A Method Integrating Rule and HMM for Chinese Part-of-speech Tagging

Hui Ning, Hua Yang and Zhihui Li
Computer science and technology college of HarBin Engineering University
HarBin Engineering University, Street Nantong 145#,Harbin
150001,China

***Abstract*-In this paper, we study the lexical category disambiguation and the disambiguation strategy using rule techniques and HMM (hidden Markov model) is introduced. With the above method, a system of disambiguation is materialized. The experimental results show that the tagging accuracy is raised by using rule techniques and hidden Markov model. The disambiguation accuracy of close test and open test is 92.97% and 91.21% respectively, and the overall accuracy is 97.84% and 96.71% respectively.**

I. INTRODUCTION

In natural language processing, there are many kinds of ambiguity problems. One of the most important is lexical category ambiguity. Part-of-speech tagging is the act of eliminating syntax ambiguity with appropriate method, base on syntax relations in context. Result of part-of-speech tagging direct affects research in many fields such as parsing, semantic analysis, speech recognition, machine translation, information search, information filtration and so on, therefore it all through attracts people's attention [1]. The difficulty of part-of-speech tagging is lexical category disambiguation. Much research has been done to tag Chinese part-of-speech using several different models and methods, including: statistical model, rule-based method, genetic algorithms and artificial neural network etc.

Rule-based technique is a conventional method, and its advantage is that it can make the best of production of linguistic research. For some special ambiguity combination, this method can get high accuracy of disambiguation by describing characteristic information of words and lexical category deeply and exactly. But, rule can't include all answer of problems of lexical category ambiguity. The dominant method is statistical method today. Its advantage is that it can get good coherence and very high coverage-rate, since the model get its knowledge by learning from corpus. But, the essence of statistical method is choosing part-of-speech tag has the highest probability. That is the maximal likelihood, but it is not exclusive case. Therefore statistical method must limit the part-of-speech tagging accuracy [2]. The preferable method is integrating two methods. This paper study the method integrating rule-based and HMM (hidden Markov model) method.

II. TAGGING METHOD

The tagging process of the method this paper introduces is shown in Figure 1:

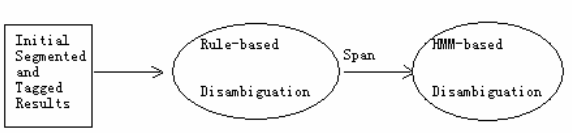

Fig.1. Integrating Rule and HMM POS Tagging Flow.

Step 1: We use Chinese participle tool to syncopate text, and tag the part-of-speech of the words has determinate part-of-speech;

Step 2: We utilize rules to eliminate lexical category ambiguity in primal segmented and tagged text. At this step, we can't eliminate all lexical category ambiguity, and leave un- eliminated ambiguity to step 3;

Step 3: We use HMM to eliminate lexical category ambiguity leaved at step 2.

A. Rule-based Disambiguation

Rule base is the foundation of rule-based disambiguation. It must be constructed firstly for rule-based disambiguation. The transformation-based error-driven learning method proposed by Eric Brill is used to construct rule base. In order to reduce trouble brought by rule collision, rules are divided into two ranks.

a. Expression of Rule

Rule chunk is constructed for each lemma. For example:

lex =广播

1^pos=n; ^1 pos=n; → pos=v;
1^pos=p; ^1 pos=n; → pos=v;
1^pos=p; ^1 pos=b; → pos=v;
1^pos=c; ^1 pos=n; → pos=v;

It denotes that current lemma "广播 " has four rules, and the first rule means that if the POS of the lemma left the current lemma is n, and the POS of the lemma right the current lemma is n, the POS of the current lemma is v.

The Virtual Value of Rule V: Measurement of rule validity in practical tagging.

$V=f_c / f_a$;

Where, f_c is the time of the rule tagging right, f_a is the time of the rule effecting.

Rule Span Len: Test condition of the rule, denotes that Len words left the current lemma and Len words right the current lemma. In example above, Len=1.

