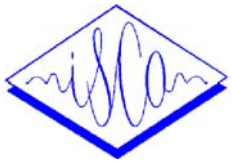

6th European Conference on
Speech Communication and Technology
(EUROSPEECH'99)
Budapest, Hungary, September 5-9, 1999

A PRIMARY STUDY ON THE RANDOMNESS CONTROL OF THE PROSODIC BOUNDARY INDEX FOR NATURAL SYNTHETIC SPEECH

*Ki-Wan Eom, *Jin-Young Kim, and **Sun-Mi Kim

*Chonnam National University, **Seoul National University

*{eom, kimjin}@dsp.chonnam.ac.kr, **sunmi@acoustics.snu.ac.kr

10.21437/Eurospeech.1999-398

ABSTRACT

In a text-to-speech (TTS) system a proper prosody control is necessary for natural synthetic speech. But synthetic speech generated by regular rules can make a listener irritated and bored, because the implemented prosody is always same for the same sentence. In this paper, the randomness of the prosodic boundary index (PBI), as a primary study on irregularity of prosody is discussed.

We examined the PBI data of 1,800 spoken sentences and concluded that there were irregularities. We applied the conventional stochastic method (CSM) to the PBI prediction. However, CSM could not implement PBI irregularity. So we proposed *local constraint Viterbi search algorithm (LCVS*) as an alternative method. *LCVS* was evaluated by computer simulation.

Keywords: PBI randomness, local constraint Viterbi search algorithm

1. INTRODUCTION

The goal of a TTS system is to translate a written text into intelligible and natural speech. Generating natural prosody is one of the main problems for obtaining a good synthetic speech. The first process in a prosody generation is a PBI assignment for a given text. This is based on the fact that every speaker tends to group words into phrases in natural speech. Of course this phrasing is very important to understanding the utterances. Thus it is necessary to assign break levels between words in a high quality TTS system.

A number of algorithms have been proposed for a PBI assignment by a number of researchers from the simple deterministic rules to the complex stochastic algorithms[1-4]. The TTS system adopted may make a listener irritated or bored with the synthetic speech. The reason is that the implemented prosody is always same for the same sentence. In other words the rules are stochastic, but the implemented prosody is deterministic.

We have studied the random phenomena of prosody. In natural speech the prosody of a spoken sentence was perturbed every time, in spite of the same syntactic structure. Our target is the randomness control of the PBI assignment. We propose a modified PBI assignment algorithm, which generates different PBI sequence for the same sentence in every trial.

To predict PBI for the sentences we used only text-based features of POS sequences, especially the trigram approach. Including a non-break boundary, we assume that all prosodic boundaries are between every pair of words in a sentence. The boundary between adjacent words is then marked as one of the boundary types, i.e. no-break, weak-break, and strong-break. We applied the Markov model to the PBI prediction problem.

First, we tested the trigram PBI estimation method against 1,800 spoken sentences. Second, we classified the trigram rules into two classes. One is the path-independent (PI) and the other is the path-dependent (PD). PIs are trigram rules of which PBIs are independent of syntactic structure. PDs are dependent on syntactic environment. Third, we devised a randomness control method. To decide the best sequence of the PBI for each sentence we modified the Viterbi search algorithm to make the boundaries using PI trigram rules. In the PI case, PBI is randomly determined, based on the given occurrence probability of each boundary type. PBIs of PDs are determined by a local constraint Viterbi search (*LCVS*) algorithm. PBIs are fixed at the PI nodes in the *LCVS* and the probability is maximized only for the PD nodes.

We evaluated the proposed method by a computer simulation. For each trigram rule the simulated PBIs had the similar distribution of the original PBI.

2. SPEECH MATERIAL

For our study we collected speech DB of 150 sentences, with various syntactic structures and a similar number of words. All sentences are declarative sentences. Those sentences were read by six speakers in their 30's, with a Seoul dialect in a conversation style. Each sentence was read twice at a normal speed. Thus, the test

material consisted of 1,800(6 speakers x 2 times x 150 sentences) utterances in which there were 22,818 word boundaries in the lexical text. Then, on the basis of a phonetician's auditory impression, the utterances spoken by the six different speakers were marked respectively between every pair of words. The sentences were labeled in terms of three levels of boundaries, i.e. no-break (NB), weak-break (WB), and strong-break (SB). Also we manually tagged the sentences using the 18 part-of-speeches (POS) which was devised by a special linguist[5].

