平文談話コーパスからの統語規則の自動獲得：人間の統語処理を考慮した解析システム（処理の詳細）

渋木　英潔
小樽商科大学大学院商学研究科

佐山　公一
小樽商科大学商学部

本発表では，入力素材から直接得られる情報（すなわち，コーパスに使われている文字列の形態情報）のみを利用しながら，事前情報のない純粋な平文談話コーパスを処理し，統語規則を自動的に獲得するシステムを提案する．本システムは，入力された文の中の単語の“順序”および，過去に獲得された統語規則と文法に関するデータにもとづいて統語規則を新たに獲得する．本システムは，どのような自然言語の文であっても統語規則を学習できる．

本発表では，本システムにおける処理のアルゴリズムを，日本語の実例を引きながら，具体的かつ詳細に説明する．システムの概要についてはワークショップで発表する．以下，第1節は“分節化”部門のアルゴリズムを，第2節は“統語解析”部門の一部である“統語カテゴリーの割りあてと置き換え”のアルゴリズムを説明する．第3節は“文法学習”部門のアルゴリズムを述べる．第4節はまとめである．

1．分節化

システムは，入力された文を，まず分節化する（“基本的な分節化”）．“分節化”部門では，以前に獲得した単語の形態情報を手がかりにし，文を単語に切り分け単語の列を作る．文の一部または全体に対応する単語の列のことを本システムでは“ピース”と呼ぶ．

“分節化”部門は，従来の形態素解析にあたる．ただし，“分節化”部門は解析部の一部であり，“統語解析”部門と処理結果のやりとりを行いながら，最終的な処理結果を出す点で，従来の形態素解析とは異なる．

システムは，入力文に対し，分節化を行いピースを切り出す．従来の形態素解析であれば，そのまま次の単語を切り出すことになるが，本システムはそうはせず，ピースを“統語解析”部門に渡す．“統語解析”部門では，そのピースに対して“ガイド”（解析木を作成するための適用順序の情報を含んだ統語規則の列）の作成を試みる．ガイドが作れた場合に限り，“基本的な分節化”を続け，その後すべてうまく行けば，最終的に文全体に対するガイドを作ることになる．

しかし，途中でガイドが作れなかった場合，システムは，直前の基本的な分節化の処理を誤りと見なす．そして，ピースの最後の1文字を削除した後，基本的な分節化をやり直す（“再分節化”）．それでもガイドの作成に失敗すれば，さらに再分節化を行う．

もしそのようにして文頭に達した場合，システムは再分節化にも失敗したと判断し，初めて再分節化を試みた段階の前の段階，すなわちガイドの作成に失敗した基本的な分節化の段階に戻る．そして，さらにもう一つ前の段階のピース（ガイドを作ることに成功した）の次の1文字を読み飛ばし，その読み飛ばした文字の次の文字から始まる既知語を見つけようとする．もし見つからなければ，見つかるまで読み飛ばし，飛ばした文字列を未知語と見なして心内辞書に登録し，未知語を含んだピースの作成を試みる（“未知語が存在する場合の分節化”）．未知語が存在する場合の分節化のイメージを図1に示す．

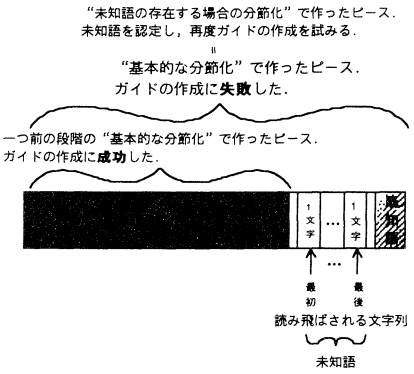

図1　未知語が存在する場合の分節化．現在の分節化，再分節化に失敗した後に試みる．

もしそのピースに対してもガイドを作ることができなければ，すなわち，“基本的な分節化”，“再分節化”，“未知語が存在する場合の分節化”のいずれによっても，ガイドの作成に失敗すれば，システムは，これらの方法とは異なる方法（“強制的な分節化”）を採る．この場合に限り，“分節化”部門だけで，ピースを強制的に作り出す．

1.1 基本的な分節化

システムは，それまでにガイドを作成していたピースの直後の文字（以前に当該文に対する処理結果のない場合は文頭の文字）から始まる既知語が心内辞書に登録されていないかどうかを検索する．既知語が複数見つかった場合は，文字数の多い既知語を使用する．次に長い既知語は，引き続く“統語解析”部門においてガイドの作成に失敗した場合にのみ使用する．

