# An Unsupervised Learning and Statistical Approach for Vietnamese Word Recognition and Segmentation

Hieu Le Trung[1] Vu Le Anh[2] and Kien Le Trung[3]

[1] St. Petersburg State University, Saint Petersburg, Russia
[2] Hoa Sen University, 8. Nguyen Van Trang, Q1, Ho Chi Minh City, Vietnam
[3] Institue of Mathematics, Arndt University, Germany

**Abstract.** There are two main topics in this paper: (i) Vietnamese words are recognized and sentences are segmented into words by using probabilistic models; (ii) the optimum probabilistic model is constructed by an unsupervised learning processing. For each probabilistic model, new words are recognized and their syllables are linked together. The syllable-linking process improves the accuracy of statistical functions which improves contrarily the new words recognition. Hence, the probabilistic model will converge to the optimum one.
Our experimented corpus is generated from about 250.000 online news articles, which consist of about 19.000.000 sentences. The accuracy of the segmented algorithm is over 90%. Our Vietnamese word and phrase dictionary contains more than 150.000 elements.

## 1 Introduction

Word recognition and segmentation of a given sentence into words are important steps in many applications of natural language processing such as text mining, text searching and document classification. These problems are not difficult in Occidental languages since words are determined by space characters. In some Oriental languages such as Vietnamese, Chinese and Japanese, they become much more difficult. Word, *a meaningful linguistic unit*, can be one syllable, or a combination of two or more syllables. Vietnamese word recognition and segmentation problems *can not be solved completely* due to the following two reasons:

*There does not exist an algorithm that segments a given Vietnamese sentence into words exactly according to its meaning if the sentence is considered isolated.* Let us consider the following sentence: *"Cái bàn là của tôi"*. This sentence has two quite different meanings depending on the different word segmentations: (i) *"It is my iron"* for the word segmentation *"Cái | bàn là | của | tôi"*, and (ii) *"The table is mine"* for the word segmentation *"Cái | bàn | là | của | tôi"*. Clearly, no word segmentation algorithm works on this input sentence. The explanation is that each syllable can be a component of different words. Moreover, a Vietnamese sentence is written as a sequence of syllables, not a sequence of words, and its meaning can not be determined without the context.

*There is no official definition of word and complete dictionary in Vietnamese.* Nowadays, Vietnamese linguists still discuss and do not agree with each other about *"What is the word definition in Vietnamese language?"* [4, 2]. For examples, *"máy tính xách tay" (laptop), "máy bay lên thẳng" (helicopter), "xe gắn máy" (motorcycle), etc.* have no final official definition that they are single words or combinations of two words. Moreover, most of new words in Vietnamese online documents, which are from foreign languages (*"avatar", "sms", ...*) or commonly used by teenagers (*"mún", "xì tin", "chảnh",...*) are not in any Vietnamese dictionaries. According to [5], the biggest Vietnamese dictionaries contain less than 33.000 words while *The Second Edition of the Oxford English Dictionary* contains over 250.000 words. Furthermore, as we know, there is no complete Vietnamese dictionary of proper names and names of places and organizations.

Our work intends to address and solve two problems: (i) Recognizing words under probability viewpoint; (ii) Constructing the optimum probabilistic model of huge corpus using an unsupervised learning process.

Our approach for the first problem is as follows. We obverse the corpus, which is a huge set of syllable sequences, and decide *which pair of syllables,* $(\alpha, \beta)$*, is probably a word or an infix of word.* $(\alpha, \beta)$ is chosen if it is *confident* and *supported*. The support $S$ is defined as the number of occurrence of event, $E$, in which $\alpha\beta$ is infix of some sentence. $\mathcal{H}$ is the hypothesis in which $(\alpha, \beta)$ is neither a word nor an infix of word. We use a probabilistic model with assumption $\mathcal{H}$, and estimate $S$ by $S'$. $\mathcal{H}$ is probably wrong if there is a big difference between $S'$ and $S$. The confidence of $(\alpha, \beta)$ (*about* $\mathcal{H}$ *is wrong*) is proportional to the popularity of event $E$ $(S)$ and the ratio, $\frac{S}{S'}$ . Obviously, if $(\alpha, \beta)$ is supported and confident enough, $(\alpha, \beta)$ is probably a word or an infix of word.

