Better Evaluation Metrics Lead to Better Machine Translation

Chang Liu[1] and **Daniel Dahlmeier**[2] and **Hwee Tou Ng**[1,2]
[1]Department of Computer Science, National University of Singapore
[2]NUS Graduate School for Integrative Sciences and Engineering
{liuchan1,danielhe,nght}@comp.nus.edu.sg

Abstract

Many machine translation evaluation metrics have been proposed after the seminal BLEU metric, and many among them have been found to consistently outperform BLEU, demonstrated by their better correlations with human judgment. It has long been the hope that by tuning machine translation systems against these new generation metrics, advances in automatic machine translation evaluation can lead directly to advances in automatic machine translation. However, to date there has been no unambiguous report that these new metrics can improve a state-of-the-art machine translation system over its BLEU-tuned baseline.

In this paper, we demonstrate that tuning Joshua, a hierarchical phrase-based statistical machine translation system, with the TESLA metrics results in significantly better human-judged translation quality than the BLEU-tuned baseline. TESLA-M in particular is simple and performs well in practice on large datasets. We release all our implementation under an open source license. It is our hope that this work will encourage the machine translation community to finally move away from BLEU as the unquestioned default and to consider the new generation metrics when tuning their systems.

1 Introduction

The dominant framework of machine translation (MT) today is statistical machine translation (SMT) (Hutchins, 2007). At the core of the system is the decoder, which performs the actual translation. The decoder is parameterized, and estimating the optimal set of parameter values is of paramount importance in getting good translations. In SMT, the parameter space is explored by a tuning algorithm, typically MERT (Minimum Error Rate Training) (Och, 2003), though the exact method is not important for our purpose. The tuning algorithm carries out repeated experiments with different decoder parameter values over a *development data set*, for which reference translations are given. An automatic MT evaluation metric compares the output of the decoder against the reference(s), and guides the tuning algorithm towards iteratively better decoder parameters and output translations. The quality of the automatic MT evaluation metric therefore has an immediate effect on the whole system.

The first automatic MT evaluation metric to show a high correlation with human judgment is BLEU (Papineni et al., 2002). Together with its close variant the NIST metric, they have quickly become the standard way of tuning statistical machine translation systems. While BLEU is an impressively simple and effective metric, recent evaluations have shown that many new generation metrics can outperform BLEU in terms of correlation with human judgment (Callison-Burch et al., 2009; Callison-Burch et al., 2010). Some of these new metrics include METEOR (Banerjee and Lavie, 2005; Lavie and Agarwal, 2007), TER (Snover et al., 2006), MAXSIM (Chan and Ng, 2008; Chan and Ng, 2009), and TESLA (Liu et al., 2010).

Given the close relationship between automatic MT and automatic MT evaluation, the logical expectation is that a better MT evaluation metric would

Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, pages 375–384,
Edinburgh, Scotland, UK, July 27–31, 2011.

lead to better MT systems. However, this linkage has not yet been realized. In the SMT community, MT tuning still uses BLEU almost exclusively.

Some researchers have investigated the use of better metrics for MT tuning, with mixed results. Most notably, Padó et al. (2009) reported improved human judgment using their entailment-based metric. However, the metric is heavy weight and slow in practice, with an estimated runtime of 40 days on the NIST MT 2002/2006/2008 dataset, and the authors had to resort to a two-phase MERT process with a reduced n-best list. As we shall see, our experiments use the similarly sized WMT 2010 dataset, and most of our runs take less than one day.

Cer et al. (2010) compared tuning a phrase-based SMT system with BLEU, NIST, METEOR, and TER, and concluded that BLEU and NIST are still the best choices for MT tuning, despite the proven higher correlation of METEOR and TER with human judgment.

In this work, we investigate the effect of MERT using BLEU, TER, and two variants of TESLA, TESLA-M and TESLA-F, on Joshua (Li et al., 2009), a state-of-the-art hierarchical phrase-based SMT system (Chiang, 2005; Chiang, 2007). Our empirical study is carried out in the context of WMT 2010, for the French-English, Spanish-English, and German-English machine translation tasks. We show that Joshua responds well to the change of evaluation metric, in that a system trained on metric M typically does well when judged by the same metric M. We further evaluate the different systems with manual judgments and show that the TESLA family of metrics (both TESLA-M and TESLA-F) significantly outperforms BLEU when used to guide the MERT search.

