



COMPARING THE PERFORMANCE OF THAI MONOSYLLABIC AND WORD SEGMENTATION

Sukchatri Prasomsuk

School of Information and Communication Technology,
University of Phayao, Thailand
e-mail: skchatri@hotmail.com

Abstract

This research proposed the performance comparison of Thai monosyllabic segmentation approach, and Thai word segmentation approach. Matching techniques were used in two main types of experimental: non-dictionary and with dictionary. The first part of non-dictionary method was separated in two approaches: rule base and pattern matching for monosyllabic word. The second part was used a word list in dictionary with a simple algorithm. Both parts of them were employed two techniques of “Regular Expression” and “Hash map” techniques. All of experiments were compared together and then also compared with other techniques. Various corpuses were used in our experiment as testing data about 10Kb per file and a huge corpus file 44Mb approximately. The result of Thai monosyllabic, using non-dictionary came out correctly 83% of F-measure and 92% accuracy of monosyllabic word dictionary, and for Thai word segmentation with a word dictionary base was perfectly 90% at high speed process.

Keywords: Thai word segmentation, Language processing, Thai monosyllable word, Regular Expression, Hash table

Introduction

At the present, computers have been used for helping people in the human language translation, especially in the internet. About Thai language, each word does not have explicit word delimiters. Sentences are written entirely without separator. Most of applications such as automatic translation, automatic speech synthesis, and spelling checks are involved to word segmentation. The approaches of word splitting have been developed in the context of text editors, trying to read and recognize syllables, then write letter by letter, and line by line. There were limited to the main memory of a machine and the ability of a language compiler. Some applications of word segmentation in the past were not suitable to use at present. Moreover, some systems worked slowly with a huge document file, and difficult to implement. And also ambiguous words were not solved clearly. For example, «หลวงตามหาบัว : Luangta/Mahaboa », can see in two meanings, if we have word segmentation as the following:

- “หลวงตามหาบัว: Luangta/Mahaboa” is the name of monk.(proper noun)
- “หลวง/ตาม/หา/บัว:Luang/Tam/Ha/Bua” is meanted “Mr.Luang is looking for Ms.Bua”. Thus, as we have just seen, "หลวงตามหาบัว" may be a proper noun or a simple sentence or other meanings depending on the cut.

Related Studies and Theories

There are many related researches and a lot of models that they were developed for solving the Thai word segmentation. Some applications in previous works were used various techniques such as *Shortest-word pattern-matching*, *Longest-word pattern-matching*, *Word-usage-frequency*, *Backtracking*, *Maximal-matching*, *Ambiguity dictionary* [1][2][4][5][6][7][8][10].

In 2003 [3], Thai text analysis system was researched with INTEX[®] program for word segmentation by “Regular expressions” and finite state machine (FSM) to solve the problem of cutting Thai words by starting from characters and phrases with a dictionary. Furthermore, in 2011 [9], a researcher used a technique of the Thai-Writing Structure Matching, and creating Thai writing structure for word segmentation with the Royal Institute Dictionary.

Research Methodology

Most of previous methods of word segmentation have used several techniques and several types of programming. Some researchers use commercial products, others use a huge programming. Some methods use utilities to build a knowledge base system with external memory using dictionary. All of our literatures, we can customize and develop a system for our project using Java with appropriate methods to create software. Regarding the segmentation method, we have done not only need to adapt other previous programs, but we also created our own work. In addition, we used our experimental models that focus on parsing and learning real data to create software. Research method for the project emphasized on the development form a simple algorithm with a normal technique of “Regular Expression” and “Hash map” techniques. After that we concentrated on comparison of the existing systems.

In this experimental, we tried to work with two main types of experiments: non-dictionary for monosyllabic word segmentation and otherwise to work with a dictionary base for word segmentation.

Non-dictionary for monosyllabic word segmentation methods :

- Monosyllabic word segmentation approaches done through analysis each character with 85 rules. The process chart of system no.1 is shown on figure 3.1.
- Otherwise approach by creating word forms and using a technique of syllable structure matching with 200 patterns. The process chart of system no.2 is also shown in the same figure 3.1.