The optimum probabilistic model is constructed by an unsupervised learning processing. The initial corpus is a huge set of sentences generated from online documents on the internet. For each learning iteration, we shall do the following steps: (i) Finding only the *local maximum confident* sequences of syllables in sentences; (ii) Linking the local maximum confident sequences of syllables together to be new syllables; (iii) Recomputing all probability values of new corpus and return to step (i). Basing on confident functions, we build a recognition function in which each pair of syllables can be determined whether they are infix of some words or undecidable. Local maximum confident sequences of syllables, which are determined by the recognition function and by comparing neighbor pairs of syllables, are probably infix of some words. Let us see following example:

Sentence $\mathcal{S}$ is *"Công việc của chúng tôi đã thành công"*. The considered pairs of syllables of the first iteration are *"công việc", "việc của", "của chúng", "chúng tôi", "tôi đã", "đã thành", "thành công"*. Recognition function shows that *"chúng tôi"* and *"thành công"* are infix of some words. Moreover, the confident of their neighbors is quite lower than theirs. $\mathcal{S}$ is rewritten as *"Công việc của chúng_ tôi đã thành_ công"* with two new syllables *"chúng_ tôi"* and *"thành_ công"*. The considered pairs of syllables of the second iteration are *"công việc", "việc của", "của chúng_ tôi", "chúng_ tôi đã", "đã thành_ công"*. Suppose *"công việc"* is the local maximum confident pair. $\mathcal{S}$ is rewritten as *"Công_ việc của chúng_ tôi đã*

*thành_ công"*.

By replacing local maximum confident sequence of syllables by the new syllables, the confusion of syllables and words is reduced and the statistical functions are more precise. Contrarily, precise probability values will improve finding the local maximum confident sequences. Therefore, the quality of the probabilistic model is improved by each iteration.

*Our contribution.* We introduce and study a new algorithm for recognizing the new Vietnamese words in huge corpus based on statistics. The unsupervised learning process for building an optimum probabilistic model and corpus is also introduced and discussed. With the experimental corpus generated from over 250.034 online news, a new Vietnamese dictionary and the optimum corpus will be introduced and used in public.

Section 1 is the introduction. Section 2 is the related works. The probabilistic model, confident functions and basic concepts are studied in section 3. The learning process is discussed in section 4. Section 5 is the experiments. Section 6 concludes the paper.

## 2 Related Works

As we known, we are the first group, who study the Vietnamese word recognition based on statistical methods. Vietnamese word segmentation has be studied by several groups [3, 7, 8, 6, 9]. There are two main approach: *manual corpus based approach* [3, 7, 8] and *unsupervised statistical approach* [6, 9].

The solutions of the first group are built around the theory of supervised learning machines. Dinh [3] is based on the WFST model and Neural Network. Nguyen [8] is based on CRF (conditional random fields) and SVM (support vector machines). Le [7] is based on hybrid algorithms with *maximal-matching method* concept. Their learning machines learn from manual dictionaries or manual annotated corpora, which are limited by human resource. [3] used 34.000-word dictionary, [8] used about 1.400 annotated news, [7] ignored the new words. They claimed the accuracy of their methods are over 90% but only for very small manual annotated corpora.

In second approach, Ha [6] applied the maximum probability of tri-gram in a given chunk of syllables over huge corpus. Thanh [9] used Mutual Information (MI) formulas for n-gram combined with Genetic Algorithm. Their works and ours have three big different points: (i) They did not have learning process to improve the accuracy of statistical information (ii) In our work, the relationship of syllables in same word are generalized by confidence concept using different probability formulas, not only MI formula [6] or maximum probability of n-gram [9] (iii) Their corpus is quite smaller than us. Therefore, the accuracies of their algorithms are 50% [6] and 80% [9], which are lower than us (90%). Moreover, our approach can apply for proper names, names of places or organizations and phrases recognition.

## 3 Probabilistic Model

### 3.1 Basic Concepts

*Syllable* is an original syllables (such as *"của", "đã"*) or a linking syllables (such as *"công_ việc", "chúng_ tôi", "thành_ công"*). Given a syllable $\beta$, we denote $\alpha \in Pre(\beta)$ ($\alpha \in Suf(\beta)$) if $\alpha$ is a prefix (suffix) of $\beta$. For example, *"công"* $\in Pre($*công_ việc*$)$, *"việc"* $\in Suf($*công_ việc*$)$, and *"của"* $\in Pre($*của*$) \cap Suf($*của*$)$. *Sentence* is a sequence of syllables. *"Công_ việc của chúng_ tôi đã thành_ công"* sentence is denoted by $S = \alpha_1\alpha_2 \ldots \alpha_5$ in which $\alpha_1$=*"công_ việc"*, $\alpha_2$=*"của"*, ..., and $\alpha_5$=*"thành_ công"*. $\beta_1\beta_2 \ldots \beta_l$ is *an infix of sentence* $S = \alpha_1\alpha_2 \ldots \alpha_k$ $(1 \leq l \leq k)$ if $\exists 1 \leq i \leq k - l + 1 : \beta_j = \alpha_{i+j-1} \forall j = 1, \ldots, l$.

