TERp System Description

Matthew Snover and **Nitin Madnani** and **Bonnie Dorr**
Laboratory for Computational Linguistics
and Information Processing
Institute for Advanced Computer Studies
Department of Computer Science
University of Maryland
College Park, MD 20742
{snover, nmadnani, bonnie}@umiacs.umd.edu

Richard Schwartz
BBN Technologies
10 Moulton Street
Cambridge, MA 02138, USA
schwartz@bbn.com

Abstract

This paper describes TER-Plus (TERp) the University of Maryland / BBN Technologies submission for the NIST Metric MATR 2008 workshop on automatic machine translation evaluation metrics. TERp is an extension of Translation Edit Rate (TER) that builds off of the success of TER as an evaluation metric and alignment tool while addressing several of its weaknesses through the use of paraphrases, morphological stemming, and synonyms, as well as edit costs that are optimized to correlate better with various types of human judgments.

1 Introduction

TER-Plus, or TERp[1], is an automatic evaluation metric for machine translation (MT) that scores a translation, the *hypothesis*, of a foreign language text, the *source*, against a translation of the source text that was created by a human translator, which we refer to as a *reference* translation. Automatic MT evaluation metrics compare the hypothesis against a set of reference translations and assign a score to the similarity.

TERp follows this methodology and builds upon an already existing evaluation metric, Translation Error Rate (TER) (Snover et al., 2006). In addition to assigning a score to a hypothesis, TER also provides an alignment between the hypothesis and the reference, enabling it to be useful beyond general translation evaluation. While TER has been shown to correlate well with human judgments of translation quality, it has several flaws, including the use of only a single reference translation and measuring similarity only with exact word matches between the hypothesis and the reference. The handicap of using a single reference can be addressed by the construction of a lattice of reference translations, a technique that has been used to combine the output of multiple translation systems (Rosti et al., 2007). TERp does not utilize this methodology[2] and instead focuses on addressing the exact matching flaw of TER. A description of TER is presented in section 2. The details of the additions and changes to TER that comprise TERp are discussed in section 3.

2 Translation Edit Rate (TER)

One of the first automatic metrics used to evaluate automatic machine translation (MT) systems was Word Error Rate (WER), which is the standard evaluation metric for Automatic Speech Recognition. WER is computed as the Levenshtein (Levenshtein, 1966) distance between the words of the system output and the words of the reference translation divided by the length of the reference translation. Unlike speech recognition, there are many correct translations for any given foreign sentence. These correct translations differ not only in their word choice but also in the order in which the words occur. WER is generally seen as inadequate for evaluation for machine translation as it fails to combine knowledge from multiple reference translations and also fails to model the reordering of words and phrases in translation.

Translation Error Rate (TER) addresses the lat-

[1] Named after the nickname–”terp”–of the University of Maryland, College Park, mascot: the diamondback terrapin.

[2] The technique of combining references in this fashion has not been evaluated in terms of its benefit when correlating with human judgments.

ter failing of WER by allowing block movement of words, called *shifts* within the hypothesis as a low cost edit, a cost of 1, the same as the cost for inserting, deleting or substituting a word. TER uses a greedy search and a number of shift constraints to both reduce the computational complexity and better model the quality of translation. Examining a larger set of shifts, or choosing them in a more optimal fashion might result in a lower TER score but would not necessarily improve the ability of the measure to determine the quality of a translation. The constraints used by TER are as follows:

1. Shifts are selected by a greedy algorithm that selects the shift that most reduces the WER between the reference and the hypothesis.

2. The sequence of words shifted in the hypothesis must exactly match the sequence of words in the reference that it is being shifted to align to.

3. The words to be shifted must contain at least one error, according to the WER, before being shifted. This prevents the shifting of words that currently correctly matched.

4. The matching words in reference that are being shifted to must also contain at least one error. This prevents shifting to align to words that already correctly aligned.

When TER is used in the case of multiple references, it scores the hypothesis against each reference indiviually. The reference with which the hypothesis has the fewest number of edits is deemed the closet reference, and that number of edits is used to determine the TER score in Equation 1.

$$\text{TER} = \frac{\text{Number of Edits}}{\text{Average Number of Reference Words}} \quad (1)$$

3 TER-Plus

TER-Plus extends the TER framework beyond the limitation of exact matches through the addition of three new types of edit operations–stem matches, synonym matches, and phrase substitutions–which are detailed in section 3.1. These changes allow a relaxing of the shifting constraints used in TER, which is presented in section 3.2. The setting of the TERp edit costs to maximize correlation with human judgments is described in section 3.4.

