Bilingual Methods for Adaptive Training Data Selection for Machine Translation

Boxing Chen Boxing.Chen@nrc-cnrc.gc.ca
Roland Kuhn Roland.Kuhn@nrc-cnrc.gc.ca
George Foster George.Foster@nrc-cnrc.gc.ca
Colin Cherry Colin.Cherry@nrc-cnrc.gc.ca
National Research Council Canada, Ottawa, ON, Canada

Fei Huang feihuang@fb.com
Facebook, New York, NY, USA

Abstract

In this paper, we propose a new data selection method which uses semi-supervised convolutional neural networks based on bitokens (Bi-SSCNNs) for training machine translation systems from a large bilingual corpus. In earlier work, we devised a data selection method based on semi-supervised convolutional neural networks (SSCNNs). The new method, Bi-SSCNN, is based on bitokens, which use bilingual information. When the new methods are tested on two translation tasks (Chinese-to-English and Arabic-to-English), they significantly outperform the other three data selection methods in the experiments. We also show that the Bi-SSCNN method is much more effective than other methods in preventing noisy sentence pairs from being chosen for training. More interestingly, this method only needs a tiny amount of in-domain data to train the selection model, which makes fine-grained topic-dependent translation adaptation possible. In the follow-up experiments, we find that neural machine translation (NMT) is more sensitive to noisy data than statistical machine translation (SMT). Therefore, Bi-SSCNN which can effectively screen out noisy sentence pairs, can benefit NMT much more than SMT.We observed a BLEU improvement over 3 points on an English-to-French WMT task when Bi-SSCNNs were used.

1 Introduction

When building a statistical machine translation (SMT) system, it is important to choose bilingual training data that are of high quality [1] and that are typical of the domain in which the SMT system will operate. In previous work, these two goals of data selection, i.e., picking high-quality data and picking data that ensure the SMT system is well-adapted to a given domain, have often been achieved separately. For instance, the papers (Munteanu and Marcu, 2005; Khadivi and Ney, 2005; Okita et al., 2009; Jiang et al., 2010; Denkowski et al., 2012) focus on reducing the noise in the data. They use different scoring functions, such as language model perplexity, word alignment score, or IBM model 1 score, to score each sentence pair, top scored sentence pairs are selected. While the papers (Zhao et al., 2004; Lü et al., 2007; Yasuda et al., 2008; Moore and Lewis, 2010; Axelrod et al., 2011; Duh et al., 2013; Axelrod et al., 2015) focus on domain adaptation. They all select monolingual or bilingual data that

[1] Within each sentence pair, the target-language sentence is a good translation of the source sentence

are similar to the in-domain data according to some criterion. These state-of-the-art adaptive data selection approaches (Axelrod et al., 2011; Duh et al., 2013; Axelrod et al., 2015) search for bilingual parallel sentences using the difference in language model perplexity between two language models trained on in-domain and out-domain data, respectively. Furthermore, (Duh et al., 2013) extends these approaches from n-gram models to recurrent neural network language models (Mikolov et al., 2010). While some previous work considers achieving the two goals simultaneously, such as (Mansour et al., 2011) which uses IBM model 1 and a language model to do data selection, (Durrani et al., 2015) uses a neural network joint model to select the in-domain data.

In a recent paper (Chen and Huang, 2016), we describe one type of neural network for carrying out data selection: a semi-supervised Convolutional Neural Network (SSCNN) that is trained on the in-domain set to score one side of each sentence in a general-domain bilingual corpus (either the source side or the target side) for its suitability as training material for an SMT system. The highest-scoring sentence pairs are chosen to train the SMT system. Experiments described in that paper, covering three different types of test domain and four language directions, show that this SSCNN method yields significantly higher BLEU scores for the resulting SMT system than for three state-of-the-art data selection methods when the amount of training data selected is held constant. The advantage of the SSCNN over the earlier methods is especially dramatic when the amount of in-domain data used to train the selection model is small (less than 800 sentence pairs): the in-domain set for the SSCNN can be as few as 100 sentences, which makes fine-grained topic-dependent translation adaptation possible. In some cases, the SSCNN is so effective at selecting good training data that it is possible to greatly reduce the amount of training data for the SMT system without negative impact on translation quality: this reduces the footprint of the system, which can be advantageous for many practical applications.

In the experiments for (Chen and Huang, 2016), we found that the best variant of the SSCNN method was one in which we trained two CNN models - one that scores source-language sentences and one that scores target-language sentences - and sum the scores of these two models to get the overall score of each sentence pair in the bilingual corpus. Thus, the SSCNN variant we used assigns high scores to sentence pairs where both the source-language sentence and its target-language partner resemble sentences in the in-domain corpus. Note what this method does **not** do: it does not check that the target sentence is a good translation of the source sentence. Essentially, it scores the extent to which both the source and target sentence are in-domain, but does not in any way penalize bad translations. We say that such a method is “symmetric”: it incorporates equal amounts of information from the source and the target language, but it is not “bilingual”: it does not incorporate information about the quality of translations.

