On the Use of Speech Recognition in Computer Assisted Translation∗

L. Rodríguez
Instituto Tecnológico de Informática.
Universidad Politécnica de Valencia. Spain
lrodrig@iti.upv.es

J. Civera, E. Vidal, F. Casacuberta, C. Martínez
Dpto. de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia. Spain
{jcivera,evidal,fcn,cmartine}@dsic.upv.es

Abstract

Computer-Assisted Translation systems can be used by human translators to increase their productivity. In these systems, the computer suggests portions of target sentence that can be accepted or amended by a human translator. In the present work we will introduce speech as a novel way to interact with these systems. The rational behind this approach is to increase the throughput and ergonomy. Different scenarios for this kind of speech interactions will be discussed along with some preliminary results.

1. Introduction

Computer-Assisted Translation (CAT) systems have proved to be a powerful tool for human translators [1]. CAT systems combine the high productivity gain that machine translation (MT) systems offer with the high quality translation achieved by professional translators. Some of such systems can work in an interactive manner in such a way that in each iteration the system provides segments of translated text to the user, who validates the whole segment or a portion of it and adds a suitable continuation [2].

The use of speech in translation environments has been explored in previous works. On the one hand, speech-to-speech translation systems have as input a spoken sentence to be translated [3]. On the other hand, speech can be used to dictate aloud the target sentence to improve the translation quality of a MT system [4]. In this case, the incorporation of speech into the CAT framework allows for a greater productivity in the interaction process.

2. Computer Assisted Translation

The simplest formalisation of a CAT process is as follows: Given a source text s and a fixed prefix of the target sentence t∗, search for a suffix of the target sentence t, that maximises the posterior probability of t, given s and t∗:

\[ \hat{t}_s = \arg \max_{t_s} \Pr(t_s | s, t^∗) . \tag{1} \]

Taking into account that \( \Pr(s, t^∗) \) does not depend on \( t_s \):

\[ \hat{t}_s = \arg \max_{t_s} \Pr(t_s | t^∗, s) . \tag{2} \]

where \( t_s t^∗ \) is the concatenation of the given prefix \( t^∗ \) and a suffix \( t_s \) suggested by the system.

∗This work has been partially supported by the European Union under grant IST-2001-32091 and Spanish project TIC 2003-08681-C02-02.

This joint distribution can be adequately modelled by means of Stochastic Finite-State Transducers (SFST) [5, 3]. In this case, the search is solved by suitable extensions of the Viterbi algorithm. This approach is followed in the CAT system presented in [6].

CAT entails an iterative process where human translator activity is included in the loop. In each iteration, a prefix of the target sentence (\( t^*_p \)) is somehow fixed by the human translator and the CAT system computes Eq. 2 in order to supply its best (or \( N \)-best) translation suffix hypothesis (\( t_s \)) to complete this prefix. Then the human translator has to accept (part of) the suggested suffix \( t_s \) (or one of the \( N \)-best suffixes) and produce some text as a continuation of the accepted part. This text may be free or it may just involve some modifications to the remaining part of the suggested prefix. The previous prefix \( t^*_p \), followed by the accepted part of \( t_s \) and the corresponding user-produced text \( k \), constitute a new prefix \( t^*_p' \). Then, \( t^*_p' \) becomes \( t^*_p \) when applying Eq. 2 in the next iteration to produce a new suggested suffix in the CAT prediction cycle. These ideas were thoroughly explored in the TT2 project [2].

3. Speech Recognition in Computer Assisted Translation

Prior to introducing the integration of speech recognition under the CAT paradigm, it is convenient to introduce a starting point based on previous works. In [4, 7, 8], the framework proposed is as follows: a human translator reads the source text and, rather than typing the target text, he or she dictates the corresponding translation. Consequently, a speech decoding process for the target language is performed, in which knowledge about the source text can be used to attempt reducing recognition errors.

Speech can be used in a CAT system to provide successive interactions between the user and a MT system. Formally, let \( s \) be the source text and \( t^*_p \) a validated prefix of the target sentence. The user is then allowed to utter some words (\( x \)) generally related to the suffix suggested by the system in the previous iteration. This utterance is aimed at accepting or correcting parts of this suffix. It can also be used to add more text as well. Moreover, the user may type some keystrokes (\( k \)) in order to correct (other) parts of this suffix and/or to add more text. Using this information, the system has to suggest a new suffix \( t_s \) as a continuation of the previous prefix, the decoded speech and the typed text. That is, the problem would be to find \( t_s \) given \( s, t^*_p, x \) and \( k \), considering all possible decodings of \( x \) (i.e., letting the decoding of \( x \) be a hidden variable).

