Automatic Mining of Verb Affixes for Myanmar Language with Unsupervised learning by using N-Grams

Shalle Soe Than, Win Pa Pa
University of Computer studies, Yangon
shallesoethan@gmail.com,winpapa76@gmail.com

Abstract

A separable verb is a verb that is composed of a verb stem and a separable affix which is called verb affixes. Mining of verb affixes is to extract the various forms of verb affixes from any Myanmar sentences. A Method for mining verb affixes in Myanmar language is presented in this paper. An unsupervised learning approach is used to extract affixes for various forms of verbs from Myanmar sentences since the verb affixes are supported to restrict verb from any Myanmar sentences. Myanmar texts sentences are analyzed to mine various forms of verbs. Unsupervised learning is the task of learning without labeled data in NLP application. Since Myanmar sentences are written with no special delimiter such as space to indicate word boundary, merging based syllable segmentation method is used to indicate word boundary. Verb affixes are extracted from these sentences by using N-Grams method. The system is implemented using java.

Keyword: *Unsupervised learning, N-Grams, Affixes, and Syllable based merging.*

1. Introduction

Morphological analysis is concerned with the process of segmenting a given corpus of words into a set of morphemes. Morphemes are the smallest units bearing a meaning in a word. In a corpus, the quantity of different morphemes is usually smaller than the quantity of different words. The purpose of the Morpho Challenge is to learn these morphemes using an unsupervised algorithm on different languages. Such a morphological analyser is useful for various applications like speech synthesis, information retrieval or machine translation where a dictionary of morphemes must be provided. Indeed, creating a dictionary of morphemes for a speech synthesis application cannot be directly carried out from raw data on all languages [2]. Morphological units can be classified into two general classes, stems (or roots) and bound morphemes, which combine to create words using various kinds of operators. The linear affixing operator combines stems and bound morphemes (affixes) using linear ordering with possible fusion effects, usually at the seams [1]. Unsupervised learning of affixal morphology in a single language is a heavily researched problem [8].

Languages may be divided into three broad categories: isolating, agglutinative, and inflective languages. Isolating languages, such as Chinese, have little or no morphology and thus do not benefit from morphological analysis. Agglutinative languages, also known as agglomerative or compounding languages, are those in which basic roots and words can be combined to make new words. These languages, such as Turkish or Finnish, tend to have many morphemes. Inflectional morphemes are used to modify a word to reflect information such as tense [3]. Myanmar language may be agglutinative language and inflective language because Myanmar word can be combined to make new word. eg: Myanmar word လူ and ငယ် is isolating word. But, if they are combined, လူငယ် is new word for each of the word and Myanmar verb is not easy to define tense like English. So, in this paper, affixes extraction for various form of verb is presented.

The set of contextually similar verbs often contains verbs with the same affix and there are many affixes for one verb. Thus, unsupervised learning method is more suited than supervised learning method to apply in this paper. Affixes extraction can be applied to decide the same-stem for words in morphological analyser. So, in this paper, affixes for various forms of verbs are extracted from Myanmar sentences because Myanmar verb is very complex to decide the stem of it. Before Affixes are extracted from any Myanmar sentences, we segment these sentences by using merging based syllable segmentation method because in Myanmar language, any Myanmar sentences are written with no special delimiter such as space to indicate word boundary. And then, we extract affixes from these sentences by using N-Grams method. Finally we get a list of

affixes from each sentence with unsupervised learning using Unicode standard encoding.

This paper is organized as follows. Section 1 is the introduction section. Section 2 will explain the related works. Section 3 will present theories of merging based syllabification and unsupervised learning N-Grams. Section 4 will present Implementation of the System, Section 5 is the Evaluation of the system and Section 6 is conclusion.

2. Related Work

The squares algorithm learns the cross-language correspondence between affixes and letter sequences. This algorithm discovers affixes and their pairing simultaneously. The squares algorithm is applied for affix learning in a single or cross-language [1]. Extraction of morpheme sequences is a hard task in languages where the word form includes sequences of prefixes or suffixes (affixes) and stem. Firstly, the boundaries of the stem are found, it is possible to extract prefix or suffix (affixes) sequences by MDL algorithm. Each word from a corpus, the left side of it is to extract the prefixes and the right side of it is to extract the suffixes [2, 10, 14].

