# Accuracy-Based Scoring for Phrase-Based Statistical Machine Translation

**Sergio Penkale**[†] **Yanjun Ma**[†] **Daniel Galron**[§] **Andy Way**[†]

[†]CNGL
School of Computing
Dublin City University
{spenkale,yma,away}@computing.dcu.ie

[§]CIMS
New York University
galron@cs.nyu.edu

## Abstract

Although the scoring features of state-of-the-art Phrase-Based Statistical Machine Translation (PB-SMT) models are weighted so as to optimise an objective function measuring translation quality, the estimation of the features themselves does not have any relation to such quality metrics. In this paper, we introduce a translation quality-based feature to PB-SMT in a bid to improve the translation quality of the system. Our feature is estimated by averaging the edit-distance between phrase pairs involved in the translation of oracle sentences, chosen by automatic evaluation metrics from the $N$-best outputs of a baseline system, and phrase pairs occurring in the $N$-best list. Using our method, we report a statistically significant 2.11% relative improvement in BLEU score for the WMT 2009 Spanish-to-English translation task. We also report that using our method we can achieve statistically significant improvements over the baseline using many other MT evaluation metrics, and a substantial increase in speed and reduction in memory use (due to a reduction in phrase-table size of 87%) while maintaining significant gains in translation quality.

## 1 Introduction

State-of-the-art Phrase-Based Statistical Machine Translation (PB-SMT) systems (Koehn et al., 2003; Och and Ney, 2002; Och, 2003) use sequences of words ("phrases") as the basic unit in translation. Given a source-language sentence, PB-SMT segments the sentence into phrases and searches for a translation that covers the input sentence while maximising the model score. This score is obtained by combining the score assigned to phrases by multiple model components, such as phrase translation and language model probabilities, which are induced in the training stage by the use of relative frequencies. Although the contribution of each component to the final score is weighted so as to optimise translation quality on held-out data via Minimum Error-Rate training (MERT) (Och, 2003), the individual components themselves only attempt to increase the likelihood of the training corpus, and none of them necessarily correlate with translation quality. Since the ultimate goal in training a PB-SMT system is to increase the quality of its translations when confronted with unseen data, in this paper we introduce a scoring method which we use to estimate a new feature, which relates to the expected translation quality of the system, and which we use to extend the model of a PB-SMT system.

There has been a range of research on the subject of translation quality-based scoring in MT. (Smith and Eisner, 2006) use minimum-risk training to improve on MERT in choosing the appropriate weights for a given set of features. (Liang et al., 2006), (Tillmann and Zhang, 2006) and (Arun and Koehn, 2007) describe methods to introduce a large number of binary features globally trained to increase BLEU (Papineni et al., 2002), although they do not report significant improvements over state-of-the-art PB-SMT systems trained with standard features. A known limitation of MERT is its difficulty to scale to weighting a larger amount of features than those present in a typical PB-SMT system (Och et al.,

2004). In another line of research, (Watanabe et al., 2007) and (Chiang et al., 2008) improve this by using the Margin Infused Relaxed Algorithm (MIRA) (Crammer et al., 2006) to estimate a large amount of syntactic and distortion features.

(Galron et al., 2009) showed that accuracy-based scoring is a crucial feature when incorporated into the Data-Oriented Translation paradigm (Poutsma, 2000; Hearne and Way, 2003). They introduce edit-distance measures to determine the similarity between candidate translation fragments and oracle translation fragments, and allow the system to benefit from knowledge of which fragments are typically involved in derivations of good translations. In this work we build upon this line of research to investigate the effects of accuracy-based scoring specifically for PB-SMT systems. We use a baseline PB-SMT system (Koehn et al., 2007) to obtain $N$-best lists, and then choose oracle translations according to a range of evaluation metrics. We then compare each phrase pair in the $N$-best list against phrases present in the oracle translations, and assign a score to each phrase pair according to how similar they are to those oracle phrase pairs. We use this information to incorporate a new feature, which indicates how likely a phrase pair is to contribute to good translations. Obtaining a score for each phrase pair out of the candidate translations for sentences in the training set is reminiscent of estimating phrase counts using forced alignments (Shen et al., 2008).

This approach differs from previous work such as that of (Watanabe et al., 2007) in that we do not attempt to replace the current features, but instead complement them by incorporating an additional feature (and rescoring existing reordering features) which brings translation quality-based knowledge into the scoring of phrase pairs. Unlike most previous work related to translation quality-driven scoring, this approach has the benefit of simplicity. This means that it can be easily performed using off-the-shelf decoders and tuning algorithms like MERT, and is therefore readily available to PB-SMT practitioners. In addition, the estimation of our feature is easily parallelizable, as sentences are processed independently of each other. Our experiments show that our approach leads not only to translation quality improvements, but also to improvements in translation speed and memory consumption.

