Automatic Sentence Segmentation and Punctuation Prediction for Spoken Language Translation

Evgeny Matusov, Arne Mauser, Hermann Ney

Lehrstuhl für Informatik 6,
RWTH Aachen University, Aachen, Germany
{matusov,mauser,ney}@informatik.rwth-aachen.de

Abstract

This paper studies the impact of automatic sentence segmentation and punctuation prediction on the quality of machine translation of automatically recognized speech. We present a novel sentence segmentation method which is specifically tailored to the requirements of machine translation algorithms and is competitive with state-of-the-art approaches for detecting sentence-like units. We also describe and compare three strategies for predicting punctuation in a machine translation framework, including the simple and effective implicit punctuation generation by a statistical phrase-based machine translation system. Our experiments show the robust performance of the proposed sentence segmentation and punctuation prediction approaches on the IWSLT Chinese-to-English and TC-STAR English-to-Spanish speech translation tasks in terms of translation quality.

1. Introduction

In recent years, machine translation (MT) research groups have increasingly considered translating speech as recognized by an automatic speech recognition (ASR) system. Almost all state-of-the-art ASR systems recognize sequences of words, neither performing a proper segmentation of the output into sentences or sentence-like units (SUs), nor predicting punctuation marks. Usually, only acoustic segmentation into *utterances* is performed. These utterances may be very long, containing several sentences. Most MT systems are not able to translate such long utterances with an acceptable level of quality because of the constraints of the involved algorithms. Examples of such constraints include reordering strategies with exponential complexity with regard to the length of the input sequence, or parsing techniques which assume the input to be a more or less syntactically correct sentence. The user of an MT system usually expects to see readable sentences as the translation output, with proper punctuation inserted according to the conventions of the target language.

Given this situation, algorithms are needed for automatic segmentation of the ASR output into SUs and for punctuation prediction. The latter can be performed either in the source or in the target language. In this paper we present a novel approach to sentence segmentation and compare three different strategies for punctuation prediction in the framework of statistical MT. In one of these approaches, the punctuation prediction is integrated with the translation process. We also show experimentally that sentence segmentation can be performed automatically without significant negative effects on the translation quality.

The paper is organized as follows. In section 2, we give a short overview of the published research on SU boundary detection and punctuation prediction. Section 3 presents some details of the statistical phrase-based MT system we use, followed by Section 4 describing the different strategies for punctuation prediction involving this MT system. In Section 5, we describe in detail a novel algorithm for automatic sentence segmentation which was designed especially for the needs of machine translation. Finally, Section 6 describes the experimental results, followed by a summary.

2. Related Work

Previous research on sentence boundary detection and punctuation prediction mostly concentrated on annotating the ASR output as the end product delivered to the user. Most authors tried to combine lexical cues (e. g. language model probability) and prosodic cues (pause duration, pitch, etc.) in a single framework in order to improve the quality of sentence boundary prediction [5]. A maximum entropy model [2] or CART-style decision trees [3] are often used to combine the different features. Various levels of performance are achieved depending on the task, but predicting SUs (i. e. complete or incomplete sentences) is reported to be significantly easier than predicting specific types of punctuation, such as commas and question marks.

Recently, [4] performed automatic punctuation restoration in order to translate ASR output for the TC-STAR 2006 evaluation. In this approach, the segments are already known and each segment is assumed to end with a period so that only commas are predicted. A comma is restored only if the bigram or trigram probability of a comma given the context exceeds a certain threshold. We are not aware of any other published work dealing with the detection of SU boundaries and punctuation in the context of machine translation.

3. Phrase-based MT system of RWTH

In this section we will briefly present the statistical MT system which we use in the experiments for this work. We will denote the (given) source sentence with $f_1^J = f_1 \dots f_J$, which is to be translated into a target language sentence $e_1^I = e_1 \dots e_I$.

