

TR-I-0110

SL-TRANSにおける、  
文節候補選択処理、問合せ処理

Selecting the Phrase Candidates  
in SL-TRANS

柿ヶ原 康二\*      桶谷 紀幸\*\*      泉 昌明\*\*\*

Koji KAKIGAHARA    Noriyuki OKEYA    Masaaki IZUMI

1989.9

概要

電話による、異なる言語間のコミュニケーションをはかる自動翻訳電話システムを実現するには、音声認識、機械翻訳、音声合成の3つの技術の確立が必要である。音声言語日英翻訳実験システムSL-TRANSは、これらの技術を計算機上で実現したものであり、音声認識を行う部分、言語処理を行う部分、音声処理と言語処理を統合する部分の3つのサブシステムから構成されている。

本稿では、SL-TRANSのシステムの中で音声認識結果をかな漢字に変換する処理、文節ラティスの候補を絞り込む処理、文節候補が誤認識の可能性が高い場合に問い合わせを行う処理について説明する。

\*ATR 自動翻訳電話研究所  
ATR Interpreting Telephony Research Laboratories

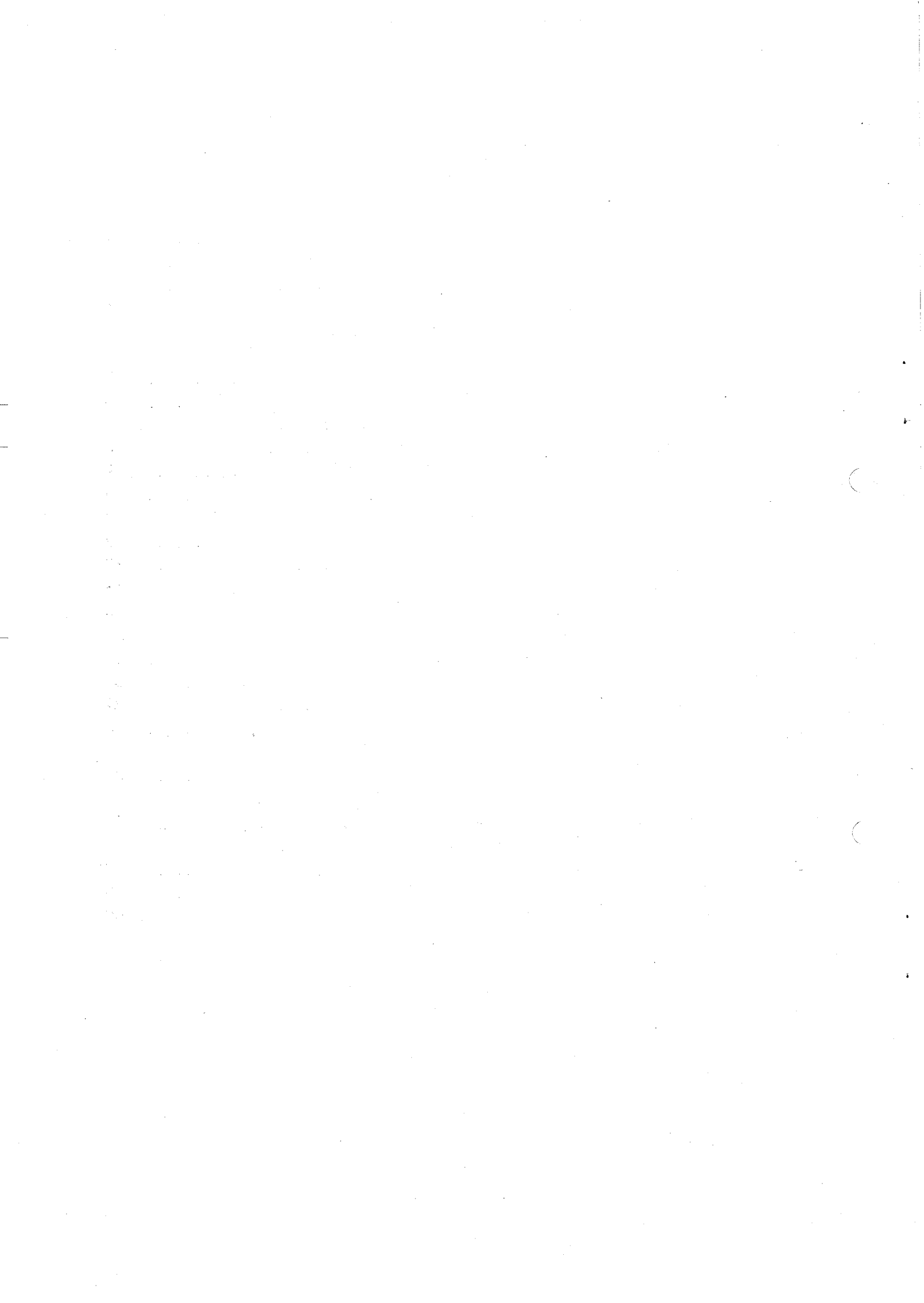
\*\* (株) 漢字情報サービス  
Kanji Information Service

\*\*\*日進ソフトウェア(株)  
NISSIN Software Company

©ATR 自動翻訳電話研究所 1989  
©ATR Interpreting Telephony Research Laboratories

## 目次

1. はじめに .....	1
2. 全体システム構成 .....	1
3. ソフトウェア構成 .....	3
3.1 かな漢字変換処理 .....	4
(1) 概要 .....	4
(2) 入出力データ構造 .....	4
(3) 内部説明 .....	5
3.2 文節候補選択処理 .....	8
(1) 概要 .....	8
(2) 入出力データ構造 .....	8
(3) 問合せ処理の必要な文節候補選択処理結果 .....	9
(4) 内部説明 .....	10
(5) 文節候補の絞り込み結果 .....	10
3.3 問合せ処理 .....	18
(1) 概要 .....	18
(2) 入出力データ構造 .....	18
(3) 内部説明 .....	19
3.4 データ構造変換処理 .....	21
(1) 概要 .....	21
(2) 入出力データ構造 .....	21
(3) 内部説明 .....	22
4. おわりに .....	23
参考文献 .....	24
<付録1 かな漢字変換処理 関数一覧> .....	26
<付録2 文節候補選択処理 関数一覧> .....	35
<付録3 問合せ処理 関数一覧> .....	83
<付録4 データ構造変換処理 関数一覧> .....	90
<付録5 文節候補の絞り込み結果> .....	96



## 1. はじめに

電話による、異なる言語間のコミュニケーションをはかる自動翻訳電話システムを実現するには、音声認識、機械翻訳、音声合成の3つの技術の確立が必要である。音声言語日英翻訳実験システムSL-TRANSは、これらの技術を計算機上で実現したものであり、音声認識を行う部分、言語処理を行う部分、音声処理と言語処理を統合する部分の3つのサブシステムから構成されている。

本稿では、SL-TRANSのシステムの中で音声認識結果をかな漢字に変換する処理、文節ラティスの候補を絞り込む処理、文節候補が誤認識の可能性が高い場合に問い合わせを行う処理について説明する。

## 2. 全体システム構成

図1にSL-TRANSのデモシステムの構成および処理の流れを示す。図中ハッチングを施した部分について概略を以下に説明する。

### (1) かな漢字変換処理

HMM-LRの文節認識結果は文法規則の番号列で出力される。その番号列からターミナルシンボルに対応する文法規則の番号を選び出し、その番号に対応する単語のかな漢字表記を語彙テーブルから求め、かな漢字表記の文節候補を作り出す処理を行う。このかな漢字表記は後の係り受けによる文節候補の選択処理、問い合わせ処理、途中結果表示処理に利用される。

### (2) 係り受けによる文節候補選択処理

HMM-LRの出力である文節ラティスには、各文節に対して5つずつの候補が存在している。この文節ラティスをそのまま言語処理部に渡すと文節候補の組合せ数が非常に多くなり言語処理部の付加が大きくなる。そこで文節間の係り受け関係を用いて文節候補数を削減する。この時言語データベースから作成した係り受けデータを参照している。

### (3) 問い合わせ処理

音声認識処理部での確信度チェックの結果、文節候補の信頼性が低いと判定された場合と、係り受けによる文節候補の選択処理の結果、文節候補の信頼性が低いと判定された場合には、問い合わせ処理を行う。問い合わせ処理では候補を画面に表示し、正しい文節候補をユーザに選んでもらう。

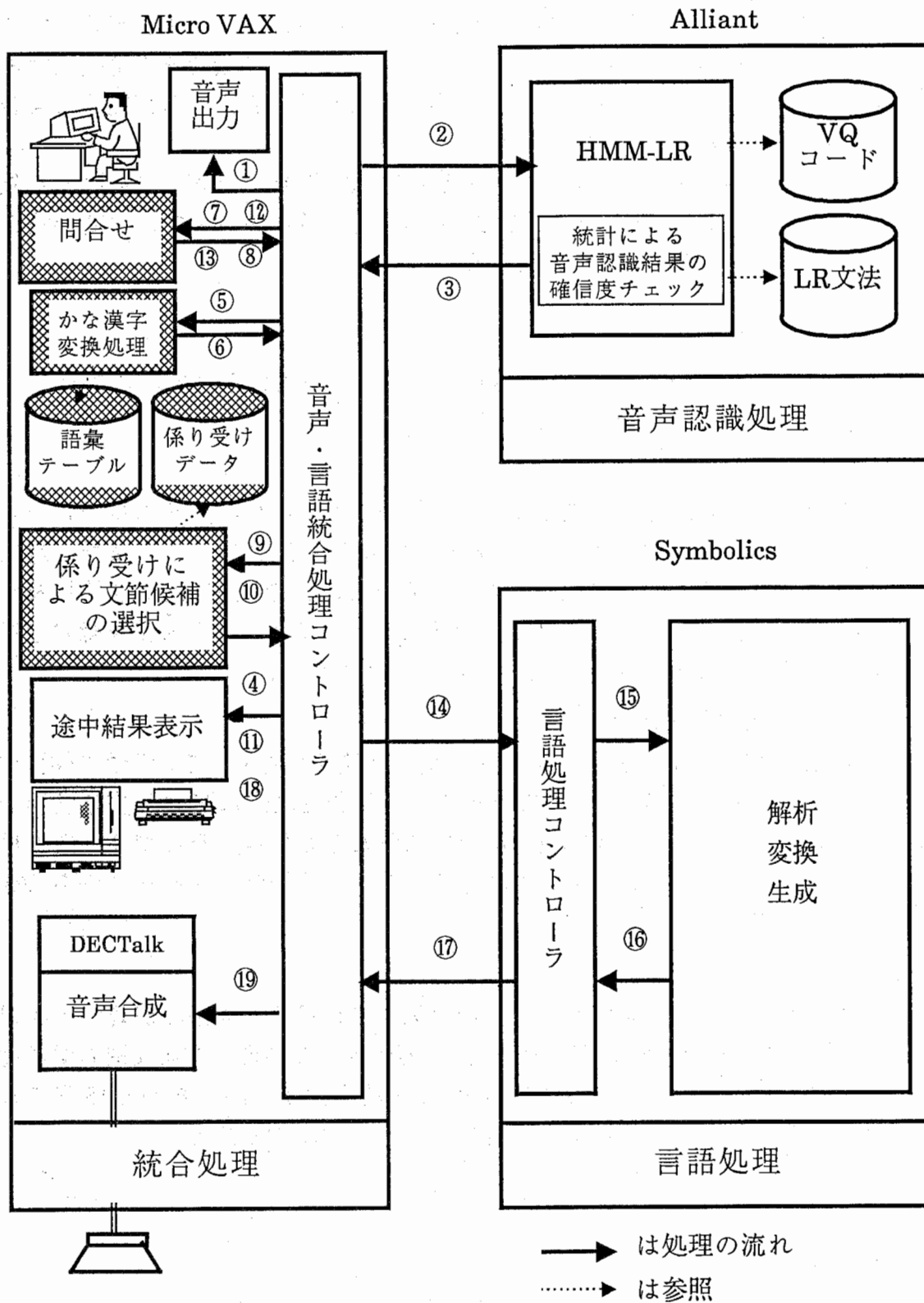


図1. SL-TRANSのデモシステムの構成および処理の流れ

(注)①から⑱は制御の各プロセスの流れを示している。

問合せを行う⑦⑧⑫⑬は例外処理の制御で通常は③⑨⑪⑯⑰と処理は進む。

### 3. ソフトウェア構成

かな漢字変換処理、文節候補選択処理、問い合わせ処理、データ構造変換処理についてこれらのモジュールのシステム全体における位置を図2に示す。これらのモジュールは統合処理部に組み込まれており、Micro Vax上でC言語で記述されている。表示環境はXWindowを用いている。

