人工知能学会誌 Vol. 15 No. 5 (Sept. 2000) [別刷]

AImap

言語理解
—SHRDLUの先にあるもの—

Understanding Natural Language
- Beyond SHRDLU -

田中 穂積
Hozumi Tanaka
東京工業大学大学院情報理工学研究科
Graduate School of Information Science and Engineering, Tokyo Institute of Technology.
tanaka@cl.cs.titech.ac.jp, http://tanaka-www.cs.titech.ac.jp/

Keywords: natural language understanding, natural language processing, parsing, rule-based, corpus-based, dependence relation, machine translation.

1. はじめに

Understanding Natural Language という言葉をはじめて耳にしたのは，スタンフォード大学の人工知能研究所に 1971 年から 1 年間滞在した折りに，同室の学生が面白いといいながら読んでいた Terry Winograd の分厚い博士論文 (MIT AI TR-17, Procedures as a Representation for Data in Computer Program for Understanding Natural Language) であった．その後 Cognitive Psychology というジャーナル一冊すべてがこの論文に割かれたりして心理学の分野にも大きな影響を与えたことがわかる．学術書としても出版されている [Winograd 72].

当時たまたま MIT を訪問する機会を得たので，人工知能研究所に滞在していた白井良明氏 (現大阪大学教授) に会い，Winograd の Technical Report (TR) が欲しいと頼んだところ，彼は「何しろ人気の TR なのであるかなー」といいながら薄暗い部屋に入り，山積みの TR の中から所望の TR を捜し出してくれた．このシステムは SHRDLU とよぶロボットが，英語の指令や質問を理解して積木の世界の様子を教えたり，積木を移動するなどの仕事をし，その様子を動画として画面に表示する．言語理解システムとはどのようなものかを，われわれにわかりやすく教えてくれる研究であった．

Winograd の論文を読む前に，スタンフォード大学で，ある授業に出ていたら，授業担当の教授が Roger Schank の授業を盛んに勧めるので聴講することにした．この授業は 3 回目あたりになると 5 人程に減ってしまい，抜け出すのも悪い気がして最後まで出席した．講義自体は大変面白く，たとえば "I saw the Grand Canyon flying to New York." という文を，彼の理論に基づき計算機に理解させるというものであった．グランドキャニオンが New York に飛ぶはずはないという知識があってはじめてこの文が理解できる訳である．これも当時最先端の言語理解の研究であり，そこには現在にも通じる問題が含まれている．

この分野の先人である二人の研究を目のあたりにして，これは面白いし奥の深いテーマのようなので，自信はなかったけれども，将来取り組んでみたいと思い，電子技術総合研究所に帰ってから，当時の上司であった渕 一博室長に相談した．新しい視点で取り組むのであれば進めて良いということになった．それ以来，渕 室長 (現東京工科大学教授) には研究を進める上で貴重な助言や示唆をいただいたことが忘れられない．

2. 1960 年代——機械翻訳と人工知能研究

前章で述べたように 1970 年代は，言語理解が人工知能研究の最前線の研究課題になっていた．ここで少し時計の針を逆転させて 1960 年代の様子を振り返ってみたい．

1960 年代初期のコンピュータの性能は極めて不十分であったにもかかわらず，機械翻訳の研究が行なわれた．新しい文法理論が登場した時期と重なったこともあって，それを利用すれば，困難が予想される意味理解の問題を避けたとしても，かなりの程度の機械翻訳が可能であると考えられていた．こうした楽観論の背景として，欧米では語順が似通った言語間の機械翻訳に取り組んでいたことがあるかも知れない．翻訳辞書を参照するだけで，単語毎の翻訳を行ない，それらをそのまま並べても，ある程度意味が通じる文 (翻訳結果) が得られることが多いからである．

1960 年代の始めには，わが国では電気試験所 (現電子技術総合研究所) で和田 弘博士が，「(数値) 計算しない計算機」の実現を標榜した英日機械翻訳システム「やまと」を開発していた．九州大学，京都大学での研究がこれに続く．「やまと」では英語と異なる語順をもつ日本語を対象とするため，個々の単語を翻訳してそれらを並べただけでは意味をなす翻訳結果が得られない．少なくとも文

の構文解析を行ない，主語や目的語の認定を行なう必要がある．日本語を対象とする場合にはこのように構文解析の問題は避けて通れない．

こうして初期の機械翻訳の研究を通じて，わが国における自然言語処理技術の研究が始まった．欧米ではプログラム言語のコンパイラの研究の一環として，高速な構文解析アルゴリズムの開発に拍車がかかり，それがまた自然言語の解析にも利用されることになる．文脈自由文法(CFG)に対するCYK法，チャート法，LR文法に対するLR法など，構文解析アルゴリズムの主なものは1960年代にほぼ出そろっている．

1960年代は人工知能研究の幕開けの時代でもあったが，この頃，意味の問題が人工知能研究の鍵であるとする考え方がMITの研究者を中心に生まれてきていた．これは“Semantic Information Processing”というスローガンのもとで推進された[Minsky 68]．このスローガンは，言語にもっとも似合いのスローガンでもある．