Our baseline system maximizes the translation probability directly using a log-linear model [9]:

$$p(e_1^I|f_1^J) = \frac{\exp\left(\sum_{m=1}^{M} \lambda_m h_m(e_1^I, f_1^J)\right)}{\sum_{\tilde{e}_1^I} \exp\left(\sum_{m=1}^{M} \lambda_m h_m(\tilde{e}_1^I, f_1^J)\right)}, \quad (1)$$

with a set of different features h_m, scaling factors λ_m and the denominator a normalization factor that can be ignored in the maximization process. We choose the λ_m by optimizing an MT performance measure on a development corpus using the downhill simplex algorithm.

The most important models in equation (1) are phrase-based models in both source to target and target to source directions. In order to extract these models, an alignment between a source sentence and its target language translation is found for all sentence pairs in the training corpus using the IBM-1, HMM and IBM-4 models in both directions and combining the two obtained alignments [10]. Given this alignment, an extraction of contiguous phrases is carried out and their probabilities are computed by means of relative frequencies [13].

Additionally we use single word based lexica in source to target and target to source direction. This has the effect of smoothing the relative frequencies used as estimates of the phrase probabilities. The phrase-based and single word based probabilities thus yield 4 features of the log-linear model. Another important feature in the log-linear model is the language model, an n-gram language model with Kneser-Ney smoothing. A length and a phrase penalty are the last models in the set of the seven basic models which are used in the system.

4. Sentence Segmentation and Punctuation Prediction in an MT framework

The issue of sentence segmentation arises when translating ASR output. It is important to produce translations of sentences or sentence-like units to make the MT output human-readable. At the same time, sophisticated speech translation algorithms (e. g. ASR word lattice translation, rescoring and system combination algorithms for (N-best) output of one or several SMT systems) may require that the number of words in the input source language SUs is limited to about 30 or 40 words.

Figure 1 depicts three alternative strategies for predicting segment boundaries and punctuation in the process of machine translation of automatically recognized speech. We have investigated each strategy in our experiments. In all three cases, we begin by taking the raw output of an ASR system, which is a long sequence of words. The sentence segmentation algorithm, which will be described in Section 5, is applied to produce sentence-like units of the length acceptable both to humans and as input to an MT system.

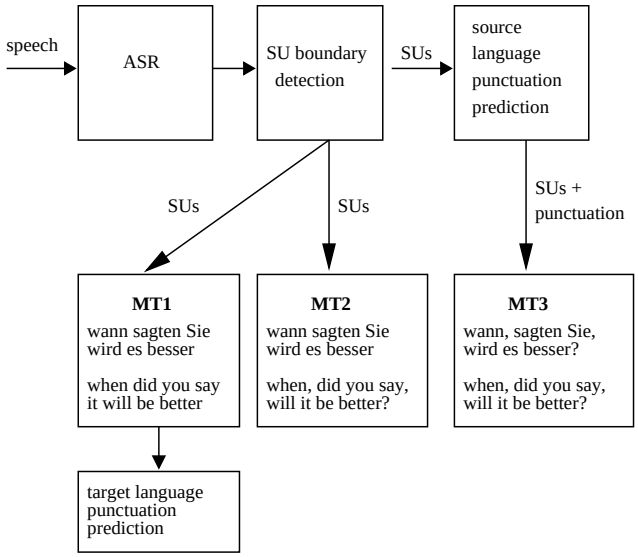

Figure 1: *Three different strategies for predicting punctuation in the process of speech recognition and machine translation.*

Although it is possible to predict punctuation marks in an unsegmented text and then use the automatically inserted periods, question marks, and exclamation marks as segment boundaries, our experiments show that this approach leads to poor segmentation results. It is much easier to predict a segment boundary (considering lexical and also prosodic features like the pause length) than to predict whether a specific punctuation mark has to be inserted or not at a given word position in the transcript. In the context of machine translation, separating sentence segmentation and punctuation prediction also allows for more flexible processing of the determined segments. Here, we are interested in having proper punctuation in the target language translation and thus may want to predict punctuation marks in the target language, where the rules and conventions for punctuation may be different from the source language.

Starting by performing sentence segmentation of the ASR output in the source language, we followed three different approaches with the goal of having punctuation in the target language translations (Figure 1). For each of the approaches, we extracted three different types of bilingual phrase pairs based on the same word alignment between the bilingual sentence pairs in the training data. Thus, three MT systems were created. They will be described in the following subsections.

