Improving Speech-to-Speech Translation Using Word Posterior Probabilities

Vicente Alabau, **Alberto Sanchis**, **Francisco Casacuberta**

Departament de Sistemes Informàtics i Computació
Universitat Politècnica de València
46071 València, SPAIN
{valabau,josanna,fcn}@dsic.upv.es

Abstract

Nowadays, speech translation is a research problem in machine translation. The problem arises as to how to combine speech recognition and machine translation in a suitable way. Some authors have shown that the speech translation can be improved by using word lattices as input of the translation system. The acoustic recognition scores from the word lattice are used for improving the translation quality. However, word lattices do not consider word co-occurrences between different hypothesis and those probabilities are not real probabilities but merely Viterbi approximations. In this work, we propose an improved word lattice representation for using posterior probabilities instead of acoustic scores. We present preliminary results of this approach compared against other common approaches on two different corpora. Although the results are not strongly conclusive, they show that this approach is worth exploring more deeply.

1 Introduction

Within the framework of speech-input translation, we are faced with the problem of integrating the speech recognition and the translation processes. The problem arises as to how to combine these two processes in a suitable way.

Different approaches to speech input translation have been investigated (E.Vidal, 1997; Ney, 1999; F. Casacuberta et al., 2004; Garca-Varea et al., 2004). The most simple approach performs the two processes in a serial manner: first, an input utterance is decoded into a sentence using a conventional automatic speech recognizer (ASR), and after, this sentence is translated using a *text-to-text* translator. The main drawback of this approach is that the output of an ASR can contain misrecognized words and, consequently, the quality of the translated sentences decrease. In order to circumvent this problem, different solutions have been proposed (Quan, 2005; E. Matusov and Ney, 2005; Bertoldi and Federico, 2005). In (Quan, 2005) N-best lists have been used for improving the quality of the translated sentences. In (E. Matusov and Ney, 2005) the translation process is performed using as input a word lattice and acoustic recognition scores. In (Bertoldi and Federico, 2005) the translation process is performed using confusion networks and posterior probabilities. All these approaches try to exploit the use of a set of the most probable hypotheses instead of only the best one.

The use of stochastic finite-state transducers provides a fully integrated recognition-translation architecture in which the source and target sentences are obtained simultaneously (E.Vidal, 1997; F. Casacuberta et al., 2004). However, the experimental results are not consistently better than serial approach (F. Casacuberta et al., 2004).

In this work, we propose the use of an improved word lattice representation for speech-to-speech translation following a semi-coupled architecture. Instead of using acoustic recognition scores (E. Matusov and Ney, 2005) we use the word posterior probabilities computed over the word lattice for improving the quality of the translated sentences.

2 Speech Input Translation: Review

In this section, a review of the formulation defined in (F. Casacuberta et al., 2004) is presented.

The problem of speech-input statistical translation can be formulated as:

$$\hat{e}_1^{\hat{I}} = \arg\max_{I,e_1^I} Pr(e_1^I|\boldsymbol{x}_1^T) \quad (1)$$

where $\boldsymbol{x}_1^T$ is the acoustic vectors and $\hat{e}_1^{\hat{I}}$ is the most probable translation of the speech utterance. The maximization is performed over all possible target sentences e_1^I and all possible lengths I.

The process can be stated as: $\boldsymbol{x}_1^T \rightarrow f_1^J \rightarrow e_1^I$ where f_1^J is the input decoding of $\boldsymbol{x}_1^T$, and e_1^I is the corresponding translation of f_1^J. Consequently, Eq. 1 can be decomposed by:

$$\arg\max_{I,e_1^I} Pr(e_1^I|\boldsymbol{x}_1^T) = \arg\max_{I,e_1^I} \sum_{f_1^J} Pr(e_1^I, f_1^J|\boldsymbol{x}_1^T) \quad (2)$$

with the practical assumption that $Pr(x_1^T|e_1^I, f_1^J)$ does not depend on the target sentence e_1^I, Eq. 2 can be decomposed by:

$$\arg\max_{I,e_1^I} Pr(e_1^I|\boldsymbol{x}_1^T) = \arg\max_{I,e_1^I} \sum_{f_1^J} Pr(e_1^I, f_1^J) Pr(\boldsymbol{x}_1^T|f_1^J) \quad (3)$$

We approximate the sum over all possible source language sentences by the maximum. The purpose is to associate a source sentence to the input utterance whose translation is the target sentence searched for. From Eq. 3,

$$\arg\max_{I,e_1^I} Pr(e_1^I|\boldsymbol{x}_1^T) \approx \arg\max_{I,e_1^I} \max_{f_1^J} Pr(e_1^I, f_1^J) Pr(\boldsymbol{x}_1^T|f_1^J) \quad (4)$$

$Pr(e_1^I, f_1^J)$ refers to the translation model and $Pr(\boldsymbol{x}_1^T|f_1^J)$ is modeled by acoustic models (typically *Hidden Markov Models* (HMM)).