In studies with human judgments case sensitivity in TERp has not been found to be beneficial to the metric, and actually significantly decreases correlation with human judgment.[3] For this reason, TERp is, by default, case insensitive. In addition, while Equation 1 allows TER to exceed 1.0 if the number of edits exceeds the number of reference words, TERp caps its error rate at 1.0.

3.1 Stem, Synonym, and Paraphrase Substitutions

TERp uses all the edit operations of TER–Matches, Insertions, Deletions, Substitutions and Shifts– as well as three new edit operations: Stem Matches, Synonym Matches and Phrase Substitutions. Rather than treating all substitutions as edits of cost one, the cost of a substitution in TERp varies so that a lower cost is used if the two words are synonyms (a Synonym Match), share the same stem (a Stem Match), or are paraphrases of each other (a Phrase Substitution). The cost of these new types of edits is set, along with the other edit costs, according to the type of human judgment that TERp is optimized towards as described in section 3.4.

TERp identifies words in the hypothesis and reference that share the same stem using the Porter stemming algorithm (Porter, 1980). Two words are determined to be synonyms if they share the same synonym set according to WordNet (Fellbaum, 1998). Sequences of words in the reference are considered to be paraphrases of a sequence of words in the hypothesis if that phrase pair occurs in the TERp paraphrase phrase table, the generation of which is discussed in section 3.3.

With the exception of the phrase substitutions, the edit operations used by TERp are fixed cost edits, meaning the edit cost is the same regardless of what the words in question are. The cost of a phrase substitution is a function of the probability of the paraphrase and the number of edits needed to align the two phrases according to TERp, without the use of phrase substitutions; in effect, the probability of the

[3] A similar effect has been observed for both TER and BLEU which are typically case-senstive. Both of these measures correlate significantly better when they ignore the case of words.

paraphrase is used to determine how much to discount the alignment of the two phrases. The cost of a phrase substitution between the reference phrase, p_1 and the hypothesis phrase p_2 is:

$$\begin{aligned}\text{cost}(p_1,p_2) =& w_1 + \\ & w_2 \,\text{edit}(p_1,p_2) \log(\Pr(p_1,p_2)) + \\ & w_3 \,\text{edit}(p_1,p_2) \Pr(p_1,p_2) + \\ & w_4 \,\text{edit}(p_1,p_2)\end{aligned}$$

This edit cost for phrasal substitutions is therefore specified by four parameters, w_1, w_2, w_3 and w_4. Only paraphrases specified in the input phrase table are considered for phrase substitutions. In addition, the cost for a phrasal substitution is limited to values greater than or equal to 0, so that the substitution cost cannot go negative.

3.2 Modification of Shift Criteria

TER only allows shifts if the two strings (the word sequence in the MT output and the word sequence in the reference) match exactly. This was originally done as a computational shortcut. However, the increased speeds from the latest version of `TERcom` remove this necessity. TERp allows shifts if the words being shifted are exactly the same, are synonyms, stems or paraphrases of each other, or any such combination. These words are not be counted as matches after the shift, only when calculating the set of possible shifts. In addition, a set of stop words limits shifts so that common words, such as punctuation, ”the”, ”a, and others, are not shifted by themselves, but only if a non-stop word is also shifted, reducing the number of shifts needed to be considered and preventing shifts that do not correspond to increased translation quality.

More relaxed shift constraints have been explored that allowed shifts even if some words did not match at all. This greatly increased the number of shifts considered, but also significantly decreased correlation with human judgment. The shift constraints imposed by TER and TERp serve not only to speed up the algorithm but also correspond to those shiftss that correspond with increased translation quality.

3.3 Paraphrase Generation

TERp uses probabilistic phrasal substitutions to align phrases in the hypothesis with phrases in the reference. It does so by looking up—in a pre-computed phrase table—paraphrases of phrases in the reference and using its associated edit cost as the cost of performing a match against the hypothesis. The paraphrases used in TERp are extracted using the pivot-based method (Bannard and Callison-Burch, 2005) with several additional filtering mechanisms to increase the precision. The corpus used for extraction was an Arabic-English newswire bitext containing a million sentences. A few examples of the extracted paraprhase pairs that were actually used in a run of TERp on development data provided by NIST are shown below:

(*brief* → *short*)
(*controversy over* → *polemic about*)
(*by using power* → *by force*)
(*response* → *reaction*)

A probability for each paraphrase pair is computed as described by Bannard and Callison-Burch (2005).

