Improving Vietnamese POS tagging by integrating a rich feature set and Support Vector Machines

Minh NGHIEM - Dien DINH
Faculty of Information Technology
University of Sciences
Ho Chi Minh, Vietnam
nqminh@fit.hcmuns.edu.vn – ddien@fit.hcmuns.edu.vn

Mai NGUYEN
Faculty of Computer Science
University of Information Technology
Ho Chi Minh, Vietnam
maintn@uit.edu.vn

***Abstract*—Part of Speech (POS) tagging is fundamental in natural language processing. So far, many methods have been applied for English and the task is well solved. However, there are few studies about this problem for Vietnamese. In this paper, we evaluate common features for English POS tagging and then propose some language specific features for Vietnamese POS tagging. Experimental results on the Vietnamese Lexicography Center's research group's corpus show that our POS tagger using this feature set trained by SVM outperforms other Vietnamese POS taggers.**

Natural Language Processing; Part of Speech Tagging; Support Vector Machines

I. INTRODUCTION

Part-of-speech (POS) tagging is fundamental in natural language processing (NLP). It is the process of marking up the words in a text as corresponding to a particular part of speech, based on both its definition, as well as its context of appearance. The POS of a word provides a significant amount of information about that word and its neighboring words, which is useful for other problems in NLP such as phrase chunking, parsing, and word-sense disambiguation.

Many methods have been applied for POS tagging based on statistical and machine learning techniques, such as the Hidden Markov Model (HMM) (Charniak et al., 1993), the Neural Networks (Schmid, 1994), the Decision Trees (Schmid, 1994), the Transformation-based Learning (Brill, 1995), the Maximum Entropy Model (Ratnaparkhi, 1996), the Support Vector Machines (SVM) (Nakagawa et al., 2001). Performances of those methods are remarkably high, evaluated on the English Wall Street Journal Corpus, using the Penn Treebank POS tag-sets. Though these methods have good performance, most studies are focused on English. So far, only one POS tagger for Vietnamese documents was made public is the HMM-based VNQTAG (Huyen Nguyen T. M et al., 2003).

Because Vietnamese language has specific characteristics, applying other taggers will lead to limited performance. Vietnamese is not a "monosyllabic" language. Vietnamese words may consist of one or more syllables. There is a tendency for words have two syllables (disyllabic) with perhaps 80% of the lexicon being disyllabic. Some words have three or four syllables- many polysyllabic words are formed by reduplicative derivation. Additionally, a Vietnamese word may consist of a single morpheme or more than one morpheme.

For example: "*com"* (cooked rice) is a mono-morphemic; "*dưa chuột"* (cucumber) is a bi-morphemic; "*vội vội vàng vàng"* (hurry-scurry) is a poly-morphemic, it is also a kind of reduplicative.

Moreover, there is a phenomenon in Vietnamese language called the "POS changing". For example: "*hạnh phúc*" (happy) is an adjective, but when it is preceded by the word "*niềm*" (sense/ feeling), its POS is noun. These problems make Vietnamese POS tagging much more difficult than other languages such as English.

In this paper, we propose a robust method for POS tagging on Vietnamese documents by using a wide variety of features, including language specific features. Our approach use SVM, one of the state of the art machine learning methods to perform tagging. The tagger we introduce in this work fulfills the requirements for being a practical tagger; experimental results on the Vietnamese Lexicography Center's research group's corpus prove that this tagger achieves high accuracy and outperforms other Vietnamese POS taggers.

The remainder of this paper is organized as follows: In section 2, we present the framework overview. We then describe the features used for POS tagger and analyze which one is good for Vietnamese language in section 3. In section 4, we describe the results of our experiments. Section 5 concludes the paper and gives avenues for future works.

II. FRAMEWORK OVERVIEW

Figure 1 shows the training and figure 2 shows the tagging process of the system.