しかし当時も今も，意味は扱いが困難であることに変わりない．そこで当時の人工知能の研究者がとったアプローチは，対象世界を狭くとり，使われる語彙や知識の数を絞り込むことであった．それにより手作業でそれらに豊富な意味記述を与え，解析のレベルを構文から意味に深めることが可能になる．

1960年代後半から1970年にかけてのWinogradの研究は，積木という極めて狭い世界を対象にして，コンピュータによる意味理解，言語理解とはどのようなものか，そこでは推論や知識がどのような役割を果たすかということを我々に示すことができたのである．

それとは対照的に当時の機械翻訳の研究では，意味理解，言語理解の問題をできるだけ避けようとする姿勢が強かった．それは当時としてはやむをえない選択でもあった．機械翻訳では対象世界を狭くするといっても，ニュース文や科学技術文献を対象にしており，積木の世界とは比較にならないほど多様な言語現象を扱わなければならなかったからである．語彙項目も相当数にのぼり，それらすべてに豊富な意味記述を与えることは手作業では不可能であった．

こうした事情もあって，機械翻訳の研究は次第に行詰まりの様相を呈する．大規模電子化辞書の構築，言語理解のための知識ベースの研究，計算言語学の推進などが急務であり，機械翻訳については基礎研究を重視すべきであるという報告書が60年代半ばに米国でまとめられ，米国での機械翻訳の研究はそれ以後冬の時代に入る．

3. 1970年代——規則ベースの時代

1970年代に入り，言語理解の研究は，対象世界をデータベース検索に絞った質問応答システムの実用化を目指す方向と，Schankらのように，コンピュータによる意味理解のための理論構築を目指す方向とに分かれて，人工知能の研究分野で活発に行なわれた．後者は認知科学という新しい学術分野の誕生を促した．

言語理解の研究では，コンピュータで扱うことが可能な知識表現が重要な研究課題となり，文法としては文脈自由文法規則(CFG規則)に手続き(プログラム)を付加して補強したものを用いることが多かった．補強により，CFGを越えた能力を文法に持たせることが可能になるからである．1980年代半ばまでの自然言語の研究の特徴は規則ベースにあった．

3·1 Lingol

筆者が言語理解の研究分野に足を踏み入れたのは1970年代半ばであった．たまたま電子技術総合研究所が招待したMITのMoses教授から，数式処理システムMACSYMAで使っているLingolとよぶ構文解析システムの存在を教えて頂いた[Pratt 75]．さっそくそのシステムを取り寄せたが，MacLispで書かれていた．プログラムの解読と，電子技術総合研究所で開発した電総研LISPシステム上への移植を行なって動作させてみた．

Lingolはなかなか良くできたシステムであった．特に解析結果を視覚的に表示したり，文法規則や辞書項目の追加と修正が容易に行なえる機能がついており，言語処理用ツールとしての使い勝手が工夫されたシステムであった．結果を急ぐあまり，こうしたソフトウェアツールの構築に時間をかけるのを厭ってはならない．ソフトウェアの開発には「急がば回れ」が必要なのである．

Lingolには横型探索法による最高速の構文解析アルゴリズムが実装されていた．構文解析結果はCFG規則を組み合わせた木構造となるが，CFG規則には関数が補強されているため，先の木構造と同形な関数木ができあがる．関数木のトップレベルの関数を評価すると，各節点に存在する関数が次々に起動され，Syntax駆動の意味解析を行なうことができる．この機構はLisp言語の特性をうまく利用しており感心させられた．この考え方はモンタギュー文法の意味解釈の方法とも親和性をもつものであった．横型探索法を採用しているので，解析結果同士の比較が容易になるのも好ましい特徴であった．

3·2 拡張 Lingol

Lingolを使っているうちに，日本語文をわかち書きして与えなければならない点に不便を感じた．そこでLingolの内部に手を入れて，CFGの制約を使いながら形態素解析を行ない，日本語文のわかち書きと同時に構文解析を行なえるようにした．形態素解析と構文解析を同時に行なう試みとしてはわが国はじめての試みであったと思う．この時のテスト文は「すももももももものうち」であった．これを拡張Lingolと名付けてプログラムソースを一般に公開した[田中 77]．

Lingolでは，構文解析が終了して完全な関数木が出来上がってからその評価に移る．そこでLingolのもう一つ

の拡張として，構文解析中途のCFG規則適用時にCFG規則に付加した関数を起動し，その結果を利用して構文解析過程を動的に制御する機構を付加した．これは元吉文男君(現電子技術総合研究所)がたちどころに実装してくれた．これを1979年の人工知能国際会議で発表したところ，エジンバラ大学のBundy教授が聞いていて，同様なことを同僚がPrologで行なっていると教えてくれた．同じことならそれほど気にする必要もなかろうとそのままにしておいた*1．

