Using Multiple Recognition Hypotheses to Improve Speech Translation

Ruiqiang Zhang, Genichiro Kikui, Hirofumi Yamamoto

ATR Spoken Language Translation Research Laboratories
2-2 Hikaridai, Seiika-cho, Soraku-gun, Kyoto, 619-0288, Japan
{ruiqiang.zhang, genichiro.kikui, hirofumi.yamamoto}@atr.jp

Abstract

This paper describes our recent work on integrating speech recognition and machine translation for improving speech translation performance. Two approaches are applied and their performance are evaluated in the workshop of IWSLT 2005. The first is direct N-best hypothesis translation, and the second, a pseudo-lattice decoding algorithm for translating word lattice, can dramatically reduce computation cost incurred by the first approach. We found in the experiments that both of these approaches could improve speech translation significantly.

1. Introduction

At least two components are involved in speech to speech translation: automatic speech recognizer and machine translation. Unlike plain text translation, the performance of speech translation may be degraded due to the speech recognition errors.

Several approaches have been proposed to compensate for the loss of recognition accuracy. [1] proposed N-best recognition hypothesis translation, which translates all the top N hypotheses and then outputs the highest scored translations by ways of weighing all the translations using a log-linear model. [2] used word lattices to improve translations. [3] used finite state transducers (FST) to convey the features from acoustic analysis and source target translation models. All these approaches realized an integration between speech recognition modules and machine translation modules so that information from speech recognition, such as acoustic model score and language model score, can be exploited in the translation module to achieve the maximum performance over the single-best translation.

In the field of machine translation, the phrase-based statistical machine translation approach is widely accepted at present. The related literature can be found in [4] [5]. But previously, word-based statistical machine translation, pioneered by IBM Models 1 to 5 [6], were used widely. In the evaluation, we used both the word-based and phrase-based systems. However, the purpose of this work is not to compare performance of word-based with phrase-based translation. We used two system for different translations. The phrase-based SMT is used in Chinese-English translation while the word-based SMT is used in Japanese-English translation.

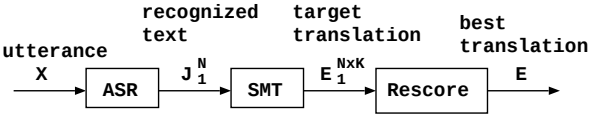

Figure 1: N-best hypothesis translation

In this paper we describe two speech translation structures. The first is a direct N-best hypothesis translation system that uses a text-based machine translation engine to translate each of the hypotheses, and the results are rescored by a log-linear model. The second is a pseudo-lattice translation system, merging the N-best hypotheses into a compact pseudo-lattice which serves as an input to our proposed decoding algorithm for lattice translation. This algorithm runs much faster than the first approach.

In the following, Section 2 describes the direct N-best hypothesis translation. Section 3 describes the pseudo-lattice translation. Section 4 introduces the experimental process and translation results in the evaluation of IWSLT2005. Section 5 presents our conclusions concerning the techniques, and some final remarks are given.

2. Direct N-best hypothesis translation

The structure of the direct N-best hypothesis translation is illustrated in Fig. 1, where there are three modules, an automatic speech recognizer(ASR), a statistical machine translation(SMT), and a log-linear model rescore(Rescore). This structure is used in Chinese to English translation in the evaluation.

2.1. ASR: automatic speech recognition

ASR functions as a decoder to retrieve the source transcript from input speech. The input is a speech signal, X. The output is a source sentence, J. The mechanism of ASR is based on HMM pattern recognition. The acoustic models and language models of the source language are required in the decoding. Because speech recognition errors are unavoidable, ASR outputs multiple hypotheses, the top N-best, to increase the accuracy.

2.2. SMT: statistical machine translation

The SMT module is to translate the source language, J, into target language, E. A phrase-based statistical machine translation decoder was used in the evaluation. The decoding process is carried out in three steps: First, a word graph is created by beam-search where phrase translation models and trigram models are used to extend beams. Second, A* search is used to find the top N-best paths in the word graph. Finally, long-range(> 3) language models are used to rescore the N-best candidates and output the best one.

In order to collect source-target translation pairs, we used GIZA++ to do bi-directional alignment, similar to [5]. In one direction alignment, one source word is aligned to multiple target words; In the other direction, one target word is aligned to multiple source words. Finally, the bi-directional alignment are merged and the phrase pairs are extracted from the overlapping alignments.

The translation probability of translation pairs were computed by relative frequency, counting the co-occurrences of the pairs in the training data.

