Adaptation Data Selection using Neural Language Models: Experiments in Machine Translation

Kevin Duh, Graham Neubig
Graduate School of Information Science
Nara Institute of Science and Technology
8916-5 Takayama, Ikoma, Japan
kevinduh@is.naist.jp
neubig@is.naist.jp

Katsuhito Sudoh, Hajime Tsukada
NTT Communication Science Labs.
NTT Corporation
2-4 Hikaridai, Seika, Kyoto, Japan
sudoh.katsuhito@lab.ntt.co.jp
tsukada.hajime@lab.ntt.co.jp

Abstract

Data selection is an effective approach to domain adaptation in statistical machine translation. The idea is to use language models trained on small in-domain text to select similar sentences from large general-domain corpora, which are then incorporated into the training data. Substantial gains have been demonstrated in previous works, which employ standard n-gram language models. Here, we explore the use of neural language models for data selection. We hypothesize that the continuous vector representation of words in neural language models makes them more effective than n-grams for modeling unknown word contexts, which are prevalent in general-domain text. In a comprehensive evaluation of 4 language pairs (English to German, French, Russian, Spanish), we found that neural language models are indeed viable tools for data selection: while the improvements are varied (i.e. 0.1 to 1.7 gains in BLEU), they are fast to train on small in-domain data and can sometimes substantially outperform conventional n-grams.

1 Introduction

A perennial challenge in building Statistical Machine Translation (SMT) systems is the dearth of high-quality bitext in the domain of interest. An effective and practical solution is *adaptation data selection*: the idea is to use language models (LMs) trained on in-domain text to select similar sentences from large general-domain corpora. The selected sentences are then incorporated into the SMT training data. Analyses have shown that this augmented data can lead to better statistical estimation or word coverage (Duh et al., 2010; Haddow and Koehn, 2012).

Although previous works in data selection (Axelrod et al., 2011; Koehn and Haddow, 2012; Yasuda et al., 2008) have shown substantial gains, we suspect that the commonly-used *n-gram* LMs may be sub-optimal. The small size of the in-domain text implies that a large percentage of general-domain sentences will contain words not observed in the LM training data. In fact, as many as 60% of general-domain sentences contain at least one unknown word in our experiments. Although the LM probabilities of these sentences could still be computed by resorting to back-off and other smoothing techniques, a natural question remains: will alternative, more robust LMs do better?

We hypothesize that the *neural language model* (Bengio et al., 2003) is a viable alternative, since its continuous vector representation of words is well-suited for modeling sentences with frequent unknown words, providing smooth probability estimates of unseen but similar contexts. Neural LMs have achieved positive results in speech recognition and SMT reranking (Schwenk et al., 2012; Mikolov et al., 2011a). To the best of our knowledge, this paper is the first work that examines neural LMs for adaptation data selection.

2 Data Selection Method

We employ the data selection method of (Axelrod et al., 2011), which builds upon (Moore and Lewis, 2010). The intuition is to select general-domain sentences that are similar to in-domain text, while being dis-similar to the average general-domain text.

To do so, one defines the score of an general-domain sentence pair (e, f) as:

$$[IN_E(e) - GEN_E(e)] + [IN_F(f) - GEN_F(f)] \tag{1}$$

where $IN_E(e)$ is the *length-normalized* cross-entropy of e on the English in-domain LM. $GEN_E(e)$ is the length-normalized cross-entropy

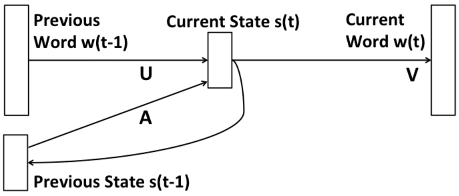

Figure 1: Recurrent neural LM.

of e on the English general-domain LM, which is built from a sub-sample of the general-domain text. Similarly, $IN_F(f)$ and $GEN_F(f)$ are the cross-entropies of f on Foreign-side LM. Finally, sentence pairs are ranked according to Eq. 1 and those with scores lower than some empirically-chosen threshold are added to the bitext for translation model training.

2.1 Neural Language Models

The four LMs used to compute Eq. 1 have conventionally been n-grams. N-grams of the form $p(w(t)|w(t-1), w(t-2), \ldots)$ predict words by using multinomial distributions conditioned on the context $(w(t-1), w(t-2), \ldots)$. But when the context is rare or contains unknown words, n-grams are forced to back-off to lower-order models, e.g. $p(w(t)|w(t-1))$. These backoffs are unfortunately very frequent in adaptation data selection.

