日本語会話文の構文木付コーパス作成

河田康裕、金城由美子、柏岡秀紀
ＡＴＲ音声翻訳通信研究所

1 はじめに

近年、電子化テキストや品詞タグ付コーパスの入手が容易になるにつれて、統計的知識や用例を用いた自然言語処理技術の研究が盛んになってきた[2, 7]。そういった研究の材料として、言語の構造に関する情報のデータは、益々有用である。構文木付コーパスが大量にあれば、言語の構造に関しての統計情報や実例が手に入る。構造に関する記述のついた大規模な言語データも整備され、入手可能となってきた[4, 3, 5]。英語については、ＡＴＲ音声翻訳研究所でも、分野限定せずに英語のテキストを収集し、詳細な情報を付与した約80万語の構文木付コーパス“The ATR/ Lancaster General English Treebank”[1]を作成した。

談話の研究や、会話の翻訳のように実用場面を想定した自然言語処理の研究では、音声で交される実際の会話に近い、話し言葉の言語データも重要である。本稿で報告するＡＴＲ日本語旅行会話コーパスは、日本語の話し言葉の構文木付コーパスである。これは、会話の書きおこしテキストを、形態素分割し、品詞タグを付与し、そして木構造を表現するラベル付括弧を付与したものである。

これは、統計処理に基づく言語解析（形態素分割、品詞タガー、構文解析器）[2, 7]の訓練データとして利用する。以下では、その特徴、作成、利用、今後の課題などについて報告する。

2 ＡＴＲ日本語会話構文木付コーパス

2.1 特徴

ＡＴＲ日本語会話構文木付コーパスの素材となったテキストの、タスク領域は、旅行会話[1]である。日本語の会話文に構文木を付与した大規模なコーパスは、公開されていない。新聞等のように整った書き言葉の文と比べると、自然な話し言葉は、平均的に短く[2]単純な構文構造の文が多いが、一方、書き言葉にはみられない、断片的な発話や省略、発話途中で意図する構文が変ってしまったと見られるような表現もある。内省的観察では、有り得ないとされるような経験的事実も含まれている。

構文木付コーパスに付与する木構造は、人間が判断して記述するが、構文解析器を使用することにより、一貫性のある揺れの少ない構文木を記述できる。構文解析器による解析結果が複数ある曖昧な構造の文も、構文木付コーパスでは、現れる文脈の中で人間が判断して、最適な構文木が一つ与えられている。構文木は、コーパスに現れた文に対する文法規則の適用履歴でもある。

2.2 テキスト収集と形態素解析

コーパスの音声会話は、4カ国語[3]で収録された。会話に関する設定内容（例、氏名、年齢、性別、日時、話者の目標、状況）のみを示した資料に基づいて行った会話や、その話題について標準会話をあらかじめ話者に提示し、会話内容を把握してもらい、後にそれを見ずに行った会話、などが収録された。収録された4カ国語の会話は、テキストデータに書きおこされ、データ管理のための属性（例、会話ＩＤ、収録日時）が記録されている。

テキストデータは、定められた仕様に従い、文分割、形態素分割、品詞タグ付与が行われ（図1）、分割の単位や付与された情報は、十分な精度が得られるまで繰り返し改良された。ここで述べる日

```
20|0020|30|70| 部屋 | ヘヤ | 部屋 | 普通名詞 | | | |
20|0020|30|80| の | ノ | の | 連体助詞 | | | |
20|0020|40|90| 予約 | ヨヤク | 予約 | サ変名詞 | | | |
20|0020|40|100| を | ヲ | を | 格助詞 | | | |
20|0020|50|110| お | オ | お | 接頭辞 | | | |
20|0020|50|120| 願い | ネガイ | 願う | 本動詞 | 五段ワ | 連
20|0020|50|130| し | シ | する | 補助動詞 | サ変 | 連用 | |
```

図1: 形態素情報の例

[1] ホテルや交通機関の予約・交渉、観光案内所での対話、など。

[2] 約一万文を抜き取り調査した結果、文長は1から44形態素、平均10.7で、4形態素からなる文が最も多く、約9.4%を占める。

[3] 日本語、英語、ドイツ語、韓国語

本語会話構文木付コーパスの日本語品詞タグは、4カ国語間の機械翻訳システム[9]で用いられている品詞である。現在のところ、統語的振る舞いに基づく基本品詞と、活用形情報の組み合わせを含めて数えると200種類余りのタグ集合である。将来は、意味的分類を反映して品詞体系を詳細化する予定である。

