Online learning of log-linear weights in interactive machine translation

Francisco-Javier López-Salcedo, Germán Sanchis-Trilles, and Francisco Casacuberta

Pattern Recognition and Human Language Technologies Group
Departamento de Sistemas Informáticos y Computación
Universitat Politècnica de València
{flopez,gsanchis,fcn}@dsic.upv.es

Abstract. Whenever the quality provided by a machine translation system is not enough, a human expert is required to correct the sentences provided by the machine translation system. In this environment, the human translator is generating bilingual data after each translation has been marked as correct, and expects the system to be able to learn from the errors made. In this paper, we analyse the appropriateness of discriminative ridge regression for adapting the scaling factors of a state-of-the-art machine translation system within a conventional post-editing scenario and also within an interactive machine translation setup. Results show that the strategies applied in the former setup cannot be directly applied in the latter framework. Hence, the discriminative ridge regression is revised and adapted for the interactive machine translation framework, with encouraging results.

Keywords: Machine translation, online learning, interactive machine translation

1 Introduction

Machine translation is not only needed in fields where the amount of data is overwhelming, but also in fields where bilingual data is less abundant, but translation quality is critical. In these scenarios, machine translation systems need to collaborate closely with human experts, with the purpose of achieving high quality translations efficiently, giving rise to the popularisation of the *computer assisted translation* (CAT) [1] paradigm. In such paradigm, the *statistical machine translation* (SMT) [2] system proposes a hypothesis to a human translator, who amends the hypothesis to obtain an acceptable target sentence. Two different user interaction schemes will be considered in this paper. The first one, *post-editing* (PE), is being embraced by more and more human translators as an efficient way of generating high-quality translations. In PE, the SMT system provides an initial translation, and then the user modifies such translation so as to correct it. The second one, *interactive machine translation* (IMT) [3, 4], is a more cutting-edge technology which has been receiving an increasing amount of attention. The IMT system attempts to predict the text the user is going to input. Whenever such prediction is wrong and the user provides feedback to the system, a new prediction is performed. Such process is repeated until the translation is considered correct.

One important problem which SMT systems need to tackle with when used for a CAT purpose is adaptability. In these scenarios, the user expects the system to learn

dynamically from its own errors, so that errors corrected once do not need to be corrected over and over again. Hence, the models need to be adapted *online*, i.e. without a complete retraining of the model parameters, since such retraining would be too costly.

The grounds of modern SMT were established in [5], by formulating the SMT problem as follows: given an input sentence $\boldsymbol{x}$ in a certain source language, the best translation $\hat{\boldsymbol{y}}$ in a certain target language is to be found:

$$\hat{\boldsymbol{y}} = \underset{\boldsymbol{y}}{\operatorname{argmax}} \ \Pr(\boldsymbol{y} \mid \boldsymbol{x}), \tag{1}$$

where $\Pr(\boldsymbol{y} \mid \boldsymbol{x})$ is modelled directly by the so-called log-linear models [6], yielding

$$\hat{\boldsymbol{y}} = \underset{\boldsymbol{y}}{\operatorname{argmax}} \sum_{m=1}^{M} \lambda_m h_m(\boldsymbol{x}, \boldsymbol{y}) = \underset{\boldsymbol{y}}{\operatorname{argmax}} \, \boldsymbol{\lambda} \cdot \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{y}) = \underset{\boldsymbol{y}}{\operatorname{argmax}} \, g(\boldsymbol{x}, \boldsymbol{y}), \tag{2}$$

where $h_m(\boldsymbol{x}, \boldsymbol{y})$ represents an important feature for the translation of $\boldsymbol{x}$ into $\boldsymbol{y}$, M is the number of models (or features) and λ_m are the weights acting as scaling factors of the score functions. $g(\boldsymbol{x}, \boldsymbol{y})$ represents the score of a hypothesis $\boldsymbol{y}$ given an input sentence $\boldsymbol{x}$, and is not treated as a probability since the normalisation term has been omitted. Common feature functions $h_m(\boldsymbol{x}, \boldsymbol{y})$ include translation models, re-ordering models or the target language model. Typically, $\boldsymbol{h}$ and $\boldsymbol{\lambda}$ are estimated by means of training and development sets, respectively. However, the domain of such sets has an important impact on the final translation quality [7], and adaptation arises as an efficient way of alleviating this fact by using very limited amounts of in-domain data. In this paper, only $\boldsymbol{\lambda}$ will be adapted, although the same methods could also be applied to adapt $\boldsymbol{h}$ [8].

