PBML

The Prague Bulletin of Mathematical Linguistics
NUMBER 108 JUNE 2017 245–256

# Applying N-gram Alignment Entropy to Improve Feature Decay Algorithms

Alberto Poncelas, Gideon Maillette de Buy Wenniger, Andy Way

ADAPT Centre, School of Computing,
Dublin City University, Dublin, Ireland

**Abstract**

Data Selection is a popular step in Machine Translation pipelines. Feature Decay Algorithms (FDA) is a technique for data selection that has shown a good performance in several tasks. FDA aims to maximize the coverage of *n*-grams in the test set. However, intuitively, more ambiguous n-grams require more training examples in order to adequately estimate their translation probabilities. This ambiguity can be measured by alignment entropy. In this paper we propose two methods for calculating the alignment entropies for *n*-grams of any size, which can be used for improving the performance of FDA. We evaluate the substitution of the n-gram-specific entropy values computed by these methods to the parameters of both the exponential and linear decay factor of FDA. The experiments conducted on German-to-English and Czech-to-English translation demonstrate that the use of alignment entropies can lead to an increase in the quality of the results of FDA.

## 1. Introduction

In recent years the amount of data available has increased significantly. Now it is possible to find vast amounts of data for use as training data in Machine Learning. The field of Statistical Machine Translation (SMT) is no exception to this phenomenon. However, as shown in Ozdowska and Way (2009), having more data does not always lead to better results. In contrast, the performance can increase by limiting the training data to a smaller but more relevant set. This is why the use of data selection techniques has become a common step in the creation of an MT pipeline.

 Corresponding author: alberto.poncelas@adaptcentre.ie
Cite as: Alberto Poncelas, Gideon Maillette de Buy Wenniger, Andy Way. Applying N-gram Alignment Entropy to Improve Feature Decay Algorithms. The Prague Bulletin of Mathematical Linguistics No. 108, 2017, pp. 245–256. doi: 10.1515/pralin-2017-0024.

The data selection technique we are using in this paper is Feature Decay Algorithms (FDA) (Biçici and Yuret, 2011; Biçici et al., 2015; Biçici and Yuret, 2015) which has obtained good results in several Workshops on both MT and quality estimation tasks. FDA collects a limited set of best sentence pairs for model training from a parallel training corpus using the (source-side) information of the test set. FDA first extracts features from the test set, and initializes them. Then, for every sentence selection iteration, FDA: 1) re-scores these features based on the already selected sentences and 2) selects the best sentence from the parallel corpus given the re-scored features, and adds it to the selected training data.

There have been previous attempts to improve FDA by using alignment entropies for unigram features (Poncelas et al., 2016). This makes sentences containing specific unigrams more (or less) likely to be selected and thus different numbers of occurrences of those unigrams are obtained in the final training data.

In this paper we propose two methods that can be used for calculating not only the alignment entropies of a unigram, but for any $n$-gram of any size. In addition we explore the performance of these methods when used to determine the value of different parameters in the mathematical model of FDA.

We perform experiments on German-to-English and Czech-to-English translation and show that it is possible to calculate a set of weights that can be used to extend FDA and obtain better results according to several evaluation metrics.

The remainder of the article is structured as follows. In Section 2 we give an outline of work that is closely related to this paper. In Section 3 we describe different extensions we propose to improve the performance of FDA. In Section 4 we describe the experiments we have designed and describe the data that has been used. In Section 5 we analyse the obtained results and perform a comparison for the different proposed extensions. We conclude in Section 6 and provide avenues for future work.

## 2. Related Work

The technique of data selection to be used is FDA. This is a method for selecting a subset from a set of parallel sentences to be used as training data for a Machine Translation System. This technique performs data selection by iteratively obtaining the most appropriate sentence pairs from a candidate pool and adding them to a selected pool, which ultimately becomes the training data when the process finishes.

### 2.1. Feature Decay Algorithms

FDA is a method that aims to maximize the coverage of $n$-grams in the test set. It does so by scoring each sentence during sentence selection as a weighted sum of the words, or more generally $n$-grams, which that particular sentence covers from the test set (the document we want to translate). Furthermore, the weight of previously selected n-grams is decreased in proportion to the number of times the $n$-gram has

already been included. This process is called feature decay. Once all the sentences have been scored, the one with the highest score will be transferred from the candidate pool and included in the selected pool. This process is iteratively repeated.

