

文テンプレートハッシュ方式によるラティス解析方式と 質問-応答システム

屋野 武秀, 笹島 宗彦, 河野 恭之

株式会社東芝 関西研究所

本稿では、キーワードラティスを高速に解析する文テンプレートハッシュ方式 (BTH) を提案する。BTH は、ラティス上のノードが持ちうる文型集合を後続ノードに伝播し、後続ノードが持ちうる文型集合との積集合を計算して考える文型を絞り込む方式である。また、BTH パーザを利用した質問-応答システムを作成した。本システムでは、複数の候補について実際に問題解決を行い、その際に問題解決器が定める問題解決のし易さに基づき、意味スコアを考慮することが特徴である。音声入力実験により、BTH は系列数 100 億以上の規模のラティスを 0.2 秒以内で解析することや、質問-応答システムでは、音響スコアで下位にあった正解候補を上位に浮上させるなど意味スコアの効果を確認した。

BTH: An Efficient Parsing Algorithm for Word-Spotting and its Application to a Spoken Dialogue System

Takehide YANO, Munehiko SASAJIMA and Yasuyuki KONO

Kansai Research Laboratories, TOSHIBA Corporation

In this paper, we propose BTH, a parsing algorithm which is able to efficiently parse key-word lattice that contains large number of false candidates. A Spoken Dialogue system that employs BTH is implemented on PC. The system receives voice input as query, and understands it by analyzing candidates of BTH result based on semantic score. Simulation results show that BTH can parse large lattice which contains over 10 billion false candidates within 0.2 seconds, and efficiency of semantic score in our system.

1. はじめに

近年、コンピュータへの入力方法として、音声による入力が注目されている。音声は人間が情報を伝える自然な手段であるため、より指示を与えやすいインターフェースが構築できるものと考えられている。実際の音声入力では、不要語、言い淀みなどを伴うため、意味解析に必要な単語のみを抽出するワードスポッティング方式を音声認識方式として採用した音声理解方式が提案されている。ワードスポッティング方式による音声認識結果はラティス形式であるため、その後処理によって入力された文章(単語系列)を推定する。このような研究では、竹林らがタスクを小規模のものに限定し、非文となる接続関係を枝刈りしながら文法で受理可能な文候補(系列)に展開し、実時間の音声対話に成功している[1]。

実アプリケーションレベルでのタスク規模を持った音声対話システムを指向した場合、認識対象語彙が多くなる。対象語彙数が多くなると、認識結果のキーワードラティスの規模は必然的に大きくなる。このようなラティスを系列に展開すると、膨大な数の系列が現れ、それぞれが文法に適合しているか検証をするには膨大な計算量を要する。また、大きなタスクを扱う場合は文法規模も大きくなるため、受理する系列パターンが増加する。その結果、許容する単語間の接続関係が増加し、同じラティスからでも受理可能な文候補が増加するため、枝刈りによる系列数抑制効果が小さくなり、計算量の増加につながる。このような状況は、即応性が重視される音声対話システムでは望ましいことではない。

本稿では、大規模なラティスに対しても、高速に解析する文テンプレートハッシュ方式 (BTH) を提案する。また、BTH パーザを使用した質問-応答システムについて報告する。このシステムは、カーナビゲーション (以下カーナビ) タスクをモチーフとして、場所に関する質問に対して回答するため、問題解決方法の選好ルールを設けることによって、この選好ルールに適合する度合を意味尤度として計算し、音響尤度とあわせて考慮することを特徴としている。

最後に計算機シミュレーションにより、BTH の処理時間と、質問-応答システムの意味処理の効果について確認する。

2. 文テンプレートハッシュ方式 (BTH)

2.1. パージング問題

カーナビタスクのためのキーワード収集を行い、得られた 750 単語を対象語彙としてワードスポッティング方式の音声認識部への入力実験を行った。その中の一入力「出口の前の最後のサービスエリアはどこ」という音声から生成されたラティスは、図 1 のようになった。この入力中で対象語彙に含まれる単語は「出口、前、最後、サービスエリア、どこ」であるが、この例では、上記の単語の他に 30 個の単語が湧き出し、考えられる単語系列は約 400 万個にも及んでいる。これらの単語系列候補のそれぞれが、文法に適合しているかどうかを確認することは、処理量の面で現実的とはいえない。

このような大規模なラティスの構文解析を行い、後の音声理解・システム応答へつなげるために、次のような方針で BTH (文テンプレートハッシュ) 方式を設計・開発した。

