IJCNLP 2008

Workshop on Technologies and Corpora for Asia-Pacific Speech Translation (TCAST)

Proceedings of the Workshop

Organizer

Asian Speech Translation Advanced Research Consortium
(A-Star)

Local Host

International Institute of Information Technology, India

January 11 2008
Hyderabad, India

Preface

This volume contains the paper accepted for presentation at the 2008 Workshop on Technologies and Corpora for Asia-Pacific Speech Translation (TCAST), which is part of the The Third International Joint Conference on Natural Language Processing held on January 7-12, 2008, in Hyderabad, India (IJCNLP2008). This workshop took place on January 11 2008.

In an age of global communication, information exchange by means of speech-to-speech technology is playing an increasingly important role. This technology is vital in breaking down language barriers and facilitating better social interaction and exchange in business and other areas. Research programs have been launched in many different countries and efforts have been made to develop successful speech-to-speech systems for several languages around the world. In the Asia-Pacific region, extensive efforts are needed to develop the field. In the region a large number of languages and dialects are spoken, some of these languages have a very rich cultural heritage. However, many of these languages have been neglected and information resources are not available.

Given this background, the objective of the workshop was to present the research and development work currently in progress for the development of corpora, data tools and techniques for the processing of Asian languages and their standardisation for applications in speech translation between Asian languages. The main aims of this workshop were to allow participants to interact and share knowledge of available resources and ongoing research, and to discuss possible avenues for future development in the field. This workshop was a part of the activities of the expert group on "Speech and Natural Language Processing" created under the ASTAP program, APEC-TEL and the A-Star project.

We would like to acknowledge the exceptional cooperation of our organizing committee members during the organization of this workshop.

Andrew Finch
Workshop Organizer
November 2007

Organization

Workshop Chair:

Satoshi Nakamura (NiCT-ATR, Japan)

Organizing Committee:

Satoshi Nakamura (NiCT-ATR, Japan)

Andrew Finch (NiCT-ATR, Japan)

Sakriani Sakti (NiCT-ATR, Japan)

Program Committee:

Satoshi Nakamura (NiCT-ATR, Japan)

S.S. Agrawal (CDAC, India)

Hammam Riza (BPPT, Indonesia)

Jun Park (ETRI, Korea)

Chai Wutiwiwatchai (NECTEC, Thailand)

Bo Xu (CAS, China)

Linshan Lee (NTU, Taipei)

Workshop Website:

http://www.slc.atr.jp/TCAST/TCAST2008/TCAST_Home.html

Workshop Program

09:00-09:30 Workshop Registration

09:30-10:00 Opening Speech
Satoshi Nakamura (NiCT-ATR, Japan)

Session 1: Machine Translation

10:00-10:30 *Transformation-based Sentence Splitting method for Statistical Machine Translation*
Jonghoon Lee, Donghyeon Lee and Gary Geunbae Lee

10:30-11:00 Coffee Break

11:00-11:30 *Speech-to-Speech Translation Activities in Thailand*
Chai Wutiwiwatchai, Thepchai Supnithi and Krit Kosawat

11:30-12:00 *Phrase-based Machine Transliteration*
Andrew Finch and Eiichiro Sumita

12:00-13:30 Lunch

Session 2: Speech Recognition

13:30-14:00 *Development of Indonesian Large Vocabulary Continuous Speech Recognition System within A-STAR Project*
Sakriani Sakti, Eka Kelana, Hammam Riza, Shinsuke Sakai, Konstantin Markov and Satoshi Nakamura

14:00-14:30 *Using Confidence Vector in Multi-Stage Speech Recognition*
Hyungbae Jeon, Kyuwoong Hwang, Hoon Chung, Seunghi Kim, Jun Park and Yunkeun Lee

14:30-15:00 *Toward Asian Speech Translation System: Developing Speech Recognition and Machine Translation for Indonesian Language*
Hammam Riza and Oskar Riandi

15:00-15:45 Discussion and Closing

Table of Contents

Transformation-based Sentence Splitting method for Satistical Machine Translation
Jonghoon Lee, Donghyeon Lee and Gary Geunbae Lee..... 1

Speech-to-Speech Translation Activities in Thailand
Chai Wutiwiwatchai, Thepchai Supnithi and Krit Kosawat.....7

Phrase-based Machine Transliteration
Andrew Finch and Eiichiro Sumita..... 13

Development of Indonesian Large Vocabulary Continuous Speech Recognition System within A-STAR Project
Sakriani Sakti, Eka Kelana, Hammam Riza, Shinsuke Sakai, Konstantin Markov and Satoshi Nakamura.....19

Using Confidence Vector in Multi-Stage Speech Recognition
Hyungbae Jeon, Kyuwoong Hwang, Hoon Chung, Seunghi Kim, Jun Park and Yunkeun Lee.....25

Toward Asian Speech Translation System: Developing Speech Recognition and Machine Translation for Indonesian Language
Hammam Riza and Oskar Riandi..... 35

Author Index

Hoon Chung..... 25
Andrew Finch..... 13
Kyuwoong Hwang.....25
Hyungbae Jeon..... 25
Eka Kelana..... 19
Seunghi Kim.....25
Krit Kosawat.....7
Donghyeon Lee..... 1
Geunbae Lee.....1
Jonghoon Lee..... 1
Yunkeun Lee.....25
Konstantin Markov.....19
Satoshi Nakamura.....19
Jun Park.....25
Oskar Riandi..... 35
Hammam Riza..... 19,35
Shinsuke Sakai.....19
Sakriani Sakti.....19
Eiichiro Sumita.....13
Thepchai Supnithi.....7
Chai Wutiwiwatchai..... 7

A Transformation-based Sentence Splitting Method for Statistical Machine Translation

Jonghoon Lee, Donghyeon Lee and Gary Geunbae Lee
Department of Computer Science and Engineering
Pohang University of Science & Technology (POSTECH)
{jh21983, semko, gblee}@postech.ac.kr

Abstract

We propose a transformation based sentence splitting method for statistical machine translation. Transformations are expanded to improve machine translation quality after automatically obtained from manually split corpus. Through a series of experiments we show that the transformation based sentence splitting is effective pre-processing to long sentence translation.

1 Introduction

Statistical approaches to machine translation have been studied actively, after the formalism of statistical machine translation (SMT) is proposed by Brown et al. (1993). Although many approaches of them were effective, there are still lots of problems to solve. Among others, we have an interest in the problems occurring with long sentence decoding. Various problems occur when we try to translate long input sentences because a longer sentence contains more possibilities of selecting translation options and reordering phrases. However, reordering models in traditional phrase-based systems are not sufficient to treat such complex cases when we translate long sentences (Koehn et al, 2003).

Some methods which can offer powerful reordering policies have been proposed like syntax based machine translation (Yamada and Knight, 2001) and Inversion Transduction Grammar (Wu, 1997). Although these approaches are effective, decoding long sentences is still difficult due to their computational complexity. As the length of an input sentence becomes longer, the analysis and decoding become more complex. The complexity causes approximations and errors inevitable during the decoding search.

In order to reduce this kind of difficulty caused by the complexity, a long sentence can be paraphrased by several shorter sentences with the same meaning. Generally, however, decomposing a complex sentence into sub-sentences requires information of the sentence structures which can be obtained by syntactic or semantic analysis. Unfortunately, the high level syntactic and semantic analysis can be erroneous and costs as expensive as SMT itself. So, we don't want to fully analyze the sentences to get a series of sub-sentences, and our approach to this problem considers splitting only compound sentences.

In the past years, many research works were concerned with sentence splitting methods to improve machine translation quality. This idea had been used in speech translation (Furuse et al, 1998) and example based machine translation (Doi and Sumita, 2004). These research works achieved meaningful results in terms of machine translation quality. Unfortunately, however, the method of Doi and Sumita using n-gram is not available if the source language is Korean. In Korean language, most of sentences have special form of ending morphemes at the end. For that reason, we should determine not only the splitting position but also the ending morphemes that we should replace instead of connecting morphemes. And the Furuse et al's method involves parsing which requires heavy cost.

In this paper we propose a transformation based splitting method to improve machine translation quality which can be applied to the translation tasks with Korean as a source language.

2 Methods

Our task is splitting a long compound sentence into short sub-sentences to improve the performance of phrase-based statistical machine translation system. We use a transformation based approach to accomplish our goal.

2.1 A Concept of Transformation

The transformation based learning (TBL) is a kind of rule learning methods. The formalism of TBL is introduced by Brill (1995). In past years, the TBL approach was used to solve various problems in natural language processing such as part of speech (POS) tagging and parsing (Brill, 1993).

A transformation consists of two parts: a triggering environment and a rewriting rule. And the rewriting rule consists of a source pattern and a target pattern. Our consideration is how to get the right transformations and apply them to split the long sentences.

A transformation works in the following manner; some portion of the input is changed by the rewriting rule if the input meets a condition specified in the triggering environment. The rewriting rule finds the source pattern in the input and replaces it with the target pattern. For example, suppose that a transformation which have a triggering environment A, source pattern B and target pattern C. We can describe this transformation as a sentence: if a condition A is satisfied by an input sentence, then replace pattern B in the input sentence with pattern C.

2.2 A Transformation Based Sentence Splitting Method

Normally, we have two choices when there are two or more transformations available for an input pattern at the same time. The first choice is applying the transformation one by one, and the second choice is applying them simultaneously. The choice is up to the characteristics of the problem that we want to solve. In our problem, we choose the former strategy which is applying the transformations one by one, because it gives direct intuition about the process of splitting sentences. By choosing this strategy, we can design splitting process as a recursive algorithm.

At first, we try to split an input sentence into two sub-sentences. If the sentence has been split by some transformation, the result involves exactly two sub-sentences. And then we try to split each sub-sentence again. We repeat this process in recursive manner until no sub-sentences are split.

In the above process, a sentence is split into at most two sub-sentences through a single trial. In a single trial, a transformation works in the following manner: If an input sentence satisfies the environment, we substitute the source pattern into the target pattern. That is, replace the connecting morphemes with the proper ending morphemes. And then we split the sentence with pre-defined position in the transformation. And finally, we insert the junction word that is also pre-defined in the transformation between the split sentences after the sub sentences are translated independently.

From the above process, we can notice easily that a transformation for sentence splitting consists of the four components: a triggering environment, a rewriting rule, a splitting position and a junction type. The contents of each component are as follows. (1) A triggering environment contains a sequence of morphemes with their POS tags. (2) A rewriting consists of a pair of sequences of POS tagged morphemes. (3) A junction type can have one of four types: ‘and’, ‘or’, ‘but’ and ‘NULL’. (4) A splitting position is a non-negative integer that means the position of starting word of second sub-sentence.

2.3 Learning the Transformation for Sentence Splitting

At the training phase, TBL process determines the order of application (or rank) of the transformations to minimize the error-rate defined by a specific measure. The order is determined by choosing the best rule for a given situation and applying the best rule for each situation iteratively. In the sentence splitting task, we maximize the machine translation quality with BLEU score (Papineni et al., 2001) instead of minimizing the error of sentence splitting.

During the training phase, we determine the order of applying transformation after we build a set of transformations. To build the set of transformations, we need manually split examples to learn the transformations.

Building a transformation starts from extracting a rewriting rule by calculating edit-distance matrix between an original sentence and its split form from the corpus. We can easily extract the different parts from the matrix.

```
BaseBLEU :=  BLEU score of the baseline system
S := Split example sentence
T := Extracted initial transformation
for each t∈ T
   for each s∈S
      while true
         try to split s with t
         if mis-splitting is occurred
            Expand environment
         else exit while loop
         if environment cannot be expanded
            exit while loop
   S' := apply t to S
   Decode S'
   BLEU := measure BLEU
   Discard t if BLEU < BaseBLEU
sort  T w.r.t. BLEU
```

Figure 1. Modified TBL for sentence splitting

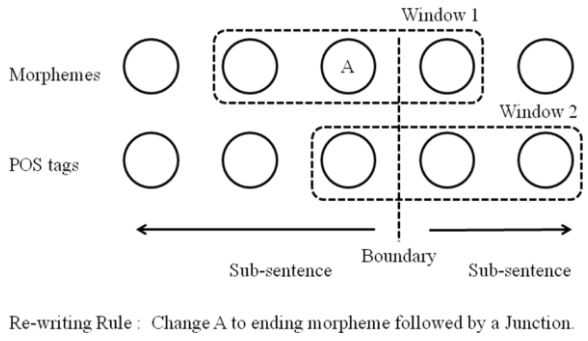

Figure 2. Window-based heuristics for triggering environments

From the difference pattern, we can make the source pattern of a rewriting rule by taking the different parts of the original sentence side. Similarly, the target pattern can be obtained from the different parts of split form. And the junction type and splitting position are directly obtained from the difference pattern. Finally, the transformation is completed by setting the triggering environment as same to the source pattern. The set of initial transformations is obtained by repeating this process on all the examples.

The Transformations for sentence splitting are built from the initial transformations through expanding process. In the expanding process, each rule is applied to the split examples. We expand the triggering environment with some heuristics (in section 2.4), if a sentence is a mis-split.

And finally, in order to determine the rank of each transformation, we sorted the extracted transformations by decreasing order of resulted BLEU scores after applying the transformation to each training sentence. And some transformations are discarded if they decrease the BLEU score. This process is different from original TBL. The modified TBL learning process is described in figure 1.

2.4 Expanding Triggering Environments

Expanding environment should be treated very carefully. If the environment is too specific, the transformation cannot be used in real situation. On the other hand, if it is too general, then the transformation becomes erroneous.

Our main strategy for expanding the environment is to increase context window size of the triggering environment one by one until it causes no error on the training sentences. In this manner, we can get minimal error-free transformations on the sentence splitting corpus.

We use two different windows to define a triggering environment: one for morpheme and another for its part of speech (POS) tag. Figure 2 shows this concept of two windows. The circles correspond to sequences of morphemes and POS tags in a splitting example. Window 1 represents a morpheme context and window 2 represents a POS tag context. The windows are independently expanded from the initial environment which consists of a morpheme 'A' and its POS tag. In the figure, window 1 is expanded to one forward morpheme and one backward morpheme while window 2 is expanded to two backward POS tags.

In order to control these windows, we defined some heuristics by specifying the following three policies of expanding windows: no expansion, forward only and forward and backward. From those three polices, we have 9 combinations of heuristics because we have two windows. By observing the behavior of these heuristics, we can estimate what kind of information is most important to determine the triggering environment.

