Proceedings of the Third IASTED International Conference
ADVANCES IN COMPUTER SCIENCE AND TECHNOLOGY
April 2–4, 2007, Phuket, Thailand
ISBN CD: 978-0-88986-656-0

AUTOMATIC RULE-BASED EXPERT SYSTEM FOR ENGLISH TO THAI TRANSCRIPTION

Chakkrit Snae[1] and *Pupong Pongcharoen[2]
[1] Department of Computer Science and Information Technology
[2] Department of Industrial Engineering
Naresuan University
Phitsanulok
Thailand
*pupongp@yahoo.com

ABSTRACT
Transliteration or transcription of names is necessary to communicate between different language communities, e.g. English to Thai writing system. Since names tend to show a certain intrinsic grade of variation this is even more the case for the transliterated or transcribed forms. Correct transcription and transliteration of names is one of the major problems in inter-cultural communication. Available standard "manual" transcription systems are often simply not used or are used inconsistently. Many computer-assisted systems are based on orthographic forms or pronunciation, rule based, and statistics-based approaches.

In this paper we discuss the problems of Romanization, e.g. ambiguities of pronunciation as well as syllable and word segmentation. These problems can be considerable guidelines an implementation of backward transcription from English to Thai. To standardise this process the author proposes an automated English to Thai transcription system, called RESETT (Rule-based Expert System for English to Thai Transcription). This tool uses rule based Royal Thai General System of Transcription, syllable pronounciation and segmentation, and a hybrid name matching algorithm called LIG3 (Levenshtein, Index of similarity, and Guth). An advantage of the name matching process is an optimised transliteration of the rather complex Thai writing system. The LIG3 algorithm helps to produce highly accurate matches for transcribed forms.

KEY WORDS
Romanization, name matching, rule based system, automated transcription

1. Introduction

Romanization is the representation of a word or language with the Roman alphabet, or a system for doing so, where the original word or language uses a different writing system (or none). Methods of romanization include transliteration, representing written text, and transcription, representing the spoken word. Each romanization has its own set of rules for pronunciation of the romanized words. Transcription is the conversion into written, typewritten or printed form, of a spoken language source. In linguistics, transcription is the process of matching the sounds of human speech to special written symbols using a set of exact rules, so that these sounds can be reproduced later. Transcription as a mapping from sound to script must be distinguished from transliteration, which creates a mapping from one script to another that is designed to match the original script as directly as possible. Transcription is often confused with transliteration, due to a common journalistic practice of mixing elements of both in rendering foreign names. The resulting practical transcription is a hybrid called both transcription and transliteration by general public.

Name or word transcription is one of the major problems in inter-cultural communication, because available standard transcription systems [1] are not used or are used inconsistently. Although there are transcription systems used by libraries and other official agencies, transcription tends to be far less predictable and highly inconsistent, even with a single individual. For example, an individual whose Thai name is “PHITSUNEE” may romanise the name on one occasion as PITSUNI and on another as PHITSOONEE. Both name representations are “correct” and can be said to be accurate Romanization of the same name. Even in cultures in which transcription systems provide a reliable standard, personal interpretation, accommodation to the spelling of another culture or perceptual confusion can cause the spelling to deviate from the standard. Thus, for example, the Thai name GOFF will vary with KOFF, because G and K are Romanization alternatives from different transcriptions systems. An observed variant of GOUGH, however, are GOFF and GOFFE, representing the influence spelling of English surnames by Reaney and Wilson [2]. The Thai Romanization tool gives KOP; on the other hand KOP is a

correct transliteration of two different Thai names: กอฟ and กอล์ฟ [3].

Thai spelling with its many-to-many sound/letter correspondences contributes to the problem of Romanization of names. Dialectal differences, historical and phonetic spellings make the Thai names somewhat unpredictable. For the rather complicated Thai writing system there exists an official standard called Royal Thai General System of Transcription [4] which is used for rendering Thai names into the Roman alphabet. It uses only straight letters for vowels, diphthongs and aspirated consonants. This standard has some drawbacks. It does not indicate the length of vowels and which of the five different tones of Thai language is used for a given syllable. From Chulalongkorn University, Bangkok, there is also an automated tool available which transcripts Thai names or terms into Roman letters. It is called "Thai Romanization" [5].