2.3 構文分析

一般に、さまざまな言語データ作成においては、自動的にできるところを機械的に処理し、できないところを人間が処理する手法が、コストが低く、データが均質で精度も高い[4]。

また、機械的な構文解析の一般的な問題は、曖昧さの解消である。構文解析器により一意に解された間違った解析結果を人間が修正するのは、かえって困難なので、曖昧な部分は、

(a) 構文解析器では何もせず、人間が解析[4]

(b) 構文解析器で全解を求め、人間が選択[1]

のいづれかが一般的である。本報告では、(b)のアプローチを採用する。いづれの場合も、前処理を行う構文解析器の精度を徹底改良することは、重要である[5]。また、人間とのインターフェースも、全体の効率に大きな影響を与えると考えられるので、使い易いツールの提供も重要である。

日本語の構文の分析は、ツリーバンカーとよばれる構文分析専門の担当者が行う。ツリーバンカーは、計算機上の構文解析器が提示する複数の構文解析結果の中から、最適解を一つ選択して構文木付コーパスに加える。計算機上の構文解析器は、素性構造付文脈自由型文法規則を参照している。

計算機による構文解析は、膨大な数の解析結果を返す場合があり、その中から最適解を選択することは、人間にとっては非常に困難な作業に思えるが、方法によっては、困難さをかなり軽減できる。例えば、千通り以上の構文解析結果の中から、一つの解を選ぶのは、構文木の非終端記号のうち、5箇所に4通りづつの曖昧さがあれば、解は千通りを越えている。ツリーバンカーは、それらの曖昧さを孕む各非終端記号の四選択一に、最悪の場合でも5回の判断をすれば、最適と判断される解に辿りつくことが可能である。実際には、曖昧な非終端記号は相互に関連しており、一つの非終端記号の曖昧さを解消すれば、別の非終端記号の曖昧さが同時に解消する場合が多い。このように、多数の構文木から妥当な構文木を選択する作業の効率を上げるためには、使い易いツールが必要であり、以下で述べるツールが開発された。

2.4 GWBTool

ツリーバンカーが使用するツールは、GWBTool (Grammarian's Work Bench)である。これは、X-Windowsのアプリケーションツールで、ツリーバンカーが、早く正確に構文木を記述するために開発された。GWBToolは、前述の"The ATR/Lancaster General English Treebank" [1]の作成にあたり開発されたものを、日本語文も扱えるように改良したものである。

GWBTool（図2）は、品詞タグ付のテキストを読み込んで表示する（図2中段のウインドゥ）。解析する文を指定すると、指定された文が黒地に反転し、その1文が上段のウインドゥに表示される。解析を指示する操作を行うと、構文解析器は、全ての可能な構文木（parse forest）を計算する。そして、もう一つウインドゥ（図2右下）が開いて、構文木が一つ表示される。その構文木は、二つ以上の部分構文木の可能性のある構文節点は、曖昧さのない構文節点とは別の色で表示されるので、どの構文節点が曖昧であるか、一見してわかる。二つ以上の部分構文木の可能性ががある構文節点をクリックすると、選択肢として、別の構文節点名のリストがポップアップ表示される。そのリストから一つを選択すれば、それに応じて部分木が描き換えられる。また、別のボタン操作で、各構文節点の素性値をポップアップ表示させることができる。

ツリーバンカーは、任意の構成素の分割を指定をして、構文解析結果を制限することができる。解析文中に手動で括弧を挿入することにより、特定の構成素分割を指定すると、GWBToolは、指定された構成素分割と矛盾しない[6]解析結果だけを計算して返す。同様に、文の一部をマウスで指定して、予め構文解析を実行し部分木を作る。その部分解析結果と矛盾しない全体の解析結果だけを計算させることもできる。ツリーバンカーは、構文解析器による解析結果の傾向に慣れてくると、構成素の指定や部分解析等を予め行い[7]、計算機による余分な構文解析を繰り返し実行したり、曖昧な構文節点での選択操作を何度も行ったりすることもなく、少ない労力で目標とする最適の構文解析結果に到達できる。

---

[4]品詞タグ付けにおいて、人手だけで行うと、機械による前処理をした場合と比べて時間は約二倍かかり、作業者間の不一致率と誤り率は、ともに5割増しになるとの報告がある[4]。