3. RANDOMNESS OF PBI ASSIGNIMENT

3.1 Stochastic Model

For the given POS tagged sequence, denoted by $d_{1\cdots N}$, in an utterance, the most likely PBI sequence can be obtained from the stochastic model. The problem is described as follows.

$$\arg\max_{bi_{1\cdots N-1}} P(bi_{1\cdots N-1} \mid d_{1\cdots N}) \quad (1)$$

And using Bayes' rule we can represent the above problem as

$$\arg\max_{bi_{1\cdots N-1}} \frac{P(d_{1\cdots N} \mid bi_{1\cdots N-1})P(bi_{1\cdots N-1})}{P(d_{1\cdots N})} \quad (2)$$

$$= \arg\max_{bi_{1\cdots N-1}} P(d_{1\cdots N} \mid bi_{1\cdots N-1})P(bi_{1\cdots N-1}) \quad (3)$$

Because the prosodic boundary index sequences are assumed to be a Markov process, the above equation can approximate this.

$$\arg\max_{bi_{1\cdots N-1}} \prod_{k=1}^{N-1} P(d_{k-1}, d_k, d_{k+1} \mid bi_k) P(bi_k \mid bi_{k-1}) \quad (4)$$

Where bi_k is the PBI between k-th and (k+1)-th words and d_k is the POS tag of k-th word[6].

To decide the best sequence of PBI for each sentence, we use a Markov model where each state represents one of the PBI types i.e. NB, WB, or SB. Observation probabilities indicate POS sequences occurring. We take the trigram to represent the POS sequence.

3.2 Classification of POS trigram set into PI and PD

We classified trigrams into two classes. One is the path-independent (PI) and the other is the path-dependent (PD). PIs are trigrams of which PBIs are independent of syntactic structure. Of course PDs are dependent on syntactic environment.

For example, the trigram shown in *table 1* occurs three times in the 150 sentences. The occurrence pattern of each boundary type is very similar every time. In this case we defined the trigram rule as PI one. In the case of the trigram rules as shown in *table 2*, we classified it as PD.

Table 1. The PI example for a POS trigram

Occurrence	NB	WB	SB
1st	9	3	0
2nd	9	3	0
3rd	10	2	0

Table 2. The PD example for a POS trigram

Occurrence	NB	WB	SB
1st	12	0	0
2nd	4	4	4
3rd	6	3	3
4th	3	9	0

We used the standard deviation value to classify trigrams. The value of the decision boundary was chosen to one. The trigram is determined as a PI trigram rule if the maximum of standard deviations of each boundary type's occurrence frequency is less than one. Then the occurrence probability of each boundary type is calculated. PI does not mean that the PI trigram has only one break index but that its statistical characteristics do not depend on its position in sentences. *Table 1* shows the probabilities of PI boundary types are 0.78 (28/36, NB), 0.22(8/36,WB), and 0(0/36, SB), respectively. On the other hand, there were a total of 885 kinds of trigrams in the 150 sentences. The PIs occur in 38% of all the trigrams.

3.3 Local Constraint Viterbi Search Algorithm

We devised a randomness control method based on the Markov model, where each state is determined by a Viterbi search algorithm. The most likely PBI sequence for a given sentence can be found. But, we don't always want the most probable PBI sequence. The result is deterministic.

Thus we modified the Viterbi search algorithm to reflect the randomness phenomena, so that the obtained path passes the predefined PBI in the PI boundaries. The Viterbi search was performed under the condition called a local constraint. The PI case, PBI, is randomly determined based on the given occurrence probabilities of three boundary types. A random number generator gives a number between zero and one with uniform distribution. Then the PBIs of PDs are determined by a local constraint Viterbi search (*LCVS*) algorithm. PBIs are fixed at the PI nodes in *LCVS* and the probability is maximized only for PD nodes.