However, the Thai Romanization is only for Thai to English and problems of Thai Romanization occur because the ambiguities of pronunciation and syllable segmentation. Some of the problems of Thai grapheme-to-phoneme conversion are shown in the following [3]:

- Some characters map to different phonemes even they are in the same position (initial or final consonant). For example, when we use the character "ท" as an initial consonant, it could map to either /d/ or /th/, as in "บุน-ดะ-ริก" (bun-**d**a-rik)-{white lotus} and "กุณ-**ฑ**ล" (kun-**th**on)-{earrings}.

- There could be linking sounds between syllables in some words derived from Pali and Sanskrit. For example, in a compound word like "พัชนี"-{beautiful woman} there is a linking syllable /cha/ between the two words, "พัช" (pat) and "นี" (nee). This word is pronounced as /pat+**cha**-nee/, rather than (pat-nee).

As a transcription system the Royal Institute's system is based on the pronunciation, orthographic forms or pronunciation [6], like some Thai-English dictionaries. There are many ways to romanise a Thai name. For example, a common name like "จักรกฤษณ์" can be romanised as chakkrit, chakkrid, jakkrit, or chakkid. Similarly, when transliterating English names into Thai names, it is usual to have different written Thai forms depending on positions of letters (e.g the letter "t" is transcribed to /ท/ when it is an initial consonant but it is transcribed /ด/ if it is a final consonant) and vowel rules [7]. For example, the name "chakkrit" can be found written as ชาคคริด, ชากกริต, จากกริช, จากกริด, จักกริด, จักกริท, จักคริท, จักคริด, or จักคิด. This backward problem, transcribing or transliterating English words or names into Thai, is a harder problem, because for a given Roman letter these usually are more than one Thai letter. For example, the letter "k" can be spelled ก, ข, ฆ, or ค. Furthermore a significant number of exceptions do exist which contribute to the problems [3].

To cope with some of the problems for Thai to English transcription, the standard Thai Romanisation has been introduced. However, people tend to romanise Thai words on their own, rather than adhering to the standard. Similarly, English to Thai transcription seems to be more complex and difficult and is not near perfect, in particular since we lack appropriate solutions to overcome the problems mentioned above.

Aroonmanakun [9] used syllable segmentation in the first process for segmenting an input string into syllables which were compared to syllable patterns that are exceptions. Most Thai words are monosyllabic and words with more than one syllable are mostly of Pali or Sanskrit origin. Then rules were used for matching pronunciations of syllables to syllable patterns. We agreed with Virongrong [10] that pronunciation disambiguation should be handled simultaneously with word segmentation but it should be aimed not only to the word level but also the syllable pronunciation and segmentation. Aroonmanakun and Rivepiboon [6] designed and implemented a model of automatic Romanization to handle ambiguities of pronunciation and syllable segmentation. They used a trigram model of syllables [6] to disambiguate syllable segmentations by generating all possible pronunciations of each syllable, then the right pronunciation is chosen based on the result of word segmentation and the statistical information of pronunciation.

The main purpose of this paper is an implementation of backward transliteration from English to Thai which works similar to automatic Romanization. However, we harnessed a hybrid name matching algorithm and the system uses the rule-based Royal Thai General System of Transcription to optimise English to Thai transliteration.

This paper firstly discusses the basic methods used to design an automated transcription system RESETT (Rule-based Expert System for English to Thai Transcription) and some methodologies and models involved. Then the testing model is explained and results from using the approach are reported. At the end, we conclude and discuss further improvements.

2. Methodology and Design

RESETT is an automated tool for English to Thai transcription which is designed and implemented using rule based Royal Thai General System of Transcription, and a hybrid name matching method [11] as a statistical model [12] for finding the most correct English transcription.