The values of the features of the selected sentence are decreased as in (1):

$$decay(f) = init(f)\frac{d^{C_L(f)}}{(1 + C_L(f))^c} \quad (1)$$

L is the selected pool, c is the linear decay factor, while d is the exponential decay factor.[1]

$C_L(f)$ is the count of the feature f in L, which makes the most frequent features decay faster, thereby allowing an increase in variability of *n*-grams in the training data. The initialization function is defined in (2):

$$init(f) = \log(|U|/C_U(f))^i |f|^l \quad (2)$$

where $|U|$ is the size of the training data, $C_U(f)$ is the count of the feature f in the training data and $|f|$ is the number of tokens of f.

### 2.2. Alignment Entropy of Unigram as Extension of FDA

FDA treats all *n*-grams equally, the default parameters of (2) are static. It does not distinguish according to how ambiguous the translation of an *n*-gram is. But intuitively, more ambiguous n-grams require more training examples in order to adequately estimate their translation probabilities. For example, proper names like "Smith" that can be unambiguously translated require fewer occurrences in a training set. Therefore the importance of this feature should decay faster than other words such as "for" or "at" which can have several possible translations.

A method for measuring how ambiguous the translations are for a given *n*-gram is to use alignment entropy. Entropy measures uncertainty, as defined in 3:

$$entropy(x) = -\sum_i p(x_i) * \log(p(x_i)) \quad (3)$$

The alignment entropy can be calculated by using the alignment probabilities in (3). These alignment probabilities can be retrieved from word-alignment models like FastAlign (Dyer et al., 2013) or GIZA++ (Och and Ney, 2003).

Let s be an *n*-gram in the source language and t an *n*-gram in the target language. We can define $A_s$ as the set of *n*-grams in the target language that are potential trans-

[1]Strictly speaking, for c in the range $(0, 1)$, c, in the denominator of formula (1), adds decay that is sublinear in $C_L(f)$, while for c in the range $(1, \infty)$ it adds decay that increases faster than linear, though not exponential. However, in the experiments in this paper, c is in the range $(0, 1)$, so the effect the factor involving c is at most linear, so we just refer to it as "linear" for simplicity.

lations of $s$, and $p(s,t)$ be the probability of $s$ being translated as $t$. Accordingly, the alignment entropy of $s$ can be calculated as in (4):

$$\mathtt{alignEnt}(s) = \frac{\sum_{t \in A_s} p(s,t) * \log(p(s,t))}{\log(|A_s|)} \quad (4)$$

In order to have alignment entropies in the $[0,1]$ range, the entropies are divided by the the $\log$ of the number of possible translations, $\log(|A_s|)$, which is the maximum possible entropy.

The score obtained in (4) can be used in (1) as the value of one of the decay factors, $d$ or $c$. As a result the alignment entropy can have an influence on the decay.

In (Poncelas et al., 2016) experiments were carried out using unigrams as features and changing the parameter $d$ in (1). The alignment probabilities were obtained by using FastAlign and GIZA++, showing that probabilities calculated by GIZA++ achieved better results.

## 3. Computing and Applying Alignment Entropies

In this paper we propose two possible alternatives for estimating the alignment entropy of a any order *n*-gram. In addition, we want to explore the performance when extending the different decay factors.

### 3.1. Extending the Exponential and Linear Decay in FDA

In FDA, the decay function (1) has two parameters: the linear decay factor $c$ in the range $[0,\infty)$ with a default value of $0.0$, and exponential factor $d$, in range $(0,1]$ with a default value of $0.5$. We are interested in exploring the impact in the performance when changing these values. The aim is to analyze the three possible combinations: change exponential decay exclusively, linear decay exclusively, and both the exponential and linear decay. Note that when changing both decay factors we are using the same set of weights in both parameters.