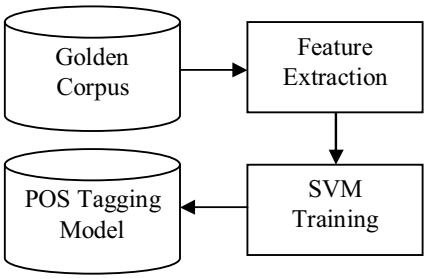

Figure 1. Training process

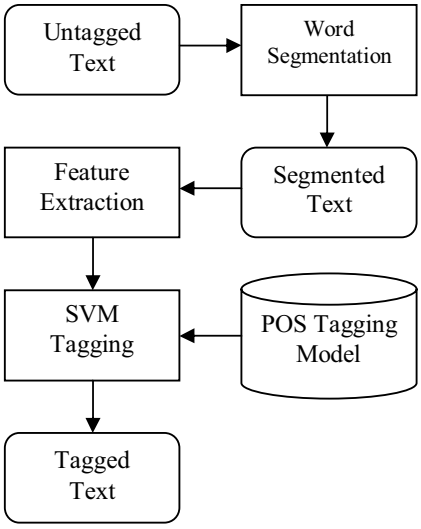

Figure 2. Tagging process

A. Word segmentation

Before POS tagging, Vietnamese language needs a step of preprocessing called word segmentation. Unlike English, Vietnamese word boundary cannot be identified by spaces. Vietnamese writing is monosyllabic in nature. So we have to do word segmentation before tagging.

In a sample Vietnamese sentence: "*Tốc độ truyền thông tin sẽ tăng cao.*" (The speed of information transmission will increase.), there are 8 syllables. These 8 syllables have their own meanings but in this sentence, some of them are only morphemes. We have many ways of segmenting words in this sentence. However, only one way is reasonable in term of semantic and grammar: "*Tốc_độ truyền thông_tin sẽ tăng cao.*" (The speed of information transmission will increase.). One case of segmentation is "*Tốc_độ truyền_thông tin sẽ tăng cao.*" (The speed of communicate news will increase.), this segmentation way is correct in grammar but not reasonable in semantic. This problem is the same as other language such as Chinese and Japanese.

B. Feature extraction

This step transforms the input data into a set of features for the system. In machine learning methods, selecting features is one of the main steps. The details of the selected features are described in section 3.

C. Training and Tagging using Support Vector Machines

After word segmentation and feature extraction, we use SVM for training and tagging.

SVM is a machine learning algorithm for binary classification which is currently considered as one of the most efficient methods in many real world applications. The theory of SVM has been developed in 60s and 70s by Vapnik and Chervonenkis, but the first practical implementation of SVM was published in the early 90s. Since then, this method is more and more popular because it outperforms most learning algorithms. SVM had been successfully applied to a number of practical problems, including NLP (Cristianini, 2000).

Given a training dataset contains n examples $\{(x_1, y_1),(x_2, y_2),...,(x_n, y_n)\}$, where each instance x_i is a vector in $\Re^L$ and $y_i \in \{-1,+1\}$ is a class label which the example x_i belongs. Suppose the hyper-plane $w.x + b = 0 \quad w \in \Re^L, b \in \Re$ separates the training data into two classes: positive examples and negative examples. While several of such separating hyper-planes exist, SVM finds the optimal hyper-plane that maximizes the margin (the distance between the hyper-plane and the nearest examples).

The classifying rule of SVM is $y = \text{sgn}(f(x))$, where $f(x) = w.x + b = 0$. The instance x will be classified to the positive class if $f(x) \geq 0$ and the negative class if $f(x) < 0$. The values of the weight vector w and the bias b is calculated by solving a quadratic optimization problem.

For linearly non-separable cases, feature vectors are mapped into a higher dimensional space by a nonlinear function $\Phi(x)$ and linearly separated there.

Since SVMs are binary classifiers, we must extend them to multi-class classifiers in order to classify more classes. Among several methods of multi-class classification for SVMs (Weston and Watkins, 1999), we use the one-versus-one approach. The tagger is implemented using the Yamcha package.

III. FEATURES

The features we used can be divided into 2 classes: common features and language specific features.

Common features are language-independent features that can be used in any language. Language specific features are features that can be used only in Vietnamese language. The common features used are similar to those used in SVMTool (Gimenez et al., 2003). We modified some features such as: while Gimenez uses tokens from w_{-3} to w_3 (from three tokens before to three tokens after), we used only the tokens w_{-1}, w_0 and w_1; the ambiguous features are extracted from the hand-made dictionary; orthographic features is merged into one feature.

Feature selection is implemented using a feature cutoff: features seen less than a small count during training will not be used.