2.1 English to Thai Transliteration System

The system transcribes from English to Thai using the rule based royal Thai general system of transcription and syllable pronounciation model based on the principle of Thai pronounciation. After that a name matching algorithm, LIG3 (LIG is abbreviated from Levenshtein, I.S.G., and Guth), is used to find matches for the transliterated Thai words in the database, to calculate the most correct transliteration and to rank them by percentage.

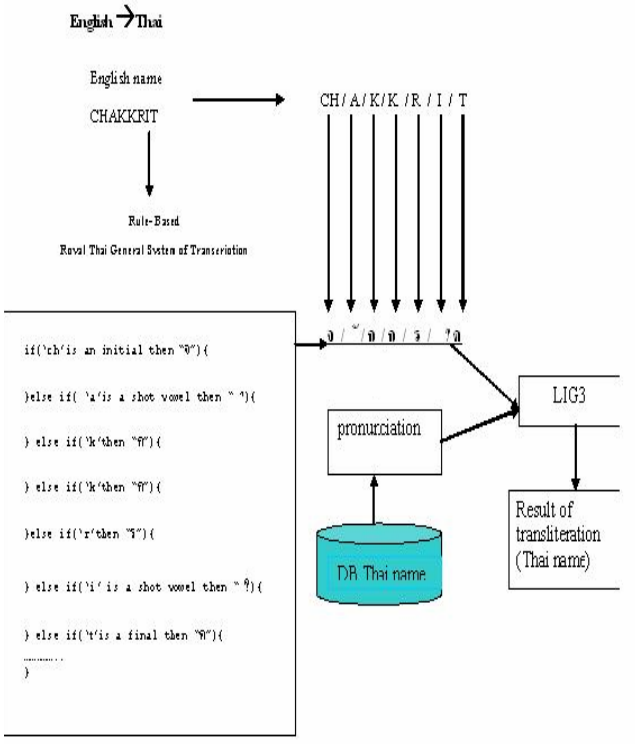

Figure 1 illustrates English -Thai-transliteration using rule based royal Thai general system of transcription and LIG3 name matching algorithm

2.2 Royal Thai General System of Transcription and Syllable Pronounciation

English to Thai transcription is a way to write English words in the Thai writing system. While the English alphabet represents far more sounds with its letters, namely 26 characters for 24 consonant and 20 vowel sounds, the Thai alphabet has far more letters than there are sounds: 44 characters for 21 consonant sounds and 19 characters (including 3 consonant characters) for 24 vowel sounds (including 6 diphthongs).

Thai syllable structures are different from that of English, phoneme sequences produced by the conversion rules have to be adjusted to comply with the Thai phonological system. Phoneme sequences that are not possible in Thai will be changed, deleted, or split into syllables with respect to the Thai writing rules [7].

Table 1 examples of classification of Thai letter

Rules of vowels	Classification of vowels	Romanization	Examples
Rule 1	เ–, โ, แ– , ไ, ใ	e, o, ae, ai	Ek
Rule 2	(รร), (–า), (–ี, –ื), (–ึ, –ื) (–ุ, –ู),	(an), (a), (i), (ue), (u)	Ka
Rule 3	–ะ, –ำ, เ–ะ, –็า	a, am, e, ua	Tua, Tam
Rule 4	–็, เ–็, ฤ, ฦ	a, e, i or e or oe, ue	Kek

As a consonant can not stand alone in Thai language as well as personal names, we consider rules for vowels only. The order is:

- Rule 1: Vowels can come first or can be followed by a first consonant, e.g. Ek
- Rule 2: Vowels can follow a first consonant without a final consonant, e.g. Ka
- Rule 3: Vowels that can not have final consonant, e.g. Tua
- Rule 4: Vowels that need final consonant, e.g. Kak

