EURASIP Journal on Applied Signal Processing 2003:2, 128–139

A Statistical Approach to Automatic Speech Summarization

Chiori Hori

Department of Computer Science, Tokyo Institute of Technology, 2-12-1 O-okayama, Meguro-ku,
Tokyo 152-8552, Japan
Email: chiori@furui.cs.titech.ac.jp

Sadaoki Furui

Department of Computer Science, Tokyo Institute of Technology, 2-12-1 O-okayama, Meguro-ku,
Tokyo 152-8552, Japan
Email: furui@furui.cs.titech.ac.jp

Rob Malkin

Interactive Systems Labs, Carnegie Mellon University, Pittsburgh, PA 15213, USA
Email: malkin@cs.cmu.edu

Hua Yu

Interactive Systems Labs, Carnegie Mellon University, Pittsburgh, PA 15213, USA
Email: hua@cs.cmu.edu

Alex Waibel

Interactive Systems Labs, Carnegie Mellon University, Pittsburgh, PA 15213, USA
Email: ahw@cs.cmu.edu

Received 20 March 2002 and in revised form 11 November 2002

This paper proposes a statistical approach to automatic speech summarization. In our method, a set of words maximizing a summarization score indicating the appropriateness of summarization is extracted from automatically transcribed speech and then concatenated to create a summary. The extraction process is performed using a dynamic programming (DP) technique based on a target compression ratio. In this paper, we demonstrate how an English news broadcast transcribed by a speech recognizer is automatically summarized. We adapted our method, which was originally proposed for Japanese, to English by modifying the model for estimating word concatenation probabilities based on a dependency structure in the original speech given by a stochastic dependency context free grammar (SDCFG). We also propose a method of summarizing multiple utterances using a two-level DP technique. The automatically summarized sentences are evaluated by summarization accuracy based on a comparison with a manual summary of speech that has been correctly transcribed by human subjects. Our experimental results indicate that the method we propose can effectively extract relatively important information and remove redundant and irrelevant information from English news broadcasts.

Keywords and phrases: speech summarization, summarization scores, two-level dynamic programming, stochastic dependency context free grammar, summarization accuracy.

1. INTRODUCTION

The revolutionary increases in the computing power and storage capacity have enabled an enormous amount of speech data, or multimedia data that includes speech, to be managed as an information source. The next step is to create a system in which speech data is tagged (annotated) by text allowing information to be retrieved and extracted from such databases. Multimedia databases including indexes can be automatically constructed using speech-recognition systems. Speech can be broadcast with captions generated by speech-recognition systems and simultaneously saved in speech and text (i.e., captions) archives in a database. Captioning can be considered a form of indexing accessible by individual words in the whole speech. One approach attempted to extract information from such a database by tracking speech through query matching to indexes based on automatic recognition results which had been synchronized with the speech data [1]. However, users attempting to retrieve information from such a speech database prefer to access abstracts rather than the whole range of data before they decide whether they are going to read or hear the entire body of information or not. The summarization of meetings/conferences will become useful if it can be developed to extract relatively important information scattered throughout the original speech. Techniques to compress and summarize information from meetings and conferences are actively being investigated [2, 3]. Speech summarization is particularly important in the closed captioning of broadcast news (BN) to reduce the number of captioned words representing speech, because the number of words spoken by professional announcers sometimes exceeds the number that people can read or understand when these are presented on a TV screen in real time.

Our goal is to build a system that extracts and presents information from spoken utterances based on the amount of information users want. Figure 1 is a flowchart of our proposed system. The output of the system can be a summarized sentence of an individual utterance or a summarization of a speech that contains multiple utterances. These outputs can be used for indexing and making closed captions and abstracts to name a few. The extracted information can be represented by original speech, text, or synthesized speech.

Although state-of-the-art speech recognition technology can obtain high recognition accuracy for speech read from a previously written text or similar types of pre-prepared language, the accuracy is quite poor for freely spoken spontaneous speech. Spontaneous speech is ill-formed and very different from written text. Even though a speech recognition system can accurately transcribe, the transcription usually includes redundant information such as disfluencies, filled pauses, repetitions, repairs, and word fragments. Irrelevant information also included in the transcription due to recognition errors is usually inevitable. Transcriptions that include such redundant and irrelevant information cannot be directly used for indexing, or preparing abstracts or minutes. A speech summarization technique that includes both information extraction and skimming technology will be required in the near future to construct a system whereby archived multimedia can be freely accessed using large vocabulary continuous recognition (LVCSR) systems.

Speech conveys both linguistic and paralinguistic (prosodic) information. Chen and Withgott [4] reported the usefulness of prosodic information in discourse speech summarization. However, Kobayashi et al. [5] reported that prosodic information was difficult to use in summarizing monologues. Since we are interested in summarizing monologues such as those in BN and presentations, this paper focuses on using the linguistic information obtained through automatic speech recognition.

Techniques for automatically summarizing written text have been actively explored throughout the field of natural language processing [6]. One of the main techniques of summarizing written text is the process of extracting important sentences. Recently, Knight and Marcu [7] proposed a sentence compression method based on training using a pair of texts and their abstracts. There is a major difference between text summarization and speech summarization due to the fact that transcribed speech is sometimes linguistically incorrect due to the spontaneity of speech and errors in recognition. A new approach to automatically summarizing speech is needed to solve these problems.

We have already proposed an automatic speech summarization technique for Japanese speech [8, 9, 10], which can effectively summarize Japanese news broadcasts and presentations. Since our method is based on a statistical approach, it can also be applied to other languages. In this paper, English news broadcasts transcribed by a speech recognizer [11] are automatically summarized and the accuracy of the technique is evaluated.

2. SUMMARY OF EACH UTTERED SENTENCE

The process of summarizing speech involves excluding recognition errors and maintaining important information. In addition, the summarized sentence should be meaningful. Therefore, our summarization approach focuses on topic-word extraction, weighting correct-word concatenations linguistically and semantically, and reliable parts of speech recognition acoustically as well as linguistically.

Our sentence-by-sentence speech summary method extracts a set of words maximizing a summarization score from an automatically transcribed sentence according to a summarization ratio, and it concatenates them to build a summary. The summarization ratio is the number of characters/words in the summarized sentence divided by the number of characters/words in the original sentence. The summarization score, indicating the appropriateness of a summarized sentence, is defined as the sum of the word significance score I, the confidence score C of each word in the original sentence, the linguistic score L of the word string in the summarized sentence [8, 9], and the word concatenation score T [10]. The word concatenation score given by the SDCFG indicates the word concatenation probability determined by the dependency structure in the original sentence.