3 Speech-to-Speech Translation Based on Posterior Probabilities

In this section we explain how the probabilities involved in speech input translation (Eq. 4) are estimated and the decoding algorithm.

3.1 Posterior Probabilities on Word Lattices

A word lattice G is a directed, acyclic, weighted graph. The nodes correspond to discrete points in time. The edges are triplets $[w,\ s,\ e]$, where w is the hypothesized word from node s to node e. The weights are the acoustic recognition scores associated to the word lattice edges. Any path from the initial to the final node forms a hypothesis f_1^J.

Given the acoustic observations $\boldsymbol{x}_1^T$, the posterior probability for a specific word (edge) $[w,\ s,\ e]$ can be computed by summing up the posterior probabilities of all hypotheses of the word lattice containing the edge $[w,\ s,\ e]$:

$$P([w,\ s,\ e] \mid \boldsymbol{x}_1^T) = \frac{1}{P(\boldsymbol{x}_1^T)} \sum_{\substack{f_1^J \in G: \\ \exists [w', s', e']: \\ w'=w, s'=s, e'=e}} P(f_1^J,\ \boldsymbol{x}_1^T) \quad (5)$$

The probability of the sequence of acoustic observations $P(\boldsymbol{x}_1^T)$ can be computed by summing up the posterior probabilities of all word lattice hypotheses:

$$P(\boldsymbol{x}_1^T) = \sum_{f_1^J \in G} P(f_1^J, \boldsymbol{x}_1^T) \quad (6)$$

These posterior probabilities can be efficiently computed based on the well-known *forward-backward* algorithm (Wessel et al., 2001).

The posterior probability defined in Eq. 5 does not perform well because of a word w can occur with slightly different starting and ending times. This effect is represented in the word lattice by different word lattice edges and the posterior probability mass of the word is scattered among the different word segmentations (see Figure 1).

To deal with this problem, we have considered a method proposed in (Wessel et al., 2001). Given a specific word (edge) $[w,\ s,\ e]$ and a specific point in time $t \in [s, e]$, we compute the posterior probability of the word w at time t by summing up the posterior probabilities of the word lattice edges $[w, s', e']$ with identical word w and for which t is within the interval time $[s', e']$:

$$P_t([w,\ s,\ e] \mid \boldsymbol{x}_1^T) = \sum_{t \in [s', e']} P([w,\ s',\ e'] \mid \boldsymbol{x}_1^T) \quad (7)$$

Based on Eq. 7, the posterior probability for a specific word $[w,\ s,\ e]$ is computed as the maximum of the frame time posterior probabilities:

$$P([w,\ s,\ e] \mid \boldsymbol{x}_1^T) = \max_{s \le t \le e} P_t([w,\ s,\ e] \mid \boldsymbol{x}_1^T) \quad (8)$$

The probability computed by Eq. 8 is in the interval $[0, 1]$ since, by definition, the sum of the word posterior probabilities for a specific point in time must sum to one (this property can be appreciated in the Figure 1). Figure 1 shows an example of the word lattice with the word posterior probabilities computed following the Eq. 5. Figure 2 shows the same word lattice after Eq. 8 is computed.

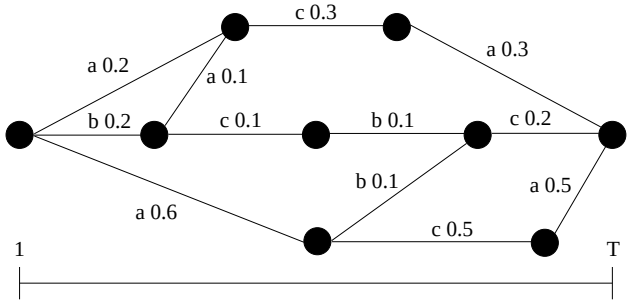

Figure 1: Word lattice with the word posterior probabilities.