2.3. Rescoring: log-linear model rescoring

Loglinear models are applied to rescore the translations which are produced by SMT. The model integrates features from both ASR and SMT. We used three features from ASR and 10 features from SMT.

The log-linear model used in our speech translation process, $P(E|X)$, is

$$P_\Lambda(E|X) = \frac{exp(\sum_{i=1}^{M} \lambda_i f_i(X,E))}{\sum_{E'} exp(\sum_{i=1}^{M} \lambda_i f_i(X,E'))} \quad \Lambda = \{\lambda_1^M\} \tag{1}$$

Features from ASR include acoustic model score, source language model score, and posterior probability calculated as below.

$$\frac{P(X|J_k)P(J_k)}{\sum_{J_i} P(X|J_i)P(J_i)} \tag{2}$$

Features from SMT include target word language model score, class language model score, target phrase language model, phrase translation model, distortion model, length model (defined as the number of words in the target), deletion model (defined as the NULL word alignment), lexicon model (obtained from GIZA++), and size model (representing the size of jump between two phrases.)

For the optimal value of λ, our goal is to minimize the translation "distortion" between the reference translations, $\mathcal{R}$, and the translated sentences, $\widehat{\mathcal{E}}$.

$$\lambda_1^M = optimize \;\; \mathcal{D}(\widehat{\mathcal{E}}, \mathcal{R}) \tag{3}$$

where $\widehat{\mathcal{E}} = \{\widehat{E}_1, \cdots, \widehat{E}_L\}$ is a set of translations of all utterances. The translation $\widehat{E}_l$ of the l-th utterance is produced by Eq. 1.

Let $\mathcal{R} = \{R_1, \cdots, R_L\}$ be the set of translation references for all utterances. Human translators paraphrased 16 reference sentences for each utterance, i.e., R_l contains 16 reference candidates for the l-th utterance.

$\mathcal{D}(\widehat{\mathcal{E}}, \mathcal{R})$ is a translation "distortion", that is, an objective translation assessment. A basket of automatic evaluation metrics can be used, such as BLEU, NIST, mWER, mPER and GTM.

Because the distortion function, $\mathcal{D}(\widehat{\mathcal{E}}, \mathcal{R})$, is not a smoothed function, we used *Powell*'s search method to find a solution [7].

The experimental results in [1] have shown that minimizing the translation distortion in development data is an effective method to improve translation qualities of test data.

3. Pseudo-lattice translation

The N-best hypothesis translation improved speech translation significantly, as found in [1]. However, the approach is inefficient, computationally expensive and time consuming.

We proposed a new decoding algorithm, pseudo-lattice decoding, to improve on the direct N-best translation. This approach can also translate the N-best hypotheses, and the processing time is shorten dramastically because the same word IDs appearing in the N-best hypotheses are translated fewer times than the direct N-best translation.

We start from the word lattice minimization produced by ASR to describe the approach.

3.1. Minimizing the source word lattice(SWL)

Because we use HMM-based ASR to generate the raw source word lattice(SWL), the same word ID can be recognized repeatedly in slightly different frames. As a result, the same word ID may appear in multiple edges in the SWL. Hence, when N-best hypotheses are generated from the word lattice, the same word ids may appear in multiple hypotheses.

Fig. 2 shows an example of lattice downsizing. The word IDs are shown in the parentheses. We use the following steps to minimize the raw SWL by removing the repeated edges. First, from the raw SWL we generate N-best hypotheses as a sequence of edge numbers. We list the word IDs of all the edges in the hypotheses, remove the duplicate words, and index the remainders with new edge IDs. The number of new edges is fewer than that in the raw SWL. Next, we replace the edge sequence in each hypothesis with a new edge ID. If more than one edge shares the same word ID in one hypothesis, we add a new edge ID for the word again and replace the edge with the new ID. Finally, we generate a new word lattice with a new word list as its edges, consisting of the N-best hypotheses only. The raw SWL becomes the downsized SWL, which is much smaller than the raw SWL. On av-

erage, the word lattice is reduced by 50% in our experiments.

As shown in Fig. 2, one hypothesis is removed after minimization.

Sometimes the downsized SWL cannot form a lattice, but the N-best ASR hypotheses with newly assigned edge IDs. So we denote the downsized SWL as a pseudo-lattice.