The phrase table for TERp contains 14,184,361 paraphrases. Paraphrases are only used if the reference side of the paraphrase occurs exactly in the reference translation, allowing us to filter the paraphrase phrase table according to a reference set, allowing a much smaller phrase table to used in evaluation without a change in results. The process of filtering the phrase table takes approximately ten minutes, depending on the evaluation set and computer speed, but only needs to be done once for a given reference set. This filtered phrase table is much smaller, often one-hundreth the size of the original or smaller, allowing it to be quickly loaded and searched.

3.4 TERp Edit Cost Optimization

The uniform weights of TER, where all edits have cost 1 except for matches which have cost 0, might prove adequate for the purpose of measuring translation quality as evidenced by correlation with human judgments for both TER and HTER, but it should not be assumed that these weights are ideal for maximizing correlation.

Different types of human judgments, such as fluency and adequacy, are likely to have different characteristics and thus different edit costs might lead to

							Phrase Substitution			
Match	Insert	Deletion	Substitution	Stem	Synonym	Shift	w_1	w_2	w_3	w_4
0.0	0.26	1.43	1.56	0.0	0.0	0.56	-0.23	-0.15	-0.08	0.18

Table 1: TERp Edit Costs

better correlations with each measure. In an extreme case, a translation could be viewed as fully fluent even if it did not correspond to the foreign text that was translated, so long as it was fluent in the target language. Adequacy on the other hand measures only whether the meaning is captured, not whether the translation is fluent in the target language. Because of these differences, one might hypothesize that the cost for a stem substitution should be very low if you wish to correlate with adequacy but much higher if we wish to correlate with fluency, as correctly translating the tense or inflection of a word would likely have no effect on capturing the meaning but would cause the text to not read as fluent English.

TERp uses 11 parameters, or edit costs, four of which are for phrasal substitutions, that require optimization. The match cost is held fixed at 0, so that only the 10 other parameters can vary during optimization. All edit costs, except for the phrasal substitution parameters, are also limited so that they cannot be below 0. A simple hill-climbing search is used to optimize the edit costs, so as to maximize the correlation of human judgments with the TERp score.

TERp was optimized to maximize segment level Pearson correlation with Adequacy on a subset of the MATR08 MT06 data. The edit costs for this optimization, which we refer to as TERp$_A$, are shown in Table 1. These weights are suitable for evaluating adequacy; different weights should be used if evaluation places an emphasis on fluency or has a desire to correlate with HTER or other measures of human judgment.

3.5 TERp Alignment

In addition to providing a score indicating the quality of a translation, TERp also generates an alignment between the hypothesis and the reference, indicating which words are correct, incorrect, misplaced, or are close to the reference translation. While the quality of this alignment is limited by the similarity of the reference translation to the hypothesis translation—a problem that is rectified when using targeted references as in HTER—it can be beneficial in diagnosing error types in MT systems.

Consider an example MT output from the MATR08 MT06 data set and one of the four reference translations, the one closest to the MT output according to TERp, shown in Figure 1. A portion of the HTML output of the alignment generated by TERp is shown in Figure 2. The alignment shown is the final alignment after all shifts are performed. Three shifts were performed by TERp: ”won the” is shifted to align with ”victory to the”, ”islamic” is shifted to align with ”muslim” and finally ”candidates” is shifted to the right to align with ”candidates” in the reference. Three phrasal substitutions were used in the final alignment:

(*the muslim* → *the islamic*) probability = 0.008016
(*victory to* → *won*) probability = 0.005643
(*election* → *electoral*) probability = 0.014986

Each word or phrase in the hypothesis is aligned to a word or phrase in the reference, with the symbol between the word or phrases indicating the type of edit: ”I” for insertions, ”D” for deletions, ”S” for substitutions, ”T” for stem matches, ”Y” for synonym matches, and ”P” for phrasal substitutions. The lack of a symbol indicates a exact match.

