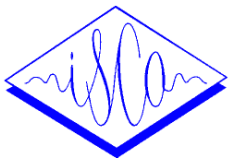


First International Conference on
Spoken Language Processing (ICSLP 90)
Kobe, Japan
November 18-22, 1990

SENTENCE SPEECH RECOGNITION USING SEMANTIC DEPENDENCY ANALYSIS

Shoichi Matsunaga and *Shigeki Sagayama*

NTT Human Interface Laboratories
Midori-cho, Musashino-shi, Tokyo 180 Japan

ATR Interpreting Telephony Laboratories
Seika-cho, Souraku-gun, Kyoto 619-02 Japan

ABSTRACT

This paper describes a sentence speech recognition system based on phoneme-based hidden Markov models (HMMs) and two grammatical constraints: a syntactic grammar of phrase structure and a semantic dependency grammar of sentence structure. A joint score, combining acoustic likelihood and linguistic certainty factors derived from phoneme based HMMs and two grammatical constraints, is maximized to obtain the optimal sentence recognition. A semantic analysis algorithm globally optimizes the joint score. This algorithm is based on two key techniques: most likely multi-phrase candidate-detection using the Viterbi algorithm, and breadth-first search for dependency parsing. Where the perplexity of the phrase syntax is 40, this system increases phrase recognition performance in the sentences by approximately 14%, showing the effectiveness of semantic dependency analysis.

I. INTRODUCTION

In Japanese sentences, the phrase order is much less constrained than in English. On the other hand, the word order within phrases, which are short sequences of words, is very regular, and the sentence structure is ordered by semantic dependency between phrases. Syntactic constraints are useful in recognizing specific tasks or short-duration utterances. However, particularly in sentence recognition for phrase-order-free languages such as Japanese, semantic constraints are more powerful than sentence syntactic constraints.

We have developed a Japanese continuous speech recognition system that obtains the most likely sentence results taking account of acoustic, syntactic, and semantic factors based on a two-level grammar approach. This approach uses two grammars: an intra-phrase transition network grammar for phrase recognition, and an inter-phrase dependency grammar for sentence recognition. The former is a syntactic grammar, and the latter is a semantic and loose syntactic grammar. The dependency grammar is compatible with the case grammar, and has robustness against missing or misrecognized words.

In ICASSP-90, we reported the effectiveness of the semantic dependency analysis approach combined with phoneme-based HMMs for speech uttered phrase-by-phrase[1]. This paper describes an extended algorithm for sentence utterances, based on two key techniques: detection of most-likely multi-phrase candidates using the Viterbi algorithm, and breadth-first dependency parsing using dynamic programming.

To concisely cover a variety of phrase structures, the syntactic structure within phrases is represented by transition networks. Taking account of pauses between phrases, syntactic constraints of sentences are represented by multiconnection of these networks. The network parser frame-synchronously parses input sentence utterances to get a lattice of the multi-phrase candidates and their likelihoods for the selected phrase boundaries extracted by the Viterbi algorithm.

The dependency parser analyzes inter-phrase dependency structures within a sentence. Semantic certainty factors are determined by taking into account grammatical cases incorporated in word dictionaries. The joint score, obtained by combining accumulated phonetic likelihood and the semantic certainty factors derived from the dependency grammar, is maximized to obtain the optimal solution. The dependency parser utilizes efficient breadth-first-search and beam-search algorithms.

The approach described here is very suitable for speech understanding systems since it can use semantic dependency structures.

II. SPEECH RECOGNITION SYSTEM USING DEPENDENCY GRAMMAR

Figure 1 shows a block diagram of the system. Inputs are sentence utterances. After feature-parameter extraction, the parameter sequence is converted into a vector code sequence. Phrase likelihood is then calculated frame-synchronously using phoneme-based HMMs and transition networks. This calculation is based on the Viterbi algorithm, and yields multi best phrase sequences for each frame whose last phrase is different from that of other sequences. The last phrase of the sequences for each end-frame are stored in lattice form. This lattice is reduced based on Viterbi phrase segmentation. Finally, using this lattice and the dependency grammar, the parser extracts the most likely sentence of a phrase sequence and its dependency structure. The verb and adjective entries in the dictionary have grammatical cases, and the noun entries are accompanied by semantic primitives.