The rest of this paper is organized as follows. In Section 2, we describe the four evaluation metrics used. Section 3 outlines our experimental set up using the WMT 2010 machine translation tasks. Section 4 presents the evaluation results, both automatic and manual. Finally, we discuss our findings in Section 5, future work in Section 6, and conclude in Section 7.

2 Evaluation metrics

This section describes the metrics used in our experiments. We do not seek to explain all their variants and intricate details, but rather to outline their core characteristics and to highlight their similarities and differences. In particular, since all our experiments are based on single references, we omit the complications due to multiple references and refer our readers instead to the respective original papers for the details.

2.1 BLEU

BLEU is fundamentally based on n-gram match precisions. Given a reference text R and a translation candidate T, we generate the bag of all n-grams contained in R and T for $n = 1, 2, 3, 4$, and denote them as $\mathrm{BNG}^{\mathrm{n}}_{\mathrm{R}}$ and $\mathrm{BNG}^{\mathrm{n}}_{\mathrm{T}}$ respectively. The n-gram precision is thus defined as

$$P_n = \frac{|\mathrm{BNG}^{\mathrm{n}}_{\mathrm{R}} \cap \mathrm{BNG}^{\mathrm{n}}_{\mathrm{T}}|}{|\mathrm{BNG}^{\mathrm{n}}_{\mathrm{T}}|}$$

To compensate for the lack of the recall measure, and hence the tendency to produce short translations, BLEU introduces a *brevity penalty*, defined as

$$\mathrm{BP} = \begin{cases} 1 & \mathrm{if} |\mathrm{T}| > |\mathrm{R}| \\ e^{1-|R|/|T|} & \mathrm{if} |\mathrm{T}| \le |\mathrm{R}| \end{cases}$$

where the $|\cdot|$ operator denotes the size of a bag or the number of words in a text. The metric is finally defined as

$$\mathrm{BLEU}(\mathrm{R}, \mathrm{T}) = \mathrm{BP} \times \sqrt[4]{P_1 P_2 P_3 P_4}$$

BLEU is a very simple metric requiring neither training nor language-specific resources. Its use of the brevity penalty is however questionable, as subsequent research on n-gram-based metrics has consistently found that recall is in fact a more potent indicator than precision (Banerjee and Lavie, 2005; Zhou et al., 2006; Chan and Ng, 2009). As we shall see, despite the BP term, BLEU still exhibits a strong tendency to produce short translations.

2.2 TER

TER is based on counting transformations rather than n-gram matches. The metric is defined as the minimum number of edits needed to change a candidate translation T to the reference R, normalized by the length of the reference, i.e.,

$$\mathrm{TER(R,T)} = \frac{\text{number of edits}}{|\mathrm{R}|}$$

One edit is defined as one insertion, deletion, or substitution of a single word, or the shift of a contiguous sequence of words, regardless of size and distance. Minimizing the edit distance so defined has been shown to be NP-complete, so the evaluation is carried out in practice by a heuristic greedy search algorithm.

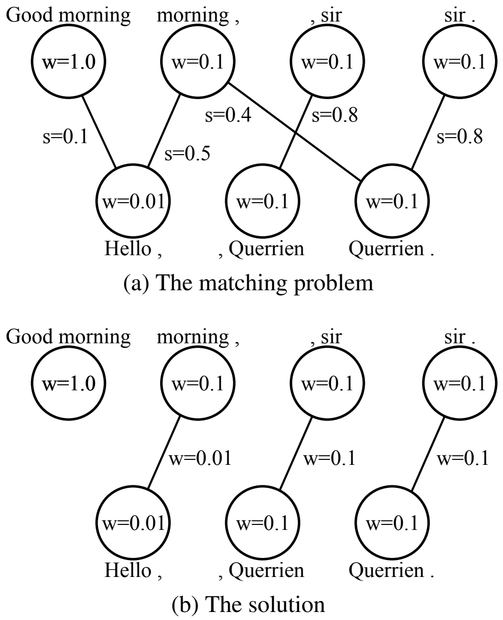
TER is a strong contender as the leading new generation automatic metric and has been used in major evaluation campaigns such as GALE. Like BLEU, it is simple and requires no language specific resources. TER also corresponds well to the human intuition of an evaluation metric.