Given a transcription result consisting of N words, $W = w_1, w_2, \ldots, w_N$, the summarization is done by extracting a set of M $(M < N)$ words, $V = v_1, v_2, \ldots, v_M$, which maximizes the summarization score given by

$$\begin{aligned} S(V) = \sum_{m=1}^{M} \{ I(v_m) + \lambda_L L(v_m | \cdots v_{m-1}) \\ + \lambda_C C(v_m) + \lambda_T T(v_{m-1}, v_m) \}, \end{aligned} \tag{1}$$

where λ_L, λ_C, and λ_T are the weighting factors to balance the dynamic ranges of L, I, C, and T. To reinforce each score, each word is accompanied by the POS (part-of-speech) information. Therefore, w actually indicates the tuple of (w, POS).

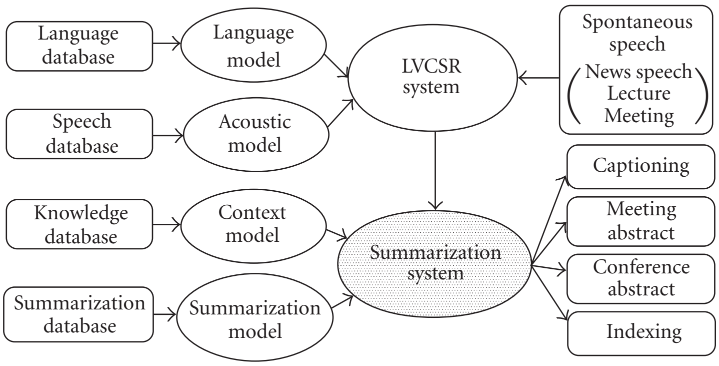


Figure 1: Automatic speech summarization system.

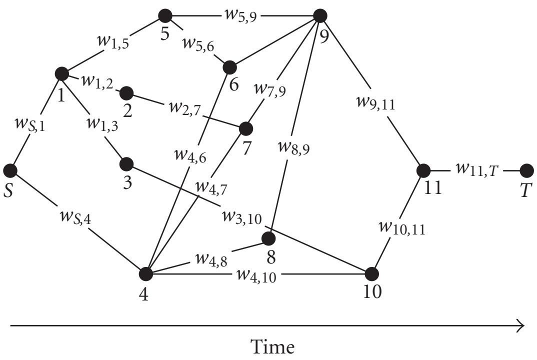


Figure 2: Example of word graph.

This method is effective in reducing the number of words by removing redundant and irrelevant information without losing relatively important information. A set of words maximizing the total score is extracted using a dynamic programming (DP) technique [8].

2.1. Word significance score

The word significance score I indicates the relative significance of each word in the original sentence [8]. The amount of information based on the frequency of each word given by (2) is used as the word significance score for topic words,

$$I(w_i) = f_i \log \frac{F_A}{F_i}, \tag{2}$$

where w_i is a topic word in the transcribed speech, f_i is the number of occurrences of w_i in the transcription, F_i is the number of occurrences of w_i in all the training documents, and F_A is the summation of all F_i in all the training documents ($= \sum_i F_i$).

The w_i which frequently occurs throughout all documents is deweighted by the measure given by (2). Our preliminary experiments revealed that this is more effective than the tf-idf measure in which w_i is deweighted, based on its homogeneous occurrence in documents in the collected data.

In this study, we choose nouns and verbs as topic words for English. We awarded a flat score to words other than topic words. To reduce the repetition of words in the summarized sentence, we also awarded a flat score to each reappearing noun and verb.

2.2. Linguistic score

The linguistic score $L(v_m | \cdots v_{m-1})$ indicates the appropriateness of the word strings in a summarized sentence and it is measured by the logarithmic value of n-gram probability $P(v_m | \cdots v_{m-1})$ [8]. In contrast with the word significance score which focuses on topic words, the linguistic score is helpful in extracting other words that are necessary to construct a readable sentence.

2.3. Confidence score

We incorporated the confidence score $C(v_m)$ to weight reliable hypotheses acoustically as well as linguistically [9]. Specifically, the posterior probability of each transcribed word, that is, the ratio of word hypothesis probability to that of all other hypotheses, is calculated using a word graph obtained through a decoder and used as a measure of confidence [12, 13]. A word graph consisting of nodes and links from the beginning node S to the end node T is shown in Figure 2.

Nodes represent time boundaries between possible word hypotheses, and the links connecting these nodes represent word hypotheses. Each link is given the acoustic log likelihood and the linguistic log likelihood of a word hypothesis.

The posterior probability of a word hypothesis $w_{k,l}$ is given by

$$C(w_{k,l}) = \log \frac{\alpha_k P_{\mathrm{ac}}(w_{k,l}) P_{\mathrm{lg}}(w_{k,l}) \beta_l}{\mathcal{G}}, \tag{3}$$

where k, l is the node number in word graph ($k < l$), $w_{k,l}$ is the word hypothesis occurring between node k and node l, $C(w_{k,l})$ is the log of posterior probability of $w_{k,l}$, α_k is the forward probability from the beginning node S to node k, β_l is the backward probability from node l to the end node

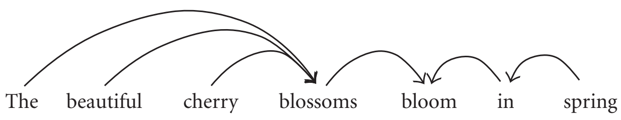


Figure 3: Example of dependency structure.

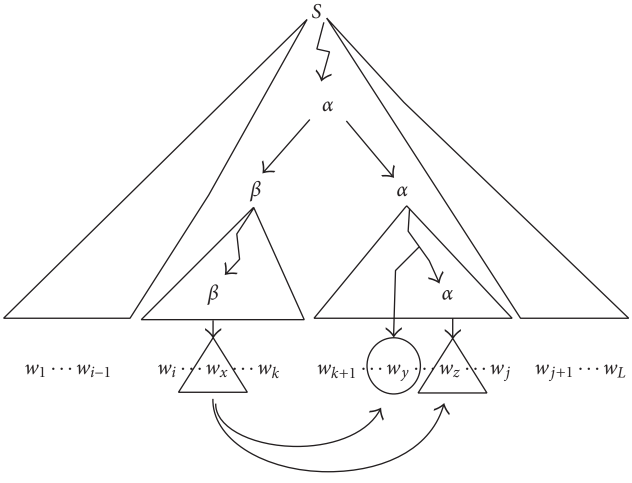


Figure 4: Phrase structure tree based on dependency structure.

T, $P_{\text{ac}}(w_{k,l})$ is the acoustic likelihood of $w_{k,l}$, $P_{\text{lg}}(w_{k,l})$ is the linguistic likelihood of $w_{k,l}$, and $\mathcal{G}$ is the forward probability from the beginning node S to the end node T $(= \alpha_T)$.

2.4. Word concatenation score

Suppose that "the beautiful cherry blossoms in Japan" is summarized as "the beautiful Japan." The summary is grammatically correct but semantically incorrect. Since its linguistic score is not powerful enough to alleviate this problem, we incorporated a word concatenation score $T(v_{m-1}, v_m)$ to penalize the concatenation between words that had no dependency in the original sentence. Every language has its own structures for dependency, and basic computation of the word concatenation score independent of the type of language is described below.