### 3.2. Computing 3-gram Alignment Entropy in FDA

While the unigram alignment entropy can be computed by using the conditional probabilities retrieved from FastAlign or GIZA++ (because they are already word-to-word translation probabilities), computing an *n*-gram alignment is not straightforward. It is not reasonable to expect that, for example, a 3-gram in the source language should be mapped to a 3-gram in the target language as well.

In order to estimate the alignment entropy for any size *n*-grams we propose the following two alternative entropy instantiations:

A *mean-of-unigram method*: Compute the alignment entropy of the unigrams using an alignment tool. For the words whose alignments could not be retrieved, we

assign them an entropy equal to the mean of the entropies of the rest of the words. Then we can estimate the entropy of the *n*-gram as the mean of the entropies of the words in the *n*-gram.

B *ngram-to-unigram method*: Assume that for every sentence pair $\langle l_s, l_t \rangle$, an *n*-grams s in the source sentence $l_s$ is only aligned to a single word (unigram) chosen from the target sentence $l_t$ with which it appears. Furthermore, assume all these alignments are equally likely. Then to compute the alignment entropy for s:

1) Extract from the parallel corpora the set L of line-pairs $\langle l_s, l_t \rangle$ that contain s in in the source side: $L = \{\langle l_s, l_t \rangle : s \in l_s\}$
2) Compute a multiset $S_s$ of translation tuples containing s:
   For every line-pair $\langle l_s, l_t \rangle \in L$, for every word $w_t \in l_t$ extract an n-gram alignment tuple $\langle s, w_t \rangle$. (Assuming every words *w* in the target side is a potential translation candidate for s)
3) Compute the alignment probability distribution $P_s$ from $S_s$ using relative frequence estimation.
4) Finally, compute the entropy over the thus computed distritutions $P_s$.

We expect *n*-grams with lower entropies to be aligned to a lower variety of words on the target side. This provides us with an estimation of how difficult is to find a translation. In addition *n*-grams that tend to appear in in-domain contexts will have less translation candidates and therefore lower entropies. The probabilities calculated using this method can be used in (4) for computing the alignment entropy of the *n*-gram.

## 4. Experiments

The goal of the designed experiments is to test the effect on the performance of the different aligment entropies (explained in Section 3.2) used when changing different decay factors (explained in Section 3.1). We will refer to this modified factor as entropy-modified decay. Therefore, the designed experiments are the following:

- Baseline experiment: Execute FDA with the default values in the parameters.
- *mean-of-unigram* experiment: Use as alignment entropy H the mean of the alignment entropy retrieved by GIZA++ of its containing words (method A in the section 3.2). Substitute H for c (linear decay), d (exponential decay) or both.
- *ngram-to-unigram* experiment: Calculate the alignment entropy H as if the *n*-grams were aligned to a single word in the target side (method B in the section 3.2). Substitute H for c (linear decay), d (exponential decay) or both.

We are interested in observing the effect of these variants in different languages and using features of different sizes. Therefore each of these experiments were carried out using German (a language with a relatively strict word order), and Czech (a language with free word order) as source languages. In addition, we used FDA1

(using unigram as features) and FDA3 (features of up to 3-grams, which is what FDA computes by default).

The data sets used in the experiments are based on the ones used in the work of Biçici (2013) and Poncelas et al. (2016): (i) *Languages*: German-to-English and Czech-to-English; (ii) *Training data*: The training data provided in the WMT 2015 (Bojar et al., 2015) translation task setting a maximum sentence length of 126 words (4.5M sentence pairs, 225M words, in German-to-English corpus and 11M sentence pairs, 355M words, in Czech-to-English corpus); (iii) *Tuning data*: We use 5K randomly sampled sentences from development sets from previous years; (iv) *Language Model*: 8-gram Language Model (LM) built using the target-language side of the selected data via the KenLM toolkit (Heafield, 2011) using Kneser-Ney smoothing; (v) *Selected sentences*: Select 66.4 million words in total (source- and target-language sides) in each experiment; (vi) *Test set*: Documents provided in the WMT 2015 Translation Task.