		SMT		Splitting	
		Korean	English	Before Split	After Split
Train	# of Sentences	123,425		1,577	1,906
	# of Words	1,083,912	916,950	19,918	20,243
	Vocabulary	15,002	14,242	1,956	1,952
Test	#of Sentences	1,577		-	-

Table 1. Corpus statistics

Test No.	Window1 policy	Window2 policy
Test 1	No expansion	No expansion
Test 2		Forward only
Test 3		Free expansion
Test 4	Forward only	No expansion
Test 5		Forward only
Test 6		Free expansion
Test 7	Free expansion	No expansion
Test 8		Forward only
Test 9		Free expansion

Table 2.Experimental setup

Test No.	# of affected sentences	BLEU score	
		Before splitting	After splitting
Test 1	209	0.1778	0.1838
Test 2	142	0.1564	0.1846
Test 3	110	0.1634	0.1863
Test 4	9	0.1871	0.2150
Test 5	96	0.1398	0.1682
Test 6	100	0.1452	0.1699
Test 7	8	0.2122	0.2433
Test 8	157	0.1515	0.1727
Test 9	98	0.1409	0.1664

Table 3. BLEU scores of affected sentences

We have at most 4 choices for a single step of the expanding procedure: forward morpheme, backward morpheme, forward POS tag, and backward POS tag. We choose one of them in a fixed order: forward POS tag, forward morpheme, backward POS tag and backward morpheme. These choices can be limited by 9 heuristics. For example, suppose that we use a heuristic with forward policy on morpheme context window and no expansion policy for POS tag context window. In this case we have only one choice: forward morpheme.

3 Experiments

We performed a series of experiments on Korean to English translation task to see how the sentence splitting affects machine translation quality and which heuristics are the best. Our baseline system built with Pharaoh (Koehn, 2004) which is most popular phrase-based decoder. And trigram language model with KN-discounting (Kneser and Ney, 1995) built by SRILM toolkit (Stolcke, 2002) is used.

Table 1 shows the corpus statistics used in the experiments. The training corpus for MT system has been built by manually translating Korean sentences which are collected from various sources. We built 123,425 sentence pairs for training SMT, 1,577 pairs for splitting and another 1,577 pairs for testing. The domain of the text is daily conversations and travel expressions. The sentence splitting corpus has been built by extracting long sentences from the source-side mono-lingual corpus. The sentences in the splitting corpus have been manually split.

The experimental settings for comparing 9 heuristics described in the section 2.4 are listed in table 2. Each experiment corresponds to a heuristic.

To see the effect of sentence splitting on translation quality, we evaluated BLEU score for affected sentenced by the splitting. The results are shown in table 3. Each test number shows the effect of transformation-based sentence splitting with different window selection heuristics listed in table 2. The scores are consistently increased with significant differences. After analyzing the results of table 3, we notice that we can expect some perfor-

Test No.	# of trans-formations (rules)	# of changes (sentences)	# of supe-rior changes	# of infe-rior changes	# of insig-nificant changes	Ratio Sup/Inf	Ratio trans/chang e
1	34	209	60	30	119	2.00	6.15
2	177	142	43	9	90	4.78	0.802
3	213	110	29	9	72	3.22	0.516
4	287	9	4	1	4	4.00	0.031
5	206	96	25	4	67	6.25	0.466
6	209	100	23	8	69	2.88	0.478
7	256	8	3	1	4	3.00	0.031
8	177	157	42	10	102	4.20	0.887
9	210	98	21	4	73	5.25	0.467

Table 4. Human evaluation results

Superior change	Reference	I saw that some items are on sale on window . what are they ?
	Baseline	What kind of items do you have this item in OOV some discount, I get a discount ?
	Split	You have this item in OOV some discount . what kind of items do I get a discount ?
Insignificant change	Reference	What is necessary to be issued a new credit card?
	Baseline	I ‘d like to make a credit card . What do I need?
	Split	I ‘d like to make a credit card . What is necessary?
Inferior change	Reference	I ‘d like to make a reservation by phone and tell me the phone number please .
	Baseline	I ‘d like to make a reservation but can you tell me the phone number , please .
	Split	I ‘d like to make a reservation . can you tell me the , please .

Table 5. Example translations (The sentences are manually re-cased for readability)

mance gain when the average sentence length is long.

The human evaluation shows more promising results in table 4. In the table, the superior change means that the splitting results in better translation and inferior means the opposite case. Two ratios are calculated to see the effects of sentence splitting. The ratio ‘sup/inf’ shows the ratio of superior over inferior splitting. And ratio trans/change shows how many sentences are affected by a transformation in an average. In most of the experiments, the number of superior splitting is over three times larger than that of inferior ones. This result means that the sentence splitting is a helpful pre-processing for machine translation.

We listed some example translations affected by sentence splitting in the table 5. In the three cases, junction words don’t appear in the results of translation after split because their junction types are NULL that involves no junction word. Although several kinds of improvements are observed in superior cases, the most interesting case occurs in out-of-vocabulary (OOV) cases. A translation result has a tendency to be a word salad when OOV’s are included in the input sentence. In this case, the whole sentence may lose its original meaning in the result of translation. But after splitting the input sentence, the OOV’s have a high chance to be located in one of the split subsentences. Then the translation result can save at least a part of its original meaning. This case occurs easily if an input sentence includes only one OOV. The Superior change of table 5 is the case. Although both baseline and split are far from the reference, split catches some portion of the meaning.

Most of the Inferior cases are caused by mis-splitting. Mis-splitting includes a case of splitting a sentence that should not be split or splitting a sentence on the wrong position. This case can be reduced by controlling the heuristics described in section 2.4. But the problem is that the effort to reducing inferior cases also reduces the superior cases. To compare the heuristics each other in this condition, we calculated the ratio of superior and inferior cases. The best heuristic is test no. 5 in terms of the ratio of sup/inf.

The test no. 4 and 7 show that a trans-formation becomes very specific when lexical information is used alone. Hence the ratio trans/change becomes below 0.01 in this case. And test no. 1 shows that the transformations with no environment expansion are erroneous since it has the lowest ratio of sup/inf.

4 Conclusion

We introduced a transformation based sentence splitting method for machine translation as a effective and efficient pre-processing. A transformation consists of a triggering environment and a rewriting rule with position and junction type information. The triggering environment of a transformation is extended to be error-free with respect to training corpus after a rewriting rule is extracted from manually split examples. The expanding process for the transformation can be generalized by adding POS tag information into the triggering environment.

The experimental results show that the effect of splitting is clear in terms of both automatic evaluation metric and human evaluation. The results consistently state that the statistical machine translation quality can be improved by transformation based sentence splitting method.

Acknowledgments

This research was supported by the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Assessment) (IITA-2006-C1090-0603-0045). The parallel corpus was courteously provided by Infinity Telecom, Inc.

References

Eric Brill. 1993. Transformation-based error-driven parsing. *In Proc. of third International Workshop on Parsing.*

Eric Brill. 1995. Transformation-based error-driven learning and natural language processing: A Case Study in Part-of-Speech Tagging. *Computational Linguistics 21(4):543-565.*

Peter F. Brown, Stephen A. Della Pietra, Vincent J.Della Pietra and Robert L. Mercer. 1993. The Mathematics of Statistical Machine Translation: Parameter estimation. *Computational Linguistics, 19(2):263-312.*

Takao Doi and Eiichiro Sumita. 2004. Splitting input sentence for machine translation using language model with sentence similarity. *In Proc. of the 20th international conference on Computational Linguistics.*

Osamu Furuse, Setsuo Yamada and Kazuhide Yamamoto. 1998. Splitting Long or Ill-formed Input for Robust Spoken-language Translation. *In Proc of the 36th annual meeting on Association for Computational Linguistics.*

Reinhard Kneser and Hermann Ney. 1995. Improved backing-off for m-gram lnguage modeling. *In Proc. of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP).*

Philipp Koehn. 2004. Pharaoh: a beam search decoder for phrase-based statistical machine translation models. *In Proc. of the 6th Conference of the Association for Machine translation in the Americas*.

Philipp Koehn, Franz Josef Och and Kevin Knight. 2003. Statistical Phrase-Based Translation. *In Proc of the of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2001. BLEU: A method for automatic evaluation of Machine Translation. *Technical Report RC22176, IBM.*

Andreas Stolcke. 2002. SRILM - an extensible language modeling toolkit. *In Proc. of the 7th International Conference on Spoken Language Processing (ICSLP).*

Dekai Wu. 1997. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics 23(3):377-404.*

Kenji Yamada and Kevin Knight. 2001. A syntax-based statistical translation Model. *In Proc. of the conference of the Association for Computational Linguistics (ACL).*

Speech-to-Speech Translation Activities in Thailand

Chai Wutiwiwatchai, Thepchai Supnithi, Krit Kosawat
Human Language Technology Laboratory
National Electronics and Computer Technology Center
112 Pahonyothin Rd., Klong-luang, Pathumthani 12120 Thailand
{chai.wut, thepchai.sup, krit.kos}@nectec.or.th

Abstract

A speech-to-speech translation project (S2S) has been conducted since 2006 by the Human Language Technology laboratory at the National Electronics and Computer Technology Center (NECTEC) in Thailand. During the past one year, there happened a lot of activities regarding technologies constituted for S2S, including automatic speech recognition (ASR), machine translation (MT), text-to-speech synthesis (TTS), as well as technology for language resource and fundamental tool development. A developed prototype of English-to-Thai S2S has opened several research issues, which has been taken into consideration. This article intensively reports all major research and development activities and points out remaining issues for the rest two years of the project.

1 Introduction

Speech-to-speech translation (S2S) has been extensively researched since many years ago. Most of works were on some major languages such as translation among European languages, American English, Mandarin Chinese, and Japanese. There is no initiative of such research for the Thai language. In the National Electronics and Computer Technology Center (NECTEC), Thailand, there is a somewhat long history of research on Thai speech and natural language processing. Major technologies include Thai automatic speech recognition (ASR), Thai text-to-speech synthesis (TTS), English-Thai machine translation (MT), and language resource and fundamental tool development. These basic technologies are ready to seed for S2S research. The S2S project has then been conducted in NECTEC since the end of 2006.

The aim of the 3-year S2S project initiated by NECTEC is to build an English-Thai S2S service over the Internet for a travel domain, i.e. to be used by foreigners who journey in Thailand. In the first year, the baseline system combining the existing basic modules applied for the travel domain was developed. The prototype has opened several research issues needed to be solved in the rest two years of the project. This article summarizes all significant activities regarding each basic technology and reports remaining problems as well as the future plan to enhance the baseline system.

The rest of article is organized as follows. The four next sections describe in details activities conducted for ASR, MT, TTS, and language resources and fundamental tools. Section 6 summarizes the integration of S2S system and discusses on remaining research issues as well as on-going works. Section 7 concludes this article.

2 Automatic Speech Recognition (ASR)

Thai ASR research focused on two major topics. The first topic aimed to practice ASR in real environments, whereas the second topic moved towards large vocabulary continuous speech recognition (LVCSR) in rather spontaneous styles such as news broadcasting and telephone conversation. Following sub-sections give more details.

2.1 Robust speech recognition

To tackle the problem of noisy environments, acoustic model selection was adopted in our system. A tree structure was constructed with each leaf node containing speaker-, noise-, and/or SNR-specific acoustic model. The structure allowed ef-

ficient searching over a variety of speech environments. Similar to many robust ASR systems, the selected acoustic model was enhanced by adapting by the input speech using any adaptation algorithm such as MLLR or MAP. In our model, however, simulated-data adaptation was proposed (Thatphithakkul et al., 2006). The method synthesized an adaptation set by adding noise extracted from the input speech to a pre-recorded set of clean speech. A speech/non-speech detection module determined in the input speech the silence portions, which were assumed to be the environmental noise. This approach solved the problem of incorrect transcription in unsupervised adaptation and enhanced the adaptation performance by increasing the size of adaptation data.

2.2 Large-vocabulary continuous speech recognition (LVCSR)

During the last few years, researches on continuous speech recognition were based mainly on two databases, the NECTEC-ATR (Kasuriya et al., 2003a) and the LOTUS (Kasuriya et al., 2003b). The former corpus was for general purposes, whereas the latter corpus was well designed for research on acoustic phonetics as well as research on 5,000-word dictation systems. A number of research works were reported, starting by optimizing the Thai phoneme inventory (Kanokphara, 2003).

Recently, research has moved closer to real and spontaneous speech. The first task collaborated with a Thai telephone service provider was to build a telephone conversation corpus (Cotsomrong et al., 2007). To accelerate the corpus development, Thatphithakkul et al. (2007) developed a speaker segmentation model which helped separating speech from two speakers being conversed. The model was based on the simple Hidden Markov model (HMM), which achieved over 70% accuracy. Another on-going task is a collection of broadcast news video. The aim of the task is to explore the possibility to use the existing read-speech model to boot broadcast news transcription. More details will be given in Section 5.

3 Machine Translation (MT)

It was a long history of the NECTEC English-to-Thai machine translation (MT) which has been publicly serviced online. The "Parsit" [1] system modified from the engine developed by NEC, Japan, which was a rule-based MT (RBMT). Over 900 parsed rules were coded by Thai linguists. The system recognized more than 70,000 lexical words and 120,000 meanings.

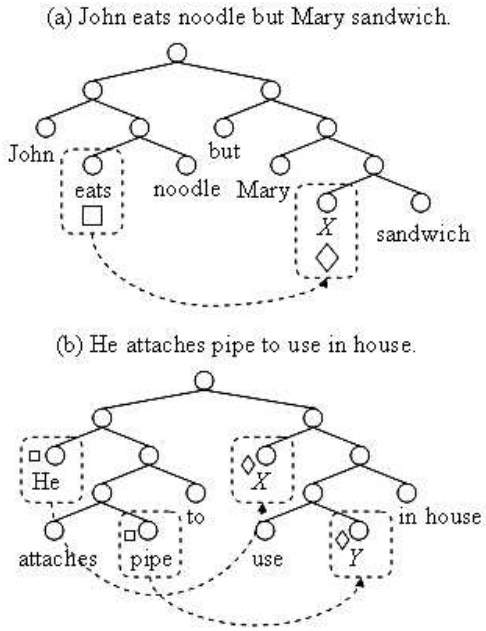

Figure 1. Examples of using MICG to solve two major problems of parsing Thai, (a) coordination with gapping and (b) verb serialization.

3.1 Thai-to-English MT

Recently, there has been an effort to develop the first rule-based system for Thai-to-English MT. The task is much more difficult than the original English-to-Thai translation since the Thai word segmentation, sentence breaking, and grammar parser are all not complete. Coding rules for parsing Thai is not trivial and the existing approach used to translate English to Thai cannot be applied counter wise. Last year, a novel rule-based approach appropriate for Thai was proposed (Boonkwan and Supnithi, 2007). The approach, called memory-inductive categorical grammar (MICG), was derived from the categorical grammar (CG). The MICG introduced memorization and induction symbols to solve problems of analytic languages such as Thai as well as many spo-

[1] Parsit MT, http://www.suparsit.com/

ken languages. In parsing Thai, there are two major problems, coordination with gapping and verb serialization. Figure 1 shows examples of the two problems with the MICG solution, where the square symbol denotes the chunk to be memorized and the diamond symbol denotes the chunk to be induced. A missing text chunk can be induced by seeking for its associated memorized text chunk.