This paper is structured as follows. Next section briefly reviews related work. Then, in Sec. 3, a short introduction to IMT is presented. Sec. 4 reviews discriminative ridge regression and the modifications needed to apply it within IMT. Experiments are described in Sec. 5, and the last section is reserved for conclusions and future work.

2 Related work

Batch adaptation (as opposed to online) is a very broad field that has been receiving a large amount of attention. In [9], adaptation in speech recognition is confronted by means of the maximum likelihood framework. In [10], the maximum likelihood framework is expanded so as to obtain maximum a posteriori estimators. In [11], adaptation is confronted as a classification problem, by extending the set of features by an additional tag. In [12], Bayesian predictive adaptation is applied for adapting $\boldsymbol{\lambda}$ in a batch setup.

However, there are also cases where there is no adaptation data at all available beforehand, and the system needs to adapt itself online without falling into an excessive time burden. Such problem led, among others, to the development of an incremental version of the Expectation-Maximisation algorithm [13]. This algorithm has been successfully applied in an IMT scenario in [14], where the models involved are incrementally updated as the user feedback is received.

In [8], an in-depth comparison of four online adaptation algorithms, i.e. passive-aggressive, perceptron, discriminative ridge regression and Bayesian predictive adaptation are studied for their application in a post-editing scenario. Both $\boldsymbol{\lambda}$ and $\boldsymbol{h}$ are

SOURCE $(\boldsymbol{x})$**:** Para encender la impresora:
REFERENCE $(\boldsymbol{y})$**:** To power on the printer:

ITER-0	$(\boldsymbol{p})$	( )
	$(\hat{\boldsymbol{s}})$	*To switch on:*
ITER-1	$(\boldsymbol{p})$	To
	(k)	power
	$(\hat{\boldsymbol{s}})$	*on the printer:*
ITER-2	$(\boldsymbol{p})$	To power on the printer:
	(k)	(#)
	$(\hat{\boldsymbol{s}})$	( )
FINAL	$(\boldsymbol{p} \equiv \boldsymbol{y})$	To power on the printer:

Fig. 1. IMT session to translate a Spanish sentence into English. Non-validated hypotheses are displayed in italics, whereas accepted prefixes are printed in normal font.

adapted, alternatively, presenting the most promising results when adapting the scaling factors $\boldsymbol{\lambda}$. In the present paper we study the application of the best-performing algorithm, namely discriminative ridge regression, within an IMT framework, showing that such algorithm may not be applied directly, and propose an alternative approach.

3 Interactive machine translation

In IMT, the purpose is to use fully-fledged SMT systems to produce full target sentence hypotheses, or portions thereof, which can be partially or completely accepted and amended by a human translator [3]. Fig. 1 illustrates a typical IMT session. Initially, the user is given an input sentence $\boldsymbol{x}$ to be translated. The reference $\boldsymbol{y}$ provided is the translation that the user would like to achieve. At iteration 0, the IMT system has to provide an initial complete translation $\hat{\boldsymbol{s}}$, as if it were a conventional SMT system. Next, the user validates a prefix $\boldsymbol{p}$ (word "*To*") and introduces a new word k. This being done, the system suggests a new suffix $\hat{\boldsymbol{s}}$. Again, the user validates a new prefix, introduces a new word and so forth. The process continues until the whole sentence is correct. In this example, a potential user of the IMT system would have typed only one word out of five, i.e., a potential effort reduction of 80% with respect to translating the whole sentence from scratch. If a PE environment is assumed as baseline, the user would have typed three words, versus only one in the case of IMT: an effort reduction of 66%.

Formally, IMT is specified as an evolution of the SMT framework. However, Eq. 1 needs to be modified according to the IMT scenario in order to take into account the part of the target sentence that is already translated, that is $\boldsymbol{p}$ and k:

$$\hat{\boldsymbol{s}} = \underset{\boldsymbol{s}}{\operatorname{argmax}}\, Pr(\boldsymbol{s}|\boldsymbol{x}, \boldsymbol{p}, k) \tag{3}$$

where the maximisation problem is defined over the suffix $\boldsymbol{s}$. This allows us to rewrite Eq. 3, by eliminating constant terms, achieving the equivalent criterion

$$\hat{\boldsymbol{s}} = \underset{\boldsymbol{s}}{\operatorname{argmax}}\, Pr(\boldsymbol{p}, k, \boldsymbol{s}|\boldsymbol{x}). \tag{4}$$

An example of the intuition behind these variables is shown in Fig. 1.