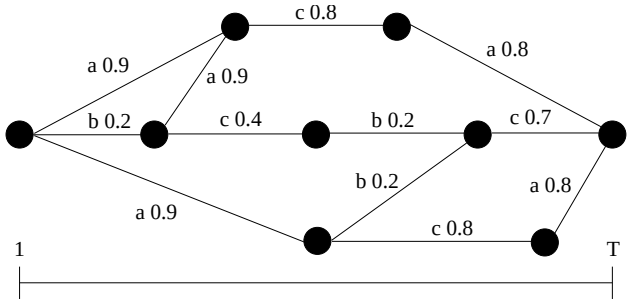

Figure 2: Word lattice after the frame posterior probabilities are computed.

We will use these posterior probabilities to compute the conditional probability of the acoustic signal given a source hypothesis:

$$Pr(\boldsymbol{x}_1^T|f_1^J) = \prod_{j=1}^{J} P([w_j, s_j, e_j]|\boldsymbol{x}_1^T) \quad (9)$$

where s_j and e_j are the starting and the ending time, respectively, of the source word w_j.

3.2 Stochastic Finite-State Transducers

The joint probability distribution $Pr(e_1^I, f_1^J)$ in Eq. 4 can be adequately modelled by means of a statistical finite-state transducer (SFST). SFSTs have been thoroughly studied (Vidal et al., 2005a; Vidal et al., 2005b) and several approaches to infer SFSTs from corpora have been proposed in recent years (Kumar et al., 2005; Casacuberta and Vidal, 2004; Allauzen et al., 2004).

We have used the Grammatical Inference and Alignments for Transducer Inference (GIATI) technique for inferring the SFST (Picó, 2005). This technique uses a finite sample of bilingual pairs (parallel corpus) for inferring the SFST in three steps:

1. Building training strings. Each training pair is transformed into a single string from an extended alphabet to obtain a new sample of strings. The transformation of a parallel corpus into a corpus of single sentences is performed with the help of statistical alignments: each word (or substring) is joined with its translation in the output sentence, creating an *extended* symbol.

2. Inferring a (stochastic) regular grammar. Typically, a smoothed n-gram is inferred from the sample of strings obtained in the previous step.

3. Transforming the inferred regular grammar into a transducer. The symbols associated to the grammar rules are transformed into source/target symbols by applying an adequate transformation.

An interesting feature of SFSTs is that the maximization of Eq. 4 can be performed in a fully integrated recognition-translation manner. This is possible since each transition of the SFST is labelled with a source word and its corresponding target translation. Thus, each transition is expanded by the acoustical representation of the source words. Following the standard speech recognition searching algorithm over the SFST, the optimal source and target sentences are obtained simultaneously.

We use the SFST in a semi-coupled manner as is explained in the next section. Comparative results between semi-coupled and integrated architectures are presented in Section 4.

3.3 Decoding Algorithm

The decoding algorithm is composed of three major steps:

- *Speech Recognition step*: In this step we perform the recognition of the speech utterance using a conventional speech recognizer. This is done by searching for a sequence of source words f_1^J such that:

$$\hat{f}_1^{\hat{J}} \approx \arg\max_{f_1^J} Pr(f_1^J) Pr(\boldsymbol{x}_1^T | f_1^J)$$

  where $Pr(\boldsymbol{x}_1^T | f_1^J)$ is modelled by HMMs and $Pr(f_1^J)$ by an input language model.

  The output of this step is also a word lattice G which represents the most probable hypotheses.

- *Compute the word posterior probabilities*: Each edge $[w, s, e]$ in the lattice G is scored with the word posterior probability following Eq. 8.

- *Translation step*: In this step, we use Eq. 4 to obtain the most probable target sentence. The maximization for each input sentence f_1^J, is only computed on a subset of possible f_1^J, i.e. those belonging to the word lattice G and the SFST. The search algorithm process corresponds to the following equation:

$$\hat{e}_1^{\hat{I}} \approx \arg\max_{I, e_1^I} \max_{f_1^J \in G \cap SFST} Pr(e_1^I, f_1^J) Pr(\boldsymbol{x}_1^T | f_1^J) \quad (10)$$

  $Pr(e_1^I, f_1^J)$ is modelled by the SFST and $Pr(\boldsymbol{x}_1^T | f_1^J)$ is computed using the word posterior probabilities of the word lattice G following Eq. 9.