4.1. Phrase-based MT without Punctuation Marks

In the first system $MT1$ we removed punctuation marks from the source and the target training corpus, adjusting the indices of the alignment accordingly. Thus, the phrases extracted using the modified training corpora and alignment do not contain punctuation marks. With this system, the target language translation of the ASR output also does not contain punctuation marks. Punctuation marks have to be inserted based on the lexical context in the automatically produced translation, e.g. using a hidden-event target language model and the method of [12].

The advantage of this method is the possibility to optimize the parameters of the MT system with the goal of improving the lexical choice independent of any punctuation marks. Also, the absence of punctuation marks allows for better generalization and longer matches of bilingual phrase pairs (see also Section 4.3).

One drawback of the approach is that the punctuation marks then have to be predicted using only language model information. Moreover, this prediction is performed on the translation hypotheses which may contain errors with respect to both word choice and word order. In the current state of technology, these errors are much more numerous than the speech recognition errors. The presence of these errors may result in poor quality of the automatically predicted punctuation. Another drawback is that any prosodic features which are characteristic to a certain punctuation type (e.g. the pitch at the end of a question) cannot be directly used in the target language punctuation prediction. Transferring these features as the annotation of the translation hypothesis may be possible, but is complicated due to the reordering performed in MT.

4.2. Implicit Punctuation Mark Prediction in the MT process

The second system $MT2$ was created by removing punctuation marks only from each source language training sentence, together with their alignment connections to the words in the corresponding target sentence. Thus, the punctuation marks in the target sentence which had been aligned with punctuation marks in the source sentence became non-aligned. Next, in the phrase extraction phase, for the same sequence of words followed or preceded by a punctuation mark, two different phrase pairs were extracted, one containing the target phrase with the punctuation mark, and one with the punctuation mark omitted from the target phrase. In the example in Figure 1, this would mean that e. g. for the phrase `sagten Sie` the MT system would memorize four translations:

```
  did you say
, did you say
  did you say ,
, did you say ,
```

With this heuristic, target phrases with punctuation marks compete with phrases without punctuation marks in the search, and the language model and other features help to select the best hypothesis (see Section 3). It is also possible to optimize the scaling factors of the models involved in the MT system to obtain the best translation performance as measured using reference translations with punctuation marks. This aspect makes the approach more robust than the one where punctuation marks are predicted using only the target language model, in a postprocessing step. In practical terms, this implicit approach is easy to use, since it requires neither preprocessing nor postprocessing with respect to punctuation. This is especially of advantage when taking alternative ASR hypotheses (e. g. ASR word lattices) as input for MT.

Alternatively, the systems $MT1$ and $MT2$ can be trained "from scratch" by removing punctuation marks from the source and target training corpora or only the source training corpus, respectively, and then performing the word alignment training and phrase extraction. This may improve the alignment estimation, especially for small training corpora.

4.3. Phrase-based MT with Punctuation Marks

Finally, for the system $MT3$ the phrase pairs were extracted including punctuation marks both in the source and the target training corpus.

Generally, a system like $MT3$ can be a standard system for translating written text input with correctly placed punctuation marks. In order to use this system for the ASR output, the punctuation has to be predicted in the source language. This is a good strategy if prosodic features are used to improve the performance of the punctuation prediction algorithm. However, if the punctuation prediction algorithm is not robust enough and makes many errors, this may have a significant negative effect on the machine translation quality. For instance, long source phrases with good translations may not match the input due to an extra or missing comma, so that shorter phrases will have to be used, with a negative influence on the fluency and adequacy of the produced translation.

Nowadays, leading MT systems are capable of translating ASR word lattices with alternative ASR hypotheses in order to overcome the negative impact of speech recognition errors. Using the system $MT3$ for lattice translation would mean that punctuation will have to be predicted within a lattice. This is a non-trivial problem, for which an efficient and robust solution is hard to find. Thus, the system $MT3$ is probably not suitable for processing ASR word lattices.