2.3 TESLA-M

TESLA[1] is a family of linear programming-based metrics proposed by Liu et al. (2010) that incorporates many newer ideas. The simplest variation is TESLA-M[2], based on matching bags of n-grams (BNG) like BLEU. However, unlike BLEU, TESLA-M formulates the matching process as a real-valued linear programming problem, thereby allowing the use of weights. An example weighted BNG matching problem is shown in Figure 1.

Two kinds of weights are used in TESLA-M. First, the metric emphasizes the content words by discounting the weight of an n-gram by 0.1 for every function word it contains. Second, the *similarity* between two n-grams is a function dependent on the lemmas, the WordNet synsets (Fellbaum, 1998), and the POS tag of every word in the n-grams.

Each node in Figure 1 represents one weighted n-gram. The four in the top row represent one BNG, and the three at the bottom represent the other BNG. The goal of the linear programming problem is to assign weights to the links between the two BNGs, so as to maximize the sum of the products of the link weights and their corresponding similarity scores.



(a) The matching problem

(b) The solution

Figure 1: Matching two weighted bags of n-grams. w denotes the weight and s denotes the similarity.

The constraints of the linear programming problem are: (1) all assigned weights must be non-negative, and (2) the sum of weights assigned to all links connecting a node cannot exceed the node's weight. Intuitively, we allow splitting n-grams into fractional counts, and match them giving priority to the pairs with the highest similarities.

The linear programming formulation ensures that the matching can be solved uniquely and efficiently. Once the solution is found and let the maximized objective function value be S, the precision is computed as S over the sum of weights of the translation candidate n-grams. Similarly, the recall is S over the sum of weights of the reference n-grams. The precision and the recall are then combined to form the F-0.8 measure:

$$F_n = \frac{\text{Precision} \times \text{Recall}}{0.8 \times \text{Precision} + 0.2 \times \text{Recall}}$$

This F-measure gives more importance to the recall, reflecting its closer correlation with human judgment. F_n for $n = 1, 2, 3$ are calculated and averaged to produce the final score.

TESLA-M gains an edge over the previous two metrics by the use of lightweight linguistic features such as lemmas, synonym dictionaries, and POS

[1] The source code of TESLA is available at nlp.comp.nus.edu.sg/software/

[2] M stands for *minimal*.

Metric	Spearman's rho
TESLA-F	.94
TESLA-M	.93
meteor-next-*	.92
1-TERp	.90
BLEU-4-v13a-c	.89

Table 1: Selected system-level Spearman's rho correlation with the human judgment for the into-English task, as reported in WMT 2010.

Metric	Spearman's rho
TESLA-M	.93
meteor-next-rank	.82
1-TERp	.81
BLEU-4-v13a-c	.80
TESLA-F	.76

Table 2: Selected system-level Spearman's rho correlation with the human judgment for the out-of-English task, as reported in WMT 2010.

tags. While such tools are usually available even for languages other than English, it does make TESLA-M more troublesome to port to non-English languages.

TESLA-M did well in the WMT 2010 evaluation campaign. According to the system-level correlation with human judgments (Tables 1 and 2), it ranks top for the out-of-English task and very close to the top for the into-English task (Callison-Burch et al., 2010).

2.4 TESLA-F[3]

TESLA-F builds on top of TESLA-M. While word-level synonyms are handled in TESLA-M by examining WordNet synsets, no modeling of phrase-level synonyms is possible. TESLA-F attempts to remedy this shortcoming by exploiting a phrase table between the target language and another language, known as the pivot language.