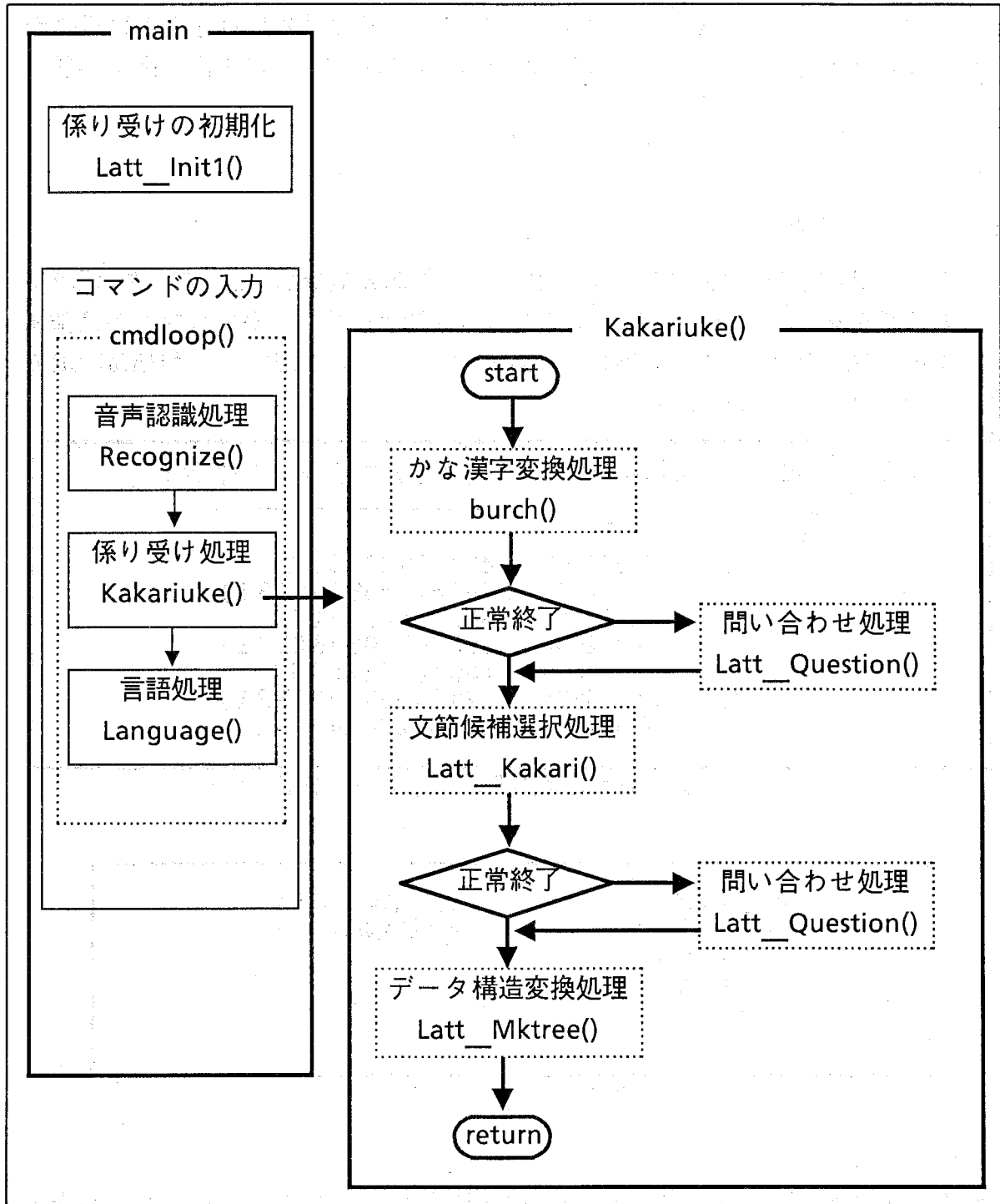


図2 ソフトウェア構成

### 3.1 かな漢字変換処理

#### (1) 概要

HMM-LRの文節認識結果は文節番号、文節候補の確率、適用した文法ルール番号列で出力される。この文法ルール番号はルールを適用した順に並んでおり、語彙に対応するものと語彙に対応しないものが混在している。そこで、その番号列から語彙に対応する文法規則の番号を選び出し、その番号に対応する単語のかな漢字表記を語彙テーブルから求め、かな漢字表記の文節候補を作り出す処理を行う。

#### (2) 入出力データ構造

##### (i) 入力データ構造

かな漢字変換処理への入力データ、すなわち文節認識結果の出力データは図3に示される構造をしている。これは「そちらは会議事務局ですか」に対する認識結果である。文節ラティスの形で表すと図4のようになる。図中の'-'はHMM-LRの文法カテゴリーの区切りを示している。

文番号	文節番号	文節候補の確率	適用した文法ルール番号列
	(2)	(0.949367	402 403 37 32 3 1)
		(0.012658	400 411 403 36 32 3 1)
		(0.012658	402 411 35 32 3 1)
		(0.012658	402 408 403 36 32 3 1)
		(0.012658	402 406 34 32 3 1))
	(3)	(0.935724	623 39 383 551 15 2 1)
		(0.064276	623 39 383 14 2 1))

図3 かな漢字変換処理の入力データ構造

そちら-は	会議事務局-です-か
いつ-から-は	会議事務局-です
そちら-から	
そちら-と-は	
そちら-を	

図4 文節ラティス

##### (ii) 出力データ構造

かな漢字変換処理からの出力データは図5に示される構造をしている。かな漢字表記は認識結果の文法ルール番号列をかな漢字表記に変換したもの。正規表現はかな漢字表記の文字列の自立語部分に対して共通の意味内容を示す単語で表現し

たもので、活用語は終止形で表現している(a)。「です」フラグはその文節に助動詞「だ」、「です」が含まれてダ文表現になっているかどうかを示しており、ダ文表現の場合に1とする。

文番号	かな漢字表記	正規表現
↓ 2.	↓	↓
2:	(そちら-は, 其方, 0.949367, 0 いつ-から-は, いつ, 0.012658, 0 そちら-から, 其方, 0.012658, 0 そちら-と-は, 其方, 0.012658, 0 そちら-を, 其方, 0.012658, 0)	
3:	(会議事務局-です-か, 事務局, 0.935724, 1 会議事務局-です, 事務局, 0.064276, 1)	
↑ 文節番号		↑ 文節候補の確率      ↑ ‘です’フラグ

図5 かな漢字変換処理の出力データ構造

### (3) 内部説明

図6にかな漢字処理の流れを示す。

付録1に個々の関数を簡単に説明する。

かな漢字変換処理には漢字辞書が必要であり、そのファイル名はkanji.dicである。

(a) 現状は先頭の文法カテゴリーに対応する単語の正規表現を用いている。



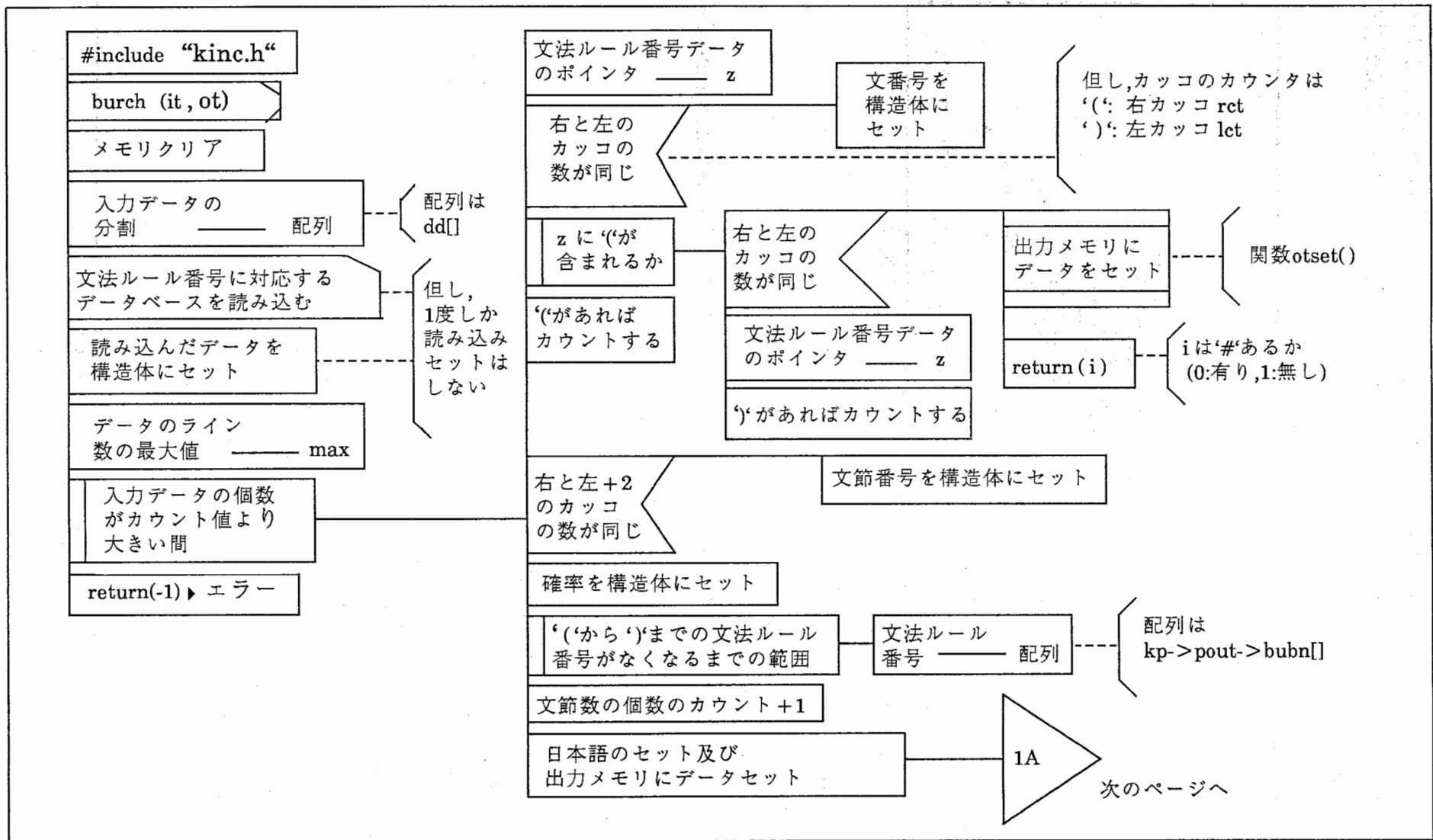


図6(a) かな漢字変換処理プログラムのフロー図 (burch)

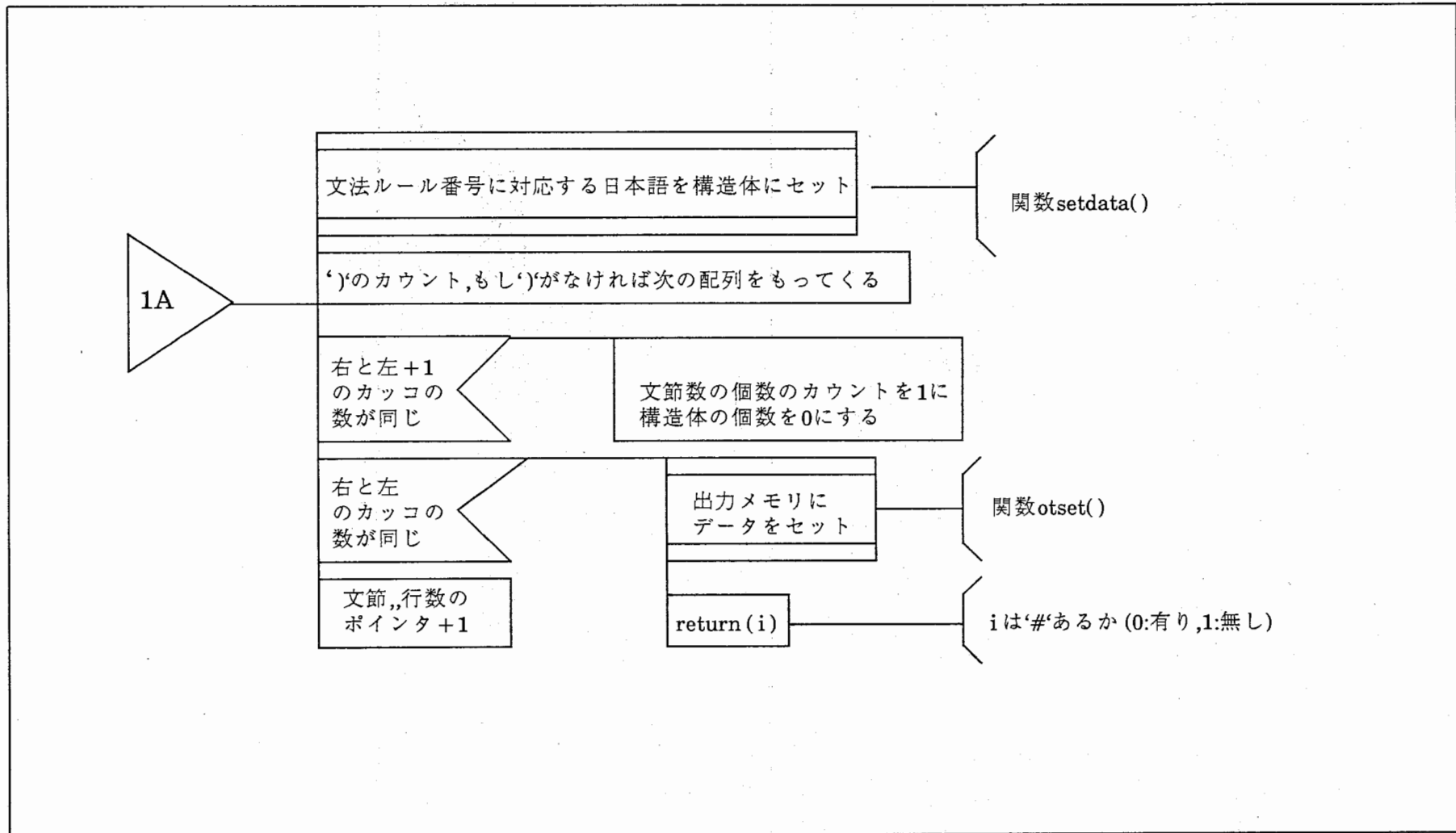


図6 (b) かな漢字変換処理プログラムのフロー図 (burch)

### 3.2 文節候補選択処理

#### (1) 概要

HMM-LRの出力である文節ラティスには、各文節に対して最大5つずつの候補が存在している。この文節ラティスをそのまま言語処理(翻訳処理)の入力とすると文節候補の組合せ数が非常に多くなり言語処理部の付加が大きくなる。そこで文節間の係り受け関係の情報を用いて発話された可能性の高い文節候補を選択し、結果として文節候補数を削減する。この時言語データベースから作成した係り受けデータを参照している。

#### (2) 入出力データ構造

##### (i) 入力データ構造

文節候補選択処理への入力データはかな漢字変換処理の出力データであり、図7に示される。問合せ処理を行った後に文節候補選択処理が行われる場合の入力データは図8に示される。この例は文節番号2番の文節に対して問合せが行われた場合であり、問合せ処理を行った文節の文節候補が1つとなり、その確率は1である。問合せ処理については後で説明する。

