Data Selection with Feature Decay Algorithms Using an Approximated Target Side

Alberto Poncelas, Gideon Maillette de Buy Wenniger, Andy Way

ADAPT Centre, School of Computing,
Dublin City University, Dublin, Ireland
{firstname.lastname}@adaptcentre.ie

Abstract

Data selection techniques applied to neural machine translation (NMT) aim to increase the performance of a model by retrieving a subset of sentences for use as training data.

One of the possible data selection techniques are transductive learning methods, which select the data based on the test set, i.e. the document to be translated. A limitation of these methods to date is that using the source-side test set does not by itself guarantee that sentences are selected with correct translations, or translations that are suitable given the test-set domain. Some corpora, such as subtitle corpora, may contain parallel sentences with inaccurate translations caused by localization or length restrictions.

In order to try to fix this problem, in this paper we propose to use an approximated target-side in addition to the source-side when selecting suitable sentence-pairs for training a model. This approximated target-side is built by pre-translating the source-side.

In this work, we explore the performance of this general idea for one specific data selection approach called Feature Decay Algorithms (FDA).

We train German-English NMT models on data selected by using the test set (source), the approximated target side, and a mixture of both. Our findings reveal that models built using a combination of outputs of FDA (using the test set and an approximated target side) perform better than those solely using the test set. We obtain a statistically significant improvement of more than 1.5 BLEU points over a model trained with all data, and more than 0.5 BLEU points over a strong FDA baseline that uses source-side information only.

1. Introduction

Supervised machine learning aims to learn predictive models from a set of labeled examples (training data) so that it can accurately predict the labels of new, unlabeled, examples. Having more data may seem at first glance to be beneficial to building more accurate models, but upon closer inspection this is not necessarily always the case. Machine learning models by design have an inductive bias that forces them to generalize over the training examples rather than just memorizing them without generalization. This means, however, that if the size of the training set is increased, this may lead to optimizing the model for predicting the labels of more examples, but which on average are less relevant at test time than would be the case for a more focused, smaller training set. The intuition of the importance of using a highly relevant set of training examples is captured well by the K-nearest neighbour model, which essentially computes at test time on-the-fly a very localized density estimate for every test example, based on the K training examples closest to the test example. It then uses this density estimate for classification. For the K-nearest neighbour model, increasing K too much is at the expense of basing predictions on an increasing number of less relevant examples. Furthermore similar to the K-nearest neighbour model, other predictive models which typically discard the original training examples and keep only a learned generalization over these examples, can suffer if the training data becomes bigger but on average less relevant to the test set.

In Machine Translation (MT), the data used to build the models are parallel sentences (pairs of sentences in two languages, which are translations of each other) and we encounter the same problem when the amounts of data become excessively large. Too much training data may cause the model to be too generic, and not perform well if $test_{src}$ (the document to be translated, i.e. the test set), belongs to a specific domain.

Data selection techniques aim to solve that problem by selecting a subset of training data. Models that are trained on a small set of parallel sentences can perform better than those trained on all training data [1, 2].

Within the data selection field we can find several approaches to reduce the data: select sentences of good translation quality (*data quality*), select sentences relevant for a particular domain (*domain adaptation*), or select sentences that are relevant for $test_{src}$ (*transductive learning*). We focus on this last type, and so in this paper we propose new methods to build Neural Machine Translation (NMT) models that are tailored towards a $test_{src}$.

Transductive learning [3] aims to find the best training instances given an unlabeled example. In MT this means finding the best parallel sentences given a document $test_{src}$ to be translated. In our work, the transductive data-selection method that we explore is Feature Decay Algorithms (FDA) [4, 5, 6]. Standard FDA uses the *n*-grams of $test_{src}$ to retrieve training sentence pairs with source-side most similar to $test_{src}$. FDA has demonstrated good performance in Sta-

tistical Machine Translation (SMT) and NMT [2].