4. 1980年代——Prolog,DCG,BUP,SAX

4·1 DCG

一方渕氏は，エジンバラ大学のTRとして発表されていたDCGの論文をすでに手に入れ読んでいた．このシステムを入手して動かしてみようということになった．1980年のことであったと思う．エジンバラ大学に手紙を書いたところ，すぐにシステムを送ってくれた．DCGは論理型プログラム言語のPrologシステムに組み込まれていたので，コンパイラ付きのエジンバラ版Prologが電子技術総合研究所で動作することになった．

DCG形式の文法規則では，CFG規則に現れる文法記号が引数付きの形をしている．それらを利用して必要ならPrologプログラムによる補強もできる．DCG形式の文法規則はPrologに組み込みのトランスレータによりPrologプログラムに変換して動作させる．動作はユニフィケーションをベースにしたPrologの基本計算機構をそのまま巧みに利用する．こうして構文解析や意味解析ができる．DCGは，実際に動作し得る文法の枠組であり，言語のもつ種々の制約を宣言的に記述することができる．言語理論の新しい潮流となるユニフィケーション文法(HPSGなど)の有用性を実証する役割も果たした．

Prologプログラムを初めて作ることになり，例題として，かつて日本初の機械翻訳システム「やまと」が対象とした文の英日翻訳を試みた．DCGでプログラムを記述するとそれは即座に完成し時代の進歩を感じた．そのプログラムに日本語を入力すると英語の翻訳結果が得られる．同じプログラムが英日翻訳にも日英翻訳にも使える．これは理屈でわかっていても，いざ動かして結果が出ると驚くものである．拡張Lingolで可能なことが，もっとスマートにDCGで可能になるということで，それからしばらくDCGの世界にはまりこんだ．同じようにPrologと自然言語処理の世界に夢中になったのが松本裕治君(現奈良先端科学技術大学院大学教授)である．

4·2 BUP, SAX など

DCGをしばらく動かしているうちに，これはトップダウン縦型探索の構文解析を行なうので，左再帰のDCG規則があると具合が悪いことが気になりはじめた．これを避けるために，ボトムアップな構文解析システムをPrologで記述しなくてはならないのかと考えていた頃，松本君が，DCG形式で書いた文法を，Left Cornerでボトムアップに解析を行なうPrologプログラムに変換する方法を考えはじめていた．これは彼らしい奇抜なアイディアであった．Prologのもつトップダウンの計算機構によるボトムアップ計算のシミュレートという側面を持っていたからである．松本君は実際にトランスレータを作成してそれが可能であることを実証した．このシステムはBUPとよび，ケンブリッジ大学の自然言語処理システムでも使われた[Alshawi 92, Matsumoto 83]．

BUPを利用していると，Prologがバックトラックを行なうたびに過去の計算履歴を捨て去り再計算を繰り返すので効率が悪い．これを防ぐために一度計算した結果を記憶して再利用しようということになった．これはほんの数行のプログラムを付加するだけで可能なことがわかり，それで5倍ほど高速化した．

BUPの論文は，論理と自然言語理解に関する国際会議で発表したが，横型探索のアルゴリズムについての質問があったことを海外留学中の松本君に手紙で知らせたところ，留学先でさっそくDCGを，ボトムアップ横型探索のチャート法で構文解析を行なうPrologプログラムに変換する方法を考え出した．これがSAXであり，彼は第五世代コンピュータ計画に参加し，その後PAXとよぶ並列計算版の構文解析システムに発展させている．

4·3 規則ベースからコーパスベースへ

自然言語処理システムが扱う世界が，マニュアル文，ニュース文，科学技術文献などに拡大するにつれて，これらに含まれる言語現象すべてをカバーする文法規則を人手で書き尽くすことが次第に困難になってきた．これが規則ベースの方法の第1の問題である．第2は，規則の適用に曖昧性が生じるとき，どのような基準で(計算法で)もっともらしい解析結果を選択するかという問題である．拡張Lingolにも解析結果にスコアを付ける機能はあったが，スコアの計算方法が確立していないことが問題であった．

1980年代半ばになり，これまでなかなか実用のレベルに至らなかった音声認識が，大量の例文(コーパス)に含まれる単語連鎖の統計情報を利用して認識率を上げることに成功する．これはn-gramとよぶ言語モデルである[Manning 99]．モデルが単純であるだけに，話し言葉にしばしば現れる冗長語(「えーと」,「あのー」など)や，言い直しなどの言語現象にも対応可能で頑健であることの他に，解析結果に付与する確率的なスコアを利用して，もっとも妥当な解析結果を選び出し曖昧性を解消する方法が有効であることがわかった．