- 実時間解析のためにはラティスを展開せずに解析することが必須である。
- 実際の音声入力は、書き言葉のように厳密な文法を記述することは困難であると考えられる。また、構文解析が可能であったとしても、後段のアプリケーションが解析結果を扱えなければ意味がない。あるいは、解析結果を扱える場合でも、再帰やループを伴い無限の文を生成する文法では、高度な処理を後段のア

プリケーションに強いることになり、アプリケーション実行時の負担が大きくなる。よって、有限個の品詞系列による文テンプレート (以下文型と呼ぶ) に基づく文法を使用する。このような文法は文脈自由文法などに比べ、文法記述の完全性、網羅性は損なわれる。しかし、倒置文受理の簡便さや後段の処理を考慮すると妥当な選択であると思われる。

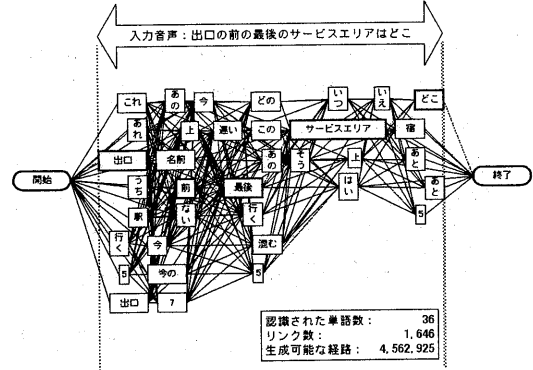


図 1 入力ラティスの例

2.2. アルゴリズム

BTH は、大きく 2 つの処理からなる。第一に、左から右に受理可能な文型の集合を絞り込む処理であり、第二に、最終的に絞り込まれた文型集合に該当する系列を抽出する処理である。文型絞り込み処理は、(1)ラティス上のノードが持ちうる文型集合を後続ノードに伝播し、(2)後続ノードが持ちうる文型集合との積集合を計算して考える文型を絞り込み、(3)絞り込まれた文型をその後続ノードに伝播し同様の処理を繰り返す、というものである。この処理に必要な集合演算の回数は、ノード同士のリンク数に比例する。以下にその詳細について述べる。

BTH パーザは、品詞辞書、文型辞書、文テンプレートハッシュ辞書、処理中ノードリストから構成される。

品詞辞書: 単語と品詞の関係を記述。

文型辞書: 受理可能な品詞系列と文型 ID とを記述。

文テンプレートハッシュ辞書: 文型辞書をコンパイルし、辞書に登録されている各品詞について文中の出現箇所と、所属する文型の集合との対応関係を要素とする表。つまり、「ある品詞

に属す単語が文中のある位置（順序）にあるならば、その単語を採用した文はどの文型に所属するか」という記述を要素を持つ。

処理中ノードリスト：現在文型絞り込み処理対象としうる入力ラティス上のノードのリスト。

また、ラティス上の各ノードには、以下の構成要素がある。

前ノードリスト：各ノードの前方に接続されているノードのリスト。

未処理前ノードリスト：前ノードリスト中で文型絞り込み処理を終了していないノードのリスト。

解析途中経過リスト：処理中の段階で、ある順序にそのノードに存在しうる文型 ID 集合を記録。

BTH パーザは構文解析の初期設定として、以下の手順を実行する。

- (1) 全てのノードの未処理前ノードリストに前ノードリストをコピー。
- (2) 単語系列の先頭となりうる単語のノードを処理中ノードリストに含める。
- (3) 処理中ノードリストに含めたノードの解析途中経過リストに初期値（文テンプレートハッシュ辞書から得られる）を設定する

この処理は、文頭にあたるノード上で考える文型をリストアップすることに相当する。

次に、文型絞り込み処理を行う。文型絞り込み処理は、処理中ノードリストが空になるまで以下の工程を繰り返す。

1. 処理中ノードリストから一つのノードを選択。
2. 未処理前ノードリストが空であるかどうかを確認する。空でなければ工程 1. に戻る。
3. あるノード(A)の未処理前ノードリストが空であれば、解析途中経過が確定しているので、ノード A に関する確定処理を行う。確定処理は、考えられる出現位置毎に解析途中経過リストと、その出現位置に該当する文テンプレートハッシュ辞書の要素との積集合を計算する処理である（図 2）。この確定処理の結果選られる解析途中経過リストの中身の文型 ID 集合が、ノード A までの全ての経路を考慮して得られる文型の集合となっている。そして、後続ノードに解析途中経過リストの内容を伝播し、ノード A を処理中ノードリストから除く。尚、伝播時には解析