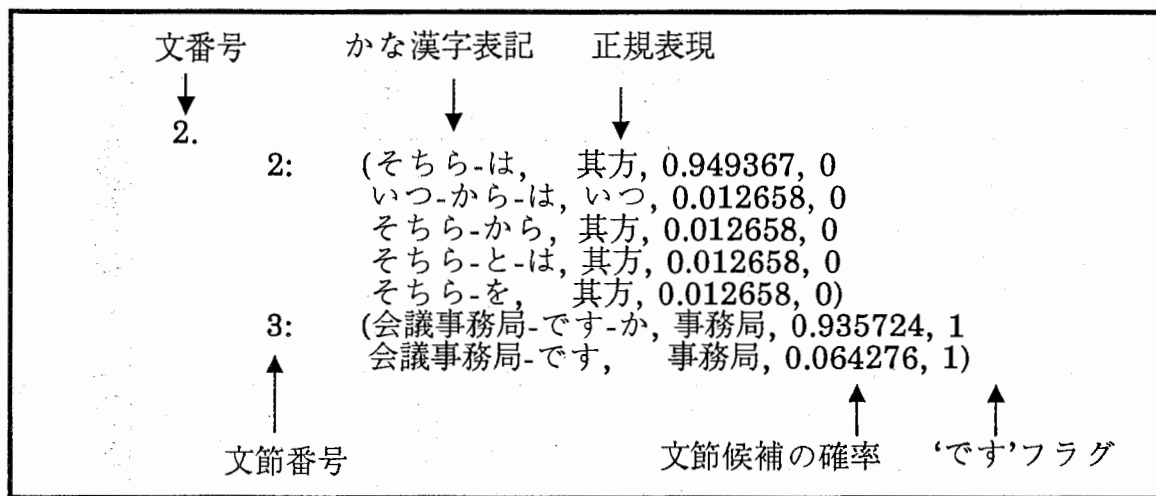


図7 文節候補選択処理の入力データ構造

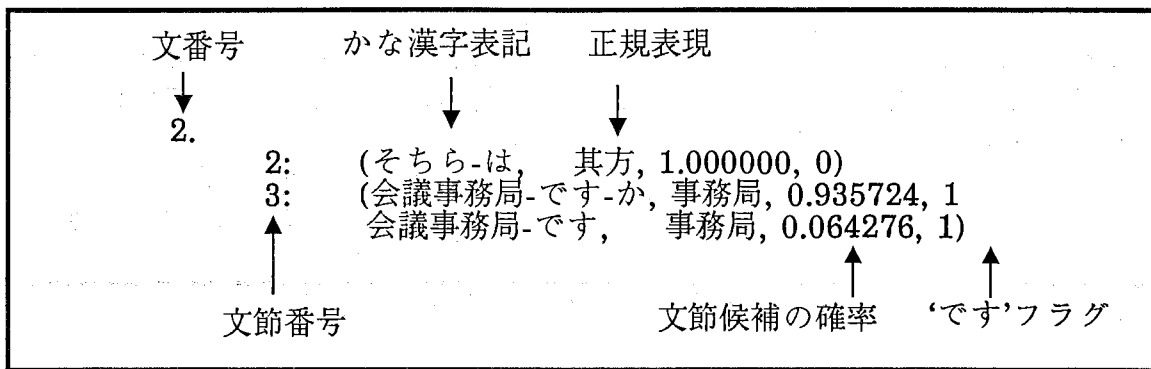


図8 問合せ後の文節候補選択処理の入力データ構造

(ii) 出力データ構造

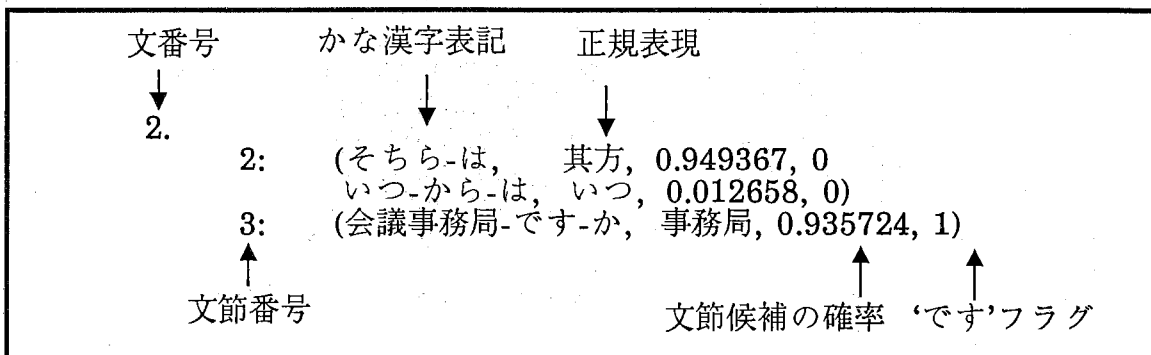


図9 文節候補選択処理の出力データ構造

文節候補選択処理の出力データ構造は図9に示される。これは構造としては入力データ構造と同じであるが、係り受け関係により選択された文節候補が出力されておりその数は最大2個まで削減される。

(3) 問合せ処理の必要な文節候補選択処理結果

HMM-LRの出力である文節ラティスに対して係り受け関係を用いて文節候補選択を行う際に、係り受け関係の付けられない文節候補群が存在する場合がある。例えば「住所は大阪市東区徳井町一の二です」に対する認識結果の文節ラティスは図10に示される。この文節ラティスに対して係り受け関係を用いて文節候補の選択処理を行うと第2文節の認識結果の候補である「希望させる」と「希望される」のどちらも他のどの文節候補とも係り受け関係が成立せずに残る。このような場合「希望させる」と「希望される」は認識誤りである可能性が高い。この時の文節候補選択処理からの出力は図11に示されるように問合せの必要な文節の文節番号の前に#記号を付加する。

住所は	希望させる	東区	徳井町	一の	二です
十七は	希望される	東区を	徳井町に	一も	二ですか
十四とは		東区も	送ります	一を	二ですね
十四は		東区で	送りたい	いつの	二十
十七が		東区が		一にも	二でしょう

図10 文節候補選択処理の入力データ構造

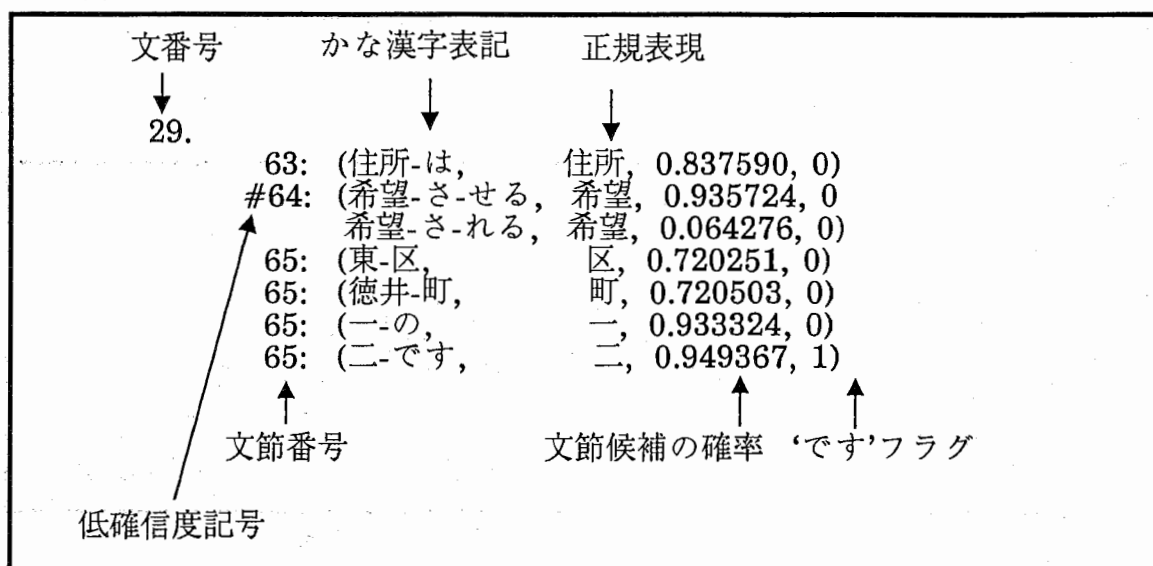


図11 問合せの必要な文節候補選択処理の出力データ構造

#### (4) 内部説明

文節候補選択処理のフローを図12に示す。

付録2に文節候補選択処理の関数を簡単に説明する。

詳細についてはテクニカルレポート TR-I-109 「係り受け関係を用いた文節候補選択処理」[1]を参照のこと。

文節候補選択処理で用いている係り受けデータは言語データベースから作成したものであり[3]、このデータをさらに文節候補選択処理で使い易いように加工して利用している。手順を図13に示す。個々の関数については付録2に示す。

文節候補選択処理には係り受け辞書データファイルとして、Context.dat0、Context.dat1、Context.dat2、Context.dat3、Context.string の5つのファイルと、係り受け辞書に無い文節候補に対する正規表現と意味素性表現を記述したファイルとしてNsosei.dat0が必要である。

#### (5) 文節候補の絞り込み結果

音声認識結果の文節ラティスと、その文節ラティスに対して係り受け関係を用いて文節候補の選択を行った結果を付録5に示す。1文節の文に対しては上位2つの文節を選択しており、付録では結果は省略している。

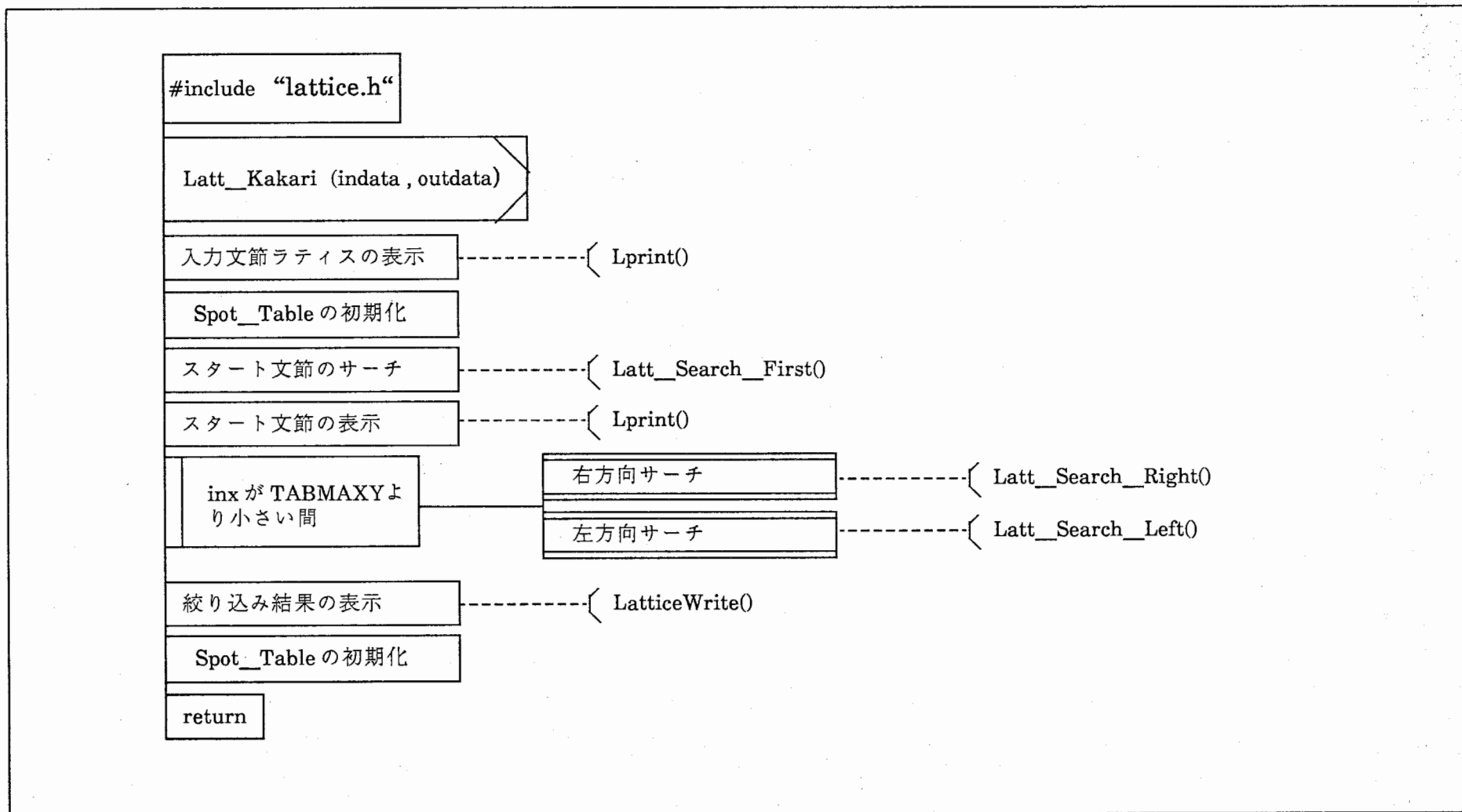


図12(a) 文節候補選択処理プログラムのフロー図 (Latt\_Kakari)

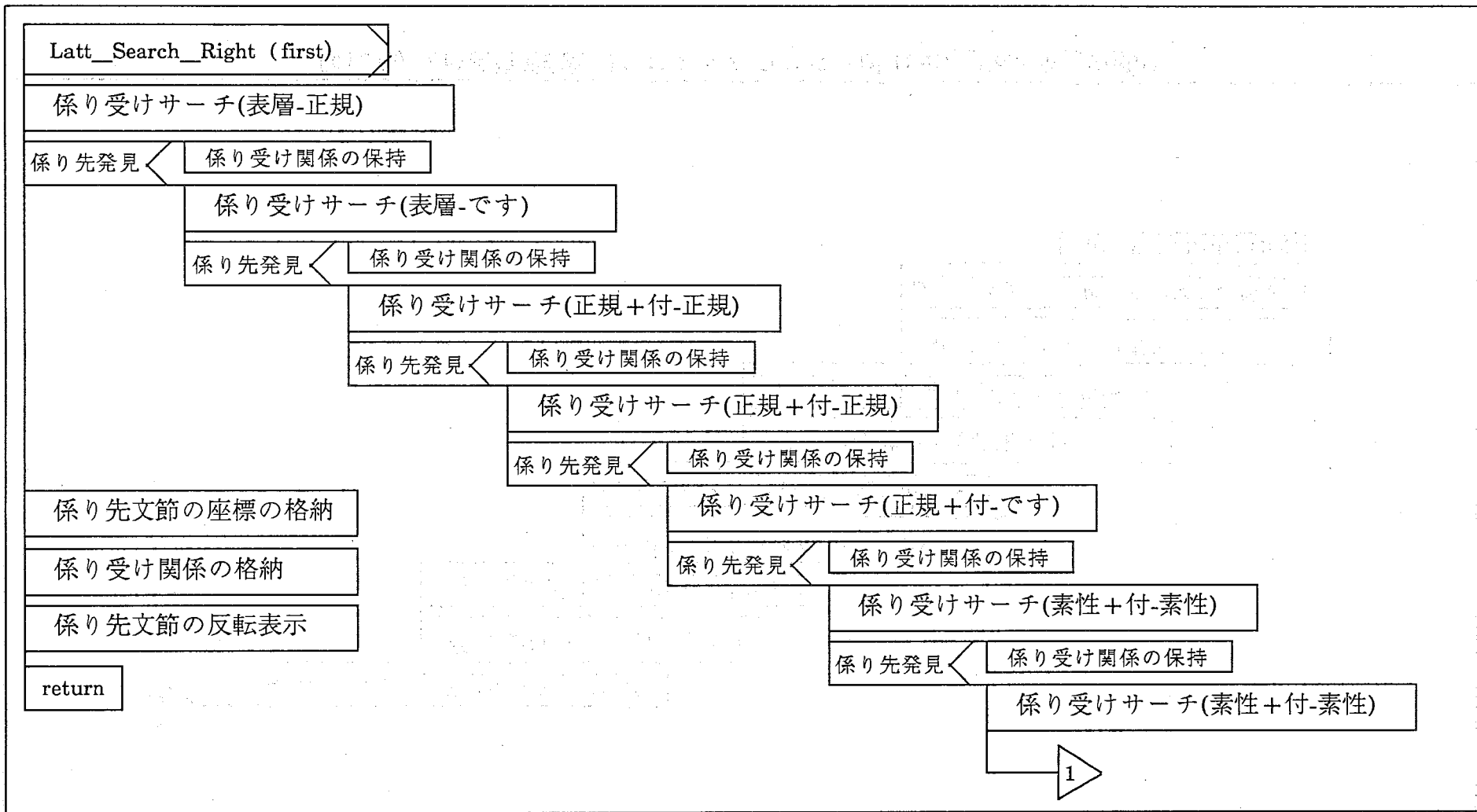


図12 (b) 文節候補選択処理プログラムのフロー図 (Latt\_Search\_Right)