Note that, since $(\boldsymbol{p}\, k\, \boldsymbol{s}) = \boldsymbol{y}$, Eq. 4 is very similar to Eq. 1. The main difference is that the argmax search is now performed over the set of suffixes $\boldsymbol{s}$ that complete $(\boldsymbol{p}\, k)$, instead of complete sentences ($\boldsymbol{y}$ in Eq. 1). This implies that we can use the same models if the search procedures are adequately modified [3].

Typically, the IMT system makes use of the word graph generated for a given sentence in order to complete the validated prefixes [15]. Specifically, the system finds the best path in the word graph associated with a given prefix. A word graph is a weighted directed acyclic graph, where each node represents one or more partial translation hypotheses. The edges represent transitions between such nodes, and are labelled each with one word of the target sentence, and weighted by a score which evaluates how likely it is to emit such word after having already emitted the current prefix.

4 Discriminative ridge regression

The main purpose of discriminative Ridge regression [8] (DRR) is that *good* hypothesis within a given N-best list score *higher*, and *bad* hypotheses score *lower*. It implements the estimation of $\boldsymbol{\lambda}$ as a regression problem between $g(\boldsymbol{x}, \boldsymbol{y})$, with $\boldsymbol{y} \in nbest(\boldsymbol{x})$, and the translation quality of $\boldsymbol{y}$.

In an online learning framework, the learning algorithm processes observations sequentially. The purpose is then to modify the prediction mechanisms according to the user's feedback in order to improve the quality of future translations. Considering that the user's feedback is the reference translation $\boldsymbol{y}^\tau$, Eq. 2 is redefined as follows

$$\hat{\boldsymbol{y}}_t = \underset{\boldsymbol{y}}{\operatorname{argmax}}\, \boldsymbol{\lambda}_t \cdot \boldsymbol{h}(\boldsymbol{x}_t, \boldsymbol{y}), \tag{5}$$

where the log-linear weights $\boldsymbol{\lambda}_t$ vary according to samples $(\boldsymbol{x}_1, \boldsymbol{y}_1^\tau), \ldots, (\boldsymbol{x}_{t-1}, \boldsymbol{y}_{t-1}^\tau)$ seen before time t. To simplify notation, we will omit subindex t from input sentence $\boldsymbol{x}$ and output sentence $\hat{\boldsymbol{y}}$, although it is always assumed. Either $\boldsymbol{h}_t$ or $\boldsymbol{\lambda}_t$ can be adapted, or even both at the same time. However, in this paper, we focus on adapting only $\boldsymbol{\lambda}_t$.

The hypothesis $\hat{\boldsymbol{y}}$ that maximises the likelihood is not necessarily the hypothesis with the highest quality from a human perspective or in terms of a certain quality measure. Let $\boldsymbol{y}^*$ be the hypothesis with the highest quality, but which might have a lower likelihood[1]. Our purpose is to adapt the model parameters so that $\boldsymbol{y}^*$ is rewarded and achieves a higher score according to Eq. 2.

We define the *difference* in translation quality between the proposed hypothesis $\hat{\boldsymbol{y}}$ and the best hypothesis $\boldsymbol{y}^*$ in terms of a given quality measure $\mu(\cdot)$:

$$l(\hat{\boldsymbol{y}}) = |\mu(\hat{\boldsymbol{y}}) - \mu(\boldsymbol{y}^*)|, \tag{6}$$

where the absolute value has been introduced in order to preserve generality. The score difference between $\hat{\boldsymbol{y}}$ and $\boldsymbol{y}^*$ is related to $\phi(\hat{\boldsymbol{y}})$, which is defined as

$$\phi(\hat{\boldsymbol{y}}) = g(\boldsymbol{x}, \boldsymbol{y}^*) - g(\boldsymbol{x}, \hat{\boldsymbol{y}}). \tag{7}$$

[1] $\boldsymbol{y}^*$ does not necessarily match the reference translation $\boldsymbol{y}^\tau$ due to eventual coverage problems.