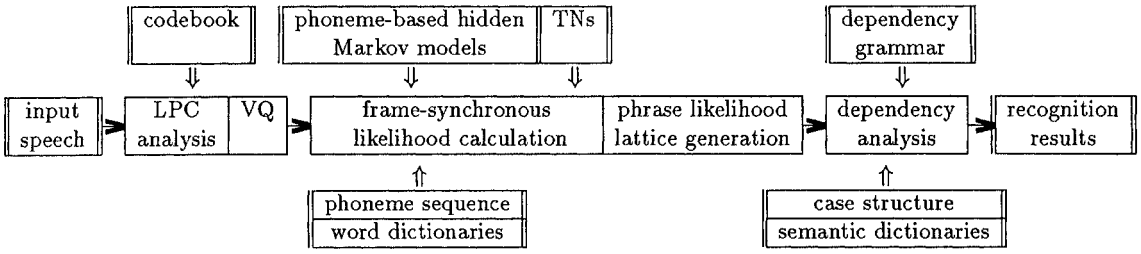


Figure 1. Block Diagram of the Sentence Speech Recognition System

III. SYNTACTIC GRAMMAR

3.1 Syntactic Constraints of Phrases

Japanese phrases are composed of a stem part and a suffix part. The stem part is a verb, a noun, or an adjective. The suffix part is composed of suffixes such as auxiliary verbs and particles. Although connection of these words is very regular, there are many kinds of connection rules. Thus, to concisely cover this variety, syntactic structure is represented by transition networks (TNs) composed of sub-networks.

3.2 Speech Recognition of Phrase Sequences

The syntactic constraints of sentences are ordered by connection of phrase networks. When the frame-synchronous parser parses the input from left to right using phoneme-based HMMs and networks based on the Viterbi algorithm, this calculation generates M-best phrase sequences whose last phrases are different from each other. The last phrases are stored in lattice form. In other words, there are M-phrases for each end-frame and the total number of phrase candidates in the lattice is approximately NM, if input is N-frame speech. After generation of the whole lattice, significant phrase boundaries, which make consecutive phrase sequence, are back-traced from the end of speech to the beginning using Viterbi phrase segmentation. Phrase boundary candidates are then selected and a reduced lattice is generated. A simple example, where M is 2 and N is 6, is shown in Figure 2. The left side of this figure is an original lattice. Three sets of (t_1, t_2, t_6), (t_1, t_5, t_6) and (t_1, t_2, t_5, t_6) can generate consecutive phrase sequences, and the new reduced lattice is on the right.

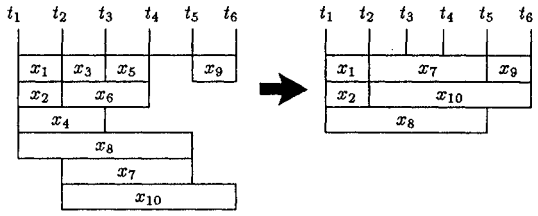


Figure 2. An example of a reduced phrase lattice

IV. INTER-PHRASE DEPENDENCY GRAMMAR

4.1 Dependency Grammar

Dependency grammar is based on semantic dependency relationships between phrases. The syntactic rules satisfy the only two constraints. First, every phrase except the last must modify one and only one later phrase. This modification is called a dependency relationship or dependency structure. Second, no modification relationship between phrases in the sentence cross.

The semantic certainty factors of the dependency structure are easily provided using grammatical cases. There are two kinds of factors. One is associated only with dependency relationships of the modifier and modificant phrases: agreement between the semantic primitive of the modifier and that required by the modificant, agreement between the modifier case and that required by the modificant, idiomatic expressions, and so on. The other factor is associated with all the dependency structures of the phrase sequence: a phrase with the obligatory case required by the modificant, no modification of the same phrase by different phrases having the same case, simplicity of the sentence structure, and so on. The certainty factor values for these items are given heuristically.