(1a) ここではきものを　ぬいでください．
(1b) ここでは着物を　ぬいでください．
(1c) ここで履き物を　ぬいでください．

たとえば，文例 (1a) の“ここではきものを”に対しては，(1b)，(1c) のような分節化が可能である．ここでは，分節化の様子を分かりやすくするために，漢字で表記してある（以下の例でも同様とする）．システムは (1b) を採り“ここでは”と分節化する．ガイドを作成できた時点で処理は終了し，“ここで”で同時に分節化を行うことはない．

1.2 再分節化

新たに切りだした文字列にあてはまる既知語がなくガイドの作成に失敗した場合，システムは一つ前の段階に戻って基本的な分節化をやり直す．システムは“統語解析”部門ですでにガイドを作成していた直前の基本的な分節化の処理を誤りと見なす．システムは，再分節化の対象となった文字列（基本的な分節化によってピースの作られていた文字列）の最後の 1 文字を削除し，それを新たな文字列とし，その文字列の中に既知語が存在するかどうかを検索する．存在しなければ，さらに 1 文字，合わせて 2 文字削除した文字列の中に既知語があるかどうかを調べる．以下，既知語が見つかるまで，順に末尾の文字を削除していき検索を続ける．もし文頭の文字まで戻っても発見できなかった場合，再分節化は失敗となり，改めて“未知語が存在する場合の再分節化”を行う．

(2a) ここではきものを　はきかえてください．
(2b) ここでは着物を　はきかえてください．
(2c) ここで履き物を　はきかえてください．

たとえば，(2b) のように“ここでは着物を”と分節化を行った後，“はきかえて”が心内辞書になく，“ください”のみが登録されていたとすると，システムは，直前の分節化の段階に戻り，“ここではきもの”，“ここではきも”，．．．“ここでは”，“ここで”と文字の削除と既知語の検索を繰り返し，“ここで”が心内辞書に登録されているのを見つける．そこで，システムは，“ここで”を新たなピースとし，以下，前節で述べた基本的な分節化を行い，(2c) のように“ここで履き物を”と新たに分節化し直す．

文例 (2) では，“ここではきものを”に対し，再分節化を試み，“ここで”が既知語であることを見つけたが，“ここではきものを”から新たな既知語“ここで”を除いた部分“はきもの”，すなわち，順に削除していった文字からなる文字列も既知語として登録されている必要がある．つまり，新たに発見した既知語の次の文字から，最初に削除した文字までの文字列が，一つ以上の既知語で分節化できなくてはならない．もしこれができなければ，別の既知語を検索しなくてはならない．

1.3 未知語が存在する場合の分節化

基本的な分節化，再分節化のいずれの方法でもガイドを作成できなかった場合，システムは，基本的な分節化を行いピースを作っていながらガイドの作成に失敗した最初の段階に戻る．そして，ガイドの作成に成功したその前の段階のピースから，現在のピースを作るために付け加えた文字列の最初の 1 文字を飛ばし，その次の文字から始まる既知語を検索しようとする．それでも既知語を発見できなければさらにもう 1 文字飛ばして 3 文字目から始まる既知語を見つけようとする．以下同様にして，既知語を発見しようとする．もし発見できた場合，飛ばした文字列を未知語と見なし，新たに心内辞書に登録する．そして，未知語を含んだピースに対し基本的な分節化を行う．

(3a) ここではきものを　はかないでください．
(3b) ここでは着物を　はかないでください．
(3c) ここで履き物を　はかないでください．

たとえば，(3b) のような分節化を行った後，“はかないで”が心内辞書になく，“ください”のみが登録されていたとする．システムは，1.2 で述べた方法で，(3c) の“ここで履き物を”のように再分節化するが，それでもガイドが作れなかったので，一旦，“ここでは着物を”と分節化した段階に戻る．その後，システムは，“はかないで”の“は”から“で”までを読み飛ばし，“ください”を既知語とし，“はかないで”を未知語とし新たに登録し，基本的な分節化を行う．

もし文末まで文字を飛ばしても既知語を発見できなかった場合，システムは，読み飛ばし始めた文字から文末の文字までを一つの未知語として登録し分節化する．

人間も，たとえば人名や地名などの固有名詞を含む文の場合のように，頻繁に未知語に出くわす．しかし，そうした状況であっても，人間はある程度問題の文を理解できるように思われる．こうした人間の言語理解の性質と“未知語が存在する場合の分節化”は似ている．

1.4 強制的な分節化

1.1 から 1.3 までの分節化の方法がいずれも使えない場合には，すでに分節化された文字列の次の文字から最長の既知語を検索し，発見できればそれをピースとする．発見できなければ，発見できるまで開始文字を 1 文字ずつ読み飛ばし既知語を検索していく．そのようにして発見された場合は，読み飛ばした文字列を未知語として分節化を行う．文末まで読み飛ばした場合は開始文字から文末までを一つの未知語とみなす．

