統計的統語解析

松本　裕治

日本語や英語などの自然言語の文の統語構造を解析することは，言語解析の最も重要な処理である．従来は，文法規則を列挙し一般的な統語解析アルゴリズムによって文の構造を得るという手法が取られてきたが，多数の解析結果が得られることが多く，曖昧性の解消が重要な問題であった．近年，大規模な解析済み例文から統計的機械学習を用いることにより，従来の人手による規則の記述に基づく統語解析を遥かに凌ぐ手法が提案されている．本稿では，最近の統計的統語解析の2つのアプローチを概観する．

キーワード：自然言語処理，文法，統計学習，句構造解析，依存構造解析

1．はじめに

日本語や英語などの自然言語の文は，単語の列からなるが，実際には平坦な構造ではなく，いくつかの単語が組み合わさってまとまりをもつ単位（句と呼ばれる）になり，さらにそれらがまとまってより大きな単位となるといった階層的な構造をもつ．統語解析は，文のそのような構造を明らかにするためのもので，自然言語処理の中で最も重要な解析と考えられている．

本稿では，統語解析の簡単な歴史を振り返り，次に，過去十数年の間に飛躍的に進歩を遂げた統計的統語解析技術について概観する．

2．統語解析とは

言語文を理解するためには，その統語的な構造を明らかにすることが必要である．英語や日本語の文の統語構造は，句構造解析，あるいは，係り受け構造解析によって行われることが多い．前者の句構造解析（phrase structure analysis）は，例えば，英語の文が名詞句と動詞句からできており，さらに名詞句は冠詞と名詞からできているなど，句の構造を階層的に記述することによって行われる．これらは，次のような規則で書くことができる．

- 文→名詞句 動詞句
- 名詞句→冠詞（形容詞）名詞
- 動詞句→動詞 名詞句

統語解析の主流は，このような句構造規則という文法規則を記述することによって言語の可能な構造を規定し，それに基づいて任意の文の構造を解析する手法の研究であった．そこでの大きな問題は，曖昧性（一つの文が何通りにも解析されてしまう）の問題であった．また，一方で，どのように詳細な文法規則を記述しても，必ず例外的な構造をもつ文が発見され，完全な文法規則集合を得るということが極めて困難だという問題だった．

一方，近年では，依存構造解析（dependency structure analysis），あるいは，係り受け解析といって，文中の単語の直接の修飾・被修飾関係（あるいは，係り受け関係）を解析する手法の研究が盛んになってきた．

句構造解析，依存構造解析のいずれの解析においても，解析結果としての文は木構造の形で表現される．図1に，“Estimated volume was a light 2.4 million ounces.”という文の句構造解析木（左側）と依存構造解析木（右側）の例を挙げる．句構造木では，“S→NP VP”, “NP→VBN NN”, “VP→VBD NP”[1]のような句構造規則が使われていることがわかる．一方，依存構造木では，このような句を定義する文法規則は明示的に存在せず，単語の間の係り受け関係だけが解析される．

日本語では，単語の間の依存関係ではなく，まず文を構成する単語列が文節[2]という単位にまとめあげられ，文節間の係り受け関係を解析するのが普通である．

まつもと　ゆうじ
奈良先端科学技術大学院大学
〒 630-0192 生駒市高山町 8916-5

[1] S, NP, VP, VBN, VBD, NN はそれぞれ，文，名詞句，動詞句，動詞の過去分詞形，動詞の過去形，名詞を表す．図1の文は Penn TreeBank[11]から抜粋した．