A. Common features

- Lexicon Feature: The simplest and most obvious feature set is the string of the current word. This group contains a large number of features (one for each token string present in the training data).
- Word context: The string of the word preceding the current word and the string of the word succeeding the current word.
- POS context: The part of speech tags of words preceding the current word (these POS tags are guessed by the system).
- Ambiguous feature: Due to the limited amount of training material, tag dictionary have been found to be useful in the POS tagging task. Tag dictionary provides the lists of POS tags for words. This dictionary was also used by Ratnaparkhi to reduce the number of possible POS tags (Ratnaparkhi, 1996). For words that are neither in the dictionary nor in the training data, all possible POS tags are taken as candidates. The tag dictionary can be extracted from the training data. However, such dictionary was provided by the Vietnamese Lexicography Center's research group so we integrated this dictionary to our system. In next section, we show that this is one of the most important features for POS tagging.
- Orthographic feature: Word characteristics are covered by the orthographic features. This feature regards to: how is the word capitalized (initial capitalized, internal capitalized or fully capitalized); the kind of characters that form the word (contains digits, contains symbols, all digits); the presence of punctuation marks (contains dots, contains hyphen).
- One other orthographic feature is “orthographic form” similar to Collins (2002), the system replaces capital letters with ‘A’, lowercase letters with ‘a’, digits with ‘0’, and all other characters with ‘_’, then collapses consecutive identical characters into one.

B. Specific features

Besides common features, we proposed 2 more language specific features: reduplication and affixes.

1) Reduplication:

Reduplication, in linguistics, is a morphological process of creating a new word by repeating either a whole word or part of a word (vowel or syllable). It is often used when a speaker adopts a tone more "expressive" or figurative than ordinary speech and is also often, but not exclusively, iconic in meaning. Reduplication is found in a wide range of languages and language groups, though its level of linguistic productivity varies: Indo-European, Chinese, Japanese, Persian, Khmer, Vietnamese, etc.

In Vietnamese, this called “*từ láy*”. It is used when one want to increase or decrease the intensity of the adjective and is often used as a literary device (like alliteration) in poetry and other compositions, as well as in everyday speech. It makes the sentences become more likely, present the meaning of the writer more exactly. For example: In the sentence: "*Gió thổi nhè nhẹ*" (The wind is blowing gently). Instead of using "*nhẹ*", we use "*nhè nhẹ*" to present the gentle sensation more exactly.

Reduplicative word has the same part-of-speech as the word which forms it. For example:

- *nhẹ* → *nhè nhẹ*: soft → soft (less): adjective
- *xinh* → *xinh xinh*: pretty → cute: adjective
- *đỏ* → *đo đỏ*: red → somewhat red: adjective
- *mạnh* → *mạnh mẽ*: strong → very strong: adjective
- *người* → *người người*: people → everyone: noun
- *cười nói* → *cười cười nói nói*: talk and laugh → keep talking and laughing: verb

Vietnamese language has many reduplicative words; every word can form a reduplicative word by following some rules. It is obvious that we cannot save all these words in the dictionary. To overcome this problem, we propose a reduplication feature. If the word is a reduplicative word, then this feature is set to 1. Moreover, the list of possible POS tags of ambiguous feature is also reduced to possible POS tags of the root word.

2) Suffixes and Prefixes:

Vietnamese also has suffixes and prefixes. Many affixes come from the Sino-Vietnamese. For example:

- Prefix *bán-* (half): *nguyệt* (moon) → *bán nguyệt* (semicircular, semi-monthly); *đảo* (island) → *bán đảo* (peninsula).
- Prefix *phi-* (not): *nghĩa* (righteousness) → *phi nghĩa* (unethical); *chính phủ* (government) → *phi chính phủ* (non-governmental).
- Suffix *–gia* (profession): *chính trị* (politics) → *chính trị gia* (politician); *khoa học* (science) → *khoa học gia* (scientist).
- Suffix *–học* (field of study): *ngôn ngữ* (language) → *ngôn ngữ học* (linguistics); *động vật* (animal) → *động vật học* (zoology).

Words with same affixes tend to have same POS. This feature is extracted by taking a syllable at the beginning of the words as prefix and taking a syllable at the end of the words as suffix.

Figure 3 shows a sample of a sentence with extracted features.

	WORD	AMBIGOUS TAGS	ORTHO -GRAPHIC	ORTHO -GRAPHIC FORM	PREFIX	POSTFIX	REDUPLI -CATION	TAG
POS: -4	Bình thường	J	FirstCap	A_a	bình	thường	No	J
POS: -3	không	R-Q-J-N	Letters	a	không	không	No	R
POS: -2	người	N	Letters	a	người	người	No	N
POS: -1	đàn ông	N	Letters	a_a	đàn	ông	No	N
POS: 0	nào	P-M-U-R	Letters	a	nào	nào	No	P
POS: +1	gọi	V	Letters	a	gọi	gọi	No	V
POS: +2	vợ	N	Letters	a	vợ	vợ	No	N
POS: +3	như	C-R	Letters	a	như	như	No	C
POS: +4	thế	P-M-N	Letters	a	thế	thế	No	P
POS: +5	.	.	Punctuation	_	.	.	No	.