According to this very general discussion, it might be assumed that the user can type with independence of the result of the speech decoding process. However, it can be argued that
this generality is not realistically useful in practical situations. Alternatively, it is much more natural that the user waits for a system outcome \( d \) from the spoken utterance, prior to start typing amendments \( k \) to the (remaining part of the previous) system hypothesis. Furthermore, this allows the user to fix possible speech recognition errors in \( d \).

In this more pragmatic and simpler scenario, an alternative problem can be formulated in two steps. The first step is to rely on the source text \( s \) and the previous target prefix \( t_p \), in order to search for a target suffix \( \hat{t}_s \):

\[
i_s = \text{argmax} \Pr(t_s \mid s, t_p) .
\]

This equation exactly corresponds to the CAT scenario discussed in Section 2 (Eq. 1) and it can be approached using the same techniques already mentioned in that section (see, e.g., [9, 10]).

Once \( i_s \) is available, the user can produce some speech, \( x \), and the system has to decode \( x \) into a target sequence of words, \( d \):

\[
d = \text{argmax} \Pr(d \mid s, t_p, i_s, x) .
\]

Finally, the user can enter adequate amendment keystrokes \( k \), if necessary, and produce a new consolidated prefix, \( t_p \), based on the previous \( t_p \), \( d \), \( k \) and parts of \( i_s \). The process will continue in this way until \( t_p \) is completely accepted by the user as a full target text which is an appropriate translation of \( s \).

Since Eq. 3 corresponds to a well-known problem in the CAT framework, we focus now on different manners to approach Eq. 4. To start with, we can write:

\[
d = \text{argmax} \Pr(d \mid s, t_p, i_s, x) \\
= \text{argmax} \Pr(d \mid s, t_p, i_s) \cdot \Pr(x \mid s, t_p, i_s) .
\]

and, by assuming that \( \Pr(x \mid s, t_p, i_s) \) only depends on \( d \):

\[
d = \text{argmax} \Pr(d \mid s, t_p, i_s) \cdot \Pr(x \mid d) .
\]

\( \Pr(x \mid d) \) can be modelled by the acoustic models of the words in \( d \) and \( \Pr(d \mid s, t_p, i_s) \) can be provided by a target language model constrained by the previous prefix \( t_p \), by the source sentence \( s \) and by the suffix \( i_s \) produced at the beginning of the current iteration.

Eq. 6 will be instantiated in four different scenarios depending on the assumptions and constraints adopted in each case. In the next sections, these scenarios will be presented in a least-to-most restricted order. It is important to remark that all the scenarios involve sentence-fragment decoding instead of complete-sentence decoding.

3.1. First Scenario : DEC

DEC is the least constrained setting, involving just conventional speech decoding in the target language:

\[
d = \text{argmax} \Pr(d) \cdot \Pr(x \mid d) .
\]

The language model for \( \Pr(d) \) is implemented as a (smoothed) \( n \)-gram, estimated from the same target sentences used to estimate the translation models for other scenarios. Since the \( n \)-gram is estimated from complete target sentences, but \( x \) is typically an utterance of a sentence fragment, this language model has to be adapted to properly accept any possible subsequence of words. To this end, the initial and final state probabilities in the language model are modified so that the search can start in any \( n \)-gram state and finish in any other state. Obviously, the resulting search space is significantly larger than that of the original \( n \)-gram (it should be noted that this scenario is more complex than simple speech recognition).

3.2. Second Scenario : DEC-PREF

The second scenario is DEC-PREF. In this case, an additional constraint is introduced: the available prefix \( t_p \); again, no information about the source text is used. In this case,

\[
d = \text{argmax} \Pr(d \mid t_p) \cdot \Pr(x \mid d) .
\]

Eq. 8 is similar to Eq. 7, except for the search, which is now constrained to start only in \( n \)-gram states which can be reached using \( t_p \).

3.3. Third Scenario : CAT-PREF

CAT-PREF is the least restricted CAT scenario. It corresponds to a realization of Eq. 6, in which the source sentence, \( s \), the current-iteration prefix, \( t_p \), and a human-translator utterance, \( x \), are available. The goal of the CAT system is to decode \( x \) into an optimal continuation \( d \) of \( t_p \) and to produce a suggested suffix \( \hat{t}_s \) as a continuation of this decoding.

As in Eq. 8, thanks to the restrictions coming from the user-validated target prefix \( t_p \), the language model used for estimating \( \Pr(d \mid s, t_p) \) should exhibit lower perplexity than that involved in Eq. 7. Moreover, here even lower perplexity can be achieved by using information provided by the knowledge that \( t_p, d, \hat{t}_s \) must be a translation of \( s \).