3.2 TM and SMT

In order to improve the performance of our translation service, we have adopted a translation memory (TM) module in which translation results corrected by users are stored and reused. Moreover, the service system is capable to store translation results of individual users. A naïve user can select from the list of translation results given by various users. Figure 3 captures the system interface.

Due to powerful hardware today, research has turned to rely more on statistical approaches. This is also true for the machine translation issue. Statistical machine translation (SMT) has played an important role on modeling translation given a large amount of parallel text. In NECTEC, we also realize the benefit of SMT especially on its adaptability and naturalness of translation results. However, a drawback of SMT compared to RBMT is that it works quite well on a limited domain, i.e. translating in a specific domain. This is actually suitable to the S2S engine which has been designed to work in only a travel domain. Therefore, in parallel to RBMT, SMT is being explored for limited domains. Two parallel text corpora have been constructed. The first one, collected by ATR under the Asian speech translation advanced research (A-STAR)[2] consortium, is a Thai incorporated version of the Basic travel expression (BTEC) corpus (Kikui et al., 2003). This corpus will seed the development of S2S in the travel domain. The second parallel corpus contains examples of parallel sentences given in several Thai-English dictionaries. The latter corpus has been used for a general evaluation of Thai-English SMT. Details of both corpora will be given in the Section 5.

4 Text-to-Speech Synthesis (TTS)

Thai TTS research has begun since 2000. At present, the system utilizes a corpus-based unit-selection technique. A well-constructed phonetically-balanced speech corpus, namely "TSynC-1", containing approximately 13 hours is embedded in the TTS engine, namely "Vaja"[3]. Although the latest version of Vaja achieved a fair speech quality, there are still a plenty of rooms to improve the system. During the past few years, two major issues were considered; reducing the size of speech corpus and improving unit selection by prosody information. Following sub-sections describe the detail of each issue.

4.1 Corpus space reduction

A major problem of corpus-based unit-selection TTS is the large size of speech corpus required to obtain high-quality, natural synthetic-speech. Scalability and adaptability of such huge database become a critical issue. We then need the most compact speech corpus that still provides acceptable speech quality. An efficient way to reduce the size of corpus was recently proposed (Wutiwiwatchai et al., 2007). The method incorporated Thai phonetics knowledge in the design of phoneme/diphone inventory. Two assumptions on diphone characteristics were proved and used in the new design. One was to remove from the inventory the diphone whose coarticulation strength between adjacent phonemes was very weak. Normally, the corpus was designed to cover all tonal diphones in Thai. The second strategy to reduce the corpus was to ignore tonal levels of unvoiced phonemes. Experiments showed approximately 30% reduction of the speech corpus with the quality of synthesized speech remained.

4.2 Prosody-based naturalness improvement

The baseline TTS system selected speech units by considering only phoneme and tone context. In the past few years, analyses and modeling Thai prosodic features useful for TTS have been extensively explored. The first issue was to detect phrasal units given an input text. After several experiments (Tesprasit et al., 2003; Hansakunbuntheung et al., 2005), we decided to develop a classification and decision tree (CART) for phrase break detection.

The second issue was to model phoneme duration. Hansakunbuntheung et al. (2003) compared several models to predict the phoneme duration.

[2] A-STAR consortium, http://www.slc.atr.jp/AStar/

[3] Vaja TTS, http://vaja.nectec.or.th/

Mainly, we found linear regression appropriate for our engine as its simplicity and efficiency. Both two prosody information were integrated in our Vaja TTS engine, which achieved a better synthesis quality regarding subjective and objective evaluations (Rugchatjaroen et al., 2007).

5 Language Resources and Tools

A lot of research issues described in previous sections definitely requires the development and assessment of speech and language corpora. At the same time, there have been attempts to enhance the existing language processing tools that are commonly used in a number of advanced applications. This section explains the activities on resource and tool development.

5.1 Speech and text corpora

Table 1 summarizes recent speech and text corpora developed in NECTEC. Speech corpora in NECTEC have been continuously developed since 2000. The first official corpus under the collaboration with ATR was for general purpose (Kasuriya et al., 2003a). The largest speech corpus, called LOTUS (Kasuriya et al., 2003b), was well-designed read speech in clean and office environments. It contained both phonetically balanced utterances and news paper utterances covering 5,000 lexical words. The latter set was designed for research on Thai dictation systems. Several research works utilizing the LOTUS were reported as described in the Section 2.2.

The last year was the first-year collaboration of NECTEC and a telephone service provider to develop the first Thai telephone conversation speech corpus (Cotsomrong et al., 2007). The corpus has been used to enhance the ASR capability in dealing with various noisy telephone speeches.

Regarding text corpora, as already mentioned in the Section 3, two parallel text corpora were developed. The first corpus was a Thai version of the Basic travel expression corpus (BTEC), which will be used to train a S2S system. The second corpus developed ourselves was a general domain. It will be used also in the SMT research. Another important issue of corpus technology is to create golden standards for several Thai language processing topics. Our last year attempts focused on two sets; a golden standard set for evaluating MT and a golden standard set for training and evaluating Thai word segmentation. Finally, the most basic but essential in all works is the dictionary. Within the last year, we have increased the number of word entries in our lexicon from 35,000 English-to-Thai and 53,000 Thai-to-English entries to over 70,000 entries both. This incremental dictionary will be very useful in sustaining improvement of many language processing applications.

Table 1. Recent speech/text corpora in NECTEC.

Corpus	Purpose	Details
LOTUS	Well-designed speech utterances for 5,000-word dictation systems	- 70 hours of phonetically balanced and 5,000-word coverage sets
TSynC-1	Corpus-based unit-selection Thai speech synthesis	- 13 hours prosody-tagged fluent speech
Thai BTEC	Parallel text and speech corpora for travel-domain S2S	- 20,000 textual sentences and a small set of speech in travel domain
Parallel text	Pairs of Thai-English sample sentences from dictionaries used for SMT	- 0.2M pairs of sentences
NECTEC-TRUE	Telephone conversation speech for acoustic modeling	- 10 hours conversational speech in various telephone types

5.2 Fundamental language tools

Two major language tools have been substantially researched, word segmentation and letter-to-sound conversion. These basic tools are very useful in many applications such as ASR, MT, TTS, as well as Information retrieval (IR).

Since Thai writing has no explicit word and sentence boundary marker. The first issue on processing Thai is to perform word segmentation. Our baseline morphological analyzer determined word boundaries and word part-of-speech (POS) simultaneously using a POS n-gram model and a predefined lexicon. Recently, we have explored Thai named-entity (NE) recognition, which is expected to help alleviating the problem of incorrect word segmentation. Due to the difficulty of Thai word segmentation, we initiated a benchmark evaluation on Thai word segmentation, which will be held in 2008. This will gather researchers who are inter-

ested in Thai language processing to consider the problem on a standard text corpus.

The problem of incorrect word segmentation propagates to the letter-to-sound conversion (LTS) module which finds pronunciations on the word basis. Our original LTS algorithm was based on probabilistic generalized LR parser (PGLR). Recently, we proposed a novel method to automatically induce syllable patterns from a large text with no need for any preprocessing (Thangthai et al., 2006). This approach largely helped alleviating the tedious work on text corpus annotation.

Another important issue we took into account was an automatic approach to find pronunciations of English words using Thai phonology. The issue is particularly necessary in many languages where their local scripts are always mixed with English scripts. We proposed a new model that utilized both English graphemes and English phonemes, if found in an English pronunciation dictionary, to predict Thai phonemes of the word (Thangthai et al., 2007).

6 Speech-to-Speech Translation (S2S)

In parallel to the research and development of individual technology elements, some efforts have been on the development of Thai-English speech-to-speech translation (S2S). Wutiwiwatchai (2007) already explained in details about the activities, which will be briefly reported in this section.

As described briefly in the Introduction, the aim of our three-year S2S project is to develop an S2S engine in the travel domain, which will be given service over the Internet. In the last year, we developed a prototype English-to-Thai S2S engine, where major tasks turned to be the development of English ASR in the travel domain and the integration of three core engines, English ASR, English-to-Thai RBMT, and Thai TTS.

6.1 System development

Our current prototype of English ASR adopted a well-known SPHINX toolkit, developed by Carnegie Mellon University. An American English acoustic model has been provided with the toolkit. An n-gram language model was trained by a small set of sentences in travel domain. The training text contains 210 patterns of sentences spanning over 480 lexical words, all prepared by hands. Figure 2 shows some examples of sentence pattern.

```
Call <DIGIT>
A CONTAINER of DRINK
Check the bill
I come from COUNTRY
I want to go to PLACE
I want to go to the nearest STOP
I would like to have FOOD
What time does VEHICLE go
```

Figure 2. Examples of sentence patterns for language modeling (uppercases are word classes, bracket means repetition).

In the return direction, a Thai ASR is required. Instead of using the SPHINX toolkit[4], we built our own Thai ASR toolkit, which accepts an acoustic model in the Hidden Markov toolkit (HTK)[5] format proposed by Cambridge University. The "iSpeech"[6] toolkit that supports an n-gram language model is currently under developing.

The English ASR, English-to-Thai RBMT, and Thai TTS were integrated simply by using the 1-best result of ASR as an input of MT and generating a sound of the MT output by TTS. The prototype system, run on PC, utilizes a push-to-talk interface so that errors made by ASR can be alleviated.

6.2 On-going works

To enhance the acoustic and language models, a Thai speech corpus as well as a Thai-English parallel corpus in the travel domain is constructing as mentioned in the Section 5.1, the Thai version of BTEC corpus. Each monolingual part of the parallel text will be used to train a specific ASR language model.

For the MT module, we can use the parallel text to train a TM or SMT. We expect to combine the trained model with our existing rule-based model, which will be hopefully more effective than each individual model. Recently, we have developed a TM engine. It will be incorporated in the S2S engine in this early stage.

In the part of TTS, several issues have been researched and integrated in the system. On-going works include incorporating a Thai intonation

[4] CMU SPHINX, http://cmusphinx.sourceforge.net/
[5] HTK, Cambridge University, http://htk.eng.cam.ac.uk/
[6] iSpeech ASR, http://www.nectec.or.th/rdi/ispeech/

model in unit-selection, improving the accuracy of Thai text segmentation, and learning for hidden Markov model (HMM) based speech synthesis, which will hopefully provide a good framework for compiling TTS on portable devices.

7 Conclusion

There have been a considerable amount of research and development issues carried out under the speech-to-speech translation project at NECTEC, Thailand. This article summarized and reported all significant works mainly in the last few years. Indeed, research and development activities in each technology element, i.e. ASR, MT, and TTS have been sustained individually. The attempt to integrate all systems forming an innovative technology of S2S has just been carried out for a year. There are many research and development topics left to explore. Major challenges include at least but not limited to the following issues:

- The rapid development of Thai-specific elements such as robust Thai domain-specific ASR and MT
- Migration of the existing written language translation to spoken language translation

Recently, there have been some initiations of machine translation among Thai and other languages such as Javi, a minor language used in the southern part of Thailand and Mandarin Chinese. We expect that some technologies carried out in this S2S project will be helpful in porting to the other pairs of languages.

Acknowledgement

The authors would like to thank the ATR, Japan, in initiating the fruitful A-STAR consortium and in providing some resources and tools for our research and development.

References

Boonkwan, P., Supnithi, T., 2008. *Memory-inductive categorial grammar: an approach to gap resolution in analytic-language translation*, To be presented in IJCNLP 2008.

Cotsomrong, P., Saykham, K., Wutiwiwatchai, C., Sreratanaprapahd, S., Songwattana, K., 2007. *A Thai spontaneous telephone speech corpus and its applications to speech recognition*, O-COCOSDA 2007.

Hansakunbuntheung, C., Tesprasit, V., Siricharoenchai, R., Sagisaka, Y., 2003. *Analysis and modeling of syllable duration for Thai speech synthesis*, EUROSPEECH 2003, pp. 93-96.

Hansakunbuntheung, C., Thangthai, A., Wutiwiwatchai, C., Siricharoenchai, R., 2005. *Learning methods and features for corpus-based phrase break prediction on Thai*, EUROSPEECH 2005, pp. 1969-1972.

Kanokphara, S., 2003. *Syllable structure based phonetic units for context-dependent continuous Thai speech recognition*, EUROSPEECH 2003, pp. 797-800.

Kasuriya, S., Sornlertlamvanich, V., Cotsomrong, P., Jitsuhiro, T., Kikui, G., Sagisaka, Y., 2003a. *NECTEC-ATR Thai speech corpus*, O-COCOSDA 2003.

Kasuriya, S., Sornlertlamvanich, V., Cotsomrong, P., Kanokphara, S., Thatphithakkul, N., 2003b. *Thai speech corpus for speech recognition*, International Conference on Speech Databases and Assessments (Oriental-COCOSDA).

Kikui, G., Sumita, E., Takezawa, T., Yamamoto, S., 2003. *Creating corpora for speech-to-speech translation*, EUROSPEECH 2003.

Tesprasit, V., Charoenpornsawat, P., Sornlertlamvanich, V., 2003. *Learning phrase break detection in Thai text-to-speech*, EUROSPEECH 2003, pp. 325-328.

Rugchatjaroen, A., Thangthai, A., Saychum, S., Thatphithakkul, N., Wutiwiwatchai, C., 2007. *Prosody-based naturalness improvement in Thai unit-selection speech synthesis*, ECTI-CON 2007, Thailand.

Thangthai, A., Hansakunbuntheung, C., Siricharoenchai, R., Wutiwiwatchai, C., 2006. *Automatic syllable-pattern induction in statistical Thai text-to-phone transcription*, INTERSPEECH 2006.

Thangthai, A., Wutiwiwatchai, C., Ragchatjaroen, A., Saychum, S., 2007. *A learning method for Thai phonetization of English words*, INTERSPEECH 2007.

Thatphithakkul, N., Kruatrachue, B., Wutiwiwatchai, C., Marukatat, S., Boonpiam, V., 2006. *A simulated-data adaptation technique for robust speech recognition*, INTERSPEECH 2006.

Wutiwiwatchai, C., 2007. *Toward Thai-English speech translation*, International Symposium on Universal Communications (ISUC 2007), Japan.

Wutiwiwatchai, C., Saychum, S., Rugchatjaroen, A., 2007. *An intensive design of a Thai speech synthesis corpus*, To be presented in International Symposium on Natural Language Processing (SNLP 2007).