This work represents a significant improvement over that of (Galron et al., 2009), in that unlike theirs, ours does not require large amounts of additional held-out data, as we estimate the new features using only the parallel data used to train the baseline system. Furthermore, unlike (Galron et al., 2009) we evaluate the impact of different evaluation metrics when selecting oracles (namely BLEU and the F-Score). In addition, we present methods to rescore each model component and provide an in-depth analysis of how and why this scoring process works.

The remainder of the paper is organised as follows. Section 2 gives a brief introduction to state-of-the-art PB-SMT and motivates our scoring method, which Section 3 describes. In section 4, our experimental setup is presented and section 5 shows the experimental results and corresponding analysis. We conclude and point out avenues for future research in section 6.

## 2 Log-linear Phrase-Based SMT

In PB-SMT (Koehn et al., 2003), an input sentence $\mathbf{f} = w_1 \dots w_n$ composed of $n$ words is segmented into $I$ phrases $f_1^I$. Each source phrase $f_i$ in $f_1^I$ is translated into a corresponding target phrase $e_i$, resulting in a target sentence $\mathbf{e} = e_1^I$ and an alignment $a$. For each target phrase $e_i$, this alignment specifies a pair of integers $a(e_i) = (l, m)$, indicating that the phrase $e_i$ is translated from the source sentence span $w_l \dots w_m$. Target phrases might be reordered.

To select among the many phrase translation options and possible input segmentations, we choose the target sentence $\mathbf{e}$ that maximises $P(\mathbf{e}|\mathbf{f})$, which is modelled directly by a log-linear model (Och and Ney, 2002) as in (1):

$$P(e_1^I|f_1^I) = \exp(\sum_{i=1}^{M} \lambda_i h_i(e_1^I, f_1^I)) \qquad (1)$$

Here each $h_i(e_1^I, f_1^I)$ is a feature function and each $\lambda_i$ the corresponding feature weight. Typical features include an $n$-gram language model over the target translations, and the product of the conditional phrase translation probabilities $p(f_i|e_i)$ and $p(e_i|f_i)$. These probabilities are estimated using relative frequency over the multiset of phrases extracted from the parallel corpus, and are smoothed by "lexical

weighting" features which measure how often were words in a phrase pair aligned in the parallel corpus.

State-of-the-art PB-SMT also incorporates lexicalised reordering features (Koehn et al., 2005), which assign a probability to the orientation between a phrase and the previously translated phrase. The modelled orientations are: monotone (a phrase directly follows the previous phrase), swap (a phrase is swapped with the previous phrase) and discontinuous (neither monotone nor swap). Analogous orientations are computed by considering the next phrase.

Although the weights $\lambda_i$ in (1) are normally estimated by MERT (Och, 2003) to optimise translation quality metrics such as BLEU (Papineni et al., 2002), the estimation of the features themselves does not have any correlation with these translation quality metrics, given that these probabilities are directly estimated by relative frequency over the parallel corpus. This motivates our work on accuracy-based scoring, which is aimed at incorporating additional features to the PB-SMT model which relate to the contribution that each phrase pair typically brings to the quality of the output translation.

## 3 Accuracy-Based Scoring

As mentioned above, we attempt to incorporate a notion of translation quality to the scoring of phrases. We will assume that translation quality is measured by a function $\mathbf{E}(\mathbf{f}, \tilde{\mathbf{e}}, \vec{\mathbf{e}})$ which assigns a score between 0 and 1 to a translation $\tilde{\mathbf{e}}$, measuring how "good" it is as a translation of the sentence $\mathbf{f}$, taking the sentences in the vector $\vec{\mathbf{e}}$ as references.

Given an input sentence $\mathbf{f}$, let $T_{\mathbf{f}}$ be the set of target translations that our system is able to produce using its extracted phrase pairs. Our ultimate goal is to score phrases in such a way that the system output is the sentence $\hat{\mathbf{e}}$ such that:

$$\hat{\mathbf{e}} = \underset{\tilde{\mathbf{e}} \in T_{\mathbf{f}}}{\operatorname{argmax}} \mathbf{E}(\mathbf{f}, \tilde{\mathbf{e}}, \vec{\mathbf{e}}) \quad (2)$$

Accomplishing this is extremely difficult, since the quality $\mathbf{E}$ of a translation depends not only on the quality of each individual phrase pair involved in it, but on the entire series of phrases it is composed of. This means that while a phrase pair can be completely adequate as the translation of some source words in one sentence, the same phrase pair translating the same source segment can be inadequate when taking a different source context into account.