The main motivation for this paper is to explore CNN-based data selection techniques that are bilingual. It is based on semi-supervised CNNs that use bitokens as units instead of source or target words (Marino et al., 2006; Niehues et al., 2011). For the bitoken semi-supervised CNN, we should use the abbreviation “Bi-SSCNN”. We also experiment with the bilingual method that combines IBM model 1 and language model (LM) scores and neural network joint model.

In this paper, we carried out experiments reported on two language pairs: Chinese-to-English and Arabic-to-English. We fix the number of training sentences to be chosen for the data selection techniques so that they can be fairly compared, and measure the BLEU score on test data from the resulting MT systems. It turns out that three techniques have roughly the same performance in terms of BLEU: the symmetric but non-bilingual word-based SSCNN method, and two symmetric, bilingual techniques - the simple IBM-LM method, and the NNJM method. The Bi-SSCNN method, on the other hand, outperforms all other methods: by +0.5 BLEU for

Chinese-to-English task, and by +0.3 BLEU for Arabic-to-English task.

Because the main motivation for the paper is exploration of bilingual methods for SSCNN-based data selection, we perform another set of experiments to see how good each method is at rejecting sentence pairs whose target side is not a translation of the source side. We do this by permuting 50% of the pairs in the bilingual corpus. We then count the proportion of pairs chosen by each method that are mismatched. In these experiments, all three bilingual methods - IBM-LM, NNJM, and Bi-SSCNN outperform the other methods (they chose a smaller proportion of mismatched sentence pairs) and Bi-SSCNN is even more effective than the other two bilingual methods.

2 Four Data Selection Methods

In this section, we focus on four methods for data selection: the IBM-LM method, the NNJM method, the original word-based SSCNN method, and the Bi-SSCNN method. The IBM-LM method is similar to Mansour et al. (2011), which is a simple bilingual method that is a good baseline for other bilingual methods. The NNJM-based data selection (Durrani et al., 2015) is the first bilingual NN method. The word-based SSCNN method is described in (Chen and Huang, 2016). The SSCNN method is symmetrical but not bilingual. The Bi-SSCNN method is a newly proposed method, which is symmetrical as well as bilingual.

2.1 Data Selection with IBM1 and Language Models

The IBM-LM method is straightforward, since state-of-the-art methods use IBM models to measure the mutual translation quality, and language models to select in-domain data. We combine length-normalized scores from these models into a global score, i.e., target-given-source IBM model 1 score, source-given-target IBM model 1 score, source language model cross entropy difference, and target language model cross entropy difference, are normalized by the corresponding sentence length and then averaged to obtain one score. The IBM models are trained on whole general-domain data plus the in-domain set, while the language models are trained on the in-domain set or a small (equal size to the in-domain set), random subset of the general-domain corpus (excluding the small in-domain set).

To select a subset of data to be used for training an SMT system from a bilingual corpus, the user must specify the number N of sentence pairs to be chosen. The N sentence pairs with the highest global scores $S(s,t)$ will be selected. This method is symmetrical - the roles of the source-language and target-language sides of the corpus are the same - and bilingual, because the IBM model 1 measures the degree to which each target sentence t is a good translation of its partner s, and vice versa.

2.2 Data Selection with Neural Net Joint Model (NNJM)

The Neural Network Joint Model (NNJM), as described in (Devlin et al., 2014), is a joint language and translation model based on a feedforward neural net (NN). It incorporats a wide span of contextual information from the source sentence, in addition to the traditional n-gram information from preceding target-language words. Specifically, when scoring a target word w_i, the NNJM inputs not only the $n-1$ preceding words $w_{i-n+1},...,w_{i-1}$, but also $2m+1$ source words: the source word s_i most closely aligned with w_i along with the m source words $s_{i-m},...,s_{i-1}$ to the left of s_i and the m source words $s_{i+1},...,s_{i+m}$ to the right of s_i.

The NNJMs used in our experiments input 4-grams on the target side and windows of size 11 on the source side (the aligned word, along with 5 words to the left of it and 5 words on its right). Training is done in two passes: a pass over general-domain data plus the in-domain set, followed by a pass over the in-domain set. While in the second pass, the source and target word embeddings are fixed. For a particular language pair, for instance, Chinese-to-English, four NNJMs are trained. There are two NNJMs informed by a wide-source language (Chinese) context that act as LMs for the target language (English): a positive one NNJM(+,Chinese-to-English) modeling in-domain target-language (English) sentences, and a negative one NNJM(-,Chinese-to-English) modeling out-of-domain target-language (English) sentences. Both are initialized with parameters and fixed source and target word embeddings which are learned on the entire general-domain corpus plus the in-domain set. The second phase of training for NNJM(+,Chinese-to-English) is on the in-domain set. It would be nice if we could train NNJM(-,Chinese-to-English) on sentence pairs that are known to be out-of-domain, but there is no easy way of obtaining such sentence pairs, so we simply carry out the second phase of training for this negative NNJM on a small, random subset of the general-domain corpus (excluding the small in-domain set). The difference between the two scores, score(NNJM(+,Chinese-to-English)) minus score(NNJM(-,Chinese-to-English)), as calculated on a sentence pair is an indicator of how close to the in-domain set the sentence pair is.