Phrase-based Machine Transliteration

Andrew Finch
NiCT-ATR
"Keihanna Science City"
Kyoto, JAPAN
andrew.finch@atr.jp

Eiichiro Sumita
NiCT-ATR
"Keihanna Science City"
Kyoto, JAPAN
eiichiro.sumita@atr.jp

Abstract

This paper presents a technique for transliteration based directly on techniques developed for phrase-based statistical machine translation. The focus of our work is in providing a transliteration system that could be used to translate unknown words in a speech-to-speech machine translation system. Therefore the system must be able to generate arbitrary sequence of characters in the target language, rather than words chosen from a pre-determined vocabulary. We evalauted our method automatically relative to a set of human-annotated reference transliterations as well as by assessing it for correctness using human evaluators. Our experimental results demonstrate that for both transliteration and back-transliteration the system is able to produce correct, or phonetically equivalent to correct output in approximately 80% of cases.

1 Introduction

Dictionaries and corpora are only able to cover a certain proportion of language. Those words and phrases that are unknown to a translator/machine translation system present a problem. Examples of such words include people's names, place names, and technical terms. One solution to the problem is to transcribe the source language and use the transcription directly in the target language. Usually these transcrptions will be phonetically similar. This process of transcription is known as *transliteration* and in this paper we will present a technique for automatically transliterating between English and Japanese, although the technique is general and is able to be appied directly to other language pairs. Of particular interest to us is the application of such a system within a speech-to-speech machine translation (MT) system. Typically words not seen by the MT system, known as out of vocabulary words (OOVs), are either left untranslated or simply removed from the output. Common examples of OOVs are named entities such as personal names, place names and technical terms, unknown occurences of which could benefit from being transliterated into the MT system's output during translation between Japanese and English. Moreover, in the case of a transation system that translates directly to speech, the transliteration system does not necessarily need to produce the correct transliteration as any one of a set of phonetically equivalent alternatives would be equally acceptable.

1.1 English-Japanese Transliteration

In Japanese there are three separate alphabets, *kanji* (the Chinese character set), *hiragana* (used as an alternative to the kanji, and to express functional elements such as particles etc.) and *katakana* (used to express foreign loan words, and relatively new words in the language, for example "karaoke"). Figure 1 shows some examples, the first line is the English source, the second line is the Japanense and the last line is a direct transcription of the Japanese katakana into the roman alphabet with spaces delimiting the character boundaries. As can be seen from the examples, transliteration is not a straghtforward process. Example 1 of Figure 1 shows an example of a transliteration which is a reasonably direct phonetic transfer. The word "manga" in English is a loan word from Japanese and has more-or-less the same pronunciation in both languages. In Example 2 we have an ambiguity, the "aa" at the end of the word *kompyutaa*, corresponds to the

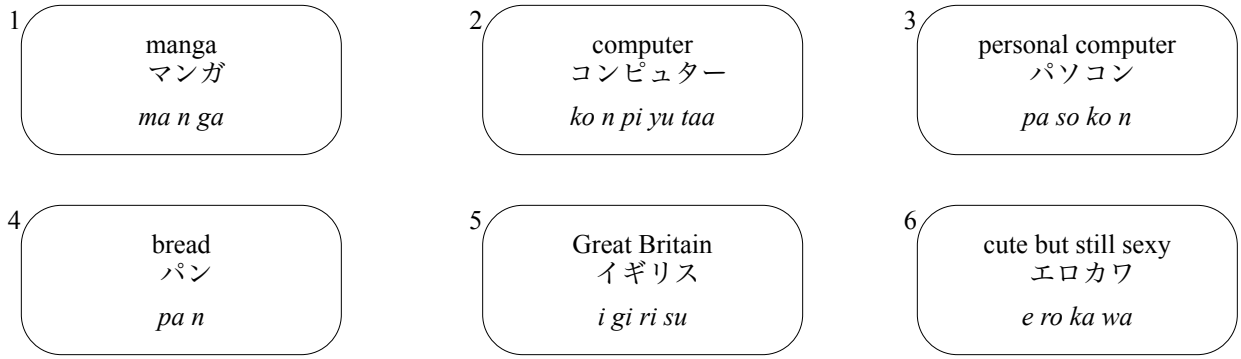

Figure 1: Example English-Japanese Transliterations

"er" of "computer". However, although incorrect the sequences *kompyuta* or *kompyuuta* are also plausible transliterations for the word. Example 4 shows a contraction. The English word has been transfered over into Japanese, and then shortened. In this case "personal" has been shortened to *paso* and "computer" has been contracted into *con*. In Example 4 the Japanese loan word has come from a language other than English, in this case French, and these words are usually transliterated according to the pronunciation in their native language. In Example 5, the etymology is quite complex. The word has entered the language from the Portugese for "English": *inglese*, but has come to mean "Great Britain". Example 6 is a creative modern mixture of an imported loan word *ero* a contraction of the transliteration *erochikku* of the English word "erotic", concatenated with a contraction of the Japanese word *kawaii* (usually written in kanji/kana) meaning "cute". Not only is the English phrase phonetically unrelated in this case, but the expression is difficult to translate without using a number of English words since it represents quite a lot of information.

2 Related Work

2.1 Machine Transliteration

This paper is directly related to an important paper by Knight and Graehl (1996). Their transliteration system was also evaluated by English-Japanese (back-)transliteration performance. Our system differs from theirs in a number of aspects. The most important of which is that their system outputs word sequences whereas our system outputs character sequences in the target language. The difference reflects the intended application of the transliteration system. Their system was intended to transliterate from the output of an OCR system, and must therefore be robust to errors in the input, whereas our system has been developed with machine translation in mind, and the input to our system is likely to consist of out-of-vocabulary words. This flexibility is a double-edged sword in that: on the one hand our system is able to handle OOVs; whereas on the other hand our system is free to generate non-words. A second difference between the approaches is that, Knight and Graehl's model models the pronunciation of the source word sequences using a pronunciation dictionary in an intermediate model. Our system transforms the character sequence from one language into another in a subword-level character sequence-based manner. Our systems relies on the the system being able to implicitly learn the correct character-sequence mappings through the process of character alignment. Our system is also able to re-order the translated character sequences in the output. The system can be easily constrained to generate the target in the same order as the source if necessary, however, often in Japanese names (including foreign names) are written with the family name first, therefore for the purposes of our experiments we allow the system to perform reordering.

2.2 Phrase-based Statistical Machine Translation (SMT)

Our approach couches the problem of machine transliteraion in terms of a character-level translation process. Character-based machine translation has been proposed as a method to overcome segmention issues in natural language

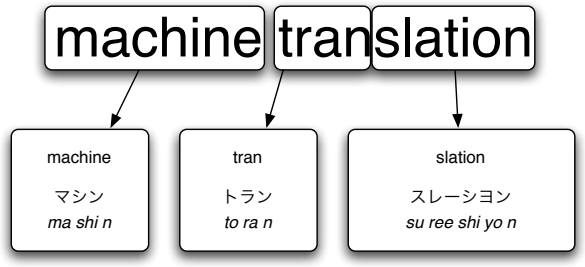

Figure 3: The phrase-translation process

processing (Denoual and Lepage, 2006) and character-based machine translation systems have already been developed on these principles (Lepage and Denoual, 2006). Our system also takes a character-based approach but restricts itself to the translation of short phrases. This is to our advantage because machine translation systems struggle in the translation of longer sequences. Moreover, the process of transliteration tends to be a monotone process, and this assists us further. We will give only a brief overview of the process of phrase-based machine translation, for a fuller account of statistical machine translation we refer the reader to (Brown et al., 1991) and (Koehn, Och, and Marcu, 2003).

During the process of phrase-based SMT the source sequence is segmented into sub-sequences , each sub-sequence being translated using bilingual sequence pairs (called phrase pairs when the translation proceeds at the word-level). The target generation process (for English-to-Japanese) at the character level is illustrated in Figure 3. The example is a real system output from an unseen phrase. The source sequence is segmented by the system into three segments. The translations of each of these segments have been gleaned from alignments of these segments where they occur in the training corpus. For example "machine⇒マシン" may have come from the pair "Turing machine⇒チューリングマシン (*chi yuu ri n gu ma chi n*)" that is present in the Wikipedia component of the training corpus. The "slation" in this example certainly came from the film title "Lost in Translation" since the Japanese translation of the English word "translation" is usually written in kanji.

3 Experimental Methodology

3.1 Experimental Data

The data for these experiments was taken from the publicly available EDICT dictionary (Breen, 1995) together with a set of katakana-English phrase pairs extracted from inter-language links in the Wikipedia[1]. These phrase pairs were extracted in a similar fashion to (Erdmann, et al., 2007) who used them in the construction of a bilingual dictionary. An inter-language link is a direct link from an article in one language to an article in another. Phrase-pairs are extracted from these links by pairing the titles of the two articles. We collected only phrase pairs in which the Japanese side consisted of only katakana and the English side consisted of only ASCII characters (thus deliberately eliminating some foreign language "English" names that would be hard to transliterate correctly). Data from both sources was combined to make a single corpus. Thus corpus was then randomly sub-divided into training (33479 phrase pairs), development (2000 phrase pairs) and evaluation (2000 phrase pairs) sub-corpora. For the human evaluation a sample of 200 phrase-pairs was chosen randomly from the test corpus. In addition a small corpus of 73 US politicians' names was collected from a list of US presidents and vice presidents in the Wikipedia. Duplictate entries were removed from this list and the training set was also filtered to exclude these entries.

3.2 Back-transliteration Accuracy

Following Knight and Graeh (1996), we evaluated our system with respect to back-transliteration performance. That is, word sequences in katakana were used to generate English sequences. As a point of reference to the results in this paper, we back-transliterated a list of American polititians' names. The results are shown in Table 1. The number of exacty correct results is lower than the system of Knight and Graehl, but the total number of correct + phonetically equivalent results is about the same. This can be explained by the fact that our system is able to generate character sequences more freely in order to be able to handle unknown words. Altogether around 78% of the back-transliterations were judged either correct or phonetically equivalent to a correct result. We included a class to respesent those results that were not equivalent in terms of English phonology but

[1] http://www.wikipedia.org

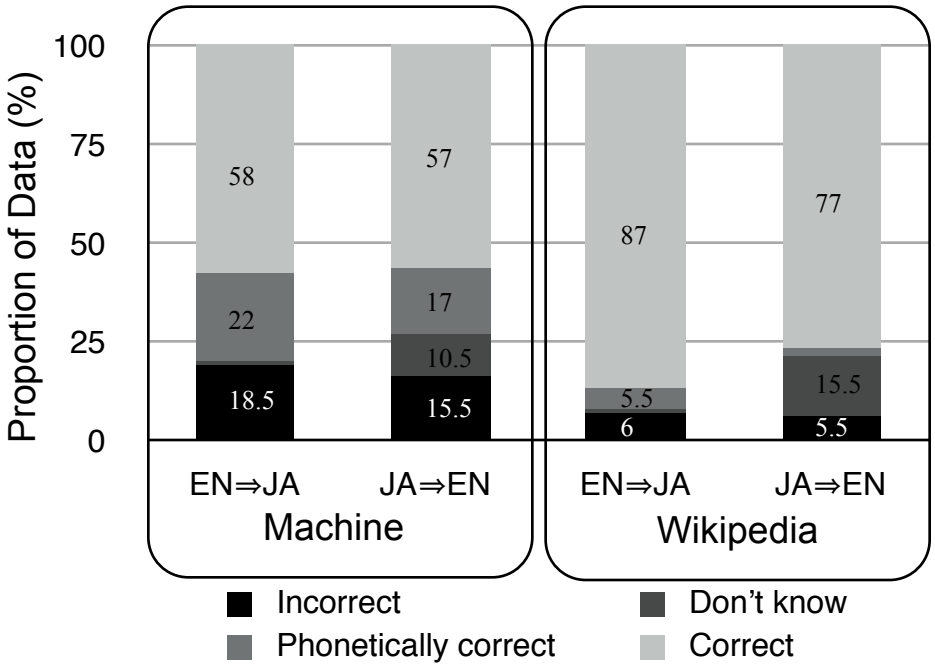

Figure 2: Human Judgement of Quality Transliteration Performance and Wikipedia Data

were "reasonable errors" in terms of Japanese phonology, for example "James Polk" was back-transliterated as "James Pork", the "r" and "l" sound being hard to discrimitate in Japanese becasue the two sounds are combined into a single sound. The reason for making this distinction was to identify the proportion (around 10%) of more pathological errors caused by errors such as incorrect phrase pairs extracted due to erroneaous word alignments.

3.3 Human Assessment

Figure 2 shows the results of the human evaluation. Transliterated text from English to Japanese was graded by a professional translator who was fluent in both languages but native in Japanese. Conversely the back-transliterated phrases were judged by a native English-speaking translator who was also fluent in Japanese. The evaluation data was graded into 4 categories:

(1) The transliteration or back-transliteration was correct.

(2) The transliteraton was not correct however the result was phonetically equivalent to a correct result.

Correct	57.53%
Phonetically equivalent (EN)	20.54%
Phonetically equivalent (JA)	10.96%
Incorrect	10.96%

Table 1: Back-transliteration performance on politicians' names

(3) The transliteration or back-transliteration was incorrect.

(4) The annnotator was unsure of the correct grade for that example.

Transliteration examples:

Grade 1: worm gear ⇒ *u oo mu gi ya*
Grade 2: worm gear ⇒ *waa mu gi a*
Grade 3: marcel desailly ⇒ *ma ru se ru de sa i ri*
Grade 4: agnieszka holland ⇒ ?

	BLEU	NIST	WER	PER	GTM	METEOR	TER
EN⇒JA	0.627	9.17	0.31	0.29	0.8	0.81	30.67
JA⇒EN	0.682	10.023	0.277	0.237	0.83	0.81	27.14

Table 2: System performance according to automatic machine translation scoring schemes

The example of Grade 1 is the Wikipedia entry and is the normal way of expressing this phrase in Japanese. The Grade 2 example is output from our system, the pronunciation of the string is almost the same as the Grade 1 version, however the form of expression is unusual. The Grade 3 example is also a system output. Here the system has made a reasonable attempt at generating the katakana, but has transliterated it in terms of the English pronunciation rather than the French from which the name dervies. The correct transliteration from this name would be: *ma ru se ru de sa ii*. This problem has been caused by the nature of the training data which contains mainly English expressions. The word "desailly" had not occurred in the training data.