A probabilistic model $\mathcal{P}$ is defined as a triple $(\mathcal{C}, \Sigma_\mathcal{C}, \mathcal{F}_\mathcal{C})$.

*Corpus*, $\mathcal{C}=\{s_1, s_2, \ldots, s_n\}$, is a finite set of sentences. $\Sigma_\mathcal{C}$ is the set of syllables, which are infix of some sentence $s_i$ of $\mathcal{C}$. $\mathcal{F}_\mathcal{C}$ is *the set of statistical functions*. A probabilistic function $f_C \in \mathcal{F}_\mathcal{C}$ is a map from ${\Sigma_\mathcal{C}}^*$ to $\mathcal{R}$. It can be a constant $(\emptyset \mapsto \mathcal{R})$, a function of syllable $(\Sigma_\mathcal{C} \mapsto \mathcal{R})$ or a function of pair of syllables $(\Sigma_C^2 \mapsto \mathcal{R})$ and so on. Here are basic statistical functions used in our work:

Suppose $\alpha, \beta \in \Sigma_\mathcal{C}$. $N(\alpha)$ is denoted for the number of occurrence of $\alpha$ in $\mathcal{C}$. We define: $N_p(\alpha) = \sum_{\beta:\alpha\in Pre(\beta)} N(\beta)$; $N_s(\alpha) = \sum_{\beta:\alpha\in Suf(\beta)} N(\beta)$; $N_1 = \sum_{\alpha\in\Sigma_C} N(\alpha)$. The probabilities of the events that $\alpha$ occurs in $\mathcal{C}$ as independent syllable or prefix, suffix of some syllable are estimated respectively as follows: $P(\alpha) = \frac{N(\alpha)}{N_1}$; $P_p(\alpha) = \frac{N_p(\alpha)}{N_1}$ and $P_s(\alpha) = \frac{N_s(\alpha)}{N_1}$. Similarly, $N(\alpha\beta)$ is denoted for the number of occurrence of $\alpha\beta$ in some sentence of $\mathcal{C}$, $N_2 = \sum_{\alpha,\beta\in\Sigma_C} N(\alpha\beta)$. The probability of event $\alpha\beta$ occurs in $\mathcal{C}$ is estimated by $P(\alpha\beta) = \frac{N(\alpha\beta)}{N_2}$.

### 3.2 Confident Functions and Word Recognition

The *optimum corpus* is the one in which each sentence is segmented into sequence of words exactly according to its meaning. Each syllable in optimum sentence is a word. We have shown in the introduction section that there is no algorithm to construct the optimum if each sentence is considered isolated. However, words are recognized with the help of confident functions.

*Confident functions* are statistical functions of pair of syllables, which measure how probably the given ordered pair of syllables is an infix of word. Suppose $\mathcal{H}$ is the hypothesis that $\alpha\beta$ is not infix of any word. Each confident function $f_{C,M}(\alpha, \beta)$ is based on a probabilistic model, $\mathcal{M}$, in which: (i) $\mathcal{H}$ is assumed to be true (ii) the probability of event, $E$, $\alpha\beta$ occurs in $\mathcal{C}$, is estimated as $P'(\alpha\beta)$. $f_{C,M}(\alpha, \beta)$ is proportional with the popularity of $E$, and the ratio $\frac{P(\alpha\beta)}{P'(\alpha\beta)}$.

**Definition 1.** Suppose $c \in \mathcal{R}$ is a constant. The confident function $f_{C,M}(\alpha, \beta) : {\Sigma_\mathcal{C}}^2 \mapsto \mathcal{R}$ over probabilistic model , $\mathcal{M}$, and corpus, $\mathcal{C}$, is defined as:

$$f_{C,M}(\alpha, \beta) = c * \frac{P(\alpha\beta)^2}{P'(\alpha\beta)}$$

We choose randomly two neighbor syllables $x_1x_2$ in some sentence, and $A$ is the event $x_1 = \alpha$ is suffix of some words in the optimum sentence; $B$ is the event in which $x_2 = \beta$ is prefix of some word in the optimum sentence. $\mathcal{H}$ implies that for each occurrence $\alpha\beta$ in $\mathcal{C}$: (i) $\alpha$ must be suffix of some words and (ii) $\beta$ must be prefix of some words. Hence, $P'(\alpha\beta) = P(A \cap B) \equiv P(AB)$. Here are different models for $P(AB)$ estimation:

