The 2015 IEEE RIVF International Conference on Computing & Communication Technologies
Research, Innovation, and Vision for Future (RIVF)

Identifying Reduplicative Words for Vietnamese Word Segmentation

Ngoc Anh, Tran
Dept. Information Technology
Le Quy Don Technical University
Hanoi, Vietnam
anhtn69@gmail.com

Phuong Thai, Nguyen
Dept. Information Technology
UET, Vietnam National University
Hanoi, Vietnam
thainp@vnu.edu.vn

Thanh Tinh, Dao
Dept. Information Technology
Le Quy Don Technical University
Hanoi, Vietnam
tinhdt@mta.edu.vn

Hong Quan, Nguyen
Dept. Information Technology
Quang Ninh Industrial University
Quang Ninh, Vietnam
cdmhongquan@gmail.com

***Abstract*—This paper proposes a method based on linguistic word-formation rules and dictionaries for determining reduplicative words in Vietnamese. The key idea for identifying whether adjacent syllables in a text can form a reduplicative word based on its formation rules. For 2-syllable reduplicative words, this paper uses rules that describe the repeating and the opposing between pairs of initial consonants, rhymes and tones. Then the method is expanded to identify reduplicative words that have 3 or 4 syllables from 2-syllable ones for the Vietnamese word segmentation task. Experimental results showed that the F1-score was improved to 98.61% and that word segmentation errors were reduced significantly, 1.26%.**

Keywords—reduplicative word; reduplicative rules; Vietnamese word segmentation

I. INTRODUCTION

Vietnamese word segmentation (VWS) is one of the fundamental problems in natural language processing. Structurally, a Vietnamese word is often composed of one or more syllables, so the space does not distinguish the words like English and many other languages. On the other hand, word boundaries and meanings depend on its order[12], splitting or combination, and context, for example, its left and right words. Thus, the determining word boundaries is a difficult task, especially to deal with ambiguity and to identify new words.

For example, with the input:

Mọi người chuẩn bị đón tiếp tân Thủ tướng

The output will be:

Mọi người chuẩn_bị đón_tiếp tân Thủ_tướng .

People prepare to welcome new Prime Minister .

For years, the VWS has been studied by many different approaches such as: maximum matching by dictionary [15], machine learning with supervised [3], [13] or unsupervised [8], [11], and especially, the hybrids, combinations of them together for better results ([2],[5],[10],[17],[18],[19],[21]). There are two difficult problems in VWS: (1) Identifying new words; (2) Solving the ambiguity of word boundaries.

For problem (2), the ambiguities of word boundaries have been researched and solved in [2][10][17][18][19]. The problem (1) is studied by statistical methods from corpora, particularly, unsupervised learning in [8] and [11].

For new complex words which do not exist in the training corpus and the lexical dictionary, we can not use statistical methods or dictionary for identifying them. One of the methods of determining new words that linguists often use is based on the formation rules of complex word in the linguistic. By [4], Vietnamese words can be classified as shown in Figure 1.

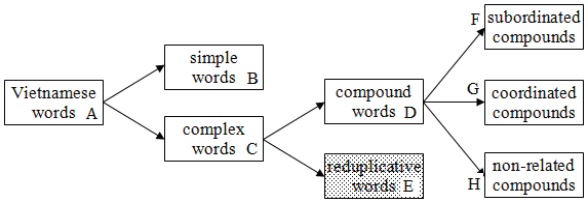

Fig. 1. Vietnamese word classification according to word formation

Recently, [20] proposed a method to identify coordinated compounds (block G in Fig.1) for VWS by using rules describing possible structures of Vietnamese words. The approach is as follows: if two adjacent simple words are the same in part-of-speeches (POS), are synonyms or antonyms, or are highly similar by their definitions in the Vietnamese Computational Lexicon (VCL)[22], then the sequence of these two syllables can be a candidate of coordinated compounds. The authors of [20] also used mutual information of two adjacent simple words from internet (online web pages) to verify whether candidates are really coordinated compounds. Moreover, extension rules are used to identify compound words or idioms that have three or four syllables. Results of experiments showed that the approach in [20] is effective and improves the accuracy of Vietnamese word segmentation.

978-1-4799-8044-4/15/$31.00 ©2015 IEEE

Similarly, since the class of reduplicative words (block E in Fig.1) takes a significant proportion in Vietnamese vocabulary (about 10%, computed using the VCL), they need to be researched and identified. In particular, reduplicative words have distinct structural features of phonetic, while they also have common characteristics in the composition of the combinations include words/terms 3 to 4 syllables. In [9], the authors used finite-state automata to represent 2-syllable reduplicative words. However, the work in [9] did not discover new reduplicative words and did not evaluate the impact of this word kind on the accuracy of Vietnamese word segmentation.