We train SMT systems on the selected data using the Moses toolkit (Koehn et al., 2007) with the standard features and using GIZA++ for word alignment. We include several evaluation metrics: BLEU (Papineni et al., 2002), NIST (Doddington, 2002), TER (Snover et al., 2006), METEOR (Banerjee and Lavie, 2005) and CHRF (Popovic, 2015). These scores give an estimation of the quality of the output of the experiment when comparing to a translated reference. In general, the higher the score is, the better the translation is estimated to be (except for TER, which is a translation error measure and so lower is better).

## 5. Results

In Table 1 and Table 2 we present the mean of 4 MERT (Och, 2003) tuning executions for the different experiments. In the columns we show the baseline (FDA with default values, $\mathtt{d} = 0.5$ and $\mathtt{c} = 0.0$), exponential decay (FDA substituting the entropies for d and keeping $\mathtt{c} = 0.0$), linear decay (FDA substituting the entropies for c and keeping $\mathtt{d} = 0.5$) and linear and exponential decay (substituting the entropies for both c and d). In Table 1 and Table 2 we also compute statistical significance at level p=0.01 when compared with the baseline using Bootstrap Resampling (Koehn, 2004) for BLEU, TER and METEOR scores.

We can observe that choosing good alignment entropies combined with changing the proper decay factors can obtain better results than the default baseline. In this section we compare the performance of the extensions for FDA1 and FDA3, the comparison of changing different decay factors, and the comparison of the obtained alignment entropies.

### 5.1. Comparison of FDA1 with FDA3

Considering that the features extracted in FDA1 are a subset of the ones from FDA3 one would expect to have better results when using features of larger order *n*-grams.

| | **baseline** | | **entropy-modified exponential decay** | | **entropy-modified linear decay** | | **entropy-modified linear and exponential decay** | |
|---|---|---|---|---|---|---|---|---|
| | FDA1 | FDA3 | FDA1 | FDA3 | FDA1 | FDA3 | FDA1 | FDA3 |
| de → en | | | | | | | | |
| BLEU | 0.2285 | 0.2282 | 0.2170 | 0.2235 | 0.2276 | **0.2307*** | 0.2198 | 0.2232 |
| NIST | 6.9407 | 6.9237 | 6.7984 | 6.8734 | 6.9124 | **6.9573** | 6.8345 | 6.8825 |
| TER | 0.5966 | 0.5955 | 0.6035 | 0.5982 | 0.5989 | **0.5918*** | 0.6002 | 0.5981 |
| METEOR | 0.2864 | 0.2851 | 0.2804 | 0.2827 | 0.2842 | **0.2859*** | 0.2819 | 0.2832 |
| CHRF3 | 50.124 | 49.937 | 49.001 | 49.528 | 49.854 | 49.884 | 49.321 | 49.743 |
| CHRF1 | 50.727 | 50.705 | 49.841 | 50.265 | 50.553 | **50.836** | 50.077 | 50.301 |
| cs → en | | | | | | | | |
| BLEU | 0.2127 | 0.2184 | 0.2102 | 0.2146 | 0.2121 | **0.2190** | 0.2073 | 0.2137 |
| NIST | 6.6518 | 6.6983 | 6.6295 | 6.6375 | 6.6408 | **6.7004** | 6.5740 | 6.6247 |
| TER | 0.5973 | 0.5955 | 0.6221 | 0.6205 | 0.6202 | 0.6152 | 0.6252 | 0.6200 |
| METEOR | 0.2805 | 0.2827 | **0.2815** | 0.2806 | 0.2805 | **0.2832** | 0.2790 | 0.2796 |
| CHRF3 | 48.178 | 48.578 | 48.078 | 48.316 | 48.029 | **48.605** | 47.647 | 48.160 |
| CHRF1 | 49.250 | 49.604 | 49.145 | 49.245 | 49.201 | 49.589 | 48.822 | 49.161 |

*Table 1. Results of the average of the scores after 4 tuning executions for the baseline, and* mean-of-unigram *experiment. The results in bold indicate an improvement over the baseline. The asterisk means the result is statistically significant.*

However, we observe that it is not always the case. An example of this is the German-to-English translation for the default FDA. As we can see in the baseline column in Table 1 or Table 2 the results when using features of size 1 are better than those of size 3 for the BLEU, NIST, METEOR, CHFR3 and CHFR1 evaluation scores.

