TER-Plus: Paraphrase, Semantic, and Alignment Enhancements to Translation Edit Rate

Matthew G. Snover (snover@umiacs.umd.edu)
Laboratory for Computational Linguistics and Information Processing
Institute for Advanced Computer Studies, University of Maryland, College Park

Nitin Madnani (nmadnani@umiacs.umd.edu)
Laboratory for Computational Linguistics and Information Processing
Institute for Advanced Computer Studies, University of Maryland, College Park

Bonnie Dorr (bonnie@umiacs.umd.edu)
Laboratory for Computational Linguistics and Information Processing
Institute for Advanced Computer Studies, University of Maryland, College Park
Human Language Technology Center of Excellence

Richard Schwartz (schwartz@bbn.com)
BBN Technologies

Abstract. This paper describes a new evaluation metric, TER-Plus (TERP) for automatic evaluation of machine translation. TERP is an extension of Translation Edit Rate (TER). It builds on the success of TER as an evaluation metric and alignment tool and addresses several of its weaknesses through the use of paraphrases, stemming, synonyms, as well as edit costs that can be automatically optimized to correlate better with various types of human judgments. We present a correlation study comparing TERP to BLEU, METEOR and TER, and illustrate that TERP can better evaluate translation adequacy.

Keywords: machine translation evaluation, paraphrasing, alignment

1. Introduction

TER-Plus, or TERP [1] (Snover et al., 2009), 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. The set of possible correct translations is very large, possibly infinite, and any one reference translation represents a single point in that space. Frequently, multiple reference translations—typically 4—are provided to give broader sampling of the space of correct translations. Automatic MT evaluation metrics compare the hypothesis against this set of reference translations and assign a score to the similarity, such that a better score is given when the hypothesis is more similar to the references.

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 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 translation quality, it has several flaws: it only considers exact matches when measuring the similarity of the hypothesis and the reference, and it can only compute this measure of similarity against a single reference. The handicap of using a single reference can be addressed by constructing a lattice of reference translations–this technique has been used to combine the output of multiple translation systems (Rosti et al., 2007). TERP does not utilize this methodology and instead addresses the exact matching flaw of TER.

In addition to aligning words in the hypothesis and reference if they are exact matches, TERP uses stemming and synonymy to allow matches between words. It also uses probabilistic phrasal substitutions to align phrases in the hypothesis and reference. These phrase substitutions are generated by considering possible paraphrases of the reference words. Matching using stems and synonyms (Banerjee and Lavie, 2005) as well as using paraphrases (Zhou et al., 2006; Kauchak and Barzilay, 2006) have been shown to be beneficial for automatic MT evaluation. Paraphrases have been shown to be additionally useful in expanding the number of references used for evaluation (Madnani et al., 2008) although they are not used in this fashion within TERP. The use of synonymy, stemming, and paraphrases allows TERP to better cope with the limited number of reference translations provided. TERP was one of the top metrics submitted to the NIST Metrics MATR 2008 challenge (Przybocki et al., 2008), having the highest average rank over all the test conditions (Snover et al., 2009).

We first discuss the original TER metric in Section 2. In Section 3, we present the details of our various enhancements to TER. We then briefly review the alignment capability of TERP along with some examples in Section 4. Finally, in Section 5, we show the results of optimizing TERP for human judgments of adequacy and compare with other established evaluation metrics, followed by an analysis of the relative benefits of each of the new features of TERP in Section 6.

2. Translation Edit Rate (TER)

One of the first automatic metrics used to evaluate automatic machine translation (MT) systems was Word Error Rate (WER) (Niessen et al., 2000), which remains the standard evaluation metric for Automatic

Speech Recognition. WER is computed as the Levenshtein distance between the words of the hypothesis 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 lexical choice but also in the order in which the words occur. WER is inadequate for evaluating machine translation output as it fails to combine knowledge from multiple reference translations and cannot accurately model the reordering of words and phrases that accompanies translation.