途中経過リストに記録されている出現位置を一つ遅らせる。

4. ノード A の後続ノードを処理中ノードリストに加える。また、後続ノードでは、伝播されてくる解析結果の和集合を計算し、ノード A を未処理前ノードリストから除く。そして、工程 1. に戻る。

そして、上記 4. の処理が終了した時点で処理中ノードリストが空になると、終了ノードの解析途中経過リストに含まれている文型 ID 集合が、ラティスで受理可能な文型 ID の集合となっている。これで文型絞り込み処理が終了し、最終的に受理可能な単語系列は、上記文型 ID を利用して、受理可能な各文型に対応する単語系列をスキャンすることによって得られる。

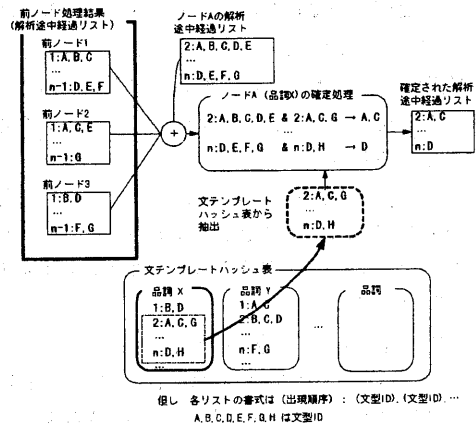


図 2 文型確定処理

2.3. 文法の階層化

BTH に適用される文法は、品詞系列の形式で現されなければならない。しかし、同じ意味を成す異なる部分品詞系列を利用した品詞系列をすべて別の文型として登録すると、受理する文型数が非常に大きくなるという問題が生じる。この問題は、文テンプレートハッシュ表の要素の大きさが膨大になり、パーズング時の集合演算の規模も膨大なものとなって、計算量が增大することにつながる。例えば、カーナビタスク向けに構築された文法では、「場所」に関する部分品詞系列が 200 個近く登録され、「場所 A から場所 B まで」という文の形式一つだけで

少なくとも約4万個の文型が必要となり、全体で120万個以上の文型が現われた。そして、要素数数十万という非現実的な規模の集合演算を計算する必要が生ずるという問題があった。

そこで、BTHを次のように修正した。

文法を句構造文法に変更：複数の品詞系列を一つのグループ(句)の中に登録し、句IDを一つの品詞と同等に扱えるように変更。また、文法で定義された各句に、その句を構成するために必要な他の句のIDを保持する必要句リストを持たせる。

文テンプレートハッシュ辞書の階層化：文テンプレートハッシュ辞書を各句毎に構成する。句に関する文テンプレートハッシュ辞書は「ある品詞に属す単語が句中のある位置(順序)にあるならば、その単語を採用した句はどの句の文型に所属しうるか」という記述を要素に持つことになる。

処理済句リスト：パーザに既に系列が得られた句IDを保持するリスト。これをBTHパーザの構成に加える。

そして、構成された各句ごとにパーズングを進め、最終的に文章として必要とする系列を得ることを目的とし、パーザの動作を以下のように動作させるように変更した。

1. 処理済句リストを空に初期化。
2. 必要句リストの中身が処理済句リストの中身に包含される句(処理対象句)を探索する。そして、処理対象句を文法として、前節のBTHアルゴリズムを適用する。
3. BTHによって得られた系列毎に対応するノードをラティス上に作成し、ノードの品詞を処理対象句のIDとする。また、処理済句リストに処理対象句のIDを登録する。
4. 上記2. 3. のステップを、処理対象句がなくなるまで繰り返す。

このように文法を階層化することにより、BTHを繰り返し行うことになり、集合演算の回数は句の数だけ倍増する。しかし、部分系列を一つにまとめたことにより、集合演算の規模を大幅に縮小させることに成功した。先ほどのカーナビタスクの例では、場所を現す「場所句」を作ることにより、約200通

りの部分品詞系列をまとめた。その結果、「場所Aから場所Bまで」という一文を記述するのに必要な文型は、場所句約200個と「場所Aから場所Bまで」用の文型1個に減少する。また、全体の文型数は120万個から場所句約200個と、場所句を利用した文章用の文型約500個の合計700個まで減少した。集合演算をその要素数に比例した回数と比較演算とすると、この場合は集合演算回数の倍増を考慮しても全体の比較演算回数は大幅に減少している。

