A Fuzzy Synset-Based Hidden Markov Model for Automatic Text Segmentation

Viet Ha-Thuc[1], Quang-Anh Nguyen-Van[1], Tru Hoang Cao[1] and Jonathan Lawry[2]

[1] Faculty of Information Technology, Ho Chi Minh City University of Technology, Vietnam
viettifosi@yahoo.com
nvqanh2003@yahoo.com
tru@dit.hcmut.edu.vn

[2] Department of Engineering Mathematics, University of Bristol, UK
j.lawry@bristol.ac.uk

Summary. Automatic segmentation of text strings, in particular entity names, into structured records is often needed for efficient information retrieval, analysis, mining, and integration. Hidden Markov Model (HMM) has been shown as the state of the art for this task. However, previous work did not take into account the synonymy of words and their abbreviations, or possibility of their misspelling. In this paper, we propose a fuzzy synset-based HMM for text segmentation, based on a semantic relation and an edit distance between words. The model is also to deal with texts written in a language like Vietnamese, where a meaningful word can be composed of more than one syllable. Experiments on Vietnamese company names are presented to demonstrate the performance of the model.

1 Introduction

Informally speaking, text segmentation is to partition an unstructured string into a number of continuous sub-strings, and label each of those sub-strings by a unique attribute of a given schema. For example, a postal address consists of several segments, such as house number, street name, city name, zip code, and country name. Other examples include paper references and company names. As such, automatic segmentation of a text is often needed for further processing of the text on the basis of its content, namely, information retrieval, analysis, mining, or integration, for instance.

The difficulty of text segmentation is due to the fact that a text string does not have a fixed structure, where segments may change their positions or be missing in the string. Moreover, one word, in particular an acronym, may have different meanings and can be assigned to different attributes. For example, in a paper reference, its year of publication may be put after the author names or at the end, and the publisher name may be omitted. For dealing with that uncertainty, a probabilistic model like HMM has been shown to be effective, for general text ([5]) as well as specific-meaning phrases like postal addresses or bibliography records ([1], [3]).

V. Ha-Thuc et al.: *A Fuzzy Synset-Based Hidden Markov Model for Automatic Text Segmentation*, Advances in Soft Computing **6**, 365–372 (2006)
www.springerlink.com

However, firstly, the above-mentioned HMMs did not consider the synonymous words in counting their common occurrence probabilities. That affects not only the performance of text segmentation, but also information retrieval later on. Secondly, the previous work did not tolerate word misspelling, which is often a case. The segmentation performance would be better if a misspelled word could be treated as its correct one, rather than an unknown word. Besides, the segmentation task is more difficult with a language like Vietnamese, where a meaningful word can be composed of more than one syllable. For example, "*công ty*" in Vietnamese means "*company*".

In this paper, we propose an HMM that overcomes those limitations. Firstly, words having the same meaning are grouped into one synonym set (synset), and the emission probability for each state is distributed over those synsets instead of individual words. Secondly, the probability is fuzzified by using a string matching distance measure such as edit distance to deal with the word misspelling noise. Thirdly, the standard Viterbi algorithm is extended to group syllables into words for Vietnamese or an alike language.

The paper is organized as follows. Section 2 summarizes the basic notions of HMM and its application to text segmentation. Section 3 present our proposed fuzzy synset-based HMM and its extension for multi-syllable words. Experimental results are presented in Section 4. Finally, Section 5 concludes the paper with some remarks and suggestion for future work.

2 HMMs for Text Segmentation

2.1 Hidden Markov Models

An HMM is a probabilistic finite state automaton ([8]), consisting of the following parameters:

- A set of one start state, one end state, and n immediate states
- An $n \times n$ transition matrix, where the ij^{th} element is the probability of making a transition from state i to state j.
- A vocabulary set V_s for each immediate state s, containing those words that can be emitted from s.
- An emission probability p distributed over V_s for each immediate state s, where $p(w|s)$ measures the probability for s emitting word w in V_s.