The results reveal several things about the data, the task and the system performance. Looking at the scoring of the Wikipedia data, there is a reasonable level of disagreement between the two annotators, but the overall number of pairs judged as correct (back-)transliterations is nonetheless reasonably high; in the 80-90% range. Secondly, the annotators judged the quality of the transliteration and back-transliteration systems to be approximately the same. We found this result surprising since the English generation, intuitively at least, appears to be harder than Japanese generation because there are fewer constraints on graphemic structure. The most significant result is that the number of cases labelled "correct" or "phonetcially equivalent to a correct result" was around 80% for both systems, which should be high enough to allow the system to be used in a speech translation system, especially since by visual inspection of the data, many of the the "incorrect" results were near misses that would be easy for a user of the system to understand. For example the transliteraton *ko-roo-ra* for "Corolla" was judged correct, however *ko-ro-ra* was judged incorrect and not phonetically equivalent.

3.4 Assement using automatic machine translation evaluation methods

Table 2 shows the results from evaluating the output of our transliteration and back-transliteration systems according to a range of commonly-used automatic machine translation scoring schemes. We believe these techniques are an effective way to evaluate transliteration quality, and are therefore provided here as a reference. The difference between the WER and PER scores is interesting here as the WER score takes sequence order into account when comparing to a reference whereas PER does not. There is a larger difference when the target is English indicating that this process has more issues related to character order.

4 Conclusion and Future Directions

This paper has demonstrated that transliteration can be done effectively by a machine translation system, and has quantified this empirically. It is clear that by leaving the system 'open' and free to generate any sequence of characters in the target language there is a price to pay since the system is able to generate non-words. On the other hand, restricting the system so that it is only able to generate words is for many applications unrealistic, and in particular it is necessary for the speech translation application this system has been developed for. Our results show that our system generates correct or phonetically correct transliterations around 80% of the time. This figure serves as a lower bound estimate for the proportion of practically useful transliterations it will produce. Perhaps a compromise between these two approaches can be achieved by introducing a lexically-based language model into the system in addition to the existing high-order character-based language model. Furthermore, we are also interested in investigating the use of the models generated by training our system in the process of word alignment for statistical machine translation, and as a precursor to this the models might be used in filtering the training data in a pre-processing stage. Lastly it is impor-

tant to mention that Wikipedia (which provided us with most of our corpus), is growing very rapidly, and considerably more training data for statistical transliteration systems should be available in the near future.

References

J.W. Breen. 1995. Building an electronic Japanese-English dictionary. *Japanese Studies Association of Australia Conference*. Queensland, Australia.

Peter Brown, S. Della Pietra, V. Della Pietra, and R. Mercer (1991). The mathematics of statistical machine translation: parameter estimation. Computational Linguistics, 19(2), 263-311.

Etienne Denoual and Yves Lepage. 2006. The character as an appropriate unit of processing for non-segmenting languages, *Proceedings of the 12th Annual Meeting of The Association of NLP*, pp. 731-734.

Maike Erdmann, Kotaro Nakayama, Takahiro Hara,and Shojiro Nishio. 2007. Wikipedia Link Structure Analysis for Extracting Bilingual Terminology. *IEICE Technical Committee on Data Engineering*. Miyagi, Japan.

Kevin Knight and Jonathan Graehl. 1997. Machine Transliteration. *Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, pp. 128-135, Somerset, New Jersey.

Yves Lepage and Etienne Denoual. 2006. Objective evaluation of the analogy-based machine translation system ALEPH. *Proceedings of the 12th Annual Meeting of The Association of NLP*, pp. 873-876.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical Phrase-Based Translation. *In Proceedings of the Human Language Technology Conference 2003* (HLT-NAACL 2003), Edmonton, Canada.

Development of Indonesian Large Vocabulary Continuous Speech Recognition System within A-STAR Project

Sakriani Sakti[1,2], **Eka Kelana**[3], **Hammam Riza**[4], **Shinsuke Sakai**[1,2]
Konstantin Markov[1,2], **Satoshi Nakamura**[1,2]
[1]National Institute of Information and Communications Technology, Japan
[2]ATR Spoken Language Communication Research Laboratories, Japan
[3]R&D Division, PT Telekomunikasi Indonesia, Indonesia
[4]Agency for the Assessment and Application of Technology, BPPT, Indonesia
{sakriani.sakti,shinsuke.sakai,konstantin.markov,satoshi.nakamura}@atr.jp,
eka_k@telkom.co.id, hammam@iptek.net.id

Abstract

The paper outlines the development of a large vocabulary continuous speech recognition (LVCSR) system for the Indonesian language within the Asian speech translation (A-STAR) project. An overview of the A-STAR project and Indonesian language characteristics will be briefly described. We then focus on a discussion of the development of Indonesian LVCSR, including data resources issues, acoustic modeling, language modeling, the lexicon, and accuracy of recognition. There are three types of Indonesian data resources: daily news, telephone application, and BTEC tasks, which are used in this project. They are available in both text and speech forms. The Indonesian speech recognition engine was trained using the clean speech of both daily news and telephone application tasks. The optimum performance achieved on the BTEC task was 92.47% word accuracy.

1 A-STAR Project Overview

The A-STAR project is an Asian consortium that is expected to advance the state-of-the-art in multilingual man-machine interfaces in the Asian region. This basic infrastructure will accelerate the development of large-scale spoken language corpora in Asia and also facilitate the development of related fundamental information communication technologies (ICT), such as multi-lingual speech translation, multi-lingual speech transcription, and multi-lingual information retrieval.

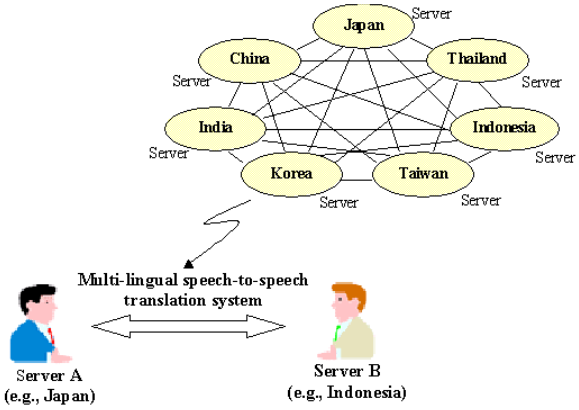

Figure 1: Outline of future speech-technology services connecting each area in the Asian region through network.

These fundamental technologies can be applied to the human-machine interfaces of various telecommunication devices and services connecting Asian countries through the network using standardized communication protocols as outlined in Fig. 1. They are expected to create digital opportunities, improve our digital capabilities, and eliminate the digital divide resulting from the differences in ICT levels in each area. The improvements to borderless communication in the Asian region are expected to result in many benefits in everyday life including tourism, business, education, and social security.

The project was coordinated together by the Advanced Telecommunication Research (ATR) and the

National Institute of Information and Communications Technology (NICT) Japan in cooperation with several research institutes in Asia, such as the National Laboratory of Pattern Recognition (NLPR) in China, the Electronics and Telecommunication Research Institute (ETRI) in Korea, the Agency for the Assessment and Application Technology (BPPT) in Indonesia, the National Electronics and Computer Technology Center (NECTEC) in Thailand, the Center for Development of Advanced Computing (CDAC) in India, the National Taiwan University (NTU) in Taiwan. Partners are still being sought for other languages in Asia.

More details about the A-STAR project can be found in (Nakamura et al., 2007).

2 Indonesian Language Characteristic

The Indonesian language, or so-called Bahasa Indonesia, is a unified language formed from hundreds of languages spoken throughout the Indonesian archipelago. Compared to other languages, which have a high density of native speakers, Indonesian is spoken as a mother tongue by only 7% of the population, and more than 195 million people speak it as a second language with varying degrees of proficiency. There are approximately 300 ethnic groups living throughout 17,508 islands, speaking 365 native languages or no less than 669 dialects (Tan, 2004). At home, people speak their own language, such as Javanese, Sundanese or Balinese, even though almost everybody has a good understanding of Indonesian as they learn it in school.

Although the Indonesian language is infused with highly distinctive accents from different ethnic languages, there are many similarities in patterns across the archipelago. Modern Indonesian is derived from the literary of the Malay dialect. Thus, it is closely related to the Malay spoken in Malaysia, Singapore, Brunei, and some other areas.

Unlike the Chinese language, it is not a tonal language. Compared with European languages, Indonesian has a strikingly small use of gendered words. Plurals are often expressed by means of word repetition. It is also a member of the agglutinative language family, meaning that it has a complex range of prefixes and suffixes, which are attached to base words. Consequently, a word can become very long.

More details on Indonesian characteristics can be found in (Sakti et al., 2004).

3 Indonesian Phoneme Set

The Indonesian phoneme set is defined based on Indonesian grammar described in (Alwi et al., 2003). A full phoneme set contains 33 phoneme symbols in total, which consists of 10 vowels (including diphthongs), 22 consonants, and one silent symbol. The vowel articulation pattern of the Indonesian language, which indicates the first two resonances of the vocal tract, F1 (height) and F2 (backness), is shown in Fig. 2.

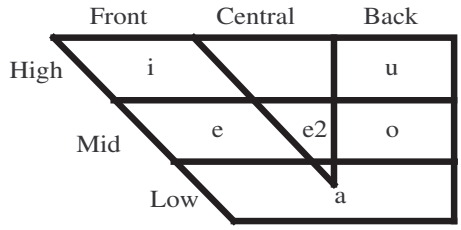

Figure 2: Articulatory pattern of Indonesian vowels.

It consists of vowels, i.e., /a/ (like “a” in “father”), /i/ (like “ee” in “screen”), /u/ (like “oo” in “soon”), /e/ (like “e” in “bed”), /e2/ (a schwa sound, like “e” in “learn”), /o/ (like “o” in “boss”), and four diphthongs, /ay/, /aw/, /oy/ and /ey/. The articulatory pattern for Indonesian consonants can be seen in Table 1.

4 Indonesian Data Resources

Three types of Indonesian data resources available in both text and speech forms were used here. The first two resources were developed or processed by the R&D Division of PT Telekomunikasi Indonesia (R&D TELKOM) in collaboration with ATR as continuation of the APT project (Sakti et al., 2004), while the third one was developed by ATR under the A-STAR project in collaboration with BPPT. They are described in the following.

Table 1: *Articulatory pattern of Indonesian consonants.*

	Bilabial	Labiodental	Dental/Alveolar	Palatal	Velar	Glotal
Plosives	p, b		t, d		k, g	
Affricates				c, j		
Fricatives		f	s, z	sy	kh	h
Nasal	m		n	ny	ng	
Trill			r			
Lateral			l			
Semivowel	w			y		

4.1 Text Data

The three text corpora are:

1. Daily News Task
   There is already a raw source of Indonesian text data, which has been generated by an Indonesian student (Tala, 2003). The source is a compilation from "KOMPAS" and "TEMPO", which are currently the largest and most widely read Indonesian newspaper and magazine. It consists of more than 3160 articles with about 600,000 sentences. R&D TELKOM then further processed them to generate a clean text corpus.

2. Telephone Application Task
   About 2500 sentences from the telephone application domain were also generated by R&D TELKOM, and were derived from some daily dialogs from telephone services, including tele-home security, billing information services, reservation services, status tracking of e-Government services, and also hearing impaired telecommunication services (HITSs).

3. BTEC Task
   The ATR basic travel expression corpus (BTEC) has served as the primary source for developing broad-coverage speech translation systems (Kikui et al., 2003). The sentences were collected by bilingual travel experts from Japanese/English sentence pairs in travel domain "phrasebooks". BTEC has also been translated into several languages including French, German, Italian, Chinese and Korean. Under the A-STAR project, there are also plans to collect synonymous sentences from the different languages of the Asian region. ATR has currently successfully collected an Indonesian version of BTEC tasks, which consists of 160,000 sentences (with about 20,000 unique words) of a training set and 510 sentences of a test set with 16 references per sentence. There are examples of BTEC English sentences and synonymous Indonesian sentences in Table 2.

Table 2: *Examples of English-Indonesian bilingual BTEC sentences.*

English	**Indonesian**
Good Evening	Selamat Malam
I like strong coffee	Saya suka kopi yang kental
Where is the boarding gate?	Di manakah pintu keberangkatan berada?
How much is this?	Harganya berapa?
Thank you	Terima kasih

4.2 Speech Data

The three speech corpora are:

1. Daily News Task
   From the text data of the news task described above, we selected phonetically-balanced sentences, then recorded the speech utterances. Details on the phonetically-balanced sentences, the recording set-up, speaker criteria, and speech utterances are described in what follows:

   - Phonetically-Balanced Sentences
     We selected phonetically-balanced sentences using the greedy search algorithm

(Zhang and S.Nakamura, 2003), resulting in 3168 sentences in total (see Table 3).

Table 3: *Number of phonetically-balanced sentences resulting from greedy search algorithm.*

Phone	# Units	# Sentences
Monophones	33	6
Left Biphones	809	240
Right Biphones	809	242
Triphones	9667	2978
Total		3168

- Recording Set-Up
  Speech recording was done by R&D TELKOM in Bandung, Java, Indonesia. It was conducted in parallel for both clean and telephone speech, recorded at respective sampling frequency of 16 kHz and 8 kHz. The system configuration is outlined in Fig. 3.

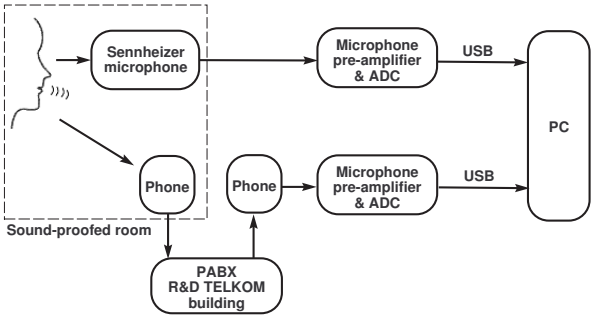

Figure 3: Recording set-up.

- Speaker Criteria
  The project will require a lot of time, money, and resources to collect all of the possible languages and dialects of the tribes recognized in Indonesia. In this case, R&D TELKOM only focused on the major ethnic accents in Bandung area where the actual telecommunication services will be implemented. Four main accents were selected, including: Batak, Javanese, Sundanese, and standard Indonesian (no accent) with appropriate distributions as outlined in Fig. 4. Both genders are evenly distributed and the speakers' ages are also distributed as outlined in Fig. 5. The largest percentage is those aged 20-35 years who are expected to use the telecommunication services more often.

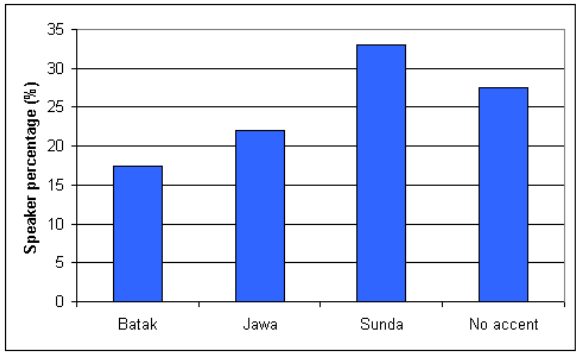

Figure 4: Accent distribution of 400 speakers in daily news and telephone application tasks.