In most cases, FDA is used as a single step in the pipeline of building a model, using $test_{src}$ to extract a subset of parallel sentences. In this paper, we propose a different configuration of use of FDA for building NMT models (see left side of Figure 1). In particular, we propose executing FDA not only using the $test_{src}$ (source-side language), as is common, but additionally on a pre-translated test set (approximated target-side). In order to avoid confusion, in this work we use $test_{src}$ to indicate the test set (in the source-side language) and $test_{trg}$ to indicate the pre-translation of the test set (in the target-side language). The outputs of these two executions can be combined into one training set to build a model that produces better translations than models built using FDA having only $test_{src}$ as input.

Considering both the source side and target side of the parallel sentences as selection criteria is especially useful when using a corpus that includes sentences from subtitles in different languages. There are particular problems concerning parallel sentences comprising subtitles. For example, both sentences in the source and target side are limited to be displayed in the same time window (assuming they are synchronized). As the length of the same sentences in different languages can be different, this may causes the longest one to be rephrased, split in two, or have words omitted so it meets the time requirement.

In our work we use an approximated, synthetic target-side using a technique we call pre-translation. One way to look at this is as a form of synthetic-data generation. As such it is somewhat reminiscent of synthetic source-data generation using a target-to-source translation model, a technique known as back-translation introduced by Sennrich et al.(2016) [7].

2. Related Work

Data selection techniques aim to select a subset of data such that the models trained on that subset perform better. There are multiple approaches to achieve those improvements, such as domain adaptation or noise reduction approaches [8].

Methods based on domain adaptation include the work of Moore and Lewis (2010) [9], who propose to use language models (LM) to select data. An LM is a distribution over sequences of words in a monolingual text, and is often used by SMT systems to model the fluency of the outputs. Given a string s and a language model LM_d, $H_d(s)$ is the entropy of the distribution of s according to LM_d.

Moore and Lewis build an in-domain language model LM_I and an out-of-domain language model LM_O, and determine how likely each sentence s is to be in-domain by computing the entropy difference $[H_I(s) - H_O(s)]$. Axelrod et al. (2011) [10] extend the method by using LMs in both the source-side and target-side languages, defining the bilingual cross-entropy difference.

Another method, proposed by van der Wees et al. (2017) [1], is to gradually remove out-of-domain sentences each η epochs when training the NMT model.

In our work, we select data that is similar to $test_{src}$ (and so, more relevant for use as training data). Previous research on selecting data considering the test set includes the work of Li et. al. (2018) [11] where they fine-tune a pre-built NMT model using training data selected based on $test_{src}$. They use similarity measures, such as Levenshtein distance [12] or the cosine similarity of the average of the word embeddings, [13].

The method that we use to select data is FDA [4, 5, 6], which has already proven to be useful in SMT [14, 15, 16] and NMT [2]. Selecting a small subset of sentences from a parallel corpus using FDA is enough to train SMT systems that perform better than systems trained using the whole parallel corpus.

FDA takes as input a set of parallel sentences U and a seed (generally the $test_{src}$). Given U and the seed, FDA retrieves an ordered sequence of sentences L from U. Sentences are ordered based on the amount of *n*-grams they share with the seed, with more shared *n*-grams meaning higher preference, while also considering the variability of the *n*-grams in the selected sentences.

The algorithm initializes L as a void sequence and iteratively selects one sentence $s \in U - L$ and appends it to L. The sentence s to select at each step is the one most relevant to $test_{src}$, based on the number of *n*-grams that s shares with the $test_{src}$. The score of the relevance is computed as in (1):

$$score(s) = \frac{\sum_{f \in F_s} 0.5^{C_L(f)}}{\text{\# words in s}} \quad (1)$$

where F_s is the set of *n*-grams present in s and $test_{src}$ (by default the order of the *n*-grams ranges from 1 to 3). $C_L(f)$ is the count of the *n*-gram f in the sequence L of selected sentences. Including $C_L(f)$ in the computation of the score causes the algorithm to penalize *n*-grams that have been selected several times, and hence favouring the selection of sentences that contain new *n*-grams.

3. Using an Approximated Test Target-side

FDA uses $test_{src}$ as seed to retrieve a subset from a set of parallel sentences. In order to retrieve the sentences it scores the *n*-grams of $test_{src}$ (source-side language). We show the pipeline of usage of FDA on the left side of Figure 1. Here, the files $test_{src}$ and *parallel text* are used as input, and FDA retrieves a subset of the sentences to be used for building a model that is adapted to $test_{src}$.