The alignment generated for a hypothesis and reference depends upon the edit costs used. For instance, the edit costs for stem matches, synonym matches and matches were all equal in TERp when generating the example alignment, causing TERp to score the alignment where ”gains” in the reference is a deletion and ”made” in the reference is aligned to ”gains” in the hypothesis (a synonym according to WordNet) equally to aligning ”gains” in the reference and hypothesis to each other and marking ”made” as a deletion. The two possible align-

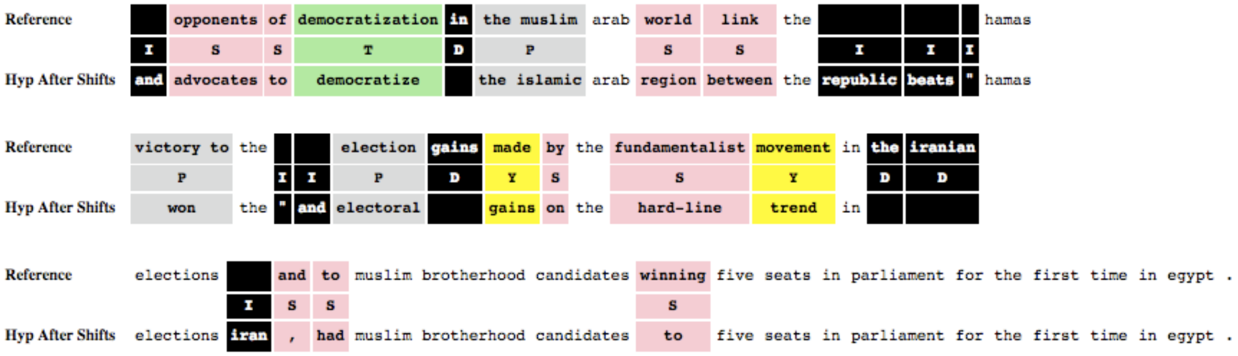
Reference

Opponents of democratization in the Muslim Arab world link the Hamas victory to the election gains made by the fundamentalist movement in the Iranian elections and to Muslim Brotherhood candidates winning five seats in parliament for the first time in Egypt.

Hypothesis

And advocates to democratize the Arab region between the Islamic Republic Beats ”Hamas” and electoral gains on the hard-line trend in elections Iran, candidates had won the Muslim Brotherhood to five seats in parliament for the first time in Egypt.

Figure 1: Example Hypothesis and Reference



```
Reference             opponents of democratization in the muslim  arab world  link    the                  hamas
                  I   S         S  T               D  P                S      S                I        I     I
Hyp After Shifts  and advocates to democratize        the islamic arab region between the republic beats " hamas

Reference         victory to  the         election  gains made  by the fundamentalist movement in the iranian
                  P               I I   P         D     Y     S      S              Y           D   D
Hyp After Shifts  won         the " and electoral       gains on the hard-line      trend    in

Reference         elections      and to  muslim brotherhood candidates winning five seats in parliament for the first time in egypt .
                            I    S   S                                 S
Hyp After Shifts  elections iran ,   had muslim brotherhood candidates to      five seats in parliament for the first time in egypt .
```

Figure 2: Example of TERp HTML Alignment Output

ments are scored equally and the former is arbitrary picked by TERp. If the cost of a synonym match was greater than zero, then the latter alignment would have been preferred.

Availability

TERp is available on the web for download at: http://www.umiacs.umd.edu/∼snover/terp/.

Acknowledgments

This work was supported, in part, by BBN Technologies under the GALE Program, DARPA/IPTO Contract No. HR0011-06-C-0022.

References

Colin Bannard and Chris Callison-Burch. 2005. Paraphrasing with Bilingual Parallel Corpora. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*, pages 597–604, Ann Arbor, Michigan, June.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press. http://www.cogsci.princeton.edu/˜wn [2000, September 7].

V. I. Levenshtein. 1966. Binary Codes Capable of Correcting Deletions, Insertions, and Reversals. *Soviet Physics Doklady*, 10:707–710.

Martin F. Porter. 1980. An algorithm for suffic stripping. *Program*, 14(3):130–137.

Antti-Veikko Rosti, Spyros Matsoukas, and Richard Schwartz. 2007. Improved word-level system combination for machine translation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 312–319, Prague, Czech Republic, June. 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 Association for Machine Translation in the Americas*.