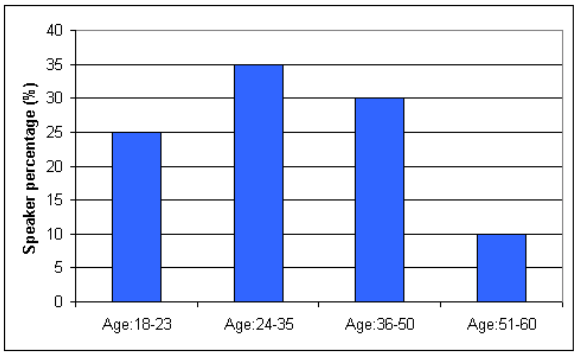

Figure 5: Age distribution of 400 speakers in daily news and telephone application tasks.

- Speech Utterances
  The total number of speakers was 400 (200 males and 200 females). Each speaker uttered 110 sentences resulting in a total of 44,000 speech utterances or about 43.35 hours of speech.

2. Telephone Application Task
   The utterances in speech of 2500 telephone application sentences were recorded by R&D TELKOM in Bandung, Indonesia using the same recording set-up as that for the news task

corpus. The total number of speakers, as well as appropriate distributions for age and accent, were also kept the same. Each speaker uttered 100 sentences resulting in a total of 40,000 utterances (36.15 hours of speech).

3. BTEC Task
   From the test set of the BTEC text data previously described, 510 sentences of one reference were selected and the recordings of speech were then done by ATR in Jakarta, Indonesia. BPPT helped to evaluate the preliminary recordings. For this first version, we only selected speakers who spoke standard Indonesian (no accent). There were 42 speakers (20 males and 22 females) and each speaker uttered the same 510 BTEC sentences, resulting in a total of 21,420 utterances (23.4 hours of speech).

5 Indonesian Speech Recognizer

The Indonesian LVCSR system was developed using the ATR speech recognition engine. The clean speech of both daily news and telephone application tasks were used as the training data, while the BTEC task was used as an evaluation test set. More details on the parameter set-up, acoustic modeling, language modeling, pronunciation dictionary and recognition accuracy will be described in the following.

5.1 Parameter Set-up

The experiments were conducted using feature extraction parameters, which were a sampling frequency of 16 kHz, a frame length of a 20-ms Hamming window, a frame shift of 10 ms, and 25 dimensional MFCC features (12-order MFCC, Δ MFCC and Δ log power).

5.2 Segmentation Utterances

Segmented utterances according to labels are usually used as a starting point in speech recognition systems for training speech models. Automatic segmentation is mostly used since it is efficient and less time consuming. It is basically produced by forced alignment given the transcriptions. In this case, we used an available Indonesian phoneme-based acoustic model developed using the English-Indonesian cross language approach (Sakti et al., 2005).

5.3 Acoustic Modeling

Three states were used as the initial HMM for each phoneme. A shared state HMnet topology was then obtained using a successive state splitting (SSS) training algorithm based on the minimum description length (MDL) optimization criterion (Jitsuhiro et al., 2004). Various MDL parameters were evaluated, resulting in context-dependent triphone systems having different version of total states. i.e., 1,277 states, 1,944 states and 2,928 states. All triphone HMnets were also generated with three different versions of Gaussian mixture components per state, i.e., 5, 10, and 15 mixtures.

5.4 Language Modeling

Word bigram and trigram language models were trained using the 160,000 sentences of the BTEC training set, yielding a trigram perplexity of 67.0 and an out-of-vocabulary (OOV) rate of 0.78% on the 510 sentences of the BTEC test set. This high perplexity could be due to agglutinative words in the Indonesian language.

5.5 Pronunciation Dictionary

About 40,000 words from an Indonesian pronunciation dictionary were manually developed by Indonesian linguists and this was owned by R&D TELKOM. This was derived from the daily news and telephone application text corpora, which consisted of 30,000 original Indonesian words plus 8,000 person and place names and also 2,000 of foreign words. Based on these pronunciations, we then included additional words derived from the BTEC sentences.

5.6 Recognition Accuracy

The performance of the Indonesian speech recognizer with different versions of total states and Gaussian mixture components per state is graphically depicted in Fig. 6. On average, they achieved 92.22% word accuracy. The optimum performance was 92.47% word accuracy at RTF=0.97 (XEON 3.2 GHz), which was obtained by the model with 1.277 total states and 15 Gaussian mixture components per state.

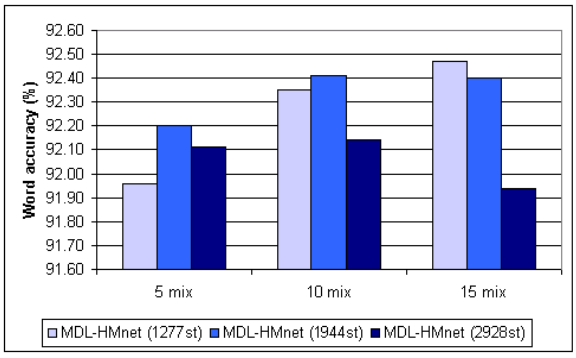

Figure 6: Recognition accuracy of Indonesian LVCSR on BTEC test set.

6 Conclusion

We have presented the results obtained from the preliminary stages of an Indonesian LVCSR system. The optimum performance achieved was 92.47% word accuracy at RTF=0.97. A future development will be to implement it on a real speech-to-speech translation system using computer terminals (tablet PCs). To further refine the system, speaker adaptation as well as environmental or noise adaptation needs to be done in the near future.

References

H. Alwi, S. Dardjowidjojo, H. Lapoliwa, and A.M. Moeliono. 2003. *Tata Bahasa Baku Bahasa Indonesia (Indonesian Grammar)*. Balai Pustaka, Jakarta, Indonesia.

T. Jitsuhiro, T. Matsui, and S. Nakamura. 2004. Automatic generation of non-uniform HMM topologies based on the MDL criterion. *IEICE Trans. Inf. & Syst.*, E87-D(8):2121–2129.

G. Kikui, E. Sumita, T. Takezawa, and S. Yamamoto. 2003. Creating corpora for speech-to-speech translation. In *Proc. EUROSPEECH*, pages 381–384, Geneva, Switzerland.

S. Nakamura, E. Sumita, T. Shimizu, S. Sakti, S. Sakai, J. Zhang, A. Finch, N. Kimura, and Y. Ashikari. 2007. A-star: Asia speech translation consortium. In *Proc. ASJ Autumn Meeting*, page to appear, Yamanashi, Japan.

S. Sakti, P. Hutagaol, A. Arman, and S. Nakamura. 2004. Indonesian speech recognition for hearing and speaking impaired people. In *Proc. ICSLP*, pages 1037–1040, Jeju, Korea.

S. Sakti, K. Markov, and S.Nakamura. 2005. Rapid development of initial indonesian phoneme-based speech recognition using cross-language approach. In *Proc. Oriental COCOSDA*, pages 38–43, Jakarta, Indonesia.

F. Tala. 2003. *A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia*. Ph.D. thesis, The Information and Language System (ILPS) Group, Informatics Institute, University of Amsterdam, Amsterdam, Netherland.

J. Tan. 2004. Bahasa indonesia: Between faqs and facts. http://www.indotransnet.com/article1.html.

J. Zhang and S.Nakamura. 2003. An efficient algorithm to search for a minimum sentence set for collecting speech database. In *Proc. ICPhS*, pages 3145–3148, Barcelona, Spain.

Using Confidence Vector in Multi-Stage Speech Recognition

Hyungbae Jeon, Kyuwoong Hwang, Hoon Chung, Seunghi Kim, Jun Park, Yunkeun Lee
Speech/Language Information Research Center
Electronics and Telecommunications Research Institute (ETRI)
161 Gajeong-dong, Yuseong-gu, Daejeon, 305-350, Korea
{hbjeon, kyuwoong, hchung, seunghi, junpark, yklee}@etri.re.kr

Abstract

This paper presents a new method of using confidence vector as an intermediate input feature for the multi-stage based speech recognition. The multi-stage based speech recognition is a method to reduce the computational complexity of the decoding procedure and thus accomplish faster speech recognition. In the multi-stage speech recognition, however, the error of previous stage is transferred to the next stage. This tends to cause the deterioration of the overall performance. We focus on improving the accuracy by introducing confidence vector instead of phoneme which typically used as an intermediate feature between the acoustic decoder and the lexical decoder, the first two stages in the multi-stage speech recognition. The experimental results show up to 16.4% error reduction rate(ERR) of word accuracy for 220k Korean Point-of-Interest (POI) domain and 29.6% ERR of word accuracy for hotel reservation dialog domain.

1 Introduction

Recently, demand of fast and small size speech recognition engine is increasing. One example is the mobile automatic speech translation device. To implement the speech translation function, we need to integrate the speech recognition, the text translation, and the speech synthesis modules all together into a single device. Another example is recognition of Point-of-Interests (POIs) on navigation device. The number of POIs used in a commercial navigation system is about several hundreds of thousands to one million. To make a fast and more efficient speech recognition engine, many computationally efficient methods were reported and multi-stage speech recognition is one of competitive approaches [1][2].

The multi-stage speech recognition completes recognition procedure through sequential two stage decoding, the acoustic decoding and the lexical decoding. The optional rescoring stage can follow the lexical decoding, depending on the application. In the acoustic decoding stage, we find the phoneme sequence which has the maximum likelihood for a sequence of input acoustic feature vector. In the lexical decoding, we recover the optimal word sequence whose phone sequence has a minimal edit distance from an input phoneme sequence from the acoustic decoding stage.

Comparing to the one-stage speech recognition, the multi-stage speech recognition can significantly reduce computation load. This is because the lexical decoding stage in the multi-stage speech recognition is repeated for every phoneme in the input phoneme sequence, while the lexical decoding stage in the one-stage speech recognition is repeated for typically every ten milliseconds. Since the duration of a phoneme is up to several hundred milliseconds, this computational saving is quite big, especially for cases with a large lexical search space. However, the multi-stage speech recognition has a shortcoming that the overall performance is highly affected by accuracy level of the phoneme recognizer. Moreover, the performance of the phoneme recognizer is generally poor since it depends only on the acoustic feature without using any higher level of information like a language model.

In this paper, we propose a new method of using a confidence vector as an input feature for lexical decoding stage to improve performance of multi-stage speech recognition system. For a phoneme segment, we introduce a confidence vector in which each element is defined as the likelihood of

each phoneme for the given segment. And, we use this confidence vector as the input feature for the lexical decoder rather than just the phoneme sequence as in the conventional multi-stage speech recognition. This means that more information is transferred from acoustic decoder to lexical decoder.

This paper is organized as follows. At first, we present the structure of multi-stage speech recognition in section 2. In section 3, we describe the proposed method that uses confidence vector as an input feature of lexical decoding. In section 4, we explain the experiment environment and results. Finally, we summarize our work in section 5.

2 Multi-Stage Speech Recognition

Multi-stage speech recognition system is composed of acoustic decoder, lexical decoder and optional n-best rescoring [2][3].

The purpose of acoustic decoding is to convert an input acoustic feature sequence into a corresponding phoneme sequence as correctly as possible while minimizing the consumption of computational power. In most cases, CDHMM-based automatic phone recognizer is used for acoustic decoding. In this work, we also used automatic phone recognizer for the acoustic decoding, in which CDHMMs corresponding to 46 Korean phonemes including silence are used and a finite state network representing Korean syllable composition structure is used for pronunciation model.

The purpose of the lexical decoding is to recover an optimal word sequence from input phoneme sequence given as a result of acoustic decoding. Acoustic decoder may generate three types of phone errors, substitution, insertion and deletion for a reference phoneme due to inaccurate acoustic modeling, surrounding noises and so on. We model these error patterns in the form of probabilistic edit cost. For each speech utterance in training DB, the lexical decoder aligns phoneme sequence from the acoustic decoder with the phoneme sequence of reference words by dynamic time warping (DTW). By accumulating the alignment results for all the utterances in training DB, we obtain the substitution probability of each phoneme. While decoding, the cost at each node is derived from the substitution probability of each phoneme.

3 Confidence Vector Based Approach

In this paper, we introduce a phoneme confidence vector as an input feature for lexical decoding. This confidence vector based approach comprises four stages, phoneme segmentation, confidence vector extraction, lexical decoding, n-best rescoring. Figure 1 illustrates a block diagram of the multi-stage speech recognition using confidence vector.

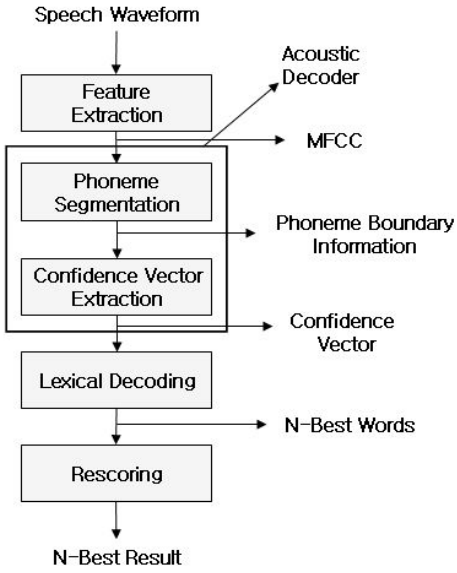

Figure 1. Block diagram of multi-stage speech recognition using confidence vector.

Firstly, the automatic phoneme recognizer finds optimal phoneme boundary information for an input utterance and then every acoustic likelihood corresponding to each of 46 Korean phoneme units is calculated for each segment, according to the phone segmentation result. Secondly, we define the confidence value for each phoneme at a given phoneme segment as follows,

$$confidence(i) = \frac{likelihood(i)}{\sum_{j=1}^{N} likelihood(j)} \quad (1)$$

where confidence(i) means confidence value of i'th phoneme and N is the number of phonemes.

Because lexical decoding is a sort of dynamic programming, we need to define a local match cost to weight distance between two phonemes. For confidence vector approach, we defined lexical

model as mean of confidence vectors of each phoneme. The cost of DTW is defined with mean of confidence vectors of reference phoneme and confidence vectors of input speech segment. We use three cost definitions that represent distance between two confidence vectors.