We train in similar manner a positive and a negative NNJM that act as LMs for the source language (Chinese) while consulting a wide context in the target language (English): NNJM(+,English to Chinese) and NNJM(-,English to Chinese). The global, symmetrical NNJM score S for a sentence pair is made up of equal contributions from these four models.

Since this metric contains information about the translation relationship between each source sentence and its target counterpart, and since the ways in which the source and target languages are used are mirror images of each other, the NNJM data selection method is both bilingual and symmetrical.

2.3 Data Selection with Semi-Supervised CNN

As described in more detail in (Chen and Huang, 2016), we were inspired by the success of convolutional neural networks (CNNs) applied to image and text classification (Krizhevsky et al., 2012; Kim, 2014; Johnson and Zhang, 2015a,b) to use CNNs to classify training sentences in either the source language or the target language as in-domain or out-of-domain.

Convolutional neural networks (CNNs) (LeCun and Bengio, 1998) are feed-forward neural networks that exploit the internal structure of data through convolutional and pooling layers; each computation unit of the convolutional layer processes a small region of the input data. When CNNs are applied to text input, the convolution layers process small regions of a document, i.e., a sequence of words. CNNs are now used in many text classification tasks (Kalchbrenner et al., 2014; Johnson and Zhang, 2015b; Wang et al., 2015). Chen and Huang (2016) use CNNs to classify sentence pairs to in-domain and out-of-domain sentence pairs.

In many of these studies, the first layer of the network converts words to word embeddings using table lookup; the embeddings are sometimes pre-trained on an unnlabeled data. The embeddings remain fixed during subsequent model training. A CNN trained with small number of labled data and pre-trained word embeddings on large unlabeled data is termed "semi-supervised". Because we were interested in data selection scenarios where only small amounts of in-domain data are available, we chose to use semi-supervised CNNs (SSCNNs) with $word2vec$ embeddings (Mikolov et al., 2013) pre-trained on a large general-domain, monolingual corpus. The input region vector that represents a segment of data can be either a concatenation of word vectors, in which the order of concatenation is the same as the word order in the sentence, or it can be a bag-of-word/n-gram vector. The bag-of-word (BOW) representation loses word order information but is more robust to data sparsity. A CNN whose input being BOW representation is called bow-CNN while input with concatenation of vectors is called seq-CNN.

On the other hand, the input sentence can also be represented with one-hot vectors, where each vector's length is the vocabulary size, value 1 at index i indicates word i appears in the

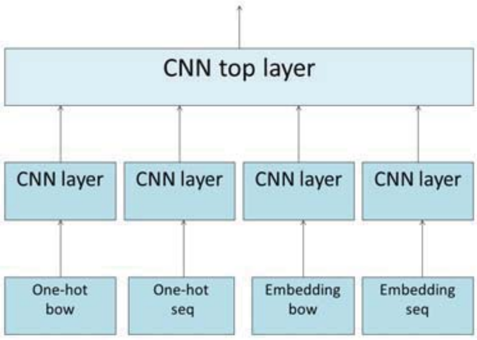

Figure 1: Semi-supervised CNN structure.

sentence, and 0 indicates its absence. The one-hot vector can be either one-hot "BOW" or one-hot "seq" too. We input all four kinds of representations, i.e., bag-of-word one-hot vectors (one-hot BOW), concatenation of one-hot vectors (one-hot seq), bag-of-word embedding vectors (embedding BOW), concatenation of embedding vectors (embedding seq), to the CNN layers to train the classification model, as shown in Figure 1.

To train the CNN itself, we take the in-domain set as the positive training sample and randomly select the same number of sentences from the general-domain training data as the negative training sample. The positively and negatively labeled data and the word embeddings are fed to the convolution layer to train the final classification model. The resulting SSCNN is then used to score each sentence in the general-domain corpus. As with the other methods described above, the SSCNN was applied symmetrically - two SSCNNs were trained, one on the source-language half of the in-domain set, the other on the target-language half, and their scores were summed to obtain a global score for each sentence pair in the big general-domain corpus. The top N sentence pairs are selected to train the SMT system. Note that though this SSCNN method is symmetrical, it is not bilingual: there is no evaluation of whether the two halves of each sentence pair are good translations of each other.

The experimental results given in (Chen and Huang, 2016) for the SSCNN method on four different language directions and a variety of genres (SMS, tweets, Facebook posts, etc) show that the resulting SMT systems typically outperform baseline systems trained with all the general-domain data, and (for a fixed amount of selected training data) previously state-of-the-art data selection methods. The advantage of the SSCNN method over other data selection techniques is especially strong when the size of the initial in-domain data (the dev set) is small. Therefore, if we wish to build a large scale topic-specific MT system with hundreds of topics, we only need to collect a few hundreds sentence pairs for each topic. This makes fine-grained topic-dependent translation adaptation possible.

2.4 Data Selection with a Bitoken Semi-supervised CNN

Despite the excellent performance of the SSCNN method described in the previous paragraphs, we believed that further improvement might be possible if we devised a method based on SSCNNs that was not only symmetrical, but also bilingual. The problem with SSCNN is that unlike the IBM-LM method or the NNJM method nothing about it filters out sentence pairs whose source and target halves are bad translations of each other. We decided to experiment with SSCNNs that take as input the bitokens of (Marino et al., 2006; Niehues et al., 2011).