4.2 Parser for Dependency Structure

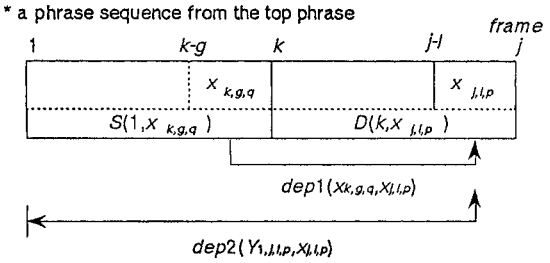
Consecutive phrases, whose number is H, of a sentence on the phrase lattice are given as follows.

$$x_{j_1,l_1,p_1} \cdots x_{j_h,l_h,p_h} \cdots x_{j_H,l_H,p_H}$$

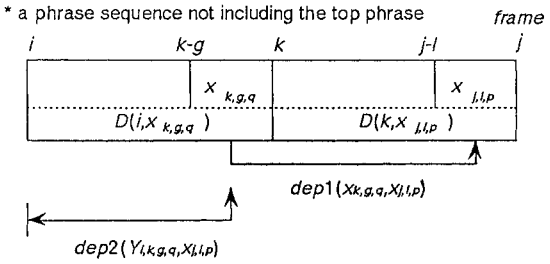
where $1 \leq h \leq H$, $1 \leq j \leq N$, $1 \leq p \leq M$, N is the number of frames of the input sentence, and M is the number of phrase candidates of each segment. The abbreviation of j_h, l_h, p_h is j, l, p. The term $x_{j,l,p}$ is a candidate of the $j-l$-th to j-th frame with the p-th best likelihood.

This parsing is equivalent to solving the following objective function using the constraints of dependency structure grammar.

$$T = \max_{H,\{p\}}[\sum_{h=1}^{H} c(x_{j,l,p}) + \max_{Y} \sum_{h=1}^{H} dep(\boldsymbol{w}_{1,j}, x_{j,l,p}|Y_{1,j,l,p})] \quad (1)$$



$$S(1, x_{j,l,p}, r) = r^{th} \max_{k,g,q,r1,r2} [S(1, x_{k,g,q}, r1) + D(k, x_{j,l,p}, r2) + dep1(x_{k,g,q}, x_{j,l,p}) + dep2(Y_{1,j,l,p}, x_{j,l,p})]$$



$$D(i, x_{j,l,p}, r) = r^{th} \max_{k,g,q,r1,r2} [D(i, x_{k,g,q}, r1) + D(k, x_{j,l,p}, r2) + dep1(x_{k,g,q}, x_{j,l,p}) + dep2(Y_{i,k,g,q}, x_{k,g,q})] \quad (i \neq 1)$$

Figure 3. Illustration of deriving the recurrence relation among the objective function $i \leq k \leq j-1$, $i+g \leq g \leq j-l$, $1 \leq q \leq M$, and $1 \leq r, r1, r2 \leq L$. $r, r1$ and $r2$ indicate the beam ranks, L is the maximum number of beams, $S(1, x_{j,l,p}, r)$ and $D(i, x_{j,l,p}, r)$ are the r-th value of the element whose phrase sequence is $X_{i,j,l,p}$, and the dependency structure is $Y_{i,j,l,p}$. Here, r^{th} max[·] is a function for deriving the r-th best value.

where $c(x_{j,l,p})$ is its log-likelihood. A phrase sequence with one phrase candidate for i-th to j-th frame and whose last phrase $x_{j,l,p}$ is denoted by $X_{i,j,l,p}$. The term $Y_{i,j,l,p}$ is one of the dependency structures of $X_{i,j,l,p}$, and $\boldsymbol{w}_{i,j}$ is the set of phrases that modify $x_{j,l,p}$ in the sequence $X_{i,j,l,p}$. Here, $dep(\boldsymbol{w}, x|Y)$ is the linguistic certainty factor of dependency relationships between $\boldsymbol{w}$ and x taking Y into account. The first item of the term on the right in Eq.(1) is the summation of phonetic likelihoods of the hypothesized sentence composed of its phrase sequence, and the second item is the summation of linguistic certainty factors. Maximizing Eq.(1) gives the sentence and its dependency structure as the speech recognition result.

