AN AUTOMATIC SENTENCE BOUNDARY DETECTOR BASED ON A STRUCTURED LANGUAGE MODEL

Shinsuke Mori *Nobuyasu Itoh* *Masafumi Nishimura*

Tokyo Research Laboratory, IBM Japan
1623-14, Shimotsuruma, Yamato-shi, Kanagawa-ken, Japan
mori@trl.ibm.co.jp

ABSTRACT

In this paper we describe an automatic sentence boundary detector, which inserts a period (sentence boundary marker) to a word sequence output by a speech recognizer. The state-of-the-art automatic sentence boundary detectors insert a period at a position selected by a word tri-gram model from among candidates (long pauses) offered by an accoustic model. In contrast, the automatic sentence boundary detector presented in this paper is based on a structured language model (SLM), which regards a sentence as a word sequence with a syntactic structure. In the experiment we applied our automatic sentence boundary detector to Japanese broadcast lectures and compared the result with an automatic sentence boundary detector based on a word tri-gram model. The accuracy of our detector was 95.7%, which was higher than that for the state-of-the-art detector (95.2%). This result shows that an SLM works better than a word tri-gram model as an automatic sentence boundary detector.

1. INTRODUCTION

Currently, the state-of-the-art speech recognizers can take dictation with a satisfactory accuracy. Now later stage of natural language processing (NLP), such as grammatical disambiguation of the dictation results, are coming into focus. Since most of these NLP systems take a sentence as an input unit, speech recognizers should insert a sentence boundary mark (period) automatically. In state-of-the-art speech recognizers, the acoustic model (AM) offers the positions of long pauses as candidates, and the language model (LM) selects linguistically reasonable ones from among them. As we mentioned above, most of the NLP systems depend on sentence boundary information, so a missrecognition of a sentence boundary is a critical, even fatal, error for these NLPs. Therefore, sentence boundary detection accuracy is more important than word recognition accuracy.

The LM of most speech recognizers is based on a word tri-gram model, which regards a sentence as a simple word sequence. The sentence boundary detection task may, however, need information about the syntactic structure of the word sequence. Thus, a word tri-gram model, which refers to only a few words around a candidate (long pause), may not have sufficient ability to detect the sentence boundary [1]. An LM based on sentence structure may outperform the state-of-the-art speech recognizers in sentence boundary detection accuracy.

Recently, a structured language model (SLM) [2], which uses structural information for word prediction, was proposed, aiming at overcoming the weakness of the word tri-gram models. The predictive power is reported to be slightly higher than an word tri-gram model. In contrast with word n-gram models, SLMs use the syntactic structure (partial parse tree) covering the preceding words at each step of word prediction. The syntactic structure also grows in parallel with the word predictions. Because of data-sparseness problems, the early SLMs are obliged to refer to only a limited and fixed part of the histories for each step of word and structure prediction. This problem has been addressed by the arboreal context tree (ACT) [3], which provides a flexible history reference mechanism.

In this paper, first we describe an SLM with ACTs. Next, we describe our sentence boundary detector. Finally, we report an experimental accuracy comparison of our automatic sentence boundary detector based on an SLM and one based on a word tri-gram model. The parameters of these models were estimated from 1,535 syntactically annotated sentences from a set of broadcast lectures in Japanese. We then tested them on 359 sentences from the same lectures. The accuracy of the sentence boundary detection of the SLM-based detector was 95.7%, higher than the accuracy of the detector based on a word tri-gram model (95.2%). This proved experimentally that an SLM improves a sentence boundary detection.

2. STRUCTURED LANGUAGE MODEL BASED ON ARBOREAL CONTEXT TREES

In this section, first we describe a structured language model (SLM) for Japanese, then the flexible history reference mechanism called arboreal context trees (ACTs), and finally, an