2.4.1 Dependency structure

The arches in Figure 3 show the dependency structure represented by a dependency grammar. In a dependency grammar, one word is designated as the "head" of the sentence, and all other words are either a "dependent" of that word, or dependent on some other word which is connected to the "head" word through a sequence of dependencies [14]. The word at the tail of the arrow in the arches is the "modifier," and the word at the point of the arrow is the "head." For instance, the dependency grammar of English consists of both right-headed dependency indicated by the arrows pointing right and left-headed dependency indicated by the arrows pointing left. These dependencies can be represented by a phrase structure grammar, that is, a dependency context free grammar (DCFG), using the following rewriting rules based on Chomsky's normal form:

$$\begin{aligned} &\alpha \longrightarrow \beta\alpha \text{ (right-headed)},\\ &\alpha \longrightarrow \alpha\beta \text{ (left-headed)},\\ &\alpha \longrightarrow w, \end{aligned} \tag{4}$$

where α and β are nonterminal symbols and w is a terminal symbol (word). Figure 4 has an example of a phrase structure tree based on a word-based dependency structure for a sentence which consists of L words, $w_1, \ldots, w_L$. The w_x modifies w_z when a sentence is derived from the initial symbol S and the following requirements are fulfilled: (1) the rule $\alpha \rightarrow \beta\alpha$ is applied; (2) $w_i \cdots w_k$ is derived from β; (3) w_x is derived from β; (4) $w_{k+1} \cdots w_j$ is derived from α; and (5) w_z is derived from α.

2.4.2 Dependency probability

Since the dependencies between words are usually ambiguous, whether or not there are dependencies between words must be estimated by a dependency probability that one word is being modified by the others. In this study, the dependency probability is calculated as a posterior probability estimated by the inside-outside probabilities [15] based on the SDCFG. The probability that the w_x and w_z relationship has a right-headed dependency structure is calculated as a product of the probabilities of the above steps from (1) to (5). However, left-headed dependency probability is calculated as the product of probabilities when rule $\alpha \rightarrow \alpha\beta$ is applied. Since English has both right and left dependencies, the dependency probability is defined as the sum of the right-headed and left-headed dependency probabilities. If a language has only right-headed dependency, the right-headed dependency probability is used for dependency probability. For simplicity, the dependency probabilities between w_x and w_z are denoted by $d(w_x, w_z, i, k, j)$, where i and k are the indices of the initial and final words derived from β, and j is the index of the final word derived from α. The dependency probability is calculated as

$$\begin{aligned} &d(w_m, w_l, i, k, j)\\ &\quad = \Big\{ \sum_{\alpha\beta} f(i, j|\alpha)P(\alpha \longrightarrow \beta\alpha)h_m(i, k|\beta)h_l(k+1, j|\alpha)\\ &\qquad\quad + \sum_{\alpha\beta:\alpha\neq\beta} f(i, j|\alpha)P(\alpha \longrightarrow \alpha\beta)h_m(i, k|\alpha)h_l(k+1, j|\beta) \Big\}, \end{aligned} \tag{5}$$

where P is the rewrite probability and f is the outside probability given by (A.3) in the appendix. The h is the head-dependent inside probability that w_n is the head of a word string derived from α, which is defined as

$$h_n(i,j|\alpha) = \begin{cases} \displaystyle\sum_{\beta} \Big\{ \sum_{k=i}^{n-1} P(\alpha \longrightarrow \beta\alpha)\, e(i,k|\beta)\, h_n(k+1,j|\alpha) \\ \quad \displaystyle + \sum_{k=n}^{j-1} P(\alpha \longrightarrow \alpha\beta)\, h_n(i,k|\alpha) \\ \qquad \times e(k+1,j|\beta) \Big\}, & \text{if } i<j, \\ P(\alpha \longrightarrow w_n), & \text{if } i=j=n, \\ 0, & \text{otherwise,} \end{cases} \tag{6}$$

where e is the inside probability given by (A.2) in the appendix.

2.4.3 Word concatenation probability

In general, as Figure 4 shows, a modifier derived from β can be directly connected with a head derived from α in a summarized sentence. In addition, the modifier can also be connected with each word which modifies the head. The word concatenation probability between w_x and w_y is defined as the sum of the dependency probabilities between w_x and w_y, and between w_x and each of the $w_{y+1} \cdots w_z$. Using the dependency probabilities $d(w_x, w_y, i, k, j)$, the word concatenation score is calculated as the logarithmic value of the word concatenation probability given by

$$T(w_x, w_y) = \log \sum_{i=1}^{x} \sum_{k=x}^{y-1} \sum_{j=y}^{L} \sum_{z=y}^{j} d(w_x, w_z, i, k, j). \tag{7}$$

2.4.4 SDCFG

The SDCFG is constructed using a manually parsed corpus. The SDCFG parameters are estimated using the inside-outside algorithm. In our SDCFG based on Ito et al. [16], we only determined the number of nonterminal symbols and considered all possible phrase trees. We applied rules consisting of all combinations of nonterminal symbols to each rewriting symbol in a phrase tree. The nonterminal symbol in this method is not given a specific function such as that of a noun phrase, and the functions of nonterminal symbols are automatically learned from data. The probabilities for frequently used rules increase and those for rarely used rules decrease. Since words in the learning data for SDCFG are tagged with POS, the dependency probability of words excluded from the learning data can be calculated based on their POS. Even if the transcription results obtained by a speech recognizer are ill-formed, the dependency structure can be robustly estimated by the SDCFG.

2.5. DP for automatic summarization

Given a transcription result consisting of N words, $W = w_1, w_2, \ldots, w_N$, summarization is done by extracting a set of M ($M < N$) words, $V = v_1, v_2, \ldots, v_M$, which maximizes the summarization score given by (1). The algorithm is as follows.

Algorithm 1. (1) *Definition of symbols and variables*

$\langle s\rangle$ is the beginning symbol of sentence, $\langle /s\rangle$ is the ending symbol of sentence, $P(w_n|w_k w_l)$ is the linguistic score, $I(w_n)$ is the word significance score, $C(w_n)$ is the confidence score, $T(w_l, w_n)$ is the word concatenation score, $s(k,l,n)$ is the summarization score of each word $s(k,l,n) = I(w_n) + \lambda_L L(w_n|w_k w_l) + \lambda_C C(w_n) + \lambda_T T(w_l, w_n)$, $g(m,l,n)$ is the summarization score of subsentence $\langle s\rangle, \ldots, w_l, w_n$, consisting of m words, beginning from $\langle s\rangle$ and ending at w_l,w_n $(0 \le l < n \le N)$, $B(m,l,n)$ is the back pointer.

