Towards Better Language Modeling for Thai LVCSR

Markpong Jongtaveesataporn[†], *Issara Thienlikit*[†],
Chai Wutiwiwatchai[‡], *Sadaoki Furui*[†]

[†] Department of Computer Science, Tokyo Institute of Technology, Japan
[‡]National Electronics and Computer Technology Center (NECTEC), Thailand
{marky,issara}@furui.cs.titech.ac.jp, chai@nectec.or.th, furui@cs.titech.ac.jp

10.21437/Interspeech.2007-447

Abstract

One of the difficulties of Thai language modeling is the process of text corpus preparation. Because there is no explicit word boundary marker in written Thai text, word segmentation must be performed prior to training a language model. This paper presents two approaches to language model construction for Thai LVCSR based on pseudo-morpheme merging. The first approach merges pseudo-morphemes using forward and reverse bi-grams. The second approach utilizes the C4.5 decision tree to merge pseudo-morphemes based on multiple features. The performance of ASR systems with language models built using these methods are better than systems which use only pseudo-morpheme or lexicon-based word segmentation. These approaches produce results comparable to that obtained by the system utilizing manual segmentation.

Index Terms: Thai LVCSR, Thai language model, Word segmentation, Pseudo-morpheme

1. Introduction

Large vocabulary continuous speech recognition (LVCSR) systems have been successfully developed for many major languages. High-accuracy acoustic models along with large-scale language models are key components in the LVCSR systems; however, research on Thai LVCSR is moving forward slowly, partly because of the lack of large text corpora for training language models[1]. Since there are no explicit word boundary markers in written Thai text, segmentation is necessary. Therefore, building a Thai text corpus is not a trivial task. In our previous research, we manually compiled a text corpus containing around 55,000 words by extracting sentences consisting of most frequent 5,000 words from another large text corpus having around 2,500,000 words. This task required extensive manual work and resources, and required over 6 months of manual segmentation and inspection to complete.

In order to eliminate the need for manual work, normally a word segmentation tool is employed to segment text. A word segmentation tool typically relies on a hand-coded lexicon plus several statistical techniques. One existing tool, named SWATH [2], is quite effective but still far from perfect. The quality of the segmentation results depends on the coverage of the lexicon with regard to the vocabulary in the text corpus. Theoretically, it can be improved easily by increasing the size of the tool's lexicon. However, it is difficult and time-consuming to build a large lexicon or a text corpus to train the tool. Moreover, new words and foreign words which are initially not included in the lexicon are created everyday. Even if numerous lexical entries are added to the lexicon, they cannot be compiled to cover all new words. Hence, there is no guarantee that the result will ever be 100% error free.

This paper introduces two approaches to building Thai language models. A pseudo-morpheme is defined here as a written form of a syllable-like unit. Both approaches are based on the concept of pseudo-morpheme merging. The first approach uses mutual information as a key to merge pseudo-morphemes together. The second approach merges pseudo-morphemes by utilizing the C4.5 decision tree trained with statistical and linguistic information of the pseudo-morphemes. The vocabulary lists are then enumerated and pronunciations of them are generated automatically by a grapheme-to-phoneme (G2P) program. Language models are built based on these segmented corpora and used to construct Thai LVCSR systems.

2. Thai Language Characteristics

In written Thai text, there is no explicit word boundary marker. Sometimes a space is used to separate phrases and sentences but it can also be inserted within a sentence for aesthetic reasons. Grammatically, Thai words are categorized into two major types: 1) simple words, each of which consists of one or more syllables. Each syllable may also have a meaning but it is not connected to the meaning of the whole word; 2) compound words, each of which consists of two or more simple words. The meaning of a compound word can be completely different or linked to the meaning of each simple word. With this criterion, the determination of word boundaries on real data is still ambiguous and may not be consistent for different people.

3. Recognition Units for Thai LVCSR

Due to the characteristics of the Thai language mentioned above, the process of preparing a text corpus for training a language model requires word segmentation. The lexical units in the dictionary used by the LVCSR system, known as recognition units, thus, depend on how the text corpus is segmented.