Ideally, we would like differences in $l(\cdot)$ to correspond to differences in $\phi(\cdot)$: if hypothesis $\boldsymbol{y}$ has a translation quality $\mu(\boldsymbol{y})$ that is very similar to the translation quality of $\mu(\boldsymbol{y}^*)$, we would like this to be reflected in translation score g, i.e., $g(\boldsymbol{x}, \boldsymbol{y})$ is very similar to $g(\boldsymbol{x}, \boldsymbol{y}^*)$. Hence, the purpose of our online procedure should be to promote this correspondence after each sample $(\boldsymbol{x}_t, \boldsymbol{y}_t^\tau)$.

For computing the new scaling factors $\boldsymbol{\lambda}_t$, the previously learnt $\boldsymbol{\lambda}_{t-1}$ is combined, for a certain learning rate α, with an appropriate update step $\check{\boldsymbol{\lambda}}_t$, yielding [8]:

$$\boldsymbol{\lambda}_t = (1-\alpha)\boldsymbol{\lambda}_{t-1} + \alpha\check{\boldsymbol{\lambda}}_t. \tag{8}$$

Although adapting $\boldsymbol{\lambda}$ is a coarse-grained strategy, its effect cannot be underestimated, since it implies adjusting the importance of every single model in Eq. 2.

4.1 Discriminative ridge regression in post-editing

In a conventional post-editing scenario where the hypotheses are provided by a regular SMT system, the DRR algorithm requires an N-best list of hypotheses in decreasing order of likelihood. Let $nbest(\boldsymbol{x})$ be such a list computed by our models for sentence $\boldsymbol{x}$. For adapting $\boldsymbol{\lambda}$, we define an $N \times M$ matrix $\mathrm{H}_{\boldsymbol{x}}$, where M is the number of features in Eq. 2, containing the feature functions $\boldsymbol{h}$ of every hypothesis:

$$\mathrm{H}_{\boldsymbol{x}} = [\boldsymbol{h}(\boldsymbol{x}, \boldsymbol{y}_1), \ldots, \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{y}_N)]'. \tag{9}$$

Additionally, let $\mathrm{H}^*_{\boldsymbol{x}}$ be a matrix such that

$$\mathrm{H}^*_{\boldsymbol{x}} = [\boldsymbol{h}(\boldsymbol{x}, \boldsymbol{y}^*), \ldots, \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{y}^*)]', \tag{10}$$

where all rows are identical and equal to the feature vector of the best hypothesis $\boldsymbol{y}^*$ within the N-best list. Then, $\mathrm{R}_{\boldsymbol{x}}$ is defined as

$$\mathrm{R}_{\boldsymbol{x}} = \mathrm{H}^*_{\boldsymbol{x}} - \mathrm{H}_{\boldsymbol{x}}\ . \tag{11}$$

The key idea is to find a vector $\check{\boldsymbol{\lambda}}_t$ such that differences in scores are reflected as differences in the quality of the hypotheses. That is,

$$\mathrm{R}_{\boldsymbol{x}} \cdot \check{\boldsymbol{\lambda}}_t \propto \mathbf{l}_{\boldsymbol{x}}\ , \tag{12}$$

where $\mathbf{l}_{\boldsymbol{x}}$ is a column vector of N rows such that

$$\mathbf{l}_{\boldsymbol{x}} = [l(\boldsymbol{y}_1) \ldots l(\boldsymbol{y}_n) \ldots l(\boldsymbol{y}_N)]'\ ,\ \ \forall \boldsymbol{y}_i \in nbest(\boldsymbol{x}). \tag{13}$$

The objective is to find $\check{\boldsymbol{\lambda}}_t$ such that

$$\check{\boldsymbol{\lambda}}_t = \operatorname*{argmin}_{\boldsymbol{\lambda}} |\mathrm{R}_{\boldsymbol{x}} \cdot \boldsymbol{\lambda} - \mathbf{l}_{\boldsymbol{x}}| \tag{14}$$

$$= \operatorname*{argmin}_{\boldsymbol{\lambda}} ||\mathrm{R}_{\boldsymbol{x}} \cdot \boldsymbol{\lambda} - \mathbf{l}_{\boldsymbol{x}}||^2, \tag{15}$$

where $||\cdot||^2$ is the Euclidean norm. Although Eqs. 14 and 15 are equivalent (i.e. the $\hat{\boldsymbol{\lambda}}$ that minimises the first one also minimises the second one), Eq. 15 allows for a direct

implementation thanks to the ridge regression[2], such that $\check{\boldsymbol{\lambda}}_t$ can be computed as the solution to the overdetermined system $\mathrm{R}_{\boldsymbol{x}} \cdot \check{\boldsymbol{\lambda}}_t = \mathbf{l}_{\boldsymbol{x}}$, given by