Consonants

Letter	Initial	Final
ก	k	k
ข	kh	k
ฃ	kh	k
ค	kh	k
ฅ	kh	k
ฆ	kh	k
ง	ng	ng
จ	ch	t
ฉ	ch	t
ช	ch	t
ซ	s	t
ฌ	ch	t
ญ	y	n
ฎ	d	t
ฏ	t	t
ฐ	th	t
ฑ	d	t
ฒ	th	t
ณ	n	n
ด	d	t
ต	t	t
ถ	th	t
ท	th	t
ธ	th	t
น	n	n
บ	b	p
ป	p	p
ผ	ph	p
ฝ	f	p
พ	ph	p
ฟ	f	p
ภ	ph	p
ม	m	m

Consonants

Letter	Initial	Final
ย	y	-
ร	r	n
ฤ	rue, ri, roe	-
ฤๅ	rue	-
ล	l	n
ฦ	lue	
ฦๅ	lue	-
ว	w	-
ศ	s	t
ษ	s	t
ส	s	t
ห	h	-
ฬ	l	n
อ	h	-

Vowels

Letter	Romanisation
–ะ, –ั, รร (with final), –า	a
รร (without final)	an
–ำ	am
–ิ, –ี	i
–ึ, –ื	ue
–ุ, –ู	u
เ–ะ, เ–็, เ–	e
แ–ะ, แ–	ae
โ–ะ, –, โ–, เ–าะ, –็	o
เ–อะ, เ–ิ, เ–อ	oe
เ–ียะ, เ–ีย	ia
เ–ือะ, เ–ือ	uea
–ัวะ, –ว, –ว–	ua
ใ–, ไ–, –ัย, ไ–ย, –าย	ai
เ–า, –าว	ao
–ุย	ui
โ–ย, –อย	oi
เ–ย	oei
เ–ือย	ueai
–วย	uai
–ิว	io
เ–็ว, เ–ว	eo
แ–็ว, แ–ว	aeo
เ–ียว	iao

Figure 2 Thai transcriptions according to the royal Thai general system

The Royal Thai General System of Transcription **(**Figure 2**)** and the principles of Thai pronunciation are used to transcribe English words/names to Thai words/names directly. For consonants, the transcription is different depending on the location of the letter within the syllable. In the part of Figure 2, showing the vowels, a dash ("–") indicates the relative position of the initial consonant belonging to the vowel.

Syllable pronounciation model is used to convert a Thai transcribed name from English into a Thai syllable pronunciation based on the principle of Thai pronunciation. This is an easy way to transcribe and merge English characters to Thai syllables.

2.3 Hybrid name Matching Algorithms

Hybrid name matching algorithms are phonetic and spelling based approaches which use similarity measure as probability [7]. We classified such algorithms as *hybrid* and introduced three into the literature called LIG1, LIG2, and LIG3 [13]. The advantageous characteristics of these algorithms can be summarized as follows: (1) simple design, which can lead to accuracy improvements without decreasing the performance, (2) use of probabilistic similarity measures based on distance and weight, (3) increase correct positive and reduce negative matches to maximize the overall accuracy, (4) provide phonetic tuning to address multi-cultural names without depending on the language.

For the English-Thai pronunciation it can be difficult to find a correct solution. To solve this problem we use LIG3 to find the variation of name/word and similarity value of name/word.

The LIG 3 algorithm used in this paper is a hybrid name matching algorithm and has been introduced by Snae [13]:

$$\text{LIG 3} = \frac{2I}{2I + C} \qquad (1)$$

where I is the number of identical letters in the two names calculated using the Levenshtein method; and C is a Levenshtein cost calculation [14]. The Levenshtein cost is defined for strings of arbitrary length and counts differences between strings in terms of the number of character replacement, insertions, and deletions needed to convert one into the other, the minimum edit distance is then the similarity.

For example, for arbitrary characters a, b we may define
c(a, a) = 0; denotes a match,
c(a, b) = 1 for a ~= b; denotes replacement of a (in s) by b (in t), where a ~= b,
c(a, -) = c(-, b) = 1; denotes deletion of character a and (-, b) denotes insertion of character b.