3.1. Pseudo-morpheme

The term "pseudo-morpheme" (PM) is defined here to represent a written form of a syllable-like unit in order to avoid confusion with the definition of a syllable in the sound system. Table 1 shows three examples of word and their corresponding pseudo-morphemes and syllables. The first example "หน้าต่าง" contains two pseudo-morphemes. Also this word is composed of two syllables and each syllable corresponds to each pseudo-morpheme. On the other hand, there are two pseudo-morphemes in the second example "วิทยา" while there are three syllables. The first pseudo-morpheme "วิท" represents two syllables (wit3 ta3) in this word. Similarly the third exam-

Table 1: Word, PM, and syllable

Word	PM (pronunciation)	Syllable (pronunciation)
หน้าต่าง	หน้า-ต่าง	หน้า-ต่าง
	(naa2 taang1)	(naa2 taang1)
วิทยา	วิท-ยา	วิด-ทะ-ยา
	(wit3 jaaz0)	(wit3 ta3 jaaz0)
พลศึกษา	พล-ศึก-ษา	พะ-ละ-สึก-สา
	(phon0 svk1 saaz4)	(pha3 la3 svk1 saaz4)

ple "พลศึกษา" holds three pseudo-morphemes and four syllables since the first pseudo-morpheme "พล" encloses two syllables (pha3 la3) in this case.

A significant problem for Thai word segmentation is caused by the lack of a clear definition for what a word is. The problem can be solved by pseudo-morpheme segmentation, based on the fact that pseudo-morphemes are more well-defined and can be more consistently analyzed than words [3]. Pseudo-morpheme segmentation is therefore more reliable. Once a number of pseudo-morpheme patterns are defined, every input string can be matched to these patterns. Trigram statistics of pseudo-morphemes then can be used to determine the best segmentation result.

Since pseudo-morpheme segmentation accuracy is very high, the language model trained by a pseudo-morpheme-segmented text corpus for a pseudo-morpheme based LVCSR system could be powerful. However, the pseudo-morpheme is not suitable to be used as a unit for LVCSR systems for several reasons. Firstly, pseudo-morpheme segmentation produces many short lexical units which generate high acoustic confusion. In addition, the span of an N-gram language model is significantly smaller since the unit is short. Therefore, the language model cannot perform efficiently comparing to word-based language model. Moreover, since some pseudo-morphemes may have variation in pronunciation depending on the context, the automatic G2P conversion process may not give correct pronunciation. As described above, the first pseudo-morpheme in the second example of Table 1 is pronounced in two syllables (wit3 ta3) while alone it represents only one syllable (wit3). In a similar manner, the first pseudo-morpheme in the third example is uttered in two syllables (pha3 la3) whereas it is pronounced as one syllable (phon0) when it is alone.

3.2. Word

Since words are longer than pseudo-morphemes in general, using words as recognition units would help mitigate the problem caused by the LVCSR system based on pseudo-morphemes regarding the acoustic model, the language model, and the G2P conversion. Here we classify words used as recognition units into two types.

3.2.1. Lexicon-based word

Lexicon-based words are grammatically defined and typically recognized by people. Existing Thai language processing systems mostly rely on lexicon-based word segmentation tools to segment a text corpus into lexicon-based words. The tool often yields good results when no unknown word is included in the input. However, since the tool cannot cover all possible words, segmentation errors always occur when input contains a word that is not registered in the tool. These errors affect the performance of the language model. The performance of the tool could be improved by increasing the size of its lexicon but this strategy should be avoided since it requires as much effort as that required by the text corpus preparation process for building a language model.

3.2.2. Data-driven word

Data-driven words are defined as words derived from pseudo-morpheme merging. Since no hand-coded lexicon is used in the process of pseudo-morpheme segmentation and merging, the process to generate data-driven words does not face the problem of unknown word segmentation errors that occur in lexicon-based word segmentation. Thus, this approach would yield a better language model for the LVCSR system.

4. Thai Language Modeling

This section introduces two approaches to build a Thai language model. We focus on data-driven word unit and both methods generate data-driven words. First, a text corpus is segmented into pseudo-morphemes. Next, the first approach merges frequent pseudo-morphemes into larger units based on mutual information. While pseudo-morphemes are merged depending on the decision tree in the second approach.