Since in PB-SMT (with the exception of language model scores) phrase pairs are scored independently of each other, the approach that we take is that we attempt to differentiate between phrases that *on average* lead to good translations from phrases that typically do not. We encode this information as a function $Acc(f_i, e_i)$ that quantifies how similar a phrase pair $(f_i, e_i)$ is on average to a phrase pair involved in an oracle translation. Using this function we incorporate a new feature $h_{Acc}$ into the log-linear model of Equation (1), as in (3):

$$h_{Acc}(e_1^I, f_1^I) = \prod_{i=1}^{I} Acc(f_i, e_i) \quad (3)$$

We estimate the function $Acc$ using a held-out parallel corpus (cf. Section 3.4). For each sentence in this corpus we use a baseline system to obtain an $N$-best list, from which we determine the sentence that maximises our translation quality metric $\mathbf{E}$. Details of this procedure are explained in Section 3.1. We refer to the hypothesised translations in the $N$-best list as the *candidate translations* $\mathcal{C}$, and to the sentences which maximise $\mathbf{E}$ over this list as the *oracle translations* $\mathcal{O}$. We consider the phrase pairs involved in building each of the candidate translations, and use the metrics that we define in Section 3.2 to compare how similar these phrase pairs are vis-à-vis those used to build the oracle translations. From this comparison we obtain a similarity score. We repeat this process for all sentences on the held-out data, and assign to each phrase pair $(f_i, e_i)$ a score $Acc(f_i, e_i)$ equal to the average of the similarity scores it obtained. Note that a phrase pair in the baseline phrase-table needs to be observed in at least one $N$-best list in order to receive a score. Section 4.1 explains how this can be dealt with.

### 3.1 Oracle selection

Since it is infeasible to search for oracles over the entire set $T_{\mathbf{f}}$, we approximate this by only considering the $N$-best translations $T_N$ produced by our system prior to the incorporation of our new feature, and we limit ourselves to scoring phrase pairs appearing only in this $N$-best list.

Let $T_N = \{\tilde{\mathbf{e}}_1, \ldots, \tilde{\mathbf{e}}_N\}$ be the $N$-best translations where each translation $\tilde{\mathbf{e}}_i$ is associated with

an alignment $a_i$ indicating the mapping between the source sentence $\mathbf{f}$ and the translation $\tilde{\mathbf{e}}_i$ at phrase level. We choose as the oracle translations the candidate translations that maximise the translation quality function $\mathbf{E}$, in a similar way to what (Liang et al., 2006) call *local updating*. Noting that many target translations may receive the same highest score $\mathbf{E}$, we define the set of oracles $\mathcal{O}$ as in (4):

$$\mathcal{O} = \underset{\tilde{\mathbf{e}} \in T_N}{\operatorname{argmax}} \mathbf{E}(\mathbf{f}, \tilde{\mathbf{e}}, \vec{\mathbf{e}}) \quad (4)$$

We assume that we can recover the phrase alignment $a$ used in each candidate translation, a feature most decoders provide. Using this, a function $\Omega$ indicating the mapping (as defined by $a$) between a source sentence span $(l, m)$ and the corresponding set of target phrases in oracle translations $\mathcal{O}$ (*oracle phrases*) is defined in (5):

$$\Omega(l, m) = \{\tilde{e}_o | \exists\, \tilde{\mathbf{e}} \in \mathcal{O} : \tilde{e}_o \in \tilde{\mathbf{e}} \wedge a(\tilde{e}_o) = (l, m)\} \quad (5)$$

We experiment with two translation-quality estimators for $\mathbf{E}$, namely BLEU and the F-Score, which the following two subsections describe. In the future we will consider evaluating the effects of additional evaluation metrics, such as TER (Snover et al., 2006).

#### 3.1.1 BLEU

The BLEU score (Papineni et al., 2002) computes a geometric mean of the unigram to $N$-gram precisions between a candidate sentence and a set of references (typically $N = 4$). If there is not at least one $N$-gram match, the BLEU score is 0. Since our aim is to use BLEU not at the document level where this phenomenon would be rare, but at the sentence level in Equation (4), this is problematic because in practice BLEU will be 0 for most sentences. We thus follow (Liang et al., 2006) and approximate BLEU by a smoothed version that combines the scores of BLEU for various $N$, as in (6):

$$\text{sBLEU} = \sum_{i=1}^{N} \frac{\text{BLEU}_i}{2^{4-i+1}} \quad (6)$$

Note that the direct use of document-level approximations of BLEU such as those used in (Watanabe et al., 2007) would be impractical in our approach, as it would introduce dependencies across sentences which would limit parallelisation.

#### 3.1.2 F-Score

The General Text Matcher (GTM) (Turian et al., 2003) computes the F-Score between a candidate translation and a reference using the notions of precision and recall. This computation is parameterised by an exponent, which adjusts the weights of longer $n$-grams in the score. In this work we use the F-Score with an exponent of 1.5, which was estimated by evaluating the quality of the oracles obtained on held-out data.

### 3.2 Similarity Metrics

To estimate the function $Acc$ in (3), we need a notion of similarity between the target phrases present in a candidate translation $\tilde{e}_c$ and the ones present in an oracle translation $\tilde{e}_o$. We relate target phrases in candidate translations to phrases in oracle translations by considering the source-sentence span they translate. To achieve this, the mapping $a$ between the source-sentence span and the target phrases as determined by the decoder is required. The estimation of $Acc$ will be limited to those phrases in a candidate translation for which oracle phrases exist which translate the same source span, i.e. we only score target phrases for which $\Omega(a(\tilde{e}_c)) \neq \emptyset$.