We also observe that the extensions proposed in this paper affect FDA3 and FDA1 differently. Extensions that improve an evaluation metric in FDA1 do not necessarily translate into improvements in FDA3. The METEOR score for Czech-to-English translation in Table 1 (entropy-modified exponential decay column) increases from 0.2805 (the baseline) to 0.2815, while the same evaluation score in FDA3 decreases from 0.2875 to 0.2806. The opposite is also true, not all the extensions yielding improvements with FDA3 do the same with FDA1.

### 5.2. Exponential Decay vs Linear Decay

Looking at Table 1 and Table 2, we observe that it is not necessarily preferable to change one decay factor over the other. Different sets of weights perform better

| | **baseline** | | **entropy-modified exponential decay** | | **entropy-modified linear decay** | | **entropy-modified linear and exponential decay** | |
|---|---|---|---|---|---|---|---|---|
| | FDA1 | FDA3 | FDA1 | FDA3 | FDA1 | FDA3 | FDA1 | FDA3 |
| de → en | | | | | | | | |
| BLEU | 0.2285 | 0.2282 | 0.2271 | **0.2297** | 0.2247 | **0.2286** | 0.2278 | **0.2305*** |
| NIST | 6.9407 | 6.9237 | 6.9270 | **6.9618** | 6.9107 | **6.9284** | 6.9367 | **6.9713** |
| TER | 0.5973 | 0.5955 | 0.5997 | 0.5974 | 0.5982 | 0.5967 | 0.5982 | 0.5966 |
| METEOR | 0.2864 | 0.2851 | 0.2851 | **0.2869*** | 0.2846 | 0.2849 | 0.2856 | **0.2867*** |
| CHRF3 | 50.124 | 49.937 | 50.075 | **50.221** | 49.957 | 49.771 | 50.070 | **50.263** |
| CHRF1 | 50.727 | 50.705 | 50.640 | **50.826** | 50.517 | 50.679 | 50.721 | **50.857** |
| cs → en | | | | | | | | |
| BLEU | 0.2127 | 0.2184 | 0.2088 | **0.2202*** | **0.2145*** | **0.2197** | **0.2142*** | **0.2211*** |
| NIST | 6.6518 | 6.6983 | **6.5560** | **6.7224** | **6.6712** | **6.7136** | **6.6630** | **6.7447** |
| TER | 0.6187 | 0.6154 | 0.6296 | **0.6140** | **0.6182** | **0.6152** | **0.6184** | **0.6127*** |
| METEOR | 0.2805 | 0.2827 | 0.2799 | **0.2844*** | **0.2816*** | **0.2832** | **0.2817*** | **0.2851*** |
| CHRF3 | 48.178 | 48.578 | 47.866 | **48.768** | **48.293** | **48.666** | **48.365** | **48.827** |
| CHRF1 | 49.250 | 49.604 | 48.950 | **49.736** | **49.344** | **49.630** | **49.392** | **49.850** |

*Table 2. Results of the average of the scores after 4 tuning executions for the baseline, and* ngram-to-unigram *experiment. The results in bold indicate an improvement over the baseline. The asterisk means the result is statistically significant*

changing different decay factors. For example, in FDA3, the scores obtained in the *mean-of-unigram* experiment work better for most of the scores when changing the linear decay factor, while in *ngram-to-unigram* experiment changing the exponential decay performs better for almost every score.

In FDA1 the use of our novel extension is even more unclear, as the only statistically significant improvement occurs in Czech-to-English translation when changing the linear decay (BLEU and METEOR rows in Table 2).

### 5.3. Changing One Decay Factor vs Changing Both Decay Factors

In Section 5.2, we have concluded that the performance of the decay factor depends on the set of weights used as inputs. Note that in these experiments we change both factors with the same values, so we propose as future work a more fine-grained evaluation of the performance using different entropies in each decay factor. Despite the dependency on the weights, we find that, in FDA3, it is possible to find a set (Table

| | de → en | | cs → en | |
|---|---|---|---|---|
| | mean | std | mean | std |
| mean-of-unigram | 0.6008 | 0.2035 | 0.5333 | 0.1926 |
| ngram-to-unigram | 0.7450 | 0.1244 | 0.7314 | 0.1310 |

*Table 3. Mean and standard deviation of the aligment entropies distribution for FDA3.*

2, last column) that can improve the baseline for almost every score[2], and it is the only extension in obtaining statistically significant improvement for more than one evaluation metric in both languages.