Identifying new reduplicative words using linguistic rules is a rather new approach in VWS. Therefore, based on linguistic studies such as [1] and [6], this paper proposes several techniques to identify reduplicative words (block E, bold in Fig.1) which often cause errors in VWS.

In the VCL [22], the number of 2-syllable reduplicative words is much higher than the number of the 3-or-4-syllable reduplicative words. Furthermore, governed by linguistic rules, 2-syllable reduplicative words can generate many new 3-4-syllable words or idioms. Therefore, this paper proposes the solution to identify reduplicative words that include two phases as follows:

Phase 1. Building a dictionary of 2-syllable reduplicative words: This dictionary contains 2-syllable reduplicative words extracted from the VCL and new words. The new 2-syllable reduplicative words discovered by applying linguistic rules described in linguistic literatures such as [1][6] and using the mutual information to determine their existence. (in Section II).

Phase 2. Applying extension rules described in [1][6] to identify 3-to-4-syllable reduplicative words for VWS: These rules make use of the generation capability of 2-syllable reduplicative words in the dictionary built from Phase 1. (in Section III).

The rest of this paper is organized as follows. In Section II (Phase 1), rules for identifying new reduplicative words are represented. After that, in Section III (Phase 2), extension rules are applied to VWS. Then in Section IV, experiments and evaluation results are presented and discussed. Finally, in Section V we give a number of conclusions and future works.

II. IDENTIFYING REDUPLICATIVE WORDS

To identify reduplicative words, first, structural characteristics need to be considered. After that, structural rules will be presented and propose solutions to perform that.

A. Structural Characteristics of Reduplicative Words

According to [1], every reduplicative word (RW), due to its particular structure, is composed of two parts: a root is and a reduplicative part, which is the repeat of the root. Syllables in a reduplicative word do not necessarily have a meaning. However, in many cases, the root syllable has a clear meaning, while reduplicative syllables have vague meanings even meaningless. And determining the root syllable or the reduplicative syllable in a word without clear meaning syllables, usually based on the status of the same type of reduplicative words that contain a clear meaning syllable. For example: "*ngay ngáy*/anxious" in the same style "*tôi tối*/slight dark", "*chầm chậm*/slowly", and "*thình lình*/suddenly" in the same type "*lòng thòng*/dangling"..., i.e the root syllable is behind the reduplicative syllable [1][6].

In the VCL, reduplicative there are 3411 reduplicative words (with 3933 meanings), including: 3215 2-syllable reduplicative words, 12 3-syllable reduplicative words, 184 4-syllable reduplicative words.

Example (Ex.): *lấp lánh/sparkle, gọn gàng/tidy, lơ tơ mơ/vague, rát ràn rạt/keen, ấm a ấm ức/displeased, lử đử lừ đừ/fatigue*,...

Hoang V.H.[6] collected and classified reduplicative words into 10 different patterns: 8 patterns for 2-syllable RWs, 2 patterns for 3- or 4- syllable RWs.

Obviously, 3 or 4 syllable reduplicative words have both structure and meaning originated from 2-syllable reduplicative words. They have the ability to generate very strong that can be hard to list all. For example:

- 3-syllable reduplicative words "*xốp xồm xộp/very spongy*" and "*lơ tơ mơ/very vague*" from "*xốp xộp/spongy*" and "*lơ mơ/ vague*" respectively.

- 4-syllable reduplicative words "*hì hà hì hục / very zealously*", "*hăm hăm hở hở / very eagerly*" from "*hì hục / zealously*", "*hăm hở / eagerly*" respectively.

So, the next section will present the rules to identify 2-syllable RWs based on references [1][6][16] and the VCL[22].

B. Identifying and Building a Dictionary of 2-syllable Reduplicative Words

Based on reference of RWs in VCL dictionary and the rules that identify RWs: reduplicating the whole, reduplicating initial consonant and reduplicating the rhyme based on repeating style and opposing style [1][6]. By searching 2-syllable RWs in the large corpora as VietTreeBank corpus [14] and Vietnamese raw large corpus to create a dictionary of 2-syllable RWs. To perform that, We build a module to identify two adjacent syllables (A1 A2) are RW whether or not.