*Model 1*: Assumption: $A$, $B$ are independent events, $P(A) = c_1P(\alpha)$, $P(B) = c_2P(\beta)$ ($c_1$, $c_2$ const.), and $P(AB) = c_1c_2P(\alpha)P(\beta)$. Hence for $c = \frac{1}{c_1c_2}$:

$$f_{C,1}(\alpha, \beta) = \frac{P(\alpha\beta)^2}{P(\alpha)P(\beta)}$$

In reality, $P(\alpha)P(\beta)$ is much more smaller than $P(AB)$ since in natural language $\alpha$, $\beta$ never stand by each other purely random. We suggest $P(AB) = P(\alpha)^{\varphi_s(\alpha)}P(\beta)^{\varphi_p(\beta)}$. The experiments shows that $\varphi_s(\alpha) = 1 - \frac{H(X_{s,\alpha})}{\log N_s(\alpha)}$ ($\varphi_p(\beta) = 1 - \frac{H(Y_{p,\beta})}{\log N_p(\beta)}$) is the good estimation in which $X_{s,\alpha}$ ($Y_{p,\beta}$) is a variable presenting syllables appear before (after) the syllable $\alpha$ ($\beta$), and $H(\cdot)$ is an *entropy operator*. Because of the limit of this paper, we will study the construction and the properties of $\varphi_s, \varphi_p$ functions in another work.

*Model 2*: Assumption: $P(AB) = c_1P(\alpha)^{\varphi_s(\alpha)}P(\beta)^{\varphi_p(\beta)}$ ($c_1$ const.). Let's $c = \frac{1}{c_1}$:

$$f_{C,2}(\alpha, \beta) = \frac{P(\alpha\beta)^2}{P(\alpha)^{\varphi_s(\alpha)}P(\beta)^{\varphi_p(\beta)}}$$

*Model 3*: Assumption: $A$, $B$ are independent events. Obviously, $P(A) \simeq P_s(\alpha)$ and $P(B) \simeq P_p(\beta)$. $P(AB)$ is estimated by $P_s(\alpha)P_p(\beta)$. Here, we take $c = 1$:

$$f_{C,3}(\alpha, \beta) = \frac{P(\alpha\beta)^2}{P_s(\alpha)P_p(\beta)}$$

Connector words (such as *"và" (and), "thì" (then), "là" (is), "của" (of)*, etc.) are important factors in Vietnamese. The occurrences of these words are very high comparing to normal ones. There is a famous assumption [1] about Vietnamese word recognition which says that $\alpha\beta$ is a word in given sentence if and only if we can not place any connector word between them that not change the meaning of the sentence. Suppose $W$ is the set of connector words. $N_W(\alpha\beta)$ is denoted for the number of occurrence of event, $E$, $\delta\gamma\eta$ is an infix of some sentence of $\mathcal{C}$ in which $\gamma \in W$, $\alpha \in Suf(\delta)$ and $\beta \in Pre(\eta)$. $N_3 = \sum_{\alpha,\beta\in\Sigma_C} N_W(\alpha\beta)$. The probability of event $E$ is estimated by $P_W(\alpha\beta) = \frac{N_W(\alpha\beta)}{N_3}$. The number of occurrence of event $AB$ is proportional to the number of occurrent $E$.

*Model 4*: Assumption: $P(AB) = c_1P_W(\alpha\beta)$. Hence for $c = \frac{1}{c_1}$:

$$f_{C,4}(\alpha, \beta) = \frac{P(\alpha\beta)^2}{P_W(\alpha\beta)}$$

.

The extent version of the confident function $f_{C,M}$, $f_C^* : {\Sigma_C}^+ \mapsto \mathcal{R}$, is defined as follows: $f_C^*(w) = P(w)$ if $w \in \Sigma_C$. Otherwise, $f_C^*(w) = \frac{P^2(w)}{P^*(w)}$ in which $P^*(w) = Max_{w=uv}P'(u,v)$ where $u$, $v$ are prefix, suffix of $w$ and $w = uv$. $P'(u,v)$ is the estimated probability of the event that $w$ is not word and is segmented into $u|v$ using $M$ and necessary statistical values.