These four parameters are learned from data in the training phase. Then in the testing phase, given a text, the most probable path of states, from the start state to the end, can be computed, where each state emits and corresponds to a word of the text in that sequence.

2.2 Learning Parameters

Learning the HMM parameters requires only a single pass over the training data set ([3]). Each training instance is a sequence of state-word pairs. The learned set of immediate states simply comprises all states appearing in the training data. The vocabulary set of each state can also be learned easily as the set of all words paired with that state in the training data.

Let N_{ij} be the number of transitions made from state i to state j, and N_i be the total number of transitions made from state i, according to the training data. The transition probability from state i to state j is learned as follows:

$$a_{ij} = N_{ij}/N_i$$

For the emission probability distribution of state s, suppose that the vocabulary set $V_s = \{w_1, w_2 \ldots w_M\}$ and the raw frequency, i.e., number of occurrence times, of each w_i in state s in the training data is f_i. Then, the probability that s emits w_i is computed as below:

$$p(w_i|s) = f_i/\sum\nolimits_{j=1,M} f_j$$

The above formula would assign probability of zero to those words that do not appear in training data, causing the overall probability for a text string to be zero. So the model would not be applicable to a text string containing one or more unknown words. To avoid this, a non-zero probability is assigned to an unknown word with respect to a state, and the emission probability distribution of that state is adjusted accordingly. Such a smoothing technique is the Laplace one, as follows:

$$p(\text{``}unknown\text{''}\,|\,s) = 1\,/\,(\sum\nolimits_{j=1,M} f_i + M + 1)$$

$$p(w_i\,|\,s) = (f_i + 1)/(\sum\nolimits_{j=1,M} f_i + M + 1)$$

One can see that $\sum_{j=1,n} a_{ij}$= 1 and $\sum_{i=1,M} p(w_i|s) + p$ (“unknown word” $|s) = 1$, satisfying the normalized conditions.

2.3 Text Segmentation

For text segmentation, given an input string $u = w_1, w_2 ... w_m$ and an HMM having n immediate states, the most probable state sequence, from the start state to the end state, that generates u can be obtained by the Viterbi algorithm as follows ([8]). Let 0 and $(n+1)$ denote the start and end states, and $Pr_s(i)$ be the probability of the most probable path for $w_1, w_2 ... w_i$ $(i \leq m)$ ending at state s (i.e., s emits w_i). As such, $Pr_0(0) = 1$ and $Pr_j(0) = 0$ if $j \neq 0$.
Then Pr$_s$(i) can be recursively defined as follows:

$$Pr_s(i) = \text{Max}_{t=1,n}\{Pr_t(i-1) \times a_{ts}\} \times p(w_i\,|\,s)$$

where a_{ts} is the transition probability from state t to state s, and the maximum is taken over all immediate states of the HMM. The probability of the most probable path that generates u is given by:

$$Pr(u) = \text{Max}_{t=1,n}\{Pr_t(m) \times a_{t(n+1)}\}$$

This probability function can be computed using dynamic programming in O(mn^2) time.

3 Fuzzy Synset-Based HMMs

3.1 A Case Study: Vietnamese Company Names

For testing the performance of our proposed model as presented in the following sections, we have chosen the domain of Vietnamese company names. Figure 1 shows the HMM learned from our training data, where each immediate state corresponds to a field that a company name may contain:

- *Kind of Company* such as “*công ty*” (company), “*nhà máy*” (factory), ...
- *Kind of Possession* such as “*TNHH*” (Ltd.), “*cổ ph`ân*” (stock), “*tu nh`ân*” (private), “*liên doanh*” (joint-venture), ...
- *Business Aspect* such as “*xăng d`âu*” (petroleum), “*du lịch*” (tourism), “*yt*” (medical)...
- *Proper Name* such as “*Sài Gòn*”, “*Microsoft*”, “*Motorola*”, ...