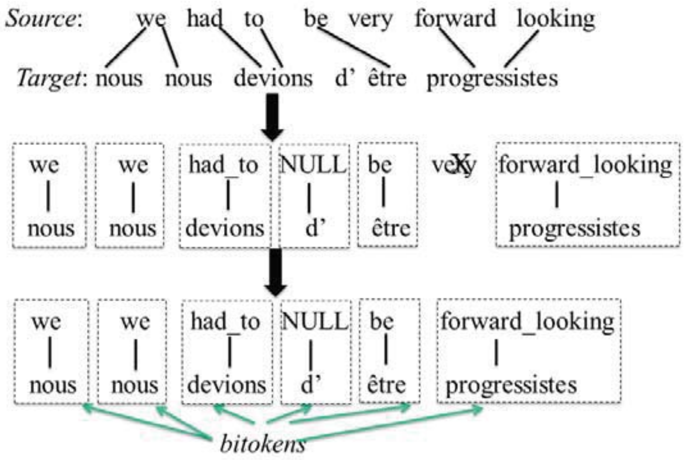

Figure 2: Bitoken sequence.

The paper (Niehues et al., 2011) describes a "bilingual language model" (biLM): the idea that SMT systems would benefit from wider contextual information from the source sentence. BiLMs provide this context by aligning each target word in the training data with source words to create bitokens. An n-gram bitoken LM for the sequence of target words is then trained.

Figure 2 (taken from (Stewart et al., 2014)) shows how a bitoken sequence is obtained from a word-aligned sentence pair for the English to French language pair. Unaligned target words (e.g., French word "d´' in the example) are aligned with NULL. Unaligned source words (e.g., "very") are dropped. A source word aligned with more than one target word (e.g., "we") aligned with two instances of "nous" is duplicated: each target word aligned with it receives a copy of that source word.

The word embeddings for bitokens are learned directly by word2vec, treating each bitoken as a word. For instance, in the French sentence shown in Figure 2, there are two occurrences of the new target-language word *nous/we*, one occurrence of *devions/had_to*, etc. To reduce the size of the bitoken vocabulary, we exclude low-frequency bitokens from consideration: to be taken into account, a Chinese-English bitoken must occur at least 10 times in the training data, and an Arabic-English bitoken must occur at least 5 times.

After this embedding has been performed, the Bi-SSCNN is trained similarly as was described above for the original word-based SSCNN. The only difference is the pooling strategy. For the word-based SSCNN, we use maximum-pooling, while for the bitoken-based SSCNN, we use average-pooling. This is because if a sentence contains one or more in-domain words, it is very likely an in-domain sentence, so we use maximum pooling to highlight those typical in-domain words for the word-based SSCNN. The reason for choosing average-pooling for Bi-SSCNN is that if a sentence pair contains one or more in-domain bitokens, it is not necessarily a good quality in-domain sentence pair. It is only when a sentence pair contains some in-domain bitokens, and most of the bitokens are good translations, that it is considered to be a high quality in-domain sentence pair.

language	zh2en	ar2en
test domain	webforum	nw&wl
train size	12.20M	5.29M
dev size	2,748	1,360
test size	1,224	5,812

Table 1: Summary of the data. Data is given as the number of sentence pairs, "M" represents "million". "nw" stands for "newswire", "wl" stands for "weblog".

As with the other methods, this data selection method is applied in a bilingual manner: the gobal score used to evaluate sentence pairs is the unweighted sum of a score for two bi-SSCNNs, one of which reverses the polarity of the source language (Chinese or Arabic) and the target language (English).

3 Experiments on SMT

Our goal is to adapt the MT system when only a small amount of in-domain data is available. So in most of our experiments, we ignored domain information about the training data, such as the source of each corpus. What we have is a small development set (dev) and one or more test sets (test) which are in the same domain.

3.1 Data setting

We carried out experiments in two different data settings. The first setting is the Chinese-to-English "webforum" task from the BOLT project (zh2en); the test domain is a combination of news and web posts. The training data from LDC[2] are a mixture of newswire, web crawls, UN proceedings, etc. The second setting is the NIST 2012 Arabic-to-English task (ar2en). Again, the training data are available from LDC, and the test domain is a combination of newswire and weblog. We use the NIST 2008 test data as the dev set and the NIST 2012 test set as the test.

Table 1 summarizes the statistics of the training, dev, and test data for both tasks. Note that for both, the test domain includes newswire data, and the training data include a proportion of newswire data are extracted from comparable data, around 10% for the Chinese-to-English task and 20% for the Arabic-to-English task. We used all the comparable data available from LDC. Some of the comparable data are quite noisy, making the task of data selection more challenging both for two tasks.

Once a subset of the large in-domain, bilingual corpus has been selected by one of the methods described above, that subset is used as training data for a standard phrase-based SMT system.