Another disadvantage of this system originates in the differences in punctuation rules and conventions between languages, which make the task of translating punctuation marks from the source to the target language a very ambiguous one. For example, some commas in Chinese are not translated into English. Also, the Mandarin language has two types of commas which have to be either omitted in translation or translated to the ASCII comma in English, etc. Due to this ambiguity, the translation of punctuation marks is not error-free. Thus, we cannot expect much better performance

of $MT3$ which translates punctuation marks than of the system $MT2$ which inserts punctuation marks in the translation process.

5. Novel Sentence Segmentation Algorithm

State-of-the-art approaches to sentence segmentation treat segment boundaries as hidden events. A posterior probability for a possible boundary after a word is determined for each word position. Then, the boundaries are determined by selecting only those positions, for which the posterior probability of a segment boundary exceeds a certain threshold. This means that although the segmentation granularity can be controlled, the length of a segment may take any value from 1 to several hundred words. This may be a disadvantage for further processing of the segmented transcript, which may require the sentence units to be at least m and/or at most M words long.

Our approach to segmentation of ASR output originates from the work of [12] and thus also uses the concept of hidden events to represent the segment boundaries. A decision regarding the placement of a segment boundary is made based on a log-linear combination of language model and prosodic features. However, in contrast to existing approaches, we optimize over the length of each segment (in words) and add an explicit segment length model. Thus, we perform HMM-style search with explicit optimization over the length of a segment. A similar approach to topic segmentation was presented by [7]. Such an approach makes it possible to introduce restrictions on the minimum and maximum length of a segment, and nevertheless produce syntactically and semantically meaningful sentence units which pass all the relevant context information on to the phrase-based MT system.

In the following we present the details of the approach. We are given an (automatic) transcription of speech, denoted by the words $w_1^N := w_1, w_2, \ldots, w_N$. We would like to find the optimal segmentation of this word sequence into K segments, denoted by $i_1^K := (i_1, i_2, \ldots, i_K = N)$. Among all the possible segmentations, we will choose the one with the highest posterior probability:

$$\hat{i}_1^{\hat{K}} = \underset{K, i_1^K}{\operatorname{argmax}} \left\{ Pr(i_1^K | w_1^N) \right\} \quad (2)$$

The posterior probability $Pr(i_1^K|w_1^N)$ is modeled directly using a log-linear combination of several models:

$$Pr(i_1^K|w_1^N) = \frac{\exp\left(\sum_{m=1}^M \lambda_m h_m(i_1^K, w_1^N)\right)}{\sum_{K', i'^{K'}_1} \exp\left(\sum_{m=1}^M \lambda_m h_m(i'^{K'}_1, w_1^N)\right)} \quad (3)$$

The denominator is a normalization factor that depends only on the word sequence w_1^N. Therefore, we can omit it during the search process. As a decision rule, we obtain:

$$\hat{i}_1^{\hat{K}} = \underset{K, i_1^K}{\operatorname{argmax}} \left\{ \sum_{m=1}^M \lambda_m h_m(i_1^K, w_1^N) \right\} \quad (4)$$

5.1. Feature functions

In practice, the features used in Eq. 4 depend on the words within the hypothesized adjacent boundaries at position $i := i_{k-1}$ and at position $j := i_k$ as well as on the prosodic information at the boundary i. To compute probabilities for a hypothesized segment w_{i+1}^j starting with word position $i+1$ and ending at position j, we interpolate log-linearly the following probabilistic features.

The **language model** probability $p_{LM}(w_{i+1}^j)$ for a segment is computed as a product of the following three probabilities:

$$p_{LM}(w_{i+1}^j) = p_S(w_{i+1}^j) \cdot p_I(w_{i+1}^j) \cdot p_E(w_{i+1}^j)$$

These probabilities are modeled as described below (assuming a trigram language model):

- probability of the first two words of a segment (segment **S**tart), conditioned on the last segment boundary represented by a hidden event <s>:

$$p_S(w_{i+1}^j) = p(w_{i+1}|\texttt{<s>}) \cdot p(w_{i+2}|w_{i+1}, \texttt{<s>})$$

- probability of the other words within a segment (**I**nternal probability):

$$p_I(w_{i+1}^j) = \prod_{k=i+3}^{j} p(w_k|w_{k-1}, w_{k-2})$$

- LM probability of the segment boundary (**E**nd) in dependency on the last two words of a segment:

$$p_E(w_{i+1}^j) = p(\texttt{<s>}|w_j, w_{j-1})$$

The probabilities are integrated into the log-linear model by using the negative logarithm of the corresponding probability value as a feature value. The extension to a larger (e. g. 4-gram) context is straightforward.