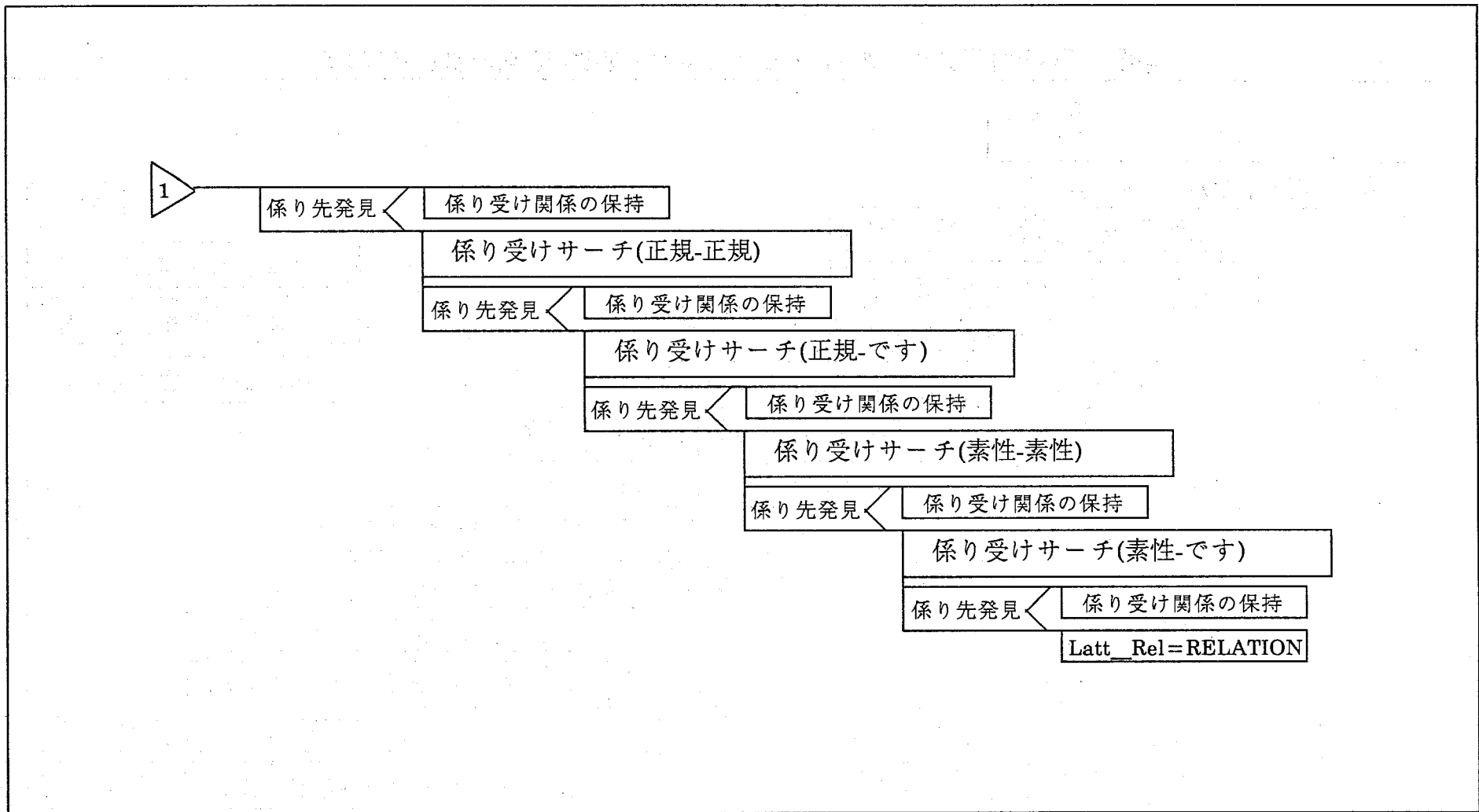


図12(c) 文節候補選択処理プログラムのフロー図 (Latt\_Search\_Right)

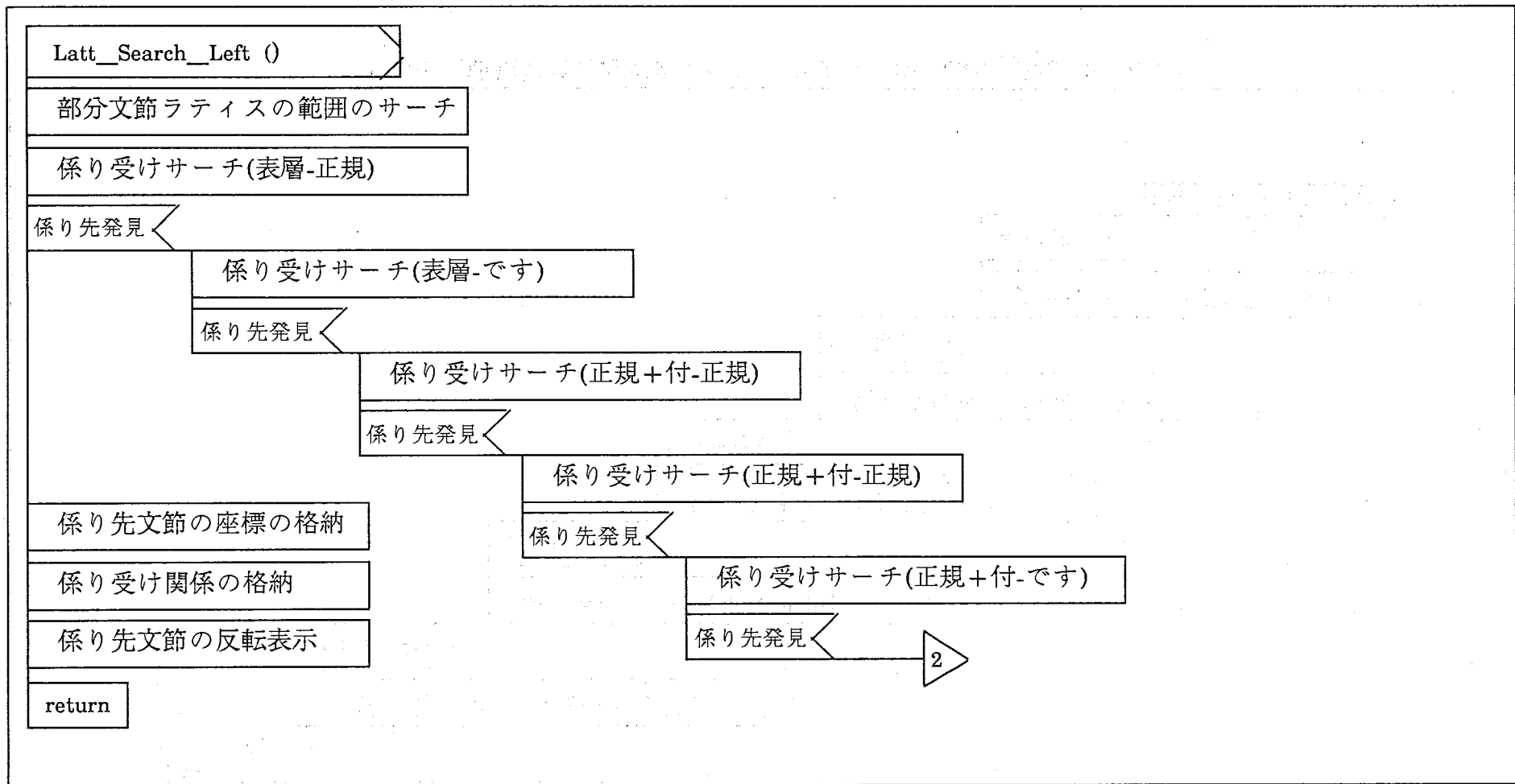


図12(d) 文節候補選択処理プログラムのフロー図 (Latt\_Search\_Left)

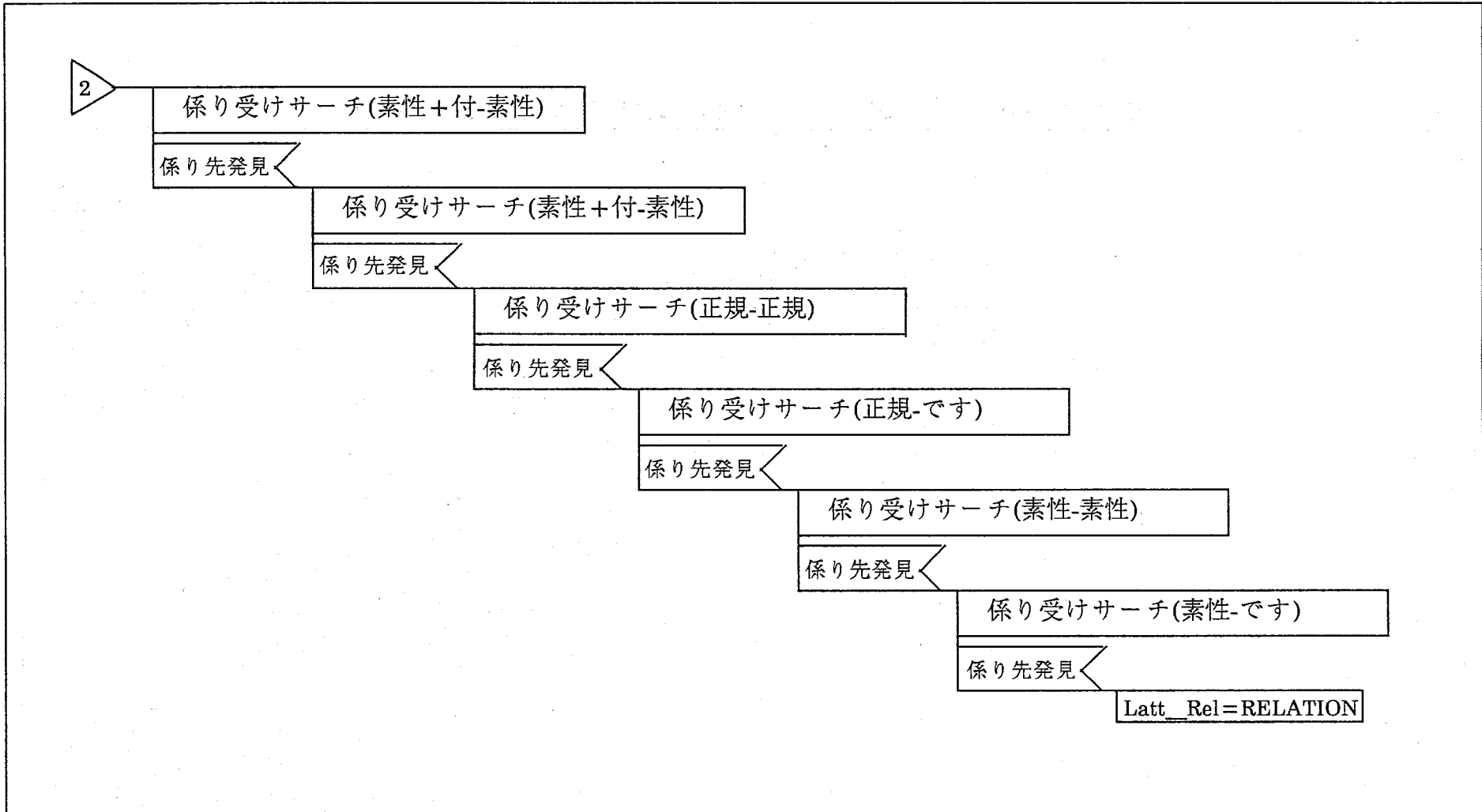


図12(e) 文節候補選択処理プログラムのフロー図 (Latt\_Search\_Left)