#### 3.2.1 Edit distance scoring

To compare two phrase pairs with the same source side and different target translations, we use the (word-level) Levenshtein distance $\delta_{dl}(\tilde{e}_c, \tilde{e}_o)$ (Damerau, 1964) between the target side of the phrase pairs. This measures the amount of insertions, deletions, or substitutions of words needed to transform the candidate phrase into the oracle phrase. For a phrase $\tilde{e}_c$ (in the candidate translation) which is translated from a source span $a(\tilde{e}_c)$, we assign as a score the exponential of the negative edit distance between $\tilde{e}_c$ and the oracle phrase $\tilde{e}_o$ it is most similar to, as in (7):

$$Acc_{ed}(f_i, \tilde{e}_c) = \max_{\tilde{e}_o \in \Omega(a(\tilde{e}_c))} \exp(-\delta_{dl}(\tilde{e}_c, \tilde{e}_o)) \quad (7)$$

Note that after repeating this for all sentences in the held-out set, the score assigned to a phrase pair is the average of the scores it obtained.

#### 3.2.2 Normalised Edit Distance

A potential problem with the metric in (7) is that we would expect that on average the edit-distance

between a candidate phrase $\tilde{e}_c$ and an oracle phrase $\tilde{e}_o$ would grow with phrase length. Since this could introduce and unwanted bias against long phrases, we also experiment with a score that normalises the edit distance by the amount of words in the target phrase, as in (8):

$$Acc_{norm}(f_i, \tilde{e}_c) = \max_{\tilde{e}_o \in \Omega(a(\tilde{e}_c))} 1 - \frac{\delta_{dl}(\tilde{e}_c, \tilde{e}_o)}{\max(|\tilde{e}_c|, |\tilde{e}_o|)} \quad (8)$$

### 3.3 Reordering Model

We also re-estimate the lexicalised reordering model by considering the order between phrases involved in oracle translations. For each phrase pair involved in an oracle translation, we obtain the orientation by considering both the previous and next phrases as in (Koehn et al., 2005). We thus obtain a list of triples $(f_i, e_i, o)$, where $(f_i, e_i)$ is a phrase pair and $o \in \{\text{monotone}, \text{swap}, \text{discontinuous}\}$, which we use to estimate $p_{Acc}(o|f_i, e_i)$.

To incorporate this information into the model, phrases for which we did not extract orientation information are assigned a default score equal to the median score for a particular orientation of the scored phrases. Then, for some constant $q$, we interpolate this new reordering score with the original score $p(o|f_i, e_i)$, as in (9):

$$p_r(o|f_i, e_i) = q \cdot p(o|f_i, e_i) + (1 - q) \cdot p_{Acc}(o|f_i, e_i) \quad (9)$$

### 3.4 Estimation Corpus

Unlike (Galron et al., 2009), we do not make use of additional training data to estimate the accuracy-based feature, but instead use Deleted Estimation (Jelinek and Mercer, 1985), a technique that has successfully been used in Data-Oriented Parsing (Zollmann and Sima'an, 2005) and a wide range of machine learning approaches such as decision tree induction (Breiman et al., 1984). In a way similar to 10-fold cross validation, we create a new training corpus $T$ by keeping 90% of the sentences in the original training corpus, and a new estimation corpus $H$ by using the remaining 10% of the sentences. Using this scheme we make 10 different pairs of corpora $(T_i, H_i)$ in a way such that each sentence from the original training corpus is in exactly one $H_i$ for some $1 \leq i \leq 10$, which ensures that each sentence is observed during estimation. We train 10 different systems using each $T_i$, and use each system to estimate $Acc$ on its corresponding held-out set $H_i$. We then consider all of the scores obtained by each phrase pair in any $H_i$, and assign as final estimate to each phrase pair the average of those scores (Jelinek and Mercer, 1985). The new feature is then added to the baseline system, which was trained on the whole original training set.

## 4 Experimental Setup

We empirically evaluate the impact of our new feature by performing Spanish-to-English translation and comparing against a baseline system trained using standard parameters. In all of our experiments we use the Moses toolkit (Koehn et al., 2007). We train on the training section of the Spanish–English Europarl corpus as provided for the Fourth Workshop on Statistical Machine Translation (WMT09).[1] We discarded sentences with more than 40 words, which left us with 1,083,773 sentences for training. We use the first 500 sentences of dev2006 as a tuning set for MERT. We use test2006 as a development test set, and test2008 as the final test set (each containing 2,000 sentence pairs). We use the 5,000-best translations returned by our decoder to select oracles and perform the scoring (note that (Galron et al., 2009) report obtaining oracles from the 10,000-best parse trees of the input sentence). We report our results using a range of evaluation metrics, namely BLEU (Papineni et al., 2002), NIST (Doddington, 2002), Meteor (Banerjee and Lavie, 2005) and the F-Score (Turian et al., 2003). In our discussion, absolute scores for BLEU, Meteor and the F-score are reported as percentages.