[2] 日本語の文節とは，自立語（動詞，名詞など）の後に付属語（助詞，助動詞など）が続くまとまりのこと．

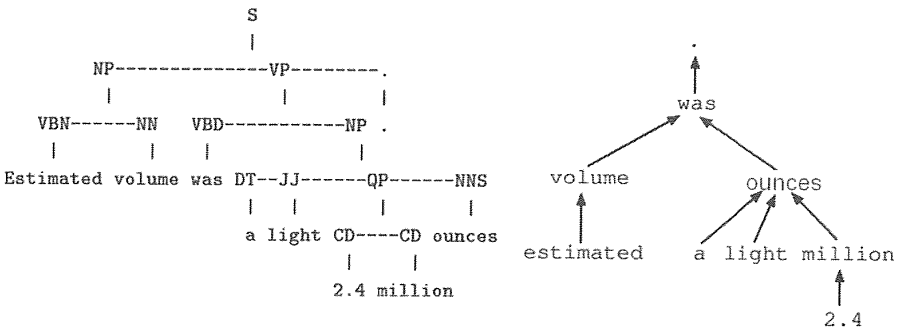

図1　句構造木と依存構造木

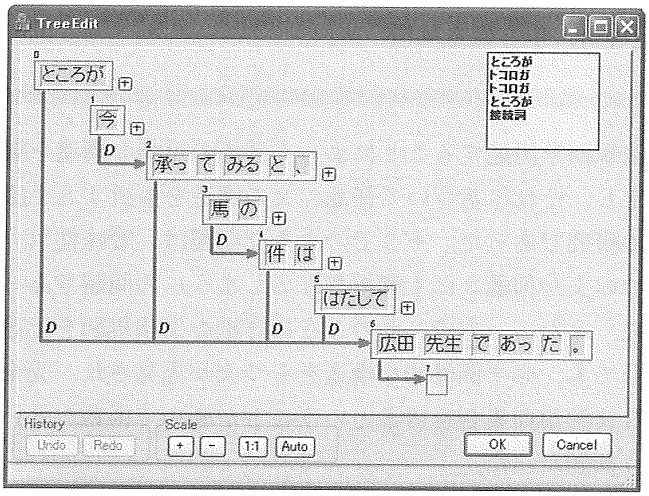

図2　係り受け解析木の例

日本語の係り受け解析木の例を図2に示す．図1の右側の依存構造木と図2では，単語や文節の配置が異なるが，基本的に同様の構造を表している．

言語の統語構造は，主として，句構造規則を詳細に記述し，句構造文法を対象とする効率よいアルゴリズムによって解析するという考え方が取られることが多かった．英語や日本語などの現実的な句構造文法は，数百また場合によっては数千もの句構造規則を含むものになってしまうことがあり，文法規則の管理だけでなく，それから生じる解析時の曖昧性の解消は重要な問題であった．

依存構造解析では，文節や語がどのような文節(語)に係り得るかという規則を書くことによって行われる．最も単純な日本語の係り受け規則は，文節が連体修飾か連用修飾かを判定し，前者は体言（名詞）に後者は用言（動詞，形容詞など）に係るとするものであり，ここでも複数の係り先候補が存在する場合に生じる曖昧性の問題は深刻な問題であった．

3.　統計的統語解析

90年代に入り，電子化された大量の文書が入手可能になるにつれ，大規模テキストデータ（コーパス）を利用した統計的言語処理が盛んになってきた．そのきっかけになったのは，IBMのグループが手がけた統計的機械翻訳に関する研究で，対訳例文集が大量にあれば，源言語の単語列が最も高い確率で対応付く目的言語の単語列を生成する確率モデルを作ることにより，英仏間の翻訳を実践してみせたものであった[1]．同じ時期に，単語の品詞同定や統語解析でも統計的手法が取り入れられるようになり，現在では，統計的手法は規則に基づく言語解析を完全に凌駕してしまうようになった．90年代前半から現在に至るまでの統計的統語解析法について以下に説明する．

3.1　確率文脈自由文法

90年代以前から文脈自由文法[3]の個々の文法規則に出現確率を割り当て，文全体の確率が最大になる統語構造を選択することで曖昧性を解消を行うことが考えられていた．確率文脈自由文法では，左辺に共通の句の名前をもつ文法規則，例えば，動詞句（VP）を左辺にもつ文法規則（VP → VBD NP，VP → BE NPなど）の確率値の合計が1になるというように，個々の文法規則が左辺の句を条件とする条件付確率をもつとして定義され，文の解析結果として得られる句構造木の確率は，その句構造木中の全規則の確率値の積として定義された．句構造文法を動的計画法によって効率よく解析する統語解析アルゴリズムを用いれば，文確率値が最大の句構造木を効率よく求めることができる．

[3] 文脈自由文法と句構造文法は同義．ここでは慣例により「確率文脈自由文法」という用語を使うが，以後は句構造文法に統一する．

確率文脈自由文法は，さらに，Inside-Outsideアルゴリズムという EM（Expectation-Maximization）アルゴリズムの一種によって，未解析あるいは部分的に解析された言語データから文法規則の確率値を推定することができるため，言語データの大規模化とともに広く使われるようになった．