3.2. Pseudo-lattice decoding algorithm

We use beam search followed by a A* search in pseudo-lattice translation. This approach has been used in text translation by [8]. We extend the approach to speech translation in this work. It is a two-pass decoding process. The first pass uses a simple model to generate a word graph to save the most likely hypotheses. It amounts to converting the pseudo word lattice into a target language word graph (TWG). Edges in the SWL are aligned to the edges in the TWG. Although the SWL is a faked lattice, the generated TWG has a true graph structure. The second pass uses a complicated model to output the best hypothesis by traversing the target word graph.

We describe the two-pass WLT algorithm in the following two sections.

3.2.1. First pass — from SWL to TWG

The bottom of Fig. 3 shows an example of a translation word graph, which corresponds to the recognition word lattice in the top. Each edge in the TWG is a target language word which is a translation of a source word in the SWL. The edges that have the same structure(including alignment and target context) are merged into a node. The node has one element indicating the source word coverage up to the current node. The coverage is a binary vector with size equal to the number of edges in the SWL, indicating the number of translated source edges. If the j-th source word was translated, the j-th element is set to 1; otherwise it equals 0. If a node covers all the edges of a full path in the SWL, it connects to the last node, the terminal node, in the TWG.

There are two main operations in expanding a node into edges: DIRECT and ALIGN. DIRECT extends the hypothesis with a target word by translating an uncovered source word. The target word is chosen based on current target N-gram context and possible translations of the uncovered source word.

ALIGN extends the hypothesis by aligning one more uncovered source word to the current node to increase fertilities of target word, where the target word is a translation of multiple source words.

The edge is not extended if the resulted hypothesis does not align to any hypothesis in the SWL. If the node has covered a full path in the SWL, this node is connected to the end node. When there is no nodes available for possible extension, the conversion is completed. A simple example of conversion algorithm is shown in Algorithm 1. The whole process equals to growing a graph. The graph can be indexed in time slices because the new nodes are created based on the old nodes of the last nearest time slice. New nodes are created by DIRECT or ALIGN to cover the uncovered source edge and connect to the old nodes. The new generated nodes are sorted in the graph buffer and merged if they share the same structure: the same coverage, the same translations, and the same N-gram sequence. If the node covers a full hypothesis in the SWL, the node connects to the terminal node. If no nodes need to be expanded, the conversion terminates.

Algorithm 1 Conversion Algorithm from SWL to TWG

```
1: Initialize graph buffer G[0]=0; t=0
2: DO
3:     FOR EACH node n=0,1,..., #(G[t]) DO
4:         IF (n cover A FULL PATH) NEXT
5:         FOR EACH edge l=0,1,...,#(EDGES) DO
6:             IF (n cover l) NEXT
7:             IF (n not cover ANY SWL PATH) NEXT
8:             generate new node and push to G[t+1]
9:         merge and prune nodes in G[t+1]
10:    t= t+1
11:WHILE (G[t] is empty)
```

In the first pass, we incorporate a simpler translation model into the log-linear model: only the lexical model, IBM model 1. The ASR posterior probabilities P_{pp} are calculated by partial hypothesis from the start to the current node. P_{pp} uses the highest value among all the ASR hypotheses under the current context. The first pass serves to keep the most likely hypotheses in the translation word graph, and leave the job of finding the optimal translation to the second pass.

3.3. Second pass — by an A* search to find the best output from the TWG

An A* search traverses the TWG generated in the last section – the best first approach. All partial hypotheses generated are pushed into a priority queue with the top hypothesis popping first out of the queue for the next extension.

To execute the A* search, the hypothesis score, $D(h,n)$, of a node n is evaluated in two parts: the forward score, $F(h,n)$, and the heuristic estimation, $H(h,n)$, $D(h,n) = F(h,n) + H(h,n)$. The calculation of $F(h,n)$ begins from the start node and accumulates all nodes' scores belonging to the hypothesis until the current node, n. The $H(h,n)$ is defined as the accumulated maximum probability of the models from the end node to the current node

