

**International Workshop on Spoken Language Translation
Pittsburgh, USA
October 24-25, 2004**

**Evaluating Machine Translation Output
with Automatic Sentence Segmentation**

Evgeny Matusov, Gregor Leusch, Oliver Bender, and Hermann Ney

**Human Language Technology and Pattern Recognition
Lehrstuhl für Informatik VI
Computer Science Department
RWTH Aachen University
D-52056 Aachen**

Content

- 1. related work**
- 2. motivation**
- 3. state-of-the-art translation error measures**
- 4. document-level vs. sentence-level evaluation**
- 5. description of the algorithm**
- 6. experimental results**
- 7. summary**

Related work

- **L. Rabiner and B. H. Juang. *Fundamentals of Speech Recognition*. Prentice Hall, chapter 7.**
- **BLEU: (Papineni et al., 2001), NIST: (Doddington, 2002)**
- **preprocessing/normalization for MT Evaluation: (Leusch et al., 2005)**

Motivation - 1

- **evaluation plays a crucial role in machine translation research and acceptance of MT technology**
- **human evaluation is time-consuming and expensive**
- **automatic evaluation is preferred, but its quality still leaves to be desired**
- **some well-established evaluation measures exist**
 - **WER, PER, BLEU, NIST, ...**
- **all objective measures include the concept of sentences or segments**
- **all use (multiple) reference translations of these segments**
- **each evaluation algorithm expects exactly one target language segment for each source language segment**

Motivation - 2

- **concept of sentences is in general not well-defined for speech translation**
 - **current situation in ASR+MT evaluations:**
 - **humans transcribe the acoustic signal and define segment boundaries**
 - **ASR systems are forced to generate segment boundaries at the same timeframes**
 - **segments may be too short/long, ASR/MT systems may lose context information**
 - **more realistic conditions:**
 - **ASR system suggests segment boundaries based on prosodic or LM features**
 - **MT system may split or merge these segments to meet its constraints or modeling assumptions**
 - **BUT: the segments in the produced translations and manual references will be different!**
- ⇒ **existing MT error measures will not be applicable**

Solution

- **align the output of an MT system with the reference translations**
- **re-segment the MT output based on the segmentation of the reference translations**
- **make use of the Levenshtein edit distance algorithm**
- **take multiple references into account**

Existing Error Measures - 1

- all well-established measures are based on segment-level comparisons
- scores for the whole document are obtained by summation over all segments and normalization
- WER is the Levenshtein (edit) distance
- PER is similar to WER, but ignores the order of words within a segment
- WER and PER are normalized by the total reference length which can be computed in several ways (Leusch et al., 2005)
- BLEU is an m -gram precision measure
- NIST extends BLEU with information weights
- BLEU and NIST use a global brevity penalty to avoid a bias towards short candidate translations

Document-level vs. Segment-level Evaluation

- **evaluation at document level:**
 - assume the whole candidate document and each reference document to have only one segment
 - **computation of WER at document level is possible only using a single reference document (as in ASR)**
 - **PER, BLEU and NIST can be computed at document level, but the estimates of translation quality will be too optimistic**
 - e.g. an m -gram starting with the first word in the candidate document will match an m -gram starting with the 500th word in a reference document
- ⇒ **segment-level evaluation is preferable (with a proper definition of segments)**

Alignment algorithm: Notation

- $w_1, \dots, w_n, \dots, w_N$ is a reference document segmented into K segments
- reference segmentation is defined by indices $n_1, \dots, n_k, \dots, n_K := N$
- candidate document $e_1, \dots, e_i, \dots, e_I$
- goal: find a Levenshtein alignment between the two documents
- mark words which are aligned to w_{n_k} and obtain the segmentation of the candidate document $i_1, \dots, i_k, \dots, i_K := I$.