We propose to use both the test source-side $test_{src}$ and the approximated test target-side $test_{trg}$ as features in FDA, when selecting the set of sentences from the parallel text.

We show the pipeline of our approach on the right side of Figure 1. First, $test_{src}$ is translated (*translate* step). Then, using FDA, we select a subset of parallel sentences given: (a) $test_{src}$ as seed (FDA_{src}), and (b) $test_{trg}$ as seed (FDA_{trg}). These two sets can be combined into one set which serves as training data to build an MT model.

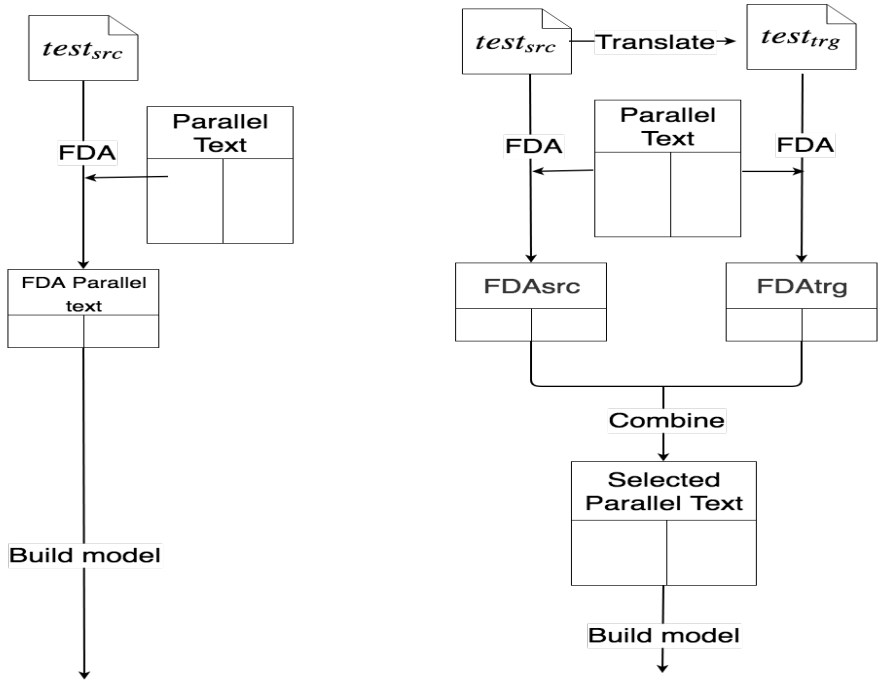

Figure 1: Pipeline of the traditional usage of FDA (left) and pipeline of our proposal, using the target-side (right).

In the following subsections we explain in more detail two issues that are yet unanswered in the pipeline : (1) how to build $test_{trg}$ (addressed in Section 3.1), and (2) how to combine the outputs of FDA (addressed in Section 3.2).

3.1. Pre-Translation of $test_{src}$

The first step in our approach consists of building $test_{trg}$ (*translate* step on the right side of Figure 1) so it can be used as the seed to extract parallel sentences using the target side. In order to perform this pre-translation we need to build a model, which we refer to as the *initial model*.

There are several approaches to build the *initial model*, such as using SMT or NMT. These models can be trained using the full training data or subsets (such as randomly sampled, selected according to a particular domain, etc.). In this work we use an NMT model built with the full training data.

3.2. Combining FDA outputs

In order to combine the sentences of FDA_{src} and FDA_{trg} into one training set of N sentences, various strategies are possible such as retrieving the intersection or the union of sentences. In this work we explore the strategy of concatenating both outputs (allowing the repetition of sentences) and propose as future work alternative methods for merging both parallel datasets.