TER addresses the latter failing of WER by allowing block movement of words, called *shifts*. within the hypothesis. Shifting a phrase is assumed to have the same *edit cost* as inserting, deleting or substituting a word, regardless of the number of words being shifted. While a general solution to WER with block movements is NP-Complete (Lopresti and Tomkins, 1997), TER computes an approximate solution by using a greedy search to select the words to be shifted, as well as imposing additional constraints on these words. These constraints are intended to simulate the way in which a human editor might choose the words to shift. Other automatic metrics exist that have the same general formulation as TER but address the complexity of shifting in different ways, such as the CDER evaluation metric (Leusch et al., 2006).

The shifting constraints used by TER serve to better model the quality of translation as well as to reduce the model's computational complexity. Examining a larger set of shifts, or choosing them in a more optimal fashion might result in a lower TER score but it 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 chooses the shift that yields the largest reduction in 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 will align with after the shift.

3. The words being shifted, and the matching reference words, must each contain at least one error, according to WER, before the shift occurs. This prevents the shifting of words that are already correctly matched.

When TER is used with multiple references, it does not combine the references, but, instead, scores the hypothesis against each reference

individually – as is the case with metrics such as METEOR (Banerjee and Lavie, 2005). The reference against which the hypothesis has the fewest number of edits is deemed to be the closet reference, and the final TER score is the number of edits in between the hypothesis and this closest reference divided by the average number words across all of the references.

3. TER-Plus (TERp)

TER-Plus extends the TER metric beyond the limitation of exact matches through the addition of three new types of edit operations, detailed in Section 3.1: stem matches, synonym matches, and phrase substitutions using automatically generated paraphrases. These changes allow a relaxing of the shifting constraints used in TER, which is discussed in Section 3.2. In addition, instead of all edit operations having a uniform edit cost of 1—as is the case in TER—the edit costs for TERP can be learned automatically in order to maximize correlation with human judgments. The details of this optimization are presented in Section 3.3.

3.1. STEM, SYNONYM, AND PARAPHRASE SUBSTITUTIONS

In addition to the edit operations of TER—Matches, Insertions, Deletions, Substitutions and Shifts—TERP also uses three new edit operations: Stem Matches, Synonym Matches and Phrase Substitutions. Rather than treating all substitution operations as edits of cost 1, the cost of a substitution in TERP varies so that a lower cost is used if two words are synonyms (a Synonym Match), share the same stem (a Stem Match), or if two phrases are paraphrases of each other (a Phrase Substitution). The cost of these new edit types is set, along with the other edit costs, according to the type of human judgment for which TERP is optimized, as described in section 3.3.

TERP identifies stems and synonyms in the same manner as the METEOR metric (Banerjee and Lavie, 2005), where words are determined to share the same stem using the Porter stemming algorithm (Porter, 1980), and words are determined to be synonyms if they share the same synonym set according to WordNet (Fellbaum, 1998).

Phrase substitutions are identified by looking up—in a pre-computed *phrase table*—probabilistic paraphrases of phrases in the reference to phrases in the hypothesis. The paraphrases used in TERP are automatically extracted using the pivot-based method (Bannard and Callison-Burch, 2005) with several additional filtering mechanisms to increase precision. The pivot-based method identifies paraphrases as English

phrases that translate to the same foreign phrase in a bi-lingual phrase table. The corpus used for paraphrase extraction was an Arabic-English newswire bi-text containing a million sentences, resulting in a phrase table containing approximately 15 million paraphrase pairs. While an Arabic-English corpus was used to generate the paraphrases, the resulting phrase pairs are English only and can be applied to regardless of the source language. We have previously shown that the choice of data for paraphrasing is not of vital importance to TERP's performance (Snover et al., 2009). A few examples of the extracted paraphrase pairs that were actually used by TERP in experiments described later are shown below:

brief ⇒ short
controversy over ⇒ polemic about
by using power ⇒ by force
response ⇒ reaction

Some paraphrases, such as *brief* and *short* are redundant with other edit types used by TERP such as synonym and stem matching.