Dealing with Multiple References

- extend the algorithm to work with multiple reference documents $r = 1, \dots, R$
- without loss of generality, assume that a reference translation of a segment k has the same length across reference documents
 - achieved by inserting artificial “empty word” symbols \$ at the end of reference segments which are shorter than the translation with the maximum length
- consequently, the reference words are indexed by w_{nr} ,
 $n = 1, \dots, N, r = 1, \dots, R$

Algorithm: Within-segment Alignment

- for each candidate word index i , reference word index n , and reference index r , compute Levenshtein distance recursively with dynamic programming using the auxiliary quantity D :

$$D(i, n, r) = \min \begin{cases} D(i-1, n-1, r) + 1 - \delta(e_i, w_{nr}), \\ D(i-1, n, r) + 1, \\ D(i, n-1, r) + 1 - \delta(w_{nr}, \$) \end{cases}$$

- determine, which possibility has lower costs:
 - a match, substitution, insertion or deletion
- special case: deletion with no costs
 - ⇒ reference that does not have the maximum length is already processed
- the index of the last locally best segment boundary is saved in a backpointer $B(i, n, r)$
- the backpointer of the best predecessor hypothesis is passed on in each recursion step

Recombination at Reference Segment Boundaries

- two consecutive candidate segments can be scored with segments from different reference documents

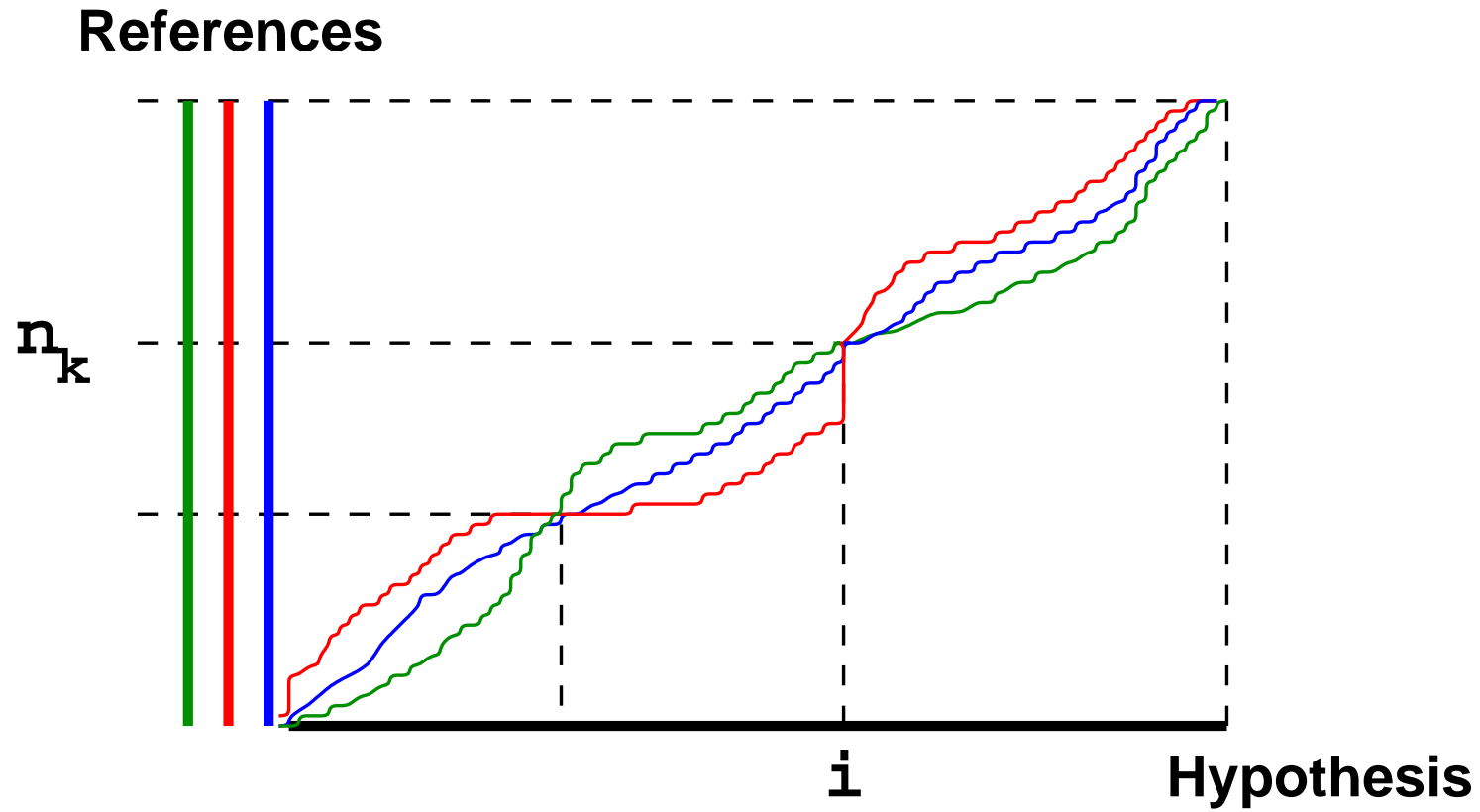
$$D(i, n = n_k, r) = \min_{r'=1, \dots, R} D(i, n - 1, r')$$