The outputs of FDA_{src} and FDA_{trg} can be seen as an ordered sequence of sentences as in equation (2) and equation (3):

$$FDA_{src} = (s_1^{(src)}, s_2^{(src)}, s_3^{(src)}, ...s_N^{(src)}) \quad (2)$$

$$FDA_{trg} = (s_1^{(trg)}, s_2^{(trg)}, s_3^{(trg)}, ...s_N^{(trg)}) \quad (3)$$

In order to obtain a training set that combines the outputs of FDA_{src} and FDA_{trg}, we concatenate the top sentences of each subset to obtain a new list of sentences of size N, as in equation (4)

$$FDA = (s_1^{(src)}, ...s_{N*\alpha}^{(src)}, s_1^{(trg)}, ...s_{N*(1-\alpha)}^{(trg)}) \quad (4)$$

where $0 \leq \alpha \leq 1$ indicates the proportion of sentences that are selected from FDA_{src} and FDA_{trg}.

Note that some of the sentences may be replicated; it may happen that $s_i^{(src)} = s_j^{(trg)}$, i.e. those that have been retrieved by both executions FDA. In this work we decided to keep the duplicates as it may be beneficial to oversample those sentences in which there is an agreement of both executions of FDA. However, we propose as future work to investigate the effect of removing those duplicate sentences.

The core of our approach is combining the outputs of the two executions of FDA (using the test and translated sets). Given the concatenation method presented in this section, the outputs can be classified as one of the three options:

- Source-side only: use only the output of FDA_{src} for building the model. It is the configuration where $\alpha = 1$ in Equation (4), which is equivalent to the traditional procedure of using FDA (left side of Figure 1, so we use this approach as the baseline.
- Target-side only: use the output of FDA_{trg} for building the model, which is the configuration where $\alpha = 0$ in Equation (4).

- Source-and-target-side: combine FDA_{src} and FDA_{trg}. This is the configuration where different values of α in equation (4) are set. In our work we explore the values $\alpha = 0.25$, $\alpha = 0.50$ and $\alpha = 0.75$.

4. Experiments

4.1. Experimental Settings

We experiment with models for German-to-English direction. The parallel data used for the experiments is the training data provided in the WMT 2015 [17] (4.5M sentence pairs, 225M words). The dev set of the NMT models (both the initial model and those trained using the selected datasets) are 5K randomly sampled sentences from development sets from previous years. All the models presented here are evaluated using the same test set which comprises documents provided in WMT 2015 translation task as $test_{src}$.

In order to build the NMT models we use OpenNMT-py, which is the Pytorch port of OpenNMT [18]. All the NMT models we build use the same settings (we only change the training data used to build them). The value parameters are the default ones of OpenNMT-py (i.e. 2-layer LSTM with 500 hidden units, vocabulary size of 50000 words for each language). All the models are executed for 13 Epochs.

In the experiments we build models with the data selected by using FDA_{src} and FDA_{trg}. We explore selecting different sizes of selected data: 500K, 1M and 2M sentence pairs.

5. Results

	baseline
BLEU	0.2474
TER	0.5525
METEOR	0.2798
CHRF3	48.9473

Table 1: Results of the model trained with all available training data; also the no-FDA baseline.

First, we show in Table 1 the quality of the pre-translated $test_{trg}$. This has been produced by the *initial model*, an NMT model trained with all training data. This result also serves as a no-FDA baseline to asses the benefit of using FDA in general with.

The evaluation metrics presented in Table 1 give an estimation of the similarity between the model output and a human-translated reference. The evaluation metrics we use are: BLEU [19], TER [20], METEOR [21] and CHRF [22].

The results of the models are shown in Table 2. The columns show the different configurations used to build the set of selected sentences (i.e. the value of α in equation (4) used). This means that the column $\alpha = 0.75$ shows the results of the model trained with the sentences from the top-750K sentences of FDA_{src} and the top-250K sentences of FDA_{trg}.

First, one may wonder whether FDA data selection is at all helpful? Comparing the scores in Table 2 to the baseline system trained on all data in Table 1, we see that all FDA systems outperform it, with the best one obtaining more than 1.5 BLEU points improvement (a relative improvement of 6%).

We have marked in bold the scores that outperform the second baseline: FDA applied using $test_{src}$ only (i.e the configuration using FDA_{src} and $\alpha = 1$), as proposed in [2], and computed the statistical significance (marked with an asterisk) with multeval [23] for BLEU, TER and METEOR when compared to the baseline at level p=0.01 using bootstrap resampling [24].