$\mathcal{P} = (\mathcal{C}, \Sigma_\mathcal{C}, \mathcal{F}_\mathcal{C})$ is a probabilistic model. $m_{sup}, M_{sup}, m_{con}, M_{con} \in \mathcal{F}_\mathcal{C}$ are constant functions, in which $0 < m_{sup} \leq M_{sup}$ and $0 < m_{con} \leq M_{con}$. $f_C \in \mathcal{F}_\mathcal{C}$ is a confident function. Word recognition function is defined as follows:

**Definition 2.** $f_R : \Sigma_C^2 \mapsto \{-1, 0, 1\}$ is *the word recognition function* of $f_C$ over $\mathcal{P}$ with parameters $(m_{con}, M_{con}, m_{sup}, M_{sup})$ in which:

$$f_R(\alpha,\beta) = \begin{cases} 1 & \text{if } (f_C(\alpha,\beta) \geq M_{con}) \wedge (N(\alpha\beta) \geq M_{sup}); \\ -1 & \text{if } (f_C(\alpha,\beta) < m_{con}) \vee (N(\alpha\beta) < m_{sup}); \\ -0 & \text{otherwise.} \end{cases}$$

If return value of recognition word function is 1, the input pair of syllables are supported and confident and it is probably infix of some word. If return value is -1, the input pair of syllables belongs two different words. We have no decision in the case the return value is 0. Obviously, if $m_{sup} = M_{sup}$ and $m_{con} = M_{con}$, the return values can not be 0 and there does not exist undecidable case. In the case, we have different confident functions and different recognition word functions. We can combine them by some fuzzy rules to be only one *universal word recognition function*, $f_R^*$.

## 4 Learning Process and Main Results

### 4.1 Learning rules and Learning Process

Suppose $\mathcal{P} = (\mathcal{C}, \Sigma_\mathcal{C}, \mathcal{F}_\mathcal{C})$ is a probabilistic model; $f_C, f_R^* \in \mathcal{F}_\mathcal{C}$ is a confident function and the universal word recognition function respectively; $D_{con} \in \mathcal{F}_\mathcal{C}$ is a positive constant; $s = \alpha_1\alpha_2\ldots\alpha_k \in \mathcal{C}$ is a sentence in corpus, and $w = \alpha_l\alpha_{l+1}\ldots\alpha_{l+m}$ is an infix of $s$ $(1 \leq l < k, 0 < m \leq k - l)$.

**Definition 3.** $w$ is a *local maximum confident sequence* (LMC for short) of $s$ over $\mathcal{P}$ with $f_C, f_R^*$ and $D_{con}$, if it satisfies following conditions:
(i) $\forall i = l, \ldots, m-1 : f_R^*(\alpha_i, \alpha_{i+1}) = 1$;
(ii) if $l > 1$ :$f_R^*(\alpha_{l-1}, \alpha_l) = -1$ or
$f_R^*(\alpha_{l-1}, \alpha_l) = 0 \wedge f_C(\alpha_l, \alpha_{l+1}) > f_C(\alpha_{l-1}, \alpha_l) + D_{con}$
(iii) if $l + m < k : f_R^*(\alpha_{l+m}, \alpha_{l+m+1}) = -1$ or
$f_R^*(\alpha_{l+m}, \alpha_{l+m+1}) = 0 \wedge f_C(\alpha_{l+m-1}, \alpha_{l+m}) > f_C(\alpha_{l+m}, \alpha_{l+m+1}) + D_{con}$

Condition (i) guarantees that all pairs of neighbor syllables of $w$ are infixes of some words. In condition (ii) and (iii), the neighbors of $w$, $(\alpha_{l-1}, \alpha_l)(l > 1)$ and $(\alpha_{l+m}, \alpha_{l+m+1})$ $(l + m < k)$, are considered. They guarantee that the neighbors do not effect to $w$ under confident viewpoint. Therefore, $w$ is a sequence of words.

Suppose $w = \beta_1\beta_2 \ldots \beta_l$ occurs $T(w)$ times as a LMC in some sentence. Here are learning rules sorted by the priority.

*Rule 0. If $Link(w) \in \Sigma_{\mathcal{C}}$: Replace $w$ by $Link(w)$.* $Link(w) \in \Sigma_{\mathcal{C}}$ implies that in the past we have learned that $w$ is infix of some word.

*Rule 1. If $m = 1$: Replace $w$ by $Link(w)$.* $w$ is a two syllables word.

*Rule 2. Sorting the neighbor pairs of $w$ by the confident value. If $(\beta_i, \beta_{i+1})$ is the first one and the difference of the confident values of first- and second pairs more than $D_{con}$: Replace $\beta_i\beta_{i+1}$ by $Link(\beta_i\beta_{i+1})$.* The difference of the confident values guarantees $\beta_i\beta_{i+1}$ does not belong two different words probably.