(2) *Initialization*

The summarization score is calculated for each subsentence hypothesis consisting of one word. The value of $-\infty$ is awarded for each word which is never selected as the first word in the summarized sentence consisting of M words,

$$g(1,0,n) = \begin{cases} I(w_n) + \lambda_L L(w_n|\langle s\rangle) + \lambda_C C(w_n), & \textit{if } 1 \le n \le (N-M+1), \\ -\infty, & \textit{otherwise.} \end{cases} \tag{8}$$

(3) *DP process*

DP recursion is applied to each pair of the last two words (w_l, w_n) for each subsentence hypothesis consisting of m words,

```
for $m = 2$ to $M$,
  for $n = m$ to $N - m + 1$,
    for $l = m - 1$ to $n - 1$,
        $g(m,l,n) = \max_{k<l} \{g(m-1,k,l) + s(k,l,n)\}$,
        $B(m,l,n) = \arg\max_{k<l} \{g(m-1,k,l) + s(k,l,n)\}$.
                                                              (9)
```

(4) *Select the optimal path*

The best complete hypothesis consisting of M words is determined by selecting the last two words $(w_{\hat{l}}, w_{\hat{n}})$,

$$\begin{aligned} S(V) &= \max_{\substack{N-M<n\le N \\ N-M-1<l\le N-1}} g(M,l,n) + \lambda_L L(\langle /s\rangle|w_l w_n), \\ (\hat{l}, \hat{n}) &= \arg\max_{\substack{N-M<n\le N \\ N-M-1<l\le N-1}} g(M,l,n) + \lambda_L L(\langle /s\rangle|w_l w_n). \end{aligned} \tag{10}$$

(5) *Backtracking*

We can get the word sequence $V = v_1 \cdots v_M$ with the best summarization result by tracking the back pointers retained in (3),

$$\begin{gathered} \textit{for } m = M \textit{ to } 1, \quad v_m = w_{\hat{n}}, \\ l' = B(m, \hat{l}, \hat{n}), \quad \hat{n} = \hat{l}, \quad \hat{l} = l'. \end{gathered} \tag{11}$$

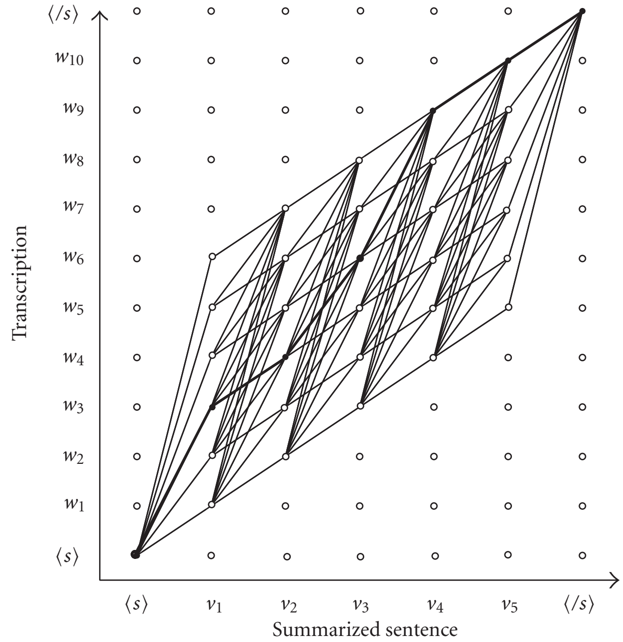


FIGURE 5: Example of DP alignment to summarize an individual utterance.

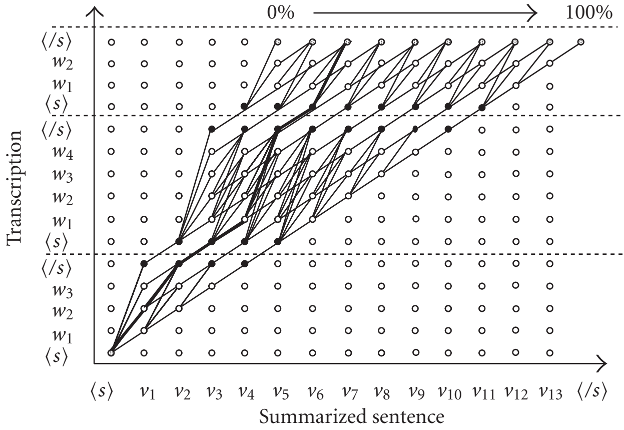


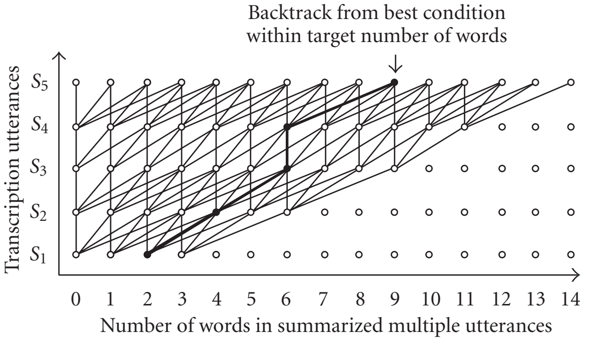
FIGURE 6: Example of DP process to summarize multiple utterances.

Figure 5 shows the two-dimensional space for the DP process. The vertical axis represents the transcription consisting of 10 words ($N = 10$), and the horizontal axis represents the summarized sentence having 5 words ($M = 5$). All possible sets of 5 words extracted from the 10 words are traced by paths from the bottom-left corner to the top-right corner. The path which maximizes the summarization score is selected.

3. SUMMARIZATION OF MULTIPLE UTTERANCES

3.1. Basic algorithm

Our proposed technique to automatically summarize the speech in individual sentences can be extended to summarizing a set of multiple utterances (sentences) by incorporating a rule which provides restrictions at sentence boundaries [10, 17]. In multiple utterances summarization, original sentences including many informative words are preserved, and sentences including few informative words are deleted or shortened. Given the total summarization ratio for multiple utterances, the summarization ratio for each utterance is automatically calculated so that the total score can be maximized. Figure 6 illustrates the DP process for summarizing multiple utterances. This technique incorporates the summarization method, developed in the field of natural language processing to extract important sentences, into our sentence-by-sentence summarization method.



Backtrack from best condition within target number of words
Transcription utterances
S_5
S_4
S_3
S_2
S_1
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
Number of words in summarized multiple utterances

FIGURE 7: Example of two-level DP process to summarize multiple utterances.

3.2. Summarization of multiple utterances using two-level DP

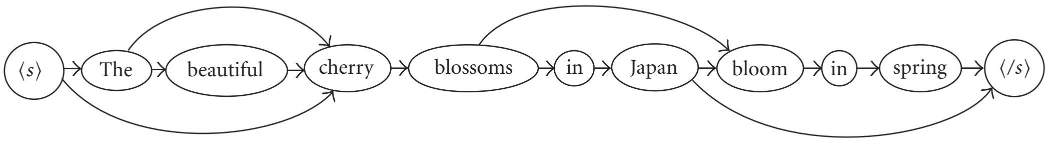
However, the amount of calculation required to select the best combination of all those possible in multiple utterances increases as the number of words in the original utterances increases. To alleviate this problem, we propose a new method in which each utterance is summarized, based on all possible summarization ratios, and then the best combination of summarized sentences for each utterance is determined according to a target compression ratio using a two-level DP technique. Figure 7 illustrates the two-level DP technique for summarizing multiple utterances. The algorithm is as follows.