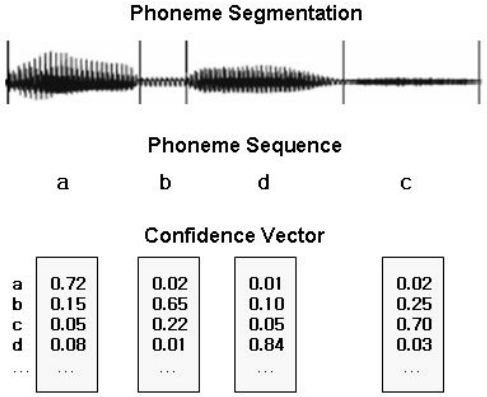

Figure 2. An input feature for lexical decoding, phoneme sequence and confidence vectors

3.1 Kullback-Leibler Divergence

Kullback-Leibler divergence measures a difference between two probability distributions. If we regard the confidence vector as a probability distribution, we can define the difference as cost. Equation 2 is a cost definition by Kullback-Leibler divergence for reference phoneme 'p'.

$$\cos t(f \mid p) = \sum_{i=1}^{N} \left(f(i) \times \log \frac{f(i)}{m_p(i)} \right) \quad (2)$$

In equation 2, f(i) is confidence value of i'th phoneme and mp(i) is i'th value of mean vector of phoneme 'p'.

3.2 Weighted Sum of Confidence

The second definition of cost is weighted sum of confidence. As weight values, mean of confidence vector that was previously trained is used. This weighted sum of confidence corresponds to substitution probability with consideration of prior probability distribution. It takes –log to make cost value from probability. Equation 3 is a cost definition by weighted sum of confidence.

$$\cos t(f \mid p) = -\log \left(\sum_{i=1}^{N} \left(m_p(i) \times f(i) \right) \right) \quad (3)$$

In equation 3, weight term can be considered as loss term of minimum risk decoding [4]. If the weight value is loss quantity of decision to phoneme 'p' with confidence f(i), decision for minimum cost is identical to decision for minimum risk.

3.3 Smoothing of Confidence

Even though we use weighted sum of confidence instead of substitution probability of best phone sequence, the cost is highly sensitive to the result of acoustic decoder. To prevent extreme decision of phoneme recognizer, we add smoothing parameters, α and β, like equation 4. α and β are assigned to be less then 1 and this gives effect to emphasize small values of probability.

$$\cos t(f \mid p) = -\log \left(\sum_{i=1}^{N} \left(m_p(i)^{\alpha} \times f(i)^{\beta} \right) \right) \quad (4)$$

4 Experiment Results

4.1 Experiment on POI domain

To evaluate the performance of the proposed algorithm, we performed experiments on Korean POI task domain which is an isolate word recognition composed of 220k POI entries. Test set comprises with 7 speakers, each speaker makes 99 utterances, so total test set is 693 utterances.

The speech signal was sampled at 16kHz, and the frame length was 20ms with 10ms shift. Each speech frame was parameterized as a 39-dimentional feature vector containing 12 Mel-Frequency Cepstral Coefficients (MFCCs), C0 energy, their delta and delta-delta coefficients. In order to evaluate the robustness of the proposed algorithm on acoustically, lexically mismatched condition, we took the experiments in two different test environment where N-fold test means the test is done in lexically matched condition and open test in acoustically and lexically mismatched condition.

N-fold Test

In order to reflect phoneme error patterns to the lexical decoding, we used the phoneme result of 6 speaker's utterance to train lexical model. We performed test for remained one speaker data, and we did 7 times with different test speakers. Each case training data of lexical model is 594 utterances and test data is 99 utterances. Lexical model is substi-

tution probability or mean of confidence vector for reference phoneme.

Because training data of lexical model has the same vocabulary with test data and the same channel condition, training of lexical model in N-fold test can train error pattern of same vocabulary and environment condition. Table 1 shows the result of N-fold test experiment. Conventional multi-stage speech recognition method uses phoneme sequence as an input feature for lexical decoding, and its performance is the lowest among the rest methods. Because the phoneme accuracy of acoustic decoder is about 60~70% for this open test set, the performance of conventional method is degraded. If we use confidence vector as input feature for lexical decoding, the performance is increased by almost 10%. This performance enhancement proves that the confidence vector contains more useful information at lexical decoding. The cost definition with weighted sum of confidence has a similar performance with Kullback-Leibler divergence that is a well known distance measure. Especially with smoothing of confidence we get the best performance, but smoothing makes the search network enlarge and slow down a little because smoothing reduces the differences between confidence vectors.

Lexical Feature	Cost Definition	N-best Word Recognition Rate (%)		
		1	5	10
Phoneme Sequence	Substitution Prob.	64.5	75.6	78.6
Confidence Vector	KL Divergence	74.5	88.7	91.8
	Weighted sum of Confidence	74.5	88.5	92.4
	Smoothing of Confidence	78.8	90.9	93.8

Table 1. Word Recognition Rate of N-fold test

Lexical Model Training with Open set

In N-fold test, environment condition and error pattern of certain vocabulary is trained although we do not intend to do so. For an open test, we trained lexical model using word DB. We used ETRI phonetically optimized word (POW) DB which has 92k utterances.

Table 2 shows the results of open test experiments. If we use phoneme sequence as input feature for lexical decoder, performance is better than N-fold test. Because training DB for lexical model is enlarged to 92k, substitution probability might be modeled correctly. But confidence vector approach shows some performance degradations from N-fold test. In N-fold test, the performance is highly affected by the training of error pattern of certain vocabulary. In open test, KL divergence has better performance than cost definition by weighted sum of confidence in n-best aspect. And smoothing method has the best performance.

Lexical Feature	Cost Definition	N-best Word Recognition Rate (%)		
		1	5	10
Phoneme Sequence	Substitution Prob.	68.4	83.6	86.0
Confidence Vector	KL Divergence	71.4	86.6	90.9
	Weighted sum of Confidence	71.3	83.8	87.2
	Smoothing of Confidence	73.6	88.0	91.9

Table 2. Word Recognition Rate of Open test

N-best Rescoring

N-best rescoring stage does one-pass search among N-best result words. We used 500-best result for N-best rescoring. Table 3 shows the result of N-best rescoring. In the N-best rescoring test we used the lexical model from POW DB.

500-best performance of conventional multi-stage speech recognition system is 97.3%, and when we use confidence vector as input feature for lexical decoding with Kullback-Leibler divergence cost, our 500-best performance is 98.8% and 98.9% when we use weighted sum of confidence vector as cost. When we use smoothing method, 500-best performance is 99.1%. Since the performance of N-best rescoring is highly dependent with 500-best performance, conventional multi-stage method has the worst rescoring performance. Because 500-best performance of weighted sum of confidence and smoothing method are similar, the performance of N-best rescoring is almost same for these two cost definitions. If we use conventional one-pass decoder, word recognition rate is 82.9%. By N-best rescoring we can achieves the same level of performance with the one-pass decoder.

Lexical Feature	Cost Definition	N-best Word Recognition Rate (%)		
		1	5	10
Phoneme Sequence	Substitution Prob.	81.2	92.4	94.2
Confidence Vector	KL Divergence	82.0	94.2	95.5
	Weighted sum of Confidence	83.2	94.5	95.9
	Smoothing of Confidence	82.0	94.5	95.8

Table 3. Word Recognition Rate of N-best Rescoring

Speed Improvement

By dividing search procedure into acoustic and lexical parts, we can have the advantage of speed. In the acoustic decoder we used context-dependent model and a real-time factor of acoustic decoder is 0.03 on 3G Hz Dual-core CPU machine. Table 4 shows the recognition speed of multi-stage speech recognition system. Multi-stage approach is about 4-times faster than conventional one-pass decoder. If we use confidence vector as input feature for lexical decoding, we need additional amount of computation to extract confidence values. But the total speed is faster than conventional phoneme sequence feature. Since confidence vector feature makes more discriminative cost in lexical decoding, the total number of active nodes of search network is decreased and we can avoid increasing time.

Decoder/Lexical Feature		Real Time Factor
One-pass		1.16
Multi-pass	Phoneme Sequence	0.29
	Confidence Vector	0.27

Table 4. Speed of multi-stage speech recognition system

4.2 Experiment on hotel reservation domain

We applied the proposed method to the dialog speech recognition. The acoustic decoder is same as in the POI domain experiment. The lexical decoder searches the finite state network with class as its node which covers the 25,000 sentence templates in the hotel reservation domain. The test speech data is composed of 2,161 sentences uttered by 50 speakers, which is covered by the sentence templates. Table 5 shows that the use of the confidence vector is shown to be very effective, showing 29.6% error reduction rate to the conventional multi-stage method. Also, the proposed approach is shown to be competitive to the one-stage speech recognition while the execution speed is improved more than five times.

Decoder/Lexical Feature		Real Time Factor	Word Accuracy(%)
One-Pass		1.17	95.90
Multi-pass	Substitution Prob.	0.33	95.71
	KL Divergence	-	96.18
	Weighted sum of Confidence	0.22	96.39
	Smoothing of Confidence	-	96.98

Table 5. Word accuracy and speed of multi-stage speech recognition for hotel reservation domain

5 Conclusion

In this paper, we proposed a new method of using the confidence vector as an input feature for lexical decoding in the multi-stage speech recognition. We also introduced diverse cost definitions to measure the difference between confidence vector and reference vector. By transferring more information in the form of confidence vector to the lexical decoding, we can achieve better performance than the conventional method. The experiment results show up to 16.4% ERR of word accuracy for 220k Korean Point-of-Interest (POI) domain and 29.6% ERR of word accuracy for hotel reservation dialog domain.

Also, comparing to the one-pass decoder, the proposed method is shown to be far faster while maintaining competitive performance. Thus, this method is appropriate to build an embedded speech recognition engine on a mobile device.

As a future research, we will investigate the possibility of integrating the confidence vector approach with a phoneme lattice as an input feature for lexical decoding. A plain phoneme lattice can convey more information to lexical decoding, but mmay increase the computational complexity without any performance gain. We expect its advantage is maximized with combining the confidence vector approach.

References

[1] Victor Zue, James Glass, David Goodine, Michael Phillips, and Stephanie Seneff. The SUMMIT Speech Recognition System: Phonological Modeling and Lexical Access. In Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing, pp. 49-52, 1990.

[2] Hoon Chung, Ikjoo Chung. Memory efficient and fast speech recognition system for low resource mobile devices. In IEEE Transactions on Consumer Electronics, Volume: 52, Issue: 3, pages 792 – 796, 2006.

[3] Hyungbae Jeon, Kyuwoong Hwang, Hoon Chung, Seunghi Kim, Jun Park, and Yunkeun Lee. Multi-stage Speech Recognition for POI. In Proc. Conference of Korean Society of Phonetic Sciences and Speech Technology, pp. 131-134, 2007.

[4] Vaibhava Goel, Shankar Kuman, and William Byrne. Confidence Based Lattice Segmentation and Minimum Bayes-Risk Decoding. In Proc. Conference of Eurospeech, 2001.

Toward Asian Speech Translation System: Developing Speech Recognition and Machine Translation for Indonesian Language

Hammam Riza
IPTEKNET
Agency for the Assessment and Application of Technology
Jakarta, Indonesia
hammam@iptek.net.id

Oskar Riandi
ICT Center
Agency for the Assessment and Application of Technology
Jakarta, Indonesia
oskar@inn.bppt.go.id

Abstract

In this paper, we present a report on the research and development of speech to speech translation system for Asian languages, primarily on the design and implementation of speech recognition and machine translation systems for Indonesia language. As part of the A-STAR project, each participating country will need to develop each component of the full system for the corresponding language. We will specifically discuss our method on building speech recognition and stochastic language model for statistically translating Indonesian into other Asian languages. The system is equipped with a capability to handle variation of speech input, a more natural mode of communication between the system and the users.

1 Introduction

Indonesia is one of the ten most populous nations in the world with the population of about 235 million people as of 2004 and is located strategically within the Asia region. The exchange of people, goods and services as well as information increases and should not be hindered by language barrier. Even though, English language may be used as the main global communication language, the more direct and more natural way of communication is preferred by local and native people to ensure the smooth exchange of information among people of different languages.

It would be beneficial for Indonesia people, if there were a system that is able, to some extent in a certain domain, to capture either a speech or digital text based on Indonesian language and process it in order to output into meaningful text into other languages such as English, Japanese and other world languages. In addition to above mentioned benefit, large numbers of Indonesian people, statistically, have problem in using and comprehending any information presented in English language, The language barrier problem is compounded by the problem of the explosion of digital information whose majority uses English language via either Internet or any digital / printed form which may overwhelms potential users and pose a threat of inequality of access of information due to the language barrier (digital divide) especially for the common Indonesian people. We are now part of a multi national project to develop speech to speech translation system for Asian languages facilitated by ATR-Japan.

Our most recent work is focusing on developing Indonesian speech recognition engine and a statistical language model for machine translation. Our approach to MT is based on the integration of stochastic and symbolic approaches to be used for analyzing Indonesian. For creating the stochastic language model, it is worthwhile to utilize annotated data when it is available and use supervised learning mechanism to estimate model's parameter. In this case, an annotated corpus is created for multiple genres of documents. Of course, the costs of annotation are prohibitively labor intensive, and the resulting corpora sometimes are susceptible to a particular genre. Due to this limitation of annotated corpora, it is necessary that we use unsuper-

vised and weakly supervised learning techniques, which do not require large, annotated data sets.

Unsupervised learning utilizes raw, unannotated corpora to discover underlying language structure such as lexical and contextual relationships. This gives rise to emergent patterns and principles found in symbolic systems. In this system, the language model is trained using weakly supervised learning on small annotated corpus to seed unsupervised learning using much larger, unannotated corpora. Unsupervised and weakly supervised methods have been used successfully in several areas of NLP, including acquiring verb sub-categorization frames, part-of-speech tagging, word-sense disambiguation and prepositional phrase attachment.

The significant contribution of this preliminary research is the development of ASR using speaker adaptation technique and a statistical language model for translating from/to Indonesian language as well as Indo-Malay language in some extent. The major language found in Indonesia, Malaysia, Brunei, Singapore, Southern Thailand and Philippines can be categorized into a single root Indo-Malay language spoken in different dialects. Creating an ideal language model for Indo-Malay language is expected to be used by more than 260 million people in the region.

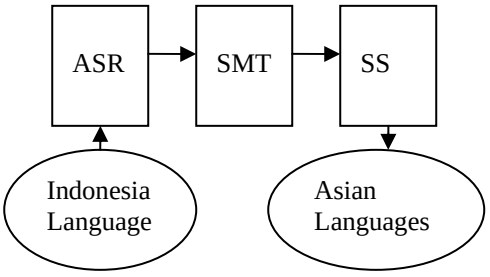

Figure 1. Scope of Indonesian-Asian Languages

2 Recognizing Indonesian Speech

Achievement of a high performance is often the most dominating design criterion when implementing speech recognition system. The current state of the art speech recognition technology is able to produce speaker independent recognizers which have extremely high recognition rates for small/medium vocabularies.

