J. Inf. Technol. Appl. Manag. 23(2): 11~28, June 2016
http://dx.doi.org/10.21219/jitam.2016.23.2.011

ISSN 1598-6284 (Print)
ISSN 2508-1209 (Online)

Ternary Decomposition and Dictionary Extension for Khmer Word Segmentation

Thaileang Sung* · Insoo Hwang**

Abstract

In this paper, we proposed a dictionary extension and a ternary decomposition technique to improve the effectiveness of Khmer word segmentation. Most word segmentation approaches depend on a dictionary. However, the dictionary being used is not fully reliable and cannot cover all the words of the Khmer language. This causes an issue of unknown words or out-of-vocabulary words. Our approach is to extend the original dictionary to be more reliable with new words. In addition, we use ternary decomposition for the segmentation process. In this research, we also introduced the invisible space of the Khmer Unicode (char\u200B) in order to segment our training corpus. With our segmentation algorithm, based on ternary decomposition and invisible space, we can extract new words from our training text and then input the new words into the dictionary. We used an extended wordlist and a segmentation algorithm regardless of the invisible space to test an unannotated text. Our results remarkably outperformed other approaches. We have achieved 88.8%, 91.8% and 90.6% rates of precision, recall and F-measurement.

Keywords : Word Segmentation, Decomposition, Natural Language Processing, Khmer

Received : 2015. 10. 27.　Revised : 2015. 12. 30.　2nd Revised : 2016. 05. 04.　Final Acceptance : 2016. 05. 04.

※ This article is a condensed form of the first author's master's thesis from Jeonju University in February 2015.

* Graduate Student of Information Systems Dept., Jeonju University, e-mail : thaileang@live.com

** Corresponding Author, Professor of Smart Media, Jeonju University, 303 Cheonjam-ro, Wansan-gu, Jeollabuk-do, 55069 Korea, Tel : +82-63-220-2757, e-mail : insoo@jj.ac.kr

1. Introduction

In the Natural Language Processing (NLP) fields, word segmentation is actively studied by many researchers. It has been used in several areas, including information retrieval, information extraction, part-of-speech tagging, machine translation, and other areas related to NLP.

Just like other natural languages, particularly Asian languages, Khmer NLP also has a number of issues related to word segmentation, including ambiguity, unknown words identification, and a lack of resources.

There has not been much research done on the topic of word segmentation for Khmer. Seng et al. [2009] proposed multiple text segmentation for statistical language modelling where the longest matching algorithm and the maximum matching algorithm have been investigated and N-grams are counted in order to train the language model. In [Seng et al., 2009], the techniques relied heavily on the quality of dictionaries. This fact raises the issue of out-of-vocabulary (OOV) words. As the number of OOV words increases, the more segmentation performances decrease.

The second research on Khmer word segmentation was made by a research group of Cambodia PAN Localization[1] [Chea et al., 2004]. They proposed two approaches : the word bigram model and the orthographic syllable bigram model. The word bigram model outperforms the orthographic syllable bigram model. However, neither model can handle the many types of unknown words due to the difficulty of identifying whether a string is an unknown word or a possible word.

To solve the segmentation errors caused by unknown words or OOV words, [Channa Van et al., 2010] proposed a rule based approach obtained by statistical analysis and the specific linguistic rules of Khmer. They used rule learning algorithm to detect OOV words. The algorithm is based on a SEQUITUR algorithm [CG Nevill-Manning, 1997]. In order to learn rules from a training corpus, they used the longest word matching algorithms to segment the text using a Khmer word list. After they implemented a rule learning algorithm and acquired the array of strings and rules of input text as the results, they used the rule-matching algorithm and linguistic rule-matching to obtain the final segmented words. As a result, when compared with PAN, the baseline slightly outperformed the PAN in terms of precision and F-measure, while a slightly better recall could be observed for PAN.

Channa Van et al. [2010] claimed that the most common approach in word segmentation is a lexicon-based one, where the longest matching algorithm or the maximum matching algorithm was used. The accuracy of this approach totally depends on a dictionary or word list which cannot cover all the words of a language. This circumstance results in an issue of unknown words or OOV words. Chea et al. [2004], Channa Van et al. [2010] and other researchers claimed that the same major problems occur in Khmer word segmentation : word segmentation ambiguity and

[1] http://www.panl10n.net/.

unknown word identification.

In this paper, we demonstrated the effectiveness of the Khmer dictionary on Khmer word segmentation. We proposed a new Khmer word segmentation algorithm using the ternary decomposition technique and Khmer dictionary extension. In section 2, we will overview Khmer language and Chuon Nath Dictionary. In section 3, we will review the details concerning the previous segmentation approaches mentioned earlier. In section 4, the proposed approach is presented, and including the details for the segmentation algorithm and dictionary extension. Section 5 describes the experimental setup and displays the results. Finally, we conclude our study and describe the future work related to this research in section 6.

2. Khmer Language Overview

2.1 Khmer Language

Khmer is the official language of Cambodia spoken by over 15 million people. Khmer has been considerably influenced by Sanskrit and Pali, especially in the royal and religious registers through the vehicles of Hinduism and Buddhism.[2] However, Khmer still possesses its own specific characteristics so far such as the word derivation rules, word reduplication technique, a specific written rule, etc. [Channa et al., 2010] [Chenda Nou et al., 2010]. Khmer differs from neighboring languages such as Thai, Burmese, Lao and Vietnamese in that it is not a tonal language. The writing system is from left to right. Khmer script is written continuously without spaces between words. In Khmer context, space can be seen when it is used to separate long phrases or as comma (,) punctuation in English. For instance; មានប៊ិចសៀវភៅ និងជ័រលុបក្នុងកាបូប។ (there are pens, books, and eraser in the bag). Khmer contains 33 consonants, 31 subscript consonants, 24 dependent vowels, 12 independent vowels, 2 consonant shifters, and a dozen diacritic signs and other symbols including number signs [Huffman, 2010].