そこで音声認識と同様に自然言語処理においても，コーパスベースの手法を取り入れて，確率によるスコア付け

*1 1980年にこれは確定節文法(DCG)というタイトルでAIジャーナルの論文として発表される[Pereira 80]．

を行なうべきであると考えられるようになった．1980年代後半のことである．もっとも単純なn-gram言語モデルは，日本語の場合，形態素解析に導入され成果をあげる．形態素解析より上位の処理の場合には，n-gram言語モデルだけでは到底不十分であり，確率CFG言語モデルが使われた．これは規則ベースと統計ベースを統合した言語モデルであり，英語の構文解析でその有効性をはじめて実証したのは，IBMの藤崎哲之助氏であったと思う．

4·4 大学での自然言語処理の研究

1983年に電子技術総合研究所から東京工業大学に移ったが，当時気になっていた問題を学生に話すと，数カ月後に解決してくれるという喜びを味わった．主なものをあげると，木下聡君 (現東芝) は英語の関係代名詞節の (先行詞として) 左に外置された部分を自動的に検出する機能をBUPに組み込む方法を考案した．徳永健伸君 (現東京工業大学助教授) は，これまで東京工業大学で行なってきたBUPに対する拡張機能を集大成したLangLabシステムを開発してパブリックドメイン上で公開した．奥村学君 (現東京工業大学助教授) は，文を読み進むにつれて漸進的に単語の語義の曖昧性を解消するモデルを開発した．高倉 伸君 (現IBM) は，DCGによる大規模な英語文法を開発した．その一部は冨田 勝氏 (現慶應義塾大学教授) の執筆したGLRの本の付録として掲載されている [Tomita 86]．大学に移った時期が，コンピュータの値段も下がり，かなりの計算機パワーを大学の研究室でも持てる時代にさしかかっていたのは幸いであった．

4·5 GLR

当時の助手の田村直良君 (現横浜国立大学教授) は，LR法 [Aho 72] を拡張してCFGが扱えるようにする研究を行なっていた．彼は決定的な解析が不可能になった時点で縦型探索を行なう戦略を採用していた．ちょうどその頃CMUの学生であった冨田氏が田中研に1カ月ほど滞在した．冨田氏はその後，それを横型探索で解決するGLR(Generalized LR；冨田法) を開発している [Tomita 86]．これは経験的にチャート法を凌ぐ構文解析速度を持つことが知られている．

LR法は先読み語から得られる情報を利用することが特徴であり，人工知能の研究者が考案したWait&Seeパーザの考え方 [Marcus 80] を先取りしていた．横型探索であるため，並列計算にも適したアルゴリズムであること，CFGに含まれる文法制約をあらかじめプリコンパイルした表 (LR表) の形式にして構文解析で利用すること，そして先読み語の情報を利用することなどが気にいった．

4·6 言語理解の研究と知識ベース

1980年代の自然言語処理の要素技術の研究成果は多い．全般に言語理解の研究は多くない．言語理解には意味の問題を避けることができないが，1970年代の積木の世界と異なり，扱う言語現象が多様で複雑になってきた．語彙数も増大する．そのためコンピュータで扱うことができる大規模な辞書の構築が問題になってきた．

ちょうどこの頃，20万を越える語彙数をもつ電子化辞書開発計画 (EDR計画) が1980年代後半に通産省の主導で世界に先駆けて始まる．機械翻訳システムの実用化の時期とも重なり，それがこの計画に拍車をかけた．機械翻訳システムでは単語の語義の曖昧性解消 (word-sense disambiguation) が必要になる．そのためには少なくとも概念間の上位・下位関係の知識が必要になる．電子化辞書計画では上位・下位関係の知識ベースも開発されている．EDR計画に刺激されて，米国では百科辞典的な知識ベース構築計画CYCが始まっている．

5. 1990年代——コーパスの利用

1990年代はコーパスを利用した研究成果が激増した時代である．我々の研究室でも，それにコミットした研究を行なったが規則ベースの良さを捨て去る気にはならなかった．

5·1 MSLR—形態素解析と構文解析の統合化

日本語の形態素解析システムでは隣接した文法カテゴリー間の接続制約を表にした接続表を用いる．LR表にはすでにCFGの制約が組み込まれている．接続制約をLR表に組み込むことができれば，GLR法による解析により，形態素解析用の制約と構文解析用の制約とを同時に使いながら，二つの解析を完全に統合化して行なうことができる．わかち書きが終わると同時に構文解析結果が得られるわけである．これは田中研究室助手の白井清昭君と学生の植木裕之君，橋本泰一君らにより実装されMSLRとして公開されている*2．

MSLRの考え方で面白いのは，LR表に接続制約を組み込むことは，LR表の中の解析動作のあるものを取り除くことであり，それにより，LR表のサイズはむしろ小さくなり単純化するということである．LR表中の解析動作を削除することで，GLRパーザの動作を制御することが可能になる．どの解析動作を削除すべきかは接続表が決めてくれる．これと類似の研究はないかと探していたところ，Ahoらが解析動作の削除により決定的な解析を行なう方法をすでに提案していた [Aho 75]．一般にLR表中の解析動作の削除により，構文解析過程を制御する技法を我々は「LR表工学」と名付けてみた．