The algorithm is given below *table 3*.

Table 3. Local Constraint Viterbi Search Algorithm

1. Initialization :

for i=1 to N

$$\Delta_1(i) = \begin{cases} \pi_i b_i(o_1) & \text{, if } o_1 = \text{PD} \\ \pi_i \delta(i\text{-}k) & \text{, if } o_1 = \text{PI} \end{cases}$$

$$\varphi_1(i) = 0$$

end

Where k is predefined PBI of o_1.

2. Recursion :

for j=1 to N

for i=1 to N

$$\Delta_t(j) = \begin{cases} \max\limits_{1\le i\le N}[\Delta_{t-1}(i)a_{ij}]b_j(o_t) & \text{, if } o_t = \text{PD} \\ \max\limits_{1\le i\le N}[\Delta_{t-1}(i)a_{ij}]\delta(j-k) & \text{, if } o_t = \text{PI} \end{cases}$$

end

$$\varphi_t(j) = \arg\max_{1\le i\le N}[\Delta_{t-1}(\mathrm{i})a_{ij}]$$

end

Where k is predefined PBI of o_t.

3. Termination :

$$p_T = \max_{1\le i\le N}[\Delta_T(i)]$$
$$q_T = \arg\max_{1\le i\le N}[\Delta_T(i)]$$

In the above algorithm, the observation sequence $O = (o_1\, o_2 \cdots o_T)$ shows POS trigrams in a given sentence, the a_{ij} means the state (boundary types) transition probabilities and δ is the Dirac delta function.

The *figure 1* shows our concept of the *LCVS* algorithm.

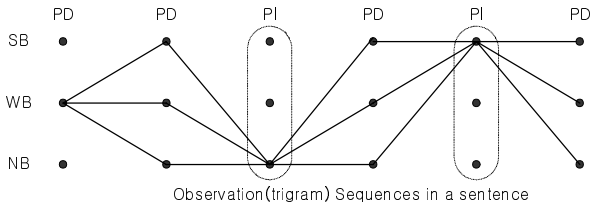

Figure 1. State transition (trellis) diagram in the local constraint Viterbi search algorithm

4. EXPERIMENTAL RESULTS

Experimental results pertaining to randomly control assignment of the appropriate prosodic boundary index are presented in *figure 2*.

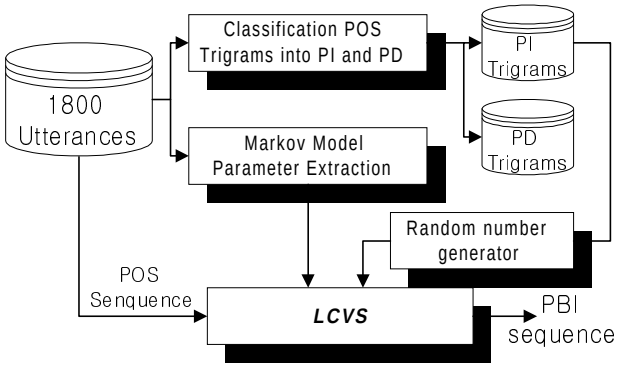

Figure 2. Randomness control for PBI assignment

Firstly, we assessed the performance of a prosodic boundary index prediction by means of Viterbi search.

The tests were conducted on other sets (in testing) for each utterance set (in training). In the PBIs prediction test, we achieved around 73 to 84% accuracy.

Secondly, we judged the success of the randomness control of the PBIs using the proposed method in this paper. In this test, we generated the PBI sequences for the given sentences using the *LCVS* method. An example of the randomness control results for a given sentence is shown in *table 3* on the next page. In the table, NB, WB, and SB are denoted by “0”, “1”, and “2”, respectively

If the PBIs are predicted by an original Viterbi search algorithm, the result will be same every time.

Unfortunately, it is not easy to evaluate the success of the randomness control of the prosodic boundary index assignment. So we compared the simulated PBIs by proposed method with the distribution of the original PBIs in the corpus. We evaluated the proposed method by a computer simulation. For each trigram rule the simulated PBIs had the similar distribution of the original PBIs.