ក្ក	ខ្ខ	គ្គ	ឃ្ឃ	ង្ង
ច្ច	ឆ្ឆ	ជ្ជ	ឈ្ឈ	ញ្ញ
ដ្ដ	ឋ្ឋ	ឌ្ឌ	ឍ្ឍ	ណ្ណ
ត្ត	ថ្ថ	ទ្ទ	ធ្ធ	ន្ន
ប្ប	ផ្ផ	ព្ព	ភ្ភ	ម្ម
យ្យ	រ្រ	ល្ល	វ្វ	ស្ស
ហ្ហ	ឡ្ឡ	អ្អ		

〈Figure 1〉 Khmer Consonants and Subscript Consonants

ក/kaa/ is the first consonant in Khmer alphabets which has ◌្ក/jerng kaa/ as a subscript consonant. Since ◌្ត/jerng taa/ serves as subscript for both ដ and ត, and ឡ has no subscript form, there are only 31 subscripts, compared with 33 consonants.

◌ា	◌ិ	◌ី	◌ឹ	◌ឺ	◌ុ
◌ូ	◌ួ	◌ើ	◌ឿ	◌ៀ	◌េ
◌ែ	◌ៃ	◌ោ	◌ៅ	◌ុំ	◌ំ
◌ា	◌ះ	◌ុះ	◌េះ	◌ោះ	

〈Figure 2〉 Khmer Vowels

[2] http://en.wikipedia.org/wiki/Khmer_language.

Some of these vowels in <Figure 2> are put in the top or bottom, to the left or to the right of a consonant, and some surround a consonant. For example, កា, កិ, កុ, កោ, កៀ...

The word "ខ្មែរ' means Khmer.

ខ្មែរ

ខ+◌្+ម+ែ◌+រ

1 2 3 4 5

<Figure 3> An Example of Khmer Spelling

According to Chuon Nath Dictionary spelling rule [Khin, 2007], we spell :

- Consonant ខ/khaa/
- ◌្(char\u17d2) to convert a consonant to a sub-consonant
- So the consonant ម/maa/becomes the sub-consonant ◌្ម (◌្ + ម = ◌្ម)/jerng maa/
- Spelled with vowel ែ◌ /ae/
- Final consonant រ/raa/ which is muted.

In Khmer vowels, there are two invisible vowels (or inherent vowels) "អ"/ɑː/ and "អ៊"/ɔː/ in all consonants. Therefore, Khmer consonants are divided into two groups : a group of consonants with inherent vowel, "អ" and another group of consonants with inherent vowel, "អ៊" can be shifted by the consonant shifters "◌៉" and "◌៊" [Huffman, 2010; Khmer Dictionary, 2005].

These independent vowels are capable to spell with/without a consonant. For example, ឥឡូវ, ឮ,រឭក... Khmer Grammar written by [Khin, 2007] and [Huffman, 2010] give the details of Khmer language.

Nowadays, computers all around the world understand Khmer language when typed using Unicode fonts. Unicode became a new international standard for Khmer.

ឥ	ឦ	ឱ	ឳ	ឯ	ឰ
ឧ	ឪ	ឫ	ឬ	ឭ	ឮ

<Figure 4> Khmer Independent Vowels

2.2 Chuon Nath Dictionary

Chuon Nath Dictionary was written by Samdech Preah Sangreach Chuon Nath[3] and published for the first time by the Buddhist Institute in Phnom Penh in 1938. Chuon Nath Dictionary is the official dictionary approved by the Cambodia government and the fifth edition of the dictionary was published by Institute BOUDDHIQUE, Phnom Penh, in 1967. After Khmer Rouge (1975~1979), the dictionary was almost forgotten. However, it had been discovered and has been revised and extended by the Buddhist Institute of Cambodia. Due to the lack of resources for the Khmer language, there is only the Chuon Nath Dictionary, which is recognized as the only official dictionary in Cambodia.

Chuon Nath Dictionary has 18,947 entries with 17,664 head words, which means each word may have multiple entries or definitions. Total words count in the dictionary is 1,312,732 words. This dictionary has been used in several researches so far, especially in the NLP field. [Seng et al., 2009] used a word list made from the Chuon Nath Dictionary to segment the text corpus using the longest matching algorithm or the maximum matching algorithm in his multiple segmentation approach. [Chea et al., 2004] also used the dictionary to match the string in KCE list

[3] http://en.wikipedia.org/wiki/Chuon_Nath.

in order to segment the text corpus using KCC bigram. [Channa Van et al., 2010] used longest word matching algorithm to segment the texts to get the output of an array of extracted terms in the first process of Rule Learning. Meaning [Channa Van et al., 2010] also used the dictionary. In the system architecture of [Chenda Nou et al., 2010] after the sentence is segmented into words, the tagger lookups in the wordlist (the dictionary) to retrieve the most frequent tag for each word. However, even though Chuon Nath Dictionary is the best choice for Khmer Lexicon Development, it does not have enough information to complete the Khmer Lexicon data dimensions.