しかし，以下の1，2に示す例文は，いずれも「名詞—助詞—動詞—名詞—助詞—動詞」という品詞列となっており，品詞や句だけを対象に文法規則を記述する句構造文法では，これらの構造の違い（例文1では「双眼鏡で」という文節が「監視した」に係るのに対し，例文2では「海で」が「泳ぐ」に係る）を区別することができないという欠点をもっていた．

1．双眼鏡で泳ぐ子供を監視した

2．海で泳ぐ子供を監視した

確率文脈自由文法の確率値の推定方法や初期の統計的言語処理の手法については，文献[2][9]が詳しい．

3.2 統計的句構造解析

上記のような語の意味の違いを考慮しなければ正しい結果が得られない文を解析するためには，単語や単語がもつ意味情報を利用する必要がある．Magerman[8]が，句や品詞だけでなく単語をも属性として用いる決定木学習を利用した統語解析を提案し，上記のような曖昧性に対応できる手法を示して以来，統計的機械学習を利用した句構造解析が盛んに研究されるようになった．Collins[4]は，句構造を作り上げる際に，各句の中心になる単語（主辞（head）と呼ばれる）の間の共起確率をより積極的に用いる統計的句構造解析法を提案した．例えば，上例文において，「双眼鏡で」と「泳ぐ」が句としてまとめ上げられる確率は非常に低く，「双眼鏡で」と「監視する」を主辞として含む句とまとめ上げられる確率に遥かに及ばない．逆に，「海で」は「泳ぐ」と強く共起し，これらが一つの句としてまとまるのが自然な解析である．句構造規則の確率を求める際に，その構成要素となっている句の主辞となっている単語の共起確率を考慮することにより，上記のような例文の句構造を区別できるようになった．Ratnaparkhi[13]は，単語を属性とし，最大エントロピー法を用いて，大規模な属性集合についても頑健な確率推定を行うことのできる句構造解析法を提案した．Charniak[3]は，確率文脈自由文法で得られる上位数十の解析結果を利用し，個々の句構造規則の文中での確率値を見積もるのにその句構造規則のさらに上位の句の情報などを属性として用い，最大エントロピー法を利用した確率値推定を行うことにより，より高精度の句構造解析を実現した．これらの一連の研究は，Penn Treebank[11]の一部を学習データとし，他の共通の一部をテストデータとすることにより，同じ土俵上での比較が行われている．Magermanの最初のパーザの精度が約85%だったのに比べ，最も性能のよいCharniakパーザでは，約90%の性能を示している．Charniakパーザはフリーソフトとして公開されており，以下から入手可能である．

http://www.cs.brown.edu/people/ec/

Collins[5]は，その後，自らの統計的パーザによって上位数十の句構造解析結果を出力し，その中から正解の句構造木を最も上位にランク付けするという機械学習を提案した．Tree Kernelを用いることにより句構造木内のすべての部分句構造木を属性としてランキングの学習を行うことができる．

これらの手法は，文の統語解析がある特定の句構造文法（句構造規則の集合）に基づいて行われると仮定し，Penn Treebankの約4万文に付与された句構造を前提として学習を行っている．同じ手法を他言語に適用しようとすると，その言語で可能な言語表現をカバーする句構造規則を網羅する必要があるが，これはもちろん簡単なことではなく，多大な労力を必要とする．

3.3 統計的依存構造解析

前節の最後に述べたように，句構造文法は，言語表現をカバーする文法規則を記述するのが容易ではない．また，現実の文を人手によって解析するにしても作業者間の揺れが生じる場合があり，解析済みデータの品質を保証するのが難しい．特に，作業者は高度な文法に関する知識が必要である．近年は，医学生物学分野，法律，特許など専門性の高い分野のテキストの解析が現実問題として要求されるようになっているが，そのような分野に特有な用語や言語表現に精通し，かつ，句構造文法の知識をもった作業者を得ることはほとんど不可能であるため，英語以外の言語やPenn Treebankが扱う経済以外の分野で充分な性能を発揮する統語解析システムを構築するのが困難であった．