In addition to the language model probability, we use a prosodic feature, namely the normalized **pause duration** between the words w_i and w_{i+1} located directly before and after the hypothesized boundary. For the normalization, the probability of a segment boundary is set to 1 if the pause is 10 or more seconds long. Other prosodic features can be included with a separate scaling factor, assuming that they also provide a single (posterior) probability for a segment boundary at each word position.

Since the length of the segment is known, we also include an explicit **sentence length** probability feature $-\log p(j-i)$. We usually estimate this distribution on the corpus used to estimate the source language model. We chose the log-normal

distribution for sentence length modeling, because it reflects the actual length histogram most accurately. The parameters of this distribution were determined using maximum likelihood estimation.

We also include a **segment penalty** $h_{\text{SP}}(i_1^K, w_1^N) = K$ in the log-linear combination. This is a simple heuristic that helps to additionally control the segmentation granularity. If the scaling factor λ_{SP} of this model is negative, generally more segments are produced because more segments reduce the total cost of the segmentation. Similarly, for $\lambda_{\text{SP}} > 0$, in general fewer segments are produced by the presented algorithm.

The scaling factors in the log-linear combination of the presented models are currently tuned manually on a development set by computing precision and recall with respect to human reference SUs.

5.2. Search

In search, the word sequence w_1^N is processed from left to right. For all hypothesized segment end positions j, we optimize over the position of the last segment boundary i and calculate the loglinear combination of the scores for the segment w_{i+1}^j as described above. The optimal sentence segmentation solution for words up to position i has already been computed in a previous recursion step and is added to the score for the current segment. The globally optimal sentence segmentation for the document is determined when the last word of the document is reached.

Note that the minimum and/or maximum sentence lengths l and L can be explicitly set by limiting the values of i to $l \leq j - i \leq L$. Since usually the maximum length L does not exceed 50 or 60 words, the algorithm is rather fast: e.g. 30 000 words are segmented in less than a second.

6. Experimental Results

6.1. Evaluation Criteria

To evaluate the quality of the sentence segmentation algorithm as described in Section 5, we compute precision and recall in comparison to the sentence boundaries defined by humans. In case of ASR output, the reference boundaries are inserted in the automatically produced transcript by aligning it with the correct (reference) transcript with the minimum edit distance algorithm.

The quality of machine translation is evaluated with objective error and correctness measures. These measures compare the MT output against human reference translations. We use the common metrics BLEU, NIST, WER, and PER. BLEU [11] and NIST [1] are correctness measures based on the similarity of subsequences of MT output and reference translation. The word error rate WER measures the word insertions, deletions and substitutions between the automatic translation and the reference. The position-independent word error rate PER computes the distance between the sets of words contained in MT output and reference translation.

Table 1: Quality of sentence segmentation measured with Precision (P) and Recall (R) in % for the TC-STAR English ASR output (minimum sentence length set to 3, maximum to 50 words).

	Development		Test	
	P	R	P	R
baseline (4-gram LM only)	54.2	52.1	54.0	50.4
+ length model	54.7	52.5	55.3	51.7
+ pause model	68.8	68.4	70.5	69.7
baseline + pause model	68.1	68.3	69.9	70.3

When translating ASR output with automatic sentence segmentation, the number of automatically determined segments may be different from the number of segments in the human reference translations. In this case, we use the tool of [6] to determine the alignment with the multiple reference translations based on the word error rate and, using this alignment, to re-segment the translation output to match the number of reference segments. Then, the usual MT evaluation measures are computed.