5·2 確率GLR言語モデル

1989年に台湾の計算言語学会大会に招待された折りに，当時台湾の清華大学のSu氏の研究に興味を憶えた．

*2 http://tanaka-www.cs.titech.ac.jp/pub/mslr/index.html 参照．

確率 CFG(PCFG) では，文脈に依存した確率のスコアを，各構文解析結果 (構文木) に与えることができない．彼はこの問題を解決する一つの考え方を提示していたのであるが，それは GLR 法の枠組にフィットしているという印象をもった．

GLR 法では，先読み語と GLR パーザの現在の状態とから LR 表を参照して，次に行なう解析動作を決定する．先読み語とパーザの状態は，それぞれ解析動作を行なうときの右文脈と左文脈におよそ対応しているからである．したがって，LR 表中の各解析動作に確率を与えることができれば，それらを用いて文脈依存の確率 (スコア) を各構文木に与え各構文木の優先順位を決めることができる．

1993 年に入り，徳永君が，ケンブリッジ大学の Briscoe と Carroll の論文を手に研究室に入ってきた [Briscoe 93]．ざっと目を通すと，我々の考えと同じに見える．Su 氏の論文も参照しているし実験結果も示している．先を越されてしまったと思いつつ詳しく読んでみると，どうも LR 表中の解析動作に与える確率の計算法に理論的な根拠がないように思えた．最終的に乾 健太郎君 (現九州工業大学助教授) と一緒に，Briscoe らの計算法の問題点を明らかにするとともに，理論的に妥当な確率の計算法を求めることができた [Inui 97]．これが確率 GLR (PGLR) 言語モデルである．

5·3 言 語 資 源

PGLR 言語モデルを使うためには，LR 表中の各動作に確率を振らなければならない．そのためには (正しい) 構文木付きの訓練用例文 (コーパス) が大量に要る．ところが 1990 年のはじめに，この種のコーパスはわが国には皆無であった．コーパスの収集を急ぐ必要があると考えた．

米国ではコーパスの開発と流通を目指したコンソーシアム LDC が 1992 年に活動を始めていた．ヨーロッパでは辞書編纂に利用する目的で早くから大量の例文が電子化されていた．1995 年には言語資源の流通を目的とした ELRA が設立される．欧米に比べてわが国の立ち後れは歴然としている．大規模コーパスの構築とそれへのタグ付け，構文構造付けの作業は自動化が困難で，大量の人的資源と時間を要する．しかしタグ付き，構造付きの大規模コーパス構築の意義は一部の人を除き，今でもわが国ではなかなか理解されない．

資金の裏付けが得られるのを待ってからでは遅いと考えて，言語資源共有機構 (GSK) という組織を昨年 5 月に立ち上げることにした．言語資源とは，タグや構造付きコーパス，電子化辞書，シソーラス，自然言語処理用ツールなどの総称である．GSK の活動に興味のある方は次のホームページアドレスを参照して欲しい．そこからわが国のさまざまな言語資源とその入手法を知ることができるはずである．

http://tanaka-www.cs.titech.ac.jp/gsk/

5·4 コーパスと言語学

言語資源について言語処理の立場からその必要性を述べたが，実はそれは言語学にも役立つ．コーパスはさまざまな言語現象の宝庫であり，コーパスを分析することで新しい言語現象が発見されることが十分に考えられるからである．コーパスが電子化されているので，コンピュータの助けを借りて言語現象を容易に取り出すことができる．それによりともすれば理論が先行し，それから理論の検証のためにコーパスを調べるというトップダウン的なアプローチだけではなく，言語データの分析から新しい言語現象を発見し言語理論を組み上げるボトムアップな研究も行ないやすくなるだろう．コーパスは辞書の構築にも役立つことは言を待たない．

5·5 なぜコーパスベースが流行したのか

1990 年代の自然言語処理技術を特徴付けるとしたら，それはコーパスベースの時代であると言える．その理由を以下にまとめてみる．

(1) 比較的規模の大きな電子化されたコーパスが入手可能になった．

(2) 音声認識の分野に適用して，コーパスベースの言語モデル (n-gram モデル) が成果をあげた．

(3) 解析結果に確率的な優先順位 (スコア) を振ることが可能になり，それを曖昧性解消の尺度の一つとして利用することができる．

(4) コーパスに含まれる表層の単語連鎖をベースにした確率を用いることにより以下の好ましい結果を得ることができる．

- a 単語の意味に踏み込むことなく解析結果に確率に基づく優先順位を付与することが可能である．
- b 比較的容易に日本語の形態素解析の精度が向上する．
- c 依存関係の解析精度が向上する．
- d これらの精度は少ない労力で規則ベースの精度に近付く．
- e 言語モデルが単純であるので，規則に合わない言語現象に対する頑健性をもち，解析可能な言語現象の数が規則ベースのそれを越えている．

(5) 表層の単語レベル (語彙レベル) より上位の構文的な統計情報を利用するコーパスベースの方法として，PCFG，PGLR などさまざまな確率的言語モデルが考案され，有効であることが明らかになった．CFG を用いるものが多く，これらはコーパスベースと規則ベースを統合した言語モデルである．

音声認識の研究と同様に，単語の連鎖を扱うため学習用のパラメータ数が膨大となり，それに伴いコーパスの量がどうしても不足する．PCFG，PGLR 言語モデルの場合には，人手の介入がどうしても必要な構文木付きのコーパスが必要になり，その量的不足は明らかである．この問題を克服する技術 (スムージング技術) や，タグ付き，

構造付きの大規模コーパス構築技術の研究も盛んに行なわれた.