Using the cost of Levenshtein distance for C in this example, we obtain the following cost:

string1: *CHAKKRIT* -
string2: *CHAK-RAVI*

If one reads the alignment column-wise, a protocol of edit operations that lead from s1 to s2 has to be followed.

Match (C,C), (H,H), (A,A), (K,K), and (R,R) =0
Insert (-, K) = 1
Replace (A, I) and (V,T) = 2
Delete (I,-) = 1

The alignment shows one Insert (K), two Replace (Replace I and T) and one Delete (Delete I), and the other edit operations are Matches (five matches: C,H,A,K, and R) and thus the C (cost of edit distance) is two and I is five. For example, the Thai name from the Royal Thai General System of Transcription is compared to the pronunciation name from database and the results are reported only the calculation of LIG3 is more than 0.5 (50%).

The LIG3 method uses formula (1) to compute and return LIG3 values for name1 and name2. The algorithm is structured as follows:

1. initialise same, string lengths and position for name1 and name2
2. compute the variable same and diff by indicating how many identical and different characters there are between nm1 and name2. It matches characters up outside of position
3. *if* characters the same, increment same
   else compare current name1 character against all characters in name2 starting with last matched position to end
   if match found increment same, increase diff by number of characters skipped and update last matched position
4. create a matrix lmx of the requisite height and width (name1 across the top and name2 down the side) and get minimum of C (cost of Levenshtein distance)
5. calculate and return LIG3 value (2*same/2*same+c) which is described in formula (2)
6. return true, indicating that name1 matches name2 if LIG2 > 0.5

2.4 Database

The Database comprises two tables: a table dictionary of Thai words (containing some 28,500 words) and a table database of Thai names (containing some 6,000 first names). These tables have three fields: Thai word/name, pronunciation and meaning. The system uses a database of Thai-English pronunciation and a name-matching algorithm to get the most likely pronunciation of Thai names from the database. We find words/names which

sound similar and let the user choose an appropriate name/word if necessary.

3. Testing and Results

In the syllable pronunciation and segmentation testing, we use a dictionary database (Section 2.4) which contains the names and correct pronunciation. Then we compare each correct pronunciation in the database to each syllable pronunciation generated by the syllable pronunciation and segmentation model (Figure 3). Finally we calculate the percentage of accuracy for our model that generates correct pronunciation and incorrect pronunciation using the following formula:

$$A = \frac{T}{N} \times 100 , \qquad (2)$$

where A is the accuracy, N is the total number of names in the database and T is the number of names that have correct pronunciation.

Inaccuracy of the syllable pronunciation and segmentation model can be computed as:

$$I = \frac{S}{N} \times 100 \qquad (3)$$

where I is the inaccuracy and S is the number of names that have incorrect pronunciation.

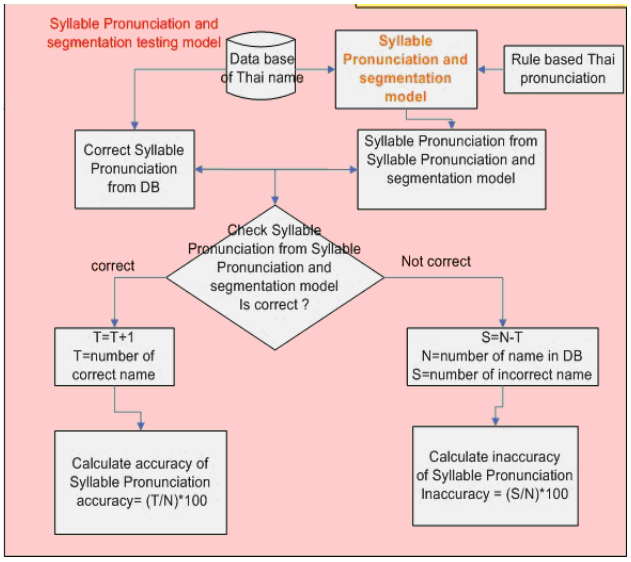

Figure.3 syllable pronunciation and segmentation model

The results of accuracy of syllable pronunciation and segmentation of the system using two databases (approximately 6,000 Thai names and 28,500 words from a dictionary of Thai words) can be shown in Table 2.