The search process is performed on a plane of a 3-D Viterbi search over the word lattice and the SFST. A third dimension considered as the length of the path is necessary, since two hypotheses that arrive at the same point in the search process, with different lengths, must be treated as two possible solutions; thus we need to define the best single 3-D path as the sequence of states which maximizes the Eq. 10.

4 Experimental Results

4.1 Corpora

This section is devoted to evaluate and analyze the approach described in this paper and compare it against the standard approaches. To do that, a series of experiments were run on two different tasks of increasing complexity.

The Eutrans-I task (F. Casacuberta et al., 2004), the simpler of them, is composed of pairs of Spanish-English sentences that represent the translation of queries, requests and complains made by telephone to the front desk of a hotel. The sentences were semi-automatically generated from a series of travel booklets and, as a consequence, the variety of expressions is limited, which is reflected in a low perplexity. With regard to the acoustic models, 26 context-independent phones were trained with the HTK Toolkit (Young et al., 1997). The speech corpus amounted to approximately 3.8 hours of 8 kHz telephone signal. A back-off 4-gram GIATI model was used as translator and a back-off trigram as input language model.

On the other hand, the Eutrans-II FUB task (Italian-English) is significantly more complex and closer to a real situation than the Eutrans-I task. Although both corpora represent the same scenario, in this case the speech corpus consisted of acquisitions of real phone calls to the front desk of a hotel. Thus, this corpus is highly spontaneous and contains many non-speech artifacts. The corpus was obtained by manually transcribing the acquired Italian utterances and translating them into corresponding English sentences. The speech training corpus consists of 7.9h of microphone speech. 1500 context-dependent models were trained and the LDA technique was used to improve the feature representation of the speech signal. We used a back-off trigram as input language model. For the GIATI model we estimated a back-off bigram.

Statistics of these corpora are summarized on Table 1.

		Eutrans-I		FUB	
		Spanish	English	Italian	English
Train	Sentences	10000		3038	
	# words	132198	134922	61232	72446
	Vocabulary	686	513	2459	1701
Test	Sentences	336		278	
	# words	2828	2940	5381	6198
	Perplexity	8.6	6.3	31	25
	ASR WER	13.3	–	29.4	–

Table 1: *Corpus statistics for Eutrans-I and FUB.*

4.2 Evaluation Measures

The experiments have been assessed through several different evaluation measures. To measure the translation performance, we have used Translation Word Error Rate (TWER) and Position-Independent Word Error Rate (PER), which inherit from speech WER, as they have been the traditional translation measures. Nevertheless, new measures have arisen recently: BLEU (Papineni et al., 2001) and NIST (Doddington, 2002). Both measure the n-gram co-occurrences and are said to be well correlated with human evaluation. We also present the results with these measures because currently they are the most used in machine translation. In all the experiments, punctuation marks have been removed, while capital letters have been kept.

4.3 Parameter Tunning

One important question regarding performance is to adjust the model parameters. In our case, given that probabilities in Eq. 10 are not true distributions, it is necessary to find an interpolation lambda parameter for the word lattice probability that minimizes the translation error. This adjustment is typically carried out by means of a development set. However, we have optimized this lambda parameter over the test-set for both corpus since they do not have development test. In the experiments, we show the results for the lambda for which the higher BLEU is achieved.

It has been shown that translation accuracy depends directly on the density of the word lattices (E. Matusov and Ney, 2005). The density of a word lattice depends on a parameter k which restricts the maximum history length of the states. For the FUB task it was not necessary to use this parameter and the word lattices were generated with the highest density possible. However, the Eutrans-I task required to set the k parameter to 10 due to computational requirements.

4.4 Experiments

Depending on how $Pr(\boldsymbol{x}_1^T|f_1^J)$ was estimated, two different sets of speech-to-speech translation experiments were run for both corpus. For the first one (labelled as *ac*), we used the acoustic recognition scores of the word lattice as it is proposed in (E. Matusov and Ney, 2005). For the second one (labelled as *post*), we used the posterior probabilities computed as it has been proposed in Section 3.1.

Three different architectures have been evaluated:

- The serial approach (serial). Using the Viterbi algorithm, the best sentences were obtained from the word lattice using the acoustic recognition scores and posterior probabilities. When the acoustic recognition scores were used, the output was the same as the output in the speech recognizer. Subsequently, the best sentence was translated.