Algorithm 2. (1) *Definition of symbols and variables*

$s_n(l)$ is the summarization score for a sentence consisting of l words summarized from sentence S_n, $0 \le l \le L_n$, $1 \le n \le N$.

(2) *Initialization*

$$g(1,l) = s_1(l),$$
$$B(1,l) = l, \quad 0 \le l \le L_1, \tag{12}$$
$$M = L_1.$$



The word string most similar to the automatic summarization in the network	⟨s⟩ Cherry blossoms bloom in spring ⟨/s⟩
Automatic summarization of automatic transcription	⟨s⟩ Chill ⟨DEL⟩ bloom in spring ⟨/s⟩
Summarization accuracy	5 − (1 + 0 + 1)/5 ∗ 100 = 60%

Figure 8: Example to calculate summarization accuracy using a word network. The underlined word and ⟨DEL⟩ in automatic summarization represent a substitution error and a deletion error. The summarization accuracy is given by (15).

(3) *DP process*

$$\begin{aligned}
&\textit{for } n = 2 \textit{ to } N,\\
&\quad M = M + L_n,\\
&\quad\quad \textit{for } m = 0 \textit{ to } M,\\
&\quad\quad\quad g(n,m) = \max_{m-L_n \le l \le m,\, l\ge 0} \{g(n-1,l) + s_n(m-l)\},\\
&\quad\quad\quad B(n,m) = \operatorname*{arg\,max}_{m-L_n \le l \le m,\, l\ge 0} \{g(n-1,l) + s_n(m-l)\}.
\end{aligned}\tag{13}$$

(4) *Backtracking*

$$\begin{aligned}
&\textit{for } n = N \textit{ to } 1,\\
&\quad l_n = M - B(n,M),\\
&\quad M = B(n,M),\\
&\quad \textit{for } n = 1 \textit{to } N,\\
&\quad\quad \textit{Output } S_n(l_n).
\end{aligned}\tag{14}$$

4. EVALUATION

4.1. Word network of manual summarization results used for evaluation

Correctly transcribed speech is manually summarized by human subjects and used as correct targets to automatically evaluate summarized sentences. The manual summarization results are merged into a word network which approximately expresses all possible correct summarizations including subjective variations. The summarization accuracy given by (15) is calculated using the word network [10]. The word string that is the most similar to the automatic summarization results extracted from the word network is considered the correct target for automatic summarization. The accuracy, comparing the summarized sentence with the target word string, is a measure of linguistic correctness and retention of the original meanings of the utterance,

$$\begin{aligned}
&\text{Summarization accuracy}\\
&\quad = \frac{\text{Len} - (\text{Sub} + \text{Ins} + \text{Del})}{\text{Len}} \times 100[\%],
\end{aligned}\tag{15}$$

where Sub is the number of substitutions compared with target word string, Ins is the number of insertions compared with target word string, Del is the number of deletions compared with target word string, and Len is the number of words in target word string.

Figure 8 shows an example of calculating summarization accuracy using a word network. In this example, "cherry" is misrecognized as "chill" by the recognition system and is extracted into a summarized sentence. The summarization accuracy is defined by the word accuracy based on the word string extracted from the word network that is most similar to the automatic summarization results.

4.2. Evaluation data

We used the TV news broadcasts in English (CNN news) recorded in 1996 by NIST as a test set for topic detection and tracking (TDT) and tagged it with Brill's tagger (http://www.cs.jhu.edu/∼brill/) to evaluate our proposed method. Five news articles consisting of 25 utterances on average were transcribed by the JANUS [11] speech recognition system. Multiple utterances were summarized in each of the five news articles at summarization ratios of 40% and 70%. Fifty utterances were arbitrarily chosen from the five news articles and used for sentence-by-sentence summarization with the 40% and 70% ratios. The mean word recognition accuracies for the utterances used for multiple utterance summarization and those for sentence-by-sentence summarization were 78.4% and 81.4%, respectively. Seventeen native English speakers generated manual summaries by removing or

Table 1: Examples of automatic summarization and the corresponding target extracted from a manual summarization word network. In each summarization ratio, upper sentence represents a set of words extracted from summarization network which is the most similar to automatic summarization, and lower sentence represents automatic summarization of recognition results. The underlined word in the recognition result is a recognition error. ⟨INS⟩ and ⟨DEL⟩ indicate an insertion error and a deletion error in summarization.

Recognition result	VICE PRESIDENT AL GORE SAYS THE GOVERNMENT HAS A PLAN TO AVOID THE INEVITABLE PROSPECT OF INCREASED AIRPLANE CRASHES AND FATALITY IS
70% summarization	VICE PRESIDENT AL GORE SAYS THE GOVERNMENT HAS A PLAN TO AVOID THE INCREASED AIRPLANE CRASHES
	VICE PRESIDENT AL GORE SAYS THE GOVERNMENT HAS A PLAN TO AVOID <DEL> INCREASED AIRPLANE CRASHES
40% summarization	<INS> THE GOVERNMENT HAS A PLAN TO AVOID THE INCREASED AIRPLANE CRASHES
	GORE THE GOVERNMENT HAS A PLAN TO AVOID THE INCREASED AIRPLANE CRASHES

extracting words, and they were merged to build word networks.

4.3. Structure of transcription system

The English news broadcasts were transcribed under the following conditions.

4.3.1 Feature extraction

Sounds were digitized at 16-kHz sampling and 16-bit quantization. Feature vectors had 13 elements consisting of MFCC. Vocal Tract Length Normalization (VTLN) and cluster-based cepstral mean normalization were used to compensate for speakers and channels. Linear Discriminant Analysis (LDA) was applied to produce a 42-dimensional vector from a set of features in each segment consisting of 7 frames.

4.3.2 Acoustic model

We used a pentphone model with 6000 distributions sharing 2000 codebooks. There were about 105-k Gaussians in the system. The training data was composed of 66 hours of BN.

4.3.3 Language model

The bigram and trigram were constructed using a BN corpus with a vocabulary of 40 k.

4.3.4 Decoder

A word-graph-based 3-pass decoder was used for transcription. In the first pass, a frame-synchronous beam search was conducted using a tree-based lexicon, the above-mentioned hidden Markov models (HMMs) and a bigram model to generate a word graph. In the second pass, a frame-synchronous beam search was conducted again using a flat lexicon hypothesized in the word graph by the first pass and a trigram model. In the third pass, the word graph was minimized and rescored using the trigram language model.