We analyze two syllables into components: initial consonants (P), rhymes (V) and tones (D) as follows:

A1 A2 = (P1, V1, D1) (P2, V2, D2).

Symbol of 6 tones Vietnamese:

(level, curve, falling, broken, rising, drop) = (- \ ? ~ / .)

and symbol "|" is for OR operator.

On that basis, with 8 rules for 2-syllable RWs, which [6] has found out the repeating and opposing rules as follows:

Rule 2.1: repeating completely: repeating the initial consonant, repeating the rhyme, repeating the tone.

P1 = P2; V1 = V2; D1 = D2 = (-) | (\ )

Ex: *lăm lăm/attempt, hao hao/slighlly like, kìn kìn/in flocks.*

Rule 2.2: repeating completely: repeating the initial consonant, repeating the rhyme, opposing the tone.

P1 = P2; V1 = V2; D1 D2 = (- ?) | (- /) | (\ ~) | (\ .) | (/ .)

Ex: *đo đỏ/slightly red, ngay ngáy/anxious,...*

Rule 2.3: repeating completely: repeating the initial consonant, opposing the rhyme, repeating the tone.

P1 = P2; D1 = D2; V1 opposes V2;

Opposing the rhyme in main vowel.

V1 V2 : (u - i)|(ô - ê)|(o - e)|(u - ơ)|(u - ă)|(ô - a)|(ê - a)

Ex: *chúm chím/smiling, gồ ghề/rough, thòm thèm/desirous*.

Rule 2.4: repeating completely: repeating the initial consonant, opposing the rhyme, opposing the tone.

P1 = P2; V1 opposes V2; D1 D2 = (- /) | (\ .)

opposing the rhyme in last consonant;

V1 V2 : (m - p) | (n - t) | (ng - c) | (nh - ch)

opposing the tone according to opposing the rhyme:

Ex: *ăm ắp/overbrimed, phơn phớt/slightly, vằng vặc/bright.*

Rule 2.5: repeating the component: opposing the initial consonant, repeating the rhyme, repeating the tone (the root is the second).

P1 opposes P2; V1 = V2; D1 = D2;

P1 P2 = (l – every consonant, except {n-, g-})|(b - nh)|
(b - l)|(b - ng)|(b-kh)|(b - r)|(ch - b)|(ch - h)|(ch - m)|
(ch - v)|(c/k - n)|(c/k-nh)|(kh - n)|(t - m)|(t - h)|(th - d)

Ex: *lòng thòng/dangling, lom khom/stoop* with the root *is: thòng/dangling, khom/stoop.*

Rule 2.6: repeating the component: opposing the initial consonant, repeating the rhyme, repeating the tone (the root is the first).

P1 opposes P2; V1 = V2; D1 = D2;

P1 P2 = (kh - l)|(th - l)|(ch - l)|(x - l)|(m - l)|(b - l)|(v - l)|
(t - l)|(x – r)|(k - r)|(kh - r)

Ex: *khéo léo/clever, thò lò/run* with the root is: *khéo/clever, thò/thrust*.

Rule 2.7: repeating the component: repeating the initial consonant, opposing the rhyme (the root is the second).

P1 = P2; V1 opposes V2; D1 D2 = (- ?)|(- /)|(\ ~)|(\ .)|(/ .)

V1 V2: with V1={a, âc, âm, ân, âp, e, i, o, ơ, ôn, ơn, uc, um, ung, ươt} (by [6])

Ex: *lập loè/blink, chí choé/argues* with the root is: *loè/bluft, choé/bright and translucent.*

Rule 2.8: repeating the component: repeating the initial consonant, opposing the rhyme (the root is the first).

P1 = P2; V1 opposes V2; D1 D2 = (- ?)|(- /)|(\ ~)|(\ .)|(/ .)

opposing the rhyme:

V1 V2: with V2={a, ac, ach, ai, am an, ang,…, ăc, ăn,...., ươm, ương, ươt} (by [6])

Ex: *đỏ đắn/in the pink, chắc chắn/reliable* with the root is: *đỏ/red, chắc/stable*.

With each rule, we use a list or array to save pairs together opposing.

C. Existence of 2-syllable Reduplicative Words

In the text, two syllables of 2-syllable RWs often appear side by side with some frequency. We can use the mutual information (MI) of two syllables of 2-syllable RWs to determine their existence. By [20], the mutual information of syllables can be defined as follows:

$$MI(A\,B) = \frac{C(A\,B)}{C(A) + C(B) - C(A\,B)} \quad (1)$$

where: + $MI(A\ B)$ is linking of two syllables ($A\ B$)

+ $C(A\ B)$ is the count of syllable bigram ($A\ B$)

+ $C(X)$ is the count of syllable unigram (X).