2. 統語カテゴリーの割りあてと置き換え

2.1 終端カテゴリーの割りあて

文 (4a) が入力されたとしよう．システムは，心内辞書を参照しながら，この文を分節化しピースを作り，ピース内の単語に対し，たとえば (4b) のように終端カテゴリーを割りあてる．以下，統語カテゴリーを，括弧 [] と統語カテゴリーの略称を使い，[TC1]，[NC98] のように書き表す．括弧内の数字は，統語カテゴリーが獲得された時間的な順番を表す．

(4a) 太郎 は 東京 へ 汽車 で 行 く .
(4b) [TC2][TC13][TC41][TC23][TC42][TC21][TC33][TC43][TC18]

この文の単語のうち，“太郎”，“は”，“へ”，“で”“行”，“.”は，以前に入力された文に存在した既知の単語であり，すでに統語カテゴリーを割りあて済みである．そのため，これらにはそれぞれ [TC2]，[TC13]，[TC23]，[TC21]，[TC33]，[TC18] というようなとびとびの数字を与えている．また，“東京”，“汽車”，“く”は，システムにとって未知の単語なので，新たに統語カテゴリーを与え，ここでは [TC41] から順に番号を割りあてている．

2.2 統語カテゴリーの置き換え

(4b) の統語カテゴリー列の中の各カテゴリーに対し，統語知識を参照し，統語規則の適用を試みる．たとえば，[TC13]，[TC23]，[TC21]，[TC33]，[TC18] に適用できる“[PC17] → [TC13]”のような規則を見つけ，(4b) のカテゴリーの並びを (4c) のように変形する．

(4a) 太郎 は 東京 へ 汽車 で 行 く .
(4b) [TC2][TC13][TC41][TC23][TC42][TC21][TC33][TC43][TC18]
(4c) [TC2][PC17][TC41][PC28][TC42][PC26][PC34][TC43][PC20]

もし一つの統語カテゴリーに適用することのできる統語規則が二つ以上あれば，それら統語規則の使用頻度を比べ，頻度の高いものから順に適用する．

3. 文法学習

システムは，“文法学習”部門（学習部）において，解析部の処理結果をもとにして統語規則の学習を試みる．ここでの処理の対象となる解析部の処理結果は，2.2 で述べたピースに対する統語カテゴリー列である．“文法学習”部門での文法学習は，文に対するガイドがない場合に行われる．もし文のピースに加え文のガイドもうまく作成できていれば，ガイド作成のために必要とされる統語規則が統語知識の中にすでに存在していたことになり，文法学習は必要ない．

文法学習には，ピースの統語カテゴリー列と保留統語カテゴリー列の異なる部分を比較する方法と，“統語規則の候補”を使用頻度の情報を使って統語規則へ格上げする方法とがある．

文法学習は，一つの統語知識を学習したことが引き金になり連鎖的に別の統語規則の学習を引き起こすことがある（“連鎖的な文法・語彙学習”）．また，システムは，“文法学習”部門で，使用されない統語知識の削除も行う．統語知識の削除は，使用頻度の情報にもとづく．

3.1 ピースの統語カテゴリー列と保留統語カテゴリー列の異なる部分を比較した文法学習

システムが，文例 (4a) のピースに対する統語カテゴリー列を (4c) のように作成したとする．

(4a) 太郎 は 東京 へ 汽車 で 行 く .
(4c) [TC2][PC17][TC41][PC28][TC42][PC26][PC34][TC43][PC20]

(5) [TC2][PC17][TC41][PC28][TC9][PC26][PC34][TC43][PC20]

システムは (4c) と部分的に同じ順序の並びの保留カテゴリー列が統語知識内に記憶されていないかどうかを調べる．システムは，保留カテゴリー列 (5) を見つけ，(4c) と (5) に共通する統語カテゴリー列を対応づける．そして，同一の部分（＿＿と＿＿の部分）に挟まれた異なる二つの統語カテゴリー [TC42] と [TC9] とが同じ上位の統語カテゴリーに分類されるものと見なし，それらに共通の前終端記号を次のように割りあてる．

(6a) [PC44] → [TC42] (6b) [PC44] → [TC9]

異なる二つの統語カテゴリーが端にある場合もこれに準じて処理する．

(6)a，b の統語規則を獲得した後，システムは (4c)，(5) の統語カテゴリー列を一つにまとめ，(7) を作る．そして，以前の並び (5) を統語知識から削除し，代わりに (7) を保留カテゴリー列として記憶する．以後，(7) をピースから作られた統語カテゴリー列と同じように扱う．

(7) [TC2][PC17][TC41][PC28] [PC44] [PC26][PC34][TC43][PC20]