- The semi-coupled approach (lattice). The word lattice was translated as it is explained in Section 3.3 using the acoustic recognition scores and posterior probabilities.

- The integrated approach (integrated). A word lattice was obtained from the speech input translation using the SFST in a fully integrated manner. As in the serial approach, a Viterbi algorithm was applied to compute the best sentences from the word lattice using the acoustic recognition scores and the posterior probabilities.

Furthermore, the correct transcriptions were also translated to compute the baseline translation error of the SFST used in all the experiments.

4.5 Eutrans-I Spanish-English

The translation results for the Eutrans-I corpus are given in Table 2. This is an easy task as can be observed by the high baseline BLEU score. Furthermore, the integrated approach (approach *A*) outperforms the rest of architectures in most of the measures. This fact is due to the good ratio between amount of training data and complexity of the task, which allows good parameter estimates.

However, word lattice decoding with posterior probabilities (approach *B*) obtains the best NIST score. A possible explanation of this inconsistency is that the average output length in approach *B* is 1.45% shorter than sentences in approach *A*. Although it does not seems a huge difference, the brevity penalty (BP) factor in the BLEU measure changes dramatically the results. In fact, the best BLEU score without BP found in the approach *A* is 85.9 while for approach *B* is 86.6. It has been observed in (Koehn, 2004; Doddington, 2002) that the BLEU measure heavily penalizes short sentences. As a matter of fact, in (Doddington, 2002) it is stated that NIST is more stable than BLEU regarding the length of the sentences. Thus, we attribute this bad results in BLEU to the fact that our translation system does not perform any kind of output brevity penalization and, therefore, the BLEU scores are affected.

It is also important to note that the results presented in this table were obtained with downsized word lattices. Despite the fact that all the experiments with lattices were affected by this restriction, it must be noted that posterior probabilities were specially affected since the probabilities of the observed hypothesis were worse estimated.

4.6 FUB Italian-English

Table 3 shows the translation results for the FUB Italian-English corpus. In this case, word lattice with posterior probabilities (approach *A*) outperformed the rest of approaches except for the BLUE measure, for which word lattice with

architec.	score	TWER	PER	NIST	BLEU
baseline	correct	4.8	4.8	9.53	94.0
serial	ac	14.1	13.7	8.29	81.3
	post	14.3	14.1	8.38	81.0
lattice	ac	13.3	13.1	8.32	81.7
	post	13.4	13.2	**8.43**	81.9
integrated	ac	**12.6**	**12.3**	8.37	**82.6**
	post	13.7	13.4	8.29	81.1

Table 2: *Eutrans-I Translation results (TWER in %) for different architectures.* ac *in column* score *indicates that acoustic scores were used.* post *indicates that posterior probabilities were used.*

acoustic scores (approach *B*) obtained the best result. If we compare in detail both approaches, we will notice that the output sentences in approach *A* are 4.1% shorter respect to approach *B*. As in the previous task, if we compare BLEU scores without BP, approach *A* scores higher (52.1 vs 51.3)

On the other hand, the posterior probabilities consistently outperform the acoustic scores for all the architectures except for the BLEU measure in this task. It may be due in part to the fact that we used the whole lattice in all the experiments. Therefore, posterior probabilities were better estimated and, as a consequence, the results against acoustic scores were improved. Besides, this task is more complex so the distribution probabilities are further from the real distributions. Hence, posterior probabilities take more advantage of the word lattice information.

architecture	score	TWER	PER	NIST	BLEU
baseline	correct	27.7	22.0	8.44	59.3
serial	ac	43.8	34.9	6.62	45.0
	post	40.3	31.7	7.04	46.3
lattice	ac	39.2	31.1	7.11	**49.4**
	post	**38.3**	**30.3**	**7.27**	48.3
integrated	ac	44.7	35.4	6.55	44.5
	post	43.3	34.1	6.74	43.7

Table 3: *FUB Translation results (TWER in %) for different architectures.* ac *in column* score *indicates that acoustic scores were used.* post *indicates that posterior probabilities were used.*

5 Conclusions

In this paper, we have shown a new approach to integrate speech and translation by using an improved word lattice representation. We propose the use of word posterior probabilities computed over the word lattice for improving speech input translation. We have test our system on two tasks and the results show that this approach can improve the translation process.