If ($A\ B$) is a candidate of 2-syllable RW and $MI(A\ B)$ is greater than threshold MI_0 then ($A\ B$) is a 2-syllable reduplicative word.

Subsections *II.B* and *II.C* give an algorithm follows:

The algorithm of looking for new RWs:

Step 1: carry out word segmentation for Vietnamese raw large corpus (54 MBs), then adding VietTreeBank (10 MBs).
Step 2: for each sentence in segmented corpus {
+ assign elements in the array of words $w[1..n]$
+ for each *word* in $w[1..n]$ {
if ($w[i]$ has 2 syllables, $w[i] \notin$ dictVCL) {
segment $w[i]$ to 2 syllables $A\ B$;
if (isRW2Rules(A, B))
add ($A\ B$) to RW2List;
}else if ($w[i]$ && $w[i+1]$ is 2 syllables) {
$A \leftarrow w[i]$; $B \leftarrow w[i+1]$;
if (isRW2Rules(A, B)) && ($MI(A\ B) \geq MI_0$) {
add ($A\ B$) to RW2List;
} // end if check words: $w[i]$ / $w[i]$ $w[i+1]$
} // end for each word in sentence
} // end for each sentence in corpus
Step 3: reorder and remove duplicate elements.
Step 4: print RW2List.

The result is a list of 1125 candidates of RWs. The linguistic experts evaluated and detected 101 errors. The assessment results in Table I.

TABLE I. RESULTS BY DETECTING NEW RWS.

No. of detected RWs	No. of corrected RWs	No. of errors	Precision (%)
1125	**1024**	101	91.02

Based on the result, we have discovered and added 1024 corrected new RWs, combined with 3215 RWs from VCL into the dictionary has 4239 2-syllable RWs.

III. APPLYING FOR VIETNAMESE WORD SEGMENTATION

A. Identifying 3-syllable Reduplicative Words

Analyzing 3 adjacent syllables to the initial consonant P, rhyme V and tone D. On this basis, applying some rules to identify 3-syllable RWs as follows:

A1 A2 A3 = (P1,V1,D1) (P2,V2,D2) (P3,V3,D3)

Rule 3.1: repeating the initial consonant, repeating the rhyme, opposing the tone.

P1 = P2 = P3; V1 = V2 = V3;

D1 D2 D3 = (? \ -)|(~ \ -)|(. \ -)|(/ \ \)|(~ \ \)|(- \ \)

Ex: *dửng dừng dưng /unconcern, mõm mòm mom / too ripe, tẹo tèo teo / tiny, đứ đừ đừ / stiff*.

Rule 3.2: repeating the initial consonant, repeating the rhyme in two sides, opposing (<>) the tone in two sides.

P1 = P2 = P3; V1= V3, V2 opposes V3; D1 D2 D3= (/ \ .)

Ex: *khít khìn khịt / close-fitting, tất tần tật / whole*.

Rule 3.3: repeating the initial consonant, repeating the two last rhyme, opposing the first tone with last tone.

P1 = P2 = P3; V2 = V3, V1<>V3; D1 D2 D3 = (/ \ \)|(. \ -)

Ex: *ngút ngùn ngùn / curl upwards, sạch sành sanh / completely empty*.

Rule 3.4: opposing the initial consonant, repeating the rhyme, repeating tone.

P1 P2 P3 = (t / l / m) | (l / t / m)

V1 = V2 = V3; D1 = D2 = D3 = (-) | (\ )

Ex: *tơ lơ mơ / vague, tờ lờ mờ / faint, lù tù mù / indicstint.*

B. Identifying 4-syllable Reduplicative Words

Analyzing 4 adjacent syllables to the initial consonant P, rhyme V and tone D. On this basis, applying some rules to identify 4-syllable RWs as follows:

A1A2A3A4 = (P1,V1,D1)(P2,V2,D2)(P3,V3,D3)(P4,V4,D4)

Rule 4.1: If AB is 2-syllable RW then AABB is 4-syllable RW.

Ex: *hối hối hả hả/hurriedly, vội vội vàng vàng/hastily*.

Rule 4.2: If BC is 2-syllable RW then ABAC is 4-syllable RW.

Ex: *đen thui đen thủi/coal black, cuống cà cuống kê / become rattled, thơm phưng thơm phức / delicious*,...