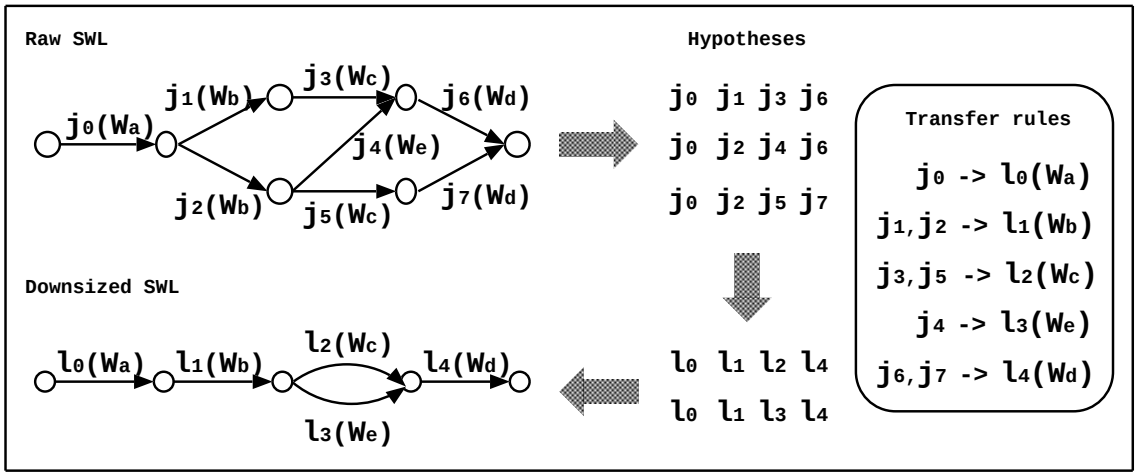

Figure 2: An example of word lattice reduction

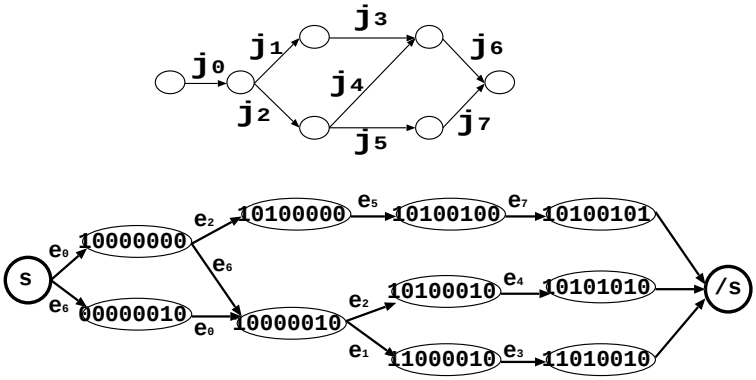

Figure 3: Source language word lattice (top) and target language word graph (bottom)

n.

In the second pass we incorporated the features of IBM Model 4 into the log-linear model. However, we cannot use IBM Model 4 directly because the calculations of the two models, $P(\Phi_0|E)$ and $\mathcal{D}(E, J)$, require the source sentence, but in fact this is unknown. Hence, the probability of $P(\Phi_0|E)$ and $\mathcal{D}(E, J)$ cannot be calculated precisely in decoding. Our method to resolve this problem is to use the maximum over all possible hypotheses. For the above two models, we calculated the scores for all the possible ASR hypotheses under the current context. The maximum value was used as the model's probability.

4. Experiments in the IWSLT2005 evaluation

There are five languages involved in the evaluation: English, Chinese, Japanese, Korean and Arabic. The available translation directions are: Chinese to English, English to Chinese, Japanese to English, Korean to English, and Arabic to English. Of these choices we participated in two tasks: Chinese to English translation and Japanese to English translation.

Regarding the data for training the translation engine, the participants must conform to four data tracks: supplied data provided by the organizer; supplied data+tools, which allows the participant to make word segmentation and morphlogical analysis of the supplied data; unrestricted data, any public data from public sources like LDC or webs; C-STAR data, with no restraints on the data, including the full BTEC corpus and proprietary data.

In this evaluation, we took part in two data tracks: supplied data+tools and the C-STAR track.

In the first track, we used our in-house part-of-speech tagging tool to make a morphological analysis of the supplied data. In the second track, we used the BTEC corpus; but, for Chinese to English translation, we used only the BTEC1 data. For Japanese to English translation we used the BTEC1-BETC4 data.

As described in the previous sections, we used the phrase-based statistical machine translation for Chinese-to-English translation and word-based SMT for Japanese-to-English translation. We trained the phrase-based translation model by carrying out bi-directional alignment first and then extracted the phrase translation pairs. The phrase translation probability was calculated by counting the phrase pair co-occurrences. Additional models used in the phrase-based approach consist of N-gram language models, distortion models and lexicon models.

For training the word-based pseudo-lattice translation models, we used GIZA++ to train an IBM Model1 and Model4. The IBM Model1 is used in the first pass of pseudo-lattice decoding and IBM Model4 used in the A* search. In addition, some models such as language models, jump size models, and target length models are integrated with the IBM Model4 log-linearly.