3.2 Experimental setup

We employ the dev set as in-domain data. All the supervised CNN models (both the SSCNN ones and the Bi-SSCNN ones) are trained with the in-domain dev data as positive examples and an equal number of randomly selected general-domain sentences as negative examples. All the meta-parameters of the CNN are tuned on held-out data; we generate one-hot based bow-regions and seq-regions, word-embedding-based bow-regions and seq-regions and input them to the CNN. We set the region size to 5 and stride size to 1. The non-linear function we chose is "ReLU", the number of weight vectors or neurons is 500. We use the online available CNN

[2] https://catalog.ldc.upenn.edu/

toolkit $conText$[3]. To train the general domain word embedding, we used $word2vec$[4]. The size of the vector was set to 300.

The baseline SMT system for each language direction, "alldata" is trained using all general-domain data. All other systems are trained with a subset of the general-domain data of fixed size: 1.8 million sentence pairs (about 15% of the available training data) for the Chinese-to-English task, and 1.4 million sentence pairs (about 26% of the available training data) for the Arabic-to-English task. We experimented with the following five data selection methods. Note that the last three methods are bilingual - they measure not only the quality of the source and target sides of a sentence pair, but also the degree to which one is a good translation of the other - but the first two are not:

1. LM: Data selection by 3-gram LMs with Witten-Bell [5] smoothing, using bilingual cross entropy difference as the criterion. This is considered to be a state-of-the-art data selection method for domain adaptation (Axelrod et al., 2011). The "sum LM" variant uses the sum of the source and target LM scores for a sentence pair.

2. SSCNN: Data selection by semi-supervised CNN based on monolingual tokens (Section 2.3)

3. IBM-LM: Data selection by both IBM and language models (Section 2.1)

4. NNJM: Data selection by neural network joint models (Section 2.2)

5. Bi-SSCNN: Data selection by bitoken based semi-supervised CNN (Section 2.4)

3.3 Experimental results

We evaluated the system using the BLEU (Papineni et al., 2002) score on the test set. Following (Koehn, 2004), we apply the bootstrap resampling test to do significance testing. Table 2 summarizes the results for each task. The number of selected sentence pairs for each language pair (1.8 million pairs for Chinese-to-English, and 1.4 million pairs for Arabic-to-English) was decided on the basis of tests on held-out data using the IBM-LM method. That is, 1.8 million was the value of N that maximized the BLEU score of the final SMT system when IBM-LM was used to select N sentence pairs as training data for Chinese-to-English, and 1.4 had the same property for Arabic-to-English.

In the table, the bilingual methods are the ones below the horizontal line. All methods shown were applied in their symmetrical version, where the global score is obtained by adding a source-based score to a target-based score.

It can be seen from the table that the three NN-based data selection methods - the original word-based SSCNN, NNJM, and bitoken-based SSCNN - outperform the other methods. Among these three, Bi-SSCNN is significantly better than the other two, outperforming the original SSCNN by +0.5 BLEU on Chinese-to-English and +0.3 BLEU on Arabic-to-English.

What about the original motivation for devising data selection methods based on IBM-LM, NNJMs and Bi-SSCNN: that methods employing bilingual information will do a better job of screening out noisy sentence pairs, i.e., sentence pairs where each side is a bad translation of the other? The BLEU results above do not make this aspect of the behaviour of the various methods clear. For instance, the bilingual IBM-LM method obtains lower scores for both the Chinese-to-English and the Arabic-to-English task than the non-bilingual original SSCNN method.

[3] http://riejohnson.com/cnn_download.html

[4] https://code.google.com/archive/p/word2vec/

[5] For small amounts of data, Witten-Bell smoothing performed better than Kneser-Ney smoothing in our experiments

	symmetrical	bilingual	zh2en	ar2en
alldata	–	–	24.6	45.7
LM	yes	no	24.7	45.2
SSCNN	yes	no	25.1**	45.9
IBM-LM	yes	yes	24.9	45.4
NNJM	yes	yes	25.0*	45.8
Bi-SSCNN	yes	yes	25.6**++	46.2**+

Table 2: Summary of BLEU results. */** means result is significantly better than the "alldata" baseline at $p < 0.05$ or $p < 0.01$ level, respectively. +/++ means result is significantly better than the "SSCNN" method at $p < 0.05$ or $p < 0.01$ level, respectively.

We therefore decided to test the ability of the methods above to screen out noisy sentence pairs directly. Inspired by the experimental approach of (Goutte et al., 2012), we deliberately corrupted a randomly chosen 50% of the sentence pairs in the two large general-domain corpora for Chinese-to-English and Arabic-to-English by permuting the order of the target-language (English) sentences, while leaving the rest of each general-domain corpus (and the dev set on which data selection methods are trained) untouched. We can then see what percentage of the sentence pairs chosen by each data selection method have a mismatched source-language and target-language side. Because the NNJM-based and bitoken SSCNN-based approaches use word alignment information, we re-ran word alignment on all the general-domain data (using the IBM and HMM models trained on the original version of the data).