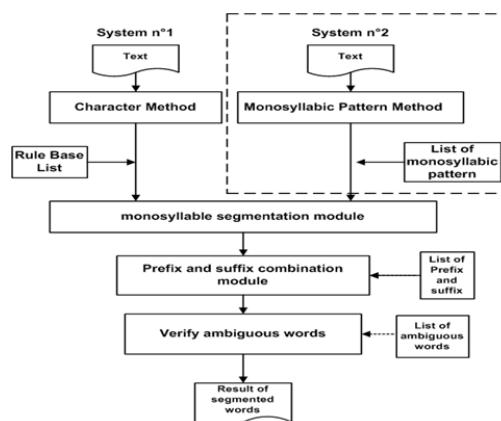


Figure 3.1 Process of Monosyllabic word segmentation

Dictionary base for word segmentation methods :

This part, we used the Royal Institute Dictionary (in 1999) 37,052 words and 35,520 words from LeXiTron v.2.6 dictionary excluding abbreviations and repetitive words, names of person and place, etc... Thus, total remain available words for this experiment were 52,605 words.

- Word segmentation approaches done with “Regular Expression” with a longest word technique and some ambiguous words problem solving. The process chart is shown in figure 3.2.
- Word segmentation approach works with a “Hash map” simple algorithm, including the same problems solving as above. The process chart is used in the same figure 3.2.

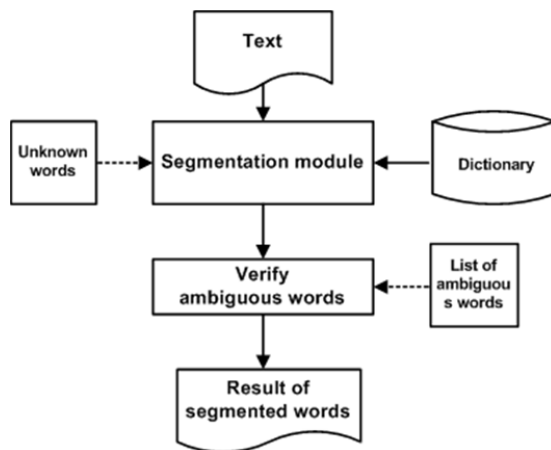


Figure 3.2 Process of word segmentation with dictionary base

Research Outcome

From the process in figure 3.1, we have got the successful result as the following.

An example below of system no.1.

Before Process : |ก|ร|ระ|บ|อ|ก| (Start from Rule 1)

After Process : |กระ|บอ|ก|

Another complex phrase example:

“มนุษย์เราไถนาโดยมุ่งหวังที่จะมีการเก็บเกี่ยวในฤดูเก็บเกี่ยวที่จะมาถึง”

[Sumran KUMYING, 1992, P.190]

Before Process :

|ม|น|ุ|ช|ย|ั|เ|ร|า|ไ|ถ|น|า|โ|ด|ย|ม|ุ|ง|ห|วัง|ที่|จะ|มี|การ|เก็บ|เกี่ยว|ใน|ฤดู|เก็บ|เกี่ยว|ที่|จะ|มา|ถึง|

After Process :

|ม|น|ุ|ช|ย|เรา|ไ|ถ|นา|โ|ด|ย|ม|ุ|ง|ห|วัง|ที่|จะ|เก็บ|เกี่ยว|ใน|ฤดู|เก็บ|เกี่ยว|ที่|จะ|มา|ถึง|

Note : For system no.2 retrieved the same result as above

List of example phrases through the process in figure 3.2 are illustrated as below.