### 4.1 Dealing with unestimated phrase pairs

As mentioned in Section 3.4, each sentence will appear in one held-out set. Even though this ensures that all of the training sentences will be decoded in the estimation process, this does not guarantee that every phrase pair will receive a score according to $Acc$, as a phrase pair needs to occur in an $N$-best list translating the same source span as an oracle phrase pair in order to receive a score. In fact, out of the 46,994,471 phrase pairs in the baseline

[1] http://statmt.org/wmt09/

phrase table, only 6,056,274 of them can obtain an accuracy-based score, when using the F-Score to select oracles, and 5,994,142 when using sBLEU. We experiment with two ways of dealing with unscored phrases. Firstly, we follow (Galron et al., 2009) and calculate a default score equal to the median score among phrase pairs that receive a score, and assign this score as the $Acc$ estimation for phrase pairs for which no accuracy-based score was obtained. Secondly, we build a system which uses only the phrase pairs that receive some score, namely just 13% of the phrase table in the baseline system.

## 5 Experimental Results

In this section, we evaluate the effect of using different metrics for phrase scoring, the impact of rescoring the reordering-model, and the effects of using different oracle selection metrics. We also discuss issues related to speed and memory use. Statistical significance tests were performed using paired bootstrap resampling (Koehn, 2004). While developing our system, we repeatedly tested our incremental improvements on a development test set (reported as scores between squared brackets), and then performed our evaluation on the test set to obtain our final results. Results with single underlines are statistically significantly better than the baseline at $p = 0.05$ and those with double underlines are significantly better at $p = 0.01$. Unless specifically mentioned, the oracle selection metric in our experiments is the F-Score.

### 5.1 Accuracy-Based Feature and Reordering Table Rescoring

We used the methods described in Section 3 to assign new scores to phrase pairs and to rescore the reordering model. Our Accuracy-Based (AB) feature encoding the average similarity between a phrase in a candidate translation and one in an oracle translation was calculated using two metrics, namely the edit-distance ("ed") described in Section 3.2.1, and the normalised edit-distance ("norm") of Section 3.2.2.

To single out the contribution of phrase rescoring, we first conducted experiments with an Accuracy-Based feature and a default reordering table, assigning a default score to unscored phrase pairs. The

| System | BLEU% | NIST | M% | $F_1$% |
|---|---|---|---|---|
| Baseline | 32.72 [32.29] | 7.7941 | 56.55 | 64.88 |
| AB feature + default reordering (F-Score Oracles) | | | | |
| ed | 32.64 [32.41] | 7.7962 | 56.55 | 65.04 |
| norm | 33.16 [32.76] | 7.8449 | 56.97 | 65.30 |
| AB feature + rescored reordering (F-Score Oracles) | | | | |
| ed | 33.03 [32.71] | 7.8582 | 56.77 | 65.18 |
| norm | **33.41** [**32.83**] | **7.8879** | **57.14** | **65.43** |
| AB feature + rescored reordering (sBLEU Oracles) | | | | |
| ed | 33.11 [32.56] | 7.8379 | 56.97 | 65.30 |
| norm | 33.11 [32.65] | 7.8513 | 56.91 | 65.28 |

Table 1: System performance with Accuracy-Based (AB) features and default score for unscored phrases. "ed" and "norm" stand for the metrics of Sections 3.2.1 and 3.2.2 respectively. M stands for Meteor and $F_1$ for F-Score

first four rows in Table 1 show the performance of the baseline system (without our AB feature) and the system with the baseline features and the addition of our new feature. The effect of using two different similarity metrics (ed v.s. norm) is also presented. As expected, the normalised edit-distance metric ($4^{\text{th}}$ row in Table 1) yields higher translation quality compared to the (absolute) edit-distance. While using the "ed" metric cannot produce significantly better translations, using "norm" leads to statistically significant gains over the baseline across all evaluation metrics we used. Using this setup, there is a 0.44 absolute improvement in BLEU, corresponding to a 1.34% relative improvement. This contrasts with the results reported in (Galron et al., 2009), where their normalised edit-distance underperforms when compared to the absolute edit-distance. We believe this might be a result of our use of MERT, which can determine a weight for our feature that properly scales it to the magnitudes of the other model components, while they only report experiments where the weight of their feature is arbitrarily assigned on a manual basis.