However, the results are not fully consistent due to an inconsistency in the BLEU scores. This may be provoked by the brevity penalty used in BLEU. Therefore, it is imperative to add a brevity penalization to our transducer in order to overcome this situation and to assure that the results are definitely conclusive.

Furthermore, a beam search decoder is needed that can handle large lattices which could benefit from better estimated posterior probabilities.

Acknowledgements

Work supported by the "Agència Valenciana de Ciència i Tecnologia" under grant GRUPOS03/031, the EC (FEDER), the Spanish MEC under grant TIN2006-15694-CO2-01, and the "Programa d'Incentiu a la Investigació 2004 UPV".

References

Allauzen, C., Mohri, M., Riley, M., and Roark, B. (2004). A generalized construction of integrated speech recognition transducers. In *IEEE International Conference on Acoustic, Speech and Signal Processing*, volume 1, pages 761–764. IEEE Press.

Bertoldi, N. and Federico, M. (2005). A new decoder for spoken language translation based on confusion networks. In *IEEE Automatic Speech Recognition and Understanding Workshop*.

Casacuberta, F. and Vidal, E. (2004). Machine translation with inferred stochastic finite-state transducers. *Computational Linguistics*, 30(2):205–225.

Doddington, G. (2002). Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the the second international conference on human language technology research (HLT 2002)*, pages 138–145, San Diego, California.

E. Matusov, S. K. and Ney, H. (2005). On the integration of speech recognition and statistical machine translation. In *9th European Conference on Speech Communication and Technology, Interspeech*, pages 3177–3180, Lisbon, Portugal.

E.Vidal (1997). Finite-State Speech-to-Speech Translation. In *Int. Conf. on Acoustics Speech and Signal Processing, Vol.1*, pages 111–114, Munich.

F. Casacuberta, H. N., Och, F., Vidal, E., Vilar, J., Barrachina, S., Garcia-Varea, I., D. Llorens, C. M., Molau, S., Nevado, F., Pastor, M., Pico, D., and Sanchis., A. (2004). Some approaches to statistical and finite-state speech-to-speech translation. *Computer Speech and Language*, 18:25–47.

Garca-Varea, I., Sanchis, A., and Casacuberta, F. (2004). A decoding algorithm for speech input statistical translation. In *Text, Speech and Dialogue: Proceedings of the 7th International Conference (TSD 2004)*, volume 3206 of *Lecture Notes in Computer Science*, pages 305–314. Springer-Verlag, Brno, Czech Repiblic.

Koehn, P. (2004). Statistical significance tests for machine translation evaluation. In *EMNLP*, Barcelona, Spain.

Kumar, S., Deng, Y., and Byrne, W. (2005). A weighted finite state transducer translation template model for statistical machine translation. *Journal of Natural Language Engineering*, 12(1):35–75.

Ney, H. (1999). Speech Translation: Coupling of Recognition and Translation. In *Int. Conf. on Acoustics Speech and Signal Processing*, pages 1149–1152, Phoenix.

Papineni, K., Roukos, S., Ward, T., and Zhu, W. (2001). Bleu: a method for automatic evaluation of machine translation. Technical Report RC22176, Thomas J. Watson Research Center.

Picó, D. (2005). *Combining Statistical and Finite-State Methods for Machine Translation*. Tesis doctoral en informtica, Departamento de Sistemas Informticos y Computacin, Universidad Politcnica de Valencia.

Quan, V. (2005). Integrated n-best re-ranking for spoken language translation. In *9th European Conference on Speech Communication and Technology, Interspeech*, Lisbon, Portugal.

Vidal, E., Thollard, F., C. de la Higuera, F. C., and Carrasco, R. (2005a). Probabilistic finite-state machines - part I. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(7):1013–1025.

Vidal, E., Thollard, F., C. de la Higuera, F. C., and Carrasco, R. (2005b). Probabilistic finite-state machines - part II. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(7):1025–1039.

Wessel, F., Schluter, R., Macherey, K., and Ney, H. (2001). Confidence measures for large vocabulary continuous speech recognition. *IEEE Trans. Speech and Audio Processing*, 9(3).

Young, S., Odell, J., Ollason, D., Valtchev, V., and Woodland, P. (1997). *The HTK Book (Version 2.1)*. Cambridge University Department and Entropic Research Laboratories Inc.