MDL algorithm first splits some words in two and treats the first piece as a stem and the second as an affixes. For each stem, it builds a cluster of affixes. Then, it associates to each cluster of affixes a number of stems that appear with that cluster of affixes. For example, if there is a large set of words ending in ***-ion*** and ***-ive***, the function described affixes ***-on*** and ***-ve*** in these cases, and place the ***-i*** in the stems, not in the affixes [7, 8]. Unsupervised same-stem decision algorithm proceeds in two phases. In the first phase, a ranked list of salient affixes is extracted from an unlabeled text corpus of a language. In the second phase, an input word pair is aligned to shortlist affixes that could potentially be added to a common stem to alternate between the two. This analysis depends strongly on the ranked affix list from the first phase. For example, the word pair ***sting*** and ***station*** align to ***-ing*** and ***-ation*** which are both salient affixes but they do not systematically contrast [5,6].

For a given input word form is segmented into morphs (as opposed to morphemes) by letter successor variety (LSV) using contextual similarity of word forms based machine learning step. It is based on the assumption that any grammatical function is expressed with only a small amount of different affixes. For example, plural is expressed with only five different morphs in German ***-en***, ***-s***, ***-e***, ***-er*** and (***zero***) [15].

3. N-Grams with Unsupervised learning

3.2 Unsupervised Learning

Unsupervised learning means learning without knowing where morpheme borders are, or which morphemes exist in which word. Unsupervised methods avoid labor intensive annotation required to produce the training materials for supervised methods [4].

Unsupervised methods have advantages for less-studied languages, but for the well established languages, we have access to fair amounts of training material in the form of analyzes in context of more frequent words. In addition, there are a host of large but shallow hand-made morphological descriptions available. Unsupervised learning methods have many attractive features for morphological modeling, such as language-independence, independence of any particular linguistic theory, and easy portability to a new language [15]. Thus, verb affixes are not needed to predefine in this system by applying unsupervised learning method.

3.3 Merging based Syllable Segmentation

The writing system of Myanmar language does not use any delimiter to explicitly indicate word boundaries. Thus, in this paper, any Myanmar sentences are segmented by merging based syllabification. Merging based syllable segmentation method used syllabic words file that store possible syllabic words to segment any Myanmar sentences by comparing these syllabic words with every Myanmar sentences.

If the input sentence is

"ကျောင်းသားတိုင်းစာကျက်သင့်သည်။"

It will be syllabificated as

"ကျောင်း-သား-တိုင်း-စာ-ကျက်-သင့်-သည်-။"

And then, N-Grams method is applied to extract verb affixes from Myanmar sentences.

3.4 Mining of Verb Affixes

Affixes mining is the important task of morphological analyser in NLP application such as same stem decision translate from one language to the cross-language, classify the word type from any language etc. In English, if we have the words governed, governing, government, governor, governs, and govern in that corpus, **govern** is (stem) verb and affixes are **ing, s, ment**, or but all affixes are not verb affixes. Because if **govern** and **ment** are combine, government is became but is not Verb. This is Noun. Thus, every combination of verb and affixes are not verb affixes.

Having a list of salient affixes is not sufficient to parse a given word into stem and affix (es). For example, **sing** happens to end in the most salient suffix yet it is not composed of **s** and **ing** because crucially, there is no ***s, *sed** etc. Thus to parse a given word we have to look at additional evidence

beyond the word itself, such as the existence of other inflections of potentially the same stem as the given word, or further, look at inflections of other stems which potentially share an affix with the given word [5].

In the same way, Myanmar language can be mined verb affixes from any Myanmar sentences. eg: shown in Table 1.

Table 1. Mining affixes from various patterns of verb

various patterns of verb	verb affixes
အိပ်သည်။	**သည်။**
အိပ်ပါသည်။	**ပါသည်။**
အိပ်နေပါသည်။	**နေပါသည်။**
အိပ်ခဲ့ပါသည်။	**ခဲ့ပါသည်။**
အိပ်လိမ့်မည်။	**လိမ့်မည်။**
အိပ်သလား။	**သလား။**
အိပ်ပါတယ်။	**ပါတယ်။**
မအိပ်ပါဘူး။	**ပါဘူး။**
အိပ်ရမည်။	**ရမည်။**
အိပ်ရမှာပေါ့။	**ရမှာပေါ့။**
အိပ်နေ၏။	**နေ၏။**
အိပ်နိုင်သည်။	**နိုင်သည်။**
အိပ်ပြီးပြီ။	**ပြီးပြီ။**
အိပ်မည်မဟုတ်ပါ။	**မည်မဟုတ်ပါ။**
အိပ်ပေတော့။	**ပေတော့။**
အိပ်ကြလေ၏။.....etc	**ကြလေ၏။**