4.1. Mutual information based unit merging

There have been several attempts to automatically build compound words based on statistical-based methods[4][5]. First, the text corpus is segmented into pseudo-morphemes. We, then, use the geometric average of the direct and reverse bigrams:

$$M(w_i, w_{i+1}) = \sqrt{P(w_{i+1}|w_i)P(w_i|w_{i+1})} \qquad (1)$$

as a measure to judge whether any pair of two consecutive units w_i and w_{i+1} should be compounded. The measure M is between 0 and 1. A high value for M means that both the direct and the reverse bigram values are high for (w_i, w_{i+1}). This makes the pair a good candidate for compounding, since the co-occurrence probability of (w_i, w_{i+1}) is high. In our implementation, first, we train a bigram LM using the initial set of pseudo-morpheme units and compute M for all the bigrams. The bigrams that have an M value higher than a threshold are merged and added to the lexicon. Then we modify the training text using the new lexicon, train a new bigram LM and choose another subset of bigrams to merge as a new iteration. This process can be repeated a number of times to create longer units.

4.2. Decision tree based unit merging

Instead of using only statistical information to compound lexical units, we also propose a unit merging technique using multiple features, i.e. statistical and linguistic information. Here we choose the decision tree approach as it is quick and easy to implement. In this paper, we employ the C4.5 decision tree induction program [6] as the learning algorithm.

First, the text corpus is segmented into pseudo-morphemes. Then, we utilize the C4.5 decision tree to consider whether any pair of consecutive lexical units should be merged together. In order to train the decision tree, a text corpus is segmented into words manually, while it is also segmented into pseudo-morphemes. The attributes for each pair of consecutive pseudo-morphemes are computed. To tag the goal attribute, every pair of pseudo-morphemes is compared with the manually segmented text corpus to see whether they are parts of a word.

If they are parts of a word, it indicates that these two pseudo-morphemes should be merged together. With this approach, the goal attribute, i.e. "merge" and "notmerge" is determined. The following attributes are used for the learning algorithm.

1. Geometric average of direct and reverse bi-grams as expressed by Equation 1.
2. Whether or not the first consonants of two consecutive lexical units are the same. We include this attribute because, in Thai, many compound words happen to have the same consonant in each starting syllable.
3. Length (number of characters) of combined lexical units.
4. Length of the former lexical units.
5. Length of the latter lexical units.
6. Number of lexical units from the last proper noun's prefix.
7. Number of lexical units from the last preposition.
8. Number of lexical units from the last nominalizer.
9. Number of lexical units from the last conjunction.
10. Whether the lexical units that are being considered contain words identifying numerical values or not.

In order to calculate the attributes 6, 7, 8, 9, and 10, a set of 142 words composed of proper noun's prefixes, prepositions, nominalizers, conjunctions, and numerical values in Thai are provided in advance. With all these attributes and a goal attribute, a decision tree is constructed.

Another text corpus is used to test the decision tree. It is first segmented into pseudo-morphemes. Then, all attributes are calculated for each pair of consecutive pseudo-morphemes by the same process as in the training set. Finally, the decision tree is used to identify "merge" or "notmerge" pseudo-morphemes. All pairs of pseudo-morphemes with the "merge" attribute will then be compounded together and a newly modified text corpus can be achieved. Each pair of consecutive tokens in this derivative corpus is evaluated again through iterative processes.

5. Experiments

5.1. Experimental conditions

A gender-dependent acoustic model (AM) of 1000-tied-state triphones with 8 Gaussian mixtures is trained with a phonetically-balanced speech corpus with 18 male-speakers' voices using the HTK [7]. 25-dimensional feature vectors consist of 12 MFCCs, their delta, and a delta energy. Tone information is not used. A 17MB text corpus is gathered from a Thai newspaper website in several specific columns and used to train language models. Two experiments are performed to process this text corpus using the methods described in Section 4.1 and 4.2. The dictionary to be used for ASR system is enumerated from the list of words in the final processed text corpus. Then, bi-grams and tri-grams are trained from the segmented text corpus using the CMU SLM Toolkit [8]. Pronunciation for each entry in the dictionary is obtained from the automatic G2P conversion tool [9]. In order to avoid manual work in the experiments, words that fail in the G2P conversion process are excluded from the dictionary. The sentences containing excluded words are also removed from the corpus. JULIUS[10] version 3.4 is used as a speech decoder. Evaluation speech data consist of 1,000 sentences from 5 male speakers.