4.4. Training data for summarization models

A word significance model, a bigram language model, and SDCFG were constructed using approximately 35-M words (10681 sentences) from the Wall Street Journal corpus and the Brown corpus in the Penn Treebank (http://www.cis.upenn.edu/~treebank/).

4.5. Evaluation results

We summarized both manual transcription (TRS) and automatic transcription (REC). Table 1 shows examples of automatic summarization and the corresponding target extracted from a manual summarization word network. Figure 9 shows summarization accuracies of utterance summarizations at 40% and 70% summarization ratios, and Figure 10 shows those for summarizing articles with multiple utterances at 40% and 70% summarization ratios. In these figures, I, L, C, and T indicate, word significance scores, linguistic scores, confidence scores, and word concatenation scores, respectively. We compared conditions with and without the word confidence score ($I\_L\_C\_T$) and (I_L_T) in the REC summarization. To summarize both TRS and REC, we compared conditions with and without the word concatenation score (I_L_T, $I\_L\_C\_T$) and (I_L, I_L_C).

The summarization accuracies for manual summarization (SUB) were considered to be the upper limit for automatic summarization accuracy. To ensure that our method was sound, we produced randomly generated summarized sentences (RDM) according to the summarization ratio and compared them with those we obtained with our proposed method.

These results indicated that our proposed automatic speech summarization technique is significantly more effective than RDM. By using the word concatenation score (I_L_T, $I\_L\_C\_T$), changes in meaning were reduced compared with when it was not used (I_L, I_L_C). The results obtained when using the word confidence score ($I\_L\_C\_T$) compared with when it was not used (I_L_T) indicate that summarization accuracy is improved by the confidence score. Table 2 shows the number of word errors and the number of sentences including word errors in the automatic summarization. Recognition errors are effectively reduced by the confidence score.

Table 2: Number of recognition errors in summarized sentences ($(\cdot)$ is the number of sentences including recognition errors).

	Individual utterance		Multiple utterances	
REC	180(45)		326(94)	
Summarization ratio	40%	70%	40%	70%
I	42 (27)	111 (40)	99 (56)	199 (71)
I_L	44 (28)	87 (37)	86 (53)	166 (69)
I_L_C	23 (15)	49 (22)	34 (28)	82 (47)
I_L_T	46 (27)	84 (37)	90 (56)	173 (69)
$I\_L\_C\_T$	22 (13)	51 (24)	25 (17)	80 (47)
RDM	82 (30)	87 (21)	89 (45)	169 (65)

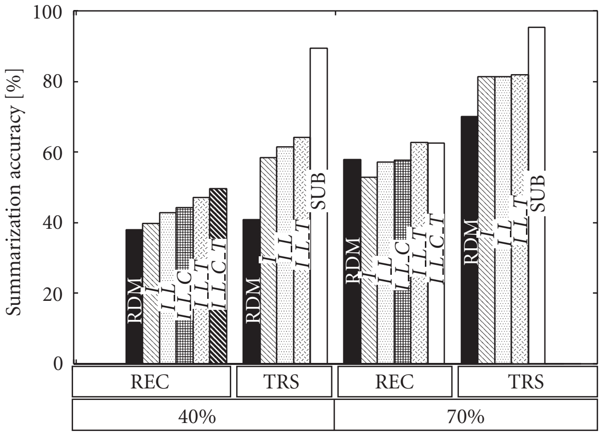


Figure 9: Individual utterance summarization at 40% and 70% summarization ratios. REC: summarization of recognition results, TRS: summarization of manual transcriptions, RDM: random word selection, C: confidence score, I: significance score, L: linguistic score, I_L: combination of 2 scores, I_L_C, I_L_T: combination of 3 scores, $I\_L\_C\_T$: combination of all scores, and SUB: subjective summarization.

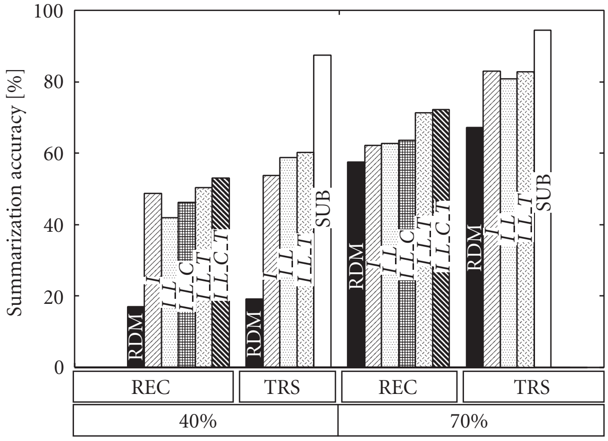


Figure 10: Article summarization at 40% and 70% summarization ratios. REC: summarization of recognition results, TRS: summarization of manual transcriptions, RDM: random word selection, C: confidence score, I: significance score, L: linguistic score, I_L: combination of 2 scores, I_L_C, I_L_T: combination of 3 scores, $I\_L\_C\_T$: combination of all scores, and SUB: subjective summarization.

5. CONCLUSIONS

Individual utterances and a whole news article consisting of multiple utterances taken from English news broadcasts were summarized by our automatic speech summarization method based on the following: word significance score, linguistic likelihood, word confidence measure, and word concatenation probability. The experimental results revealed that our method can effectively extract relatively important information and remove redundant and irrelevant information from English news broadcasts in the same way as it does in Japanese news broadcasts.

In contrast with the confidence score which was incorporated into the summarization score to exclude word errors by the recognizer, the linguistic score effectively reduces out-of-context word extraction both from recognition errors and human disfluencies. In summarizing the speech of Japanese news broadcasters, the confidence measure improved summarization by excluding in-context word errors. In the English case, the confidence measure not only excluded word errors, but also helped extract clearly pronounced important words. Consequently, the use of the confidence measure yielded a larger increase in the summarization accuracy for English than it did for Japanese.