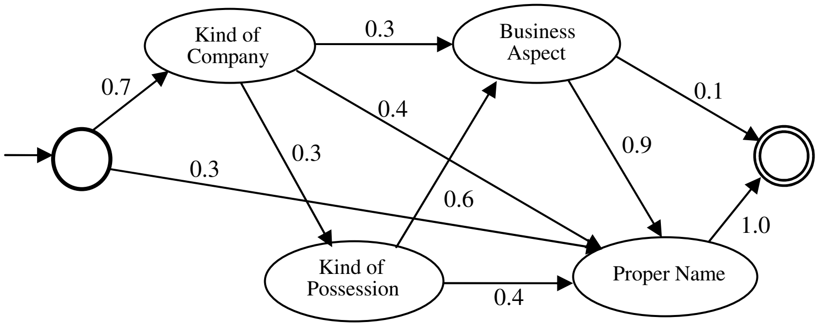

Fig. 1. An HMM for Vietnamese Company Names

3.2 Synset-Based HMMs

As mentioned above, synonymous words should be treated as the same in semantic matching as well as in counting their occurrences. For example, in Vietnamese, “*trách nhiêm hũu hạn*”, “*TN hũu hạn*”, and “*TNHH*” are full and acronyms of the same word meaning “*Ltd.*”. So we propose synset-based HMMs in which words having the same meaning are grouped into a synset. Each training instance is a sequence of state-word-synset triples created manually. The probability of a synset emitted by a state is defined as the sum of the probabilities of all words in that synset emitted by the state, as exemplified in Table 1. Then, the model would operate on a given text as if each word in the text were replaced by its corresponding synset.

Since one ambiguous word may belong to different synsets, the one with the highest emission probability will be chosen for a particular state. For example, “*TN*”

Table 1. Emission Probabilities in a Synset-Based HMM

Word w	$p(w\|$ state = $Kind\ of\ Possession)$
trách nhiệm hũu hạn	0.05
TNHH	0.25
TN hũu hạn	0.1
cổ phần	0.15
CP	0.15
tu nhân	0.2
TN	0.1

Synset W	$p(W\|$ state = $Kind\ of\ Possession)$
trách nhiệm hũh hạn	0.4
TNHH	
TN hũu hạn	
cổ phần	0.3
CP	
tu nhân	0.3
TN	

in the two following company names has different meanings, where in the former it is an abbreviation of "*tu nhân*" (private) and in the latter of "*thiên nhiên*" (natural):

Công ty	*TN*	*Duy Lợi*
company	private	proper name

Cty	*nuóc khoáng*	*TN*	*La Vie*
company	mineral water	natural	proper name

Figure 2 illustrates the most probable paths of the two names in the proposed synset-based HMM, found by using the Viterbi algorithm. We note that using synsets also helps to fully match synonymous words emitted from the same state, such as "*Công ty*" and "*Cty*" in this example.