[5]京都大学テキストコーパス・プロジェクトでは、「解析システムそのものの改良を徹底的に行」[8]っている。

[6]両者間で交差する左右括弧対が存在しない。

[7]例えば、等位構造の範囲や副詞の修飾先の範囲は、括弧を挿入して予め指定してから構文解析を実行する。

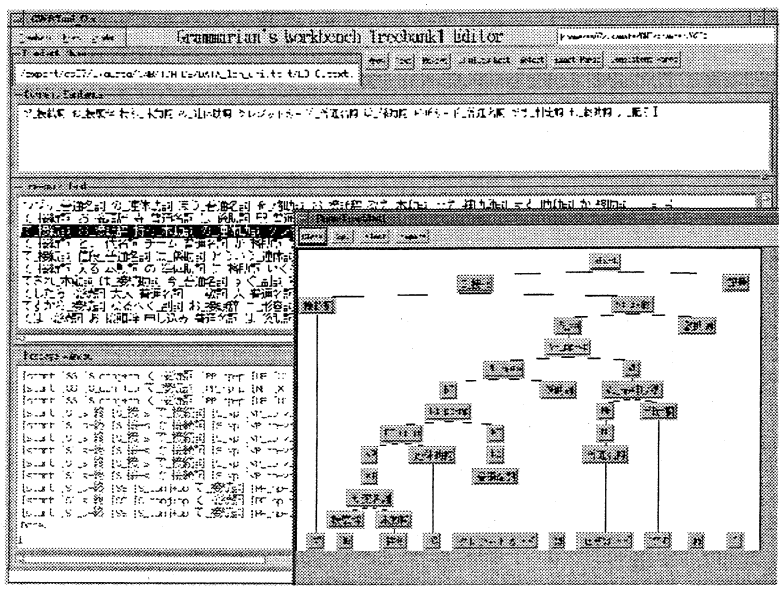

図 2: GWBTool の画面

2.5 構文木付コーパス構築の経過

GWBTool を使用して、品詞タグ付テキストデータから構文木付コーパスを構築する際の、これまでの作業の経過を述べる。GWBTool のアルゴリズムやソフトウェア作成の担当者、構文解析器の文法を作成の担当者、日本語の記述分析の担当者が、定期的に会合を持ちながら構文木付コーパス作成を進めている。

(1) **構文解析器の文法規則作成** 目的コーパスの文の中から、異なる文長の文、様々な表現のサンプルを抜き出して、GWBTool の構文解析器が参照する文法規則を作成[8]。

(2) **構文木の記述** ツリーバンカーが、GWBTool を使用して構文木を記述し、構文木付コーパスを作成。

(3) **構文解析器の文法規則改良** 初期の段階では、解析するコーパスの中に、GWBTool の解析器の文法が予測していない言語現象が現れるので、結果が出ない、結果の候補中に妥当な解がない、又は、おびただしい数の曖昧さが出る、等が起きる。その場合、文法開発者とツリーバンカーが議論して、妥当な構文解析結果が出せるように GWBTool の構文解析文法を、改良変更（(1) へ）。

この場合、データの一貫性と均質さを保つため、記述済みの構文木付コーパスの中で、変更された文法規則が関わる可能性のある部分は、再度分析（(2) へ）。

以上のサイクルを繰り返し続けて行くと、GWBTool の文法規則が成熟し、(1) と (3) が少なくなり、(2) が長期間続く状態になっていく。

現在まで、およそ半年で約 15 万語[9]の日本語会話文構文木付コーパスを作成した。引き続き、更に約 1 年間かけて、1 年半で総計約 40 万語の構文木付コーパスを作成する予定である。

3 構文木付コーパスの利用

確率付決定木を用いた言語解析処理 [2, 7] では、形態素分割、品詞タグ付与、そして構文解析を行う各々の決定木の学習を行うために、相当量の正解データが必要である。構築した構文木付コーパスは、これらのシステムの、訓練データとして使用している。システムは、構文木付コーパスに現れたそれぞれの分析記述を、事実として学習し、それらの事実に照らして、最も尤度の高い決定を

[8] 100 余りの規則であるが、素性により一般化した規則なので、規則数には大きな意味はない。全部で 17 の素性があり、ほとんどの非終端記号は、4 つ以上の素性について値をもっている。

[9] 構文木記述作業に先行したソフト開発、文法開発の期間は含まない。2 名のツリーバンカーの作業による。

行う確率付決定木を、エントロピーの減少に基づくアルゴリズムにより作り上げる。システムは出来上がった決定木を使って未知の事例の言語処理を行う。これは、非常に頑健なモデルである[10]。