Table 3 shows the proportion of mismatched sentence pairs in the top N selected sentence pairs. For each language direction, two different values of N are tried: $N = 1.8M$ and $N = 200K$ for Chinese-to-English, and $N = 1.4M$ and $N = 200K$ for Arabic-to-English. Data selection based on only a source LM ("src LM") or target LM ("tgt LM") is completely ineffective at screening out mismatched sentence pairs: their proportion in the selected subset is around 50%, just as it was in the large corpus the subset was selected from. Symmetrizing LM data selection by adding the source LM and target LM yields an improvement that is particularly noticeable when only 200K sentence pairs are selected for each language direction: 38% of the highest-scoring 200K Chinese-English pairs and 39% of the highest-scoring 200K Arabic-English pairs are mismatched. The variants of the original SSCNN method show a similar pattern, with the symmetrical version performing better than the versions that rely only on the source-side or target-side scores.

A definite improvement in performance is seen when we consider the three bilingual data selection methods: IBM-LM, NNJM, and Bi-SSCNN. By far the best performer of the three is Bi-SSCNN. When it ranks sentence pairs in the Chinese-English corpus, of the 1.8M pairs with the highest scores, about 29% are mismatched; of the 200K pairs with the highest scores, about 11% are mismatched. The mismatch proportions for the highest-scoring Arabic-English pairs are about 28% for the 1.4M subset and 10% for the 200K subset.

These results show that the three bilingual methods - IBM-LM, NNJM, and Bi-SSCNN - are more effective at screening out noisy sentence pairs than the non-bilingual methods. It may seem surprising that all three of these methods do not therefore also have the highest BLEU scores in Table 2. A possible reason is that the general-domain data we used to train the system are not too noisy. Moreover, (Goutte et al., 2012) showed that phrase-based MT is highly robust to the sentence alignment errors: "performance is hardly affected when the misalignment rate is below 30%, and introducing 50% alignment error brings performance down less than 1 BLEU point." Thus, the proportion of misaligned sentence pairs in a training corpus and the BLEU

	symmetrical	bilingual	zh2en	ar2en	zh2en	ar2en
#selected			1.8M	1.4M	200K	200K
src LM	no	no	0.501	0.502	0.497	0.489
tgt LM	no	no	0.496	0.505	0.495	0.492
sum LM	yes	no	0.461	0.454	0.376	0.389
src SSCNN	no	no	0.487	0.495	0.481	0.481
tgt SSCNN	no	no	0.488	0.498	0.476	0.488
sum SSCNN	yes	no	0.453	0.455	0.365	0.401
IBM-LM	yes	yes	0.411	0.417	0.355	0.312
NNJM	yes	yes	0.428	0.402	0.376	0.298
Bi-SSCNN	yes	yes	0.292	0.280	0.113	0.100

Table 3: The proportion of mismatched sentence pairs in the top N sentence pairs selected from the 50% sentence pairs permuted corpora.

language	zh2en	ar2en
alldata	24.6	45.7
Bi-SSCNN	25.6	46.2
clean-in	25.4	46.0

Table 4: BLEU Results for Manually Selected Data.

score obtained from a system on that corpus are not highly correlated.

For most practical purposes, we care more about a data selection method ability to produce a training corpus that will yield an SMT system with a high BLEU score than its ability to screen out noisy sentence pairs. Nevertheless, it is reassuring that the method which yields the highest BLEU score in our experiments, Bi-SSCNN, is also the one that is far better than the other methods at screening out noisy pairs.

In another experiment, we manually selected subsets made up of data that were likely to be in-domain (because they came from newswire and weblogs, just like the test set) and clean (we excluded comparable data). We selected around 1.4M sentence pairs for Chinese-to-English task and 1.0M for Arabic-to-English task.

As Table 4 shows, the resulting "clean-in" training data set yielded an SMT system that performed about as well, or slightly worse, than an SMT system trained on data selected by "Bi-SSCNN". This is a strong argument in favour of using Bi-SSCNN, which is fully automatic and doesn't require any outside knowledge about the sentence pairs in the large general-domain corpus.

3.4 Discussion

When we examined a random selection of sentence pairs in the big general-domain corpora and looked at their scores as assigned by the Bi-SSCNN method, we made an interesting observation. The sentence pairs with lowest scores are out-of-domain, as expected, but also tend to have very good translation quality. The overall ordering (from highest to lowest score) tends to be 1. clean and in-domain pairs 2. noisy and in-domain pairs 3. noisy and out-of-domain pairs 4. clean and out-of-domain pairs.

This tendency, which is surprising at first glance, is understandable. Both the positive samples in the dev set and the sampled negative examples used to train the Bi-SSCNN classification model are of good quality. Thus, the Bi-SSCNN has learned two top priorities: selecting **for** clean, in-domain sentence pairs, and selecting **against** clean, out-of-domain sentence pairs. It thus scores the former type of sentence pair highest, and the latter type lowest. Noisy sentence pairs thus receive intermediate scores. This behaviour may be desirable from a practical point of view: clean, out-of-domain sentence pairs are dangerous in the sense that they will populate the phrase table with phrase translations that are likely to compete with the correct translations for the given domain. Noisy out-of-domain sentence pairs will have a more random effect, sprinkling the phrase table with low-frequency, unlikely translations, thus doing less harm (and noisy in-domain pairs will probably have a mildly positive effect on the performance of the resulting SMT system).