Rule 4.3: If AB is 2-syllable RW then AaAB is 4-syllable RW.

Pa = PB; Va = a; DA Da=(- -)|(/ -)|(? -)|(? \)|(\ \)|(~ \)|(. \)

Ex: *đủng đà đủng đỉnh/fishtail-palm, õng a õng ẹo/mincing*

Rule 4.4: AB is 2-syllable RW: opposing the initial consonant, repeating the rhyme, repeating the tone, include of:

VA = VB; DA = DB; PA opposes PB;

with cases as follows:

+ DA = DB = (\ ), DA' = DB' = (?): A'B'AB is a RW.

Ex: *tẩn ngẩn tần ngần /hang back, bổi hổi bồi hồi /fret*

+ DA = DB = (?), DA' = DB' = (\ ): ABA'B' is a RW.

Ex: *lảm nhảm làm nhàm/talk nonsense, lởm chởm lờm chờm/rugged.*

+ DA = DB = (.), DA' = DB' = (/): A'B'AB is a RW.

Ex: *loáng choáng loạng choạng/stagger, lốm cốm lộm cộm/disorder*.

+ DA = DB = (/), DA' = DB' = (.): ABA'B' is a RW.

Ex: *bắng nhắng bặng nhặng / fuss*.

Rule 4.5: If AB is RW when change phonetic, then A'B'AB, ABA'B' are RWs.

Ex: *lông bông lang bang / be on the tramp, bô lô ba la / at random, linh tinh lang tang / miscellaneous, lơ chơ lỏng chỏng / few and disorderly, lơ thơ lẩn thẩn / wander…*

With experiments on Vietnamese Corpora, identifying 3-to-4-syllable reduplicative words has achieved the precision is 100%. So, we take this identifying into an integrated method for Vietnamese word segmentation.

C. Integrated Method for Vietnamese Word Segmentation

The problem of VWS can be presented as follows: given a sentence as a sequence of *n* syllables:

$S = s_1\ s_2\ s_3\ \ldots\ s_{n-1}\ s_n$

Find an optimal sequence of segmented *m* words ($m \le n$):

$S = w_1\ w_2\ w_3\ \ldots\ w_{m-1}\ w_m$

To do that, [18] and [19] proposed a score model by integrating method as follows:

- Using a 2-dimension array score[1..*n*, 1..*n*] to score each word. If a sequence of syllables ($s_i \ldots s_j$) can be a word in the dictionary or training corpus or by linguistic rules then:

$score(w_{ij}) = score(s_i \ldots s_j) = score[i, j] = 1.$

With the maximal matching, the number of segmented words (m) is minimal. Each word has a score, hence the sum of their scores must be minimized. With this approach, then we need to initialize: $score[i,j] = +\infty$; $1 \le i, j \le n$.

$SC_k(S)$ is a score sum of segmentable words with *k*-scheme of sentence *S*. So, the dynamic programming formula will be:

$$\min\{SC^k(S)\} = \min\left\{\sum_{i=1}^{m^k} score(w_i^k)\right\} \qquad (2)$$

where, w_i^k is the *i*-word segmented by *k*-scheme.

m_k is number of segmented words by *k*-scheme.

- *The integrated algorithm for VWS as follows*:

Step 1. Using the maximal matching method with the VCL dictionary and subdictionaries of 2-syllable words (coordinated compounds and reduplicative words) to segment the input sequence. Each segmentable word has a score equal to 1.

Step 2. Detecting new words (complex words) that have 3 or 4 syllables with two groups of extension rules: (1) for coordinated compounds [20], and (2) for reduplicative words (in III.A and III.B) to identify and score them.

Step 3. Detecting ambiguities (OA - overlapping ambiguities or CA - combination ambiguities) and scoring them by word bi-gram probability or mutual information of syllable n-gram model in [18] and [19].

Step 4. Using a dynamic programming algorithm to find the optimal sequence of segmented words by the formula (2).

To speed up our word segmentation program and to reduce the memory of data, we do some works as follows:We implemented the dictionary using the minimum weight finite state automaton - MWFSA by [7] or [10], in which the value at the final states of MWFSA is the sum of the weights, and is used as the order of words in the dictionary. We used these two automata, one for the dictionary of 6950 syllables and one for the VCL dictionary of 31158 words. The syllable automaton is used for n-gram statistics and computing the *MI* of syllables, and the word automaton is used for maximum matching and computing the probabilities of word bi-grams.