To investigate the build-up effect of using both the Accuracy-Based feature and a rescored reordering model, we estimated a new reordering model using $q = 0.5$ in Equation (9). Clearly we see an add-on value by doing this, as gains are observed across all evaluation metrics. The best system, i.e. using both an AB feature and a rescored reordering model with "norm" as the similarity metric, outperforms

| Feature | Baseline | Def. Reo. | Resc. Reo. |
|---|---|---|---|
| $\text{Acc}_{norm}$ | - | 0.4513 | 0.4707 |
| Lang. Model | 0.2512 | 0.1756 | 0.1895 |
| $p(f \mid e)$ | 0.1600 | 0.0818 | 0.0952 |
| $lex(f \mid e)$ | 0.1498 | 0.0939 | 0.1252 |
| $p(e \mid f)$ | 0.1166 | 0.0274 | 0.0822 |
| $lex(e \mid f)$ | 0.0064 | 0.0309 | 0.0118 |
| Phrase Penalty | 0.3687 | 0.2732 | 0.1405 |
| Word Penalty | -0.0530 | -0.1344 | -0.1152 |

Table 2: Weight assigned by MERT to each (non-reordering) feature in the models of the baseline system, the system with our AB feature + default reordering, and the system with the AB feature + rescored reordering

the baseline by 0.69 BLEU points, corresponding to a 2.11% relative improvement. We note also that the 0.25 absolute BLEU points improvement between the system with rescored reordering (7th row in Table 1) and the system with default reordering (4th row) is statistically significant at $p = 0.01$.

We give in Table 2 the normalised weights assigned by MERT to each of the (non-reordering) features, for the baseline system and for the systems with an Accuracy-Based feature estimated using the normalised edit distance. We see that after adding the AB feature, most of the original features have ceded their contribution to the overall scoring, which is now dominated by this feature. This shows that the translation quality improvements observed in Table 1 are due to the introduction of our feature. Interestingly, while 30% of the contribution of the language model feature to the overall score has been given away to our new feature, the phrase translation probabilities on each direction cede 48% and 76% of their weight. The greater loss in weight for the source-to-target direction is expected, as this is the direction our technique assumes. An experiment using a system with our accuracy-based feature and no phrase translation or lexical weighting probabilities (i.e. using also phrase and word penalties, features which are not estimated during training) results in a BLEU score of 31.26 when decoding our development test set. While this is a significant drop of 1.03 BLEU points, we nonetheless find it remarkable that the system is able to perform at a level not so distant from the baseline while ignoring relevant features such as phrase translation probabilities, especially considering that these features have been considered integral components in both word- and phrase-based SMT since their inception.

### 5.2 Oracle Selection Metric

The last two rows in Table 1 show the effect of using sBLEU instead of the F-Score to select oracles. As can be seen, the systems with F-Score oracle selection consistently outperform those using sBLEU, where the best system using the F-Score has a gain of 0.3 absolute BLEU points over using sBLEU, corresponding to a 0.9% relative improvement. While this gain appears to be modest, it is statistically significant at $p = 0.01$, demonstrating the advantage of using the F-Score over sBLEU for oracle selection. This is not surprising given that, as noted in Section 3.1.1, BLEU is specifically designed for document-level evaluation while the F-Score is more suitable for evaluation at sentence-level.

We collected oracle selection statistics in order to further investigate this process. It turned out that 92.24% of the oracles are not the top hypothesis in the N-best list. In fact, if we evaluate the score obtained by using the 1-best hypotheses when decoding the training set to obtain the N-best lists, we obtain a BLEU score of 40.46, while using the oracle translations obtained by the F-Score yields 52.86 in BLEU. The corresponding score for sBLEU oracle selection is a BLEU score of 53.39. Given that the top hypothesis in the N-best list is the most likely translation according to the current model parameters, it is clear that there is plenty of space for improving the model to allow for the best translation in the N-best list (the oracle) to be scored the highest. This confirms the rationale of our methods which can improve the model parameterisation and subsequently the translation results.

In order to show the rank of the oracle in the N-best lists, we plotted the frequency distribution of the ranks as shown in Figure 1. We can see that oracles are very frequently selected from the top 100 hypotheses of the N-best list. Hypotheses with a rank above 1000 may still be selected as the oracle, but with a much lower frequency (corresponding to the dense tail on the right of the graph). As a matter of fact, 7.76% of the oracle translations are the 1-best hypothesis (corresponding to the point at the top left corner of the graph), 11.41% are selected from the top-10 hypotheses, 19.88% from the top-

| System | BLEU% | NIST | M% | $F_1$% |
|---|---|---|---|---|
| Baseline | 32.72 [32.29] | 7.7941 | 56.55 | 64.88 |
| AB feature + rescored reordering (F-Score Oracles) | | | | |
| ed | 33.04 [32.66] | 7.8665 | 56.93 | 65.29 |
| norm | 33.23 [32.60] | 7.8835 | **57.21** | **65.58** |
| AB feature + rescored reordering (sBLEU Oracles) | | | | |
| ed | **33.35** [32.56] | 7.8590 | 57.20 | 65.42 |
| norm | 33.25 [**33.19**] | **7.9236** | 57.06 | 65.54 |

Table 3: System performance using only scored phrases

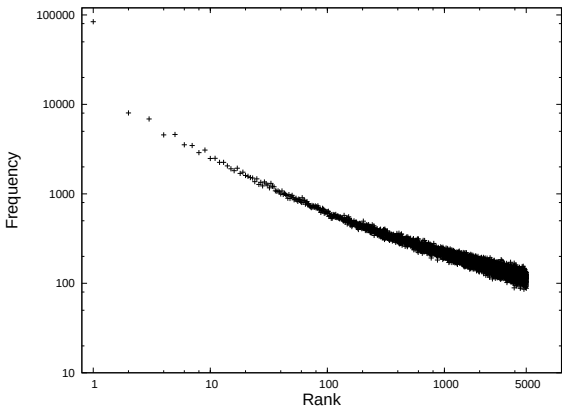


Figure 1: Oracle rank frequencies (log scale)

100 hypotheses, 45.34% from the top-1000 hypotheses, and the remaining 54.66% are selected from hypotheses ranking from 1000 to 5000. It is clear that a large N-best list is crucial in order to select a better oracle translation.