Neural LMs, in contrast, model word probabilities using continuous vector representations. Figure 1 shows a type of neural LMs called recurrent neural networks (Mikolov et al., 2011b).[1] Rather than representing context as an identity (n-gram hit-or-miss) function on $[w(t-1), w(t-2), \ldots]$, neural LMs summarize the context by a hidden state vector $s(t)$. This is a continuous vector of dimension $|S|$ whose elements are predicted by the previous word $w(t-1)$ and previous state $s(t-1)$. This is robust to rare contexts because continuous representations enable *sharing* of statistical strength between similar contexts. Bengio (2009) shows that such representations are better than multinomials in alleviating sparsity issues.

Now, given state vector $s(t)$, we can predict the probability of the current word. Figure 1 is expressed formally in the following equations:

$$w(t) = [w_0(t), \ldots, w_k(t), \ldots w_{|W|}(t)] \tag{2}$$

$$w_k(t) = g\left(\sum_{j=0}^{|S|} s_j(t) V_{kj}\right) \tag{3}$$

$$s_j(t) = f\left(\sum_{i=0}^{|W|} w_i(t-1) U_{ji} + \sum_{i'=0}^{|S|} s_{i'}(t-1) A_{ji'}\right) \tag{4}$$

Here, $w(t)$ is viewed as a vector of dimension $|W|$ (vocabulary size) where each element $w_k(t)$ represents the probability of the k-th vocabulary item at sentence position t. The function $g(z_k) = e^{z_k} / \sum_k e^{z_k}$ is a softmax function that ensures the neural LM outputs are proper probabilities, and $f(z) = 1/(1 + e^{-z})$ is a sigmoid activation that induces the non-linearity critical to the neural network's expressive power. The matrices V, U, and A are trained by maximizing likelihood on training data using a ”backpropagation-through-time” method.[2] Intuitively, U and A compress the context ($|S| < |W|$) such that contexts predictive of the same word $w(t)$ are close together.

Since proper modeling of unknown contexts is important in our problem, training text for both n-gram and neural LM is pre-processed by converting all low-frequency words in the training data (frequency=1 in our case) to a special ”unknown” token. This is used only in Eq. 1 for selecting general-domain sentences; these words retain their surface forms in the SMT train pipeline.

3 Experiment Setup

We experimented with four language pairs in the WIT^3 corpus (Cettolo et al., 2012), with English (en) as source and German (de), Spanish (es), French (fr), Russian (ru) as target. This is the in-domain corpus, and consists of TED Talk transcripts covering topics in technology, entertainment, and design. As general-domain corpora, we collected bitext from the WMT2013 campaign, including CommonCrawl and NewsCommentary for all 4 languages, Europarl for de/es/fr, UN for es/fr, Gigaword for fr, and Yandex for ru. The in-domain data is divided into a training set (for SMT

[1] Another major type of neural LMs are the so-called feed-forward networks (Bengio et al., 2003; Schwenk, 2007; Nakamura et al., 1990). Both types of neural LMs have seen many improvements recently, in terms of computational scalability (Le et al., 2011) and modeling power (Arisoy et al., 2012; Wu et al., 2012; Alexandrescu and Kirchhoff, 2006). We focus on recurrent networks here since there are fewer hyper-parameters and its ability to model infinite context using recursion is theoretically attractive. But we note that feed-forward networks are just as viable.

[2] The recurrent states are unrolled for several time-steps, then stochastic gradient descent is applied.

	en-de	en-es	en-fr	en-ru
In-domain Training Set				
#sentence	129k	140k	139k	117k
#token (en)	2.5M	2.7M	2.7M	2.3M
#vocab (en)	26k	27k	27k	25k
#vocab (f)	42k	39k	34k	58k
General-domain Bitext				
#sentence	4.4M	14.7M	38.9M	2.0M
#token (en)	113M	385M	1012M	51M
%unknown	60%	58%	64%	65%

Table 1: Data statistics. ”%unknown”=fraction of general-domain sentences with unknown words.

pipeline and neural LM training), a tuning set (for MERT), a validation set (for choosing the optimal threshold in data selection), and finally a testset of 1616 sentences.[3] Table 1 lists data statistics.