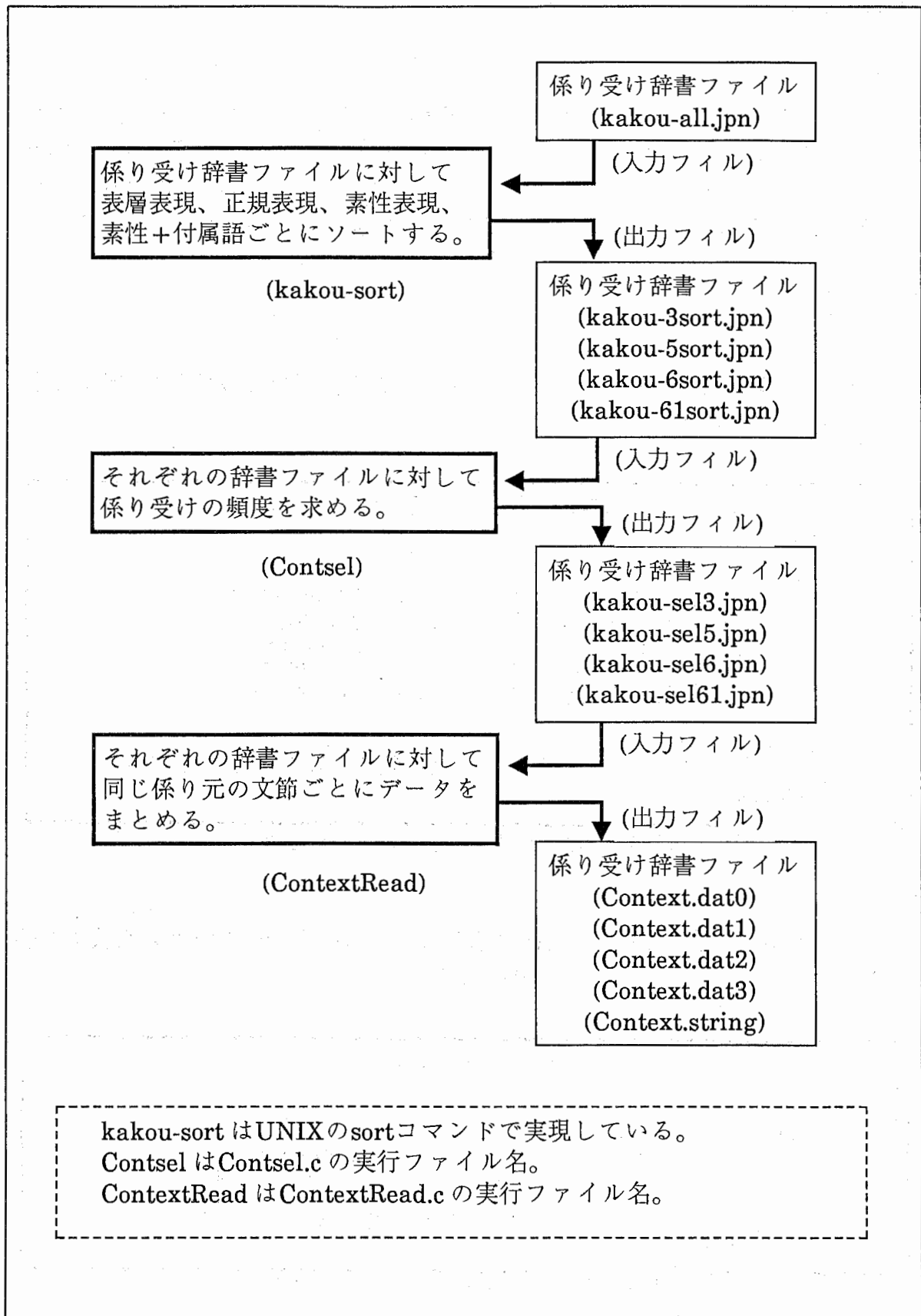


図13 係り受け辞書の加工手順

### 3.3 問合せ処理

#### (1) 概要

音声認識処理部での確信度チェックの結果、文節候補の信頼性が低いと判定された場合と、係り受けによる文節候補の選択処理の結果、文節候補の信頼性が低いと判定された場合には、問い合わせ処理を行う。問い合わせ処理では候補を画面に表示し、正しい文節候補をユーザに選んでもらう。(b)

#### (2) 入出力データ構造

##### (ii) 入力データ構造

問合せ処理の入力データは図14に示されるように問合せの必要な文節の文節番号の前に#記号が付加されている。

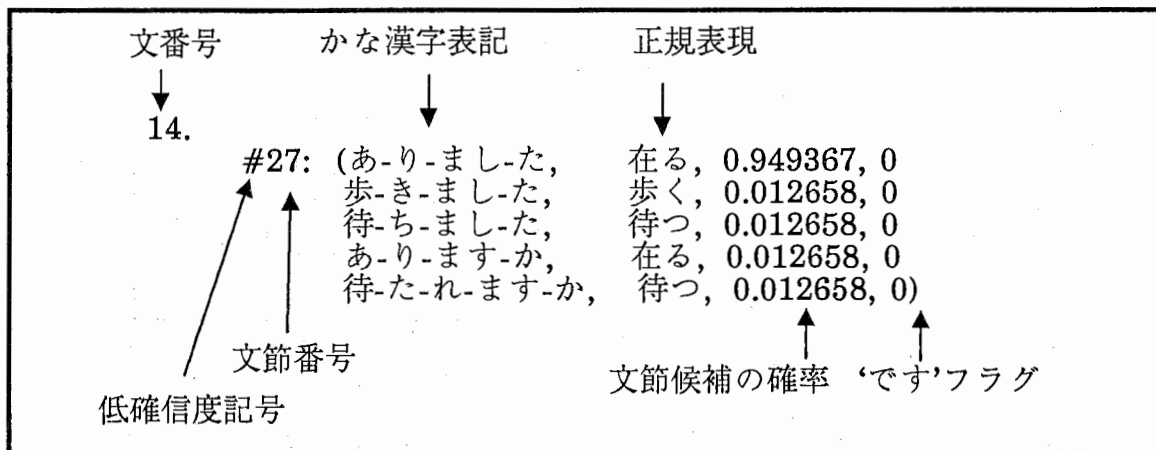


図14 問合せの処理の入力データ構造

##### (ii) 出力データ構造

問合せ処理の出力データは図15に示されるように問合せの行われた文節の候補が1つに確定され、その確率は1になる。

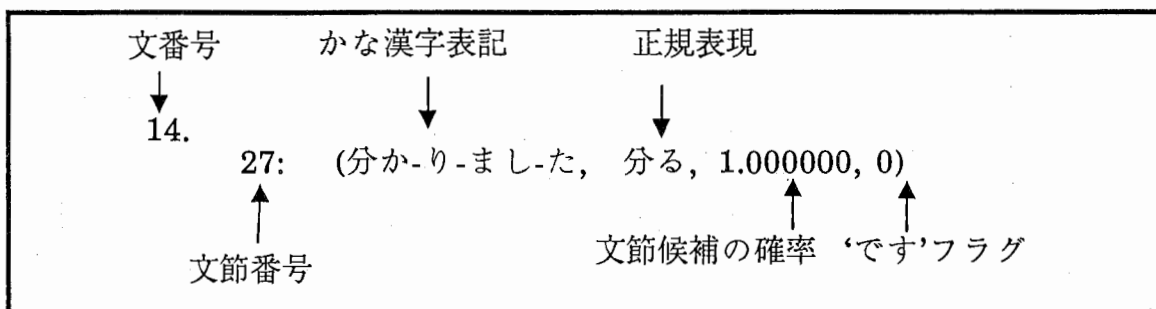


図15 問合せの処理の出力データ構造

(a) 将来的には音声による再入力などが考えられるが、現状は各文節の正解をデータとして保持しており、認識結果の候補に正解の文節候補を付け加えてユーザに問合せを行う。

### (3) 内部説明

\_\_ 図16に問合せ処理のフローを示す。

付録3に問合せ処理の関数を簡単に説明する。

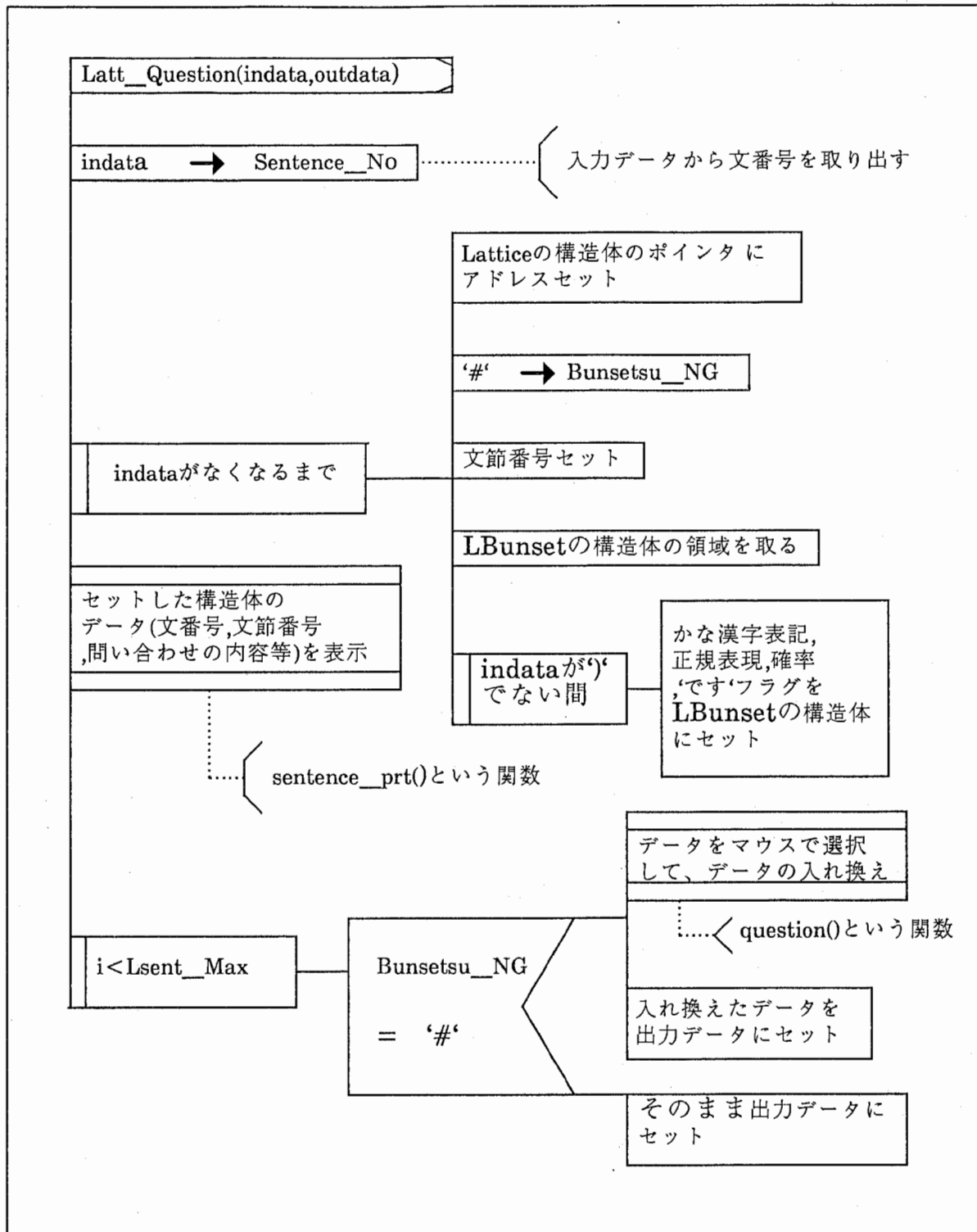


図16 問合せ処理プログラムのフロー図 (Latt\_Question)

### 3.4 データ構造変換処理

#### (1) 概要

文節候補の選択処理の出力は問合せの入力データとなる可能性があるため、かな漢字変換処理の出力データと同じ構造になっている。そこで言語処理にデータを渡す前に構造の変換を行う。これは言語処理がLISP言語で構築されているため、その処理に適用が容易な構造に変換する。

#### (2) 入出力データ構造

##### (i) 入力データ構造

入力データ構造は図17に示される構造であり、これは文節候補選択処理の出力データであるか、あるいは問合せ処理を行った後のデータである。

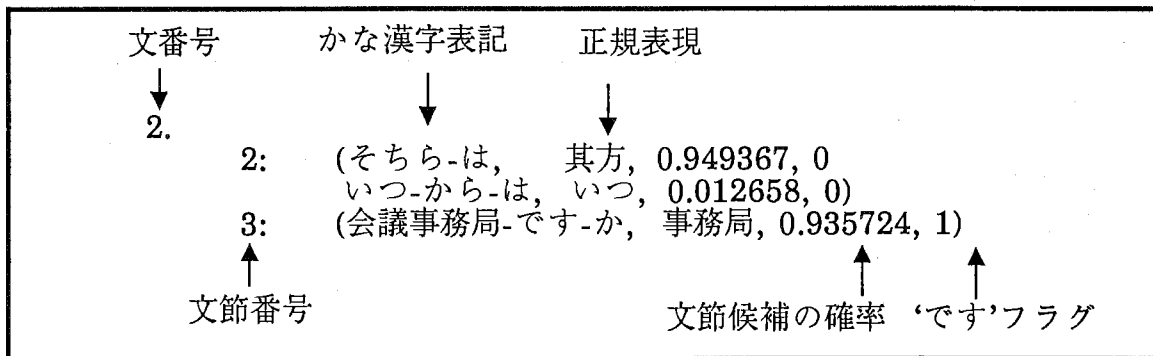


図17 データ構造変換処理の入力データ構造

##### (ii) 出力データ構造

データ構造変換処理の出力データは図18に示されるようにLISP言語のリスト構造である。この出力データに対して言語処理部はさらに文字単位のリスト構造を作成している。



```

(DEF-BUNSETSU-LATTICE l1
  v1 v999
  ((e1 v1 v2 "そちら")
   (e2 v2 v999 "は" (:prob 0.949367))
   (e3 v1 v4 "いつ")
   (e4 v4 v5 "から")
   (e5 v5 v999 "は" (:prob 0.012658)))
  (:nframes 229 :input "sochirawa"))

(DEF-BUNSETSU-LATTICE l2
  v1 v999
  ((e1 v1 v2 "会議事務局")
   (e2 v2 v3 "です")
   (e3 v3 v999 "か" (:prob 0.935724)))
  (:nframes 421 :input "kaigijimukyokudesuka"))

(DEF-LATTICE-SENTENCE s1
  (l1 l2))

```

図18 データ構造変換処理の出力データ構造

### (3) 内部説明

データ構造変換処理の関数について付録4に簡単に説明する。

#### 4. おわりに

音声言語日英翻訳実験システム(SL-TRANS)の中で音声認識結果をかな漢字に変換する処理、文節ラティスの候補を絞り込む処理、文節候補が誤認識の可能性が高い場合に問い合わせを行う処理、文節候補選択結果のデータ構造を変換する処理について説明した。現在このシステムで扱っているサンプル会話はまだその数が少なく簡単なものである。今後はさらに多くの会話に対応できるように機能拡張をはかって行きたい。