*Rule 3. If $m = 3, 4$ and $T_w \geq M_{sup}$ and $f^*_C(w) \geq M_{con}$: Replace $\omega$ by $Link(\omega)$.*

Rule 0 is always considered as it is the nature of learning. Rule 1 and 2 find pairs of syllables which are infix of some word. The priority of Rule 1 is higher than Rule 2 since Rule 1 helps us founding out the "two-syllables" words. Rule 3 have the lowest priority since it needs many statistical values and computing resources. The case in which $m > 4$ are ignored since there does not exist 5-syllable word in Vietnamese. Learning process for Rule 0, 1 and 2 is as follows:

```
1.  Learning-Process-1
2.  repeat
3.    repeat
4.      for each sentence s
5.        for each w is LMC of s
6.          if w satisfies Rule 0 or 1 then
7.              s is rewritten by replacing
                w with Link(w)
8.              Update, create new necessary
                statistical values
9.      until No linking pair is found
10.     for each sentence s
11.       for each w is LMC of s
12.         if infix αβ of w satisfies Rule2 then
13.             s is rewritten by replacing αβ
                with Link(αβ)
14.             Update, create new necessary
                statistical values
15.   until No linking pair of Rule 2 is found
```

*Learning Strategy.* The system of parameters (*SP for short*), $(m_{con}, M_{con}, m_{sup}, \ldots)$, decides the quality of learning process. For example, as higher as $M_{con}$, the number of new words is smaller but the quality of learning is higher. Our proposed strategy of learning process is *"Slowly but Surely"*. Suppose $(m^*_{con}, M^*_{con}, m^*_{sup}, \ldots)$ is the desired SP for the optimum corpus. We increase the quality of learning process by generating sequences of SPs: $(m_{1,con}, M_{1,con}, m_{1,sup}, \ldots)$, ..., $(m_{n,con}, M_{n,con}, m_{n,sup}, \ldots) = (m^*_{n,con}, M^*_{n,con}, m^*_{n,sup}, \ldots)$ in which the quality of learning of system $i$-th is higher than system $(i+1)$-th: $m_{i+1,con} \leq m_{i,con}$, $M_{i+1,con} \leq M_{i,con}$, ... Here is the proposed learning process for Rule 0, 1, 2 and 3 and desired SP $(m^*_{n,con}, M^*_{n,con}, m^*_{n,sup}, \ldots)$:

```
1.  Learning-Process
2.      Computing necessary statistical values
        of syllable and pairs of syllables
3.  for i = 1 to n do
4.      Setting the parameters to
        (m_{i,con}, M_{i,con}, m_{i,sup}, ...)
5.      Learning-Process-1
6.  for each sentence s
7.      for each local maximum confident
        sequence w of s
8.          if w satisfies Rule3 then
9.              s is rewritten by replacing w
                with Link(w)
```

The **Learning-Process-1** loops $n$-times (line 3-5) with $n$ systems of parameters, which converge to the desired one. The way we choose these systems

guarantees the quality of learning process. Rule 3 is considered (line 6-9) which guarantees all 3-4 syllables words are recognized.

### 4.2 Optimum Segmentation Algorithm, Dictionary

Suppose $\mathcal{P}^* = (\mathcal{C}^*, \Sigma^*_{\mathcal{C}}, \mathcal{F}^*_{\mathcal{C}})$ is the probabilistic model which is the result of the learning process, $M^*_{sup} \in \mathcal{F}^*_{\mathcal{C}}$ is the minimum support of words. The dictionary of words, $\mathcal{D}$, generated from $\mathcal{P}^*$ is defined as: $\mathcal{D} = \{w \in \Sigma^*_C | N_1(w) \geq M^*_{sup}\}$. Experiments show that if desired parameters is chosen exactly, we can extract not only the dictionary of Vietnamese words but also phrases, names of organizations and so on.

Different word segmentation algorithms introduced in another works [3, 7, 8] can use the optimum corpus as the annotated corpus. Learning process itself is an word segmentation algorithm. The input sentence is segmented by the learning algorithm introduced in previous subsection. It uses the statistical values which are produced by the optimum probabilistic model. Experiments shows that the accuracy our algorithm is about 90%.

## 5 Experiments

*Corpus.* Our corpus is generated from 250.034 articles in the *Tuoi Tre* (The Youth) online newspaper. After applying the data normalization (fix the code font and repair spelling mistake of syllables) and sentences segmentation (using punctuation mark, comma, question mark, semicolon, etc.), the initial corpus has 18.676.877 sentences whose total length is 131.318.974 syllables.