3. 質問応答システム

次に、BTHパーザを利用した質問-応答システムについて述べる。この質問応答システムは、カーナビタスクをモチーフとして、場所に関する質問を受け付けるものである。構成は地図を表示し、経路を移動しながら現在地を更新する地図表示モジュールと、BTHパーザと問題解決器を含む対話モジュールの大きく2つに分けられる。

本システムは、BTHパーザが出力する単語系列候補列を受け取り次のように動作する。

- 候補の中で音響尤度の高いものから順に解析を開始する。
- 各候補に該当する文型ID及び句IDをもとに、系列候補をフレーム表現に変換する。この処理は、文型ID、句IDが決定すると、系列の並びが一意となり、部分系列、単語が入るべきスロットも一意に決定できる。
- 次に、問題解決器でフレーム表現から質問の対象となっている場所を検索する。この時、発話時の状況とマッチしていない質問の場合は、意味スコアを減点する(後述)。
- 各候補について、総合尤度として、音響尤度と意味スコアをあわせたスコアを計算し、最上位になる候補を選択し、回答を返す。

次節で、意味スコアの計算方法について述べる。

3.1. 意味スコアの算出

本稿では、意味スコアを「質問されている状況に適合している度合い」と定める。状況に適合しているかどうかは、解決しようとする問題によって異なるので、各問題解決器が問題を解く時に適合度合

いを計算する。本システムで使用する問題解決器である、場所検索器での意味スコア計算ルールの概要を次に挙げる。

- (R1) 設定経路上に存在する場所を優先して選択する。設定経路外の場所を指定する表現は大きく減点する。
- (R2) 設定経路上では、場所指定を唯一に指定できない表現は減点する。
- (R3) 設定経路上で指示対象が複数存在する場合は、利用者が何らかの動作をするところ(ガイドポイント)を優先して選択する。ガイドポイント以外の場所を指定する表現は減点する。

このように、カーナビタスクにおいて、利用者は設定経路上にある場所で、かつガイドポイントに関する質問がメインであると想定し、場所検索器を設計している。場所検索器に関する意味スコア計算基準は、上記想定に基づく回答として与える場所の選好基準を与え、その選好基準に合えば状況に適合するとしており、言い換えれば「問題解決のし易さ」を意味スコアの尺度としている。

4. シミュレーション

本質問-応答システムをPC(CPU:PentiumII 375MHz DualCPU)上に実現し、音声入力による簡単なシミュレーションを行った。

対象とした文法はカーナビタスクをモチーフとしたもの(場所句文型数約 200, 全体文章文型数約 500)である。また、発話者は男性 1 名で、文型リストから無作為に取り出した文型に適合する文章 50 文を提示し、読み上げを依頼した。

最初に BTH パーザの処理時間を測定した。対象となるキーワード数は 750 単語で、音声入力を 15 回行った。スポッティングされた単語数は 20~100 個であり、文型に必要な単語数の 10 倍~30 倍の湧き出し誤りが生じていた。この条件での BTH の処理時間を表 1 に示す。

この表より、BTH パーザは、平均 10 億個近くの系列を保持するラティスの解析を約 0.1 秒で終わっていることがわかる。また、実験中に現れた最大のラティスは 100 億個以上の系列を保持していたが、そのラティスの解析を 0.2 秒で終わっており、ラティスの規模によらず安定して高速な解析が可能である

といえる。

表 1 BTH パーザの処理時間

	平均	最大
単語数	72	100
系列数	9.5×10^9	1.09×10^{11}
処理時間(msec)	105	172

次に、質問-応答システム内の意味処理の効果を調べた。対象とするキーワードは、場所を問うタスクに必要な約 300 単語で、音声入力を 50 回行った。また、意味スコアに加減量については、実験的に決定した。そして、正解基準を、入力意図表現と音声入力内容が一致することとして実験した。結果を表 2 に示す。

この表において、音響順位はパーザを通過した際に与えられる正解候補の音響スコアの順位で、意味順位とは、音響スコアに加えて意味処理によるスコアを加味した正解候補の総合順位である。また、音響順位の「欠落」とは、必要なキーワードがスポッティングされなかったことを指す。また、各順位で、二位以下の順位をすべて下位とまとめた。表 2 より、意味処理を加えることによって、音響スコアが下位であった正解候補が一位に浮上する例が 8 例(16%)見られ、意味処理時のスコアの見直しによる効果が見られる。