From Table 2 the accuracy of syllable pronunciation and segmentation is high (more than 90%). This means that our syllable pronounciation is sufficient for segmenting words correctly based on sound and the principle of Thai pronunciation. However, there is still a complex problem of returning incorrect pronounciation due to there is no rules exist in word/syllable pronounciation and segmentation that can be used to cope with the problem

Table 2 results of accuracy of syllable pronunciation

Database	Total size	correct	Accuracy (%)
Thai names	6000	5890	98.16 %
a dictionary of Thai words	28500	26505	93 %

The system is tested with English to Thai transcription model (Figure 4). First rule based Royal Thai General System of Transcription is used to transliterate English to Thai name. Then LIG 3 is used to compare the Thai name to each correct pronunciation in database, to calculate the similarity of name transliteration and to report the most correct transcription alternative. Then the user can select the best name from the resulting list of names with their meaning. A screenshot of the working system is presented in Figure 4 which shows the input and output model of RESETT.

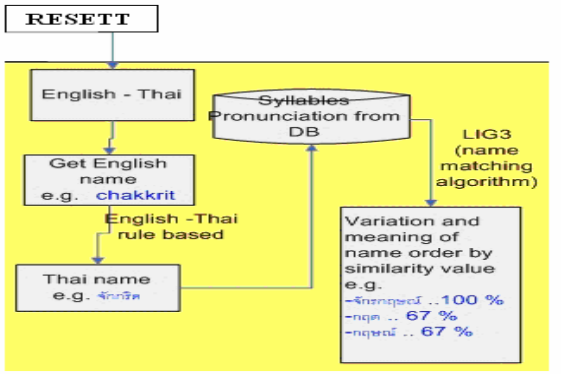

Figure.4 RESETT testing model

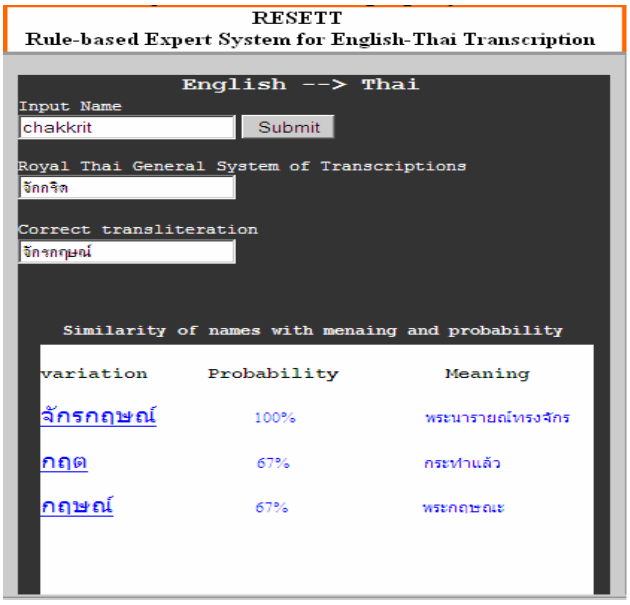

Figure 5 Interface of RESETT

For English to Thai the system uses the English name/word and transcribes it according to the rule-based Royal Thai General System of Transcription. Then the LIG3 algorithm is used to get the correct or most likely transcription. The results are sorted by similarity. From testing, results of name variants that matched using LIG3 are not really similar. This because LIG 3 returns more matches which unlikely to be similar to names that are translated in Thai.

4. Conclusion

We have proposed an automated transcription system for English to Thai writing system, called RESETT. This tool performs on the basis of rule based Royal Thai General System of Transcription, syllable pronounciation and segmentation and the LIG3 algorithm has the best performance in terms of producing most accurate true matches in transliteration.