6. 2000年代——これから何をすべきか

前章で述べたように1990年代は欧米だけでなく日本でも,コーパスベースの自然言語処理技術の研究が活発に行なわれた.しかし最近コーパスベースの技術にも飽和現象が見られ,ドラスティックな解析精度の向上がみられなくなってきた.1990年代には言語理解の本格的な研究が見当たらなかったのも特徴といえるかも知れない.

6·1 コーパスベースの限界

ここでコーパスベースの方法の限界をみておきたい.コーパスベースの単語連鎖に基づく単純な言語モデルにより,音声認識の精度は1980年代後半から1990年前半にかけて確かに著しく向上した.市販の安価な音声認識システムでもマイクに向かって喋ると,かなりの精度の音声認識結果が得られる.認識結果が次々に出力されるので,音声認識システムが言語を理解していると錯覚するほどである.実は音声認識システムは単語の列を出力するだけで喋った言葉の意味を理解していない.

このことは語彙レベル(単語連鎖)の統計情報を用いて形態素解析を行なう場合も同様である.言語理解システムを構築するためには,音声認識結果や形態素解析結果(わかち書きの結果)に対して,再び構文解析や意味解析などの処理を施す必要がある.この時おそらく文法を使うことになるので,規則ベースの処理が登場する.規則のもつ制約とコーパスのもつ統計的な制約とを共に用いた言語モデルがPGLR言語モデルである.

前節では単語連鎖の統計情報を利用するコーパスベースの方法の利点を幾つかあげた.これはしばしば語彙化という用語で特徴付けられる.1960年代の依存文法では単語と単語の間の依存関係(これも広義の意味での単語連鎖関係)を扱う.係り受け関係にある単語同士は必ずしも隣接している必要はないが,表層的な単語連鎖の統計情報を利用して依存関係を決めることができる.大量のコーパスが蓄積され,コーパスベースの方法での語彙化の流れにのり,1990年代に依存文法が再び息を吹き返す.

現在,コーパスベースの方法で得られる係り受け関係同定の精度は80%程度である.したがって,文中のすべての係り受け関係が正しいとされる確率はまだ低く,到底十分であるとはいえない.言語理解では少なくとも係り受け関係,さらには深層格関係(後述)の抽出などが必要になる.この時,文中の一箇所の係り受け関係を誤ると,機械翻訳などでは惨めな翻訳結果を出力することになるだろう.

6·2 依存関係,依存構造を越えて

依存関係,依存構造は言語理解に役立つ情報を含んではいるが,それだけで十分であるとはいえない.以下では依存関係,依存構造以上に深い意味構造の抽出が言語理解には必要になることを説明する.

依存構造はラベルなしの木構造(括弧付き構造)で表すことができる[Manning 99].「戸を叩く音がする」と「戸を叩く人がいる」の依存構造を括弧付き構造で表現すると以下のようになる.

(1) (((戸を 叩く) 音が) する)
(2) (((戸を 叩く) 人が) いる)

両者とも連体修飾節を含んでおり依存構造としては同型である.ここで読者は,各々の文を英語に直してみられるとよい.「戸を叩く人がいる」の場合には「人」を先行詞にした関係代名詞節を含む英語の文で表現することができる.これに対して「戸を叩く音がする」の場合には関係代名詞節を含む文で表現することができない.

両者の違いを考えてみると,「人」は「叩く」の深層格(動作主格)となるが,「音」は「叩く」の深層格にならないことがわかる.このような深層格関係の違いを考慮しないと正しい翻訳ができない.言語理解では,依存関係,依存構造を越えた深い意味関係,意味構造をとらえることが必要になる.文の各構成要素の文法機能を明らかにすることは当然のこととして,深層格関係程度は少なくとも抽出しておく必要があるのである.

依存構造は「係り-受け」という二項関係をベースに構成されるため,次の問題も考慮する必要がある.「開く」という受けの語に対して「鍵で 開く」と「花が 開く」が成り立つと,「花が 鍵で 開く」を正常な文として受け付けてしまう.文の意味的異常さが検出できないのである.「花が 鍵で 開く」は三項関係を考慮して初めて意味的異常さがわかる文である.二項関係を越えた立場から言語理解の問題を考えなければならない.一般にこれは結合価の問題であるが,言語学的にも,認知科学的にも今後取り組むべき興味深い研究課題である.