For each language pair, we built a baseline **in-data** SMT system trained only on in-domain data, and an **alldata** system using combined in-domain and general-domain data.[4] We then built 3 systems from augmented data selected by different LMs:

- **ngram**: Data selection by 4-gram LMs with Kneser-Ney smoothing (Axelrod et al., 2011)
- **neuralnet**: Data selection by Recurrent neural LM, with the RNNLM Toolkit.[5]
- **combine**: Data selection by interpolated LM using n-gram & neuralnet (equal weight).

All systems are built using standard settings in the Moses toolkit (GIZA++ alignment, grow-diag-final-and, lexical reordering models, and SRILM). Note that standard n-grams are used as LMs for SMT; neural LMs are only used for data selection. Multiple SMT systems are trained by thresholding on $\{$10k,50k,100k,500k,1M$\}$ general-domain sentence subsets, and we empirically determine the single system for testing based on results on a separate validation set (in practice, 500k was chosen for fr and 1M for es, de, ru.).

4 Results

4.1 LM Perplexity and Training Time

First, we measured perplexity to check the generalization ability of our neural LMs as language models. Recall that we train four LMs to compute each of the components of Eq. 1. In Table 2, we compared each of the four versions of **ngram**, **neuralnet**, and **combine** LMs on in-domain test sets or general-domain held-out sets. It re-affirms previous positive results (Mikolov et al., 2011a), with **neuralnet** outperforming **ngram** by 20-30% perplexity across all tasks. Also, **combine** slightly improves the perplexity of **neuralnet**.

Task	ngram	neuralnet	combine
In-Domain Test Set			
en-de de	157	110 (29%)	110 (29%)
en-de en	102	81 (20%)	78 (24%)
en-es es	129	102 (20%)	98 (24%)
en-es en	101	80 (21%)	77 (24%)
en-fr fr	90	67 (25%)	65 (27%)
en-fr en	102	80 (21%)	77 (24%)
en-ru ru	208	167 (19%)	155 (26%)
en-ru en	103	83 (19%)	79 (23%)
General-Domain Held-out Set			
en-de de	234	174 (25%)	161 (31%)
en-de en	218	168 (23%)	155 (29%)
en-es es	62	43 (31%)	43 (31%)
en-es en	84	61 (27%)	59 (30%)
en-fr fr	64	43 (33%)	43 (33%)
en-fr en	95	67 (30%)	65 (32%)
en-ru ru	242	199 (18%)	176 (27%)
en-ru en	191	153 (20%)	142 (26%)

Table 2: Perplexity of various LMs. Number in parenthesis is percentage improvement vs. ngram.

Second, we show that the usual concern of neural LM training time is not so critical for the in-domain data sizes used domain adaptation. The complexity of training Figure 1 is dominated by computing Eq. 3 and scales as $O(|W| \times |S|)$ in the number of tokens. Since $|W|$ can be large, one practical trick is to cluster the vocabulary so that the output dimension is reduced. Table 3 shows the training times on a 3.3GHz XeonE5 CPU by varying these two main hyper-parameters ($|S|$ and cluster size). Note that the setting $|S| = 200$ and cluster size of 100 already gives good perplexity in reasonable training time. All neural LMs in this paper use this setting, without additional tuning.

[3] The original data are provided by http://wit3.fbk.eu and http://www.statmt.org/wmt13/. Our domain adaptation scenario is similar to the IWSLT2012 campaign but we used our own random train/test splits, since we wanted to ensure the testset for all languages had identical source sentences for comparison purposes. For replicability, our software is available at http://cl.naist.jp/∼kevinduh/a/acl2013.

[4] More advanced phrase table adaptation methods are possible. but our interest is in comparing data selection methods. The conclusions should transfer to advanced methods such as (Foster et al., 2010; Niehues and Waibel, 2012).

[5] http://www.fit.vutbr.cz/∼imikolov/rnnlm/

$\lvert S\rvert$	Cluster	Time	Perplexity
200	100	198m	110
100	$\lvert W\rvert$	12915m	110
200	400	208m	113
100	100	52m	118
100	400	71m	120

Table 3: Training time (in minutes) for various neural LM architectures (Task: en-de de).