*Model.* Because of the limit of this paper, we present only the experiments using Model 2 with confident function $f_{C,2}$ in which the formula of SP is $(m_{con}, M_{con}, m_{sup}, M_{sup}, D_{con})$. The results of other models, the comparison and combination of different models will be represented in the extent version of this paper.

*Algorithm.* Our learning process applied the strategy *"Slowly but Surely"*. There were 9 learning iterations with 9 different SPs. In 1-4 iterations, $M_{con}$ and $M_{sup}$ are very high since we want recognize the most common two-syllable

| $\log(SP)$ | No. Linking | 1-Syl. | 2-Syl. | 3-Syl. | 4-Syl. | ($>4$)-Syl. | Sum |
|---|---|---|---|---|---|---|---|
| (-5.0,5.0,10,200,2.0) | 9.331.392 | 11.107 | 1.990 | 0 | 0 | 0 | **13.097** |
| (-4.5,3.0,10,100,1.5) | 21.150.384 | 10.415 | 7.556 | 9 | 0 | 0 | **17.980** |
| (-4.5,2.0,10,80,1.5) | 27.545.021 | 10.019 | 14.761 | 108 | 0 | 0 | **24.888** |
| (-4.0,1.0,10,50,1.0) | 32.273.932 | 9.594 | 30.479 | 582 | 6 | 0 | **40.661** |
| (-4.0,0.0,10,40,1.0) | 36.547.210 | 9.191 | 42.531 | 2.159 | 80 | 25 | **53.986** |
| (-3.5,-1.0,10,30,0.5) | 39.023.942 | 8.598 | 51.681 | 6.187 | 587 | 123 | **67.176** |
| (-3.5,-1.5,10,20,0.5) | 40.246.564 | 8.413 | 66.394 | 9.985 | 1069 | 261 | **86.122** |
| (-3.0,-2.0,10,20,0.0) | 41.763.245 | 8.180 | 85.500 | 12.985 | 1.947 | 514 | **109.126** |
| (-3.0,-3.0,10,20,0.0) | 45.516.469 | 7.676 | 106.696 | 32.573 | 5.835 | 1.788 | **154.568** |

**Table 1.** System of parameters and number of recognized words in learning iterations. $\log(SP)$ is denoted of $(\log m_{con}, \log M_{con}, \log m_{sup}, \log M_{sup}, \log D_{con})$.

| Syllables | St. 1 | St. 2 | St. 3 | St. 4 |
|---|---|---|---|---|
| cá nhân | 3.58 | reg. | reg. | reg. |
| cá cược | 3.60 | reg. | reg. | reg. |
| cá độ | 1.86 | 3.03 | reg. | reg. |
| cá heo | 0.51 | 1.31 | 2.01 | reg. |

| Syllables | St. 1 | St. 2 | St. 3 | St. 4 |
|---|---|---|---|---|
| cá cảnh | -2.55 | -2.01 | -0.66 | 1.02 |
| cá biển | -2.85 | -2.60 | -0.8 | 1.1 |
| cá lóc | 0.88 | 1.5 | 2.51 | reg. |
| cá bỏ | -7.86 | -7.49 | -6.83 | -6.05 |

**Table 2.** The confident function's values of some pairs of syllables, of which *"cá"* is the first syllable. *"reg."* is the short of *recognized word*.

words, whose average occurrence is about 30% in Vietnamese sentences. Hence linking their syllables will improve effectively the quality of statistical values. In 5-7 iterations, $M_{con}$ is decreased slowly so most of words are recognized in this time. In 8-9 iterations, all parameters are change slowly so that all proper nouns, phrases, etc are recognized. The results of the iterations are shown in Table 1. Our dictionary contains about 60.000 words, over 30.000 phrases, and 20.000 proper names, names of place, foreign words.

*Accuracy.* The accuracy of word recognition and segmentation algorithms is measured by choosing randomly some recognized words or segmented document and counting the mistakes. After checking many times and computing the average of mistake, the accuracy of our dictionary and segmented algorithm is 95% and 90% respectively. They are depended strongly on the confident function. Table 2 shows the values of the confident function of 8 pair of syllables, which are words except *"cá bỏ"* for 1-4 iterations. The confident values are increased by each iteration, and these words are recognized step by step except *"cá bỏ"*.