In [19], the authors had done a statistics about word length distribution showed that the proportion of words composed of 5 or more syllables is about 0.01% in the VietTreeBank corpus. They do not significantly affect the accuracy of VWS. So, we choose a 5-syllables window for word segmentation. Hence, the time complexity of the dynamic programming algorithm by (2) is linear.

The algorithm of the formula (2) *as follows*:

```
Step 1. a[0] ← 0;
Step 2. for i ← 1 to n {
          a[i] ← + ∞; first ← 0;
          if (i > WinSize)
             first ← i – WinSize;
          for j ← first To i – 1  {// WinSize times
             w ← vw[j];
             if (a[i] > a[j] + score[j, i]) {
                a[i] ← a[j] + score[j, i];
                for k ← j + 1 To i – 1
                   w ← w + " " + vw[k];
             }  // end of if
             q[i] ← w;  //here is the result
          } // end of for j
       } // end of for i
```

Where:

+ *n* is number of syllables in the input sentence

+ *WinSize* is the size of syllable windows

+ *vw*[*j*] is the j^{th} syllable

+ *score*[*j, i*] is score of word that include j^{th} to i^{th} sylables

+ *a*[ ] is a template array.

+ *q*[ ] is the result of word segmentation.

Clearly, for *WinSize* = 5, the time complexity of above algorithm is *O*(*n*).

Our VWS program includes a number of modules as described in Table II. Most of the intergrated modules in [18], [19], [20], only one new module RW in Table II. (bold).

TABLE II. DESCRIBE MODULES OF WORD SEGMENTATION

Modules	Describe
FMM	Forward Maximum Matching
BMM	Backward Maximum Matching
MM	Advanced Maximum Matching
NE	Named Entities
MI	Mutual Information of syllables
Pb	Probability of word bigram
CC	Coordinated Compounds
RW	**Reduplicative Words**

IV. EXPERIMENTS AND EVALUATION

A. Resources and Evaluative Method

For experiments we used the following resources:

- The VCL [22] is used for word segmentation by maximal matching with 31,158 words. A dictionary of 2-syllable coordinated compounds (4454 words) and a dictionary of 2-syllable reduplicative words (4239 words).

- A corpus for word segmentation training and testing: The corpus VietTreeBank [14] includes 70.000 sentences, for a total of 1,547,387 segmented words. The corpus is divided into two parts: (1) 70% was used for training in order to calculate the mutual information MI (mutual information) based on n-gram syllable statistics, to calculate the bigram word probabilities. (2) 30% are used for testing.

- Evaluation:

+ P (Precision): $P = \frac{\text{the number of correct words}}{\text{the number of output words}}$

+ R (Recall): $R = \frac{\text{the number of correct words}}{\text{the number of words in corpus}}$

+ F_1-score: $F = \frac{2PR}{P+R}$

+ ErrR = (No. of words in corpus) - (No. of correct words)

B. Results

Several Vietnamese word segmentation experimental results are taken from [18][19][20]. The difference is that in this study, we add a module for identifying reduplicative words (RWs). Test results are shown in Table III.

TABLE III. RESULTS OF VIETNAMESE WORD SEGMENTATION AND COMPARISONS

	Methods	ErrR	δEr %	R (%)	P (%)	F (%)	δF %
*	FMM	20079		95.57	92.09	93.80	
	BMM	19213		95.76	92.27	93.98	
	FMM+NE	9799		97.84	96.97	97.40	
	BMM+NE	8956		98.02	97.16	97.59	
i	NE+MM	8954		98.02	97.16	97.59	
	NE+MM+**RW**	8832	*-1.36*	98.05	97.22	97.63	*0.04*
	NE+MM+CC	7903		98.26	97.77	98.01	
	NE+MM+CC+**RW**	7792	*-1.40*	98.28	97.82	**98.05**	*0.04*
ii	NE+MM+MI	8616		98.10	97.20	97.65	
	NE+MM+MI+**RW**	8494	*-1.42*	98.13	97.26	97.69	*0.04*
	NE+MM+MI+CC	7539		98.34	97.81	98.07	
	NE+MM+MI+CC+**RW**	7428	*-1.47*	98.36	97.87	**98.11**	*0.04*
iii	NE+MM+Pb	5795		98.72	97.98	98.35	
	NE+MM+Pb+**RW**	5719	*-1.31*	98.74	98.02	98.38	*0.03*
	NE+MM+Pb+CC	5539		98.78	98.18	98.48	
	NE+MM+Pb+CC+**RW**	5468	*-1.28*	98.79	98.21	**98.50**	*0.02*
iv	NE+MM+MI+Pb	5876		98.70	98.22	98.46	
	NE+MM+MI+Pb+**RW**	5800	*-1.29*	98.72	98.26	98.49	*0.03*
	NE+MM+MI+Pb+CC	5624		98.76	98.41	98.59	
	NE+MM+MI+Pb+CC+**RW**	5553	*-1.26*	98.77	98.45	**98.61**	*0.02*

(i) Only using the VCL with NE and MM (+CC) for VWS. Module RW increased F1-score to 0.04%, and reduced the number of errors from 1.36% to 1.4%.