The implementation of this scenario is based on the following equation:

\[
d = \text{argmax} \Pr(d \mid s, t_p) \cdot \Pr(x \mid d) .
\]

This equation is derived from Eq. 6 assuming that \( d \) does not depend on \( t_p \). In this case, the constrained language model for estimating \( \Pr(d \mid s, t_p) \) can be implemented as an adaptation of the target language model used in DEC-PREF based on the information provided by the source sentence \( s \). Each \( n \)-gram probability, \( \Pr(t_n \mid t_{n-1}^{\hat{t}_s}) \), is recalculated according to the product of the original \( n \)-gram probability in DEC-PREF \( (\Pr(t_n \mid t_{n-1}^{\hat{t}_s})) \) and the maximum probability of translating \( t_n \) from any source word in \( s \). This can be seen as a simple interpretation of the inverted alignment models described in [11]. This translation probability is obtained from a stochastic dictionary estimated using a parallel corpus and the GIZA++ toolkit [12].

3.4. Fourth Scenario : CAT-SEL

In this scenario, the human translator can only utter exact prefixes of the suggestion made by the CAT system in the previous iteration \( t_s \). Eq. 6 can be rewritten taking into account this new constraint and assuming that \( d \) only depends on \( t_s \):

\[
d = \text{argmax} \Pr(x \mid d) \cdot \Pr(d \mid t_s) .
\]

In practice, Eq. 10 can be implemented as a search for \( d \) in the (small) set of possible prefixes of the target suffix \( t_s \); that is, \( \Pr(d \mid t_s) \) is estimated by a special finite-state language
model in which only those d that are prefixes of t, have non-null probability.

4. Experimental results

Some experiments have been conducted using the English-Spanish Xerox corpus employed in the TT2 project [2]. In all these experiments a speech decoder based on monophone HMMs was employed. Speech preprocess and feature extraction consisted in speech boundary detection, followed by computing the first ten MEL cepstral coefficients plus the energy, along with the corresponding first and second derivatives [13]. Lexical entries were (automatically) built into deterministic automata from the word-to-phoneme transcription of each entry [13]. Finally, smoothed 3-gram language models (LM) were used in all the scenarios (except CAT-SEL, which requires a special language model. See subsection 3.4).

These experiments were devoted to confirm that the error decoding rate decreases as more constraints are added into the different scenarios.

4.1. Corpus features

The acoustic models for speech recognition were trained using the ALBAYZIN corpus [14]. The features of this corpus are shown in Table 1.

The test data set is composed by a set of utterances of fragments of target-language (Spanish) sentences extracted from the test part of the corpus. All the speech data was acquired using high quality microphones and 16 KHz sampling frequency. A summary of relevant features of this corpus is shown in Tables 2 and 3.

Table 1: Features of the Spanish speech acoustic training corpus (K = ×1.000)

<table>
<thead>
<tr>
<th>Speakers</th>
<th>164</th>
</tr>
</thead>
<tbody>
<tr>
<td>Running words (4 hours)</td>
<td>42K</td>
</tr>
</tbody>
</table>

Table 2: Features of the text English-Spanish translation Xerox Corpus (K = ×1.000)

<table>
<thead>
<tr>
<th>Training-set</th>
<th>Running words</th>
<th>572K</th>
<th>657K</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vocabulary</td>
<td>26K</td>
<td>30K</td>
<td></td>
</tr>
<tr>
<td>Test-set</td>
<td>Running words</td>
<td>7.6K</td>
<td>9.4K</td>
</tr>
<tr>
<td>Running characters</td>
<td>47K</td>
<td>59K</td>
<td></td>
</tr>
<tr>
<td>Perplexity (3-gram)</td>
<td>103</td>
<td>61</td>
<td></td>
</tr>
</tbody>
</table>

Table 3: Spanish speech test utterances (from the XEROX-corpus)

<table>
<thead>
<tr>
<th>Text</th>
<th># of original complete sentences</th>
<th>128</th>
</tr>
</thead>
<tbody>
<tr>
<td># of different sentence fragments uttered</td>
<td>485</td>
<td></td>
</tr>
<tr>
<td>Average uttered fragment length (words)</td>
<td>2.4</td>
<td></td>
</tr>
<tr>
<td>Uuttered fragment length range (words)</td>
<td>[1,13]</td>
<td></td>
</tr>
<tr>
<td>Average prefix length (words)</td>
<td>4.5</td>
<td></td>
</tr>
<tr>
<td>Prefix length range (words)</td>
<td>[0,23]</td>
<td></td>
</tr>
<tr>
<td>Running words</td>
<td>1,138</td>
<td></td>
</tr>
<tr>
<td>Running characters</td>
<td>7,320</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Speech</th>
<th>Number of speakers</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of utterances</td>
<td>5,796</td>
<td></td>
</tr>
<tr>
<td>Running words</td>
<td>13,998</td>
<td></td>
</tr>
</tbody>
</table>

The set of test utterances described in Table 3 was obtained as follows. First a subset of 128 sentence pairs was selected from the XEROX text corpus [2] of printer manuals. For the target (Spanish) sentence of each of these pairs, several segmentations into prefixes and suffixes were randomly performed and, for each generated suffix, a set of prefixes was randomly derived. All the prefixes of suffixes generated in this way constitute the set of sentence fragments uttered by several speakers. An example of this process is shown in Table 4.