### 5.4. Comparison of *mean-of-unigram method* and *ngram-to-unigram method*

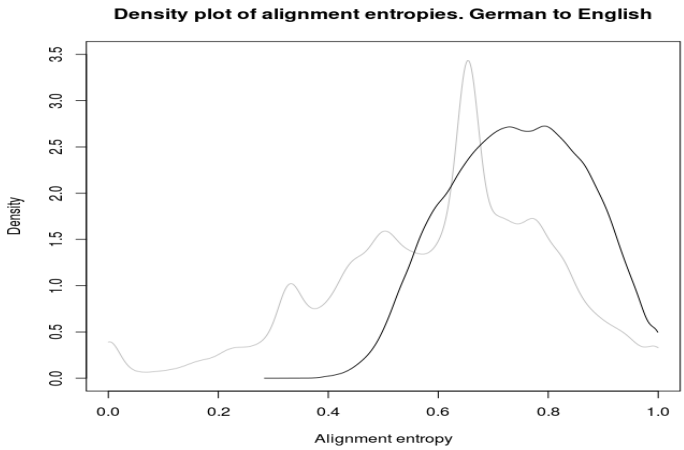


*Figure 1. Density plot of the alignment entropies obtained in* mean-of-unigram *(grey) and* ngram-to-unigram *(black) experiments for FDA3 and for German-to-English translation.*

In The *ngram-to-unigram* experiment we are assuming that every word in the target language may be a potential candidate translation for a given $n$-gram. Therefore we expect it to produce a set of higher entropies.

In order to have a deeper understanding of the distributions of the entropies in the experiments, in Figure 1 and Figure 2 we show the distribution for German-to-English and Czech-to-English translations, respectively. In Table 5.4 we also include

[2]The single case where the score is not improved, is the TER score for German-English translation.

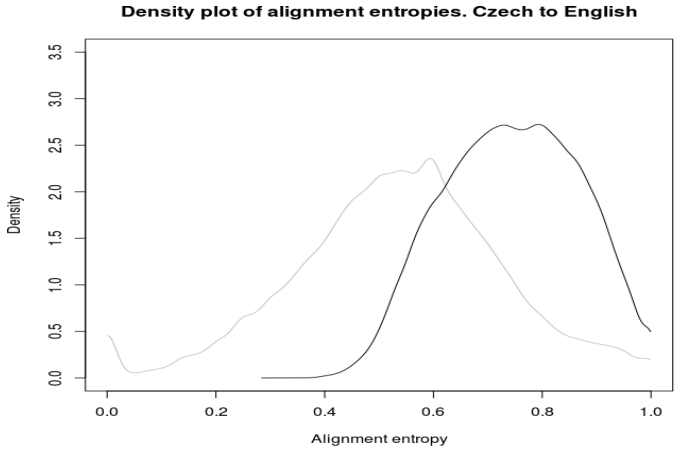


*Figure 2. Density plot of the alignment entropies obtained in* mean-of-unigram *(grey) and* ngram-to-unigram *(black) experiments for FDA3 and for Czech-to-English translation.*

the statistics of these distributions. They confirm our hypothesis that distribution for *ngram-to-unigram* is centered in higher entropies: 0.745 and 0.7314. In contrast, for *mean-of-unigram* they are 0.6008 and 0.5333. Note also that none of the entropies in *ngram-to-unigram* experiment have a value below 0.3. This makes de values of the features in *ngram-to-unigram* experiment decay slower.

We can observe that the results obtained by the *ngram-to-unigram* experiment for FDA3 are generally better than those of *mean-of-unigram*. While in the first case (Table 1) only one extension performs better than the baseline, in the second case (Table 2) in every extension we obtain improvements for at least two evaluation metrics.

For FDA1, even if the results are not equally satisfactory, we can observe statistically significant improvements in the *ngram-to-unigram* experiment for two of the extensions in Czech-to-English translation.