Some statistical properties of the experimental data and models are shown in Table 1, where language pair indicates Chinese to English translation (C/E) and Japanese to English translation (J/E). "Data size" shows the sentence numbers in the training pairs. "t-table" shows the size of source and target pairs in the translation model. Phrase-based and word-based translation models were used for Chinese-to-English and Japanese-to-English translation respectively. "Ngram" shows the number of consequent words in English language model, extracted from the training data. "perplexity" shows the source language model's perplexity in the test set and target language model's perplexity in the development data.

4.1. Evaluation results of development data and test data

Shown in table 2 and 3 are the results of development data and test data, respectively. "direct N-best" and "pseudo-lattice" mean that the speech translation are made by a direct N-best translation approach or pseudo-lattice translation approach. The development data results are of development set2, IWSLT2004, containing 500 sentences while the test data contain 506 sentences. For the Chinese ASR translation task, the organizer provides three sets of ASR output. The translations of ASR output presented in table 2 were made using the third set, the word accuracy ratio from 87.3%, single-best , to 94.5%, N-best.

After analyzing the experimental results, we can make the following conclusions:

- Undoubtedly, the translations in the C-star track are better than those of the supplied data track ,regardless of C/E or J/E, because more training data are used.

- Comparing the translation results of manual transcription, N-best, pseudo-lattice, and single-best, we found that ASR word error worsen the translations greatly because the single-best's results are much worse than the plain text's. However, using N-best hypotheses can counteract ASR word errors. N-best hypothesis translation improves single-best translation.

- In most cases, N-best translations are better than the single-best translations. The improvement by N-best translations is significant for C-star track.

- There are some inconsistence to the above analysis. The NIST score of manual transcription in the J/E supplied track is worse than the single-best's. We guess that this is because our log-linear model was optimized on the BLEU score, therefore, the NIST score was not improved.

Table 1: Properties of experimental data and models

language pair	data track	data size	t-table	Ngram	perplexity	
					testset(source language)	dev.data(target language)
C/E	supplied+tools	20,000	1.8M	97K	65.4	53.8
	C-star	172,170	5.0M	961K	69.3	52.2
J/E	supplied+tools	20,000	64K	55K	54.9	53.7
	C-star	463,365	506K	354K	22.5	31.6

Table 2: Translation results for development set2 (IWSLT2004)

translation pair	data track	translation type	BLEU	NIST	WER	PER	METEOR
C/E	supplied+tools	manual transcription	0.409	8.37	0.537	0.433	0.634
		direct N-best	0.374	7.29	0.563	0.473	0.576
		single-best	0.370	7.47	0.579	0.481	0.578
	C-star	manual transcription	0.548	9.34	0.428	0.350	0.70
		direct N-best	0.508	7.71	0.463	0.408	0.637
		single-best	0.474	7.88	0.502	0.428	0.625
J/E	supplied+tools	manual transcription	0.433	5.06	0.509	0.470	0.564
		pseudo-lattice	0.430	4.70	0.514	0.476	0.557
		single-best	0.428	4.85	0.517	0.477	0.556
	C-star	manual transcription	0.623	9.16	0.351	0.306	0.737
		pseudo-lattice	0.607	9.06	0.372	0.321	0.719
		single-best	0.596	9.02	0.377	0.328	0.716

Table 3: Translation results for test data (IWSLT2005)

translation pair	data track	translation type	BLEU	NIST	WER	PER	METEOR	GTM
C/E	supplied+tools	manual transcription	0.305	7.20	0.518	0.422	0.573	0.471
		direct N-best	0.267	6.19	0.645	0.546	0.506	0.421
		single-best	0.251	5.93	0.683	0.581	0.479	0.395
	C-star	manual transcription	0.421	8.17	0.518	0.422	0.642	0.547
		direct N-best	0.375	6.80	0.561	0.486	0.560	0.493
		single-best	0.340	6.76	0.619	0.525	0.531	0.461
J/E	supplied+tools	manual transcription	0.388	4.39	0.563	0.519	0.520	0.431
		direct N-best	0.383	4.27	0.574	0.530	0.513	0.422
		pseudo-lattice	0.378	4.18	0.578	0.534	0.511	0.420
		single-best	0.366	4.50	0.576	0.527	0.508	0.412
	C-star	manual transcription	0.727	10.94	0.289	0.243	0.80	0.716
		direct N-best	0.679	10.04	0.324	0.281	0.760	0.670
		pseudo-lattice	0.670	9.86	0.329	0.289	0.763	0.665
		single-best	0.646	9.68	0.352	0.304	0.741	0.645