1-4244-0737-0/07/$20.00 ©2007 IEEE

b. Rule Learning

First unannotated text is passed through an initial-state annotator. Where, labeling all words as nouns is used as initial-state annotator. Once text has been passed through the initial-state annotator, it is then compared to the truth. A manually annotated corpus is used as our reference for truth. All possible transformations are learned and evaluated. The virtual value of each rule is computed, and the transformations whose virtual values are not less than the minimum virtual value are selected as rule set. Then, the training corpus is updated by applying the learned transformations. Learning continues until no transformation can be found [3].

The learning algorithm is as following;

1) Annotate training corpus by applying initial-state annotator ;
2) Len=1;
3) Find all possible transformations (rules);
4) Compute the virtual value of each rule;
5) Eliminate rules has too low frequency */* Give a minimum frequency f_{min}, to the rule r_i, if the frequency $f_i < f_{min}$, eliminate r_i */*
6) Eliminate rules bring collision;
7) Eliminate rules whose virtual value $Ci<C_m$ */* C_m is a user given threshold of virtual value */*
8) Annotate training corpus by applying learned rule set;
9) Len=Len+1, if Len<MaxLen, turn to step 3) */* MaxLen is the maximal rule span given by user */*

c. Rule Classification

In order to reduce trouble brought by rule collision, rules are divided into two ranks based on character of the rule, when we use rule to eliminate lexical category ambiguity.

Absolutely determinate rules: If the words have lexical category ambiguity accord with these rules, their part-of-speech is concluded based on the rule;

Semi-determinate rules: If the words have lexical category ambiguity accord with these rules, we can't conclude part-of-speech right now. We still consider if the key in current context also accord with absolutely determinate rules or other semi-determinate rules. If not, we conclude the part-of-speech based on this rule. If there is absolutely determinate rule matching, we process as (1). If there is other semi-determinate rule matching, we consider rules conflict, and leave problem to HMM disambiguation step.

B. HMM Disambiguation

We get a series of word clusters whose first and last word has a clear and tagged part-of –speech. The word cluster is called span. HMM disambiguation processes a span once.

a. HMM Definition

HMM is a five elements tuples { S，A，V，B，Π }, where,

S= {S_1，S_2，…S_N} is hidden state set;

V= {V_1，V_2，…… V_M} is observable state set;

Π = {π_1，π_2，--- π_N} is initial state distributing probability, and

$$\pi_i = P(X_1 = S_i) \qquad 1 \le i \le N \tag{1}$$

$A=(a_{ij})_{N\times N}$ is state transfer probability matrix, and

$$a_{ij} = P(X_{t+1} = S_j \mid X_t = S_i) \tag{2}$$

It's the probability of state S_i transferring to state S_j.

$B=(b_j(v_k))_{N\times M}$ is the probability matrix of hidden state releasing observable state,

Where, $b_j(v_k) = P(O_t = V_k \mid X_t = S_j) \quad 1 \le k \le M, 1 \le j \le N$ (3)

It denotes the probability of hidden state S_j releasing observable state V_k.

When we use HMM to tag Part-of-speech, we regard part-of-speech as hidden state and un-tagged span as observable sequence $O_0O_1\cdots\cdots O_n$. Then part-of-speech tagging is finding the most possible hidden state sequence $S_0S_1\cdots\cdots S_n$ as part-of-speech tagging sequence for special span $O_0O_1\cdots\cdots O_n$. In this process, there are two problems to resolve, HMM learning and decoding problem.

b. HMM Learning

Because part-of-speech of first word O_0 of the span is clear, then

$$\pi_i = \begin{cases} 1 & i=j; \\ & 1 \le i \le N; \\ 0 & \text{others}; \end{cases} \tag{4}$$

We can use ML (Maximal Likelihood) method to get matrix A and B from training samples

$$a_{ij} = C_{ij} / \sum_{k=1}^{N} Ci,k \qquad 1 \le i, j \le N \tag{5}$$

Where, C_{ij} is the degree of part-of-speech S_i transferring to S_j in all training sample

$$b_j(v_k) = E_j(v_k) / \sum_{l=1}^{M} E_j(v_l) \qquad 1 \le j \le N, 1 \le k \le M; \tag{6}$$

Where, $E_j(v_k)$ is the degree of part-of-speech S_j releasing word V_k in all training sample.

c. HMM Decoding

When we use established HMM to tag part-of-speech, we regard span $O_0O_1\cdots\cdots O_n$ as model input, and use Viterbi algorithm to find tagging sequence which has the maximal probability for word sequence $O_0O_1\cdots\cdots O_n$, as the part-of-speech tagging sequence.