eg: In အိပ်သည်။, အိပ် is stem and **သည်။** is affix and in အိပ်ကြလေ၏။, အိပ် is stem and **ကြလေ၏။** are affixes and they all are verb affixes. In the word အိပ်ခြင်း, ခြင်း is the affixes of အိပ် but အိပ်ခြင်း are not verb. This is noun. But, there are Myanmar verb affixes at the end of Myanmar sentences and Myanmar verb (stem) is very complex to define. eg:

လုပ်အားပေးနေသည်။
အားပေးနေသည်။
ပေးနေသည်။
နေသည်။.

Thus, verb affixes are mined from any Myanmar sentences by using N-Grams method in the system. Eg: if the inputs sentence is

မောင်မျိုးနှင့်အောင်တိုးတို့သည်ကျောင်းသားများဖြစ်ကြပါသည်။

Verb affixes of these sentence is

ကြပါသည်။.

All verb affixes that mine from any sentences in this system can be applied at the decision of any verb from any Myanmar sentences.

3.5 N-Grams Method

An n-gram is a subsequence of n items from a given sequence. The items in question can be syllables, letters, and words according to the application. An *n*-gram of size 1 is referred to as a "unigram"; size 2 is a "bigram" (or, less commonly, a "digram"); size 3 is a "trigram"; and size 4 or more is simply called an "n-gram". An n-gram model is a type of probabilistic model for predicting the next item in such a sequence.

N-grams are used in various areas of statistical Natural Language Processing and genetic sequence analysis such as Speech recognition, Spelling collection, Handwriting recognition and Information retrieval.

In this system, we apply word based N-Grams to extract affixes. N-Grams are sequences of N consecutive syllables extracted from the Myanmar sentences. If the sequence of syllables longer than 2 or 3, then we use the CHAIN RULE:

$$P(W_1 \ldots W_n) = P(W_1)\,P(W_2|W_1)\,P(W_3|W_1 W_2)\ldots\ldots P(W_n|W_1 \ldots W_{n-1}) \quad \text{---------- (1)}$$

where,

$P(W_1 \ldots W_n)$ = The probability of $W_1 \ldots W_n$ (syllables) from the corpus.

The whole system will be applied by the following N-Grams equation.

$$P(W_n|W_{n-1,1}) = C(W_{n,1}) / C(W_{n-1,1}) \quad \text{---------- (2)}$$

where,

$P(W_n|W_{n-1,1})$ = The probability of W_n (syllable) by position from the corpus.

W_n = The syllable word of n position from the current sentence.

$W_{n-1,1}$ = The suffix syllable words of W_n from the current sentence.

$C(W_{n,1})$ = The count of $W_{n,1}$ (syllable) by position from the corpus.

$C(W_{n-1,1})$ = The count of $W_{n-1,1}$ (syllable) by position from the corpus.

In the system, following trigram method is applied.

$$P(W_n | W_{n-1}, W_{n-2}) = C(W_n W_{n-1} W_{n-2}) / C(W_{n-1} W_{n-2}) \quad \text{---------- (3)}$$

where,

$P(W_n | W_{n-1}, W_{n-2})$=The probability of W_n (syllable) by position from the corpus.

W_n = The syllable word of n position from the current sentence.

W_{n-1}, W_{n-2} = The suffix syllable words of W_n from the current sentence.

$C(W_n W_{n-1} W_{n-2})$ = The count of W_n W_{n-1} W_{n-2} (syllables) by position from the corpus.

C (W_{n-1} W_{n-2}) = The count of $W_{n,1}$ W_{n-2}(syllables) by position from the corpus.

eg: ကွန်ပျူတာတက္ကသိုလ်တွင်ကျောင်းဆောင်ပေါင်းများစွာတည်ဆောက်ခဲ့ပါသည်။

The probability of "▮" must be assumed 1 because the end of every sentences is "▮".

If **သည်** are calculated by applying tri-grams equation.

P (သည်|▮@) =C (သည်|▮@)/C (▮@)

And **ပါ** are calculated by applying above equation.