$$\check{\boldsymbol{\lambda}}_t = \left(\mathrm{R}'_{\boldsymbol{x}} \cdot \mathrm{R}_{\boldsymbol{x}} + \beta \mathrm{I}\right)^{-1} \mathrm{R}'_{\boldsymbol{x}} \cdot \mathbf{l}_{\boldsymbol{x}} \ , \tag{16}$$

where a small β is used as a regularisation term to stabilise $\mathrm{R}'_{\boldsymbol{x}} \cdot \mathrm{R}_{\boldsymbol{x}}$. $\beta = 0.01$ was used in the experiments described in this paper.

4.2 Discriminative ridge regression in interactive machine translation

When attempting to apply DRR within an IMT setting, the quality metric that is used in IMT is no longer inherent to a single hypothesis, but to a complete wordgraph. It is quite common to measure the quality of a given IMT system by computing the amount of interactions required in order to modify the system's hypothesis so that it matches the reference. Once a single word has been introduced, the IMT system modifies the suffix, which implies that the number of interactions cannot be computed as a function of the hypothesis, but must be computed by first simulating the interaction procedure and is a function of a given wordgraph. Hence, DRR, as described in previous section, cannot be directly applied within an IMT framework. One would think that optimising a certain translation quality metric would also optimise the amount of interactions required. However, experimental results detailed in Sec. 5 show that this assumption is not completely true. Hence, since the metric to be optimised by online learning does not depend on a single-best hypothesis, the formulation of DRR needs to be reviewed.

At this stage, it would be reasonable to consider instead of a list of N-best hypothesis a list of N-best wordgraphs. However, the concept of N-best wordgraph is somewhat fuzzy. For this reason, instead of computing a true list of N-best wordgraphs we will obtain a set of N scaling factors $\boldsymbol{\lambda}$ obtained at random, $\boldsymbol{\Lambda} = \{\boldsymbol{\lambda}^1, \ldots, \boldsymbol{\lambda}^n, \ldots, \boldsymbol{\lambda}^N\}$, and compute the wordgraph $W_{\boldsymbol{\lambda}^n}(\boldsymbol{x})$ associated to a given input sentence $\boldsymbol{x}$ and obtained for a certain set of scaling factors $\boldsymbol{\lambda}^n$. Of course, since the weights have been obtained at random, the resulting wordgraphs will not constitute a true list of possible N-best wordgraphs. However, since the purpose of DRR is to reward those hypotheses (in this case wordgraphs) that score well, and penalise those that score worse, what is really important is to have wordgraphs (i.e., samples of $\boldsymbol{\lambda}$) which score well, and wordgraphs (samples of $\boldsymbol{\lambda}$) which score bad. Hence, $\mathbf{l}_{\boldsymbol{y}}$ will be a column vector of N rows such that

$$\mathbf{l}_{\boldsymbol{y}} = [l(W_{\boldsymbol{\lambda}^1}(\boldsymbol{x})) \ldots l(W_{\boldsymbol{\lambda}^n}(\boldsymbol{x})) \ldots l(W_{\boldsymbol{\lambda}^N}(\boldsymbol{x}))] \tag{17}$$

Another aspect that needs to be taken care of when considering DRR within an IMT setting is that matrix $\mathrm{H}_{\boldsymbol{x}}$ also needs to be redefined, since the features that need to be considered in this case no longer correspond to those of the hypotheses in the N-best list, but to the wordgraphs generated with $\boldsymbol{\Lambda}$. Since a certain wordgraph $W_{\boldsymbol{\lambda}^n}(\boldsymbol{x})$ does not have a single set of features, but rather one feature vector for each one of the paths through the wordgraph, we will consider for building $\mathrm{H}_{\boldsymbol{x}}$ the feature vector $\boldsymbol{h}$ of the best path in $W_{\boldsymbol{\lambda}^n}(\boldsymbol{x})$, i.e., the feature vector of the best hypothesis in $W_{\boldsymbol{\lambda}^n}(\boldsymbol{x})$. Abusing

[2] Also known as Tikhonov regularisation.

Table 1. Characteristics of the Europarl corpus and NC11 test set. OoV stands for "Out of Vocabulary" words, k stands for thousands of elements and M for millions of elements.