To solve Eq.(1) effectively, a fast parsing-algorithm using breadth-first-search and beam-search was developed, based on the fundamental algorithms [2,3,4].

The breadth-first algorithm is formulated as follows. First, $dep(\boldsymbol{w}, x|Y)$ is divided into two terms:

$$dep(\boldsymbol{w}_{1,j}, x_{j,l,p}|Y_{1,j,l,p}) = \sum_{x \in w_{1,j}} dep1(x, x_{j,l,p}) + dep2(Y_{1,j,l,p}, x_{j,l,p}) \quad (2)$$

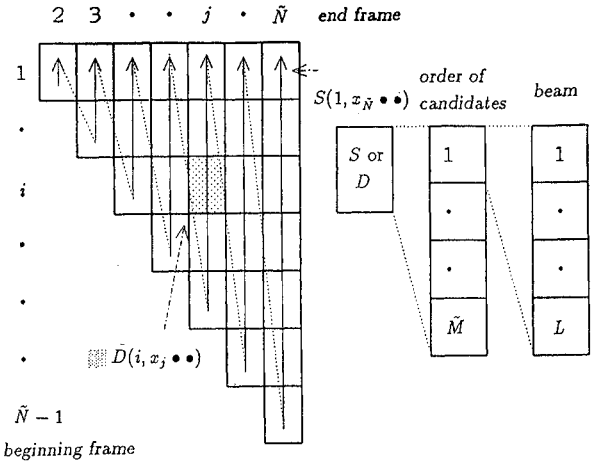


Figure 4. Configuration of parsing table

where $dep1$ is the certainty factor associated with dependency relationships of only the modifier and modificant phrases, and $dep2$ is the certainty factor associated with $Y_{1,j,l,p}$. This algorithm uses beam-search to recursively derive $S(1, x_{j,l,p})$, the objective function's value of a phrase sequence from the top phrase to phrase $x_{j,l,p}$, as well as $D(i, x_{j,l,p})$, the value of a phrase sequence not including the top phrase ($i \neq 1$) as shown in Figure 3. In these equations, each dependency structure for evaluation of $dep2$ are different.

When the equations is calculated, $Y_{i,j,l,p}$ is stored for later use evaluating $dep2$. Initial values are given using phonetic likelihood based on HMMs. After calculating the recurrence relation, the value of the objective function is obtained:

$$T = \max_{l,p} [S(1, x_{N,l,p}, 1)] \quad (3)$$

where $1 \leq p \leq M$ and $1 \leq l \leq N$. The best sentence and its dependency structure are given through $Y_{1,N,l,p}$ where p and l maximize Eq.(3). The processing amount order for this algorithm is $O(N^5 M^2 L^2)$.

4.3 Further Reduction of Processing

After selection of phrase boundaries, the reduced phrase lattice is generated. In this case, the parsing algorithm is extended as follows. Loops for frame length N and for the number of fixed-length phrase candidates M are reduced to one loop for the number of free-length candidates $\tilde{M}$. Then, the processing amount order for this extended algorithm is $O(\tilde{N}^3 \tilde{M}^2 L^2)$. If $\tilde{N} = N/10$, $\tilde{M} = M$, and the the length of the input sentence is about 310 frames, the parser reduces computation to approximately 10^{-8} of the amount for the original parser. Figure 4 shows the parsing table, with the first row corresponding to S, and others to D. The phrase sequence for the input sentence corresponds to the right-most top cell. Each cell is composed of $\tilde{M}$ sub-cells for the number

of candidates, and each sub-cell is composed of L sub-cells for the beam-width. Arrows show the sequence for calculating the recurrence relation.

V. SPEECH RECOGNITION EXPERIMENTS

Input utterances were sampled at a rate of 12 kHz. One frame was extracted every 10ms with a 30ms Hamming window and converted into 34 acoustic feature parameters: power, 16 LPC cepstra, Δ power, and 16 Δ LPC cepstra.