4.2 End-to-end SMT Evaluation

Table 4 shows translation results in terms of BLEU (Papineni et al., 2002), RIBES (Isozaki et al., 2010), and TER (Snover et al., 2006). We observe that all three data selection methods essentially outperform **alldata** and **indata** for all language pairs, and **neuralnet** tend to be the best in all metrics. E.g., BLEU improvements over **ngram** are in the range of 0.4 for en-de, 0.5 for en-es, 0.1 for en-fr, and 1.7 for en-ru. Although not all improvements are large in absolute terms, many are statistically significant (95% confidence).

We therefore believe that neural LMs are generally worthwhile to try for data selection, as it rarely underperform n-grams. The open question is: what can explain the significant improvements in, for example Russian, Spanish, German, but the lack thereof in French? One conjecture is that neural LMs succeeded in lowering testset out-of-vocabulary (OOV) rate, but we found that OOV reduction is similar across all selection methods.

The improvements appear to be due to *better probability estimates* of the translation/reordering models. We performed a diagnostic by decoding the testset using LMs trained on the same testset, while varying the translation/reordering tables with those of **ngram** and **neuralnet**; this is a kind of pseudo forced-decoding that can inform us about which table has better coverage. We found that across all language pairs, BLEU differences of translations under this diagnostic become insignificant, implying that the raw probability value is the differentiating factor between **ngram** and **neuralnet**. Manual inspection of en-de revealed that many improvements come from lexical choice in morphological variants (”meinen Sohn” vs. ”mein Sohn”), segmentation changes (”baking soda” → ”Backpulver” vs. ”baken Soda”), and handling of unaligned words at phrase boundaries.

Finally, we measured the intersection between the sentence set selected by **ngram** vs **neuralnet**. They share 60-75% of the augmented training data. This high overlap means that **ngram** and **neuralnet** are actually *not* drastically different systems, and **neuralnet** with its slightly better selections represent an *incremental* improvement.[6]

Task	System	BLEU	RIBES	TER
en-de	indata	20.8	80.1	59.0
	alldata	21.5	80.1	59.1
	ngram	21.5	80.3	58.9
	neuralnet	**21.9**$^{+}$	**80.5**$^{+}$	**58.4**
	combine	21.5	80.2	58.8
en-es	indata	30.4	83.5	48.7
	alldata	31.2	83.2	49.9
	ngram	32.0	83.7	48.4
	neuralnet	**32.5**$^{+}$	83.7	**48.3**$^{+}$
	combine	**32.5**$^{+}$	**83.8**	**48.3**$^{+}$
en-fr	indata	31.4	83.9	51.2
	alldata	31.5	83.5	51.4
	ngram	32.7	83.7	50.4
	neuralnet	**32.8**	**84.2**$^{+}$	**50.3**
	combine	32.5	84.0	50.5
en-ru	indata	14.8	72.5	69.5
	alldata	23.4	75.0	62.3
	ngram	24.0	75.7	61.4
	neuralnet	**25.7**$^{+}$	**76.1**	**60.0**$^{+}$
	combine	23.7	75.9	61.9^{-}

Table 4: End-to-end Translation Results. The best results are bold-faced. We also compare neural LMs to ngram using pairwise bootstrap (Koehn, 2004): ”+” means statistically significant improvement and ”−” means significant degradation.

5 Conclusions

We perform an evaluation of neural LMs for adaptation data selection, based on the hypothesis that their continuous vector representations are effective at comparing general-domain sentences, which contain frequent unknown words. Compared to conventional n-grams, we observed end-to-end translation improvements from 0.1 to 1.7 BLEU. Since neural LMs are fast to train in the small in-domain data setting and achieve equal or incrementally better results, we conclude that they are an worthwhile option to include in the arsenal of adaptation data selection techniques.

[6] This is corroborated by another analysis: taking the *union* of sentences found by **ngram** and **neuralnet** gives similar BLEU scores as **neuralnet**.

Acknowledgments

We thank Amittai Axelrod for discussions about data selection implementation details, and an anonymous reviewer for suggesting the *union* idea for results analysis. K. D. would like to credit Spyros Matsoukas (personal communication, 2010) for the trick of using LM-based pseudo forced-decoding for error analysis.