Viterbi algorithm:

(1)Initialization:
$$\delta_1(i) = \pi_i b_i(O_0), \quad 1 \le i \le N$$
$$\varphi_1(i) = 0, \quad 1 \le i \le N \tag{7}$$

(2)Recursion:
$$\delta_t(j) = \max_{1\le i\le N}[\delta_{t-1}(i)a_{ij}]b_j(O_t), \quad 1 \le t \le n, 1 \le j \le N$$
$$\varphi_t(j) = \arg\max_{1\le i\le N}[\delta_{t-1}(i)a_{ij}], \quad 1 \le t \le n, 1 \le j \le N \tag{8}$$

(3) End:
$$P^* = \max_{1\le i\le N}[\delta_n(i)])$$
$$q_n^* = \arg\max_{1\le i\le N}[\delta_n(i)] \tag{9}$$

q_n^* is the part-of-speech tagging of the word sequence $O_0O_1\cdots\cdots O_n$.

III. Experimental Results And Discussion

In order to test the technique presented above, we conducts a series of experiments on the corpus segmented and tagged by Peking University and Fujitsu. The corpus from February to June is selected as training sample which contains 6166139 words. And that of January is used as testing data (1118405 words). The results are summarized in Table 1.

TABLE I
Part-of-speech tagging results in different methods

	Close Testing		Open Testing	
	Disambiguating Accuracy	Tagging Accuracy	Disambiguating Accuracy	Tagging Accuracy
Rule-based Disambiguation	79.65%	92.81%	73.51%	90.15
HMM-based Disambiguation	91.36%	96.30%	89.41%	96.42%
Integrating Rule and HMM Disambiguation	92.97%	97.84%	91.21%	96.71%

As can be seen from the data in Table 1, the method integrating rule and HMM has an obviously better disambiguating and tagging accuracy than other two methods.

IV. Conclusion

After rule-based disambiguation, span can be shortened effectively. Span is shorter that means there are less potential tagging paths when we use HMM-based disambiguation. Therefore, HMM-based disambiguation has better disambiguating and tagging accuracy.

The accuracy of our approach leaves room for improvement. We can enhance accuracy by perfecting rule base and enlarging training corpus.

References

[1] Zhang Xiao-fei, Chen Zhao-xiong, Huang He-yan Cai Zhi. "An Approach of Processing New Words Based on HMM in Tagging of Speech of Part". JOURNAL OF CHINESE INFORMATION PROCESSING Vol.17 No.5 2003

[2] Huang De-gen, Zhang Li-jing, Zhang Yan-li, Yang Yuan-sheng. "Disambiguation Mechanism Using Rule Techniques and Statistics Techniques". *MINI—MICRO SYSTEMS* Vol.24 No.7 July 2003.

[3] Eric Brill. "Transformation-based Error-driven Learning and Natural Language Processing". *A Case Study in Part of Speech tagging. Computational Linguistic,*1995,21(4).

[4] Li Xiao-li, Shi Zhong-zhi. "A Data Mining Method to Acquire Part-of-speech Rules in Chinese Text". *JOURNAL OF COMPUTER RESEARCH&DEVELOPMENT* Vol. 37 No.12 December 2000.

[5] Kong Jun, Chen Yu-quan, Lu Ru-shan. "A Self-Learning Part-of-speeech Tagging Integrated with Partial Syntactic Analysis". *JOURANAL OF SHANGHAI JIAOTONG UNIVERSITY* Vol. 35 No. 9 September 2001.

[6] Yuan Li-chi, Zhong Yi-xin, "A Novel POS Tagging Model". *MICROELECTRONICS &COMPUTER*. Vol.22 NO.9 2005.

[7] Liang Yi-min, Huang De-gen "Chinese Part-of-speech Tagging Based on Full Second-order Hidden Markov Model". *Computer Engineering*. Vol.31 No. 10 July 2005.

[8] Wen Rui, Zhu Qiao-ming, Li Pei-feng. "The Application of HMM and Negative Feedback Model in POS Tagging". JOURNAL OF SUZHOU UNIVERSITY(NATURAL SCIENCE EDITION). pp.39-42 July 2005.