P (ပါ|သည်▮) =C (ပါသည်▮)/C (သည်▮)

Then, **ခဲ့** are calculated the following

P (ခဲ့|ပါသည်) =C(ခဲ့ပါသည်)/C(ပါသည်)

Finally, **ခဲ့ပါသည်▮** are calculate from these Sentences by applying chain rule with N-Grams Method.

P (ခဲ့|ပါသည်▮@) = P(ခဲ့|ပါသည်)*P(ပါ|သည်▮)*P(သည်|▮@)

If the affixes probability is less than the threshold probability, the affixes is will be extracted as following.

P (ဆောက်|ခဲ့ပါသည်▮@) =

P(ဆောက်|ခဲ့ပါ)*P(ခဲ့|ပါသည်)*P(ပါ|သည်▮)*P(သည်|▮@)

P (တည်ဆောက်ခဲ့ပါသည်▮@) =

P(တည်|ဆောက်ခဲ့)*P(ဆောက်|ခဲ့ပါ)*P(ခဲ့|ပါသည်)*P(ပါ|သည်▮)

***P(သည်|▮@)** ……..etc

4. Implementation of the System

4.1. Overview of the System

Myanmar sentences that apply in the system are downloading from the internet journal and newspaper and they include various kinds of sentences such as structural sentences, spoken sentences.

The system works as follows:

First, unlabeled Myanmar sentence or sentences are applied to mine verb affixes from these. And then, these sentences are segmented as syllabic word by using syllable segmentation based on merging method which is applied syllabic data by merging each sentences. Then, N-Grams method is used to mine verb affixes from these syllable word sequences. Finally, verb affixes are mined from each sentence by using N-Grams method with unsupervised learning as shown in Figure 1.

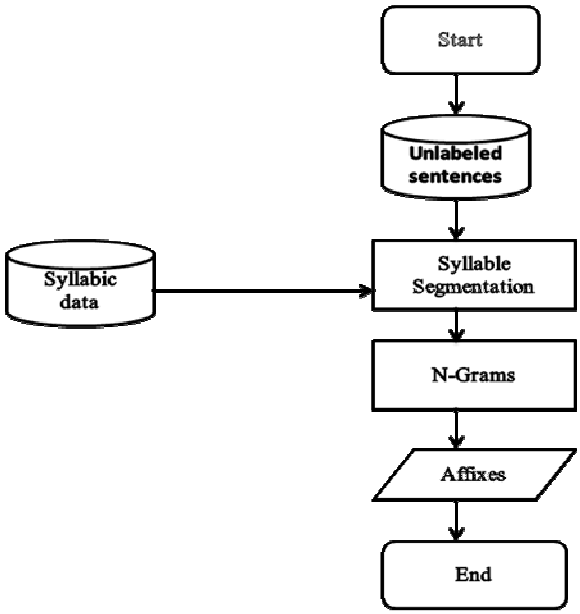

Figure 1. The System Overview

4.2. Syllable segmentation of the System

Myanmar sentences corpus is trained from text file as shown in Figure 2.

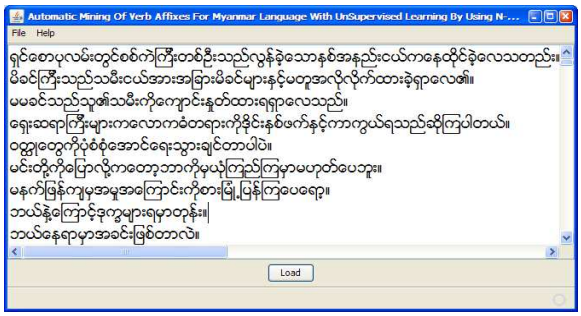

Figure 2. Loading sentences corpus from text file

And then, each Myanmar sentences are segmented as syllabic word by merging syllabic data as shown in Figure 3.

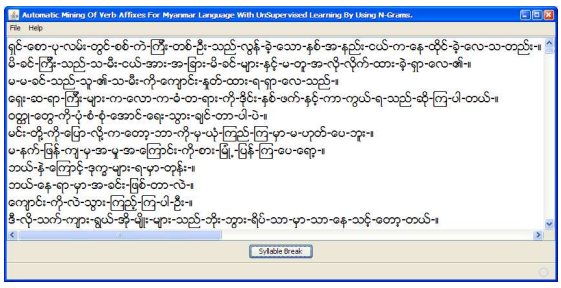

Figure 3. Each sentences segment as syllabic word

4.3. Affixes mining of the System

Any Myanmar sentences can be mined verb affixes by using Tri-Grams method by unsupervised learning. The result is shown in Figure 4. In the system, at least three syllabic words must be had to mine verb affixes from each sentences because trigram method are applied. And, at the end of each sentence is calculated to mine verb affixes by using Tri-Grams because there are verbs at the end of sentences in Myanmar language. Each syllabic word must be defined position that have own weight to apply Tri-Grams method shown in Table 1.