5.1. Ratio of data obtained using source and target side

Intuitively, models built using the data selected based on $test_{trg}$ might perform worse than using $test_{src}$ only. $test_{trg}$ may contain errors produced by the machine-generated text, so an algorithm that bases the decision on that text may not select the best sentences. Indeed, this can be seen in the column $\alpha = 0$ of Table 2, where most of the scores are worse than those in column $\alpha = 1$.

On the other hand, using only $test_{src}$ as a selection criterion also has its limitations. While it guarantees the selected source sentences to be similar to $test_{src}$, it does not provide any information about the target side of the selected sentences. Therefore, it may still select sentences with target-side translations that are wrong or not suitable given the domain of the test-set, thereby hurting the final translation accuracy.

Using training data containing parallel sentences that are not an accurate translation of each other is a problem that can be encountered when using parallel sentences obtained from subtitles. Often, translation of subtitles needs to be adapted to meet length requirements (due to the restriction of time it is displayed on screen). We present some examples of sentences that are not accurately translated in Table 3.

We find that selecting sentences based both on $test_{src}$ and on $test_{trg}$ works better than using one selection criterion alone. Thus, using an approximated target side, even if imperfect, can help. The best performance is obtained using configurations that combine outputs of FDA_{src} and FDA_{trg} ($\alpha = 0.75$, $\alpha = 0.50$ and $\alpha = 0.25$ columns).

The best results are obtained for $\alpha = 0.75$ using 1 million sentences for selection. This setting improves 1.53 BLEU points over the no-FDA baseline (model trained with all data) and 0.67 BLEU points over the baseline that uses only the source side for selection in FDA.

In Table 3 we show examples of sentences that are exclusive outputs of FDA_{src} or FDA_{trg}. These examples give an indication about how including the output of FDA_{trg} can benefit (or hurt) the quality of the selected data.

In the first row we see that the sentence "nun gibt es kein Zurück mehr ." has been selected by FDA_{src} as it matches

		$\alpha = 1$	$\alpha = 0.75$	$\alpha = 0.50$	$\alpha = 0.25$	$\alpha = 0$
500K lines	BLEU	0.2517	**0.2542**	**0.2543**	**0.2534**	0.2441
	TER	0.5601	**0.5521***	**0.5563**	**0.5544**	0.5628
	METEOR	0.2886	**0.2895**	0.2882	**0.2888**	0.2789
	CHRF3	49.8314	**50.0915**	**49.8898**	**49.9074**	48.7796
1M lines	BLEU	0.256	**0.2627***	**0.2595**	**0.2600***	0.2496
	TER	0.5497	**0.5455***	**0.5462**	**0.5493***	0.5534
	METEOR	0.2886	**0.2920***	**0.2921***	**0.2918***	0.2833
	CHRF3	50.0932	**50.6273**	**50.5226**	**50.5682**	49.5192
2M lines	BLEU	0.2585	**0.2610**	0.2580	**0.2614**	0.2547
	TER	0.5454	**0.5429**	0.5465	**0.5437**	0.5496
	METEOR	0.2894	**0.2923***	**0.2903**	**0.2927***	0.2852
	CHRF3	50.095	**50.5582**	**50.2431**	**50.5487**	49.7838

Table 2: Results of the models using different sizes of FDA_{src} and FDA_{trg}.

German	English	pos FDA_{src}	pos FDA_{trg}
nun gibt es kein Zurück mehr .	there is no going back now .	12	-
diese Zahl ist mehr als doppelt so viel , als vor 10 Jahren .	famous pieces from the 19th century include those by Delacroix , Gauguin , Monet , Renoir and Corot .	50	-
diese Aufzählung ließe sich beliebig fortführen .	and I could continue .	-	63
bitte beachten Sie , dass Sie sich registrieren lassen müssen , um einen Zugang zu den detaillierten Außenhandelsdaten zu erhalten .	all data can be downloaded free of charge .	-	92

Table 3: Examples of sentences retrieved by FDA_{src} and FDA_{trg}

"kein Zurück mehr" in the input. According to this sentence, this *n*-gram should be translated as "no going back". The translation found for "kein Zurück mehr" in $test_{trg}$ is "point where there is no return" (which, in addition, is closer to the reference "point of no return") and hence FDA_{trg} will use *n*-grams such as "point" or "no return" to retrieve sentences.