*Scalar.* According to the number of words and phrases, our dictionary is the biggest public Vietnamese dictionary. Table 3 shows the list of words, proper names, names of places, phrases which are in our dictionary and rarely found in another ones. However, our dictionary is incomplete. All missing words are rarely used in modern Vietnamese or in the professional language for newspapers. Our corpus is expanded easily with our web crawler. Currently, we have downloaded about 9.000 Vietnamese online-books at the Web http://vnthuquan.net and the news of the most popular Vietnamese online newspapers. The next version of

| Phrases | Place names | Place names | Proper names | New words |
|---|---|---|---|---|
| ách tắc giao thông | Bắc Triều Tiên | Quận Ninh Kiều | Alfred Riedl | acid béo |
| áp thấp nhiệt đới | Bờ Biển Ngà | Quận Đống Đa | Alex Ferguson | mobile phone |
| bất phân thắng bại | Bồ Đào Nha | Quận Ba Đình | Barack Obama | bản photocopy |
| bật đèn xanh | Ch Czech | Quận Bình Tân | Bảo Đại | bánh heroin |
| bất khả xâm phạm | Ch Ireland | Quận Bình Thạnh | Nguyễn Minh Triết | băng cassette |
| bặt vô âm tính | Trung Quốc | Quận Bình Thủy | Nguyễn Tấn Dũng | bánh pizza |
| cân đo đong đếm | Triều Tiên | Quận Gò Vấp | Phạm Ngọc Thạch | máy in Laser |
| lở mồm long móng | Nhật Bản | Quận Hai Bà Trưng | David Beckham | quả penalty |
| càng sớm càng tốt | Thủy Điển | Quận Hải Châu | Leonardo Da Vinci | nhạc rock |
| chất độc màu da cam | Ấn Độ | Quận Hoàn Kiếm | Công nương Diana | Windows Vista |

**Table 3.** Some special phrases, names of place, proper names, and new words in our dictionary.

our dictionary is more complete and the optimum corpus is more precise, too.

## 6 Conclusion

We have proposed a Vietnamese word recognizing algorithm based on statistic. The algorithm works well on different corpora and can extract the name of persons, places or organizations. The experiments show that the complete Vietnamese dictionary can be built with this approach.

We have also studied an unsupervised learning iterations to construct the optimum probabilistic model and perfect word segmentation algorithm. There are two main factors effect to the learning: (i) Linking the local maximum confident pairs of syllables in sentences together as new syllables; (ii) probability values of the corpus. Two factors effect to each others, and both are improved by learning process. The output of the learning iterations is the optimum probabilistic model. The experiments show that by using the optimum probabilistic model generated from our corpus, the accuracy of our word segmentation is over 90%.

The Vietnamese language is not explained and described well by grammar rules. One of our research direction is: Finding the most common formulas of Vietnamese sentences based on statistic. We believe that with computer and huge corpus, we can solve many problems of Vietnamese language processing based on statistic.

## References


1. Cao, X. H.: Vietnamese - Some Questions on Phonetics, Syntax and Semantics. Nxb Giao duc, Hanoi. (2000)
2. Chu, M. N., Nghieu, V. Đ, Phien, H. T. : Cơ sở ngôn ngữ học và tiếng Việt. Nxb Giáo dục, Hanoi, 1997, 142–152.
3. Dien, D., Kiem, H., Toan, N. V.: Vietnamese Word Segmentation. The Sixth Natural Language Processing Pacific Rim Symposium, Tokyo, Japan. (2001) 749–756
4. Giap, N. T.: Từ vựng học tiếng Việt. H., Nxb Giao duc, 2003.
5. Thu, C. B., Hien, P.:Về một xu hướng mới của từ điển giải thích (2007) http://ngonngu.net/index.php?p=319
6. Ha, L. A.: A method for word segmentation in Vietnamese. Proceedings of Corpus Linguistics 2003, Lancaster, UK. (2003)
7. Le, H. P., Nguyen, T. M. H., Roussanaly, A., Ho, T. V.: A hybrid approach to word segmentation of Vietnamese texts. In 2nd International Conference on Language and Automata Theory and Applications, Tarragona, Spain. (2008)
8. Nguyen, C. T., Nguyen, T. K., Phan, X. H., Nguyen, L. M., Ha, Q. T.: Vietnamese word segmentationwith CRFs and SVMs: An investigation. In Proceedings of the 20th Pacific Asia Conference on Language, Information and Computation (PACLIC 2006),Wuhan, CH. (2006)
9. Nguyen, T.V., Tran, H.K., Nguyen, T.T.T., Nguyen, H.: Word segmentation for Vietnamese text categorization: an online corpus approach. Research, Innovation and Vision for the Future, The 4th International Conference on Computer Sciences (2006)