6.2. Quality of sentence segmentation

The experiments for automatic sentence segmentation were performed for the TC-STAR task and for the IWSLT task. For the TC-STAR task (speech recognition and translation of Speeches in the European Parliament), we determined sentence boundaries in the English ASR output for the 2006 English-to-Spanish Speech Translation evaluation. The ASR word error rate (WER) was 6.9%. The scaling factors of the models involved, as well as the minimum and maximum segment length parameters, were tuned manually on the development set (with about 28K words and 1194 segments in the verbatim (correct) transcription) with the goal of increasing and balancing precision and recall. Then, these scaling factors were used for detecting segment boundaries in the evaluation set (with about 28K words and 1155 segments in the verbatim transcription). The precision and recall percentages for the development and test set are given in Table 1.

The baseline system for sentence segmentation only made use of a 4-gram language model trained on the English part of the European Parliament corpus (over 31 million words). The parametric sentence length model was also estimated on this data. The largest gains in performance came from using the pause duration feature, which indicates that in many cases the speakers do make pauses to mark the start of a new sentence. The best segmentation results reach 70% precision and recall.

Further experiments were performed on the IWSLT Chinese-to-English task (2006 evaluation). This task consisted of translating manually and automatically transcribed utterances related to tourism from Chinese to English. For

Table 2: Quality of sentence segmentation measured with Precision (P) and Recall (R) in % for the IWSLT Chinese-English task (minimum sentence length set to 3, maximum to 30 words). Comparison of the RWTH approach with the standard approach of SRI [5]. No prosodic features are used.

	RWTH tool		hidden-ngram	
corpus	P	R	P	R
IWSLT test 2005	84.2	84.1	84.1	85.5
IWSLT dev 2006	59.5	64.6	57.0	62.4
IWSLT test 2006	56.4	61.0	54.9	57.6
IWSLT test 2006 (ASR)	56.0	55.2	55.4	52.6

this task, we did not use the pause duration feature, since all of the utterances had been recorded separately. Instead, we compared the performance of the algorithm across different types of data. The 2005 test set with 3208 words and 506 reference segments is very similar to the training data (around 300K words) on which the 4-gram LM was trained, whereas the 2006 test set with 5550 words and 500 segments contains more spontaneous utterances. We were also interested in the effect of speech recognition errors on sentence segmentation. The Chinese character error rate was 12.8% for the development set and 15.2% for the test set.

Table 2 gives an overview of the segmentation results for this task. The system performs very well on the 2005 test data, but not as well on the more spontaneous data. The ASR errors mostly affect recall, presumably because some of the words which are typical for the beginning or the end of a sentence had not been recognized correctly.

These results are better than or comparable to the well-established approach of SRI [12] using the same language model (cf. the last two columns of Table 2. For the experiments with the SRI toolkit, the threshold for the SU posterior probability was optimized for precision/recall on the same development set.

6.3. Translation quality

Even though we can measure the performance of the sentence segmentation algorithm in terms of precision and recall of the found segment boundaries, it is not clear how automatic segmentation and punctuation prediction affect the quality of machine translation output. Therefore we evaluated the different ways of segmentation and punctuation restoration in a machine translation setup.

As for evaluating the quality of the segmentation, we use the TC-STAR 2006 English-to-Spanish and the IWSLT 2006 Chinese-to-English tasks and compare our results to the evaluation submissions. For these experiments, only single-pass search was used, i. e. no rescoring of N-best lists with additional models was performed.

The results shown in Table 3 show the effect of the various types of segmentation and punctuation restoration. The label "implicit" refers to the system where punctuation is added implicitly in the translation process, as described in Section 4.2. The labels "source" and "target" name the set-ups, where punctuation is inserted in the source language or in the target language, respectively. The MT systems for these set-ups were trained as described in Sections 4.3 and 4.1, respectively. All MT systems were optimized with respect to the BLEU measure on a development set.

For punctuation prediction either in the source or in the target language we used the `hidden-ngram` tool from the SRI toolkit [12]. We used a 4-gram hidden event language model trained as proposed by the organizers of the IWSLT 2006 evaluation.

When indicated, automatic segmentation of the ASR output was used. As an overall baseline, we used the translation of the correct transcription. There, we have no recognition errors and manual segmentation of the input. In order to separate the effects of ASR errors and segmentation, we aligned the ASR output with the correct transcription (with removed punctuation) using edit distance in order to obtain the original segmentation.