In the second row, we find an example of a sentence retrieved by FDA_{src} whose translation is not accurate (this is easily noticeable as the names "Delacroix, Gauguin, Monet, Renoir and Corot" are not present in the English-side sentence). Including this sentence in the training data causes the quality to decrease and the models to perform worse. This problem is not exclusive of FDA_{src}, as in rows 3 and 4 we see the same problem happening in the output of FDA_{trg}.

Combining the outputs of FDA_{src} and FDA_{trg} causes the training data to be reinforced with sentences with relevant translations. Note that mixing the outputs of the two executions of FDA cause some sentence pairs to be replicated, as there is an overlap of the outputs.

In Table 4 we indicate the amount of unique lines contained in the training data of the models (those presented in Table 2). In the table we observe that the number of unique lines is high in all training sets. The proportion of unique lines ranges from 82% to 94%, which shows how FDA_{src} and FDA_{trg} retrieve different sentences mostly.

	$\alpha = 0.75$	$\alpha = 0.50$	$\alpha = 0.25$
500K	471753 (94%)	460993 (92%)	471174 (94%)
1M	918506 (92%)	886685 (89%)	917087 (92%)
2M	1749015(87%)	1648727(82%)	1745142(87%)

Table 4: Number of unique sentences in the training data.

When performing a column-wise comparison in Table 4, we see how the number of unique lines is larger when the output of one of the FDA models dominates the training data ($\alpha = 0.25$ or $\alpha = 0.75$ columns) compared to those sets that contain the same amount of sentences extracted from FDA_{src} and FDA_{trg} (column $\alpha = 0.50$).

We also see that the larger the amount of selected data, the more overlap exists between the two outputs (the proportional amount of unique lines is smaller). For example, in column $\alpha = 0.50$, when 500K lines are selected, there are 92% non-repeated lines, and this decreases to 82% when selecting 2M lines. The same can be observed in the other columns. This indicates how the selected data using FDA_{src} and FDA_{trg} tend to be more similar the more sentences are retrieved.

6. Conclusion and Future Work

In this work, we explored a different pipeline in which FDA can be used. We discovered that using $test_{trg}$ (which is machine-generated) as the seed of FDA can improve the performance.

In our experiments, we built models using training sets containing replicated instances of sentence pairs (as the output of the two runs of FDA, on the source-side and target-side, may overlap). This opens the door to exploring data selection algorithms allowing the repetition of selected instances.

In the future, we want to consider other procedures for combining the outputs of FDA, as we believe that other merging strategies may achieve better results. For example, considering both *n*-grams on the source and target side in combination (rather than two separate executions of FDA) may achieve better performance.

In addition, we want to explore the performance when using a different *initial model*. Changing the initial model to produce the $test_{trg}$ causes FDA_{trg} to have a different performance. We believe that using another dataset to build the initial NMT model (or even using different paradigms such as SMT or rule-based MT) or choosing an initial model that is also closer to $test_{src}$ (e.g. using FDA to build the initial model) should boost the performance. Moreover, the use of several initial models allow us to perform concatenations of several outputs of FDA using different seeds.

Finally, we want to explore how data selection algorithms may improve when allowing the algorithm to select the same sentence pairs several times.

7. Acknowledgements

This research has been supported by the ADAPT Centre for Digital Content Technology which is funded under the SFI Research Centres Programme (Grant 13/RC/2106) and is co-funded under the European Regional Development Fund.

This work has also received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 713567.

8. References

[1] M. van der Wees, A. Bisazza, and C. Monz, "Dynamic data selection for neural machine translation," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Copenhagen, Denmark, 2017, pp. 1400–1410.

[2] A. Poncelas, G. M. de Buy Wenniger, and A. Way, "Feature decay algorithms for neural machine translation," in *Proceedings of the 21st Annual Conference of the European Association for Machine Translation*, Alacant, Spain, 2018, pp. 239–248.

[3] V. N. Vapnik, *Statistical Learning Theory*. Wiley-Interscience, 1998.