A probability for each paraphrase pair is estimated as described in Bannard and Callison-Burch (2005). However, studies (Snover et al., 2009) of these paraphrase probabilities have shown that they are not always reliable indicators of the semantic relatedness of phrase pairs and further refinements of these probability estimates might prove valuable to TERP and other machine translation evaluation metrics.

With the exception of the phrase substitutions, all of the edit operations used by TERP have fixed cost edits, i.e., the edit cost is the same regardless of the words in question. 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 without the use of phrase substitutions. In effect, the probability of the paraphrase is used to determine how much to discount the alignment of the two phrases. For a phrasal substitution between a reference phrase r and a hypothesis phrase h where Pr is the the probability of paraphrasing r as h, and $\text{edit}(r, h)$ is number of edits needed to align r and h without any phrasal substitutions, the edit cost is specified by three parameters, w_1, w_2, and w_3 as follows:

$$\text{cost}(r, h) = w_1 + \text{edit}(r, h)(w_2 \log(\text{Pr}) + w_3)$$

Only paraphrases specified in the input phrase table are considered for phrase substitutions. In addition, the total cost for a phrasal substitution is limited to values greater than or equal to 0, to ensure that the edit cost for substitution operations is always non-negative.

The parameter w_1 allows a constant cost to be specified for all phrase substitutions, while parameters w_2 and w_3 adjust the discount applied to the edit cost of the two phrases.

3.2. Additional Differences From Ter

In addition to the new edit operations, Terp differs from Ter in several other ways. First, Terp is insensitive to casing information since we observe that penalizing for errors in capitalization lowers the correlation with human judgments of translation quality. Second, Terp is capped at 1.0. While the formula for Ter allows it to exceed 1.0 if the number of edits exceed the number of words, such a score would be unfair since the hypothesis cannot be more than 100% wrong.

The shifting criteria in Terp have also been relaxed relative to Ter, so that shifts are allowed if the words being shifted are: (i) exactly the same, (ii) synonyms, stems or paraphrases of the corresponding reference words, or (iii) any such combination. In addition, a set of stop words is used to constrain the shift operations such that common words ("the", "a" etc.) and punctuation can be shifted if and only if a non-stop word is also shifted. This reduces the number of shifts considered in the search and prevents any shifts that may not correspond with an increase in translation quality.

More relaxed shift constraints have been explored that allowed shifts even if some words did not match at all. We have empirically found 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 block movement of words that correspond with increased translation quality.

3.3. Terp Edit Cost Optimization

While Ter uses uniform edit costs—1 for all edits except matches—, we seek to improve Terp's correlation with human judgments by weighting different edit types more heavily than others, as some types of errors are more harmful to translation quality than others.

Terp uses a total of eight edit costs. However, the cost of an exact match is held fixed at 0 which leaves a total of seven edit costs that can be optimized. Since the paraphrase edit cost is represented by 3 parameters, this yields a total of 9 parameters that are varied during optimization. All parameters, except for the 3 phrasal substitution parameters, are also restricted to be positive. A hill-climbing search optimizes the parameters to maximize the correlation of human judgments with the Terp score. In this paper, these correlations are measured

at the sentence, or *segment*, level. However, optimization could also be performed to maximize document level correlation or any other measure of correlation with human judgments.

While TERP can be run using a fixed set of parameters, it can be beneficial to tune them depending on the properties of translation desired. Optimization of MT evaluation metrics towards specific human judgment types has previously investigated in a similar manner by Lita et al. (2005). Depending on whether the end goal is to maximize correlation with HTER, adequacy, or fluency, different sets of parameters may better reflect translation performance (Snover et al., 2009).

4. TERp Alignment

In addition to providing a score indicating the quality of a translation, TERP generates an alignment between the hypothesis and reference, indicating which words are correct, incorrect, misplaced, or similar to the reference translation. While the quality of this alignment is limited by the similarity of the reference to the hypothesis it can be beneficial in diagnosing error types in MT systems.