Accordingly, the precision's rate of all the segmentation approaches mentioned above will be able to be increased, too, if the number of words in the dictionary is increased. Likewise, the unknown word identification problem is less worrying.

As in the description above, the needs of a reliable dictionary are very essential and necessary in most word segmentation and many other NLP research fields.

2.3 Problems in Khmer Word Segmentation

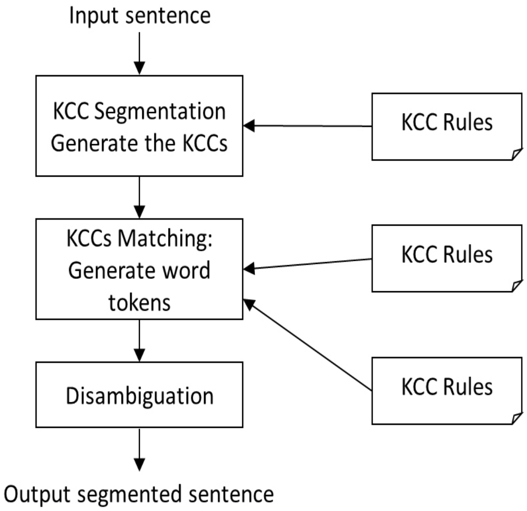
Text segmentation is a fundamental task in Natural Language Processing (NLP). Many NLP applications require the input text segmented into words before making further progress, because the word is considered the basic semantic unit in natural languages [Seng et al., 2010]. Identifying a word is trivial in English, because it uses word delimiters; so it is easier to segment English text corpus. However, Khmer does not have delimiters. To segment Khmer text into words is not a trivial task because a sequence of characters may be segmented in more than one way to produce a sequence of valid words. A sentence ពណ៌សម្ដេចថាខ្មៅ can be segmented into ពណ៌|ស|ម្ដេច|ថា|ខ្មៅ (color | white | why | say | black) or ពណ៌|សម្ដេច|ថា|ខ្មៅ (color | king | say | black) [Seng et al., 2003]. In here, we found out the ambiguity issue in Khmer segmentation. Another issue in Khmer segmentation is unknown words identification. It could be incorrectly segmented into shorter words or pieces of single syllable if the word does not exist in the dictionary. For example, a word ឆ|ហាក is a possible word, but it would be segmented into ឆ|ហាក after dictionary lookup. The difficulty is that whether a string is an unknown word or a possible word. There is some research on Khmer word segmentation that has been researched so far.

3. Research Reviews

3.1 KCC Bigram

3.1.1 General Overview on KCC Bigram

KCC is the abbreviation of Khmer Character Cluster. Chea et al. [2004] has proposed a word segmentation method that segment Khmer text into combinations of characters, called Khmer Character Cluster (KCC), and then merge those KCCs into possible word segmentations. Disambiguation module is used to select the best segmentation among those candidates. However, the disambiguation module was not clearly explained in Chea et al. [2004]; until the second research of Chea et al. [2004], bigram model was introduced to use in the disambiguation module to decide the most appropriate segmentation among the list of the candidates. Since then, it is called KCC Bigram.



〈Figure 5〉 Flow chart of the system

<Figure 5> shows the main process of the algorithm. The input sentence is segmented into combinations of characters (KCC). Then, KCC matching module reads each KCC one by one from left to right and match them. Next, it converts the KCCs into KCE (Khmer Common Expression) string. The KCE string is used to look up if it exists or not in the dictionary. Therefore, multiple possible segmentations of the input text are generated. Finally, the disambiguation module will select the best segmentation among those candidates. Here, bigram model is used.

3.1.2 Khmer Character Cluster (KCC)

KCC is another technical name of Orthographic syllable which is defined as a succession of characters with an inseparable unit. Since KCC is well defined, the boundary of the KCC might be the boundary of a Khmer word, too. Meaning, a Khmer word is the combination of one or more KCCs. For example,

- The word សាលាក្ដី is the combination of three KCCs : សា + លា + ក្ដី
- The word ចុកចាប់ is the combination of three KCCs : ចុ + ក + ចា + ប់
- The word ស្ដ្រី is one KCC word

The segmentation of text into KCC can be done by detecting the transition state for each character of the input string. If there is no possible transition of any character, it means the end of KCC is reached. The rule for forming KCC is given in term of type of characters.

To form the Khmer Character Clusters manually is really heavy work for a computer programmer. However, Lucene which is a free open source information retrieval software library works very well on Khmer cluster tokenization.

3.1.3 Khmer Common Expression (KCE)

KCE is an expression created to make Khmer strings with the same pronunciation similar. In KCCs matching module, KCE string is formed using the KCE rule. KCC, which is the combination of consonant, robat, subscripts, consonant shifter, vowels, various sign, zero width non-joiner, and zero width joiner, is used to be the fundamental component for building the KCE. Three types of rules have been categorized for building the KCE. Each type of rule handles different blocks of the KCC.

- Type 1 : focuses on the consonant and robat block of the KCC
- Type 2 : handles the subscript, consonant shifter block
- Type 3 : handles the vowels, various signs.

There are two main parts for each type of rule: matching rule and mapping rule [Chea et al., 2004]. The main idea of KCE, or Khmer Common Expression, is to encode the misspelled word and the dictionary word list into an expression that is based on how it is pronounced, which means the string with the same pronunciation has the same expression.