$$BR(i, k) = \hat{r} = \operatorname{argmin}_{r'=1, \dots, R} D(i, n - 1, r')$$

$$BP(i, k) = B(i, n - 1, \hat{r})$$

- backpointers pass on the locally optimal reference and the hypothesized segment boundary for the segment $k - 1$

Illustration



Automatic Segmentation Word Error Rate

- algorithm terminates by reaching the last word in candidate and each reference document
- the optimal number of edit operations is given by:

$$d_L = \min_r D(I, N, r)$$

- the sentence boundary decisions i_1, \dots, i_K and the optimal sequence of reference segments $\hat{r}_1, \dots, \hat{r}_K$ are recursively backtraced using the backpointer arrays
- $\hat{r}_1, \dots, \hat{r}_K$ is the new single-reference document \hat{E} with length \hat{N}
- we define the automatic segmentation word error rate (AS-WER) by:

$$\text{AS-WER} = \frac{d_L}{\hat{N}}$$

Complexity

- **memory:** $O(N \cdot R + I \cdot K)$
- **time:** $O(N \cdot I \cdot R)$
- **fast C++ implementation using integer word indices and costs**
- **e.g. 2-3 minutes and max. 400 MB of memory to align 20K words using two reference documents with 2643 segments (desktop PC)**

Experimental results

- **test the new evaluation method on the data from IWSLT 2004 and TC-STAR 2005 evaluations**
- **on the IWSLT 2004 data, compute correlation with human judgements**