### 5.3 Improved Speed and Memory Use

We also conducted experiments discarding phrases that were not scored. From Table 3, we can see that using only those phrases that received a score yields improvements over the baseline across all evaluation metrics. There is an improvement of 0.63 absolute BLEU points over the baseline using sBLEU for oracle selection and "ed" as the similarity metric, corresponding to a 1.93% relative improvement over the baseline. We also observe a modest gain over using all phrases (Table 1) across most of the metrics (except for the BLEU score of the system using the F-Score for oracle selection and "norm" as similarity measure). This is remarkable given that the system with the AB feature uses a phrase table 87% smaller than the one in the baseline, which leads to speed increases and memory consumption reductions.

### 5.4 Sentence-level evaluation

In addition to the document-level automatic evaluation, we conducted a sentence-level evaluation using Meteor. Pairwise comparison was performed for four systems including the Baseline system (B), the system with AB Feature and default reordering (AB), AB Feature and rescored reordering (AB+O), and system only using the rescored phrases (R). In the pairwise comparison, we count the number of sentences in the test set where one of the systems is better or both systems are equal.

The results in Table 4 are consistent with the document-level evaluation in Tables 1 and 3. We see that with AB-Scoring (B vs. AB), sentences with improved translations are far more numerous than those whose translations become worse (703 vs. 558). Adding both the AB feature and a rescored reordering model (B vs. AB+O and AB vs. AB+O) further improves the system performance, with more sentences receiving better translations. Using only rescored phrases yields substantial improvements over the baseline (B vs. R), and encouragingly, using only the rescored phrases does not result in any decrease in translation quality; there is instead a marginal gain of 3 sentences (AB+O vs. R).

In order to qualitatively assess some of the improvements to which our method leads, we give in Figure 2 example output from the baseline system and from our best-scoring system, when translating a sentence in our development test set. In this Figure, vertical bars represent the target-side phrase-segmentation used to build the sentence, and numbers above the phrases indicate the amount of times that the corresponding phrase pair was extracted from the training corpus. We see that the baseline system uses short phrases which are very frequent in the training corpus. In contrast, the rescored system uses fewer, longer phrases, which do not occur as frequently in the training corpus, but which yield a better translation. This might be explained by the weights given in Table 2: the rescored system does not have to heavily rely on the amount of times a phrase pair occurs in the corpus, allowing it to use longer (and more infrequent) phrases when evidence has been observed that such a phrase typically leads to a good translation. We note that while the base-

| | B vs. AB | B vs. AB+O | B vs. R | AB vs. AB+O | AB vs. R | AB+O vs. R |
|---|---|---|---|---|---|---|
| System 1 Better | 558 | 515 | 633 | 482 | 616 | 660 |
| Equal | 739 | 793 | 555 | 1006 | 712 | 677 |
| System 2 Better | 703 | 692 | 812 | 512 | 672 | 663 |

Table 4: Pairwise comparison of different systems via sentence-level evaluation

**Source:** la comunidad internacional no puede contentarse por más tiempo con esconder la cabeza como el avestruz (...)
**Reference:** the international community can no longer content itself with burying its head in the sand (...)

**Baseline:** the$^{523092}$ | international community$^{1385}$ | cannot be satisfied$^{2}$ | by$^{31382}$ | more time$^{233}$ | to$^{9965}$ | bury our$^{8}$ | heads in the sand$^{3}$ | (...)
**Rescored:** the international community$^{1}$ can no | longer$^{7}$ | be content$^{8}$ | with$^{72416}$ | burying one 's head in the sand$^{1}$ | (...)

Figure 2: Example output translation from the baseline and our best-scoring system

line system uses $32,728$ phrase pairs to translate the whole development set, the rescored system can translate it using only $26,272$ phrase pairs, an indication that this phenomenon might not be unique to this particular sentence.

## 6 Conclusions

In this paper, we introduced additional features for PB-SMT which bring translation quality-related knowledge into the scoring of phrase pairs. On the WMT09 Spanish-to-English translation task, significant gains over the baseline are obtained across many evaluation metrics. Encouragingly, our method can also lead to a substantial reduction (87%) in phrase-table size without significant loss in translation quality. Given the size of the weight associated with our AB feature (Table 2), it is not an exaggeration to conclude that gains in MT quality and efficiencies in speed and memory usage are due almost entirely to our new feature. Although discarding key features such as phrase-translation probabilities and using only our feature leads to a loss in translation quality, doing so results in a system that performs nearly as well as the baseline.