		Spanish	English
Europarl Training	Sentences	1.4M	
	Run. words	29.9M	28.9M
	Vocabulary	129.8k	85.3k
Europarl Development	Sentences	2000	
	Run. words	60.6k	58.7k
	OoV. words	164	99
NC11 test	Sentences	3003	
	Run. words	79.4k	74.7k
	OoV. words	1549	1708

notation and with the purpose of keeping notation unclogged, let $\boldsymbol{h}_{\boldsymbol{\lambda}^n}$ be such feature vector. Then, $\mathrm{H}_{\boldsymbol{x}}$ is defined for the IMT case as

$$\mathrm{H}_{\boldsymbol{x}} = [\boldsymbol{h}_{\boldsymbol{\lambda}^1}, \ldots, \boldsymbol{h}_{\boldsymbol{\lambda}^N}]' . \tag{18}$$

Equivalently, $\mathrm{H}^*_{\boldsymbol{x}}$ is defined in this case as

$$\mathrm{H}^*_{\boldsymbol{x}} = [\boldsymbol{h}_{\boldsymbol{\lambda}^*}, \ldots, \boldsymbol{h}_{\boldsymbol{\lambda}^*}] , \tag{19}$$

with $\boldsymbol{h}_{\boldsymbol{\lambda}^*}$ being the feature vector of the best hypothesis of wordgraph $W_{\boldsymbol{\lambda}^*}(\boldsymbol{x})$, and $W_{\boldsymbol{\lambda}^*}(\boldsymbol{x})$ being that wordgraph with the best performance according to the IMT metric used, from among those computed using the random set of weights $\boldsymbol{\Lambda}$.

5 Experimental results

Given that a true CAT scenario is very expensive, since it requires a human translator to correct every hypothesis, such scenario will be simulated by using the reference of the test set. Such reference will be fed one at a time, following an online setting.

Translation quality will be assessed by means of *Translation Edit Rate* (TER) [16] and *Word Stroke Ratio* (WSR) [4]. TER is an error metric that computes the minimum number of edits required to modify the system hypotheses so that they match the references. Possible edits include insertion, deletion, substitution of single words and shifts of word sequences. Hence, TER is an automatic metric which intends to measure the effort required to post-edit the hypotheses provided by a SMT system. WSR measures the amount of words (interactions in this case) a human translator would require to type within an IMT framework to correct the system's hypothesis. Both TER and WSR are measured in percentage, i.e., both are normalised by the total amount of words of the reference, multiplied by 100. Also in both cases, lower TER and WSR rates are better.

As baseline system, we trained a SMT system on the Europarl English–Spanish training data, in the partition of the Workshop on SMT of the EMNLP 2011 [7]. The Europarl corpus [17] (Table 1) is built from the transcription of European Parliament speeches published on the web. We used the open-source MT toolkit Moses [18][3] in its standard non-monotonic setup (including the `msd-reordering-fe` model [19]),

[3] Available from http://www.statmt.org/moses/

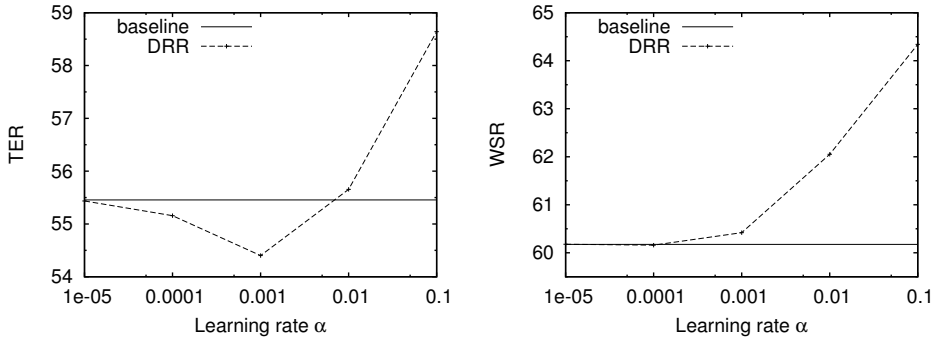

Fig. 2. Effect of the α learning rate on the effort within a PE and an IMT scenario, as measured by TER and WSR. DRR was implemented according to Sec. 4.1. N-best size was set to 1000.

and estimated $\boldsymbol{\lambda}$ using MERT [20] on the Europarl development set. The set of weights $\boldsymbol{\Lambda}$ described in Sec. 4.2 was obtained by sampling from a Gaussian distribution with mean vector the $\boldsymbol{\lambda}$ obtained by MERT and variance 0.01, following preliminary investigation. We also estimated a 5-gram LM with interpolation and Knesser-Ney smoothing [21]. Moses was also used for the purpose of building the required wordgraphs.