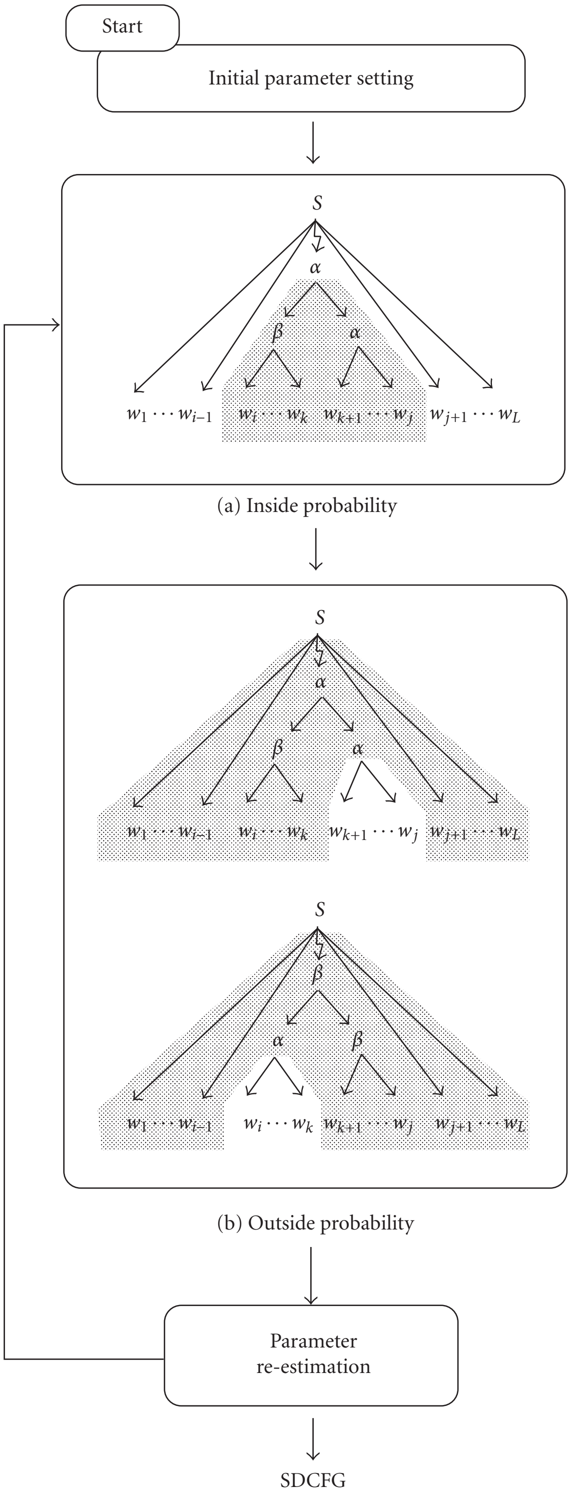
APPENDIX

PARAMETER RE-ESTIMATION IN SDCFG

The parameters of SDCFG for languages with both right and left dependency structures are estimated from a manual-parsed corpus using the inside-outside algorithm. Suppose that a sentence consists of L words,

$$S \longrightarrow w_1 \cdots w_i \cdots w_L, \tag{A.1}$$

where L is the number of words in a sentence and w_i is the ith word in a sentence.



(a) Inside probability

(b) Outside probability

FIGURE 11: Estimation algorithm for SDCFG.

The rewrite probabilities of $\alpha \rightarrow \beta\alpha$ and $\alpha \rightarrow w$ are denoted by $P(\alpha \rightarrow \beta\alpha)$ and $P(\alpha \rightarrow w)$, respectively. The algorithm for estimating the parameters of the SDCFG is described below. Figure 11 lists the estimation steps.

Algorithm A.3. (1) *Initialization*

$P(\alpha \rightarrow \beta\alpha)$ and $P(\alpha \rightarrow \alpha\beta)$ are given a flat probability and $P(\alpha \rightarrow w)$ is given random values.

(2) *Calculation of the inside probability*

The inside probability in Figure 11(a) is calculated as follows:

$$e(i, j|\alpha) = P(\alpha \longrightarrow w_i \cdots w_j) = \begin{cases} \displaystyle\sum_{k=i}^{j-1} \Big\{ \sum_{\beta} P(\alpha \longrightarrow \beta\alpha) e(i, k|\beta) e(k+1, j|\alpha) \\ \qquad + \displaystyle\sum_{\beta:\alpha\neq\beta} P(\alpha \longrightarrow \alpha\beta) e(i, k|\alpha) \\ \qquad\qquad \times e(k+1, j|\beta) \Big\}, \quad \text{if } i < j, \\ P(\alpha \longrightarrow w_i), \quad \text{if } i = j. \end{cases} \tag{A.2}$$

(3) *Calculation of the outside probability*

The outside probability in Figure 11(b) is calculated as follows:

$$\begin{aligned} f(i, j|\alpha) &= P(w_1 \cdots w_{i-1} \alpha w_{j+1} \cdots w_L) \\ &= \sum_{k=1}^{i-1} \Big\{ \sum_{\beta} P(\alpha \longrightarrow \beta\alpha) e(k, i-1|\beta) f(k, j|\alpha) \\ &\qquad + \sum_{\beta:\alpha\neq\beta} P(\beta \longrightarrow \beta\alpha) e(k, i-1|\beta) f(k, j|\alpha) \Big\} \\ &\quad + \sum_{k=j+1}^{L} \Big\{ \sum_{\beta} P(\beta \longrightarrow \alpha\beta) e(j+1, k|\beta) f(i, k|\alpha) \\ &\qquad + \sum_{\beta:\alpha\neq\beta} P(\alpha \longrightarrow \alpha\beta) e(j+1, k|\beta) f(i, k|\alpha) \Big\}. \end{aligned} \tag{A.3}$$

(4) *Estimate of parameters*

The parameters are re-estimated, using the probabilities obtained through steps (2) to (3),

$$\begin{aligned} \hat{P}(\alpha \longrightarrow \beta\alpha) &= \frac{\sum_{i=1}^{L-1} \sum_{j=i+1}^{L} \sum_{k=i}^{j-1} g(i, k, j; \alpha \longrightarrow \beta\alpha)}{e(1, L|S)}, \\ \hat{P}(\alpha \longrightarrow w_c) &= \frac{\sum_{i=1}^{L} P(\alpha \longrightarrow w) f(i, j|\alpha)}{e(1, L|S)}, \end{aligned} \tag{A.4}$$

where

$$\begin{aligned} g(i, k, j; \alpha \longrightarrow \beta\alpha) &= e(i, k|\beta) e(k+1, j|\alpha) \\ &\quad \times P(\alpha \longrightarrow \beta\alpha) f(i, j|\alpha), \\ g(i, k, j; \alpha \longrightarrow \alpha\beta) &= e(i, k|\alpha) e(k+1, j|\beta) \\ &\quad \times P(\alpha \longrightarrow \alpha\beta) f(i, j|\alpha). \end{aligned} \tag{A.5}$$

(5) *Iteration*

Steps from (2) to (4) are iterated until the parameters are saturated.

ACKNOWLEDGMENT

The authors would like to thank Dr. Yoshi Gotoh (Sheffield University) for an arrangement of generating the correct answer for automatic summarization.

REFERENCES

[1] R. Valenza, T. Robinson, M. Hickey, and R. Tucker, "Summarization of spoken audio through information extraction," in *Proc. ESCA Workshop on Accessing Information in Spoken Audio*, pp. 111–116, Cambridge, UK, 1999.

[2] Z. Klaus, "Automatic generation of concise summaries of spoken dialogues in unrestricted domains," in *Proc. 24th ACM SIGIR International Conference on Research and Development in Information Retrieval*, New Orleans, La, USA, September 2001.

[3] S. Furui, K. Maekawa, H. Isahara, T. Shinozaki, and T. Ohdaira, "Toward the realization of spontaneous speech recognition-introduction of a Japanese priority program and preliminary results," in *Proc. International Conference on Spoken Language Processing (ICSLP2000)*, vol. 3, pp. 518–521, Beijing, China, 2000.