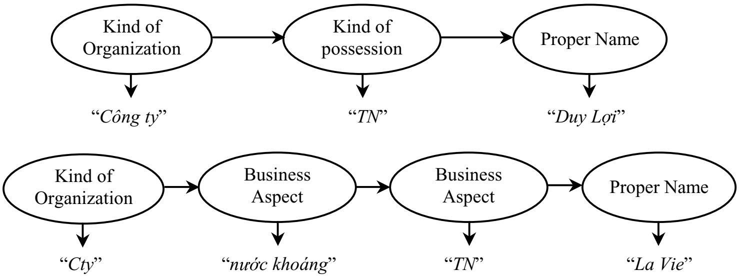

Fig. 2. A Synset-Based HMM Resolves Ambiguous Words

3.3 Fuzzy Extension

The conventional HMM as presented above does not tolerate erroneous data. If a word is misspelled, not in the vocabulary set of a state, it is treated as an unknown word with respect to that state. For instance, "*cômg ty*" and "*Micorsoft*" are misspellings of "*công ty*" and "*Microsoft*", respectively. The idea of our proposed fuzzy HMMs is that, if a word w is not contained in the vocabulary set V_s of a state s, but its smallest edit distance ([2]) to the words in V_s is smaller than a certain threshold, then it is considered as a misspelling. In the synset-based case, the distance of w to a synset in V_s is defined as the minimum of the distances of w to each word in that synset. Therefore, the fuzzy emission probability of w with respect to s is computed as follows:

```
if ($w \in W$ in $V_s$) then
        $fp(w|s) = p(W|s)$
else {
        $W_0 = Argmin_{W \in V_S} distance(w, W)$
                        // $distance(w, W) = Min_{x \in W}\, editDist(w, x)$
            if ($distance(w, W_0) < threshold_s$ ) then
            $fp(w|s) = p(W_0|s)$ // $w$ might be misspelled from $W_0$
            else
            $fp(w|s) = p(\text{“}unknown\text{”}|s)$// $w$ is an unknown word
    }
```

3.4 Extension for Vietnamese

The fact that a Vietnamese word may comprise more than one syllable makes word segmentation and part-of-speech tagging difficult ([6], [7]), as compared to English where words are separated by spaces. For example, the Vietnamese words "*xuȯsng*", "*công ty*", "*tông công ty*" contain one, two and three syllables respectively. Therefore, in order to segment company names, for instance, using the HMM present above, one would have to pre-process it to group syllables into words first.

Here we propose to do both steps in one HMM, by modifying the Viterbi algorithm as follows. Assume that the maximal number of syllables that form a word is K, in particular 4 for Vietnamese. The probability $Pr_s(i)$ of the most probable path for a syllable sequence $e_1 e_2 ... e_i$ ending at state s, among n immediate states of the HMM, is defined by:

$$Pr_s(i) = Max_{j=1,K}\{Max_{t=1,n}\{Pr_t(i-j) \times a_{ts}\} \times p(e_{i-j+1} ... \, e_{i-1} e_i | \; s)\}$$

That is $j(1 \leq j \leq K)$ syllables may form a word ending at state s, which maximizes $Pr_s(i)$. The time complexity of the algorithm is O(Kmn^2), for a syllable sequence of length m.

4 Experimental Results

The accuracy of a name segmentation method is defined to be the percentage of names correctly segmented in a testing set. We have evaluated the proposed synset-based HMM over a set of company names randomly extracted from several Vietnamese websites. Figure 3 shows that its accuracy is about 5% higher than the

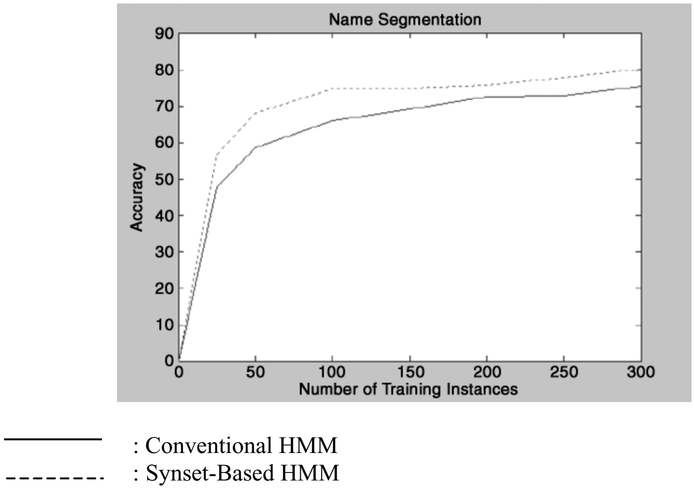

Fig. 3. Comparison between Conventional and Synset-Based HMMs

conventional HMM, being over 80% with 300 training instances. In a domain where the vocabulary sets contain many synonymous words, the improvement could be higher.