An improvement is to implement a Thai Meta-Sound algorithm (combination of Soundex and Metaphone algorithms) and use it instead of LIG3 for finding the most likely correct matches (most similar matches) from English to Thai transliteration. Also Meta-Sound algorithm can be used for character/word clustering and grouping to classify diphthong characters or group of syllables or words that have the same leading sounds in each syllable to solve the problem of words/names that can be pronounced in different ways. For example, "ปรี" (pree) can be classified into a diphthong group of ปรีณัย (pree-nai) ปรีดิกร (pree-di-korn) ปรีดิท (pree-dit) ปรีดิทา (pree-di-ta) ปรีดิทัต (pree-di-tut) ปรีดิพัทธ์ (pree-di-put) ปรียปราณ (pree-ya-pran) ปรียวิศว์ (pree-ya-wis) ปรียส (pree-yos). Furthermore, we want to use the user input during his/her choice as a starting point for a machine learning solution.

Acknowledgement

I would like to thank Associated Professor Dr. Chayan Boonyarak and Dr. Punnee Sittidech for their moral support through my research at Faculty of Science, Naresuan University.

References

[1] J. Giarratano, *Expert systems*, (Gary Riley, PWS Publishing Company, 1998).

[2] P.H. Reaney & R. M.Wilson, *A dictionary of English surnames*, (Oxford: OUP, 1997).

[3] C. Snae, N. Singhadech, B. Emapana, & M. Brueckner, Interactive transliteration tools for explanation level language system (IT-TELLS). In *Proc. 21st International Technical Conf. on Circuits/Systems, Computers and Communications*, Chiang Mai, Thailand, 2006. I-245-I248.

[4] Royal Thai General System of Transcription <http://en.wikipedia.org/wiki/Royal_Thai_General_System_of_Transcription> (access Nov. 19, 2005).

[5] W. Aroonmanakun, A Chunk-based n-gram English to Thai transliteration. In: *Proc. 6th Symposium on Natural Language Processing*, Chiang Rai, Thailand, 2005, 37-42.

[6] W. Aroonmanakun & W. Rivepiboon, A Unified model of Thai word segmentation and romanization. In: *Proc. 18th Pacific Asia Conf. on Language, Information and Computation*, Tokyo, Japan, 2004, 205-214.

[7] C. Snae & M.Brueckner, Concept and rule based naming system, *The Information Universe: Journal of Issues in Informing Science and Information Technology* , *3*, 2006, 619-634.

[8] T. Charoenporn, A. Chotimongkol, & V. Sornlertlamvanich, Automatic romanization for Thai. In: *Proc. 2nd International Workshop on East-Asian Language Resources and Evaluation (ORIENTAL COCOSDA '99*), 1999, 137-140.

[9] W. Aroonmanakun, Collocation and Thai word segmentation. In: *Proc. 5th Symposium on Natural Language Processing & 5th Oriental COCOSDA Workshop*, Pathumthani: Sirindhorn International Institute of Technology, 2002, 68-75.

[10] T. Virongrong, P.T. Charoenpornsawa, & V. Sornlertlamvanich, A context-sensitive homograph disambiguation in Thai text-to-speech synthesis. In: *Proc. Human Language Technology Conf. (HLT-NAACL 2003)*, Edmonton, Canada, 2003.

[11] C. Snae & M. Brueckner, Hybrid name matching methods for rule based Thai naming system, *Naresuan University Science Journal ,2* (2), 2006, 139-150.

[12] M. Sipser, *Introduction to the theory of computation*, (PWS Publishing Company, 1997).

[13] C. Snae & B.M. Diaz, An interface for mining genealogical nominal data using the concept of linkage and a hybrid name matching algorithm, *Journal of 3D-Forum Society*, *16*(1), 2002, 142-147.

[14] V.I. Levenshtein, Binary codes capable of correcting deletions, insertions and reversals. *Doklady Akademii Nauk SSSR 163*, 1965, 845-848 (*trans. Soviet Physics Doklady 10*, 707-710).