In the paper, this weight values are suited by analyzing various kinds of 5918 Myanmar sentences but the values of weight can be varied by kind of sentences and total number of sentences.

For example: the testing sentence is **မမသည်အလွန်စာတော်ပါသည်။**
The weight and position of syllable word from these sentence as shown in Table 2.

Table 2. Position with Weight

Position	Weight
1	1
2	1
3	1
4	0.95
5	0.7
6	0.65
7	0.4
8	0.35
9	0.1
10	0.095

Table 3. Position with Weight for testing sentence

Syllable words from sentence	Position	Weight
။	2	1
သည်	3	1
ပါ	4	0.95
တော်	5	0.7
စာ	6	0.65
လွန်	7	0.4
အ	8	0.35
သည်	9	0.1
မ	10	0.095
မ	11	0.07

The calculation of testing sentence is as following.

(3,2,1#သည်။@/2,1#။@)((1882*1.0=1882.0)/5918)
result=0.31801284217641096*(2,1,0#။@@/1,0#@@)1.0=0.31801284217641096

The result probability 0.31801284217641096 is greater than threshold probability **(0.0008)** that analyze Myanmar sentences. So, we will calculate next position. In the paper, the threshold probability is suited by analyzing various kinds of 5918 Myanmar sentences but it also can be varied by kind of sentences and total number of sentences.

(4,3,2#ပါသည်။/3,2#သည်။)((445*0.95=422.75)/1882)
result=0.2246280552603613*(3,2,1#သည်။@/2,1#။@)0.31801284217641096=0.0714346062859074
This result probability 0.0714346062859074 is also greater than. So, we calculate next position.
(5,4,3#တော်ပါသည်/4,3#ပါသည်)((2*0.7=1.4)/445) result
=0.003146067416*(4,3,2#ပါသည်။/3,2#သည်။)
0.0714346062859074=2.247380872E-4

Then, the result probability is 2.247380872E-4 less than the threshold probability that analyze Myanmar sentences. Thus, the result of verb affixes is **ပါသည်။** with probability 0.0714346062859074 as shown in Figure 4.

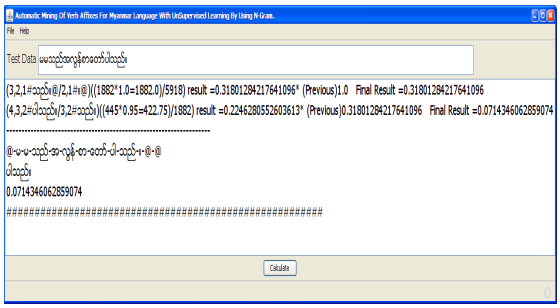

Figure 4. Automatic mining of verb affixes from sentence

5. Evaluation

5.1. Accuracy evaluation

In this system, two standard metrics *exact accuracy* and *Fscore* are used to evaluate the performance of our system on the test sentences. Exact accuracy is the percentage of the verb affixes whose proposed Extraction (Ep) is identical to the correct Extraction (Ec). F-score is simply the harmonic mean of recall and precision, as computed using the formulas below.

Precision = (H) / (H+I) ---------- **(4)**
Recall = (H) / (H+D) ---------- **(5)**
F-score = (2H) / (2H+I+D) ---------- **(6)**

where,
H is the number of Hits (*i.e.*, correctly placed boundaries)

I is the number of morpheme boundaries needed to be inserted into Ep, respectively, to make it identical to Ec.
D is the number of morpheme boundaries needed to be deleted from Ep, respectively, to make it identical to Ec [3].
Precision= 5600 / (5600+1079) = 0.838448869
Recall = 5600 / (5600+761) = 0.880364722 then,
F-score is
F-score = (2*5600)/ ((2*5600) +1079+761) =0.8588 95706= 86%

Thus, exact accuracy is 93% and F-score is 86%. In this system, there are total various kinds of 5918 Myanmar sentences. They are applied to extract verb affixes from these sentences. And there are mining of about 240 verb affixes. The errors of this paper are as following example.