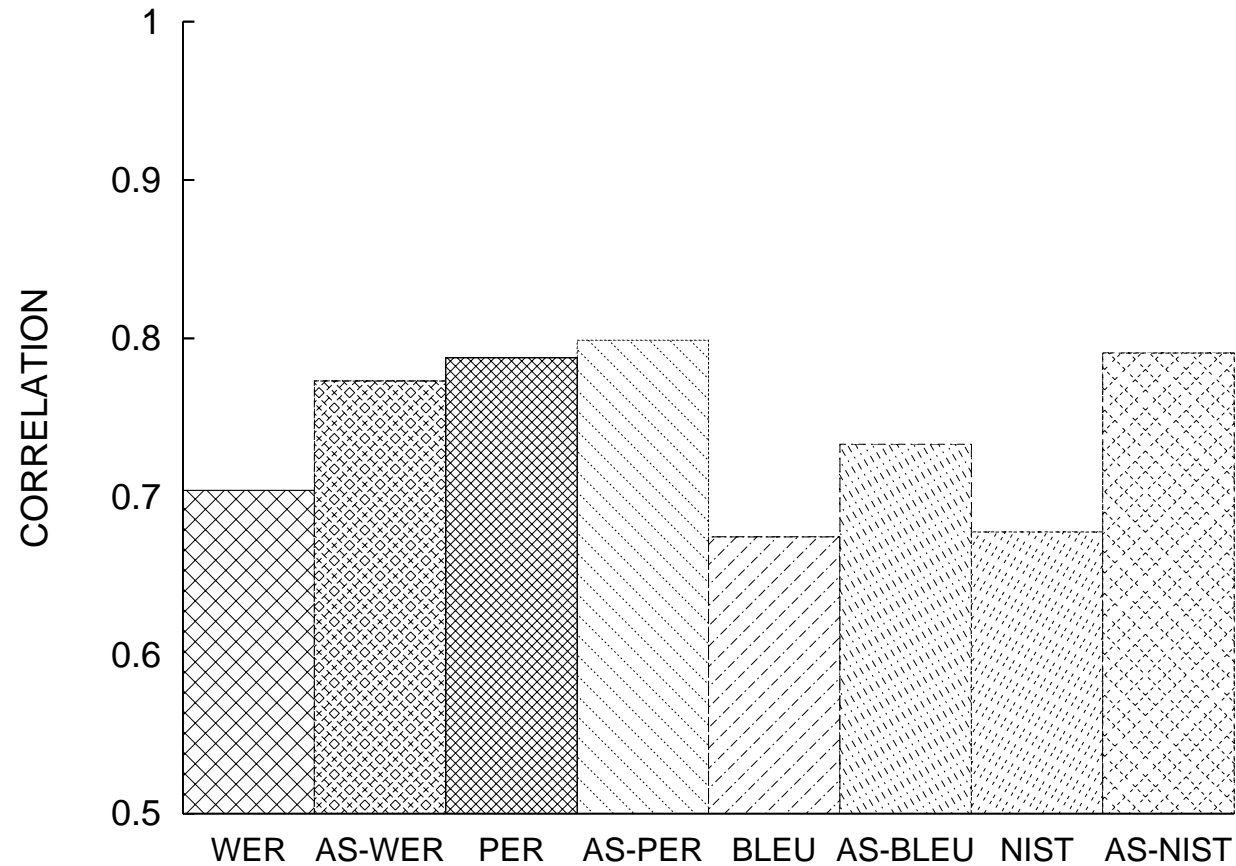
Evaluation Methodology

- **currently, the “correct” manual segmentation of the candidate translation is available**
- **compute WER, PER, BLEU, NIST using either manual or automatic segmentation**
- **compute correlation coefficients with human judgments**
 - **adequacy, fluency (Pearson’s r)**
 - **ranking (Kendall’s τ)**
- **compare relative score changes**
- **compare absolute score values**

Corpus Statistics

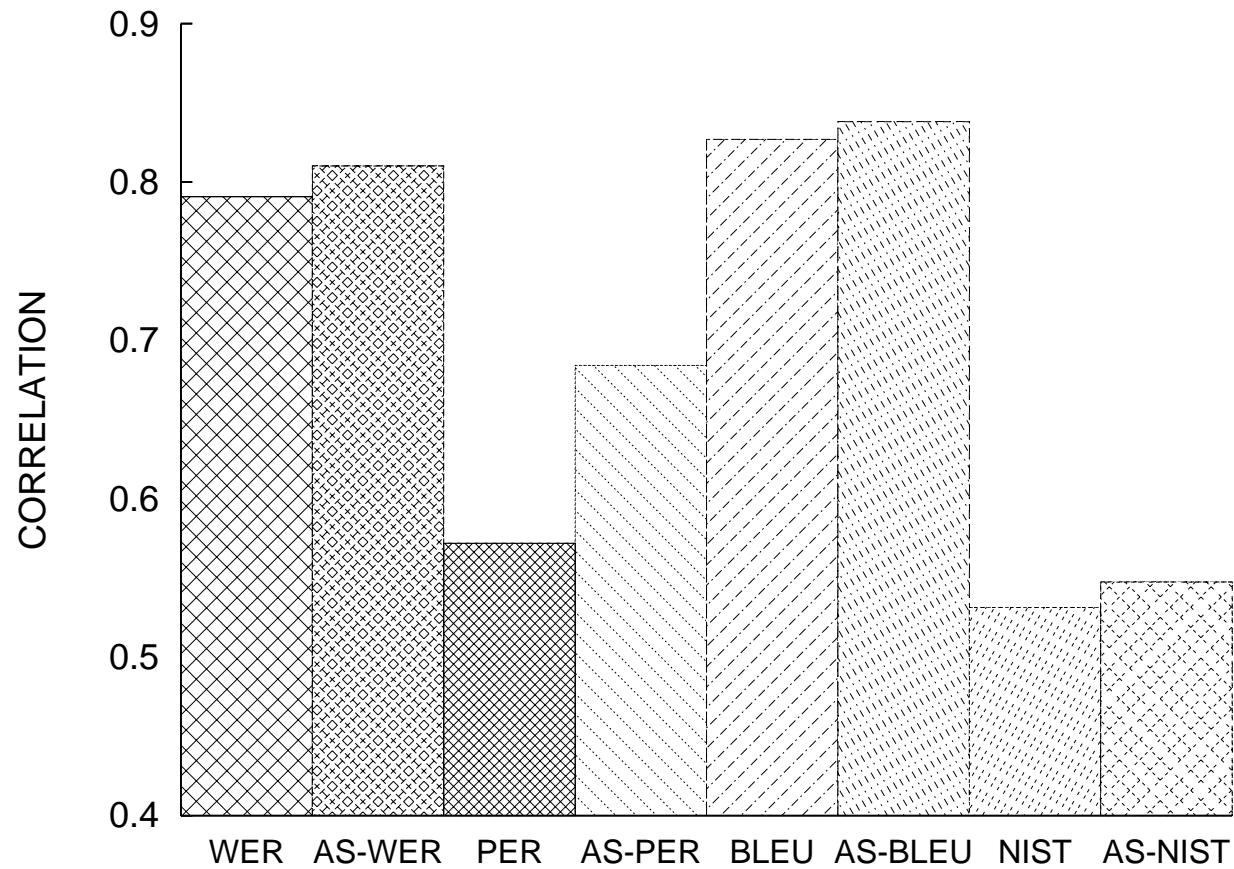
| | TC-STAR | BTEC CE |
|--------------------------|----------------|----------------|
| Source language | Spanish | Chinese |
| Target language | English | English |
| Segments | 2643 | 500 |
| Running words | 20164 | 3632 |
| Ref. translations | 2 | 16 |
| Avg. ref. length | 7.8 | 7.3 |
| Candidate systems | 4 | 20 |