(ii) Using the VCL with NE, MM (+CC) and a raw corpus for calculating MI by syllables n-gram. Module RW increased F1-score to 0.04%, and reduced the number of errors from 1.42% to 1.47%.

(iii) Using the VCL with NE, MM (+CC) and the VietTreeBank corpus to calculate the probability Pb. Module RW increased F1-score from 0.02% to 0.03%, and reduced the number of errors from 1.28% to 1.31%.

(iv) Using the VCL with NE, MM (+CC), syllable mutual information MI and bigram word probabilities Pb. Module RW increased F1-score from 0.02% to 0.03%, and reduced the number of errors from 1.26% to 1.29%.

Thus, the number of errors decreased rather consistently. Obviously, when adding the RW module, the results of VWS are better than before. (columns δF and δEr in Table III).

The following is an illustrated example for Vietnamese word segmentation. The example includes 7 sentences:

Chủ tịch UBND Thành phố Hà Nội Nguyễn Thế Thảo đã đi ...

Trường Đại học Bách khoa HN dẫn đầu phong trào ...

Tất cả chúng ta đang chuẩn bị đón tiếp tân Thủ tướng .

Họ đi một vòng quanh thành phố .

Họ đã vượt qua bao sông suối , thác ghềnh để đến đây .

Họ đã có cơm ăn áo mặc , không phải đi mưa về nắng nữa .

Từ láy : bùm bụp , cuống cuồng cuồng , khúc kha khúc khích .

The results of VWS for 7 sentences above:

Chủ_tịch UBND Thành_phố Hà_Nội Nguyễn_Thế_Thảo đã đi ...

President of Hanoi's People Committee Nguyen The Thao went ...

Trường Đại_học Bách_khoa HN dẫn_đầu phong_trào ...

HN University of Science and Technology leads the movement ...

Tất_cả chúng_ta đang chuẩn_bị đón_tiếp tân Thủ_tướng .

All of us are preparing to welcome the new prime minister .

Họ đi một vòng quanh thành_phố .

They go a round the city .

Họ đã vượt qua bao sông_suối , thác_ghềnh để đến đây .

They crossed many rivers and streams , waterfalls to come here .

Họ đã có cơm_ăn_áo_mặc , không phải đi_mưa_về_nắng nữa .

They have food and clothing , do not have to work hard anymore .

Từ_láy : bùm_bụp , cuống_cuồng_cuồng , khúc_kha_khúc_khích .

Reduplicative words : boom boom , panic-stricken , giggling .

V. Conclusion

On the basis of studying the characteristics of reduplicative words that linguists have discovered, we have developed a computational method to identify them. The precision of identifying reduplicative words reached 91.02% (Table I). Our study also got a dictionary containing 4239 reduplicative words with 1024 new words.

This study showed that the exploitation of specific structures of Vietnamese words improved the accuracy of Vietnamese word segmentation: F1-scores increased from 0.02% to 0.04%, and the proportion of errors reduced from 1.26% to 1.47%. In the future, we intend to identify Vietnamese subordinated compounds and then use new compounds for the VWS task.

Acknowledgements

This paper has been supported by the national project number KC.01.20/11-15. We would like to express thanks to Dr. Nguyen Thi Trung Thanh (Institute of Linguistics), who helped us check the list of 2-syllable reduplicative words, the list of coordinated compounds, and corrected many segmentation errors of reduplicative words and coordinated compounds in the VietTreeBank corpus.

References

[1] Diep Q. B., Hoang V. T. 2006. Vietnamese Grammar (in Vietnamese), Volumn1,2 . Education Press, Ha Noi.

[2] Dinh D., Hoang K., Nguyen V. T. 2001. Vietnamese Word Segmentation. The sixth Natural Language Processing Pacific Rim Symposium, Tokyo, Japan, 11/2001. pp. 749-756.