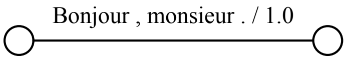


Figure 2: A degenerate confusion network in French. The phrase table maps *Good morning , sir .* to *Bonjour , monsieur .*

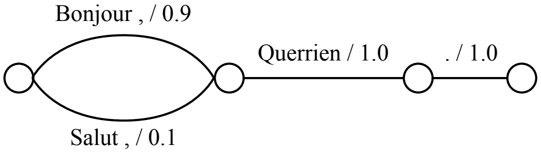


Figure 3: A confusion network in French. The phrase table maps *Hello ,* to *Bonjour ,* with $P = 0.9$ and to *Salut ,* with $P = 0.1$.

Assume the target language is English and the pivot language is French, i.e., we are provided with an English-French phrase table. Let R and T be the reference and the translation candidate respectively, both in English. As an example,

R: Good morning , sir .
T: Hello , Querrien .

TESLA-F first segments both R and T into phrases to maximize the probability of the sentences. For example, suppose both *Good morning , sir .* and *Hello ,* can be found in the English-French phrase table, and proper name *Querrien* is out-of-vocabulary, then a likely segmentation is:

R: ||| Good morning , sir . |||
T: ||| Hello , ||| Querrien ||| . |||

Each English phrase is then mapped to a bag of weighted French phrases using the phrase table, transforming the English sentences into confusion networks resembling Figures 2 and 3. French n-grams are extracted from these confusion network representations, known as pivot language n-grams. The bag of pivot language n-grams generated by R is then matched against that generated by T with the same linear programming formulation used in TESLA-M.

TESLA-F incorporates all the F-measures used in TESLA-M, with the addition of (1) the F-measures generated over the pivot language n-grams described above, and (2) the normalized language model score, defined as $\frac{1}{n} \log P$, where n is the length of the translation, and P the language model probability. Unlike BLEU and TESLA-M which rely on simple averages (geometric and arithmetic average respectively) to combine the component scores, TESLA-

[3] TESLA-F refers to the metric called TESLA in (Liu et al., 2010). To minimize confusion, in this work we call the metric TESLA-F and refer to the whole family of metrics as TESLA. F stands for *full*.

F trains the weights over a set of human judgments using a linear ranking support vector machine (RSVM). This allows TESLA-F to exploit its components more effectively, but also makes it more tedious to work with and introduces potential domain mismatch problems.

TESLA-F makes use of even more linguistic information than TESLA-M, and has the capability of recognizing some forms of phrase synonyms. TESLA-F ranked top for the into-English evaluation task in WMT 2010 (Table 1). However, the added complexity, in particular the use of the language model score and the tuning of the component weights appear to make it less stable than TESLA-M in practice. For example, it did not perform as well in the out-of-English task.

3 Experimental setup

We run our experiments in the setting of the WMT 2010 news commentary machine translation campaign, for three language pairs:

1. French-English (fr-en): the training text consists of 84624 sentences of French-English bitext. The average French sentence length is 25 words.

2. Spanish-English (es-en): the training text consists of 98598 sentences of Spanish-English bitext. The average Spanish sentence length is 25 words.

3. German-English (de-en): the training text consists of 100269 sentences of German-English bitext. The average German sentence length is 22 words.

The average English sentence length is 21 words for all three language pairs. The text domain is newswire report, and the English sides of the training texts for the three language pairs overlap substantially. The development data are 2525 four-way translated sentences, in English, French, Spanish, and German respectively. Similarly, the test data are 2489 four-way translated sentences. As a consequence, all MT evaluations involve only single references.