Figure 3. A sample of a sentence with extracted feature

IV. EXPERIMENTAL RESULTS

A. Data and Evaluation

The experiments were carried out using the datasets provided by The Vietnamese Lexicography Center's research group. These datasets contain a lexical dictionary and 7 documents belong to a number of different genres [1]. These 7 documents are tagged manually. Table 1 contains the various corpus statistics.

TABLE I. THE CORPUS STATISTICS

Document	*Number of words*	*Genre*
Chuyện tình kể trước lúc rạng đông 1 (Love story tell before dawn 1)	16787	Vietnamese novel
Chuyện tình kể trước lúc rạng đông 2 (Love story tell before dawn 2)	14698	Vietnamese novel
Hoàng tử bé (Little Prince)	18663	Foreign story
Lược sử thời gian (Brief history of time)	11626	Science book
Muối của rừng (Forest's Salt)	3537	Vietnamese tale
Những bài học nông thôn (Rural lessons)	8244	Vietnamese tale
Công nghệ (Technology)	1162	Newspaper and magazine
Total	74753	

The lexical dictionary has 37454 words; each word accompanied with its list of POS. An example of the dictionary is show in Figure 4.

Training and testing were performed using 5-fold cross-validation; the original corpus is partitioned into 5 subsets. Of the 5 subsets, a single subset is retained as the validation data for testing the model, and the remaining subsets are used as training data. The cross-validation process is then repeated 5 times, with each of the 5 subsets used exactly once as the validation data. The 5 results from the folds then are averaged to produce a single estimation.

We used 2 tag-sets in this paper: the first tag-set has 10 tags; the second tag-set has 48 tags. These tag-sets are the same as [1].

B. Word Context

From the hypothesis that better use of context will improve the accuracy, we set up an experiment to find out which context is best suitable for Vietnamese language. In this experiment, we only use the word context surrounding the current word. Results are displayed in Table 2.

TABLE II. RESULTS OF DIFFERENCE WORD CONTEXT

Word Context	Results	
	48 tags	*10 tags*
3 word before	80.79	89.39
2 word before	81.97	90.06
1 word before	82.93	90.54
1 word after	82.76	91.07
2 word after	81.72	90.15
3 word after	80.67	89.72
1 word before and 1 word after	83.29	92.71
2 word before and 2 word after	81.11	89.61
3 word before and 3 word after	78.61	88.07

On this dataset, we found that if we take one word before and one word after the current word, the result is higher than taking more preceding and succeeding words.

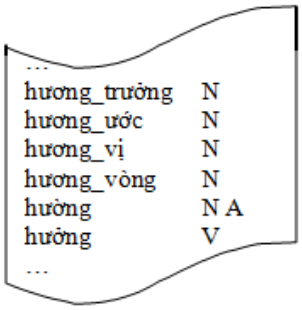

Figure 4. An example of the dictionary

C. Evaluating the SVM method

All statistical POS tagger must use a machine learning method to build a model and perform tagging. VNQTAG uses HMM method. The current tag t_0 is predicted based on the current word and 2 previous words. It uses another kind of information: a word dictionary, each word accompanied with its list of POS. This tagger only uses a few features so the accuracy is still limited.

In order to compare with this method, we setup an experiment with same features using different methods. The features we used are the same as the features used in VNQTAG [1]: word tri-gram and dictionary. Table 3 shows the results of these methods.

TABLE III. RESULTS OF DIFFERENCE METHODS

Word Context	Results	
	48 tags	*10 tags*
HMM	85.6	91.54
SVM	87.9	93.51

With the same features, SVM system is better than HMM, but the difference is not much, only 1% in accuracy.

D. Contribution of each feature

In this section, we report experiments using combination sets of features to evaluate the important of each feature for Vietnamese POS tagging. Table 4 shows the results of the system with other features. Features are added to the system seriatim.

TABLE IV. RESULTS OF DIFFERENCE FEATURES

Word Context	Results	
	48 tags	*10 tags*
1 word before and 1 word after	83.29	92.71
+ dictionary	87.9	93.51
+ orthographic, orthographic forms	87.95	93.79
+ reduplication, affixes	88.19	94.67
+ POS context	88.41	94.89

As we can see in Table 4, the performance of the system increases when adding features. The dictionary feature is the most important feature, it increase the performance of the system up to 4%. The language specific features (reduplication, affixes) increase the performance of the system up to 1%.