In the future, we plan to adapt this method to other types of SMT systems, and to evaluate its performance on additional language pairs. We are also interested in performing manual evaluation to enhance the analysis in Section 5.4 by determining which type of sentences benefit from this approach and which do not. We will also investigate the trade-off between $N$-best list size and the corresponding amount of rescored phrase pairs and translation quality improvement, and we will consider investigating the use of word lattices for oracle selection and phrase scoring.

## Acknowledgments

This research is supported by the Science Foundation Ireland (Grant 07/CE/I1142) as part of the Centre for Next Generation Localisation (www.cngl.ie) at DCU. We thank the anonymous reviewers for their helpful comments and suggestions.

## References

Abhishek Arun and Philipp Koehn. 2007. Online learning methods for discriminative training of phrase based statistical machine translation. In *Proceedings of Machine Translation Summit XI*, pages 15–20, Copenhagen, Denmark.

Satanjeev Banerjee and Alon Lavie. 2005. 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, MI.

Leo Breiman, Jerome Friedman, Charles J. Stone, and R. A. Olshen. 1984. *Classification and Regression Trees*. Chapman and Hall/CRC, 1 edition.

David Chiang, Yuval Marton, and Philip Resnik. 2008. Online large-margin training of syntactic and structural translation features. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 224–233, Honolulu, HI.

Koby Crammer, Ofer Dekel, Joseph Keshet, Shai Shalev-Shwartz, and Yoram Singer. 2006. Online passive-aggressive algorithms. *Journal of Machine Learning Research*, 7:551–585.

Fred J. Damerau. 1964. A technique for computer detection and correction of spelling errors. *Commun. ACM*, 7(3):171–176.

George Doddington. 2002. 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 Francisco, CA.

Daniel Galron, Sergio Penkale, Andy Way, and I. Dan Melamed. 2009. Accuracy-Based Scoring for DOT: Towards Direct Error Minimization for Data-Oriented Translation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 371–380, Singapore.

Mary Hearne and Andy Way. 2003. Seeing the wood for the trees: Data-oriented translation. In *Proceedings of the Ninth Machine Translation Summit*, pages 165–172, New Orleans, LA.

Fred Jelinek and Robert Mercer. 1985. Probability distribution estimation from sparse data. *IBM Technical Disclosure Bulletin*, 28:2591–2594.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the ACL*, pages 48–52, Edmonton, Canada.

Philipp Koehn, Amittai Axelrod, Alexandra Birch Mayne, Chris Callison-Burch, Miles Osborne, and David Talbot. 2005. Edinburgh system description for the 2005 IWSLT speech translation evaluation. In *Proceedings of IWSLT*, Pittsburgh, PA.

P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the Annual Meeting of the ACL, demonstation session*, pages 177–180, Prague, Czech Republic.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 388–395, Barcelona, Spain.

Percy Liang, Alexandre Bouchard-Côté, Dan Klein, and Ben Taskar. 2006. An end-to-end discriminative approach to machine translation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th annual meeting of the ACL*, pages 761–768, Sydney, Australia.

Franz Josef Och and Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 295–302, Philadelphia, PA.

Franz Josef Och, Daniel Gildea, Sanjeev Khudanpur, Anoop Sarkar, Kenji Yamada, Alex Fraser, Shankar Kumar, Libin Shen, David Smith, Katherine Eng, Viren Jain, Zhen Jin, and Dragomir Radev. 2004. A smorgasbord of features for statistical machine translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 161–168, Boston, MA.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *41st Annual Meeting of the ACL*, pages 160–167, Sapporo, Japan.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, PA.

Arjen Poutsma. 2000. Data-oriented translation. In *The 18th International Conference on Computational Linguistics*, pages 635–641, Saarbrücken, Germany.

Wade Shen, Brian Delaney, Timothy Anderson, and Raymond Slyh. 2008. The MIT-LL/AFRL IWSLT-2008 MT System. In *Proceedings of the International Workshop on Spoken Language Translation*, pages 69–76, Hawaii, USA.

David A. Smith and Jason Eisner. 2006. Minimum risk annealing for training log-linear models. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 787–794, Sydney, Australia.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. 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, MA.

Christoph Tillmann and Tong Zhang. 2006. A discriminative global training algorithm for statistical MT. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th annual meeting of the ACL*, pages 721–728, Sydney, Australia.

Joseph P. Turian, Luke Shen, and I. Dan Melamed. 2003. Evaluation of machine translation and its evaluation. In *Proceedings of the Ninth Machine Translation Summit*, pages 386–393, New Orleans, LA.

Taro Watanabe, Jun Suzuki, Hajime Tsukada, and Hideki Isozaki. 2007. Online large-margin training for statistical machine translation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, Prague, Czech Republic.

Andreas Zollmann and Khalil Sima'an. 2005. A consistent and efficient estimator for data-oriented parsing. *Journal of Automata, Languages and Combinatorics*, 10(2/3):367–388.