4.2. Comparison of pseudo-lattice translation and direct N-best translation

This section highlights the comparison of pseudo-lattice translation and direct N-best translation. As shown in Table 3, we found in the testset evaluation both direct N-best translation and pseudo-lattice translation improved on the single-best translation. The pseudo-lattice translation is slightly worse than the direct N-best translation. A twin paper [9] describes the details of our lattice decoding algorithm. We used confidence measure to filter the ASR hypotheses with low confidence. We used the same decoding parameters as the direct N-best translation, such as beam size and threshold for pruning. And also, we applied model approximations in lattice decoding. While all these methods resulted in the improvement of the single-best translation, they made the lattice translation worse than the direct N-best translation. However, the pseudo-lattice translation is much faster. The total running time for lattice translation is only 20% of that in the direct N-best translation for the results shown in Table 3. We will continue to improve pseudo-lattice translation in future work.

5. Conclusions

Integration of speech recognition and machine translation is a promising research theme in speech translation. In addition to our approaches, finite state transducers (FST) was used in [3]. However, the speech translation performance produced by FST integration structure was reported lower than that by the single-best serial structure. A latest work in FST integration [10] carried out an Italian-English speech translation task, where a significant improvement was observed for grammartically closed languages.

Our main purpose in taking part in this year's evaluation is to verify our work in speech translation, seeking an effective solution for integrating speech recognition and machine translation. In this work we proposed two approach: direct N-best hypothesis translation and pseudo-lattice translation. Both approaches achieved satisfactory improvement over single-best translation. In some cases the improvement can reach 50% of that achieved with correct manual transcription translation.

6. Acknowledgments

We would like to thank those who have gave us their sincere assistance in this work, especially, Dr.Michael Paul, Dr.Wai-kit Lo, Dr.Xinhui Hu and Mr.Teruaki Hayashi.

The research reported here was supported in part by a contract with the National Institute of Information and Communications Technology of Japan entitled "A study of speech dialogue translation technology based on a large corpus".

We also thank the reviewers for the comments and editorial corrections.

7. References

[1] R. Zhang, G. Kikui, H. Yamamoto, T. Watanabe, F. Soong, and W. K. Lo, "A unified approach in speech-to-speech translation: Integrating features of speech recognition and machine translation," in *Proc. of Coling 2004*, Geneva, 2004.

[2] S. Saleem, S. chen Jou, S. Vogel, and T. Schultz, "Using word lattice information for a tighter coupling in speech translation systems," in *Proc. of ICSLP 2004*, Jeju, Korea, 2004.

[3] F. Casacuberta, H.Ney, F.J.Och, E.Vidal, J.M.Vilar, S.Barrachina, I.Garcia-Varea, D.Llorens, C.Martinez, S.Molau, F.Nevada, M.Pastor, D.Pico, A.Sanchis, and C.Tillmann, "Some approaches to statistical and finite-state speech-to-speech translation," in *Computer Speech and Language*, 2004, pp. 25–47.

[4] D. Marcu and W. Wong, "A phrase-based, joint probability model for statistical machine translation," in *Proc. of EMNLP-2002*, Philadelphia, PA, July 2002.

[5] P. Koehn, F. J. Och, and D. Marcu, "Statistical phrase-based translation," in *HLT/NAACL*, Edmonton, Canada, 2003.

[6] P. F. Brown, V. J. D. Pietra, S. A. D. Pietra, and R. L. Mercer, "The mathematics of statistical machine translation: Parameter estimation," *Computational Linguistics*, vol. 19, no. 2, pp. 263–311, 1993.

[7] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C++*. Cambridge, UK: Cambridge University Press, 2000.

[8] N. Ueffing, F. J. Och, and H. Ney, "Generation of word graphs in statistical machine translation," in *Proc. of the Conference on Empirical Methods for Natural Language Processing (EMNLP02)*, Philadelphia, PA, July 2002, pp. 156–163.

[9] R.Zhang, G.Kikui, H.Yamamoto, and W.Lo, "A decoding algorithm for word lattice translation in speech translation," in *IWSLT'2005*, Pittsburgh, PA, 2005.

[10] E.Matusov, S.Kanthak, and H.Ney, "On the integration of speech recognition and statistical machine translation," in *Eurospeech'2005*, Lisbon, Portugal, 2005, pp. 3177–3181.