6·3 照応,省略現象など

言語には,照応,省略など一文を越えた言語現象が含まれる.照応の場合には照応先の先行詞の同定が,また省略の場合には省略語の補完が必要になる.われわれはさまざまな知識を駆使した推論を行ない,これらの言語現象を理解していると推察される.これらの問題の解決には,語義の問題も関係してくるので厄介である.文のつながり具合や首尾一貫性も考慮する必要がある.コーパスベースの方法でこの問題を解決しようとする試みもあるが,満足のいく解は得られていない.困難ではあってもこれもまた今後の重要な研究課題である.

6·4 音声と自然言語の研究者の共同研究

音声認識システムは，コーパスベースの技法を採用し，発話条件についての制約を守りさえすれば実用化されているといってよい．コーパスベースの限界の節で述べたように，現在の音声認識システムは，単語連鎖の統計情報を利用する単純な言語モデルを採用しているため，話した言葉の意味を理解していない．話し言葉を理解する音声理解システムを実現しようとするなら，話し言葉の文の構文構造，意味構造などを明らかにする必要がある．それには自然言語理解システムの研究者側の研究成果が利用できるはずである．

ところが音声と自然言語の研究者はこれまで個別に研究を進めてきており，両者を包含する立場から共同して音声理解の研究を行なうケースが少なかったように思われる．両者ともそれぞれに困難な研究課題を抱えており，他を振り返る余裕がなく自己の問題解決に全力を傾注していたからだろう．あえて言えば，自然言語から音声研究へアプローチする研究者が少なかった．これからは，言語学的にも十分な検討がなされていない話し言葉を巡り，両者が力を合わせて対話理解，音声理解の研究を進める必要がある．

7. おわりに—言語理解と行動制御

前章の最後に述べた研究と関連して，個人的な興味から一つの研究課題を述べてみたい．これまでの言語理解の研究では，機械翻訳，情報検索，文書分類・要約など，主として自然言語で書かれた文書を対象としていたのに対して，話し言葉と行動という視点から言語理解の研究を考えてみたいのである．

7·1 研 究 の 背 景

1970年代初頭のMITにおけるSHRDLUとよぶ言語理解システムをすでに紹介した．これはコンピュータの中にシュミレートされたロボットが，端末からの指令に応じて積木の世界で動作する．このように，言語によってロボットの行動を制御する研究の重要性は指摘されていたにもかかわらず，以後ほとんど進展がみられない．これは，機械的なロボットにしろコンピュータ内部の仮想空間のロボットにしろ，当時はそれらの動作機能が限られていたこと，指令文を端末からいちいち入力する手間がかかること，自然言語処理技術が未熟であったことなどが原因であった．

ところが最近コンピュータグラフィクス(CG)技術の進展により，機械的な制約を受けず，行動機能が豊富な3次元のソフトウェア・ロボットをコンピュータ内部に作りだし，それを自在に動かすことができるようになってきた．言語による指令にしたがって動作するロボットは機械的なロボットに限られなくなってきた．さらに，端末から文を入力する代りにマウス操作で代用して問題の一部を解決してきたが，音声認識システムの実用化が進み，音声で指令を与えることが可能になってきた．自然言語処理技術も1970年代と比較して格段の進歩がみられる．

そこで，音声認識技術，自然言語処理技術，CG技術，人工知能技術を集大成し，言語によりロボットの行動を制御するシステムを構築したいという夢が膨らんだ．このロボットは，機械的なロボットであってもCGによる仮想空間内の3次元ロボットであっても良い．これらを総称してエージェントと呼ぶことにすれば，エージェントは行動するだけでなく，音声，表情，身振り手振り付きでこちらに応答することも考える．このエージェントが初め「部屋の外に出る」という行動概念を知らなければ，それを丁寧に言葉で教えることにより，新しい行動概念を学習することもさせたい．

7·2 プロトタイプシステムの動作例

ここで簡単なプロトタイプシステムの動作例を示して，研究の意義の一端を説明してみたい．プロトタイプシステムでは，3人の3次元のエージェントが仮想空間内で演奏している映像をカメラが映し出しているものとしよう．この映像を映し出しているカメラもまたエージェントの一つである．カメラエージェントC1に対して音声による指令を出して，カメラの位置を変えるという動作を行なわせると，カメラが映し出す映像が連続的に変化する．

動作例：

指令1：ドラマーの後ろに回り込んでください．

(C1が映し出す映像がドラマーを中心に連続的に変化し，ドラマーの後ろ姿を映し出す)

指令2：もうちょっと後ろに．

(ドラマーの姿が少し小さくなる)

指令3：もうちょっと．

(ドラマーの姿がさらに小さくなる)

指令4：ちょっと行き過ぎ．

(ドラマーの姿が若干大きくなる)

.....