Before ambiguous word problem solving	After ambiguous word problem solving
<u>โค</u> น <u>ม</u> น <u>อน</u> <u>บน</u> <u>หญ</u> ้า	โค <u>น</u> ม <u>น</u> อน <u>บน</u> <u>หญ</u> ้า
<u>จ</u> ัน <u>จ</u> ุง <u>โค</u> ล <u>ง</u> <u>เร</u> ื่อ <u>จ</u> น <u>เร</u> ื่อ <u>โค</u> ล <u>ง</u> <u>ไป</u> มา	<u>จ</u> ัน <u>จ</u> ุง <u>โค</u> ล <u>ง</u> <u>เร</u> ื่อ <u>จ</u> น <u>เร</u> ื่อ <u>โค</u> ล <u>ง</u> <u>ไป</u> มา
<u>ไป</u> หาม <u>เห</u> ล <u>ี</u>	ไป หาม เห <u>ล</u> ี
เร <u>า</u> ก <u>ิน</u> <u>จ</u> น <u>สม</u> อ <u>ย</u> าก <u>เล</u> ย	เร <u>า</u> ก <u>ิน</u> <u>จ</u> น <u>สม</u> อ <u>ย</u> าก <u>เล</u> ย
<u>ดู</u> น <u>ัน</u> <u>ล</u> ี <u>ตา</u> ก <u>อด</u> <u>ห</u> ล <u>าน</u> <u>ดู</u> น <u>่า</u> ร <u>ัก</u> <u>จ</u> ัง <u>เล</u> ย	<u>ดู</u> น <u>ัน</u> <u>ล</u> ี <u>ตา</u> ก <u>อด</u> <u>ห</u> ล <u>าน</u> <u>ดู</u> น <u>่า</u> ร <u>ัก</u> <u>จ</u> ัง <u>เล</u> ย
เร <u>า</u> มา <u>ร</u> อ <u>ก</u> ร <u>า</u> บ <u>ห</u> ล <u>วง</u> ต <u>า</u> ม <u>ห</u> า <u>บ</u> ั <u>ว</u>	เร <u>า</u> มา <u>ร</u> อ <u>ก</u> ร <u>า</u> บ <u>ห</u> ล <u>วง</u> ต <u>า</u> ม <u>ห</u> า <u>บ</u> ั <u>ว</u>
<u>จ</u> ัน <u>เข</u> ็น <u>แพ</u> ล <u>ง</u> <u>น</u> ้ำ <u>จ</u> น <u>ข</u> า <u>แพ</u> ล <u>ง</u>	<u>จ</u> ัน <u>เข</u> ็น <u>แพ</u> ล <u>ง</u> <u>น</u> ้ำ <u>จ</u> น <u>ข</u> า <u>แพ</u> ล <u>ง</u>
ค <u>ณ</u> ะ <u>ก</u> ร <u>ม</u> การ <u>ย</u> ก <u>ร</u> าง <u>ห</u> น <u>ึ่ง</u> ส <u>ื่อ</u> ว <u>าง</u> <u>ไ</u> ว <u>้</u> ห <u>น</u> ้า <u>ห</u> อง	ค <u>ณ</u> ะ <u>ก</u> ร <u>ม</u> การ <u>ย</u> ก <u>ร</u> าง <u>ห</u> น <u>ึ่ง</u> ส <u>ื่อ</u> ว <u>าง</u> <u>ไ</u> ว <u>้</u> ห <u>น</u> ้า <u>ห</u> อง
<u>น</u> ิ่ง <u>ตา</u> ก <u>ล</u> ม <u>ส</u> บ <u>าย</u> ดี	<u>น</u> ิ่ง <u>ตา</u> ก <u>ล</u> ม <u>ส</u> บ <u>าย</u> ดี

Note: underlined words are ambiguous words.

Our technique to treatment this problem is to eliminate some errors and give the correct direction of symbol “|”. The results are shown in the right of table.

The evaluation of our systems can be separated in two part of segmentation: segmentation of syllables and words or phrases. We were also to compare our results with competing methods. In this experiment, we used sample corpus data about 10 Kb belonging to 10 different areas (Agriculture and Environment, economy, Medicine and Health, Law, Political, Sport, Engineering and Technology, Computer and internet, Travel, Society and Culture). All of sample data are encoded in Unicode, UTF-8, and/or ANSI on a computer 1.4 GHz processor and 1.5 MB main memory which is used to test our treatments. In the table as below, System no.1-7 are our applications and techniques for this research.