## 参考文献

- [1] 柿ヶ原、森元：「係り受け関係を用いた文節候補選択処理」ATRテクニカルレポート TR-I-0109 (1989)
- [2] 小倉、坂野、保坂、森元：「音声言語日英翻訳実験システム(SL-TRANS)」ATRテクニカルレポート TR-I-0102 (1989)
- [3] 柿ヶ原、泉：「文節候補選択処理のための係り受けデータ」ATRテクニカルレポート TR-I-0111 (1989)
- [4] 森元、小倉、相沢、樽松：「音声言語日英翻訳実験システム(SL-TRANS)の概要」情報処理学会第39回全国大会 (1989)
- [5] 柿ヶ原、森元：「係り受け関係を用いた文節候補選択についての一考察」情報処理学会第38回全国大会 (1989)
- [6] 柿ヶ原、森元：「SL-TRANSにおける文節候補の削減」情報処理学会第39回全国大会 (1989)
- [7] 柿ヶ原、森元：「文節間の係り受け関係を用いた文節候補選択手法」音響学会秋季研究発表会 (1989)
- [8] 北、坂野、保坂、川端：「SL-TRANSにおける文節音声認識」情報処理学会第39回全国大会 (1989)
- [9] 小暮、堂坂、加藤：「SL-TRANSにおける言語解析」情報処理学会第39回全国大会 (1989)
- [10] 飯田、長谷川、上田、相沢：「SL-TRANSにおけ変換・生成手法」情報処理学会第39回全国大会 (1989)
- [11] 尾関：「多文節間の係り受け整合度に基づき最適文節列を選択する多段決定アルゴリズム」信学論J71-D(1988-04)
- [12] 松永他：「係り受け規則により候補を限定した文節音声の認識」信学技法 SP87-29(1987-06)
- [13] 中川他：「音節標準パターンと逆時間向き係り受け解析法を用いた日本語文音声の認識」信学論J70-D(1987-12)
- [14] 好田：「文節ラティス上で最適文節列を選択するためのアルゴリズム」信学技法SP86-72(1986-12)
- [15] 井上、小倉、森元：「言語データベース用格・係り受け意味体系」ATRテクニカルレポート TR-I-0029 (1988)

- [16] 北、川端、斎藤：「HMM音韻認識と予測LRパーザを用いた文節認識」電子情報通信学会SP88-88 (1988)
- [17] 坂野、北、森元：「音声認識候補の統計処理による絞り込み」電子情報通信学会春季全国大会A-21 (1988)
- [18] 吉本、小暮：「句構造文法にもとづく日本語文の解析」ATRテクニカルレポート TR-I-0049 (1988)
- [19] 小暮：「解析過程の制御を考慮した句構造文法解析機構の検討」ATRテクニカルレポート TR-I-0064 (1988)
- [20] 荻野：「文法と意味I」朝倉書店 (1983)
- [21] 南：「文法と意味II」朝倉書店 (1985)
- [22] 大野：「類語国語辞典」角川書店 (1985)
- [23] 辺見：「実用X-Window」日刊工業新聞社 (1988)
- [24] 田中：「XWindow System, Version11 ポケットガイド Xlib編」ATRテクニカルレポート TR-I-0091 (1989)
- [25] 田中：「XWindow System, Version11 ポケットガイド Toolkit編」ATRテクニカルレポート TR-I-0092 (1989)

<付録1 かな漢字変換処理 関数一覧>

かな漢字変換処理関数一覧

かな漢字変換処理関数一覧

かな漢字変換処理関数一覧

かな漢字変換処理関数一覧

かな漢字変換処理関数一覧

かな漢字変換処理関数一覧

かな漢字変換処理関数一覧

かな漢字変換処理関数一覧

名前	係り受けデータ加工プログラム ( burch )	
機能	本プログラムは、入力データ(文法ルール番号の表現)と正規表現のデータのファイルを入力とし、前者は改行コードを境として配列に入れて後者はラインごとに構造体に読み込み(データベース),入力データから文番号,文節番号,確率を取り出し、次に文法ルール番号を取り出し、それと現在のデータベースの個数の最大値と比較して大きいと次の文法ルール番号を取り出し、小さいか同じだとその文法ルール番号に対応するデータベースから取り出し、構造体にセットして、それら取り出してきたデータをメモリーに配置,結合して出力データとする。	
呼び出し形式	int burch( it, ot )	
リターン値	1: '#なし, 0: '#有り(文節番号に付いている)	
引数名	タイプ	説明
it	char *	文法ルール番号の入力データ
ot	char *	加工されたデータをセット
備考	但し,データベースはburch関数が1度,呼ばれたときだけ構造体にセットし(フラグ sw3=1),次回呼ばれたときはセットしない。	

名前	日本語データセット ( setdata )	
機能	本プログラムはデータベースから文法ルール番号に対応した係りの自立語,正規表現,係りの付属語など,または'です'をチェックして,それぞれを構造体に領域を取りながらセットしている。	
呼び出し形式	void setdata( e )	
リターン値	無し	
引数名	タイプ	説明
e	int	文法ルール番号の個数
備考		

名前	文法ルール番号による検索(kensa)	
機能	本プログラムはデータベースから文法ルール番号に対応するデータを検索して、その先頭ポインタを返す。	
呼び出し形式	void kensa(p,a,e,ll)	
リターン値	無し	
引数名	タイプ	説明
p	char **	検索したデータのポインタ
a	int	文法ルール番号
e	int	文法ルール番号の個数
ll	int *	現在の文法ルール番号が何個目
備考		



名前	データセット (dataset)	
機能	本プログラムはあるデータから文字列の一部を取り出し,その文字数分だけデータの領域を確保して,そのポインタを返す。	
呼び出し形式	char *dataset(u, dline)	
リターン値	確保した領域のポインタ(エラー:NULL)	
引数名	タイプ	説明
u	char *	取り出したデータをセットするポインタ
dline	char *	取り出されるデータのポインタ
備考		

名前	再割り当て文字列結合( ducat )	
機能	本プログラムは1度確保した領域に再割り当てをして,文字列を結合する。	
呼び出し形式	char *ducat( m , s )	
リターン値	確保した領域のポインタ(エラー:NULL)	
引数名	タイプ	説明
m	char *	結合されるデータのポインタ
s	char *	結合するデータのポインタ
備考		

名前	文字列を取り出す( tol )	
機能	本プログラムはある文字列の空白でない文字から次の空白(含まない)までの文字列を取り出す。	
呼び出し形式	int tol( a, c )	
リターン値	文字列の文字数	
引数名	タイプ	説明
a	char *	取り出す文字列
c	char *	取り出される文字列
備考		

名前	出力メモリーセット ( otset )	
機能	本プログラムは取り込んだ構造体のデータを出力メモリーに配置する。	
呼び出し形式	char *otset( ot, s )	
リターン値	セットしたデータのポインタ	
引数名	タイプ	説明
ot	char *	出力メモリーのポインタ
s	int *	文節番号に'#'が有,無 (0,1)
備考		

## ●その他の関数の説明

関数名	機能
furkard()	出力メモリに自立語,正規表現,確率,'です'のフラグを結合する。
ktol()	文字列からカッコを取る。
ptasu()	文字列の文字数分だけポインタを更新する。
nkut()	文字列から改行を取る。
poit()	文字列の内容が空白であればポインタを更新し,空白でない場合は次の空白までポインタを更新する。
clr()	構造体の初期化する。
clr2()	文字列の文字数分空白をセットし,初期化する。
mjchk()	数字文字かチェックする。

<付録2 文節候補選択処理 関数一覧>

名前	Latt_Init1	
機能	文節候補選択処理の初期化を行う。 選択処理用のウィンドウのオープン、ウィンドウタイトルの表示、係り受け辞書の読み込み、読み込み中のメッセージ表示等を行う。	
呼び出し形式	Latt_Init1()	
リターン値	なし	
引数名	タイプ	説明
備考		

名前	Latt_Init2	
機能	<p>文節候補選択処理の初期化を行う。          整合度の計算式の重みを代入する。          Spot_D=1, Spot_S=5          Spot_Table (文節候補のindex Table) の初期化を行う。          入力文節ラティスの表示エリアのクリアと絞り込み結果の表示エリアのクリアを行う。</p>	
呼び出し形式	Latt_Init2()	
リターン値	なし	
引数名	タイプ	説明
備考		



名前	Latt_Kakari	
機能	<p>認識スコアの高い文節候補(スタート文節)から順次係り受けをサーチする。スタート文節より右方向に対して係り受け可能な文節を選択(右方向サーチ)し、OKのとき選択されていない部分文節群より係り先文節に係る文節を選択(左方向サーチ)する。</p> <p>絞り込み結果の表示とoutdataへデータ変換してLattice_Write()をコールする。</p>	
呼び出し形式	int Latt_Kakari (indata, outdata)	
リターン値	文候補数	
引数名	タイプ	説明
indata	char *	かな漢字変換処理の出力データ
outdata	char *	問合せあるいはデータ変換の入力データ
備考		

名前	Latt_Search_First	
機能	スタート文節のスコア以下の文節候補を選択する。	
呼び出し形式	double Latt_Search_First(Score)	
リターン値	スタート文節の認識スコア	
引数名	タイプ	説明
Score	double	前スタート文節の認識スコア( $0 < \text{score} < 1$ )
備考	選択された文節が任意素性るとき係り受け関係が成立しないため、その文節はスタートの文節にならないがその文節群の一候補として処理する。	

名前	Prob__Compute	
機能	係り受けの整合度の計算を行う。	
呼び出し形式	double Prob__Compute ( y1, y2, x2, prob )	
リターン値	整合度	
引数名	タイプ	説明
y1	int	Lattice_Tableの係り元のY座標
y2	int	Lattice_Tableの係り先のY座標
x2	int	Lattice_Tableの係り先のX座標
prob	double	係り先の認識スコア
備考	$\begin{aligned} \text{整合度} = & \text{頻度(prob)} \\ & - W1(\text{Spot\_D}) * \text{距離}( y2-y1 ) \\ & + W2(\text{Spot\_S}) * \text{認識スコア} \end{aligned}$	

名前	Latt_Search_Right	
機能	右方向サーチ。 ある文節iから右方向の部分文節ラティスに対して、文節iと係り受け関係があり、整合度の最も高い文節jをサーチする。この関数は再帰呼出しされる。	
呼び出し形式	int Latt_Search_Right(first)	
リターン値	S_X:係り先の文節候補のX座標	
引数名	タイプ	説明
first	int	係り元の文節候補のY座標
備考	サーチ順序 1.表層->正規 2.表層->です 3.正規+付属語->正規 4.索性+付属語->索性 5.正規+付属語->です 6.索性+付属語->です 7.正規->正規 8.正規->です 9.表層->索性 10.索性->です	

名前	Latt_Search_Right_10	
機能	係り元の表層に対する係り受け辞書より各係り先候補をサーチして整合度の最も高い文節のY、X座標をS_y、S_xにセットする。	
呼び出し形式	int Latt_Search_Right_10 (Latt_Ptr, S_y, S_x, first, cmp)	
リターン値	係り先Lattice_TableのX座標	
引数名	タイプ	説明
*Latt_Ptr	LATTICE	係り元文節ラティス構造体のポインタ
*S_y	int	係り先Lattice_TableのY座標
*S_x	int	係り先Lattice_TableのX座標
first	int	係り元Lattice_TableのY座標
cmp	int	係り受け関係
備考		

名前	Latt_Search_Right_11	
機能	係り元の表層に対する係り受け辞書より各係り先候補の付属語が“です”をサーチして整合度の最も高い文節のY、X座標をS_y、S_xにセットする。	
呼び出し形式	int Latt_Search_Right_11(Latt_Ptr, S_y, S_x, first, cmp)	
リターン値	係り先Lattice_TableのX座標	
引数名	タイプ	説明
*Latt_Ptr	LATTICE	係り元文節ラティス構造体のポインタ
*S_y	int	係り先Lattice_TableのY座標
*S_x	int	係り先Lattice_TableのX座標
first	int	係り元Lattice_TableのY座標
cmp	int	係り受け関係
備考		

名前	Latt_Search_Right_20	
機能	係り元の正規+付属語に対する係り受け辞書より各係り先候補をサーチして整合度の最も高い文節のY、X座標をS_y、S_xにセットする。	
呼び出し形式	int Latt_Search_Right_20 (Latt_Ptr, S_y, S_x, first, cmp)	
リターン値	係り先Lattice_TableのX座標	
引数名	タイプ	説明
*Latt_Ptr	LATTICE	係り元文節ラティス構造体のポインタ
*S_y	int	係り先Lattice_TableのY座標
*S_x	int	係り先Lattice_TableのX座標
first	int	係り元Lattice_TableのY座標
cmp	int	係り受け関係
備考		

名前	Latt_Search_Right_30	
機能	係り元の素性+付属語に対する係り受け辞書より各係り先候補をサーチして整合度の最も高い文節のY、X座標をS_y、S_xにセットする。	
呼び出し形式	int Latt_Search_Right_30(Latt_Ptr, S_y, S_x, first, cmp)	
リターン値	係り先Lattice_TableのX座標	
引数名	タイプ	説明
*Latt_Ptr	LATTICE	係り元文節ラティス構造体のポインタ
*S_y	int	係り先Lattice_TableのY座標
*S_x	int	係り先Lattice_TableのX座標
first	int	係り元Lattice_TableのY座標
cmp	int	係り受け関係
備考		



名前	Latt_Search_Right_21	
機能	係り元の正規+付属語に対する係り受け辞書より各係り先候補の付属語が“です”をサーチして整合度の最も高い文節のY、X座標をS_y、S_xにセットする。	
呼び出し形式	int Latt_Search_Right_21 (Latt_Ptr, S_y, S_x, first, cmp)	
リターン値	係り先Lattice_TableのX座標	
引数名	タイプ	説明
*Latt_Ptr	LATTICE	係り元文節ラティス構造体のポインタ
*S_y	int	係り先Lattice_TableのY座標
*S_x	int	係り先Lattice_TableのX座標
first	int	係り元Lattice_TableのY座標
cmp	int	係り受け関係
備考		

名前	Latt_Search_Right_22	
機能	係り元の正規に対する係り受け辞書より各係り先候補の正規をサーチして整合度の最も高い文節のY、X座標をS_y、S_xにセットする。	
呼び出し形式	int Latt_Search_Right_22 (Latt_Ptr, S_y, S_x, first, cmp)	
リターン値	係り先Lattice_TableのX座標	
引数名	タイプ	説明
*Latt_Ptr	LATTICE	係り元文節ラティス構造体のポインタ
*S_y	int	係り先Lattice_TableのY座標
*S_x	int	係り先Lattice_TableのX座標
first	int	係り元Lattice_TableのY座標
cmp	int	係り受け関係
備考		