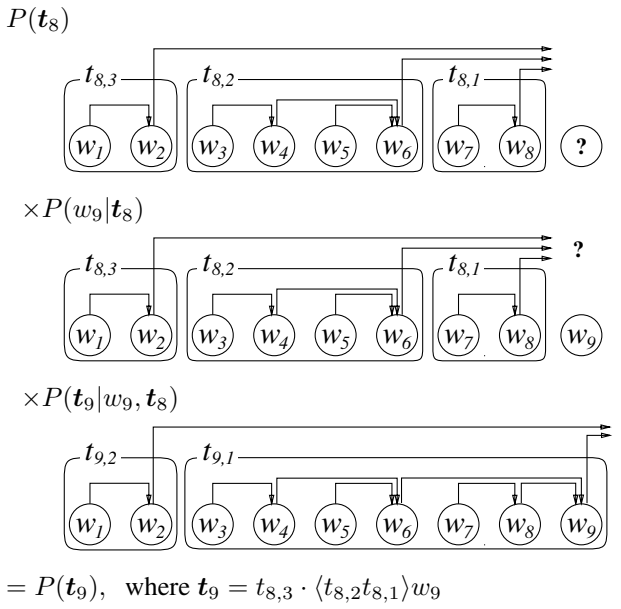

Fig. 1. Word prediction from a partial parse

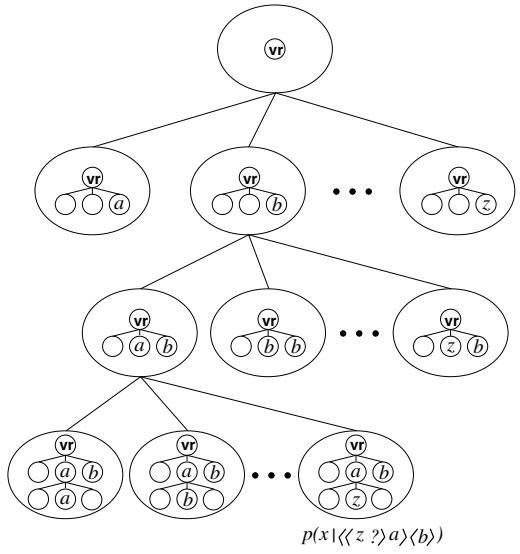

Fig. 2. An arboreal context tree (ACT).

SLM based on ACTs.

2.1. Structured Language Model

The basic idea of an SLM [2] is that each word would be better predicted from the words that may have a dependency relation with the word to be predicted than from the preceding $(n-1)$ words. Thus the probability P of a sentence $\boldsymbol{w} = w_1 w_2 \cdots w_n$ and its parse tree T is given as follows:

$$P(T) = \prod_{i=1}^{n} P(w_i|\boldsymbol{t}_{i-1}) P(\boldsymbol{t}_i|w_i, \boldsymbol{t}_{i-1}), \qquad (1)$$

where $\boldsymbol{t}_i$ is the i-th partial parse tree sequence. The partial parse tree depicted at the top of Figure 1 shows the status before the 9th word is predicted. From this status, first the 9th word w_9 is predicted from the 8th partial parse tree sequence $\boldsymbol{t}_8 = t_{8,3} t_{8,2} t_{8,1}$, and then the 9th partial parse tree sequence $\boldsymbol{t}_9$ is predicted from the 9th word w_9 and the 8th partial parse tree sequence $\boldsymbol{t}_8$ in order to get ready for the 10th word prediction.

Since in a dependency grammar of Japanese, every dependency relation is in a particular direction as shown in Figure 1 and no two dependency relations cross each other, the structure prediction model only has to predict the number of the trees depending on the next word. Thus, the second conditional probability in the right hand side of Equation (1) is rewritten as $P(l_i|\boldsymbol{t}_{i-1})$, where l is the length (number of elements) of the tree sequence $\boldsymbol{t}_i$. Our SLM for Japanese dependency grammar is defined as follows:

$$P(T) = \prod_{i=1}^{n} P(w_i|\boldsymbol{t}_{i-1}) P(l_i|w_i, \boldsymbol{t}_{i-1}). \qquad (2)$$

According to a psycholinguistic report on language structure [5], there must be an upper limit on l_i the number of the words whose modificands have not appeared yet. We set the upper limit to 9, the maximum number of slots in human short-term memory [6]. With this limitation, our SLM becomes a hidden Markov model.

2.2. Arboreal Context Tree

The problem in Equation (2) is how to classify the condition parts of the two conditional probabilities in order to predict the next word and the next structure without encountering a data-sparseness problem. In an English model [2] the next word is predicted from the two right-most exposed heads (for example w_6 and w_8 in Figure 1).

It is clear, however, that in some cases some child nodes of the tree $t_{i-1,2}$ or $t_{i-1,1}$ are useful for the next word prediction and in other cases even the consideration of an exposed head (root of the tree $t_{i-1,1}$ or $t_{i-1,2}$) causes a data-sparseness problem because of the limitation of the learning corpus size. Therefore a more flexible mechanism for history classification should improve the predictive power of the SLM.