3.1.4 Bigram Model

Bigram model is used in the disambiguation module to decide the most appropriate segmentation among the list of the candidates. The main idea is to assume that the next word can be predicted given the previous word. Therefore, the probability function is as followed : $\mathrm{P}(w_1^n) = \Pi_{k=1}^{n} P(w_k|w_{k-1})$

For example, given the Bigram probabilities table shown in <Table 1>, the probability of the sequence “I have dinner” is :

$$\begin{aligned}\mathrm{P(I\ have\ dinner)} &= \mathrm{P(I} \mid <BOS>) \\ &\quad \times \mathrm{P(have} \mid \mathrm{I)} \\ &\quad \times \mathrm{P(dinner} \mid \mathrm{have)} \\ &\quad \times \mathrm{P(}<EOS> \mid \mathrm{dinner)} \\ &= 0.3 \times 0.2 \times 0.25 \times 0.13 \\ &= 0.00195\end{aligned}$$

<Table 1> Bigram Probability Table

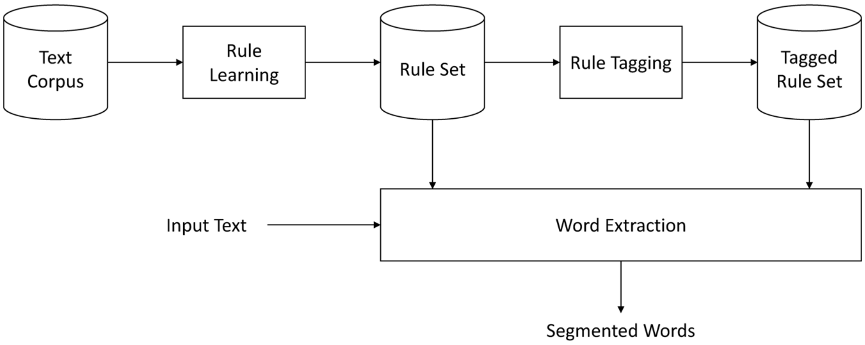
Bigram	Probability Value
<BOS> I	0.30
I have	0.20
have dinner	0.25
dinner <EOS>	0.13

However, there are a couple issues that need to be solved in the Bigram modelling; including overflow and data sparseness issues.

KCC Bigram is the Khmer word segmentation method developed by Chea et al. [2004]. It segments the input sentence into KCCs. Then, KCC matching module reads each of KCC one by one from left to right and converts KCCs into KCE string. The KCE string is used to look up whether it exists in the dictionary or not. By that, multiple possible segmentations of the input text are generated. To choose the best segmentation, the Bigram model is used in the disambiguation module. Therefore, trained text corpus is required in this process. The more text corpus is collected and trained, the higher accuracy of the segmentation is achieved.

3.2 Trainable Rule-based

Identifying OOV words, which are not detectable by using dictionary-based approach, such as compound words, proper names, acronyms and new words, are the most challenging tasks in Khmer word segmentation. By knowing that the more the OOV words increase, the more segmentation performance decreases, Channa et al. [2010] proposed the rule learning algorithm in order to discover the OOV words. Rules are obtained from the corpus by detecting the repeated character subsequence in text based on the SEQUITUR algorithm [Nevill-Manning, 1997]. A repeated character subsequence is a subsequence that occurs in a character sequence or a text more than once. In addition, the specific Khmer linguistic rules are also used to detect the possible OOV words from the text.



<Figure 6> The Trainable Rule-based Approach

<Figure 6> illustrates the trainable rule-based approach. In the rule learning step, the system extracts rules of the repeated character subsequence by using a rule-extracting algorithm. After completing the rule learning, the rule tagging step and the word extraction step are carried out. The rule tagging is done based on different kind of statistical measurements, such as entropy [Shannon, 2004], mutual information [Church et al., 1991], mutual dependency, log-frequency mutual dependency [Thanopoulos et al., 2002] and chi-square test. These statistical measurements are independently applied to weight the strength of each rule to be a word. A text is segmented into words based on the matching of the rules in the word extraction step. If a subsequence in the text matches a qualified rule, that subsequence is taken as a word and is segmented. Finally, the linguistic rules are applied to optimize the segmentation result. So there are two main tasks are carried out :

- Rule Learning : create a rule set based on the text corpus.
- Word Extraction : extract words based on the obtained rule set and the statistical measurements.

In addition, linguistic rule matching is done at the final stage of the word extraction in order to optimize the result of OOV recognition.

4. Proposed Approach

A dictionary has a unique and important role in getting high accuracy in word segmentation. In general, most of the word segmentation systems depend on a dictionary. The bigger the dictionary size is, the higher the precision of the word segmentation is achieved. However, the dictionary, which is usually used for Khmer word segmentation is not enough to cover the words in Khmer language. For this reason, increasing the number of words in the dictionary will help make the word segmentation more accurate. Therefore, the goal of our research is to extract new words from formal Khmer texts and insert those words into the Khmer dictionary [Khin, 2007] in order to achieve higher accuracy in Khmer word segmentation.