Table 4.1 Summary of applications and each technique

Method	Name of systems	Words/forms	Techniques
By syllable	1.System n°1	85	Regx/Rules
	2.System n°2	200	Regx/SylbForme
	3.System n°3	10642	Regx/monosylb
	4.System n°4	10642	Algo+HashMap/monosylb
	5.N-gram CU (/s)	-	N-gram /s
By Word	6.Longest (LexTo)	52605	Long+tri/Dic
	7.Long/Max (Swath)	Lex+RIT	Long/Max+tri/Dic
	8.Bi-gram (Swath)	Lex+RIT	Bi-gram+tri
	9.N-gram CU (/w)	RIT 37052	N-gram /w
	10.System n°5		Combiner/monosylb
	11.System n°6	52605	Regx/Dic
	12.System n°7	52605	Algorithm+HashMap/Dic

Performance and correctness measure

The evaluation technique is currently the most uses commonly. It was proposed by the "Grammar Evaluation Interest Group" (Harrison in 1991) and is often called "PARSEVAL." The formula for the evaluation is:

$$\begin{aligned}
 \text{Precision (P)} &= \frac{\text{number of correct provided answers}}{\text{number of responses}} \\
 \text{Recall (R)} &= \frac{\text{number of correct provided answers}}{\text{number of expected responses}} \\
 \text{F-Mesure (F)} &= 2PR/(P+R)
 \end{aligned}$$

The final result of comparison performance is revealed in the table as below.

Table 4.2 the result of our experiment in average values from 10 corpus tests.

Method	Method	Number of provided answers	Correct	Error (bruit)	Recall (%)	Precision (%)	F-mesure (%)	Sec.
syllable	Result of answers		679.80					
Word	Expected responses		708.60					
By syllable	1.System n°1	742.20	594.70	85.1	87.48	80.13	83.64	0.48
	2.System n°2	745.80	590.30	89.5	83.31	79.15	81.17	0.27
	3.System n°3	701.20	635.60	44.2	89.70	90.64	90.17	0.50
	4.System n°4	700.20	635.50	44.3	89.68	90.76	90.22	0.34
	5.N-gram CU (/s)	681.30	671.50	8.3	94.76	98.56	96.63	33.05
By Word	6.Longest (LexTo)	751.80	642.20	66.4	90.63	85.42	87.95	0.16
	7.Long/Max. (Swath)	677.20	503.10	205.5	71.00	74.29	72.61	0.18
	8.Bi-gram (Swath)	677.90	503.20	205.4	71.01	74.23	72.59	0.17
	9.N-gram CU (/w)	670.90	500.90	207.7	70.69	74.66	72.62	34.33
	10.System n°5	715.30	309.60	399	43.69	43.28	43.49	0.75
	11.System n°6	753.50	631.20	40.3	89.08	83.77	86.34	1.30
	12.System n°7	753.30	660.80	47.8	93.25	87.72	90.40	0.48

Moreover, we conducted a test with a large file corpus 44 Mb (from NECTEC, BEST 2009), the result is presented as the following table.

Table 4.3 the result of a large corpus (44 Mb)