[4] E. Biçici and D. Yuret, "Instance selection for machine translation using feature decay algorithms," in

Proceedings of the Sixth Workshop on Statistical Machine Translation, Edinburgh, Scotland, 2011, pp. 272–283.

[5] E. Biçici, Q. Liu, and A. Way, "ParFDA for fast deployment of accurate statistical machine translation systems, benchmarks, and statistics," in *Proceedings of the Tenth Workshop on Statistical Machine Translation*, Lisbon, Portugal, 2015, pp. 74–78.

[6] E. Biçici and D. Yuret, "Optimizing instance selection for statistical machine translation with feature decay algorithms," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 2, pp. 339–350, 2015.

[7] R. Sennrich, B. Haddow, and A. Birch, "Improving neural machine translation models with monolingual data," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Berlin, Germany, 2016, pp. 86–96.

[8] S. Eetemadi, W. Lewis, K. Toutanova, and H. Radha, "Survey of data-selection methods in statistical machine translation," *Machine Translation*, vol. 29, no. 3-4, pp. 189–223, 2015.

[9] R. C. Moore and W. Lewis, "Intelligent selection of language model training data," in *Proceedings of the ACL 2010 conference short papers*, Uppsala, Sweden, 2010, pp. 220–224.

[10] A. Axelrod, X. He, and J. Gao, "Domain adaptation via pseudo in-domain data selection," in *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, Edinburgh, Scotland, UK., 2011, pp. 355–362.

[11] X. Li, J. Zhang, and C. Zong, "One Sentence One Model for Neural Machine Translation," in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, 2018, pp. 910–917.

[12] V. Levenshtein, "Binary codes capable of correcting deletions, insertions and reversals," in *Soviet Physics Doklady*, vol. 10, no. 8, 1966, pp. 707–710.

[13] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[14] E. Biçici, "Feature decay algorithms for fast deployment of accurate statistical machine translation systems," in *Proceedings of the Eighth Workshop on Statistical Machine Translation*, Sofia, Bulgaria, August 2013, pp. 78–84.

[15] A. Poncelas, A. Way, and A. Toral, "Extending feature decay algorithms using alignment entropy," in *International Workshop on Future and Emerging Trends in Language Technology*, Seville, Spain, 2016, pp. 170–182.

[16] A. Poncelas, G. M. de Buy Wenniger, and A. Way, "Applying n-gram alignment entropy to improve feature decay algorithms," *The Prague Bulletin of Mathematical Linguistics*, vol. 108, no. 1, pp. 245–256, 2017.

[17] O. Bojar, R. Chatterjee, C. Federmann, B. Haddow, M. Huck, C. Hokamp, P. Koehn, V. Logacheva, C. Monz, M. Negri, M. Post, C. Scarton, L. Specia, and M. Turchi, "Findings of the 2015 Workshop on Statistical Machine Translation," in *Proceedings of the Tenth Workshop on Statistical Machine Translation*, Lisboa, Portugal, September 2015, pp. 1–46.

[18] G. Klein, Y. Kim, Y. Deng, J. Senellart, and A. M. Rush, "Opennmt: Open-source toolkit for neural machine translation," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics-System Demonstrations*, Vancouver, Canada, 2017, pp. 67–72.

[19] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, Pennsylvania, USA, July 2002, pp. 311–318.

[20] M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul, "A study of translation edit rate with targeted human annotation," in *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas*, Cambridge, Massachusetts, USA, 2006, pp. 223–231.

[21] S. Banerjee and A. Lavie, "Meteor: An automatic metric for MT evaluation with improved correlation with human judgments," in *Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, Ann Arbor, Michigan, 2005, pp. 65–72.

[22] M. Popovic, "chrF: character n-gram F-score for automatic MT evaluation," in *Proceedings of the Tenth Workshop on Statistical Machine Translation*, Lisbon, Portugal, 2015, pp. 392–395.

[23] J. H. Clark, C. Dyer, A. Lavie, and N. A. Smith, "Better hypothesis testing for statistical machine translation: Controlling for optimizer instability," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, Portland, Oregon, 2011, p. 176–181.

[24] P. Koehn, “Statistical significance tests for machine translation evaluation,” in *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, Barcelona, Spain, 2004, pp. 388–395.