In our experiments, NNJM-based data selection did not stand out from other methods either in terms of its impact on BLEU or in terms of its ability to screen out noisy sentence pairs. Study of the NNJMs we trained suggests that part of the problem is their limited vocabulary size: 32K words for both the source and the target language. All other words are mapped into a small number of clusters. Unfortunately, many sentence pairs of poor quality end up with several occurrences of particular cluster on both the source and the target side, which may mislead the NNJMs into thinking that these sentence pairs are of good quality.

Note also that the SMT systems trained on the selected data did not contain an NNJM feature (or, indeed, any neural component). It is possible that the NNJM-based method will work better when applied to the task of selecting data from a bilingual corpus to train a second, larger NNJM. We intend to explore this possibility in our future work.

4 Follow-up experiments on NMT

After we submitted this paper, we did some experiments with a neural machine translation (NMT) system (Sutskever et al., 2014; Bahdanau et al., 2015) using Bi-SSCNN. [6] The experiments were carried out with an open source system called Nematus (Sennrich et al., 2016), which is an attention-based NMT system (Bahdanau et al., 2015). We carried out experiments on English-to-French (en2fr) WMT task [7]. The training data contain 12 million sentence pairs; the dev set is a concatenation of newstest2012 and 2013, which contains 6,003 sentence pairs; the test set is newstest2014, which contains 3,003 sentence pairs.

Table 4 summarizes our experiments. If we introduce 30% sentence alignment error to the original data, we lost over 3 BLEU score (35.4 vs 32.0 on the 12.0M data set; 33.7 vs 30.6 on the 6.4M sampled data set.) Then, if we apply Bi-SSCNN to the 12.0M data which contains 30% sentence alignment error, by carefully manipulating the negative training samples of the Bi-SSCNN, we can turn on the noise reduction and domain adaptation separately. Experiment 5 shows that when the data contain 30% alignment error, Bi-SSCNN can screen out the noise in the data and improve the performance to 33.9 BLEU, from 30.6 BLEU for the experiment 4 baseline. Experiment 6 showed that if we only use Bi-SSCNN for domain adaptation on a data with 30% sentence alignment error, the improvement is smaller, only 0.4 BLEU. Finally, if we turn on both noise reduction and domain adaptation for Bi-SSCNN, we obtain the best result, which is 34.2: a 3.6 BLEU point improvement over the baseline in experiment 4. This series of experiments shows that 1. neural machine translation is more sensitive to noise than SMT; 2. Bi-SSCNN is effective in carrying out both noise reduction and domain adaptation.

5 Conclusions

We proposed a new method for data selection from a large bilingual corpus for the purpose of training an SMT system. The new method is based on a bitoken semi-supervised convolutional

[6] Due to the time limit, we did not finish the experiments with other data selection methods.

[7] The data is available at http://www-lium.univ-lemans.fr/ schwenk/nnmt-shared-task/

id	data size	data description	BLEU
1	12.0M	all original data	35.4
2	12.0M	30% alignment error	32.0
3	6.4M	sampled from data 1: original data	33.7
4	6.4M	sampled from data 2: with 30% alignment error	30.6
5	6.4M	Bi-SSCNN noise reduction on data 2	33.9
6	6.4M	Bi-SSCNN domain adaptation on data 2	31.0
7	6.4M	Bi-SSCNN noise reduction and domain adaptation on data 2	34.2

Table 5: BLEU Results for English-to-French WMT task with neural machine translation system.

neural networks. It outperformed its nearest competitor, a method that uses a word-based SS-CNN, by +0.5 BLEU on a Chinese-to-English task and by +0.3 BLEU on an Arabic-to-English task. Since one of the motivations underlying the creation of the Bi-SSCNN method (and two other data selection methods, those based on IBM-LM and NNJM) was the ability to screen out noisy sentence pairs, we carried out another type of experiment in which the methods were explicitly tested for the ability to do this when half the pairs in the bilingual corpus have been deliberately corrupted. According to this criterion, too, the Bi-SSCNN outperformed all other methods.

In the follow-up experiments, we find that neural machine translation is more sensitive to noisy data than statistical machine translation. Therefore, Bi-SSCNN, which can effectively screen out noisy sentence pairs, can benefit NMT much more than SMT. For instance, given a potential training corpus with 30% sentence alignment error, data selected with Bi-SSCNN yields a system with a performance gain of over 3 BLEU points above the baseline.

References

Axelrod, A., He, X., and Gao, J. (2011). Domain adaptation via pseudo in-domain data selection. In *EMNLP 2011*.

Axelrod, A., Resnik, P., He, X., and Ostendorf, M. (2015). Data selection with fewer words. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 58–65, Lisbon, Portugal.

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *ICLR*.

Chen, B. and Huang, F. (2016). Semi-supervised convolutional networks for translation adaptation with tiny amount of in-domain data. In *Proceedings of the SIGNLL Conference on Computational Natural Language Learning*, Berlin, Germany.

Denkowski, M., Hanneman, G., and Lavie, A. (2012). The cmu-avenue french-english translation system. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 261–266, Montréal, Canada. Association for Computational Linguistics.

Devlin, J., Zbib, R., Huang, Z., Lamar, T., Schwartz, R., and Makhoul, J. (2014). Fast and robust neural network joint models for statistical machine translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1370–1380, Baltimore, Maryland. Association for Computational Linguistics.