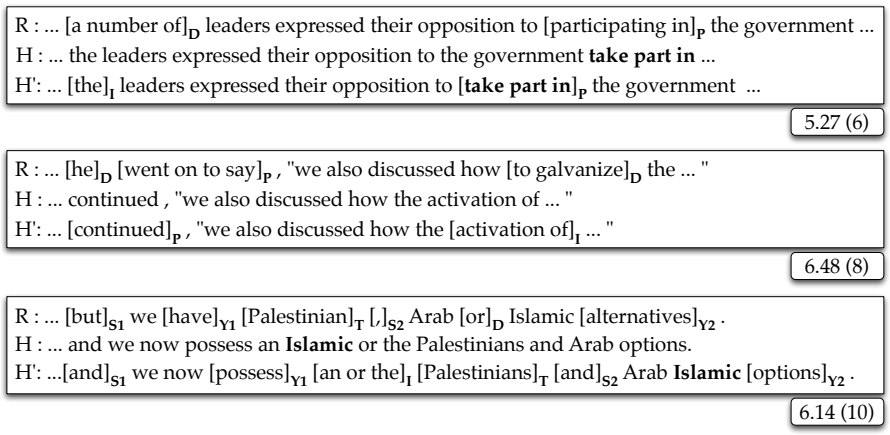

Figure 1. Examples of TERP Alignment Output. In each example, **R**, **H** and **H'** denote the reference, the original hypothesis and the hypothesis after shifting respectively. Shifted words are **bold**ed and other edits are in [brackets]. Number of edits shown: TERP (TER).

Actual examples of TERP alignments are shown in Figure 1. Within each example, the first line is the reference translation, the second line is the original hypothesis, and the third line is the hypothesis after performing all shifts. Words in **bold** are shifted, while square brackets

are used to indicate other edit types: P for phrase substitutions, T for stem matches, Y for synonym matches, D for deletions, and I for insertions.

These alignments allow TERP to provide quality judgments on translations and to serve as a diagnostic tool for evaluating particular types of translation errors. In addition, it may also be used as a general-purpose string alignment tool – TER has been used for aligning multiple system outputs to each other for MT system combination (Rosti et al., 2007), a task for which TERP may be even better suited.

5. Experimental Results

5.1. Optimization For Adequacy

In order to tune and test TERP, we used a portion of the Open MT-Eval 2006 evaluation set that had been annotated for adequacy (on a seven-point scale) and released by NIST as a development set for the Metrics MATR 2008 challenge (Przybocki et al., 2008). This set consists of the translation hypotheses from 8 Arabic-to-English MT systems for 25 documents, which in total consisted of 249 segments. For each segment, four reference translations were also provided. Optimization was done using 2-fold cross-validation. These optimized edit costs (and subsequent results) differ slightly from the formulation of TERP submitted to the Metrics MATR 2008 challenge, where tuning was done without cross-validation. Optimization requires small amounts of data but should be done rarely so that the metric can be held constant to aid in system development and comparison.

Table I. TERP Edit Costs Optimized for Adequacy

Match	**Insert**	**Deletion**	**Substitution**	**Stem**
0.0	0.20	0.97	1.04	0.10
Syn.	**Shift**	**Phrase Substitution**		
0.10	0.27	w_1: 0.0	w_2: -0.12	w_3: 0.19

TERP parameters were then optimized to maximize segment level Pearson correlation with adequacy on the tuning set. The optimized edit costs, averaged between the two splits of the data, are shown in Table I. Because segment level correlation places equal importance on all segments, this optimization over-tunes for short segments, as they have very minor effect at the document or system level. Optimization on

length weighted segment level correlation would rectify this but would result in slightly worse segment level correlations.

5.2. Correlation Results

In our experiments, we compared TERp with METEOR (Banerjee and Lavie, 2005) (version 0.6 using the Exact, WordNet synonym, and Porter stemming modules), TER (version 0.7.25), the IBM version of BLEU (Papineni et al., 2002) with a maximum n-gram size of 4 (BLEU). We also included a better correlating variant of BLEU with a maximum n-gram size of 2 (BLEU-2). TER and both versions of BLEU were run in case insensitive mode as this produces significantly higher correlations with human judgments, while METEOR is already case insensitive.