日本語では，文節単位の係り受け解析が文構造の基礎として学校で教えられており，句構造解析よりもむしろ依存構造解析の方が自然である．工藤[7]は，隣り合う文節間に係り受け関係があるかどうかを学習し，係り受け関係があると判断された文節を段階的にまとめあげていくことにより，文節係り受け解析を実現し

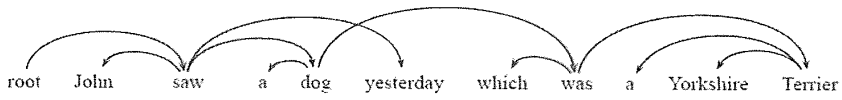

図3　依存関係の交差をもつ依存構造木

た．2つの文節間には係り受け関係があるかないかのいずれかであり，2クラス分類問題として扱うことができるので，Support Vector Machines (SVM) を学習器として用いている．本システムは「南瓜」という名前で，以下から入手可能である．

http://chasen.org/taku/software/cabocha/

南瓜は，係り受け解析が施された日本語コーパス(京都大学コーパス[4]の約3万数千文を用いて学習し，文節間の係り受け関係を約91%の精度で解析することができる．図2に示した例は，このシステムにより解析されたものである．

他の言語においても単語の間の依存関係（係り受け関係）の解析を対象とした依存文法という考え方があり，文節ではなく，単語間の依存関係の解析を行うことにより，図1の右側に示したような依存構造木を得ることができる．Yamada[14]とNivre[12]は，句構造文法の後戻り型のボトムアップ統語解析アルゴリズムであるShift-Reduce法と同様の手続き（ただし，文法規則は2つの単語の間に右から左への依存関係があるか，左から右への依存関係があるかのいずれか）を用いた統計的依存構造解析法を提案した．アルゴリズムの各ステップでの処理動作の決定を機械学習(Yamada法ではSVM, Nivre法ではk-近傍法）によって行うことにより，後戻りや複数の途中結果を保存することなく，決定的な依存構造解析を行うことができる．

依存構造解析では，文中の異なる依存関係同士は互いに交差しないと考えられることが多い．しかし，現実には図3の例文のように依存関係間に交差が起こる場合[5]がある．YamadaやNivreの方法では，句構造解析で用いられていた統語解析アルゴリズムが流用されており，交差を許さないようになっている．一方，依存関係の間に交差を許す場合には，求める解析結果は，文中の一つの単語を根（root）とする有向木（directed tree）という形で表現できる．こう考えると，文中の各単語のペアの係りやすさを事前に計算し，全体の係りやすさの合計を最大化する全域木（spanning tree）を求めることが，係りやすさの合計値が最大の依存構造木の探索することと同等であることがわかる．McDonald[10]は，最大コストの全域木を求める効率よいアルゴリズムを用いることにより，入力文長 n に対して $O(n^2)$ の計算時間で動作可能な依存構造解析法を提案した．彼らのアルゴリズムでは，単語間の依存関係の評価値をその単語対と周辺文脈から得られる様々な属性の線形結合によって定義し，その線形式の係数を最適化するために，マージン最大化の考えにしたがったMIRA[6]という学習アルゴリズムを使っている．彼らのシステム（MST parser）は，以下から入手可能である．

http://ryanmcd.googlepages.com/MSTParser.html

依存構造解析は，明示的な文法規則が不要であり，学習データさえあれば言語に依存しない統語解析法である．上記のシステムは主として英語を対象にしており（Nivreは当初スウェーデン語を対象），Penn Treebankの句構造木を依存構造木に変換したものを学習データに用いていた．依存構造解析は言語に依存しないだけでなく，文内の単語の統語的な係り受け関係という概念が理解できれば，誰にでもその情報を付与することができる．句構造解析のように難解な文法を覚える必要がなく，どの単語がどの単語を修飾するかが理解できる作業者であれば，学習データを構築することができる．句構造文法に比べると解析の精度は若干劣るものの，学習データの構築の容易さやアルゴリズムの簡便さから，近年注目を集めている統語解析手法である．

ACL (Association for Computational Linguistics) のSIGNLL (Special Interest Group on Natural Language Learning) は，毎年CoNLL (the Conference on Natural Language Learning) という学習に基づく言語処理に関する会議を主催している．1999年以降は，shared taskといって，ある特定の言