[3] Dinh D., Vu T. 2006. A Maximum Entropy Approach for Vietnamese Word Segmentation. Proc. of the 4th IEEE International Conference on Computer Science - Research, Innovation and Vision of the Future 2006, HCM City, Vietnam, pp. 247-252.

[4] Do H.C. 2004. TextBook of Vietnamese Lexicology (in Vietnamese), Education University Press, Hanoi.

[5] Hoang C.D. V., Nguyen L. N., Dinh D., Nguyen Q. H. 2006. Applying Maximum Matching Algorithm and SVM in Vietnamese Word Segmentation (in Vietnamese). Proc. of NCICT2006 (@'06)

[6] Hoang V. H. 2008. Reduplicative words in Vietnamese, Institute of Linguistics, Vietnam Academy of Sosial Sciences (in Vietnamese). Sosial Sciences Press, Hanoi, Vietnam.

[7] Jan Daciuk, Stoyan Mihov, Bruce W. Watson, Richard E. Watson. 2000. Incremental Construction of Minimal Acycle Finite-State Automata.

[8] Le A. H. 2003. A Method for Word Segmentation in Vietnamese. Proceedings of the Corpus Linguistics 2003 Conference, pp. 282-287.

[9] Le H. P., Nguyen T. M. H, Azim R. 2009. Finite-State Description of Vietnamese Reduplication. Proceedings of the 7th Workshop on Asian Language Resources, ACL-IJCNLP 2009, pages 63–69, Suntec, Singapore, 6-7 August 2009. (c) 2009 ACL and AFNLP.

[10] Le H. P., Nguyen T. M. H., Azim R., Hoang T. V. 2008. A Hybrid Approach to Word Segmentation of Vietnamese Texts. Proc. of the 2nd International Conference on Language and Automata Theory and Applications, Springer LNCS 5196, Tarragona, Spain.

[11] Le T. H., Le A. V., Le T. K. 2010. An Unsupervised Learning and Statistical Approach for Vietnamese Word Recognition and Segmentation. Proc. of ACIIDS 2010. pp.195-204.

[12] Ly T. T. 2008. Theory of Word Order in Syntax (in Vietnamese). Vietnam National University Press, Hanoi.

[13] Nguyen C. T., Nguyen T. K., Phan X. H., Nguyen L. M., Ha Q. T. 2006. Vietnamese Word Segmentation with CRFs and SVMs An Investigation. Proceedings of th 20th PACLIC, Wuhan, China, pp.215-222.

[14] Nguyen P. T., Vu X. L., Nguyen T. M. H, Le H. P., Dao M. T., Nguyen T. M. N., Le K. N, Nguyen M. V. 2009. Report of SP7.3 – VietTreeBank. Project of KC01.01/06-10. Vietnam.

[15] Nguyen T. M. H., Vu X. L., Le H. P. 2003. Word segmenatation by dictionary, and POS tagging by probability. Proc. of ICT.RDA, 2003.

[16] Nguyen T. M. H., Vu X. L., Le H. P. 2009. Guidelines for Identification of Lexical Unit in the Vietnamese Language. Report of SP8.2, Project of KC01.01/06-10.

[17] Pham D. D., Tran B. G., Pham B. S. 2007. A Hybrid Approach to Vietnamese Word Segmentation using Part of Speech tags. The 1st International Conference on Knowledge and Systems Engineering (KSE2009), pp.154-161.

[18] Tran N. A., Dao T. T., Nguyen P. T. 2011. An effective method of reducing ambiguity in contextual of problem Vietnamese word segmentation (in Vietnamese). Journal of Science and Technique. Military University of Science and Technology .Vol. 145, pp.50-62.

[19] Tran N. A., Dao T. T., Nguyen P. T. 2012. An effective context-based method for Vietnamese-word segmentation. The First International Workshop on Vietnamese Language and Speech Processing (VLSP2012). In conjunction with 9th IEEE-RIVF Conference on Computing and Communication Technologies. pp.34-40.

[20] Tran N. A., Dao T. T., Nguyen P. T. 2013. Identifying Coordianted Compound Words for Vietnamese Word Segmentation. Proceedings of the fifth international conference of Soft Computing and Pattern Recognition (SoCPaR2013).

[21] Tran T.O., Le A.C., Ha Q. T. 2010. Improving Vietnamese Word Segmentation and POS Tagging using MEM with Various Kinds of Resources., Journal of NLP 17(3):41-60.

[22] VCL (Vietnamese Computational Lexicon). 2010. Branch Themes Word Processing. Lexcicon Project of KC01.01/06-10. 2010.