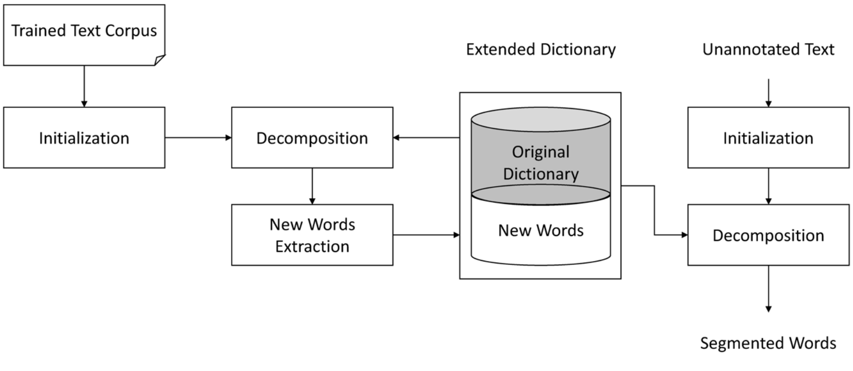
We propose a new Khmer word segmentation algorithm using the ternary decomposition technique and a Khmer dictionary extension. Here, we extract the new words from Khmer formal texts, which are considered as training texts. Since the texts use formal language we can assume that they are 100% correct-there are not any mistakes in grammar, spelling or new word in the texts. Accordingly, even if a word can't be looked up in the dictionary (original dictionary), the word can still be a possible new word; because the texts are the formal in nature and have no flaws. The texts are segmented by white spaces, invisible symbol, and the ternary decomposition technique. Then, all the words that can't be identified by the dictionary are set as new words, and inserted into the dictionary. Each new word will compare itself to other words in the extended dictionary (not the original dictionary). If the new word is the combination of two or more words in the extended dictionary, the process will remove the word out of the extended dictionary. This means the larger the training texts are the higher probability of the correct new words being found. Finally, we use the extended dictionary and our segmentation algorithm to segment unannotated text.

<Figure 7> illustrates our proposed approach. In the Initialization, the text corpus is split by symbols and white spaces and invisible spaces (char\u200B, invisible space is introduced in training section only); and the word segmentation process is started using the longest word matching algorithm based on the dictionary. Then, ternary decomposition is used. The detail of the decomposition is described in section 4.2. The new word extraction finds the words which are identified as unknown words, and sets those words as new words; and inserts the new words in to the dictionary. As the dictionary has been extended, we use the extended dictionary and our segmentation algorithm to test an unannotated text and then we acquire the segmented words as a result.

4.1 Initialization

4.1.1 Khmer Formal Texts

Before the process of the initialization of segmentation, Khmer formal texts have been collected and used as a training corpus. The texts are manually checked to assure that they have correct grammar, words, and spelling. The texts' resources are from history books, law text books, the Khmer Bible and other formal texts which are known to have reliable data. We collected



〈Figure 7〉 The Proposed Approach

only the texts typed with a Khmer Unicode keyboard authorized by NiDA that correctly typed according to Khmer writing system and NiDA's typing instruction. However, to find all those valid texts is complex, because not enough valid Khmer resources have been put up on websites or the internet in an appropriate way.

4.1.2 Preprocessing

We initialized our segmentation by splitting the texts into a list of many terms by symbols and white spaces. White spaces do not appear very often in Khmer context. We have learned that in Khmer, delimitation does not exist between words. Therefore, ambiguity occurs whenever we try to segment Khmer text. Fortunately, the latest version of NiDA Khmer Unicode has provided a better solution for this issue. People started using an invisible space (char\u200B). They use it to separate between words. Since some users of the Khmer Unicode become more aware of how to use the Khmer Unicode keyboard, they started to type Khmer with the invisible space to separate each word. For example :

ខ្ញុំនៅមានជីវិត។ → ខ្ញុំនៅមានជីវិត។

↖

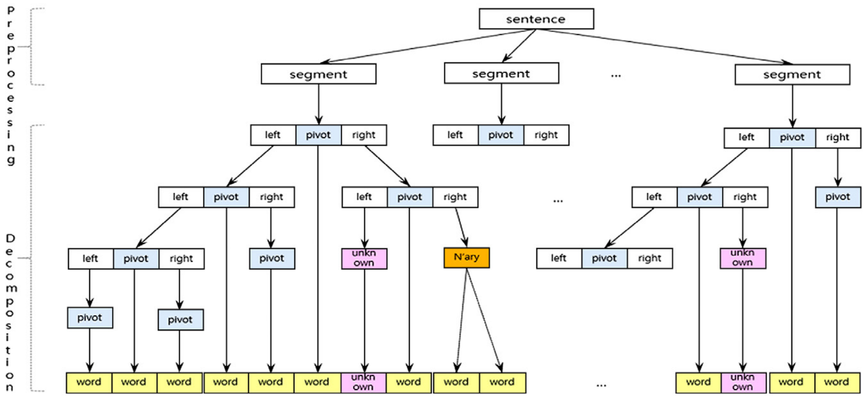
Invisible Space

The invisible space is very useful for Khmer segmentation and helps us to avoid a lot of hard work on NLP research especially Khmer ambiguity. So now, just like English, There is no need to worry about the delimitation issue anymore. However, there are many users that still haven't learned how to use the invisible space yet. Moreover, the Khmer Unicode keyboard provided by Microsoft in windows and Mac in OSX haven't included the invisible space. Accordingly, we use invisible space to help segment training corpus only in order to extract new words.

4.2 Decomposition

In the Initialization above, a number of possible words have been segmented already. Next is the continuation of segmenting the texts with decomposition. At this point, the longest word matching algorithm is used to find the longest word match based on the current dictionary [Channa et al., 2010]; and ternary decomposition technique is used.

Ternary means composition of three items. Ternary decomposition in word segmentation means splitting a string into three terms : left, pivot, and right. Here, the pivot is every word that exists in the dictionary and has a character length greater than *k*. At the end of the decomposition process, the pivot is set as *known words*. The process also checks the left and the right terms to see whether they exist in the dictionary or not. They are set as pivots if they can be looked up in the dictionary. If not, each of them is decomposed again into three terms in order to find the pivot. Otherwise, they are set as *unknown words*. The loop keeps working until the decomposition cannot look for a pivot word anymore (based on the dictionary). See <Figure 8> for the details of the Ternary Decomposition process.