Since our purpose is to analyse the performance of an online adaptation strategy, in addition to Europarl we also considered a different test set that does not belong to the parliamentary domain, such as the News Commentary[4] (NC) 2011 test set. The News Commentary corpus was obtained from different news feeds and was used as test set for the 2011 EMNLP shared task on SMT [7]. See Table 1 for NC test set statistics.

As a first step, we carried out the experimentation according to Sec. 4.1, i.e., optimising a typical SMT evaluation metric which is ought to minimise post-editing effort. Such results can be seen in Fig. 2. The plot on the left displays TER, i.e., the amount of edits required in a PE scenario, whereas the plot on the right displays WSR, i.e., the amount of interactions required in an IMT setting. As shown, DRR achieves to provide improvements when the α learning rate is about 0.001 within the PE scenario, but fails to obtain the same results within the IMT setting. This is so because DRR, as described in Sec. 4.1, was implemented using TER as translation quality metric l. However, it would be quite risky to assume that minimising the number of edits within a PE setting would also lead to minimising the number of interactions within the IMT framework, and this fact is indeed reflected by the behaviour of WSR in the right plot of Fig. 2. It is important to point out that experiments using other translation quality metrics, such as BLEU [22], lead to similar results as the ones displayed here with TER.

After verifying that DRR, as described in Sec. 4.1, is not valid for its application in an IMT setting, we carried on implementing the version of DRR described in Sec. 4.2, and the results can be seen in Fig. 3. In this case, the approach proposed improves the amount of interactions required to correct a hypothesis, as measured by WSR. However, improvements obtained are not mirrored in the PE setting, where TER is only slightly improved for a very small α, and is actually higher with α values that do improve WSR.

Table 2 sums up the results above, with the purpose of providing more precision.

[4] This corpus is available from http://www.statmt.org/wmt11/

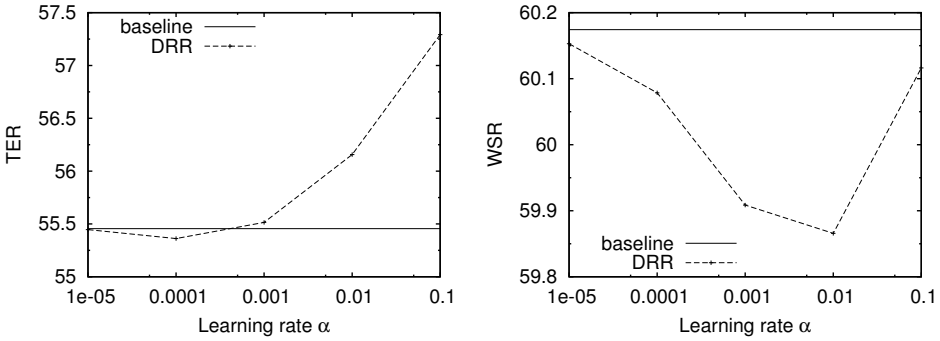

Fig. 3. Effect of α on the effort in PE and IMT, as measured by TER and WSR. DRR was implemented according to Sec. 4.2. The amount of random weights obtained was set to 500.

Table 2. TER and WSR scores for the two optimisation methods described in Sec. 4.

Optimisation method	α	TER	WSR
baseline	–	55.5	60.2
DRR (Sec. 4.1)	0.001	54.4	60.4
DRR (Sec. 4.2)	0.01	56.2	59.9

6 Conclusions and future work

In the present paper, we have analysed the applicability of discriminative Ridge regression within a simulated CAT environment. In the experiments reported, DRR was applied to update the log-linear weights of a state-of-the-art SMT system, both within a post-editing scenario and an interactive machine translation scenario. Results show that an implementation of DRR which optimises a traditional SMT evaluation metric and provides improvements within a PE scenario may fail to provide improvements in an IMT setting. Hence, a modification of DRR was carried out for its application in IMT, where the evaluation metric is not associated to a single hypothesis but to a complete wordgraph. Experiments with such modification present encouraging results.