Correlation with Adequacy



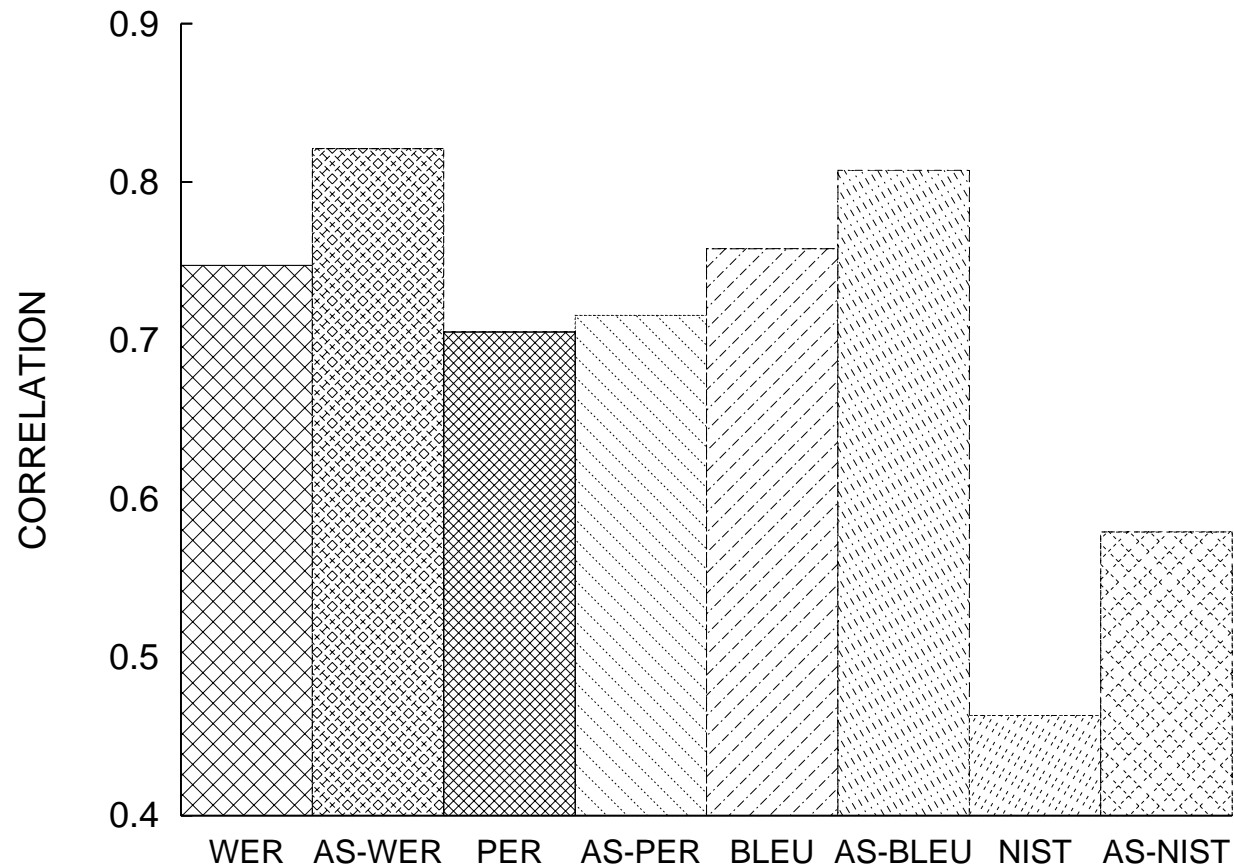
- correlation is slightly better when automatic segmentation is used