E. The importance of the dictionary

As we can see in the previous section, the list of POS tags for known words (Ambiguity classes) is one of important features for POS tagging. But building such a tag dictionary is a time consuming task. So we set up an experiment without using the given dictionary to evaluate the important of this information.

The first model uses the dictionary which was built by hand. The second model uses a tag dictionary which was extracted from the training data. This tag dictionary contains words and its tags appeared in the training data. When extracting features, tags of known words that appeared in training data are taken using this tag dictionary. For unknown words, all possible POS tags are taken as the candidates. The last model does not make use of the dictionary. The result is shown in Table 5.

TABLE V. THE IMPORTANT OF THE DICTIONARY

Word Context	Results	
	48 tags	*10 tags*
The dictionary is built by hand	88.41	94.89
The dictionary is extracted from training data	84.48	93.5
Without the dictionary	83.29	92.71

As we can see in Table 5, by using the hand built dictionary, the performance of the system increase by 2-5%.

F. Influence of Word Segmentation

To test the influence of the word segmentation on the process of POS tagging, we used a word segment tool, based on [8]. The test data was word segmented by this word segment tool before running the POS tagger.

TABLE VI. THE IMPORTANT OF WORD SEGMENTATION

Word Context	Results	
	48 tags	*10 tags*
Word segmentation is done manually	88.41	94.89
Word segment by using tool	84.76	90.8

The results obtained are displayed in Table 6. As expected, the automatic segmentation leads to lower results than the original segmentation. Although the word segmentation tool got a high result on word segmentation in our corpus (96%) on one hand, but on the other hand, it makes the total result much

lower. These examples demonstrate some errors while performing word segmentation:

- *"học (study) sinh (biology)" → "học_sinh (pupil)"*
- *"lao_động (works) từ (from)" → "lao (javelin) động_từ (verb)"*

We found out the reason was about the word context. When a word is not segmented correctly, it impact not only the current word but also neighboring words.

V. CONCLUTION

In this paper, we applied SVM to Vietnamese POS tagging using a rich feature sets and showed that our tagger it perform quite well. The resulting tagger achieves higher accuracy than other tagger on Vietnamese. We also analyze the important of each feature in POS tagging, the role of tag dictionary and the influence of word segmentation in POS tagging.

There are such various POS taggers on English documents have been published in the world today. The accuracy is very high. But when these methods are applied to Vietnamese POS tagging, the results are not high as English. This difference is caused by the structures and the specific characteristics of each language. The other cause is that lengths of training data are unequal. The corpus we use to train is rather small compare to other corpus in English or Chinese, it does not embrace all the ambiguity cases. Therefore, the results we get are still limited. To get a highest accuracy, it needs a process of long research and working. However, the first satisfactory results make us try to improve the quality of the Vietnamese POS tagger. We always hope that this POS tagger will be a practical and useful tool with NLP researches.

REFERENCES

[1] Huyen Nguyen T. M., Luong Vu X., Phuong Le H., "A case study of the probabilistic tagger QTAG for Tagging Vietnamese Texts", In Proceedings of ICT.rda'03. Hanoi Feb. 22-23, 2003.

[2] A. Ratnaparkhi, "A Maximum Entropy Model for Part-of-Speech Tagging", In Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP-1), pages 133–142, 1996.

[3] N. Cristianini and J. Shawe-Taylor, "An Introduction to Support Vector Machines and other kernel-based learning methods", Cambridge University Press, 2000.

[4] T. Nakagawa and T. Kudoh and Y. Matsumoto, "Unknown Word Guessing and Part-of-Speech Tagging Using Support Vector Machines", In Proceedings of the Sixth Natural Language Processing Pacific Rim Symposium, 2001.

[5] J. Weston and C. Watkins, "Support Vector Machines for Multi-Class Pattern Recognition", In Proceedings of the Seventh European Symposium on Artificial Neural Networks (ESANN-99), 1999.

[6] J. Gimenez and L. Marquez, "Fast and accurate part-of-speech tagging: The SVM approach revisited", In Proceedings of RANLP-2003, Borovets, Bulgaria, 2003.

[7] J. Gimenez and L. Marquez, "SVMTool: A general POS tagger generator based on support vector machines", In Proceedings of LREC '04, pp. 43--46, 2004.

[8] Dinh Dien and Vu Thuy, "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, 2006.