We follow the standard approach for training hierarchical phrase-based SMT systems. First, we tokenize and lowercase the training texts and create word alignments using the Berkeley aligner (Liang et al., 2006; Haghighi et al., 2009) with five iterations of training. Then, we create suffix arrays and extract translation grammars for the development and test set with Joshua in its default setting. The maximum phrase length is 10. For the language model, we use SRILM (Stolcke, 2002) to build a trigram model with modified Kneser-Ney smoothing from the monolingual training data supplied in WMT 2010.

	fr-en	es-en	de-en
BLEU	3:49 (4)	5:09 (6)	2:41 (4)
TER	4:03 (4)	3:59 (4)	3:59 (5)
TESLA-M	13:00 (3)	17:34 (5)	13:40 (4)
TESLA-F	35:07 (4)	40:54 (4)	40:28 (5)

Table 3: Z-MERT training times in hours:minutes and number of iterations in parenthesis

Parameter tuning is carried out using Z-MERT (Zaidan, 2009). TER and BLEU are already implemented in the publicly released version of Z-MERT, and Z-MERT's modular design makes it easy to integrate TESLA-M and TESLA-F into the package. The maximum number of MERT iterations is set to 100, although we observe that in practice, the algorithm converges after 3 to 6 iterations. The number of intermediate initial points per iteration is set to 20 and the n-best list is capped to 300 translations. Table 3 shows the training times and the number of MERT iterations for each of the language pairs and evaluation metrics.

We use the publicly available version of TESLA-F, which comes with phrase tables and a ranking SVM model trained on the WMT 2010 development data.

4 Automatic and manual evaluations

The results of the automatic evaluations are presented in Table 4. The best score according to each metric is shown in bold. Note that smaller TER scores are better, as are larger BLEU, TESLA-M, and TESLA-F scores.[4]

We note that Joshua generally responds well to the change of tuning metric. A system tuned on met-

[4] The TESLA-F scores shown here have been monotonically scaled.

tune\test	BLEU	TER	TESLA-M	TESLA-F
BLEU	0.5237	0.6029	0.3922	0.4114
TER	**0.5239**	**0.6028**	0.3880	0.4095
TESLA-M	0.5005	0.6359	**0.4170**	0.4223
TESLA-F	0.4992	0.6377	0.4164	**0.4224**

(a) The French-English task

tune\test	BLEU	TER	TESLA-M	TESLA-F
BLEU	0.5641	0.5764	0.4315	0.4328
TER	**0.5667**	**0.5725**	0.4204	0.4282
TESLA-M	0.5253	0.6246	**0.4511**	0.4398
TESLA-F	0.5331	0.6111	0.4498	**0.4409**

(b) The Spanish-English task

tune\test	BLEU	TER	TESLA-M	TESLA-F
BLEU	**0.4963**	**0.6329**	0.3369	0.3927
TER	**0.4963**	0.6355	0.3191	0.3851
TESLA-M	0.4557	0.7055	**0.3784**	**0.4070**
TESLA-F	0.4642	0.6888	0.3753	0.4068

(c) The German-English task

Table 4: Automatic evaluation scores

	P(A)	Kappa
French-English	0.6846	0.5269
Spanish-English	0.6124	0.4185
German-English	0.3973	0.0960

Table 5: Inter-annotator agreement

	fr-en	es-en	de-en
BLEU	44.1%	33.8%	49.6%
TER	41.4%	34.4%	47.8%
TESLA-M	65.8%	49.5%	57.8%
TESLA-F	66.4%	53.8%	55.1%

Table 6: Percentage of times each system produces the best translation

ric M usually does the best or very close to the best when evaluated by M. On the other hand, the differences between different systems can be substantial, especially between BLEU/TER and TESLA-M/TESLA-F.

In addition to the automatic evaluation, we enlisted twelve judges to manually evaluate the first 200 test sentences. Four judges are assigned to each of the three language pairs. For each test sentence, the judges are presented with the source sentence, the reference English translation, and the output from the four competing Joshua systems. The order of the translation candidates is randomized so that the judges will not see any patterns. The judges are instructed to rank the four candidates, and ties are allowed.

The inter-annotator agreement is reported in Table 5. We consider the judgment for a pair of system outputs as one data point. Let $P(A)$ be the proportion of times that the annotators agree, and $P(E)$ be the proportion of times that they would agree by chance. The Kappa coefficient is defined as

$$\text{Kappa} = \frac{\text{P(A)} - \text{P(E)}}{1 - \text{P(E)}}$$

In our experiments, each data point has three possible values: A is preferred, B is preferred, and no preference, hence $P(E) = 1/3$. Our Kappa is calculated in the same way as the WMT workshops (Callison-Burch et al., 2009; Callison-Burch et al., 2010).