名前	Latt_Search_Right_23	
機能	係り元の正規に対する係り受け辞書より各係り先候補の付属語が“です“をサーチして整合度の最も高い文節のY、X座標をS_y、S_xにセットする。	
呼び出し形式	int Latt_Search_Right_23 (Latt_Ptr, S_y, S_x, first, cmp)	
リターン値	係り先Lattice_TableのX座標	
引数名	タイプ	説明
*Latt_Ptr	LATTICE	係り元文節ラティス構造体のポインタ
*S_y	int	係り先Lattice_TableのY座標
*S_x	int	係り先Lattice_TableのX座標
first	int	係り元Lattice_TableのY座標
cmp	int	係り受け関係
備考		

名前	Latt_Search_Right_31	
機能	係り元の素性に対する係り受け辞書より各係り先候補の素性をサーチして整合度の最も高い文節のY、X座標をS_y、S_xにセットする。	
呼び出し形式	int Latt_Search_Right_31 (Latt_Ptr, S_y, S_x, first, cmp)	
リターン値	係り先Lattice_TableのX座標	
引数名	タイプ	説明
*Latt_Ptr	LATTICE	係り元文節ラティス構造体のポインタ
*S_y	int	係り先Lattice_TableのY座標
*S_x	int	係り先Lattice_TableのX座標
first	int	係り元Lattice_TableのY座標
cmp	int	係り受け関係
備考		

名前	Latt_Search_Right_32	
機能	係り元の素性に対する係り受け辞書より各係り先候補の付属語が“です”をサーチして整合度の最も高い文節のY、X座標をS_y、S_xにセットする。	
呼び出し形式	int Latt_Search_Right_32(Latt_Ptr, S_y, S_x, first, cmp)	
リターン値	係り先Lattice_TableのX座標	
引数名	タイプ	説明
*Latt_Ptr	LATTICE	係り元文節ラティス構造体のポインタ
*S_y	int	係り先Lattice_TableのY座標
*S_x	int	係り先Lattice_TableのX座標
first	int	係り元Lattice_TableのY座標
cmp	int	係り受け関係
備考		

名前	Latt_Search_Left	
機能	ある文節jに左接する部分文節ラテイスに対して、その右端の文節候補のなかで文節jと係り受け関係があり、整合度の最も高い文節kをサーチする。 この関数は再帰呼出しされる。	
呼び出し形式	Latt_Search_Left()	
リターン値	なし	
引数名	タイプ	説明
備考		

名前	Cont_Next_set	
機能	スポッキングポイントと係り先が可能なポイントの決定を行う。Spot_Tableのnext以降で係り先が可能なindexをNext_Table[]にセットする。	
呼び出し形式	Cont_Next_set(prev, next)	
リターン値	なし	
引数名	タイプ	説明
prev	int	部分文節ラテイスの先頭位置
next	int	部分文節ラテイスの終端位置
備考	static int Next_Tbl[50], Next_Tbl_Max;	

名前	Latt_Search_Left_10	
機能	係り元の文節候補[S_y][]のindex番目の表層に対する係り受け辞書よりNext_Table[]の正規をサーチして整合度の最も高い文節のY、X座標を係り元の文節候補[S_x]のLatt_next_y、Latt_next_xにセットする。	
呼び出し形式	int Latt_Search_Left_10(S_y, cmp)	
リターン値	係り元Lattice_TableのX座標	
引数名	タイプ	説明
S_y	int	係り元Lattice_TableのY座標
cmp	int	係り受け関係
備考		



名前	Latt_Search_Left_11	
機能	係り元の文節候補[S_y][]のindex番目の表層に対する係り受け辞書よりNext_Table[]の付属語が“です“をサーチして整合度の最も高い文節のY、X座標を係り元の文節候補[S_x]のLatt_next_y、Latt_next_xにセットする。	
呼び出し形式	int Latt_Search_Left_11(S_y, cmp)	
リターン値	係り元Lattice_TableのX座標	
引数名	タイプ	説明
S_y	int	係り元Lattice_TableのY座標
cmp	int	係り受け関係
備考		

名前	Latt_Search_Left_20	
機能	係り元の文節候補[S_y][]のindex番目の正規+付属語に対する係り受け辞書よりNext_Table[]の正規をサーチして整合度の最も高い文節のY、X座標を係り元の文節候補[S_x]のLatt_next_y、Latt_next_xにセットする。	
呼び出し形式	int Latt_Search_Left_20(S_y, cmp)	
リターン値	係り元Lattice_TableのX座標	
引数名	タイプ	説明
S_y	int	係り元Lattice_TableのY座標
cmp	int	係り受け関係
備考		

名前	Latt_Search_Left_30	
機能	係り元の文節候補[S_y][]のindex番目の素性+付属語に対する係り受け辞書よりNext_Table[]の素性をサーチして整合度の最も高い文節のY、X座標を係り元の文節候補[S_x]のLatt_next_y、Latt_next_xにセットする。	
呼び出し形式	int Latt_Search_Left_30(S_y, cmp)	
リターン値	係り元Lattice_TableのX座標	
引数名	タイプ	説明
S_y	int	係り元Lattice_TableのY座標
cmp	int	係り受け関係
備考		

名前	Latt_Search_Left_21	
機能	係り元の文節候補[S_y][]のindex番目の正規+付属語に対する係り受け辞書よりNext_Table[]の付属語が“です“をサーチして整合度の最も高い文節のY、X座標を係り元の文節候補[S_x]のLatt_next_y、Latt_next_xにセットする。	
呼び出し形式	int Latt_Search_Left_21(S_y, cmp)	
リターン値	係り元Lattice_TableのX座標	
引数名	タイプ	説明
S_y	int	係り元Lattice_TableのY座標
cmp	int	係り受け関係
備考		

名前	Latt_Search_Left_22	
機能	係り元の文節候補[S_y][]のindex番目の正規に対する係り受け辞書よりNext_Table[]の正規をサーチして整合度の最も高い文節のY、X座標を係り元の文節候補[S_x]のLatt_next_y、Latt_next_xにセットする。	
呼び出し形式	int Latt_Search_Left_22(S_y, cmp)	
リターン値	係り元Lattice_TableのX座標	
引数名	タイプ	説明
S_y	int	係り元Lattice_TableのY座標
cmp	int	係り受け関係
備考		

名前	Latt_Search_Left_23	
機能	係り元の文節候補[S_y][]のindex番目の正規に対する係り受け辞書よりNext_Table[]の付属語が“です“をサーチして整合度の最も高い文節のY、X座標を係り元の文節候補[S_x]のLatt_next_y、Latt_next_xにセットする。	
呼び出し形式	int Latt_Search_Left_23(S_y, cmp)	
リターン値	係り元Lattice_TableのX座標	
引数名	タイプ	説明
S_y	int	係り元Lattice_TableのY座標
cmp	int	係り受け関係
備考		

名前	Latt_Search_Left_31	
機能	係り元の文節候補[S_y][]のindex番目の正規素性に対する係り受け辞書よりNext_Table[]の素性をサーチして整合度の最も高い文節のY、X座標を係り元の文節候補[S_x]のLatt_next_y、Latt_next_xにセットする。	
呼び出し形式	int Latt_Search_Left_31(S_y, cmp)	
リターン値	係り元Lattice_TableのX座標	
引数名	タイプ	説明
S_y	int	係り元Lattice_TableのY座標
cmp	int	係り受け関係
備考		

名前	Latt_Search_Left_32	
機能	係り元の文節候補[S_y][]のindex番目の正規素性に対する係り受け辞書よりNext_Table[]の付属語が“です”をサーチして整合度の最も高い文節のY、X座標を係り元の文節候補[S_x]のLatt_next_y、Latt_next_xにセットする。	
呼び出し形式	int Latt_Search_Left_32(S_y, cmp)	
リターン値	係り元Lattice_TableのX座標	
引数名	タイプ	説明
S_y	int	係り元Lattice_TableのY座標
cmp	int	係り受け関係
備考		



名前	Ninni_Sosei	
機能	係り元の文節候補[S_y][]のindex番目の正規素性に対する係り受け辞書よりNext_Table[]の付属語が“です“をサーチして整合度の最も高い文節のY、X座標を係り元の文節候補[S_x]のLatt_next_y、Latt_next_xにセットする。	
呼び出し形式	Ninni_Sosei ( spot_y, spot_x, prob, cmp )	
リターン値	なし	
引数名	タイプ	説明
spot_y	int	Lattice_TableのY座標
spot_x	int	Lattice_TableのX座標
prob	double	文節候補[S_y][S_x]の認識スコア
cmp	int	係り受け関係
備考		

名前	Sentence_save	
機能	文節候補列Spot_Table[]の内容をSentence_Table[sent_inx++][]にsaveする。	
呼び出し形式	Sentence_save (sent_inx)	
リターン値	文候補数	
引数名	タイプ	説明
sent_inx	int	文候補数
備考		

名前	Sentence_Prt	
機能	文候補Sentence_Table[][]の表示を行う。	
呼び出し形式	Sentence_Prt()	
リターン値	なし	
引数名	タイプ	説明
備考		

名前	Lprint	
機能	文節候補選択処理途中の文節ラテイスの表示を行う。 係り受け関係をサーチしている文節を反転表示する。	
呼び出し形式	Lprint()	
リターン値	なし	
引数名	タイプ	説明
備考		

名前	LatticeWrite	
機能	絞り込んだ文節候補に対して同じ文節候補が存在する場合に1つにする。	
呼び出し形式	LatticeWrite ( outdata, No )	
リターン値	なし	
引数名	タイプ	説明
outdata	char *	データ構造変換処理の入力データ
No	int	文番号
備考		



名前	ContextString (辞書文字列対応ファイルREAD)	
機能	filenameで指定された辞書文字列対応ファイルの()の中の文字列をSymbol_String[]にセットしていく。 filenameはContext.string	
呼び出し形式	ContextString ( filename )	
リターン値	なし	
引数名	タイプ	説明
filename	char *	辞書文字列対応ファイル名のポインタ
備考	辞書文字列対応ファイルの例 (\$) (かかります) (掛かる) (%-を) (%) (頂く) (戴く) (始まって) (始まる) .....	

名前	NsoseiRead (意味素性対応ファイルREAD)	
機能	filenameで指定された意味素性対応ファイルの中の正規表現と意味素性をNsosei_co[]にセットする。 filenameはNsosei.dat0	
呼び出し形式	ContextString ( filename )	
リターン値	なし	
引数名	タイプ	説明
filename	char *	意味素性対応ファイル名のポインタ
備考	意味素性対応ファイルの例 意味 : abs/log 含む : pre/fkm 宜しい : adj/yrs 区 : loc/reg 限る : pre/kgr 市 : loc/reg .....	



名前	LatticeRead (文節ラテイスREAD)	
機能	<p>文節ラテイスを読み込み__Latticeの構造体にセットする(表層、正規表現)。このとき__Contextの係り受け情報(表層、正規、正規+付属語、素性、素性+付属語)をセットする。文節の候補は'('から')'までであり、各候補の区切りはスペースとする。</p> <p>自立語のグループ化の処理を行う。これは文節候補群のなかで正規表現が等しい文節候補をまとめるものである。</p>	
呼び出し形式	int LatticeRead ( Latt_data )	
リターン値	文番号(No)	
引数名	タイプ	説明
Latt_data[]	char	かな漢字変換処理の出力データ
備考		

名前	SearchSymbol (指定シンボルのサーチ)	
機能	Symbol_Tableよりbuff1あるいはbuff2と等しいIndexを返す。	
呼び出し形式	int SearchSymbol ( buff1, buff2, Grp )	
リターン値	Symbol_TableのIndex	
引数名	タイプ	説明
buff1	char *	表層表現のポインタ
buff2	char *	正規表現のポインタ
Grp	int *	正規表現のIndex (Group処理で使用)
備考		

名前	SearchContext (指定シンボルのサーチ)	
機能	Context_TableよりSymbolPtrと等しいIndexを返す。	
呼び出し形式	int SearchContext (SymbolPtr)	
リターン値	Context_TableのIndex	
引数名	タイプ	説明
SymbolPtr	int	Symbol_TableのIndex
備考		

名前	SearchContextE (指定シンボルのサーチ)	
機能	Context_TableよりSymbolPtrと等しいIndexを返す。	
呼び出し形式	int SearchContextE (SymbolPtr)	
リターン値	Context_TableのIndex	
引数名	タイプ	説明
SymbolPtr	int	Symbol_TableのIndex
備考		

名前	SearchContextE__aux (指定シンボルのサーチ)	
機能	Context_Tableより buff1+buff2と等しいIndexを返す。	
呼び出し形式	int SearchContextE__aux ( buff1, buff2 )	
リターン値	Context_TableのIndex	
引数名	タイプ	説明
buff1	char *	正規表現のポインタ
buff2	char *	付属語のポインタ
備考		

名前	SearchContextS (指定シンボルのサーチ)	
機能	Cont_inx1の素性あるいはCont_inx2の素性と等しいContext_TableのIndexを返す。 Latt_strで示される正規表現の素性が'XXXXXXXX'のときはNsosei_co[]よりサーチする。	
呼び出し形式	int SearchContextS(Cont_inx1, Cont_inx2, Latt_str)	
リターン値	Context_TableのIndex	
引数名	タイプ	説明
Cont_inx1	int	Context_TableのIndex
Cont_inx2	int	Context_TableのIndex
Latt_str	char *	正規表現
備考		

名前	SearchContextS__aux (指定シンボルのサーチ)	
機能	Cont__inx1の素性+付属語あるいはCont__inx2の素性+付属語と等しいContext__TableのIndexを返す。 Latt__strで示される表層表現の素性が'XXXXXXXX'のときはNsosei__co[]よりサーチする。	
呼び出し形式	int SearchContextS ( Cont__inx1, Cont__inx2, Latt__str )	
リターン値	Context__TableのIndex	
引数名	タイプ	説明
Cont__inx1	int	Context__TableのIndex
Cont__inx2	int	Context__TableのIndex
Latt__str	char *	表層表現
備考		

## ●その他の関数の説明

関数名	機能
SkipSpace(fp) FILE *fp	スペースをスキップする
SkipChar(fp, ch) FILE *fp; int ch;	キャラクタをスキップする
SSStr(p, inx) char *p; int *inx;	スペースをスキップする
SCStr(p, ch, inx) char *p; char ch; int *inx;	キャラクタをスキップする