〈Figure 8〉 Ternary Decomposition

We determine a *k* value as the number of characters. When we use invisible space in segmentation, we decompose only the words which have the *k* value greater than 9, otherwise, we define *k* is greater than 3. A Khmer word on average is composed of 3.2 syllables and 4.3 character clusters [Seng et al., 2010]. The majority of words in Chuon Nath Dictionary contain from two to four clusters [Puthick, 2005]. Therefore, if we define *k* as being too small, it is possible to break the word which has been already segmented.

The ternary decomposition process is shown in the diagram of <Figure 8>. Showing that each segment term in the process is split into three terms : left term, *pivot, and right term. The pivot* is the max-length word in each segment that has more than *k* characters. The *pivot* is set as the *known word,* since it is found as a word that exists in the dictionary. The process does thesame to the left and right term if they don't have a pivot yet. At the end of the process, the left and right term are set as *unknown words* if they do not consist of any words in the dictionary. Finally, we acquired the *segmented words* of *known* and *unknown words* as the result of the decomposition process.

4.3 New Word Extraction

New Word Extraction is the process of identifying new words from the segmented words resulting from the decomposition process. As mentioned in section 4.1, the text corpus must be formal and correct. Therefore, even though the words which cannot be found in the dictionary are possible new words. Those new words can be translated words, proper names, derived words, etc.

New Word Extraction takes on four roles. First, extracting the unknown words from the segmented words after the decomposing process, setting those words into new words. Second, checking whether there are duplicate words in the extended dictionary with each of the new words which have been found. Third, adding the new words to the extended dictionary if no duplication has been found. Finally, the process checks for each of the new words in the extended dictionary to find out whether there is a word, which is the combination of two or more words of the previous new word or current new word. If so, the process will remove the word and keep only the minimal word sets. Therefore, the more we use the dictionary extension process we can see more accurate new words being found.

4.4 Extended Dictionary

An extended dictionary is our proposed resolution. We chose the Chuon Nath Dictionary for the extension work. The Chuon Nath Dictionary has 18,947 entries with 17,664 head words, which means each word may have multiple entries or definitions. The total word count in the dictionary is 1,312,732 words.

1,312,732 words seem to be a lot of words, but it doesn't mean that they are helpful for users or researchers. As mentioned above, there are only 17,664 head words (stem words) in the dictionary; which means when users attempts to search for a word(s), they can only find the head head word. For example, ត្រីខ, ភាសាក្លិង្គ, គូព្រេង, ទ្បោមព័ទ្ធ, ឆបោក[4] etc. We cannot find these words in the Chuon Nath Dictionary. If we segment the text contained all these words based on Chuon Nath Dictionary, all these words will be separated into different words or wrong attempted words :

- ត្រីខ (name of a food) → ត្រី|ខ (fish | Consonant Kha)
- ភាសាក្លិង្គ (Indian language) → ភាសា|ក្លិង្គ (language | Indian People)
- គូព្រេង (predestined couple) → គូ|ព្រេង (couple | oil)
- ទ្បោមព័ទ្ធ (to surround) → ទ្បោម|ព័ទ្ធ (to surround | to bind)
- ឆបោក (to cheat) → ឆ|បោក (to lie | to throw violently)

Part-of-speech tagging, machine translation, and information retrieval are based on the accuracy of the segmentation. If the segmentation step failed to be precise in the first place, all the applications of NLP above also fail. As we have learned, the most common approach in word segmentation is lexicon-based [Channa et al., 2010]. Therefore, in this paper, the Chuon Nath Dictionary is proposed to be extended in order to get higher precision in Khmer word segmentation and make the Khmer dictionary more reliable. However, in the current approach, we are not working on how to collect and apply definition to each new word that we have found in the extended dictionary. Although, part-of-speech tagging processing is being studied, but due to the time limitation, we have decided to apply a part-of-speech tagging process in our future work.

Here, we have four functions in our algorithm code :

- Preprocessing()
- Decomposition()
- new_word_extraction()
- dictionary_extension()

The algorithm starts with the preprocessing method. It splits inputted strings into a set of segments by white spaces(and invisible spaces in the training section). Then, the decomposition method is initialized. The longest word match-

[4] http://dictionary.tovnah.com/help.

ing algorithm is used to find the max-length word based on dictionary and sets the word as the pivot. The pivot is the max-length word which is greater than k, which is the number of characters. In our algorithm, the value of kmust be defined. If the pivot is found and its length is greater than k,the segment is split into left segment, pivot, and right segment. If the left or right segment is not null, the left or right segment is split into the same three terms; but if the left or right segment exists in the dictionary, it is set as a pivot. Then the pivot is set as a known word at the end of the decomposition process. The segment (left or right segment) is set as an unknown word when the process cannot find anymore pivot.