သူများကိုဆုံးမလွန်းလို့ဝင့်လည်တာထင်ပါရဲ့။ .....etc.

These sentences are mined "**။**" like affixes and there are about 100 sentences like this error by 5918 sentences.

တစ်ကမ္ဘာလုံးကြိုက်တဲ့သီချင်းကိုတောင်လက်မခံပါလား**ဗျို့။** .....etc.

These sentences are mined "**ဗျို့။**" like affixes but this affixes is not true for this sentences and there are about 200 sentences like this error by 5918 sentences.

လှလှမနေ့ကကျောင်းသွား **ခြင်္ဂပါတယ်။**
မောင်မောင်သည်ကျောင်းသား **ခြင်္ဂပါတယ်။** ……..etc

These sentences are mined "**ခြင်္ဂပါတယ်**" like affixes but this affixes is true for first sentence and false for second sentence. There are about 1079 sentences like this error by 5918 sentences.

ကလေးများပျော်ရွှင်စွာ **မ** ကစား **ပေဘူးဗျို့။** ……..etc

These sentences are mined "**ပေဘူးဗျို့။**" like affixes but do not "**မ**" (prefix) with verb affixes in this system. There are about 761 sentences like this error by 5918 sentences.

6. Conclusion

This system mines verb affixes by applying Tri-Grams method with unsupervised learning from any Myanmar sentences. And, this system can help for many NLP applications. In future, this system can solve the errors and can extract correct verb for Myanmar language using Unicode standard encoding by applying Myanmar Orthography.

7. References

[1] A. Rappoport, "Induction of cross-language Affix and Leteer Sequence correspondence." Institude of Computer seience The Hebrew University.

[2] B. Golenia, "UNGRADE:UNsupervised GRAph DEcomposition."Machine Learning Group, Computer Science Department, University of Bristol, UK.

[3] C. Jordan, "Swordfish2:Using Kernel Density Estimation to Smooth N-Grams Histogram for Morphological analysis." Faculty of Computer Science, Dalhousie University 6050 University Avenue.

[4] C. H. Parkes,"Toward Unsupervised Extraction of verb paradigms from large Corpora" Department of Computer & Information Science University of Pennsylvania.

[5] H. Hammarstrom, "Poor Man's Stemming: Unsupervised Recognition of Same-Stem Words." Chalmer university, 412 96 Gothenburg Sweden.

[6] H. Hammarstrom, "A survey and classification of Method for (Mostly) Unsupervised learning of Morphology." Dep. of Computing Science Chalmer university, 412 96 Gothenburg Sweden.

[7] H. Blancafort, "Learining Morphology of Romance, Germanic and Slavic Language with the tool linguistica." Syllabs 15, rue Jean Baptiste Berlier, 75013 Paris, France .Universitat Pompeu Fabra Roc Boronat,138, 08018 Barcelona, Spain.

[8] J. Goalsmith, "An Algorithm for Unsupervised Learning of Morphology." Departments of Linguistics and Computer Science 1010 East 59th St. The University of Chicago Chicago IL 60637 USA, (October 2005).

[9] J. Smarr, "Automatic Classification of Previously Unseen Proper Noun Phrases into Semantic Categories Using an N-Gram Letter Model."CS 224N Final Project, Stanford University, Spring 2001.

[10] Kvnsunihta and Nkalyani, "Improving word coverage by using unsupervised morphological analyser ." *Sadhana* Vol. 34, Part 5, October 2009, pp. 703–715. © Indian Academy of Sciences.

[11] M. Mansur, "Analysis of N-Gram Based Text Categorization for Bangla in a newspaper Corpus." BRAC University, Dhaka, Bangladesh.

[12] O. Kohonen, "Allomophessor: Toward Unsupervised Morphe Analysis." Adaptive Informatics Research Centre, Helsinki University of Technology.

[13] P. Limcharoen, "Thai Word Segmentation based-on GLR Parsing Technique and Word N-gram Model."

[14] S. ARGAMON, "Efficient Unsupervised Recuisive Word segmentation Using Minimun Description Length." Illions Institude of Technology, Dep. of Computer seience, Chicago IL 60616, USA.

[15] S. Bordag, "Unsupervised and Knowledge-free Morpheme Segmentation and Analysis."

[16] Y. Singer, "Beyon Word N-Grams." Institude Computer Science,Hebrew University.