluate the probability of every n-gram.
- Not including the time of converting text file to binary file of the input for CSLM.

LMS	CPU Time1	CPU Time2	CPU Time3
BNLM	3.241 s	4.044 s	4.404 s
CSLM	42.058 s	42.372 s	38.361 s

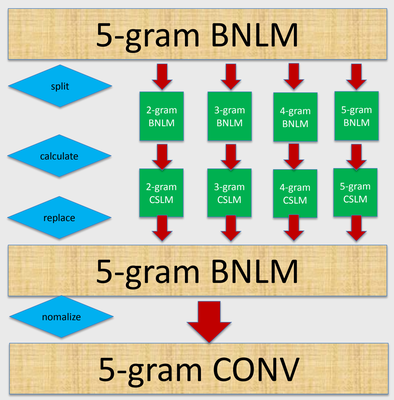
CSLM in SMT



Former Method

Our Method

Conversion Method



Experiments

- Corpus: The NTCIR-9 parallel training, development, and test data consisted of 1 million, 2,000, and 2,000 sentences, respectively.
- SMT features: five translation model scores, one word penalty score, seven distortion scores and one language model.
- Baseline: patent data for the Chinese to English patent translation subtask from the NTCIR-9 patent translation task.
- Vocabulary: the same and extracted from the 1 million training sentences.

LMs	*BNLM 42*	*BNLM 746*	*CONV 42*	*CONV 746*	*CSLM 42*
Words	42 million	746 million	42 million	746 million	42 million
Corpus (sentences)	1 million	26 million	1 million	26 million	1 million
Vocabulary	457k	457k	457k	457k	457k
smoothing	modified Kneser-Ney	modified Kneser-Ney	modified Kneser-Ney	Modified Kneser-Ney	NA
cutoffs	no	3; 4; 5-grams more than once	no	3; 4; 5-grams more than once	no
Other	*Input for CSLM42 into CONV42*	*Input for CSLM42 into CONV 746*	interpolated with BNLM42 by the weight maximizing the BLEU score on the dev data	interpolated with BNLM746 by the weight maximizing the BLEU score on the dev data	projection layer 256 hidden layer 384 output layer 8192.

Results

Language Models	1st Pass BLEU	1st Pass CPU Time	Rerank BLEU	1st Pass + Rerank CPU Time
BNLM42	31.60	3373.867 s	32.44	5057.725 s
CONV42	32.58	3431.200 s	32.98	5198.648 s
BNLM746	32.83	3510.888 s	33.36	5156.495 s
CONV746	33.22	3584.917 s	33.54	5276.182 s

Comparison of BLEU and decoding efficiency

LMs	BNLM746 Rerank	BNLM746 Rerank	CONV 746 1st Pass	BNLM746 1st Pass	CONV 42 1st Pass	BNLM42 Rerank	BNLM42 1st Pass
CONV746 Rerank	--	>>	>>	>>	>>	>>	>>
BNLM746 Rerank		--	>>	>	>>	>>	>>
CONV746 1st Pass			>>	--	>>	>>	>>
CONV42 Rerank				--	>>	>>	>>
BNLM746 1st Pass					--	>>	>>
CONV42 1st Pass						--	>>
BNLM42 Rerank							>>

- Significance test
- ">>":significantly better at $\alpha = 0.01$,
- ">": significantly better at $\alpha = 0.05$,
- "--": not significantly better at $\alpha = 0.05$