Duh, K., Neubig, G., Sudoh, K., and Tsukada, H. (2013). Adaptation data selection using neural language models: Experiments in machine translation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 678–683, Sofia, Bulgaria.

Durrani, N., Sajjad, H., Joty, S., Abdelali, A., and Vogel, S. (2015). Using joint models for domain adaptation in statistical machine translation. In *Proceedings of the Fifteenth Machine Translation Summit (MT Summit XV)*.

Goutte, C., Carpuat, M., and Foster, G. (2012). The impact of sentence alignment errors on phrase-based machine translation performance. In *Tenth Biennial Conference of the Association for Machine Translation in the Americas (AMTA-2012)*, San Diego, USA.

Jiang, J., Way, A., and Carson-Berndsen, J. (2010). Lattice score based data cleaning for phrase-based statistical machine translation. In *14th Annual Conference of the European Association for Machine Translation*, Saint-Raphael, France.

Johnson, R. and Zhang, T. (2015a). Effective use of word order for text categorization with convolutional neural networks. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 103–112, Denver, Colorado.

Johnson, R. and Zhang, T. (2015b). Semi-supervised convolutional neural networks for text categorization via region embedding. In Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M., and Garnett, R., editors, *Advances in Neural Information Processing Systems 28*, pages 919–927. Curran Associates, Inc.

Kalchbrenner, N., Grefenstette, E., and Blunsom, P. (2014). A convolutional neural network for modelling sentences. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 655–665, Baltimore, Maryland.

Khadivi, S. and Ney, H. (2005). Automatic filtering of bilingual corpora for statistical machine translation. In *Natural Language Processing and Information Systems, 10th International Conference on Applications of Natural Language to Information Systems*, pages 263–274, Alicante, Spain.

Kim, Y. (2014). Convolutional neural networks for sentence classification. *CoRR*, abs/1408.5882.

Koehn, P. (2004). Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Barcelona, Spain.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Pereira, F., Burges, C. J. C., Bottou, L., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc.

LeCun, Y. and Bengio, Y. (1998). Convolutional networks for images, speech, and time series. In Arbib, M. A., editor, *The Handbook of Brain Theory and Neural Networks*, pages 255–258. MIT Press, Cambridge, MA, USA.

Lü, Y., Huang, J., and Liu, Q. (2007). Improving Statistical Machine Translation Performance by Training Data Selection and Optimization. In *Proceedings of the 2007 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Prague, Czech Republic.

Mansour, S., Wuebker, J., and Ney, H. (2011). Combining translation and language model scoring for domain-specific data filtering. In *Proceedings of IWSLT*.

Marino, J. B., Banchs, R. E., Crego, J. M., de Gispert, A., Lambert, P., Fonollosa, J. A. R., and Costa-jussà, M. R. (2006). N-gram-based machine translation. *Computational Linguistics*, 32(4):527–549.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Mikolov, T., Karafiát, M., Burget, L., Černocký, J., and Khudanpur, S. (2010). Recurrent neural network based language model. In *Proceedings of the 11th Annual Conference of the International Speech Communication Association (INTERSPEECH2010)*, pages 1045–1048. International Speech Communication Association.

Moore, R. C. and Lewis, W. (2010). Intelligent selection of language model training data. In *ACL 2010*.

Munteanu, D. S. and Marcu, D. (2005). Improving machine translation performance by exploiting non-parallel corpora. *Computational Linguistics*, 31(4):477–504.

Niehues, J., Herrmann, T., Vogel, S., and Waibel, A. (2011). Wider context by using bilingual language models in machine translation. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 198–206, Edinburgh, Scotland. Association for Computational Linguistics.

Okita, T., Naskar, S., and Way, A. (2009). Noise reduction experiments in machine translation. In *the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, Bled, Slovenia.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: A method for automatic evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 311–318, Philadelphia. ACL.

Sennrich, R., Haddow, B., and Birch, A. (2016). Edinburgh neural machine translation systems for wmt 16. In *Proceedings of the First Conference on Machine Translation*, pages 371–376, Berlin, Germany. Association for Computational Linguistics.

Stewart, D., Kuhn, R., Joanis, E., and Foster, G. (2014). Coarse 'split and lump' bilingual language models for richer source information in smt. In *Eleventh Biennial Conference of the Association for Machine Translation in the Americas (AMTA-2014)*, Vancouver, Canada.

Sutskever, I., Vinyals, O., and Le, Q. V. V. (2014). Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27 (NIPS)*, pages 3104–3112.

Wang, P., Xu, J., Xu, B., Liu, C., Zhang, H., Wang, F., and Hao, H. (2015). Semantic clustering and convolutional neural network for short text categorization. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 352–357, Beijing, China.

Yasuda, K., Zhang, R., Yamamoto, H., and Sumita, E. (2008). Method of selecting training data to build a compact and ef?cient translation model. In *International Joint Conference on Natural Language Processing.*

Zhao, B., Eck, M., and Vogel, S. (2004). Language model adaptation for statistical machine translation with structured query models. In *Proceedings of the International Conference on Computational Linguistics (COLING) 2004*, Geneva.