[4] http://nlp.kuee.kyoto-u.ac.jp/nl-resource/corpus.html

[5] 文献[10]からの抜粋．この図では，依存関係の矢印が図1とは逆向きに記述されているのに注意．sawとyesterdayの間の依存関係がdogとそれを修飾している関係節の主動詞wasとの間の依存関係と交差している．

語処理タスクを設定し，共通のデータを用いて，機械学習に基づく言語解析システムの性能を評価するセッションを催している．同じ内容のタスクを2年連続で行うのが恒例になっており，2006年と2007年のshared taskが，“Multilingual Dependency Parsing”であった．英語を除く10以上の言語の依存構造を付与した学習用データが提供され，統計学習を用いた依存構造解析の性能が競われた．本年2007年6月の会議では，前年の多言語依存構造解析に加えて，“Domain Adaptation Track”が追加され，新しい分野のテキストに対する適応性の評価も行われた．

4. おわりに

統計的統語解析の動向として，句構造に基づく統語解析と依存構造に基づく統語解析の近年の進展について紹介した．句構造文法については，Charniak parserやCollins parserが，依存構造解析については，南瓜やMST parserなどがフリーソフトとして公開され，実応用に耐えるだけの解析精度を達成している．最近は，単に単語を指定して文書を検索する情報検索だけではなく，質問を言語文で表現し，その答えをWebから検索する質問応答システムや，特定の製品や事柄についての様々な意見を掘り出す技術として意見・評判情報マイニングなどのこれまでより深い言語解析が必要な応用が手がけられており，実応用レベルに達した統語解析システムの利用が必須となってきている．

統計的統語解析システムは，学習データを整備することで言語や分野に適合できるという特徴があるが，大規模な解析済みデータを準備することはそれほど容易ではない．今後は，解析済みデータと大規模な未解析データの両者を利用した半教師付き学習（semi-supervised learning）のような手法の利用が重要になるだろう．

参考文献

[1] Peter Brown, et al., “The Mathematics of Statistical Machine Translation: Parameter Estimation,” Computational Linguistics, Vol. 19, No. 2, pp. 263-311, 1993.

[2] Eugene Charniak, Statistical Language Learning, The MIT Press, 1993.

[3] Eugene Charniak, “A Maximum-Entropy-Inspired Parser,” 1st Meeting of the North American Chapter of the Association for Computational Linguistics, pp. 132-139, 2000.

[4] Michael Collins, “A New Statistical Parser Based on Bigram Lexical Dependencies, ” 34th Annual Meeting of the Association for Computational Linguistics, pp. 184-191, 1996.

[5] Michael Collins and Nigel Duffy, “New Ranking Algorithms for Parsing and Tagging: Kernels over Discrete Structures, and the Voted Perceptron,” 40th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, pp. 263-270, 2002.

[6] Koby Crammer and Yoram Singer, “Ultraconservative Online Algorithms for Multiclass Problems,” 14th Annual Conference on Computational Learning Theory, pp. 99-115, 2001.

[7] 工藤拓，松本裕治，「チャンキングの段階適用による日本語係り受け解析」，情報処理学会論文誌 Vol. 43, No. 6, pp. 1834-1842, 2002.

[8] David Magerman, “Statistical Decision-Tree Models for Parsing,” 33nd Annual Meeting of the Association for Computational Linguistics, pp. 276-283, 1995.

[9] Christopher D. Manning and Hinrich Schütze, Foundations of Statistical Natural Language Processing, The MIT Press, 1999.

[10] Ryan McDonald, et al., “Non-Projective Dependency Parsing using Spanning Tree Algorithms,” Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, pp. 523-530, 2005.

[11] Mitchell P. Marcus, et al., “Building a Large Annotated Corpus of English: The Penn Treebank, ” Computational Linguistics, Vol. 19, No. 2, pp. 313-330, 1993.

[12] Joakim Nivre, “An Efficient Algorithm for Projective Dependency Parsing,” 8th International Workshop on Parsing Technologies, pp. 149-160, 2003.

[13] Adwait Ratnaparkhi, “A Linear Observed Time Statistical Parser Based on Maximum Entropy Models,” 2nd Conference on Empirical Methods in Natural Language Processing, pp. 1-10, 1997.

[14] Hiroyasu Yamada and Yuji Matsumoto, “Statistical Dependency Analysis with Support Vector Machines,” 8th InternationalWorkshop on Parsing Technologies, pp. 195-206, 2003.