Method	Method	number of provided answers	Correct	Error (bruit)	Recall (%)	Precision (%)	F-mesure (%)	Sec.
Syllable	result of answers		1325803.52					
Word	Expected responses		1381971.71					
By syllable	1.System n°1	1447111.22	1158469.09	167334.424	87.38	80.05	83.56	523.86
	2.System n°2	1452962.08	1150667.95	175135.563	86.79	79.19	82.82	192.43
	3.System n°3	1367149.55	1238430.76	87372.7531	93.41	90.58	91.98	569.75
	4.System n°4	1365199.27	1238430.76	87372.7531	93.41	90.71	92.04	218.67
	5.N-gram CU (/s)	1328143.86	1308641.01	17162.5051	98.71	98.53	98.62	64456.91*
By word	6.Longest (LexTo)	1464663.78	1248182.19	133789.528	90.32	85.22	87.70	65.17
	7.Long/Max. (Swath)	1320342.72	980993.19	400978.527	70.99	74.30	72.60	281.04
	8.Bi-gram (Swath)	1320342.72	980993.19	400978.527	70.99	74.30	72.60	300.29
	9.N-gram CU (/w)	1306690.73	975142.33	406829.381	70.56	74.63	72.54	66953.27*
	10.System n°5	1394453.54	602637.96	779333.753	43.61	43.22	43.41	1014.29
	11.System n°6	1468564.35	1230629.62	151342.09	89.05	83.80	86.34	2235.12
	12.System n°7	1468564.35	1287187.88	94783.8348	93.14	87.65	90.31	167.07



Conclusion


The result of our research by “Hash map” technique was revealed an average successful at 90% with an accepted processing time. Otherwise, tries technique (LexTo) was shown nearly result at 87% accuracy with a highest speed. The Longest word/Bi-gram techniques (SWAT) got a successful result over 70% with a good speed, while the N-gram approach (ThaiCU) gave the best result over 90% correctly for monosyllabic word segmentation but the processing time slowly.

Acknowledgements

This project is supported a grant by Franco-Thai research Project in the domain of computational linguistics in 2006, and Naresuan University (Phayao), Thailand in 2009 with in the research laboratory of “Institute National Des Langues et Civilisations Orientales” (INALCO), Paris France.

References

1. Charoenpornasawat Paisarn, *SWATH: Thai Word Segmentation Program*. <http://www.cs.cmu.edu/~paisarn/software.htm>, Tokyo Institute of Technology, Japan. And NECTEC of Thailand, 2003
2. Kooptiwoot Chompunuch, *Segmentation of Ambiguous Thai Words by Inductive Logic programming*. [in Thai] Master Thesis of Science, Department of Computer Engineering, Chulalongkorn University, Bangkok. 1999
3. Kosawat Krit, *Méthodes de segmentation et d'analyse automatique de textes thai Automated methods of segmentation and analysis of Thai texts*, Doctor Thesis of « Université de Marne-La-Vallée », 8 September 2003
4. Promchan Pisit, Teng-Amnuay Yunyong 1998. *Performance Comparison of Thai Word Separation Algorithms*. Proceedings of the National Computer Science and Engineering Conference 1998 (NCSEC'98), 19th-21st October 1998. Bangkok.
5. R. Varakulsiripan, J Ngamvivit, S. Junvan, S.Jivatayakul and S. Tipjuksurat, *Thai Word Separation Using Longest Word Pattern Matching*. Papers on Natural Language Processing, Compiled by V. Sornlertlamvanich, 1995
6. R. Varakulsiripan, W. Suchaichit, S. Junvan and S. Tipjaksurat, *Word Usage Frequency Algorithm*. Papers on Natural Language Processing, Compiled by V. Sornlertlamvanich, 1995
7. M. Allen Weiss, *Data Structure and Algorithm Analysis in C*. The Benjamin/ Cummings Publishing Company, Inc., 1993
8. S. Meknawin, P. Charoenpornasawat, Boonserm Kijisirikul *Feature-based Thai Word Segmentation*. NLPRS' 97 Proceedings of the National Language Processing Pacific Rim Symposium, 1997
9. Vuttichai Vichianchai, *Thai-Word Segmentation through Thai Writing Structure Matching*, *2011 International Conference on Modeling, Simulation and Control IPCSIT vol.10 (2011) © (2011) IACSIT Press, Singapore*

- 
10. W.Kanlayanawat, S. Prasitjutrakul, *Automatic Indexing for Thai Text with Unknown Words using Trie Structure*. NLPRS' 97 Proceedings of the National Language Processing Pacific RimSymposium, 1997