名前	係り受けデータのソート(ContextRead)	
機能	ソートされた係り受け辞書ファイルに対して同じ係り受け関係の数を調べ、頻度を求める。同じ係り受け関係はまとめる。	
呼び出し形式	void ContextRead (input_fp, output_fp)	
リターン値	無し	
引数名	タイプ	説明
input_fp	FILE	入力データファイルのポインタ
output_fp	FILE	出力データファイルのポインタ
備考	プログラムファイル名はContsel.c	

名前	係り受けデータのソート(main)	
機能	係り受け辞書データのソートプログラム。 入力データファイル名と出力データファイル名の取り込みを行う。	
呼び出し形式	main()	
リターン値	無し	
引数名	タイプ	説明
備考	プログラムファイル名はContsel.c	

名前	係り受け辞書データの加工(ContextRead)	
機能	Contselによってソートされた係り受け辞書ファイルに対して同じ係り元の文節ごとにデータをまとめる処理を行う。そのとき、係り受け辞書ファイルの表層表現、正規表現を別ファイル(Context.string)にまとめてそのインデックスでサーチできるようにする。	
呼び出し形式	void ContextRead (input_fp)	
リターン値	無し	
引数名	タイプ	説明
input_fp	FILE	入力ファイルのポインタ
備考	プログラムファイル名はContextRead.c	

名前	係り受け辞書データの加工(main)	
機能	係り受け辞書データ加工プログラム(ContextRead)に対して入出力ファイル名の取り込みを行う。	
呼び出し形式	main()	
リターン値	無し	
引数名	タイプ	説明
備考	プログラムファイル名はContextRead.c	

名前	指定シンボルのサーチ(SearchSymbol)	
機能	指定されたシンボルに等しいシンボルをSymbolTableよりサーチし、対応するインデックスを返す。	
呼び出し形式	int SearchSymbol(buff)	
リターン値	インデックス	
引数名	タイプ	説明
buff	char*	サーチするシンボルのポインタ
備考	プログラムファイル名はContextRead.c	

<付録3 問合せ処理 関数一覧>

名前	問い合わせ処理(Latt__Question)	
機能	本プログラムは入力データにおいて、'#'が文節番号にあれば、マウスで選択して、入力データを選択されたデータに入れ換えて出力データとする、また、もし'#'が文節番号になれば入力データを出力データとする。	
呼び出し形式	void Latt__Question(indata,outdata)	
リターン値	無し	
引数名	タイプ	説明
indata	char *	問い合わせ処理の入力データのポインタ
outdata	char *	問い合わせ処理の出力データのポインタ
備考	但し、入れ換えるための出力データはプログラム中の外部変数の領域に格納されている。	

名前	問合せ処理の画面のクリア(Clear_Q)	
機能	本プログラムは問合せ処理のウインドウ画面をクリアする。	
呼び出し形式	void Clear_Q()	
リターン値	無し	
引数名	タイプ	説明
備考		



名前	問合せ処理の初期画面(init_window)	
機能	本プログラムは文節候補の問合せ処理のウィンドウの初期画面を作成し、また選択するためのマウスのエリアを設定している。	
呼び出し形式	void init_window()	
リターン値	無し	
引数名	タイプ	説明
備考		

名前	マウスで選択及び入れ換え処理(question)	
機能	本プログラムは選択するためのデータを表示して、それをマウスで選択し、入力データと選択されたデータとを入れ換えて出力データとする。	
呼び出し形式	void question(ptr,buff)	
リターン値	無し	
引数名	タイプ	説明
ptr	Lattice *	Latticeの構造体のポインタ
buff	char *	出力データのポインタ
備考		

名前	問い合わせのデータの表示(sentence_prt)	
機能	本プログラムは問い合わせの文番号,文節番号,内容の表示をする。	
呼び出し形式	void sentence_prt()	
リターン値	無し	
引数名	タイプ	説明
備考		

## ●その他の関数の説明

関数名	機能
get_button_1()	マウスで選択クリックしたとき、どれが選択されたか、それぞれの関数でフラグをたてる。
get_button_2()	
get_button_3()	
get_button_4()	
get_button_5()	
get_button_6()	
get_button_dummy()	

<付録4 データ構造変換処理 関数一覧>

名前	データ構造変換プログラム ( Latt_Mktree )	
機能	文節候補群に対して '-' で区切られるフレーズをエッジとして DEF-BUNSETU-LATTICE lx を作成する。lx は 1+文節番号である。 最後に DEF-BUNSETU-SENTENCE sx を作成する。sx は 1+文番号である。	
呼び出し形式	void Latt_Mktree ( indata , outdata )	
リターン値	なし	
引数名	タイプ	説明
indata	char *	文節候補選択処理の結果。入力buffのポインタ
outdata	char *	言語処理への入力データ。出力buffのポインタ
備考		

名前	エッジの出力( printnodelist )	
機能	フレーズ(node)に続く (node->chile[ ])の処理をする。	
呼び出し形式	void printnodelist( node , outdata )	
リターン値	なし	
引数名	タイプ	説明
node	Node	フレーズ先頭ポインタ
outdata	char *	出力buffのポインタ
備考		

名前	ノード名の出力(printnode)	
機能	フレーズ(node)の処理をする。nodenameの出力。	
呼び出し形式	void printnode( node , startno , outdata )	
リターン値	なし	
引数名	タイプ	説明
node	Node	フレーズポインタ
startno	int	フレーズ番号
outdata	char *	出力buffのポインタ
備考		



名前	文節の分解 (getphrase)	
機能	文節を'-'を区切りとして分解しph[]にセットする。	
呼び出し形式	void getphrase ( bunsetsu , ph )	
リターン値	なし	
引数名	タイプ	説明
bunsetsu	char *	文節文字列
ph	char *	分解した文節をセーブするエリア
備考		

## ●その他の関数の説明

関数名	機能
<code>generate__sentenceno()</code>	<code>sentanceno</code> をインクリメントし、リターン値として返す
<code>generate__latticeno()</code>	<code>latticeno</code> をインクリメントし、リターン値として返す
<code>generate__edgeno()</code>	<code>edgeno</code> をインクリメントし、リターン値として返す
<code>generate__vertexno()</code>	<code>vertexno</code> をインクリメントし、リターン値として返す
<code>resetnode()</code>	<code>curnode</code> , <code>edgeno</code> , <code>vertexno</code> をリセットする

## &lt;付録5 文節候補の絞り込み結果&gt;

- (1) もしもし。  
 (2) そちらは 会議事務局ですか。

## &lt;文節ラテイス&gt;

そちらは 会議事務局ですか  
 いつからは 会議事務局です  
 そちらから  
 そちらとは  
 そちらを

## &lt;絞り込み結果&gt;

そちらは 会議事務局ですか  
 いつからは

- (3) はい。  
 (4) そうです。  
 (5) 会議に 申し込みたいのですが。

## &lt;文節ラテイス&gt;

会議に 申し込みたいのですが  
 八に 申し込みたいのですか  
 八人 申し上げたいのですが  
 会議 乗り継ぎたいのですが  
 八人に 申し上げたいのですか

## &lt;絞り込み結果&gt;

会議に 申し込みたいのですが  
 申し込みたいのですか

- (6) 登録用紙は 既に お持ちでしょうか。

## &lt;文節ラテイス&gt;

登録用紙は 既に お持ちでしょうか  
 登録用紙が いつに 思いましようか  
 登録した 千に お持ちましようか  
 登録用紙とは 九でしょうか  
 登録したら お持ちしては

## &lt;絞り込み結果&gt;

登録用紙は いつに お持ちでしょうか  
 既に

- (7) いいえ。  
 (8) まだです。  
 (9) 分かりました。  
 (10) それでは 登録用紙を お送り致します。

<文節ラテイス>

それでは 登録用紙も お送り致します  
 三では 登録用紙を お送りします  
 すれば 登録用紙と お聞き致します  
 会議 登録用紙とも お持ち致します  
 八人に 登録費としても お送りしています

<絞り込み結果>

それでは 登録用紙も お送り致します  
 登録用紙を

- (11) ご住所と お名前を お願いします。

<文節ラテイス>

ご住所と お名前を お願いします  
 ご住所とも お名前も お願いしますか  
 お名前と お願いしない  
 お名前とも お願いしましょう  
 お名前は お願いしたい

<絞り込み結果>

ご住所と お名前を お願いします  
 お名前も

- (12) 住所は 大阪市 北区 茶屋町 二十三です。

<文節ラテイス>

住所は 希望させる 北区 茶屋町 二十三です  
 十七は 大阪市 北区を 茶屋町に 二十三ですか  
 十四は 大阪市内 北区と 茶屋町と 二十三ですね  
 十四が 希望される 北区で 茶屋町の 二十三でしょう  
 十七が 送らせる 北区が 茶屋町が

<絞り込み結果>

住所は 大阪市 北区 茶屋町 二十三です  
 二十三ですか

(13) 名前は 鈴木真弓です。

<文節ラテイス>

名前は 鈴木真弓です  
 名前が 鈴木真弓ですか  
 七は 鈴木真弓ですね  
 なれば 鈴木真弓でしょう  
 七が 鈴木真弓として

<絞り込み結果>

名前は 鈴木真弓です  
 鈴木真弓ですか

(14) 分かりました。

(15) 登録用紙は 至急 送らせていただきます。

<文節ラテイス>

登録用紙が 至急 送らせていただきます  
 登録用紙は 七十 送らせて致します  
 登録用紙から 七十を 含ませていただきます  
 登録しました 七十五 含ませて致します  
 登録用紙とは 七十も 送らせられます

<絞り込み結果>

登録用紙が 至急 送らせていただきます  
 登録用紙は

(16) 分からない 点が ございましたら いつでも お聞き下さい。

<文節ラテイス>

分からない 点が ございましたら いつでも お聞き下さい  
 送らない 意味が ございましては 一でも お送り下さい  
 分かりたい 入れるが 言われますか いつも お聞きしてある  
 分からない 四が 行かれますか いつにも  
 在りたい 言えば 五千ですか いつの

<絞り込み結果>

分からない 点が ございましたら いつでも お聞き下さい  
 送らない

(17) 有難うございます。

(18) それでは 失礼します。

<文節ラテイス>

すれば 失礼します  
 三では 失礼しますか  
 いつでは 失礼しない  
 それでは 失礼したい  
 すがが されます

<絞り込み結果>

すれば されます  
 それでは 失礼します

(19) どうも 失礼致します。

<文節ラテイス>

五の 失礼致します  
 五も して致します  
 どうも 失礼します  
 五を 失礼しています  
 五のも 失礼して致します

<絞り込み結果>

どうも 失礼致します  
 失礼します

(20) もしもし。

(21) こちらは 会議事務局です。

<文節ラテイス>

こちらは 会議事務局です  
 九からは 会議事務局ですか  
 九から 会議事務局ですね  
 六が 会議事務局でしょう  
 送れば

<絞り込み結果>

こちらは 会議事務局です  
 会議事務局ですか

(22) 会議に 参加したいのですが。

<文節ラテイス>

会議に 参加したいのですが  
 会議 参加したいのですか  
 八に 参加されますか  
 八人  
 八

<絞り込み結果>

会議に 参加したいのですが  
 参加したいのですか

(23) どうすれば よろしいですか。

<文節ラテイス>

どうすれば 用紙ですか  
 五千が よろしいですか  
 五千の 用意しますか  
 五千も 用意しました  
 五千を 用紙です

<絞り込み結果>

どうすれば 用紙ですか  
 よろしいですか

(24) 先ず 登録用紙で 手続を していただくかなくては なりません。

<文節ラテイス>

先ず 登録用紙で 手続を していただくかなくては なりません  
 まられる 登録費として 手続も 在りませんが  
 七を 登録用紙が 手続は 七千が  
 七で 登録として 手続が なりません  
 登録用紙に 手続から 名前が

<絞り込み結果>

先ず 登録用紙で 手続を していただくかなくては なりません  
 なられる

(25) もう 登録用紙は お持ちでしょうか。

<文節ラテイス>

もう	登録用紙は	お持ちでしょうか
五を	登録用紙が	お持ちしては
五も	登録した	お持ちしたら
どうも	登録したら	思いましようか
二を	登録用紙とは	お持ちした

<絞り込み結果>

もう	登録用紙は	お持ちでしょうか
		お持ちしては

(26) まだです。

(27) 用紙を 送ってください。

<文節ラテイス>

用紙を	送って下さい
用紙と	送って下さる
四十を	お聞き下さい
	お送り下さい
	送っていただく

<絞り込み結果>

用紙を	送って下さい
用紙と	

(28) では ご住所と お名前を お願いします。

<文節ラテイス>

では	ご住所と	お名前を	お願いします
五は	五十七と	お名前も	お願いしますか
二は	五十七は	お名前と	お願いしない
七		お名前とも	お願いしたい
七を		お名前は	お願いしましう

<絞り込み結果>

では	ご住所と	お名前を	お願いします
五は	五十七と		



(29) 住所は 大阪市 北区 茶屋町 一の 二です。

<文節ラテイス>

住所は	希望させる	東区	徳井町	一の	二です
十七は	希望される	東区を	徳井町に	一も	二ですか
十四とは		東区も	送ります	一を	二ですね
十四は		東区で	送りたい	いつの	二十
十七が		東区が		一にも	二でしょう

<絞り込み結果>

絞り込み不可能

(30) 名前は 清水太郎です。

<文節ラテイス>

名前は	清水太郎です
名前が	清水太郎ですか
七は	清水太郎ですね
七には	清水太郎でしょう
名前とは	

<絞り込み結果>

名前は	清水太郎です
	清水太郎ですか

(31) 分かりました。

(32) 参加料は 要るのでしょうか。

<文節ラテイス>

参加料は	要るのでしょうか
参加料が	入れるのでしょうか
参加のは	四でしょうか
参加とは	言うのでしょうか
参加料を	入れましょうか

<絞り込み結果>

参加料は	要るのでしょうか
参加料が	

(33) はい。

(34) 登録費として お一人 三万 五千円が 必要です。

<文節ラティス>

登録費として	お一人	三万	五千円が	必要です
登録として	お二人	三万を	五千が	必要ですか
登録しました	お一人に	三万に	五千円は	七十六です
登録したい	お二人に	三万も	五十四が	四十六です
お持ちして	送りたい	三万の	五千には	七十六ですか

<絞り込み結果>

登録費として	お一人	三万	五千円が	必要です
				四十六です

(35) そうですか。

(36) どうも 有難うございました。

<文節ラティス>

五の	有難うございました
五も	有難うございます
どうも	
五のも	
五人を	

<絞り込み結果>

どうも	有難うございました
	有難うございます

(37) 失礼致します。