From Table 3 it becomes clear that recognition errors account for the most of the loss in translation quality as compared to the translation of the correct transcription. In contrast, the MT evaluation measures only degrade slightly when automatic segmentation is used and the punctuation is automatically predicted. This shows that the presented approaches to SU boundary detection and punctuation restoration are robust enough to be used in a machine translation framework. The restriction on the maximum sentence length (50 words) allows for efficient translation. On the other hand, the restriction on the minimum sentence length of 3 words helps to avoid breaking apart word groups, for which a good phrasal translation exists. Sentences shorter than 3 words are usually standard expressions like "yes" and "thank you", which are translated accurately even if they become part of a longer segment.

All strategies for predicting punctuation marks work similarly well for this task, with the best translation results yielded by inserting punctuation marks in the source language. This can be explained by the low recognition error rate on this corpus, which makes punctuation prediction in the source language sufficiently reliable.

A preliminary version of the proposed segmentation algorithm was already used by all participants in the 2006 TC-STAR evaluation [8].

For the IWSLT 2006 experiments, the results shown in Table 4 indicate a similar tendency as the results for the TC-STAR task. Errors introduced by automatic speech recognition have a higher impact on the translation scores than the errors introduced from automatic segmentation.

With respect to translation quality, the best performance with punctuation is achieved by implicit prediction using the translation model. This method has the advantage that the performance of the phrase-based translation system is not de-

Table 3: Translation quality for the TC-STAR English-to-Spanish task.

transcription	segmentation	punctuation prediction	BLEU [%]	WER [%]	PER [%]	NIST
correct	correct	manual (source)	45.2	43.3	32.2	9.71
automatic	correct (aligned)	source	37.8	50.6	37.6	8.77
	automatic	source	36.7	51.2	38.1	8.70
		implicit	36.1	51.5	38.6	8.62
		target	36.3	51.3	38.4	8.66
		full stop only (source)	35.8	50.2	38.6	8.70

Table 4: Translation quality for the IWSLT 2006 Chinese-to-English task. All scores are computed case-sensitive with punctuation, as in the official evaluation. The reference translations for the 2006 evaluation data were not available. Therefore, scores using automatic segmentation can only be reported for the development set.

transcription	segmentation	punctuation prediction	BLEU [%]	WER [%]	PER [%]	NIST
DEV 2006						
correct	correct	source	19.8	70.5	54.3	5.99
		implicit	22.0	71.0	53.0	5.86
		target	18.9	70.7	55.2	6.03
	automatic	source	17.3	66.1	54.9	5.34
		implicit	20.7	62.1	52.0	5.41
		target	17.5	67.2	55.9	5.49
automatic	correct	source	15.9	73.9	58.5	5.28
		implicit	19.0	69.1	56.7	5.18
		target	15.4	73.2	58.2	5.37
	automatic	source	14.4	68.4	58.2	4.51
		implicit	17.1	64.8	55.2	4.62
		target	13.8	69.0	59.1	4.60
TEST 2006						
correct	correct	source	18.5	71.7	55.1	5.39
		implicit	21.0	67.1	54.5	5.13
		target	17.7	70.7	55.1	5.38
automatic	correct	source	15.7	73.6	58.8	4.82
		implicit	17.8	70.2	57.8	4.57
		target	15.2	73.6	59.7	4.78

teriorated by falsely inserted punctuation marks in the source side. This is especially important in the IWSLT task, since the corpus is small. Furthermore, the translation quality of the overall system including punctuation prediction is optimized as a whole. On the small task, using the translation model and the target language model in combination to generate punctuation on the target side can improve system performance.

7. Conclusions

We presented a framework for automatic detection of sentence-like units and punctuation prediction in the context of statistical spoken language translation.

The novel sentence segmentation method presented here performed at least as well as the state-of-the-art approaches in terms of precision and recall, but has the advantage that the length of the produced segments can be explicitly controlled and adjusted to the needs of machine translation algorithms. The robustness of the proposed method was also confirmed when evaluating it in terms of the resulting machine translation quality.