## 6. Conclusions and Future Work

In this work we have tried to improve the results of FDA by setting new, n-gram-specific, weights in the decay function, that depend on the uncertainty of the n-grams. In order to do that we proposed two methods for calculating the uncertainty. These methods give an insight into the amount of occurrences an *n*-gram needs in the training data, based on how ambiguous the translation is. We observe that different weights work better for different parameters. Accordingly, finding a good set of values is not enough; it is also necessary to find which parameter performs better. However we demonstrated that it is possible to find a combination that can have a positive

impact on the output. Our findings have proven to be useful both for German-to-English and Czech-to-English translation. An additional finding in this work is that when using unigram features in the default FDA set-up, the output can be as good as (or even better than) using higher order $n$-gram features.

In the future, we intend to conduct experiments to explore whether having different distributions of the entropies (e.g. more left or right skewed, or different standard deviations) can improve the results. The entropies used in this work where the same for exponential and linear decay factors. Having different sets of weights for each parameter might be beneficial. In addition we want to analyse the outcome when using alignment entropies as input to the init function as well. The source languages used in this work are morphologically richer than the target language. We are also interested in knowing if the improvements are preserved when performing the translation in the reverse direction.

Finally, we want to find a method for obtaining an optimal size of the selected training data.

## Acknowledgements

This research is supported by the ADAPT Centre for Digital Content Technology, funded under the SFI Research Centres Programme (Grant 13/RC/2106). This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 713567.

## Bibliography

Banerjee, Satanjeev and Alon 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*, pages 65–72, Ann Arbor, Michigan, 2005.

Biçici, Ergun. Feature decay algorithms for fast deployment of accurate statistical machine translation systems. 2013.

Biçici, Ergun and Deniz Yuret. Instance selection for machine translation using feature decay algorithms. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 272–283, Edinburgh, Scotland, 2011.

Biçici, Ergun and Deniz Yuret. Optimizing instance selection for statistical machine translation with feature decay algorithms. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(2):339–350, 2015.

Biçici, Ergun, Qun Liu, and Andy Way. ParFDA for fast deployment of accurate statistical machine translation systems, benchmarks, and statistics. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 74–78, Lisbon, Portugal, 2015.

Bojar, Ondřej, Rajen Chatterjee, Christian Federmann, Barry Haddow, Matthias Huck, Chris Hokamp, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Carolina Scarton, Lucia Specia, and Marco Turchi. Findings of the 2015 Workshop on Statistical

Machine Translation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 1–46, Lisboa, Portugal, September 2015. Association for Computational Linguistics. URL http://aclweb.org/anthology/W15-3001.

Doddington, George. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the second international conference on Human Language Technology Research*, pages 138–145, San Diego, CA, 2002.

Dyer, Chris, Victor Chahuneau, and Noah Smith. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings*, pages 644–648, Atlanta, Georgia, USA, 2013.

Heafield, Kenneth. KenLM: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197, Edinburgh, Scotland, 2011.

Koehn, Philipp. Statistical Significance Tests for Machine Translation Evaluation. In *Proceedings of EMNLP 2004*, pages 388–395, Barcelona, Spain, 2004.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open source toolkit for SMT. In *Proceedings of 45th annual meeting of the ACL on interactive poster & demonstration sessions*, pages 177–180, Prague, Czech Republic, 2007.

Och, Franz. Minimum error rate training in statistical machine translation. In *ACL-2003: 41st Annual Meeting of the Association for Computational Linguistics, Proceedings*, pages 160–167, Sapporo, Japan, 2003.

Och, Franz and Hermann Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, 2003.

Ozdowska, Sylwia and Andy Way. Optimal bilingual data for French-English PB-SMT. 2009.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th ACL*, pages 311–318, Philadelphia, PA, USA, 2002.

Poncelas, Alberto, Andy Way, and Antonio Toral. Extending Feature Decay Algorithms using Alignment Entropy. 2016.

Popovic, Maja. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal, 2015.

Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John 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*, pages 223–231, Cambridge, Massachusetts, USA, 2006.

**Address for correspondence:**
Alberto Poncelas
alberto.poncelas@adaptcentre.ie
ADAPT Centre, School of Computing,
Dublin City University, Ireland