In the training process, 3 training sets were used: 216 phonetically balanced words, 5240 words, and combination of 216 words and 29 sentences. These utterances were manually labeled using 25 phoneme symbols including silence. Each phoneme was modeled by HMM and had 4 states and 7 transition paths. The parameter sequence was converted into a vector code sequence and a vector codebook composed of 256 prototype vectors was generated. Each HMM was trained using the forward-backward algorithm, and the code sequence for training was cut out based on the phonetic labels. Output probabilities were floored after training.

In the testing process, acoustic feature parameters of the input were generated in the same manner. These parameters were converted into a code sequence using the same speaker's codebook generated in the training process.

Talker-dependent preliminary word recognition tests were carried out with 216 words uttered by two speakers, one male and one female. The system attained a word recognition rate of 99.5%.

Preliminary recognition tests were then performed on 100 sentences (including 668 phrases in an essay) uttered phrase-by-phrase by the same speakers. The word dictionary had 360 entries and the perplexity of the phrase syntax was 40. Certainty factors of dependency relationships were empirically determined[5] through the analysis of technical literature. If the trainig set is 216 words, the system attained a phrase recognition rate of 85.8% using only the intra-phrase syntactic parser. The dependency parser increased this rate to 90.8%.

Finally, sentence speech recognition was tested for 71 sentences (including 418 phrases) uttered by the same speakers. The maximum number of each phrase segment $\tilde{M}$ was 5, and the beam-width, L was 8. Word dictionary, word perplexity, and certainty factors were the same as those of the preceding tests. Table 1 shows that the dependency parser increased a phrase recognition rate of 69.2% to 83.1%, if the training set was the combination of 216 words and 29 sentences. These results show the effectiveness of semantic dependency analysis.

VI. CONCLUSION

This paper described a Japanese continuous speech recognition system using an intra-phrase transition network grammar and an inter-phrase dependency grammar. The best sentence was efficiently determined from input utterances using a frame-synchronous network parser and a breadth-first dependency parser. Recognition experiments showed the effectiveness of the inter-phrase dependency grammar.

Further development is currently in progress to refine phoneme-based models based on continuous HMMs that take context dependency into account.

ACKNOWLEDGMENT

The authors wish to express their appreciation to Sadaoki Furui of NTT Labs and to Masaki Kohda of the University of Yamagata for their invaluable discussions. The authors also thank Shigeru Homma of NTT for his acoustic processor, without which this work would not have been possible.

REFERENCES

[1] S.Matsunaga, et al "A continuous speech recognition system based on a two-level grammar approach", *Proc. ICASSP*, pp.589-592 , 1990, Albquerque

[2] S.Matsunaga and M.Kohda, "Post-processing using dependency structure of inter-phrases for speech recognition." 1-1-23, *Proc. ASJ annual meeting*, pp.45-46, Mar.1986 (in Japanese)

[3] K.Ozeki,"A multi-stage decision algorithm to select optimum kakariuke structures from bunsetsu lattice." *Trans.*, IEICE Japan, J70-D,12, pp.2621-2629, Dec.1987 (in Japanese)

[4] M.Kohda,"An algorithm for optimum selection of phrase sequence from phrase lattice." *Paper Tec. Group*, IECE Japan, NLC86-14, pp.9-16, Dec.1986 (in Japanese)

[5] S.Matsunaga and M.Kohda,"Linguistic processing using a dependency structure grammar for speech recognition and understanding.", *Proc. COLING*, pp.402-407, 1988, Budapest

Table 1. Speech recognition results

test data	216 words	668 phrases		71 sentences (418 phrases)					
HMM training data	216 words	216 words		216 words		5240 words		216 words + 29 sent.	
dep. analysis	–	without	with	without	with	without	with	without	with
recog. rate(SAG)	99.5	83.1	89.4	45.5	55.3	49.0	64.6	63.6	78.9
recog. rate(TAK)	99.5	88.6	92.3	54.5	75.4	–	–	74.8	87.3
average	99.5	85.8	90.8	50.0	65.3	–	–	69.2	83.1