Proposed Algorithm : Segmentation and Dictionary Extension

```
function preprocessing (String sentence) {
  Split the sentence by white spaces and symbols into a set of segments
  Then do
     decomposition (segment)
}
function decomposition (String segment) {
  Find max-length/longest matching word based in the dictionary and
          set the word as the pivot
  if the pivotis found and the length is greater than k
      Split the segment into left_segment, pivot, and right_segment
       Do decomposition (left_segment) if left_segmentis not null
      Set the pivotas a Known word
       Do decomposition (right_segment) if right_segmentis not null
  else
      The segment is composed with N words in Dictionary
      Set the segment into N known words
  Then Do
      new_word_extraction (segment)
}
function new_word_extraction (String segment) {
  Set the segment as a new word
  Do dictionary_extension (String segment)
}
function dictionary_extension (String segment){
  Insert the segment into Dictionary
  Refine dictionary to include minimal word sets only
}
```

As the result, we acquired the segmented words with both *known word* and *unknown word*. Next, the new word extraction method is initialized to look for *new words* from the decomposition process. *Unknown words* are identified and set as *new words*. We make sure to set them as *new words* due to the reliability of our Khmer text corpus. The method checks the extended dictionary to see if there are any duplicate words within the dictionary. If none is found, the method adds the *new words* to the extended dictionary. The method also checks for each of the *new words* within the extended dictionary to see whether the word exists in a combination of two or more words in previous *new words* or current *new words*. If so, the process will remove the word and keep only the minimal word sets in the extended dictionary.

For example, given previous/current *new words* : {abcde, ab, cd, e, abc}

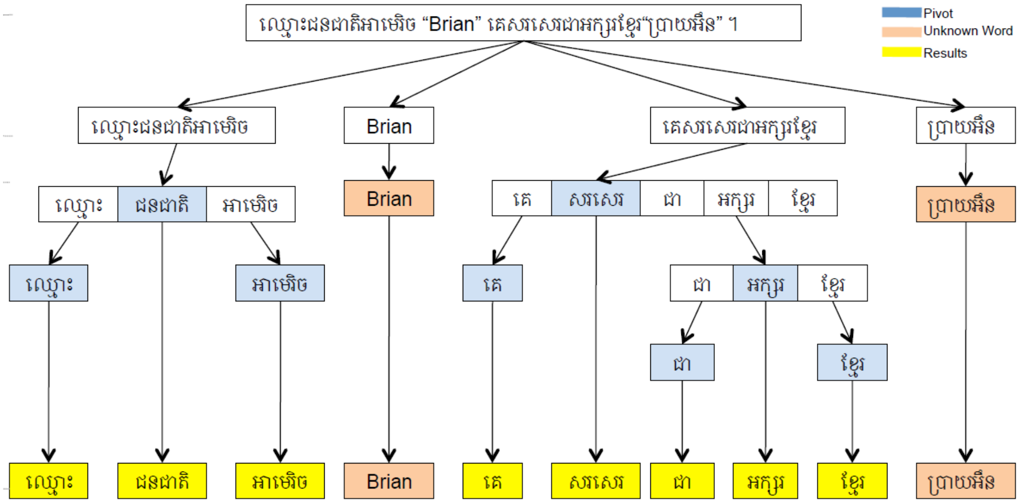
- “abcde” contains : “ab”, “cd” and “e”
- “abcde” is removed by the method
- “abc” contains only : “ab”
- So “abc” isn’t removed by the method

New words to be added into extended dictionary are abc, ab, cd, and e. In addition, to avoid unintended problems in future research, we have flagged all new words that have been found as temporary in our programming. Finally, we achieve our purpose in either dictionary extension or word segmentation. Furthermore, we use our extended dictionary for word segmentation tests. We use preprocessing and decomposition methods to test word segmentation on an unannotated text based on our extended dictionary. <Figure 9> shows an example of ternary decomposition.

5. Experiment

5.1 Experimental Setup

To make a formal text as a training corpus,



〈Figure 9〉 An Example of Ternary Decomposition we manually, collected several reliable text from Khmer history books, law text books, and even some of the Khmer bible context. We read through all the collected texts to make sure that they are formal and reliable resources for our corpus. However, we were unable to collect as many as we wanted. Our corpus has only 22,726 tokens. We used this corpus to train our system to make the dictionary extension. Furthermore, we also collected some unannotated texts to test our segmentation algorithm based on the extended dictionary. To test the performance, we manually segmented the unannotated texts. We also segmented the unannotated text with either invisible space or without invisible space.

Usually, three measures are used to evaluate the segmentation's accuracy : precision rate, recall rate and F-measure. A perfect method will have a recall and precision rate of 100% [Chea et al., 2004].

- Precision rate (P) = C/M
- Recall rate (R) = C/N
- F-Measure = (P×R×2)/(P+R)

Where, N : Number of words occurring in the manual segmentation, E : Number of words incorrectly identified by the automatic method, C = M-E : Number of words correctly identified by the automatic method, M : Number of words identified by automatic segmentation, and F-Measure : The harmonic mean of precision and recall.

Here, we manually segment the test text and we obtained N = 6,832 tokens. Using our process, we acquired M, E, and C as <Table 2> below :

〈Table 2〉 Number of Tokens of M, E and C

	M	E	C
Invisible Space	6,918	177	6,741
Without Invisible Space	7,068	791	6,277

5.2 Experimental Results

Using the Khmer formal corpus of 22,726 tokens, we achieved 1.39% of new words which do not exist in the dictionary. We can assume that the bigger the size of the corpus, the more new wordsare possible to find. Using the extended wordlist resulting from our extension process, we segmented a test text which has 6832 tokens; and our algorithm produced higher accuracy rates compared to PAN and Baseline (the trainable rule based).

<Table 3> shows the results of precision, recall and F-Measure of Khmer word segmentation. Our approach outperforms PAN and Baseline's results remarkably.