Although the average recognition rates are high, some speakers have recognition rates considerably worse than others. It is generally agreed that speaker dependent system will give the best performance in applications involving a specific speaker. This requires, however, that enough training data is available for training the system from scratch. An often used solution is to train speaker independent system using data from many speakers. But other experiments have shown that using such systems, in general, involves obtaining a lower performance than what is achievable with a speaker dependent system. This problem can be overcome, at least partially, by using speaker adaptation techniques, the aim of which is to take an initial model system which is already trained, and use a sample of a new speaker data to attempt to improve the modeling of the speaker with the current set of the model.

By collecting data from a speaker and training a model set on this speaker's data alone, the speaker's characteristics can be modeled more accurately. Such systems are commonly known as *speaker dependent* systems, and on a typical word recognition task, may have half the errors of a speaker independent system. The drawback of speaker dependent systems is that a large amount of data (typically hours) must be collected in order to obtain sufficient model accuracy. Rather than training speaker dependent models, *adaptation* techniques can be applied. In this case, by using only a small amount of data from a new speaker, a good speaker independent system model set can be adapted to better fit the characteristics of this new speaker.

Speaker adaptation techniques can be used in various different modes. If the true transcription of the adaptation data is known then it is termed *supervised adaptation*, whereas if the adaptation data is unlabelled then it is termed *unsupervised adaptation*. In the case where all the adaptation data is available in one block, e.g. from a speaker enrollment session, then this termed *static adaptation*. Alternatively adaptation can proceed incrementally as adaptation data becomes available, and this is termed *incremental adaptation*.

One of the researches on speaker adaptation techniques based on HMM is **Maximum Likelihood Linear Regression (MLLR)**. This method transforms the mean of continuous HMM. MLLR

will generate a global adaptation transform when a small amount of data is available. While more adaptation data becomes available, improved adaptation is possible by increasing the number of transformation using the regression class. The problem then occurred when the number of regression class increased while the adaptation data is static. The transformation matrices are difficult to estimate well enough when the amount of adaptation data is reduced too much due to a fine regression class division.

To overcome this problem the use of **Vector Field Smoothing (VFS)** incorporated with MLLR is a one technique. VFS is used to deal with the problem of retraining with insufficient training data. The transformation matrices produced by MLLR is then be used to calculate the transform vector of VFS continued by smoothing process.

2.1 Maximum Likelihood Linear Regression

MLLR uses a set of regression based transform to tune the HMM mean parameter to new speaker. The aim of MLL is to estimate an appropriate transformation for the mean vectors of each mixture component so that original system is tuned to the new speaker. For mixture component *s* with mean μ_s, the adapted mean estimate $\hat{\mu}_s$ is given by the following equation.

$$\hat{\mu}_s = W_s \bullet \xi_s$$

where W_s is an $n \times (n+1)$ transformation matrix and ξ_s is the extended mean vector,

$$\xi_s = [\omega, \mu_s, \cdots, \mu_{sn}]'$$

where the value of ω indicated whether an offset term is to be included: $\omega = 1$ for an offset, $\omega = 0$ for no offset. The transformation matrix is determined with a re-estimation algorithm based upon the principle of maximum likelihood estimation. In this way, the re-estimated transformation matrix is the one that maximizes the probability of having generated the observed adaptation data using the model.

2.2 Vector Field Smoothing

The vector field smoothing technique assumes that the correspondence between feature vectors from different speaker is viewed as a smooth vector field. Based on this assumption, the correspondence obtained from adaptation data is considered to be an incomplete set of observation from the continuous vector filed, containing observation errors. To achieve both better correspondence and reduction errors, both interpolation and smoothing are introduce into adaptation process.

VFS has three steps, as follows:

- **Concatenation training:** In this step, the mean vector of the Gaussian distribution is trained by concatenation training.
- **Interpolation:** In this step, the untrained mean vector is transferred to the new speaker's voice space by using an interpolated transfer vector.
- **Smoothing of transfer vector:** In this step, each transfer vector is modified in accordance with the other transfer vector.

2.3 MLLR-VFS

The technique of MLLR-VFS can be separately performed in three steps. The first step is an extension of the MLLR to multiple regression matrixes. The second step is calculating the transfer vector of VFS using the regression matrix produced by MLLR. The third step is the smoothing of transfer vector as VFS usual manner.

- Extension to multiple regression class

If *R* states $\{s_1, s_2, \cdots, s_R\}$ are shared in a given regression class, then the regression matrix $\hat{W}_s$ can be written:

$$\sum_{t=1}^{T}\sum_{r=1}^{R}\gamma_{sr}(t)\sum_{s_r}^{-1} o_t \xi'_{sr} = \sum_{t=1}^{T}\sum_{r=1}^{R}\gamma_{sr}(t)\sum_{sr_r}^{-1}\hat{W}_s \xi_{sr}\xi'_{sr}$$

- Calculation of transfer vector

The transfer vector $\Delta\hat{\mu}_i$ is calculated from the difference between the mean vector of the initial continuous density HMM and the initial continuous density HMM multiplied by the regression matrix.

$$\Delta\hat{\mu}_i = \hat{\mu}_i - \hat{W}_s\hat{\mu}_i$$

- Smoothing of transfer vector

In this step, each transfer vector is modified in accordance with the other transfer vector as an usual VFS manner.

We conduct these steps to develop the Indonesia speech recognition system with favorable result. Using the speech data provided by ATR-Japan, we obtain a promising result with accuracy rate around 90%. The signal processing model takes 12 kHz sampled data and transform it into 39-dimensional MFCC vectors every 10 ms (see Table 1, A: speaker independent, B: speaker dependent, Data is number of words for adaptation). This experiment also used Left-to-Right HMM model with single Gaussian Mixture.

Table 1. Result of Indonesian ASR

A	Data	MAP	VFS	MLLR	MLLR -VFS	B
79.7	10	85.19	81.96	85.34	86.51	92.7
	20	86.50	84.28	87.91	89.22	
	40	87.75	86.50	89.80	89.34	
	80	90.23	90.26	90.57	91.39	
	100	90.11	90.76	90.29	91.97	

Based on this result, we are now in collaboration with Telkom RDC to develop speech data to enhance the accuracy. We will also improve the speed of the system.

3 Machine Translation for Indonesian Language

A large number of Indonesian people, statistically, have problem in using and comprehending any information presented in other cross-border languages. The language barrier problem is compounded by the problem of the explosion of digital information whose majority uses English language via either Internet or any digital printed form which may overwhelms potential users and pose a threat of inequality of access of information due to the language barrier (digital divide) especially for the common Indonesian people. This is one of the motivations for us to propose a collaborative project to develop speech to Asian speech translation system, between BPPT-Indonesia, ATR-Japan, ETRI-Korea, NECTEC-Thailand, CCNOIDA-India, NTU-Taiwan and CAS-China.

In line with the research objectives, our most recent experiment is focusing on developing Indonesian statistical language model - based on the integration of stochastic and symbolic approaches - to be used for analysis stage in the machine translation engine. For creating the stochastic language model, it is worthwhile to utilize annotated data when it is available and use supervised learning mechanism to estimate model's parameter. In this case, an annotated corpus is created for multiple genres of documents. Of course, the costs of annotation are prohibitively labor intensive, and the resulting corpora sometimes are susceptible to a particular genre.

Due to this limitation of annotated corpora, it is necessary that we use unsupervised and weakly supervised learning techniques, which do not require large, annotated data sets. Unsupervised learning utilizes raw, un-annotated corpora to discover underlying language structure such as lexical and contextual relationships. This gives rise to emergent patterns and principles found in symbolic systems. In this system, the language model is trained using weakly supervised learning on small annotated corpus to seed unsupervised learning using much larger, un-annotated corpora. Unsupervised and weakly supervised methods have been used successfully in several areas of NLP, including acquiring verb sub-categorization frames, part-of-speech tagging, word-sense disambiguation and prepositional phrase attachment.

The Internet has proven to be a huge stimulus for statistical MT, with hundreds of millions of pages of text being used as corpus resources. Over the last few years, there has been an increasing awareness of the importance of corpus resources in MT research. As researchers begin to consider the implications of developing their systems beyond the level of proof-of-concept research prototypes with very restricted coverage, considerable attention is being paid to the role that existing bilingual and monolingual corpus and lexical resources can play. Such collections are a rich repository of information about actual language usage.

In developing monolingual corpus, we checked existing Indonesian news articles available on web (Purwarianti, 2007). We found that there are three candidates for the article collection. But in the article downloading, we were only able to retrieve one article collection, sourced from Tempointerakif. We downloaded about 56,471 articles which are noisy with many incorrect characters and some of them are English. We cleaned the articles semi-automatically by deleting articles with certain words as sub title. We joined our downloaded articles with the available Kompas corpus (Tala, 2003) at http://ilps.science.uva.nl/Resources/BI/ and resulted 71,109 articles.

In Indonesia, many research groups have been developing a large-scale annotated corpus to further the NLP and Speech research in trainable system. It should be clear that in statistical approach, there is no role whatsoever for the explicit encoding of linguistic information, and thus the knowledge acquisition problem is solved. On the other hand, the general applicability of the method might be doubted; it is heavily dependent on the availability of good quality of data in very large proportions, something that is currently lacking for Indonesian languages.

In order to experiment the feasibility of statistical MT for Indonesian, we build a prototype Indonesian-English MT. For that purpose, we need parallel corpus of Indonesian-English sentences, and there are none publicly available. Therefore, we have develop a collection of training and test sentences collected from a number of information sources mainly from Indonesia national news agency ANTARA, totaling 250.000 parallel sentences. We then use SRILM to build the n-gram language model and translation model, subsequently use PHARAOH (Koehn 2006) as a beam search decoder.

4 Discussion and Future Work

We are working forward to improve the quality of speech recognition and MT. Our collaboration with Telkom RDC and ATR-Japan will provide us with new speakers' data (40 speakers, 1000 words) which is expected to improve the accuracy of ASR to a better 90% level.

In other speech processing work, University of Indonesia (UI) and Bandung Institute of Technology (ITB) are also developing ASR and speech synthesis (SS) which will be integrated in the final speech translation system.

We are also building a new corpus in broadcasting news, to train the translation system, so as to enable automatic "tagline" in bilingual TV program. The experts in translation have two differing approaches toward the translation concept: universalism and monadic. We understood there is a possibility of "un-translation" which is "translation fails – or un-translability occurs when it is impossible to build functionally relevant features of the situation into contextual meaning of target language (TL) text. Broadly speaking, the cases where this happens fail into two categories. Those where the difficulty is linguistic, and those where it is cultural.

We examine further the translability concept by taking into account that most Asian language share very similar "culture" but different in language structure. We can not enforce the system and structure to target language without "knowing" the language itself. In this case, a rule-based system should be used as a preprocessing to enable the structure of source language to approximate the structure of target language. For example, in translating Indonesian-English, we need a rule-based system to transform the DM-MD rule. This rule approximates the order of noun and adjective phrase of Indonesian according to English noun or adjective phrase. For example:

```
        MD                          DM
sebuah rumah besar -> a big house
  (a)    (house) (big)
gunung  biru itu -> the blue mountain
(mountain) (blue) (the)
```

In our future work, by implementing several symbolic modules as pre-processor, it is expected that statistical MT will perform better in translating by having a "similar" language structure.

5 Conclusion

An updated report on speech to speech translation system is given together with a brief overview of some of the issues and techniques in speech recognition and statistical machine translation (SMT), which are being actively researched today in Indonesia.

It is particularly important for Indonesian language to have research on speech-to-speech translation systems, which is an ideal solution to the field of language technology. Such work is clearly important but difficult because it certainly will bring up many interesting differences of emphasis, for example in speech-to-speech work, there is an emphasis on speed, and on dealing with sentence fragments, since we would like to be able to translate each utterance as it is spoken, without waiting for the end. This gives importance to bottom up methods of language analysis, and severe restrictions on the input in terms of the type of text.

References

Ayu Purwarianti, Masatoshi Tsuchiya and Seiichi Nakagawa. 2007. Developing a Question Answering System for Limited Resource Language - Indonesian QA, submitted to *Journal of Language Resources and Evaluation*.

C.H. Lee, J.L. Gauvain. 1993. "Speaker Adaptation Based on MAP Estimation of HMM Parameters", Proc.ICASSP, Minneapolis, USA, pp.II-558-561.

C.J. Leggetter, P.C. Woodland. 1995. "Maximum Likelihood Linear Regression for Speaker Adaptation of Continuous Density Hidden Markov Models", *Computer Speech and Language*, 9(2):171-185.

F.Z. Tala. 2003. A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia, M.Sc. Thesis, University of Amsterdam.

H. Riza. 1999. The Indonesia National Corpus and Information Extraction Project (INC-IX), Technical Report, BPP Teknologi, Jakarta, Indonesia.

H. Riza. 2001. BIAS-II: Bahasa Indonesia Analyser System Using Stochastic-Symbolic Techniques, International Conference on Multimedia Annotation (MMA), Tokyo, Japan.

Heidi Christensen. 1996. "Speaker Adaptation of Hidden Markov Models using Maximum Likelihood Linear Regression", Project Report, Aalborg University, Denmark.

J.C. Junqua, J.P Haton. 1996. Robustness in Automatic Speech Recognition – Fundamental and Application, Kluwer Academic Publiser, Netherland.

Kazumi Ohkura, Masahide Sugiyama, Shigeki Sagayama. 1992. "Speaker Adaptation Based on Transfer Vertor Field Smoothing with Continuous Mixture Density HMMS, Proc of ICSLP 92, pp. 369-372.

M.J.F. Gales. 1997. "Maximum Likelihood Linear Transformations for HMM-Based Speech Recognition", TR 291, Tech. Report, Cambridge University Engineering Department.

Oskar Riandi. 2001. "A Study on the Combination of Maximum Likelihood Linear Regression and Vector Field Smoothing for Speaker adaptation", M.Sc Thesis, Japan Advanced Institute of Science and Technology (JAIST), Japan.

S.Young, G. Evermann, M.J.F. Gales, T. Hain, Dan Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, P.C. Woodland. 2005. "The HTK Book (for HTK Version 3.3)", Revised for HTK Version 3.3 April 2005, Cambridge University Engineering Department

Philipp Koehn. 2006. Statistical Machine Translation: the Basic, the Novel and the Speculative, SMT Tutorial, University of Edinburgh.

Sakriani Sakti, Konstantin Markov, Satoshi Nakamura. 2005. "Rapid Development of initial Indonesian Phoneme-Based Speech Recognition Using The Cross-Language Approach", Proceeding of O-COCOSDA, Jakarta.