現在、ＡＴＲ音声翻訳研究所では、約80万語の“The ATR/Lancaster General English Treebank” [1]を使って、英語の品詞タガーと英語構文解析器の確率付決定木を学習し、それぞれが動作している。この英語構文解析器は、試験的に英語版GWBToolにも実装された。これを用いると、ツリーバンカーは、全解の中から最適解を一つ選択する場合にも、尤度の高い順に提示される解の中から選択することができる。

日本語では、約22万語の構文木なしの品詞タグ付コーパスを使って学習した確率付決定木モデルの日本語形態素分割と品詞タガーが動作している[7]が、今後さらに、現在作成中の日本語構文木付コーパスを使って、

- 日本語形態素分割
- 日本語品詞タグ付与
- 日本語構文解析器

の各々の確率付決定木の学習訓練を行う。既に動作している形態素分割と品詞タグ付与でも、構文構造に関する情報を使うことにより、現在の性能が向上することが予測される。

これまで、構文木付コーパスのデータ作成のために使用した形態素分割・品詞タガー・構文解析器は従来手法のものである[11]が、今後、確率付決定木モデルを導入した品詞タガー、構文解析器の評価結果が向上すれば、それらを利用する。

4 おわりに

将来の実用的な自然言語処理システムの研究開発において、重要な役割をはたすと考えられる日本語会話文の構文木付コーパス作成について述べた。会話に特有の現象をも含んだ多量のテキストデータを、早く正確に分析記述して構文木付コーパスを作成するには、頑健な解析器を組み込んだ使い易いツールが役立つ。

我々は、データを作成すると同時に、それを利用する言語処理システムの研究開発にも取り組んでいる。GWBToolの構文解析器に決定木モデルを組み込む試みもされた。訓練データの蓄積によって、品詞タガーや構文解析器の性能が向上すれば、訓練データを作るコストが下がり、さらなる訓練データの蓄積を可能にしていくと考えられる。

これからの課題は、40万語の日本語旅行会話構文木付コーパス作成とともに、そのコーパスで学習した、確率付決定木を用いた日本語形態素分割、品詞タガー、日本語構文解析の実用化である。これらのシステムを構文木付コーパス作成の過程に組み込んで、構文木付コーパス作成過程の中で評価し、訓練データ量とシステムの性能の関係を明らかにしていく。

[10] [7]を参照。

[11] 形態素分割・品詞タガーはn-rigramモデル。構文解析器はチャート法の全解探索モデル。

参考文献

[1] Black, E., H. Kashioka, S. Eubank, R. Garside, G. Leech and D. Magerman. (1996) “Beyond Skeleton Parsing: Producing a Comprehensive Large-Scale General-English Treebank With Full Grammatical Analysis”, Proceedings of COLING-96, pp. 107-112.

[2] Black, E., S. Eubank, 柏岡秀紀. (1997) “The Non-Dictionay: Description and Evaluation of a Dictionaryless Semantic and Syntactic Tagger for Unrestricted English Text” 言語処理学会第3回年次大会発表論文集 pp. 309-312.

[3] EDR (1996) *EDR Electronic Dictionary Version 1.5 Technical Guide.* EDR TR2-007.

[4] Marcus, M., Santorini, B., Marcinkiewicz, M.A. (1993) “Building a large annotated corpus of English: the Penn Treebank.” *Computational Linguistics, Vol 19.*

[5] Miyoshi, H., K. Sugiyama, M. Kobayashi, and T. Ogino. (1996) “An Overview of the EDR Electronic Dictionary and the Current Status of Its Utilization” *Proceedings of COLING-96.*

[6] 柏岡秀紀, Stephen Eubank, Ezra Black. (1996) “The ATR/Lancaster General – American – English Treebank”, 言語処理学会第2回年次大会発表論文集 1996, pp. 269-272.

[7] 柏岡秀紀、河田康裕、金城由美子、A. Finch、E. Black (1998) 『確率付決定木を用いた日本語構文解析』言語処理学会第4回年次大会発表論文集.

[8] 黒橋禎夫、長尾眞 (1997) 『京都大学テキストコーパス・プロジェクト』言語処理学会第3回年次大会発表論文集 pp. 115-118.

[9] 古瀬蔵、隅田英一郎、飯田仁 (1994) 『経験的知識を活用する変換主導型機械翻訳』情報処理学会論文誌 vol. 35 No. 3.