As future work, we would like to study other possible ways of obtaining the set of random weights $\boldsymbol{\Lambda}$. An interesting possibility would be to obtain such weights by means of Markov chain Monte Carlo. In addition, the size of $\boldsymbol{\Lambda}$ might also be important, since the more weights sampled the higher the possibility of obtaining appropriate log-linear weights for a specific test sentence. We also intend to analyse this in future work.

Acknowledgements

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement Nr. 287576 (CasMaCat). Also funded by the EC (FEDER/FSE) and the Spanish MICINN under projects MIPRCV “Consolider Ingenio 2010” (CSD2007-00018) and iTrans2 (TIN2009-14511), and by the Generalitat Valenciana under grant Prometeo/2009/014.

References

1. Callison-Burch, C., Bannard, C., Schroeder, J.: Improving statistical translation through editing. In: Proc. of 9th EAMT workshop "Broadening horizons of machine translation and its applications". (April 26–27 2004) 26–32
2. Koehn, P.: Statistical Machine Translation. Cambridge University Press (2010)
3. Barrachina et al., S.: Statistical approaches to computer-assisted translation. Computational Linguistics **35**(1) (2009) 3–28
4. Toselli, A.H., Vidal, E., Casacuberta, F., eds.: Multimodal Interactive Pattern Recognition and Applications. Springer (2011)
5. Brown, P.F., Pietra, S.D., Pietra, V.J.D., Mercer, R.L.: The mathematics of statistical machine translation: Parameter estimation. Computational Linguistics **19**(2) (1994) 263–311
6. Och, F.J., Ney, H.: Discriminative training and maximum entropy models for statistical machine translation. In: Proc. of the 40th annual conf. of the ACL. (July 8–10 2002) 295–302
7. Callison-Burch, C., Koehn, P., Monz, C., Zaidan, O.F., eds.: Proceedings of the Sixth Workshop on SMT. Association for Computational Linguistics, Edinburgh, Scotland (July 2011)
8. Martínez-Gómez, P., Sanchis-Trilles, G., Casacuberta, F.: Online adaptation strategies for statistical machine translation in post-editing scenarios. Pattern Recognition **45**(9) (2012) 3193–3203
9. Christensen, H.: Speaker Adaptation of Hidden Markov Models using Maximum Likelihood Linear Regression. PhD thesis, Aalborg University, Denmark (1996)
10. Gauvain, J.L., Lee, C.H.: Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains. IEEE Transactions on Speech and Audio Processing **2** (1994) 291–298
11. Daumé III, H.: Frustratingly easy domain adaptation. In: Proc. of the 45th annual conf. of the ACL. (June 2007) 256–263
12. Sanchis-Trilles, G., Casacuberta, F.: Log-linear weight optimisation via bayesian adaptation in statistical machine translation. In: Proc. of the intl. conf. on Computational Linguistics, 2010. (August 23–27 2010) 1077–1085
13. Dempster, A., Laird, N., Rubin, D.: Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society **39**(1) (1977) 1–38
14. Ortiz-Martínez, D., García-Varea, I., Casacuberta, F.: Online learning for interactive statistical machine translation. In: Proc. of NAACL. (June 2–4 2010) 546–554
15. Och, F.: Minimum error rate training for statistical machine translation. In: Proc. of the 41st Annual Meeting of the ACL. (July 7–12 2003) 160–167
16. Snover, M., other: A study of translation edit rate with targeted human annotation. In: Proc. of the 7th biennial conf. of the AMTA. (August 8–12 2006) 223–231
17. Koehn, P.: Europarl: A parallel corpus for statistical machine translation. In: Proc. of the 10th Machine Translation Summit, 2005. (September 12–16 2005) 79–86
18. Koehn et al., P.: Moses: Open source toolkit for statistical machine translation. In: Proc. of the ACL Demo and Poster Sessions, 2007. (June 25–27 2007) 177–180
19. Koehn, P., Axelrod, A., Mayne, A.B., Callison-burch, C., Osborne, M., Talbot, D.: Edinburgh system description for the 2005 iwslt speech translation evaluation. In: Proc. of the international Workshop on SLT. (October 2005)
20. Och, F.J.: Minimum error rate training for statistical machine translation. In: Proc. of the 41st annual conf. of the ACL. (July 7–12 2003) 160–167
21. Kneser, R., Ney, H.: Improved backing-off for m-gram language modeling. In: Proc. of ICASSP. (May 9–12 1995) 181–184
22. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: A method for automatic evaluation of machine translation. In: Proc. of the 40th annual conf. of the ACL. (July 6–12 2002) 311–318