5. CONCLUSION

In this paper, we presented the randomness control problem of the prosodic boundary index as a primary study on the randomness control of prosodic boundary index.

Although for the each trigram rule the simulated PBIs had the similar distribution of the original PBIs, testing listening on synthetic speech will need a practical feasibility test of our proposed algorithm.

We are under the development of TTS system adequate to randomness control test. We will perform the listening test later.

ACKNOWLEDGEMENTS

This paper was supported by the Korean Research Foundation under Grant Interdisciplinary Research Project 97-98.

6. REFERENCES

[1] Paul Taylor and Alan W Black (1998), Assigning Phrase Breaks from Part-of-Speech Sequences. *Computer Speech and Language*, Vol. 12, pp. 99-117.

[2] Pijper, J.R. de and Sanderman, A.A. (1994), On the perceptual strength of prosodic boundaries and its relation to suprasegmental cues. *J. Acoust. Soc. Am. 96 (4)*, pp. 2037-2047.

[3] Cutler A. and Butterfield S. (1991), Word boundary cuse in clear speech: A supplementary report. *Speech Communication*, 10. Pp. 335-353.

[4] M. Ostendorf and N. Veilleux (1994), A Hierarchical Stochastic Model for Automatic Prediction of Prosodic Boundary Location. *Computational Linguistics*, Vol. 20(1), pp. 27-52.

[5] Sun-Mi Kim (1997), *Rhythmic Units and Syntactic Structures in Korean*. Ph.D. thesis, Seoul National University.

[6] N. Veilleux et al. (1990), Markov Modeling of Prosodic Phrase Structure. *ICASSP*, pp. 777-780.

Table 4. For a sentence, original PBI and generated PBI by LCVS distribution

A. Original PBI distribution for a sentence (6 speaker x 2 times)

Word(POS)	1	2	3	4	5	6	7	8	9	10	11	12
이때에(NV)	2	0	2	2	0	1	2	2	0	1	1	0
등장인물들이(NS)	0	0	0	0	0	0	1	0	0	0	1	0
사용하는(VA)	0	0	0	0	0	0	1	1	0	0	0	0
언어는(NM)	2	2	2	2	2	2	2	2	2	2	2	2
우리가(NS)	0	0	0	0	0	0	1	1	0	1	2	0
일상생활에서(NV)	0	1	0	0	0	0	0	0	0	0	2	0
사용하는(VA)	0	0	0	0	0	0	1	1	0	0	0	0
언어와(NV)	2	2	2	2	2	2	2	2	2	2	2	2
거의(E)	1	1	0	0	1	1	0	0	0	1	0	0
차이가(NS)	0	0	0	0	1	1	0	0	0	0	0	0
없다(VD).												

B. Using LCVS method, generated PBI distribution for a sentence (12 times work)

Word(POS)	1	2	3	4	5	6	7	8	9	10	11	12
이때에(NV)	2	2	2	1	0	1	1	0	2	2	2	10
등장인물들이(NS)	0	0	0	0	0	0	0	0	0	0	0	0
사용하는(VA)	0	0	0	0	0	0	0	0	0	0	0	0
언어는(NM)	2	2	2	2	2	1	2	2	2	2	2	2
우리가(NS)	0	0	0	1	2	2	0	2	0	2	0	0
일상생활에서(NV)	0	1	0	0	0	0	0	0	0	0	0	0
사용하는(VA)	0	0	0	0	0	0	0	0	0	1	2	0
언어와(NV)	2	2	2	2	2	2	2	2	2	2	2	2
거의(E)	0	0	1	1	0	1	0	0	0	0	0	0
차이가(NS)	1	0	1	0	0	1	1	0	0	0	1	0
없다(VD).												

NV: Noun + Adverbial Particle, **NS**: Noun + Subjective Particle, **NM**: Noun + Modifying Particle, **VD**: Verb + Declarative Ending, **VA**: Verb + Adnominal Clause Ending, **E**: Adverb.