4.2. Evaluation measures

Speech decoding accuracy was assessed in terms of conventional measures:

- **Sentence Error Rate (SER)**: Number of sentences that are incorrectly recognised, divided by the total number of sentences. In this framework, SER can be understood as the number of times that user has to make a correction on the decoded utterance.

- **Word Error Rate (WER)**: Minimum number of word substitution, deletion and insertion operations needed to convert a full target sentence provided by a MT system into the corresponding reference translation, divided by the total number of words [5, 15].

In addition to these speech metrics, a translation measure to assess the performance of the translation models has been considered:

- **Translation Word Error Rate (TWER)**: Minimum number of word substitution, deletion and insertion operations needed to convert a full target sentence provided by a MT system into the corresponding reference translation, divided by the total number of words in the reference translation [5, 15].

4.3. Results

The results of Table 5 show that increasing speech recognition performance is achieved as the language models become more constrained. If only prefix-derived constraints are added to DEC, a modest improvement of 2.5 points of WER and 5.8 points of SER is obtained in DEC-PREF. By further including constraints derived from the source text, a more significant improvement is achieved in CAT-PREF with respect to DEC-PREF: 5.5 points of WER and 14.4 points of SER. It is necessary to remark that the result that can be achieved in this scenario heavily depends on the quality of the underlying translation models employed and also on the difficult of the translation task. Regarding the task, it is important to mention that the corpus employed here (Table 3) is a subset of a full bilingual corpora (see Table 2) already used in several machine translation experiments. This subset has been turned up to be a specially difficult corpus (in comparison with the full one) as can be seen in the results shown in Table 6.

The constraints added in the last scenario, CAT-SEL, are derived from the (simulated) suggestions of the CAT system. In this case, the improvement is very important: 9.0 points of WER and 26.4 points of SER with respect to the previous scenario (CAT-PREF).
Table 5: Speech decoding results (in %) for different scenarios

<table>
<thead>
<tr>
<th></th>
<th>WER</th>
<th>SER</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEC</td>
<td>18.6</td>
<td>50.2</td>
</tr>
<tr>
<td>DEC-PREF</td>
<td>16.1</td>
<td>44.4</td>
</tr>
<tr>
<td>CAT-PREF</td>
<td>10.6</td>
<td>30.0</td>
</tr>
<tr>
<td>CAT-SEL</td>
<td>1.6</td>
<td>3.6</td>
</tr>
</tbody>
</table>

These results clearly suggest that using knowledge about the source sentence is more important than using only user-validated prefixes.

Table 6: MT performance on the English-Spanish full test set of Table 2 and the small complete-sentences subset of Table 3

<table>
<thead>
<tr>
<th></th>
<th>TWER (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FULL</td>
<td>42.7</td>
</tr>
<tr>
<td>SMALL</td>
<td>57.4</td>
</tr>
</tbody>
</table>

The decoding computational demands of the different systems is also worth mentioning. A dramatic reduction of memory and computing time is observed from DEC to DEC-PREF and the response time is more than halved from DEC-PREF to CAT-PREF. Finally CAT-SEL only requires very light computing, which makes it easy to implement this kind of speech-enabled CAT systems on low-end desktop computers.

5. Conclusions

In this paper, novel approaches on the use of speech in a CAT system are presented. Speech has proved to be useful in a CAT system if the performance of decoding is high enough. Such performance can be achieved by using sufficiently constrained target language models. Different approaches to achieve this goal have been assessed in an English to Spanish translation task of printer manuals. Sufficiently good speech decoding performance has been achieved at least in one scenario, which entails significant potential savings of human effort with respect to typing the whole target text or having to use non-speech cursor-positioning actions. This good performance (both speed and accuracy) is achieved thanks to search constraints derived from both the source sentence (through a translation model) and the successively consolidated prefixes of the target text.

The way in which the translation restrictions can be included into the speech decoding system is still an open problem. We have explored some elementary techniques, but there are many other alternatives that should have to be considered in future works.

6. References