Kappa coefficients between 0.4 and 0.6 are considered *moderate*, and our values are in line with those reported in the WMT 2010 translation campaign. The exception is the German-English pair, where the annotators only reach *slight* agreement. This might be caused by the lower quality of German to English translations compared to the other two language pairs.

Table 6 shows the proportion of times each system produces the best translation among the four. We observe that the rankings are largely consistent across different language pairs: Both TESLA-F and TESLA-M strongly outperform BLEU and TER. Note that the values in each column do not add up to 100%, since the candidate translations are often identical, and even a different translation can receive the same human judgment.

Table 7 shows our main result, the pairwise comparison between the four systems for each of the language pairs. Again the rankings consistently show that both TESLA-F and TESLA-M strongly outperform BLEU and TER. All differences are statistically significant under the Sign Test at $p = 0.01$, with the exception of TESLA-M vs TESLA-F in the French-English task, BLEU vs TER in the Spanish-English task, and TESLA-M vs TESLA-F and BLEU vs TER in the German-English task. The results provide strong evidence that tuning machine

A\B	BLEU	TER	TESLA-M	TESLA-F
BLEU	-	11.4% / 6.5%	29.1% / 52.1%	28.0% / 52.3%
TER	6.5% / 11.4%	-	28.6% / 54.5%	27.5% / 55.0%
TESLA-M	52.1% / 29.1%	54.5% / 28.6%	-	~~7.6% / 8.8%~~
TESLA-F	52.3% / 28.0%	55.0% / 27.5%	~~8.8% / 7.6%~~	-

(a) The French-English task. All differences are significant under the Sign Test at $p = 0.01$, except the strikeout TESLA-M vs TESLA-F.

A\B	BLEU	TER	TESLA-M	TESLA-F
BLEU	-	~~25.8% / 22.3%~~	31.0% / 50.6%	24.4% / 50.8%
TER	~~22.3% / 25.8%~~	-	31.9% / 51.0%	26.4% / 52.4%
TESLA-M	50.6% / 31.0%	51.0% / 31.9%	-	25.9% / 33.4%
TESLA-F	50.8% / 24.4%	52.4% / 26.4%	33.4% / 25.9%	-

(b) The Spanish-English task. All differences are significant under the Sign Test at $p = 0.01$, except the strikeout BLEU vs TER.

A\B	BLEU	TER	TESLA-M	TESLA-F
BLEU	-	~~21.8% / 18.4%~~	28.1% / 36.9%	27.3% / 35.3%
TER	~~18.4% / 21.8%~~	-	26.9% / 39.5%	27.3% / 37.5%
TESLA-M	36.9% / 28.1%	39.5% / 26.9%	-	~~24.3% / 21.3%~~
TESLA-F	35.3% / 27.3%	37.5% / 27.3%	~~21.3% / 24.3%~~	-

(c) The German-English task. All differences are significant under the Sign Test at $p = 0.01$, except the strikeout BLEU vs TER, and TESLA-M vs TESLA-F.

Table 7: Pairwise system comparisons. Each cell shows the proportion of time the system tuned on A is preferred over the system tuned on B, and the proportion of time the opposite happens. Notice that the upper right half of each table is the mirror image of the lower left half.

translation systems using the TESLA metrics leads to significantly better translation output.

5 Discussion

We examined the results manually, and found that the relationship between the types of mistakes each system makes and the characteristics of the corresponding metric to be intricate. We discuss our findings in this section.

First we observe that BLEU and TER tend to produce very similar translations, and so do TESLA-F and TESLA-M. Of the 2489 test sentences in the French-English task, BLEU and TER produced different translations for only 760 sentences, or 31%. Similarly, TESLA-F and TESLA-M gave different outputs for only 857 sentences, or 34%. In contrast, BLEU and TESLA-M gave different translations for 2248 sentences, or 90%. It is interesting to find that BLEU and TER should be so similar, considering that they are based on very different principles. As a metric, TESLA-M is certainly much more similar to BLEU than TER is, yet they behave very differently when used as a tuning metric.