表 2 意図処理結果

音響順位	意味順位	
1位	1位	22
下位	1位	8
1位	下位	2
下位	下位	6
欠落		12

5. 考察

従来にも、音声理解を目的としたパーザの研究がなされてきた。竹林らはタスクを小規模のものに限定し、拡張 LR パーザに基づく手法で系列(文候補)を作りながら解析を行っている[1]。その処理時間は、作成中の文候補のビーム幅に依存する。しかし、大規模なラティスに対しては多数の文候補が生

成される可能性があるため、ビーム幅を大幅に絞り込まないと、実時間でパーザが動作しない可能性がある。また、永井らが提案する音声理解方式[2],[3]は、鳥駆動方式のラティス理解方式に基づき、言語知識を導入することで系列生成の枝刈り効率を向上させており、1000単語からなる観光案内などのタスクの自由発話音声理解システムを準実時間で動作させている。

一方、BTHは、有限種類の正解系列を定義する文法を対象としているため、完全な自由発話理解のための文法よりは狭い文法を扱うことになるが、文法で受理可能な全ての系列を高速に探索することが可能である。また、自由発話理解に向けて文法を拡張することや、高精度な意図理解に向けて言語知識を考慮した文法を構築することも、品詞の細分化や文型の追加などによって可能である。しかし、追加、削除などによる副作用を十分に考慮するなど、文法の整備あるいは整備方法が課題となり、今後検討が必要であると考えられる。

質問-応答システムに関しては、音響尤度の低い正解系列を一位にする例が見られた。

発話例「吹田インターの後のインター」

1位：吹田 インター 前 インター (文1)

2位：吹田 インター 後 インター (文2)

これは、順序を表す自立語「後」にかわって「前」が湧き出し、音響スコアで一位になった例である。カーナビタスクで指定した設定経路では「吹田インター」から高速道路に乗る経路となっているため、この場合「吹田インターの前のインター」は経路上には存在しない。よって、意味スコアルール(R1)が適用され、(文1)の意味スコアは減点され、音響スコアで二位であった正解文章(文2)が一位に浮上した。このように実際に問題解決を図ることで、実際には考え難い意味合いの文章を下位に下げることが可能であると考えられる。経路外にある固有名詞が湧き出した例でも同様の結果が得られるなど、本意味処理方式で一定の効果が見られた。

しかし、音響スコアで一位だった正解文章が意味処理によって下位に転落した場合もある。その例として、「入口」などの一意に決定できない表現にかわり経路上に存在する固有名詞が湧き出し、正解文章に(R2)が適用され、減点されるケースがあった。

このような副作用を伴うため、本方式の有効性は意味尤度計算ルール設計によるところが大きい。また、システムの選好基準と一致しない入力があることも考えられる。この場合、音響スコア上位の候補が意味処理によって下位に転落することが予想されるため、そのような場合に問い返しを図るなどの処置を考慮する必要がある。問い返し内容に問題解決状況を反映させるなどの問い返し戦略は今後の重要な課題となる。

6. まとめ

キーワードラティスを高速に解析する文テンプレートハッシュ方式(BTH)を提案した。BTHは、ラティス上のノードが持ちうる文型集合を後続ノードに伝播し、後続ノードが持ちうる文型集合との積集合を計算して考える文型を絞り込む方式である。この方式は、ラティスを展開することなく解析を進めるので、大規模なラティスに対しても安定して高速に解析可能である。また、BTHパーザを利用した質問-応答システムを作成した。本システムは、複数の候補について実際に問題解決を行い、その際に問題解決器が定める問題解決のし易さに基づき、意味スコアを考慮することが特徴である。計算機シミュレーションにより、BTHは系列数100億以上のラティスを0.2秒以内で解析することを示し、質問-応答システムについては、音響スコアで下位にあった候補を上位に浮上させるなど、意味スコアの効果を確認した。

今後の課題として、BTHに関しては文法の整備方式の検討と、質問-応答システムに関しては、問い返し戦略の検討を行う予定である。

参考文献

- [1] 坪井, 竹林, 橋本: "キーワードラティスのLR解析による自由発話理解", 情報処理学会論文誌, vol.38, No.2, pp.260-269(1997-2)
- [2] 永井, 石川, 中島: "概念素に基づく意味理解における音声認識への統合手法", 情処研報, SLP7-4(1995-10)
- [3] 永井, 石川: "対話システムのための概念素理解方式による対話音声理解", 情処研報, SLP18-5(1997-10)