Correlation with Fluency



- correlation is slightly better when automatic segmentation is used

Rank Correlation



⇒ **AS-measures are suitable for evaluation and ranking of MT systems**

Error Measures on the TC-STAR task - 1

| Error measure: | System | | | |
|---------------------|--------|------|------|------|
| | A | B | C | D |
| WER [%] | 37.4 | 40.4 | 41.4 | 47.9 |
| AS-WER [%] | 36.2 | 39.1 | 40.0 | 45.7 |
| PER [%] | 30.7 | 33.7 | 33.9 | 40.6 |
| AS-PER [%] | 30.6 | 33.4 | 33.9 | 39.7 |
| BLEU [%] | 51.1 | 47.8 | 47.4 | 40.6 |
| AS-BLEU [%] | 50.9 | 47.5 | 47.2 | 40.6 |
| NIST | 10.34 | 9.99 | 9.74 | 8.65 |
| AS-NIST | 10.29 | 9.92 | 9.68 | 8.65 |
| Segmentation ER [%] | 6.5 | 8.0 | 7.8 | 9.5 |

- automatic segmentation does not change the ranking of the four systems
- absolute values of AS-measures can be slightly lower/higher (e. g. depending on the normalization method)
- segmentation error rate is small and degrades only slightly with degrading WER

Error Measures on the TC-STAR task - 2

| Error measure : | System | | | |
|----------------------------|--------|-------|-------|-------|
| | A | B | C | D |
| BLEU [%] | 51.1 | 47.8 | 47.4 | 40.6 |
| AS-BLEU [%] | 50.9 | 47.5 | 47.2 | 40.6 |
| BLEU at document level [%] | 55.3 | 50.5 | 50.9 | 47.5 |
| NIST | 10.34 | 9.99 | 9.74 | 8.65 |
| AS-NIST | 10.29 | 9.92 | 9.68 | 8.65 |
| NIST at document level | 11.57 | 11.23 | 11.12 | 10.89 |

- **BLEU and NIST scores on document level overestimate the performance of MT systems**
 - moreover, the difference between systems is significantly underestimated
- **AS-BLEU and AS-NIST give reliable estimates of translation quality**

Conclusions

- a novel method of automatic sentence segmentation to be used for evaluation of MT quality
- MT output does not have to have the same segmentation as the reference translations
- automatic segmentation is determined efficiently with a modified edit distance algorithm
- multiple reference translations are taken into account
- existing MT evaluation measures can be applied
- the measures computed using automatic segmentation correlate with human judgement at least as well as when manual segmentation is used
- the new evaluation method is especially important for evaluating translations of automatically recognized and segmented speech
- the method resolves the issue of different segmentation requirements of ASR and MT