We also observe that TESLA-F and TESLA-M tend to produce much longer sentences than do BLEU and TER. The average sentence lengths of the TESLA-F- and TESLA-M-tuned systems across all three language pairs are 26.5 and 26.6 words respectively, whereas those for BLEU and TER are only 22.4 and 21.7 words. Comparing the translations from the two groups, the tendency of BLEU and TER to pick shorter paraphrases and to drop function words is unmistakable, often to the detriment of the translation quality. Some typical examples from the French-English task are shown in Figure 4.

Interestingly, the human translations average only 22 words, so BLEU and TER translations are in fact much closer on average to the reference lengths, yet their translations often feel too short. In contrast, manual inspections reveal no tendency for TESLA-F and TESLA-M to produce overly long translations.

These observations suggest that the brevity penalty of BLEU is not aggressive enough. Neither is TER, which penalizes insertions and deletions equally. Interestingly, by placing much more emphasis on the recall, TESLA-M and TESLA-F produce translations that are statistically too long, but feel much more 'correct' lengthwise.

Another major difference between TESLA-M/TESLA-F and BLEU/TER is that the TESLAs heavily discount n-grams with function words. One might thus expect the TESLA-tuned systems to be less adept at function words; yet they translate them surprisingly well, as shown in Figure 4. One explanation is of course the sentence length effect we have discussed. Another reason may be that since the metric does not care much about function words, the language model is given more freedom to pick function words as it sees fit, without the fear of large penalties. Paradoxically, by reducing the weights of function words, we end up making better translations for them.

TER is the only metric that allows cheap block movements, regardless of size or distance. One might reasonably speculate that a TER-tuned system should be more prone to reordering phrases. However, we find no evidence that this is so.

The relative performance of TESLA-M vs TESLA-F is unsurprising. TESLA-F, being heavier and slower, produces somewhat better results than its minimalist counterpart, though the margin is far less pronounced than that between TESLA-M and the conventional BLEU and TER. Since extra resources including bitexts are needed in using TESLA-F, TESLA-M emerges as the MT evaluation metric of choice for tuning SMT systems.

6 Future work

We have presented empirical evidence that the TESLA metrics outperform BLEU for MT tuning in a hierarchical phrase-based SMT system. At the same time, some open questions remain unanswered. We intend to investigate them in our future work.

The work of (Cer et al., 2010) investigated the effect of tuning a phrase-based SMT system and found that of the MT evaluation metrics that they tried, none of them could outperform BLEU. We would like to verify whether TESLA tuning is still preferred over BLEU tuning in a phrase-based SMT system.

Based on our observations, it may be possible to improve the performance of BLEU-based tuning by (1) increasing the brevity penalty; (2) introducing

BLEU	in the future , americans want a phone *that* allow the user to . . .
TER	in the future , americans want a phone *that* allow the user to . . .
TESLA-M	in the future , *the* americans want a *cell* phone *, which* allow the user to . . .
TESLA-F	in the future , *the* americans want a phone *that* allow the user to . . .
BLEU	. . . also for interest on debt of the state . . .
TER	. . . also for interest on debt of the state . . .
TESLA-M	. . . also for *the* interest on debt of the state . . .
TESLA-F	. . . also for *the* interest on debt of the state . . .
BLEU	and it is *hardly* the end of carnival-like transfers .
TER	and it is *hardly* the end of carnival-like transfers .
TESLA-M	and it is *far from being* the end of *the* carnival-like transfers .
TESLA-F	and it is *far from being* the end of *the* carnival-like transfers .
BLEU	it is not certain that the state can act without money .
TER	it is not certain that the state can act without money .
TESLA-M	it is not certain that the state can act without *this* money .
TESLA-F	it is not certain that the state can act without *this* money .
BLEU	but the expense of a debt of the state . . .
TER	but the expense of a debt of the state . . .
TESLA-M	but *at* the expense of a *greater* debt of the state . . .
TESLA-F	but *at* the expense of a *great* debt of the state . . .