[4] F. R. Chen and M. M. Withgott, "The use of emphasis to automatically summarize a spoken discourse," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 1, pp. 229–232, San Fransisico, Calif, USA, March 1992.

[5] S. Kobayashi, N. Yoshikawa, and S. Nakagawa, "Extracting summarization of lectures based on linguistic surface and prosodic information," IPSJ Technical Report SIG-SLP-43-7, Toyohashi University of Technology, Japan, 2002.

[6] I. Mani and M. T. Maybury, *Advances in Automatic Text Summarization*, MIT Press, Cambridge, Mass, USA, 1999.

[7] K. Knight and D. Marcu, "Statistics-based summarization—step one: sentence compression," in *Proc. 17th National Conference on Artificial Intelligence (AAAI-00)*, Austin, Tex, USA, August 2000.

[8] C. Hori and S. Furui, "Automatic speech summarization based on word significance and linguistic likelihood," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 3, pp. 1579–1582, Istanbul, Turkey, 2000.

[9] C. Hori and S. Furui, "Improvements in automatic speech summarization and evaluation methods," in *Proc. 6th International Conference on Spoken Language Processing (ICSLP2000)*, vol. 4, pp. 326–329, Beijing, China, 2000.

[10] C. Hori and S. Furui, "Advances in automatic speech summarization," in *Proc. 7th European Conference on Speech Communication and Technology (Eurospeech)*, vol. 3, pp. 1771–1774, Aalborg, Denmark, 2001.

[11] A. Waibel et al., "Advances in meeting recognition," in *Proc. 1st International Conference on Human Language Technology Conference (HLT 2001)*, pp. 11–13, San Diego, Calif, USA, March 2001.

[12] T. Kemp and T. Schaaf, "Estimating confidence using word lattices," in *Proc. 5th European Conference on Speech Communication and Technology (Eurospeech)*, vol. 2, pp. 827–830, Rhodes, Greece, September 1997.

[13] V. Valtchev, J. Odell, P. Woodland, and S. Young, "MMIE training of large vocabulary recognition systems," *Speech Communication*, vol. 22, no. 4, pp. 303–314, 1997.

[14] C. Manning and H. Schutze, *Foundations of Statistical Natural Language Processing*, MIT Press, Cambridge, Mass, USA, 1999.

[15] K. Lari and S. J. Young, "The estimation of stochastic context-free grammars using the inside-outside algorithm," *Computer Speech & Language*, vol. 4, no. 1, pp. 35–56, 1990.

[16] A. Ito, C. Hori, M. Katoh, and M. Kohda, "Language modeling by stochastic dependency grammar for Japanese speech recognition," in *Proc. 6th International Conference on Spoken Language Processing (ICSLP2000)*, vol. 1, pp. 246–249, Beijing, China, 2000.

[17] C. Hori and S. Furui, "A new approach to automatic speech summarization," to appear in the IEEE Trans. Multimedia.

Chiori Hori received the B.E. and the M.E. degrees in electrical and information engineering from Yamagata University, Yonezawa, Japan in 1994 and 1997, respectively. From April 1997 to March 1999, she was a Research Associate in the Faculty of Literature and Social Sciences, Yamagata University. In April 1999, she started the doctoral course in the Graduate School of Information Science and Engineering at Tokyo Institute of Technology (TITECH) and received her Ph.D. degree in March 2002. She is currently a Researcher in NTT Communication Science Laboratories (CS Labs) at Nippon Telegraph and Telephone Corporation (NTT), Kyoto, Japan in 2002. She is a member of the IEEE, the Acoustical Society of Japan (ASJ), and the Institute of Electronics, Information and Communication Engineers of Japan (IEICE).

Sadaoki Furui is currently a Professor at the Department of Computer Science, Tokyo Institute of Technology. He is engaged in a wide range of research on speech analysis, speech recognition, speaker recognition, speech synthesis, and multimodal human-computer interaction and has authored and coauthored over 400 published articles. He is a Fellow of the IEEE, the Acoustical Society of America, and the Institute of Electronics, Information and Communication Engineers of Japan (IEICE). He is President of the Acoustical Society of Japan (ASJ), the International Speech Communication Association (ISCA), and the Permanent Council for International Conferences on Spoken Language Processing (PC-ICSLP). He is a Board of Governor of the IEEE Signal Processing Society. He is Editor-in-Chief of the Transaction of the IEICE and has served as Editor-in-Chief of Speech Communication. He has received the Yonezawa Prize and the Paper Award from the IEICE (1975, 1988, 1993) and the Sato Paper Award from the ASJ (1985, 1987). He has received the Senior Award from the IEEE ASSP Society (1989) and the Achievement Award from the Minister of Science and Technology, Japan (1989). He has received the Book Award from the IEICE (1990). In 1993, he served as an IEEE SPS Distinguished Lecturer.

Robert Malkin received the B.S. degree in computational linguistics and the Master of Language Technologies, both from Carnegie Mellon University, in 1996 and 1998, respectively. He is currently a Ph.D. candidate at Carnegie Mellon's Language Technologies Institute. Mr. Malkin's research interests include computational auditory scene analysis, machine perception, and speech recognition.

Hua Yu received his B.S. and M.S. degrees in computer science, Tsinghua University, China in 1994 and 1996, respectively. He is now a Ph.D. candidate in the School of Computer Science, Carnegie Mellon University, working on recognition of conversational speech. His research interest includes speech recognition, pattern recognition, and language technologies in general. He is a student member of the IEEE and the ACM.

Alex Waibel is a Professor of computer science at Carnegie Mellon University, Pittsburgh and at the University of Karlsruhe (Germany). He directs the Interactive Systems Laboratories (www.is.cs.cmu.edu) at both universities with research emphasis in speech recognition, handwriting recognition, language processing, speech translation, machine learning, and multimodal and multimedia interfaces. At Carnegie Mellon, he also serves as Associate Director of the Language Technology Institute and as Director of the Language Technology Ph.D. program. He was one of the founding members of the CMU's Human Computer Interaction Institute (HCII) and continues on its core faculty. Dr. Waibel was one of the founders of C-STAR, the international consortium for speech translation research and served as its chairman from 1998 to 2000. His team has developed the JANUS speech translation system, the JANUS speech recognition toolkit, and a number of multimodal systems including the meeting room, the Genoa Meeting recognizer and meeting browser. Dr. Waibel received the B.S. in Electrical Engineering from the Massachusetts Institute of Technology in 1979, and his M.S. and Ph.D. degrees in Computer Science from Carnegie Mellon University in 1980 and 1986. His work on the Time Delay Neural Networks was awarded the IEEE best paper award in 1990; his work on multilingual and speech translation systems the "Alcatel SEL Research Prize for Technical Communication" in 1994, the "Allen Newell Award for Research Excellence" from CMU in, 2002 and the Speech Communication Best Paper Award in 1992.