To evaluate the quality of an automatic metric, we examined the Pearson correlation of the automatic metric scores—at the segment, document and system level— with the human judgments of adequacy. Document and system level adequacy scores were calculated using the length weighted averages of the appropriate segment level scores.

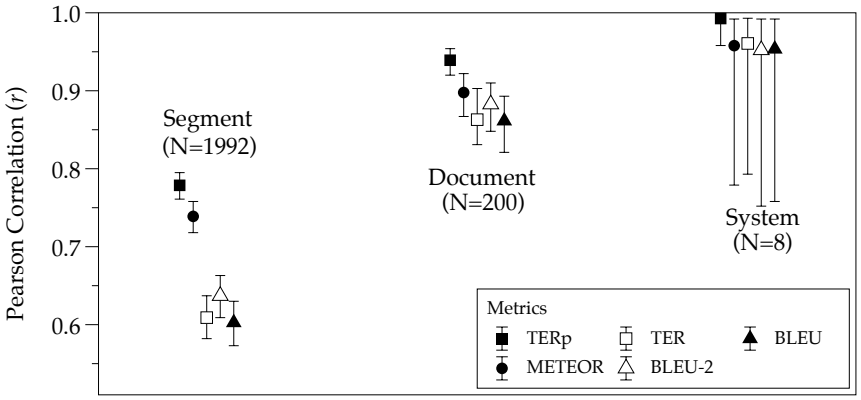

Figure 2. Metric correlations with adequacy on the Metrics MATR 2008 development set. Correlations are significantly different if the center point of one correlation does not lie within the confidence interval of the other correlation.

Pearson correlation results between the automatic metrics and human judgments of adequacy are shown in Figure 2. We can determine whether the difference between two correlation coefficients is statistically significant by examining the confidence interval of the Pearson coefficient,[2] r. If the correlation coefficient for a metric occurs within the 95% confidence interval of another metric, then the difference between the correlations of the metrics is not statistically significant.

TERp consistently outperformed all of the other metrics on the segment, document, and system level Pearson correlations, with all but

one difference being statistically significant. While TERP had higher correlation than TER on the system level, the difference is not statistically significant—the differences with all other metrics are statistically significant. Of the other metrics, METEOR consistently had the highest Pearson correlation at the segment and document level. METEOR, the only other tunable metric, might possibly correlate better by retuning for this dataset, although this is not generally done for METEOR.

6. Benefit of Individual TERp Features

In this section, we examine the benefit of each of the new features of TERP by individually adding each feature to TER and measuring the correlation with the human judgments. Each condition was optimized as described in section 5.1. Figure 3 shows the Pearson correlations for each experimental condition along with the 95% confidence intervals.