As we mentioned above, in SLMs the history is a sequence of partial parse trees. This can be regarded as a single tree, called a history tree, by adding a virtual root

node having these partial trees under it. An arboreal context tree is a data structure for flexible history tree classification. Each node of an ACT is labeled with a subtree of the history tree. The label of the root is a null tree and if a node has child nodes, their labels are the series of trees made by expanding a leaf of the tree labeling the parent node. For example, each child node of the root in Figure 2 is labeled with a tree produced by adding the right most child to the label of the root. Each node of an ACT has a probability distribution $P(x|t)$, where x is an alphabet and t is the label of the node. For example, let $\langle a_k \cdots a_2 a_1 \rangle a_0$ represent a tree consisting of the root labeled with a_0 and k child nodes labeled with $a_k, \cdots, a_2$, and a_1, so the rightmost node at the bottom of the ACT in Figure 2 has a probability distribution of the alphabet x under the condition that the history matches the partial parse trees $\langle\langle z? \rangle a \rangle \langle b \rangle$, where "?" matches with an arbitrary alphabet. Putting it in another way, the next word is predicted from the history having b as the head of the rightmost partial parse tree, a as the head of the second rightmost partial parse tree, and z as the second rightmost child of the second rightmost partial parse tree.

2.3. An SLM with ACTs

An ACT is applied to classification of the condition parts of both two conditional probabilities in Equation (2). Thus, an SLM with ACTs is defined as follows:

$$P(T) = \prod_{i=1}^{n} P(m_i | ACT_m(\langle \boldsymbol{t}_{i-1} \rangle)) \times P(l_i | ACT_s(\langle \boldsymbol{t}_{i-1} m_i \rangle)), \quad (3)$$

where ACT_m is an ACT for word prediction and ACT_s is an ACT for structure prediction. Note that this is a generalization of the prediction from the two rightmost exposed heads (w_6 and w_8) in the English model [2]. In general, an SLM with ACTs includes SLMs with fixed history reference mechanism as special cases.

3. SENTENCE BOUNDARY DETECTOR

In this section, first we discuss two sorts of information for sentence boundaries: acoustic information and linguistic information. Next, we explain how a sentence boundary detector determines positions of sentence boundaries from these sorts of information.

3.1. Acoustic Information

Normally, speakers have a tendency to put a longer pause between two sentences than between two words. Thus, the duration of a silence captured by an acoustic model is a clear clue of a sentence boundary. But the pauses between sentences are not always longer than the longest pause between words. An acoustic model can only enumerate candidates

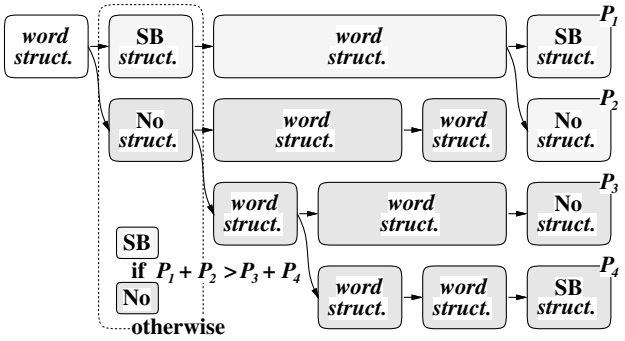

Fig. 3. Probabilistic voting for decision.

for sentence boundaries. Therefore, we set a threshold to the duration of silence, and the threshold must be low enough so that the recall will be very close to 100%. In the experiment we have conducted, the threshold was set to be 300ms and the recall in a set of samples was 100%.

3.2. Linguistic Information

Linguistic information can help distinguish sentence boundaries from simple long pauses. In each language, there are expressions which have a tendency to appear at the beginning or at the end of sentences and others which have a tendency not to appear at the beginning or end. In English, for example, the articles do not appear at the end of grammatical sentences. Conventional word tri-gram models are able to capture these tendencies. In addition, there are also structural characteristics especially at the end of sentences. Every phrasal unit in a sentence has a certain grammatical relationship with other phrasal units and each pair of phrasal units in a sentence are connected with each other directly or indirectly by grammatical relationships. Putting it another way, if there are grammatically isolated units between the beginning and some position in a set sequence, this position does not tend to be a sentence boundary. The SLMs are capable of capturing this characteristic, which is beyond the descriptive power of word tri-gram models.