To obtain noisy data sets, we use a tool that randomly generates misspelled words from their original correct ones. The noise level of a data set is defined as the ratio of the number of erroneous characters per the total number of characters of a word, for every word in the set. Figure 4 compares the performances of a synset-based HMM and its fuzzy extension over three data sets with different noise levels.

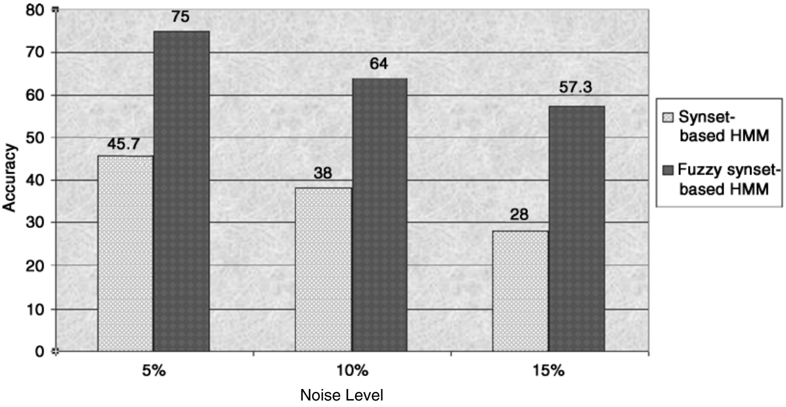

Fig. 4. Comparison between a Synset-Based HMM and its Fuzzy Extension

The experiment results show that a synset-based HMM works quite well on clean data, but its accuracy decreases with noisy data. Introducing fuzzy emission probabilities helps to reduce the misspelling effect.

5 Conclusion

We have presented an enhancement of HMMs for text segmentation whose emission probabilities are defined on synsets rather than individual words. It not only improves the accuracy of name segmentation, but also is useful for further semantic processing such as name matching. For dealing with misspelled words, we have introduced the notion of fuzzy emission probability defined on edit distances between words. Lastly, we have modified the Viterbi algorithm to segment text in Vietnamese and alike languages, where a meaningful word may comprise more than one syllable.

Conducted experiments have shown the advantage of the proposed fuzzy synset-based HMM. Other string distance measures are worth trying in calculating fuzzy emission probabilities. The model is being applied in VN-KIM, a national key project on Vietnamese semantic web, to automatically recognize named-entities in web pages and matching them for knowledge retrieval ([4]). These are among the topics that we suggest for further work.

References

[1] Agichtein E., Ganti V., (2004), Mining Reference Tables for Automatic Text Segmentation, Procs of ACM Conference on Knowledge Discovery and Data Mining (SIGKDD), pp. 20-29.

[2] Bilenko M., Mooney R., Cohen W., Ravikumar P., Fienberg S., (2003), Adaptive Name Matching in Information Integration, IEEE Intelligent Systems, Vol. 18, No. 5, 16-23.

[3] Borkar V., Deshmukh K., Sarawagi S., (2001), Automatic Segmentation of Text into Structured Records, Procs of the ACM SIGMOD Conference.

[4] Cao T.H., Do H.T., Pham B.T.N., Huynh T.N., Vu D.Q, (2005). Conceptual Graphs for Knowledge Querying in VN-KIM, Contributions to the 13th International Conference on Conceptual Structures, pp. 27-40.

[5] Freitag D., McCallum A.K., (2000), Information Extraction with HMM Structure Learned by Stochastic Optimization, Procs of the 18th Conference on Artificial Intelligence, pp. 584-589.

[6] Ha L.A., (2003), A Method for Word Segmentation in Vietnamese, Procs of Corpus Linguistics Conference, pp. 17-22.

[7] Nguyen Q.C., Phan T.T., Cao T.H., (2006), Vietnamese Proper Noun Recognition, Procs of the 4th International Conference on Computer Sciences, pp. 145-152.

[8] Rabiner L.R., (1989), A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, Procs of the IEEE, Vol. 77, No. 2, 257-286.