The text corpus used for training the C4.5 decision tree is also collected from a Thai newspaper website. Its size is 17MB containing around 346k sentences. All sentences in this text corpus are segmented into words manually.

5.2. Results

We have trained several language models based on two proposed methods in different conditions. For the first approach, we vary threshold values ($logM$) and the number of iterations. For the second approach, various language models are achieved from different iterations.

Since the text corpus is segmented in several ways, the perplexities estimated from the processed text cannot be compared. In order to compare language model perplexities of the different versions of the test set with different text lengths (the number of units in the text), we use a normalized perplexity:

$$PP^* = PP^{\frac{N_b}{N}} \quad (2)$$

where N_b is the length of the test set and N is the length of a reference version (baseline). OOV words in the text are ignored when computing the perplexity. The vocabulary lists in the dictionary for ASR built by different methods are not the same either. Therefore, we choose the character error rate (CER) as the measure for the comparison of ASR performance.

5.2.1. Mutual information based unit merging

The result of CER and PP varied by threshold values and iterations are shown in Table 2 and 3 respectively. The systems with threshold value of -1.2 and -1.6 show good results of CER; however, the generated vocabulary size of the system with threshold value of -1.6 exceeds 65k which is the limit of the tools we use after iteration 4. For threshold value of -1.2, the best CER of 9.8% is obtained from iteration 5 and 6 as shown in Figure 1.

Table 2: CER comparison

Iter. \ Thres.	-0.4	-0.8	-1.2	-1.6	-2.0
1	12.3	11.1	10.4	10.1	10.5
2	12.0	10.6	9.9	9.9	10.2
3	11.7	10.3	9.9	9.9	

Table 3: Normalized test-set perplexity comparison

Iter. \ Thres.	-0.4	-0.8	-1.2	-1.6	-2.0
1	181.4	171.0	162.0	153.7	162.2
2	172.4	163.7	159.1	146.6	168.1
3	170.2	154.5	154.6	151.6	

5.2.2. Decision tree based unit merging

The detailed results of CER and PP for each iteration are shown in Figure 1. The system with the best CER of 9.9% and the best PP is obtained from iteration 10. The graph also shows that PP is decreasing with iterations, but CER seems to be saturated after iteration 5.

6. Discussion

We also compare the performance of the systems with proposed methods to:

- The baseline system, in which text segmentation is done manually and the list of pronunciations is created by hand. We refer to this system as "Baseline".

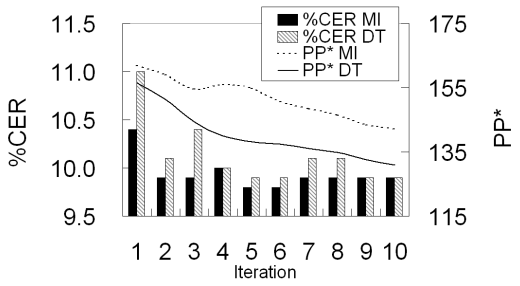

Figure 1: CER and PP* varied by iterations for mutual information based (MI) (threshold = -1.2) and decision tree based (DT) unit merging

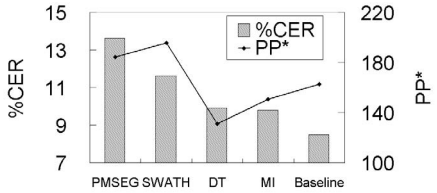

Figure 2: CER and PP* of various kind of systems and the best result from the systems with MI and DT unit merging

Method	Number of phones/word		
	Mean	SD	Max
MI	9.7	7.4	67
DT	7.6	4.0	44
Baseline	4.6	2.4	23

Table 4: Statistical information on number of phones per word from MI (threshold = -1.2, best condition at iter.6), DT unit merging (best condition at iter.10), and Baseline

- The system developed using lexical-based word segmentation, in which text segmentation is automatically done with SWATH tool and the list of pronunciations is obtained with the automatic G2P conversion tool. We refer to this system as "SWATH".
- The system based on pseudo-morpheme segmentation, in which text is segmented into pseudo-morphemes and their pronunciations are generated with the automatic G2P conversion tool. We refer to this system as "PMSEG".