Figure 4: Comparison of selected translations from the French-English task

a recall measure and emphasizing it over precision; and/or (3) introducing function word discounting. In the ideal case, such a modified BLEU metric would deliver results similar to that of TESLA-M, yet with a runtime cost closer to BLEU. It would also make porting existing tuning code easier.

7 Conclusion

We demonstrate for the first time that a practical new generation MT evaluation metric can significantly improve the quality of automatic MT compared to BLEU, as measured by human judgment. We hope this work will encourage the MT research community to finally move away from BLEU and to consider tuning their systems with a new generation metric.

All the data, source code, and results reported in this work can be downloaded from our website at `http://nlp.comp.nus.edu.sg/software`.

Acknowledgments

This research was done for CSIDM Project No. CSIDM-200804 partially funded by a grant from the National Research Foundation (NRF) administered by the Media Development Authority (MDA) of Singapore.

References

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. *Proceedings of the ACL 2005 Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*.

Chris Callison-Burch, Philipp Koehn, Christof Monz, and Josh Schroeder. 2009. Findings of the 2009 workshop on statistical machine translation. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*.

Chris Callison-Burch, Philipp Koehn, Christof Monz, Kay Peterson, Mark Przybocki, and Omar F. Zaidan. 2010. Findings of the 2010 joint workshop on statistical machine translation and metrics for machine translation. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*.

Daniel Cer, Christopher D. Manning, and Daniel Jurafsky. 2010. The best lexical metric for phrase-based statistical MT system optimization. In *Human Language Technologies: The 2010 Annual Conference of* *the North American Chapter of the Association for Computational Linguistics*.

Yee Seng Chan and Hwee Tou Ng. 2008. MaxSim: A maximum similarity metric for machine translation evaluation. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics*.

Yee Seng Chan and Hwee Tou Ng. 2009. MaxSim: performance and effects of translation fluency. *Machine Translation*, 23(2):157–168, September.

David Chiang. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*.

David Chiang. 2007. Hierarchical phrase-based translation. *Computational Linguistics*, 33(2):201–228.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. The MIT press.

Aria Haghighi, John Blitzer, John DeNero, and Dan Klein. 2009. Better word alignments with supervised ITG models. In *Proceedings of 47th Annual Meeting of the Association for Computational Linguistics and the 4th IJCNLP of the AFNLP*.

John W. Hutchins. 2007. Machine translation: A concise history. *Computer Aided Translation: Theory and Practice*.

Alon Lavie and Abhaya Agarwal. 2007. METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*.

Zhifei Li, Chris Callison-Burch, Chris Dyer, Juri Ganitkevitch, Sanjeev Khudanpur, Lane Schwartz, Wren N.G. Thornton, Jonathan Weese, and Omar F. Zaidan. 2009. Joshua: An open source toolkit for parsing-based machine translation. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*.

Percy Liang, Ben Taskar, and Dan Klein. 2006. Alignment by agreement. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*.

Chang Liu, Daniel Dahlmeier, and Hwee Tou Ng. 2010. Tesla: Translation evaluation of sentences with linear-programming-based analysis. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*.

Sebastian Padó, Daniel Cer, Michel Galley, Dan Jurafsky, and Christopher D. Manning. 2009. Measuring machine translation quality as semantic equivalence: A metric based on entailment features. *Machine Translation*, 23(2):181–193, August.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*.

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 Seventh Conference of the Association for Machine Translation in the Americas*.

Andreas Stolcke. 2002. SRILM - an extensible language modeling toolkit. In *Proceedings of the International Conference on Spoken Language Processing*.

Omar Zaidan. 2009. Z-MERT: A fully configurable open source tool for minimum error rate training of machine translation systems. *The Prague Bulletin of Mathematical Linguistics*, 91:79–88.

Liang Zhou, Chin-Yew Lin, and Eduard Hovy. 2006. Re-evaluating machine translation results with paraphrase support. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*.