3.3. Search Algorithm

As we mentioned above, a language model, given candidate positions from an acoustic model, calculates the probability that each position is or is not a sentence boundary. The problem here is the timing of a decision on a candidate position. There is no clear sentential end at the very end of speech.

In our sentence boundary detector each decision is made when the next candidate is given by an AM as shown in Figure 3. At the first candidate position, the LM holds two possibilities: a sentence boundary and a simple long pause.

Table 1. An experimental result.

Sentence boundary detector	accuracy
SLM with ACTs	95.68% (10803/11291)
word tri-gram model	95.16% (10745/11291)
baseline (always "No")	85.21% (9621/11291)

When the second candidate position is given by the AM, the LM executes a "probabilistic voting," where the LM sums up the probabilities of the nodes connected to the node labeled with sentence boundary (SB) at the previous candidate position (ex. $P_1 + P_2$ in Figure 3), and also sums up the probabilities of the nodes connected to the node not labeled with a sentence boundary (No) at the previous candidate position (ex. $P_3 + P_4$), and compares these two probabilities to make a decision on the previous candidate position. Note that a "struct." in Figure 3 represents $ACT_m(\boldsymbol{t})$ in Equation (3), which is always the same in a word tri-gram model.

4. EVALUATION

We developed a sentence boundary detector based on an SLM with ACTs and one based on an orthodox word tri-gram model. In this section, we report the results of the sentence boundary detection experiments and discuss them.

4.1. Conditions of the Experiments

The corpus used in our experiments is a set of transcribed Japanese broadcast lectures from a bachelor's degree program. The corpus contains 2,004 sentences. Each of them is segmented into words and annotated with its syntactic structure. Each word is annotated with a part-of-speech. Each sentence contains marks representing pauses between words longer than 300 ms, and some of them are marked as sentence boundaries by linguists in our lab. The task of the sentence boundary detectors in the experiment is to select these sentence boundaries from among the long pauses. The corpus was divided into ten parts; the parameters of the model were estimated from nine of them and the model was tested on the remaining one (10-fold cross validation).

4.2. Evaluation

The criterion for sentence boundary detection is the ratio of correct decisions over all decisions to be made (pauses longer than 300 ms):

$$\text{accuracy} = \frac{\#\text{correct decisions}}{\#\text{long pauses}}.$$

Table 1 shows the accuracies of the sentence boundary detector based on SLM with ACTs, one based on a word tri-gram model, and a baseline in which all long pauses are not regarded as sentence boundaries. This result shows that the SLM with ACTs reduces the errors by 10.6% compared to the state-of-the-art sentence boundary detector based on word tri-grams. Since the accuracy of the state-of-the-art method is close to 100% and sentence boundary information plays a very important role in NLP following speech recognizers, this improvement can be regarded as significant. In addition, the SLM outperforms the orthodox word tri-gram model in this task as well as in the word prediction task for a speech recognizer. This result experimentally shows an advantage of SLMs over word tri-gram models.

5. CONCLUSION

We have described a sentence boundary detector based on an SLM with ACTs and reported an experimental result on a set of broadcast lectures in Japanese. The accuracy of the SLM-based detector was better than a detector based on word tri-gram model. This proved experimentally that an SLM improves sentence boundary detection.

6. REFERENCES

[1] Douglas B. Paul and Janet M. Baker, "The design for the Wall Street Journal-based CSR corpus," in *Proceedings of the DARPA Speech and Natural Language Workshop*, 1992, pp. 357–362.

[2] Ciprian Chelba and Frederic Jelinek, "Structured language modeling," *Computer Speech and Language*, Vol. 14, pp. 283–332, 2000.

[3] Shinsuke Mori, Masafumi Nishimura, and Nobuyasu Itoh, "Improvement of a structured language model: Arbori-context tree," in *Proceedings of the Seventh European Conference on Speech Communication and Technology*, 2001.

[4] Dana Ron, Yoram Singer, and Naftali Tishby, "The power of amnesia: Learning probabilistic automata with variable memory length," *Machine Learning*, Vol. 25, pp. 117–149, 1996.

[5] Victor H. Yngve, "A model and a hypothesis for language structure," *The American Philosophical Society*, Vol. 104, No. 5, pp. 444–466, 1960.

[6] George A. Miller, "The magical number seven, plus or minus two: Some limits on our capacity for processing information," *The Psychological Review*, Vol. 63, pp. 81–97, 1956.