We can see that by combining some pseudo-morpheme units, we can reduce the CER by around 3.8% compared to the PMSEG system. Perplexities are also successfully reduced by the proposed methods. Moreover, both methods also outperform the traditional approach to automatic Thai language model training, using SWATH. The high perplexity of the SWATH system comes from the problem of segmentation errors which create many junk words. Since the SWATH system is a lexical-based word recognition where segmentation requires a dictionary, unknown words are the main problem. On the other hand, segmentation errors in the proposed methods are less severe since we basically combine short units, resulting in lower perplexities. The OOV rates are also reduced from 0.7%(SWATH) to 0.05%(MI) and 0.07%(DT). The best CER is 1.3% higher than the manually-corrected baseline.

The number of phones per word (PPW) of generated vocabularies and those from manual work are shown in Figure 4. The average and maximum PPW from the MI method are around double and triple of those of the manual system respectively. The very long units might rarely occur in other domains. This is disadvantageous in the view of generality. Using multiple features in the unit merging process can reduce the length of generated units. In addition, as shown in Figure 1, the PP of unit merging with multiple features seems to be lower than using only mutual information by around 13%. However, there is no significant difference in the CERs of both methods. By using a more effective machine learning technique, the performance of unit merging with multiple features may improve. Also, the analysis on each feature must be studied more to enhance the system.

With regard to complexity, mutual information based unit merging approach is much easier to implement as there is no need for manual labor. All tasks can be done automatically. On the other hand, a manually segmented text corpus or manual tagging on the goal attributes is required for training a decision tree.

7. Conclusions

This paper has reported two methods for automatically building language models for Thai LVCSR. Both methods are based on the concept of pseudo-morpheme unit merging. The first approach merges pseudo-morphemes using mutual information. The second approach utilizes the C4.5 decision tree to merge pseudo-morphemes based on mutual information and other linguistic information. The CERs indicate the ASR performance are significantly improved from the system using merely pseudo-morpheme units. They also outperform systems developed using traditional approaches to Thai language modeling, such as that employed by SWATH. Finally, the performance of both methods are comparable to that obtained by manual segmentation.

8. Acknowledgements

The authors would like to thank Dr. Wirote Aroonmanakun for allowing us to use his pseudo-morpheme segmentation tool and NECTEC for providing useful Thai speech resources.

9. References

[1] C. Wutiwiwatchai and S. Furui, "Thai speech processing technology: A review," *Speech Communication*, vol. 49, no. 1, pp. 8 – 27, 2007.

[2] "SWATH," http://www.cs.cmu.edu/~paisarn/software.html.

[3] W. Aroonmanakun, "Collocation and Thai word segmentation," in *Proc. SNLP and Oriental COCOSDA Workshop*, 2002, pp. 68–75.

[4] G. Saon and M. Padmanabhan, "Data-driven approach to designing compound words for continuous speech recognition," in *IEEE Trans. on Speech and Audio Processing*, vol. 9, May 2001, pp. 327–332.

[5] M. Creutz, T. Hirsimäki, M. Kurimo, A. Puurula, J. Pylkkönen, V. Siivola, M. Varjokallio, E. Arisoy, M. Saraclar, and A. Stolcke, "Analysis of morph-based speech recognition and the modeling of out-of-vocabulary words across languages," in *Proc. NAACL-HLT 2007*, April 2007, pp. 380–387.

[6] J. R. Quinlan, *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, Inc., San Mateo, CA, 1993.

[7] http://htk.eng.cam.ac.uk/.

[8] R. Rosenfeld, "The CMU statistical language modeling toolkit, and its use in the 1994 ARPA CSR evaluation," in *Proc. ARPA Spoken Language Technology Workshop*, 1995.

[9] V. Sornlertlamvanich, P. Tarsaku, and R. Thongprasirt, "Thai grapheme-to-phoneme using probabilistic GLR parser," in *Proc. EUROSPEECH*, 2001, pp. 1057–1060.

[10] http://julius.sourceforge.jp/en/julius.html.