For punctuation prediction, we compared three different approaches:

- translating input without punctuation marks followed by punctuation prediction on the resulting translations in a postprocessing step,
- implicitly generating punctuation marks in the translation process, and
- predicting punctuation in the MT input and translating

with an MT system trained on a fully punctuated corpus.

We discussed the advantages and disadvantages of each strategy and performed a contrastive evaluation on two translation tasks. For the large vocabulary task of the TC-STAR English-to-Spanish evaluation, punctuation prediction in the MT input yields best translation quality. For the small vocabulary 2006 IWSLT Chinese-to-English task, implicit generation of punctuation marks leads to superior translation quality.

In the future, we would like to investigate a tighter coupling of automatic SU and punctuation prediction and machine translation by considering "soft" segment boundaries.

8. Acknowledgements

This work was in part funded by the European Union under the integrated project TC-STAR – Technology and Corpora for Speech to Speech Translation (IST-2002-FP6-506738) and is partly based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR0011-06-C-0023. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the DARPA.

9. References

[1] Doddington, G. "Automatic Evaluation of Machine Translation Quality Using N-gram Co-Occurrence Statistics". In *Proc. ARPA Workshop on Human Language Technology*, San Diego, California, March 2002.

[2] Huang, J., and Zweig, G. "Maximum entropy model for punctuation annotation from speech". In *Proc. of ICSLP*, pp. 917-920, 2002.

[3] Kim, J., and Woodland, P. "The use of prosody in a combined system for punctuation generation and speech recognition". In *Proc. of Eurospeech*, pp. 2757-2760, 2001.

[4] Lee, Y., Al-Onaizan, Y., Papineni, K., and Roukos, S. "IBM Spoken Language Translation System". In *Proc. TC-STAR Workshop on Speech-to-Speech Translation*, pp. 13-18, Barcelona, Spain, June 2006.

[5] Liu, Y., Shriberg, E., Stolcke, A., Hillard, D., Ostendorf, M., Peskin, B., and Harper, M. "The ICSI-SRI-UW Metadata Extraction System," *ICSLP 2004, International Conf. on Spoken Language Processing*, Korea, 2004.

[6] Matusov, E., Leusch, G., Bender, O., and Ney, H. "Evaluating Machine Translation Output with Automatic Sentence Segmentation". In *Proc. of IWSLT 2005*, pp. 148-154, Pittsburgh, PA, October 2005.

[7] Matusov, E., Peters, J., Meyer, C., and Ney, H. "Topic Segmentation Using Markov Models on Section Level". In *Proceedings of the 8th IEEE Automatic Speech Recognition and Understanding Workshop (ASRU 2003)*, St. Thomas, Virgin Islands, USA, pp. 471-476, December 2003.

[8] Matusov, E., Zens, R., Vilar, D. Mauser, A., Popovic, M., and Ney, H. "The RWTH Machine Translation System". In *Proc. of The 2006 TC-STAR Workshop on Speech-to-Speech Translation*, pp. 31-36, Barcelona, Spain, June 2006.

[9] Och, F. J., and Ney, H. "Discriminative training and maximum entropy models for statistical machine translation." In *Proc. of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 295-302, Philadelphia, PA, July 2002.

[10] Och, F. J., and Ney, H. "A systematic comparison of various statistical alignment models." *Computational Linguistics*, 29(1):19–51, March 2003.

[11] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proc. of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, Philadelphia, PA, July 2002, pp. 311–318.

[12] Stolcke, A., Shriberg, E., Bates, R., Ostendorf, M., Hakkani, D., Plauche, M., Tür, G. and Lu, Y., "Automatic detection of sentence boundaries and disfluencies based on recognized words," In *Proc. of ICSLP '98, International Conf. on Spoken Language Processing*, pp. 2247-2250, Sidney, Australia, 1998.

[13] Zens, R., and Ney, H. "Improvements in phrase-based statistical machine translation." In *Proc. Human Language Technology Conf. / North American Chapter of the Association for Computational Linguistics Annual Meeting (HLT-NAACL)*, pp. 257-264, Boston, MA, May 2004.