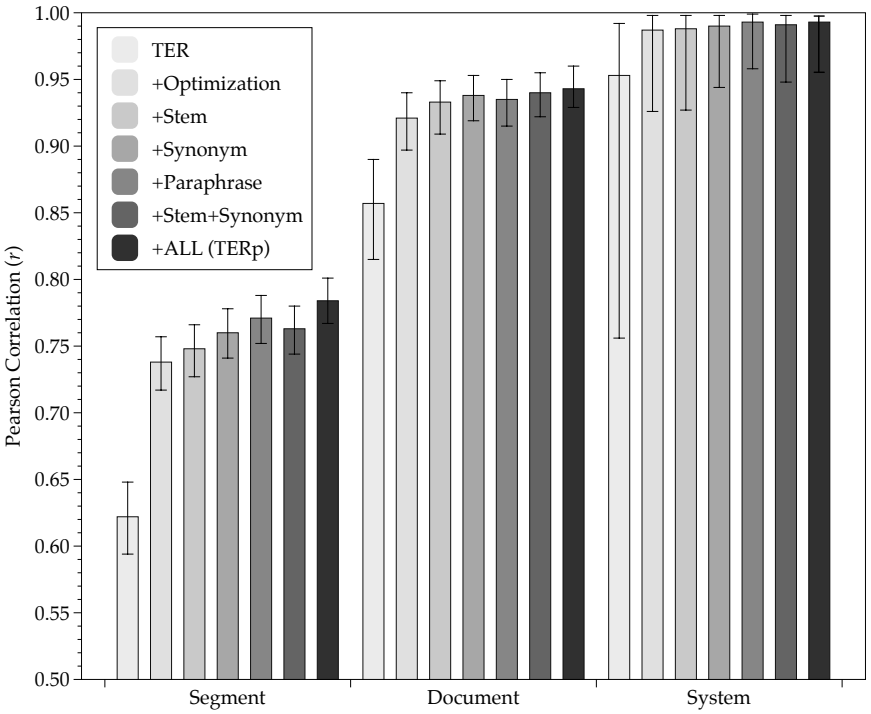

Figure 3. Pearson Correlation of TERP with Selective Features.

The largest gain over TER is through the addition of optimizable edit costs. This takes TER from being a metric with balanced insertion and deletion costs to a recall-oriented metric which strongly penalizes deletion errors, while being forgiving of insertion errors. This single

addition gives statistically significant improvements over TER at the segment and document levels. This validates similar observations of the importance of recall noted by Lavie et al. (2004).

The other three features of TERP— stemming, synonymy, and paraphrases — are added on top of the optimized TER condition since optimization is required to determine the edit costs for the new features. The addition of each of these features increases correlations over the optimized edit costs at all levels, with statistically significant gains at the segment level for the addition of synonymy or paraphrasing. The addition of paraphrasing gives the largest overall gains in correlation after optimization and is more beneficial than stemming and synonymy combined. A large percentage of synonym and stem matches are already captured in the paraphrase set and, therefore, the combination of all three features yields only a small gain over paraphrasing alone.

The TERP framework and software also provides for separate word classes with individual edit costs, so that the edit costs of various sets of words can be increased or decreased. For example, the cost of deleting content words could be set higher than that of deleting function words. It is difficult to set such costs manually as it is not clear how these phenomenon are treated by human annotators of translation quality, although these costs could be determined by automatic optimization.

7. Discussion

TERP extends the TER metric using stems, synonyms, paraphrases, and optimizable edit costs to assign a more realistic score to a translation hypothesis and generate a better alignment against the reference. Experimental results show that TERP achieves significant gains in correlation with human judgments over other MT evaluation metrics. Optimization can be targeted towards specific types of human judgments, yielding different edit costs for TERP, for use in cases when a specific notion of translation quality is desired.

Automatic MT evaluation metrics are used for two major purposes: (1) To compare two or more MT systems (or variants of the same system) in order to determine which system generates better translations. This is often used to show that the addition of a new feature to a translation system yields an improvement over a baseline system. (2) To automatically optimize or tune the parameters of a system. While we conducted this study in the context of the first purpose—showing that TERP provides significant gains in evaluating final system outputs—we have not evaluated TERP for the second purpose. It is frequently the case that automatic metrics that appear useful according to the first

criterion are not suitable for the second purpose, resulting in degenerate system parameters. To evaluate a metric's suitability for optimization, a translation system must be optimized using a baseline metric, such as Bleu, and also using the new metric being examined. The final outputs of the two systems tuned to the different metrics must then be judged by humans to determine which optimization method provides better translations. Unfortunately, this technique is also biased by the translation system that is being tuned and the method used for parameter optimization. Further explorations of this nature are needed to determine if Terp, and other metrics, are suitable for use in MT parameter optimization.

We showed that the addition of stemming, synonymy and, most importantly, paraphrasing to the Ter metric significantly improves its correlation with human judgments. We believe that further significant improvements in Terp and other automatic evaluation metrics are contingent on the use of additional linguistic features so as to better capture the fluency of a translation hypothesis and its similarity in meaning to reference translations.