References

Andrei Alexandrescu and Katrin Kirchhoff. 2006. Factored neural language models. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, NAACL-Short '06, pages 1–4, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ebru Arisoy, Tara N. Sainath, Brian Kingsbury, and Bhuvana Ramabhadran. 2012. Deep neural network language models. In *Proceedings of the NAACL-HLT 2012 Workshop: Will We Ever Really Replace the N-gram Model? On the Future of Language Modeling for HLT*, pages 20–28, Montréal, Canada, June. Association for Computational Linguistics.

Amittai Axelrod, Xiaodong He, and Jianfeng Gao. 2011. Domain adaptation via pseudo in-domain data selection. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 355–362, Edinburgh, Scotland, UK., July. Association for Computational Linguistics.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language models. *JMLR*.

Yoshua Bengio. 2009. *Learning Deep Architectures for AI*, volume Foundations and Trends in Machine Learning. NOW Publishers.

Mauro Cettolo, Christian Girardi, and Marcello Federico. 2012. Wit^3: Web inventory of transcribed and translated talks. In *Proceedings of the 16th Conference of the European Association for Machine Translation (EAMT)*, pages 261–268, Trento, Italy, May.

Kevin Duh, Katsuhito Sudoh, and Hajime Tsukada. 2010. Analysis of translation model adaptation for statistical machine translation. In *Proceedings of the International Workshop on Spoken Language Translation (IWSLT) - Technical Papers Track*.

George Foster, Cyril Goutte, and Roland Kuhn. 2010. Discriminative instance weighting for domain adaptation in statistical machine translation. In *EMNLP*.

Barry Haddow and Philipp Koehn. 2012. Analysing the effect of out-of-domain data on smt systems. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 422–432, Montréal, Canada, June. Association for Computational Linguistics.

Hideki Isozaki, Tsutomu Hirao, Kevin Duh, Katsuhito Sudoh, and Hajime Tsukada. 2010. Automatic evaluation of translation quality for distant language pairs. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 944–952, Cambridge, MA, October. Association for Computational Linguistics.

Philipp Koehn and Barry Haddow. 2012. Towards effective use of training data in statistical machine translation. In *WMT*.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *EMNLP*.

Hai-Son Le, I. Oparin, A. Allauzen, J. Gauvain, and F. Yvon. 2011. Structured output layer neural network language model. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pages 5524–5527.

Tomáš Mikolov, Anoop Deoras, Daniel Povey, Lukáš Burget, and Jan Černocký. 2011a. Strategies for training large scale neural network language model. In *ASRU*.

Tomáš Mikolov, Stefan Kombrink, Lukáš Burget, Jan Černocký, and Sanjeev Khudanpur. 2011b. Extensions of recurrent neural network language model. In *Proceedings of the 2011 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*.

Robert C. Moore and William Lewis. 2010. Intelligent selection of language model training data. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 220–224, Uppsala, Sweden, July. Association for Computational Linguistics.

Masami Nakamura, Katsuteru Maruyama, Takeshi Kawabata, and Kiyohiro Shikano. 1990. Neural network approach to word category prediction for english texts. In *Proceedings of the 13th conference on Computational linguistics - Volume 3*, COLING '90, pages 213–218, Stroudsburg, PA, USA. Association for Computational Linguistics.

Jan Niehues and Alex Waibel. 2012. Detailed analysis of different strategies for phrase table adaptation in SMT. In *AMTA*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *ACL*.

Holger Schwenk, Anthony Rousseau, and Mohammed Attik. 2012. Large, pruned or continuous space language models on a gpu for statistical machine translation. In *Proceedings of the NAACL-HLT 2012 Workshop: Will We Ever Really Replace the N-gram Model? On the Future of Language Modeling for*

HLT, pages 11–19, Montréal, Canada, June. Association for Computational Linguistics.

Holger Schwenk. 2007. Continuous space language models. *Comput. Speech Lang.*, 21(3):492–518, July.

M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *AMTA*.

Youzheng Wu, Xugang Lu, Hitoshi Yamamoto, Shigeki Matsuda, Chiori Hori, and Hideki Kashioka. 2012. Factored language model based on recurrent neural network. In *Proceedings of COLING 2012*, pages 2835–2850, Mumbai, India, December. The COLING 2012 Organizing Committee.

Keiji Yasuda, Ruiqiang Zhang, Hirofumi Yamamoto, and Eiichiro Sumita. 2008. Method of selecting training data to build a compact and efficient translation model. In *ICJNLP*.