このカメラエージェントC1の動作例から，言語理解と動作に関するさまざまな問題の一端を示すことができる．

指令1に関しては，「回り込む」という動作にはさまざまな「回り込み方」がある．そのいずれか一つを決めて，カメラの位置を変化させなければ映像が作れない．指令2では，カメラがどの程度後ろに引くかを決めなければならない．これらは，言語学者が漠然性(vagueness)とよんでいる問題である．さらに，指令2は，指令1のカメラの最終位置を基準として，ドラマーから遠ざかる方向に動かなければならない．これは，脈絡に依存した言語理解が必要になることを意味している．さらに，指令2では前文より省略語を推論し，「ドラマーのもう少し後ろに下がる」という意味であると解釈しなければならな

い．指令3でも指令2と同様な言語理解が必要になる．指令4では，文字通りの意味解釈とは異なる行動/動作をカメラに要求している．「カメラに，ドラマーの方にもう少し近付く」ことを要求しているからである．これは言語行為論として哲学者らが研究しているが，言語理解と行動の研究では，こうした問題がただちに顕在化する．「右に」，「左に」などの指令を出せば，状況に依存した方向の理解が必要になる．音楽監督になった積もりでドラマーらに演奏方法についての指示を出し，ドラマーの演奏方法を変えさせることなども考えられる*3．

このようなシステムを構築することができれば，エンターテインメントの世界ではアニメーションの制作支援，医療福祉の世界では介護ロボットの対話による制御や，これまで熟練した専門家が行なっていた機器の対話による操作も可能になる．また，分子構造，都市構造，宇宙の構造などのミクロからマクロなレベルに至るさまざまな立体モデルを映し出すカメラ動作の指示を言語によって行なうことにより，さまざまな角度からの立体構造を容易に観察することも可能になる．

以上は言語理解についての一つの夢物語である．困難ではあっても挑戦する価値のある研究課題と思われるがどうだろうか．そのためには，音声と自然言語の研究者の他に，実は，言語学者，心理学者，哲学者，認知科学との共同研究も必要になる．

◇参考文献◇

[Aho 72] Aho, A. V. and Ullman, J. D.: *The Theory of Parsing, vol.1*, Englewood Cliffd, N.J.: Prentice-Hall (1972).

[Aho 75] Aho, A., Johnson, S., and Ullman, J.: Deterministic Parsing of Ambiguous Grammars, *Comm. of ACM*, Vol. 18, No. 8, pp. 441–452 (1975).

[Alshawi 92] Alshawi, H., ed.: *Core Language Engine*, The MIT Press (1992).

[Briscoe 93] Briscoe, T. and Carroll, J.: Generalized Probabilistic LR Parsing of Natural Language (Corpora) with Unification-Based Grammars, *Computational Liguistics*, Vol. 19, No. 1, pp. 25–59 (1993).

[Inui 97] Inui, K., Sornlertlamvanich, V., Tanaka, H. and Tokunaga, T.: A New Formalization of Probabilistic GLR Parsing, in *Proc. of the Fifth International Workshop on Parsing Technologies(IWPT-97)*, pp. 123–134 (1997).

[Manning 99] Manning, C. D. and Schutze, H.: *Foundations of Statistical Natural Language Processing*, The MIT Press (1999).

[Marcus 80] Marcus, M. P.: *A Theory of Syntactic Recognition for Natural Language*, The MIT Press (1980).

[Matsumoto 83] Matsumoto, Y. et al.: BUP: A Bottom-Up Parser Embedded in Prolog, *New Generation Computing*, Vol. 1, No. 2, pp. 145–158 (1983).

[Minsky 68] Minsky, M, ed.: *Semantic Information Processing*, The MIT Press (1968).

[Pereira 80] Pereira, F. C. and Warren, D. H.: Definite Clause Grammar for Language Analysis–A Survey of the Formalism and Comparison with Augmented Transition Networks, *Artificial Intelligence*, Vol. 13, No. 3, pp. 231–278 (1980).

[Pratt 75] Pratt, V.: LINGOL—A Progress Report, in *Proc. of the 4th International Conference on Artificial Intelligence*, pp. 372–381 (1975).

[Tomita 86] Tomita, M.: *Efficient Parsing for Natural Languages*, Kluwer Dordrecht (1986).

[田中 77] 田中，佐藤，元吉：自然言語処理のためのプログラミングシステム—拡張 Lingol について，電子通信学会論文誌，Vol. J60-D, No. 12, pp. 1061–1068 (1977).

[Winograd 72] Winograd, T.: *Understanding Natural Language*, Academic Press, New York (1972).

*3 実は，これとはちょっと異なるシナリオで動作するプロトタイプシステムが田中研究室のパソコン上で動いている．

2000年6月9日 受理

著者紹介

田中 穂積(正会員)

1964年東京工業大学理工学部制御工学科卒業．1966年同大学院修士課程修了．同年電気試験所（現，電子技術総合研究所）入所，1983年東京工業大学工学部情報工学科助教授．1986年同大学教授，1994年同大学院情報理工学研究科教授となり現在に至る．工学博士．人工知能，自然言語処理の研究に従事．情報処理学会，電子情報通信学会，日本認知科学会，計量国語学会，AAAI各会員．