Acknowledgments

This work was supported, in part, by BBN Technologies under the GALE Program, DARPA/IPTO Contract No. HR0011-06-C-0022 and in part by the Human Language Technology Center of Excellence. The authors would like to thank Philip Resnik, Chris Callison-Burch, Mark Pryzbocki, Sebastian Bronsart and Audrey Le. The Terp software is available online for download at: http://www.umiacs.umd.edu/∼snover/terp/

Notes

[1] Terp is named after the nickname—"terp"—of the University of Maryland, College Park, mascot: the diamondback terrapin.

[2] Confidence intervals are calculated using the Fisher r-to-z transformation, consulting a z-table to find the upper and lower bounds of a 95% confidence interval, and then converting the values back to r scores. This is solely a function of the correlation coefficient, r, and the number of data points, N.

References

Banerjee, S. and A. Lavie: 2005, 'METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments'. In: *Proceedings of the*

ACL 2005 Workshop on Intrinsic and Extrinsic Evaulation Measures for MT and/or Summarization. pp. 228–231.

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

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

Kauchak, D. and R. Barzilay: 2006, 'Paraphrasing for Automatic Evaluation'. In: *Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL*. pp. 455–462.

Lavie, A., K. Sagae, and S. Jayaraman: 2004, 'The Significance of Recall in Automatic Metrics for MT Evaluation'. In: *Proceedings of the 6th Conference of the Association for Machine Translation in the Americas*. pp. 134–143.

Leusch, G., N. Ueffing, and H. Ney: 2006, 'CDER: Efficient MT Evaluation Using Block Movements'. In: *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics)*. pp. 241–248.

Lita, L. V., M. Rogati, and A. Lavie: 2005, 'BLANC: Learning Evaluation Metrics for MT'. In: *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP)*. Vancouver, BC, pp. 740–747.

Lopresti, D. and A. Tomkins: 1997, 'Block Edit Models For Approximate String Matching'. *Theoretical Computer Science* **181**(1), 159–179.

Madnani, N., P. Resnik, B. J. Dorr, and R. Schwartz: 2008, 'Are Multiple Reference Translations Necessary? Investigating the Value of Paraphrased Reference Translations in Parameter Optimization'. In: *Proceedings of the Eighth Conference of the Association for Machine Translation in the Americas*. pp. 143–152.

Niessen, S., F. Och, G. Leusch, and H. Ney: 2000, 'An evaluation tool for machine translation: Fast evaluation for MT research'. In: *Proceedings of the 2nd International Conference on Language Resources and Evaluation*. pp. 39–45.

Papineni, K., S. Roukos, T. Ward, and W.-J. Zhu: 2002, 'Bleu: a Method for Automatic Evaluation of Machine Traslation'. In: *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. pp. 311–318.

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

Przybocki, M., K. Peterson, and S. Bronsart: 2008, 'Official results of the NIST 2008 "Metrics for MAchine TRanslation" Challenge (MetricsMATR08)'. http://nist.gov/speech/tests/metricsmatr/2008/results/.

Rosti, A.-V., S. Matsoukas, and R. Schwartz: 2007, 'Improved Word-Level System Combination for Machine Translation'. In: *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*. Prague, Czech Republic, pp. 312–319.

Snover, M., B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul: 2006, 'A Study of Translation Edit Rate with Targeted Human Annotation'. In: *Proceedings of Association for Machine Translation in the Americas*. pp. 223–231.

Snover, M., N. Madnani, B. Dorr, and R. Schwartz: 2009, 'Fluency, Adequacy, or HTER? Exploring Different Human Judgments with a Tunable MT Metric'. In: *Proceedings of the Fourth Workshop on Statistical Machine Translation*. Athens, Greece, pp. 259–268, Association for Computational Linguistics.

Zhou, L., C.-Y. Lin, and E. Hovy: 2006, 'Re-evaluating Machine Translation Results with Paraphrase Support'. In: *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP 2006)*. pp. 77–84.