〈Table 3〉 Experimental Results

		Precision	Recall	F-measure
PAN		0.718	0.788	0.751
Baseline		0.777	0.755	0.765
Proposed Approach	Invisible Space	0.974	0.986	0.980
	Without Invisible Space	0.888	0.918	0.906

5.3 Discussion

According to the result of our experiment, the segmentation achieved an unexpectedly higher performance and had a greater performance compared to other previous research such as the PAN and Rule based approaches. However, what makes the segmentation achieve this high rate of accuracy is not based on the extended dictionary alone, but because of decomposition of splitting symbols, white spaces, and the participation of the invisible spaces hidden in the Khmer texts which was used in the training section and the ternary decomposition.

To extend the dictionary, we need a larger corpus in order to generate more new *words* and the texts corpus that are found must be qualified as formal texts. Likewise, the dictionary extension can be applied to not only be used with our word segmentation algorithm, but other segmentation approaches as well. Since the other approaches use wordlist based on the Chuon Nath Dictionary, they can also use our dictionary extension based method as well to gain the higher performance.

6. Conclusion

In this paper, we demonstrated on the effectiveness of Khmer word segmentation based on dictionary extension. We proposed a new Khmer word segmentation algorithm using the ternary decomposition technique and Khmer dictionary extension. In our work, we required the formal texts which do not have any word, grammar or spelling mistakes. In order that all the unknown words that can't be looked up in the dictionary can be extracted and set as new words and inserted into the extended dictionary.

Also, we have introduced the invisible space in the latest Khmer Unicode Keyboard which is a large contribution for Khmer word segmentation. We have used invisible space to help segment the training corpus in our training section. Khmer word segmentation can be more accurate as the extended dictionary is increased. The experimental results show that our proposed approach can achieve significantly better accuracy rates in Khmer word segmentation. Having a dictionary extended is not only useful for segmentation work, but also other Khmer NLP research fields such as information retrieval, information extraction, translate machine, and etc.

References

[1] Channa, V. and Kameyama, W., *Khmer Word Segmentation and Out -of-Vocabulary Words Detection Using Collection Measurement of Repeated Characters Subsequences*, 2010.

[2] Chea, S., Top, R., and Ros, P., *Detection and Correction of Homophonous Error Word for Khmer Language*, 2004.

[3] Chea, S., Top, R., and Ros, P., *Word Bigram Vs Orthographic Syllable Bigram in Khmer Word Segmentation*, 2004.

[4] Church, K. W., Robert, L., and Mark, L. Y., *A Status Report on ACL/DCL*, 1991, pp. 84-91.

[5] Huffman, F. E., *Cambodian System of Writing and beginning reader with Drills and*

Glossary, Yale University Press, 1970.

[6] Khin, S., "Khmer Grammar", *Royal Academy of Cambodia*, first Edition, 2007.

[7] Khmer Dictionary, *Royal Academy of Cambodia*, 2005.

[8] Mohri, M. F., Pereira, C. N., and Riley, M., "A rational design for a weighted finite-state transducer library", in *Lecture Notes in Computer Science,* Springer, 1998, pp. 144-158.

[9] Nevill-Manning, C. G., "Identifying Hierarchical Structure in Sequences A linear-time algorithm", *Journal of Artificial Intelligence Research*, Vol. 7, No. 1, 1997, pp. 67-82.

[10] Nou, C. and Kameyama, W., *Hybrid Approach for Khmer Unknown Word POS Guessing*, 2007.

[11] Puthick, H., Development of a Khmer Spell Checker Based on a Hidden Markov Model, A subthesis submitted in partial fulfillment of the degree of Master of Information Technology (eScience) at The Department of Computer Science Australian National University November, 2005.

[12] Seng, S., Abate, S. T., and Besacier, L., "Boosting N-gram Coverage for Unsegmented Languages Using Multiple Text Segmentation Approach", *Proceedings of the 1st Workshop on South and Southeast Asian Natural Language Processing* (WSSANLP), 2010, pp. 1-7.

[13] Seng, S., Besacier, L., Bigi, B., and Castelli, E., Multiple Text Segmentation for Statistical Language Modelling, 1LIG Laboratory, CNRS/UMR-5217, Grenoble, France 2MICA Center, HUT-CNRS/UMI-2954-Grenoble INP, Hanoi, Vietnam, 2009.

[14] Seng, S., Sam, S., Le, V.-B., Bigi, B., and Besacier, L., "Which Units for acoustic and language modelling for Khmer automatic speech recognition?", 38041 Grenoble Cedex 9, FRANCE, 2010.

[15] Shannon, E., "A Mathematical Theory of Communication", *Bell System Technical Journal*, Vol. 27, 1948, pp. 379-423.

[16] Thanopoulos, A., Fakotakis, N., and Kokkinakis, G., *Comparative Evaluation of Collocation Extraction Metrics*, 2002.

[17] Van, C. and W. Kameyama, "Query Expansion for Khmer Information Retrieval", *Proceedings of the 8th Workshop on Asian Language Resources*, Beijing, 2010, pp. 80-87.

Author Profile

Thaileang Sung

Thaileang Sung is currently a graduate student of Culture Technology. He received B.E in Computer Science and M.S. in Information Systems at Jeonju University. His research interests include Natural Language Processing, Big Data, and Data Mining.

Insoo Hwang

Insoo Hwang is a professor of smart media at the Jeonju University. He received his B.A, M.S., and Ph.D. degree in Management Information Systems from Korea University. His research interests include Data Mining, Big Data, and Natural Language Processing.