このようにして，統語規則を獲得するたびに，統語カテゴリー列の一部を，終端記号から前終端記号列に置きかえる．この時点で，次の文例 (8a) が入力され，(8a) のピースに対し統語カテゴリー列 (8b) を作成したとする．すると，システムは，統語カテゴリー列 (8b) を，記憶されていた (7) と対応づけ，統語規則 (9) を得る．

(8a) 太郎 は 東京 へ 自動車 で 行 く .
(8b) [TC2][PC17][TC41][PC28][TC45][PC26][PC34][TC43][PC20]

(9) [PC44] → [TC45]

先の保留カテゴリー列 (7) が作成された後，(10a) が入力されたとしよう．システムは，(4c) と (5) とを対応

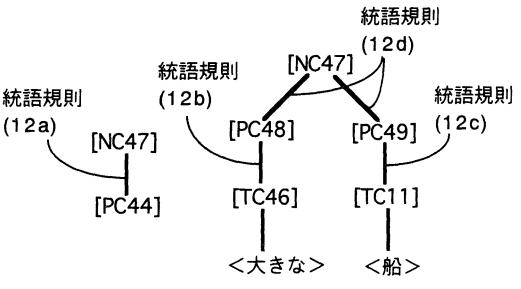

図 4 ピースの統語カテゴリー列と保留統語カテゴリー列の異なる部分を比較した文法学習．二つ以上の統語規則を獲得する場合．

づけたときと同じやり方で (7) と (10b) を対応づけ，(12)a， b， c， d の統語規則を作る．その後，システムは，(10b)， (7) を統語知識から削除し，代わりに (11) を保存する．この様子を，図 4 に示す

(10a) 太郎 は 東京 へ 大きな 船 で 行 く．
(10b) [TC2][PC17][TC41][PC28][TC46] [TC11][PC26][PC34] …
(7) [TC2][PC17][TC41][PC28] [PC44] [PC26][PC34][TC43][PC20]
(11) [TC2][PC17][TC41][PC28] [NC47] [PC26][PC34][TC43][PC20]

(12a) [NC47] → [PC44] (12b) [PC48] → [TC46]
(12c) [PC49] → [TC11] (12d) [NC47] → [PC48] [PC49]

3.2 “統語規則の候補” の登録

上の場合，統語カテゴリー列は，非終端記号の並びに置きかえられている．むろん入力されるのは常に現実の文であるから，非終端記号を非終端記号に書きかえる規則を獲得するのは，文が入力され，まず終端記号を前終端記号に書きかえる規則を獲得した後，再度統語カテゴリー列の対応づけを行う場合に限られる．この場合に限り，II型統語規則を獲得するたびに，統語カテゴリーの数のより少ない統語記号列も新たに獲得し，最終的に文を表す非終端記号一つからなる統語カテゴリー列を学習する．

システムが，単語と統語カテゴリーおよび統語規則を獲得していき，それらが十分な量に達しているとすれば，上で述べた “ピースの統語カテゴリー列と保留統語カテゴリー列の異なる部分に着目した文法学習” が多く行われる．しかし，十分な量に達していない時点，人間で言えば子供のころには，入力される文の多くは，過去に入力された文と異なる部分の方を多くもつと考えられるので，異なる部分に着目した文法学習はできないことが考えられる．そこで，保留統語カテゴリー列を記憶するたびに，統語規則の候補を作る．そして，同じ統語規則の候補ができた場合には，その候補の使用頻度を上げる．そのようにして，一定頻度に達したものを統語規則として登録するようにする．

(11) [TC2][PC17][TC41][PC28] [NC47] [PC26][PC34][TC43][PC20]

たとえば，保留統語カテゴリー列 (11) を保存した際，システムは，問題のカテゴリー列の統語カテゴリーの二つの並びをすべて登録する．カテゴリー列 (11) に対しては，(13) のようなペアを作りすべて登録する．

(13) [TC2][PC17]， [PC17][TC41] … [TC43][PC20]

3.2 “統語規則の候補” の統語規則への格上げ

(13) の 1 行目 [TC2][PC17] が一定の頻度に達したとすると，後に述べる “格上げの制限” を用いて，(14)a，b の二つの統語規則を得る．

(14a) [PC51] → [TC2] (14b) [NC52] → [PC51[NC17]

また，これら二つの統語規則を得たことにより，[TC2][PC17] は削除し，代わりに [NC52][TC41] を登録する．

格上げの制限：
統語カテゴリー二つの並びからII型統語規則（I型は不可能）を作る場合，その統語規則の左辺には NC しか来ることができないことにする．

統語規則 (14) を獲得する様子を図 5 に示す．

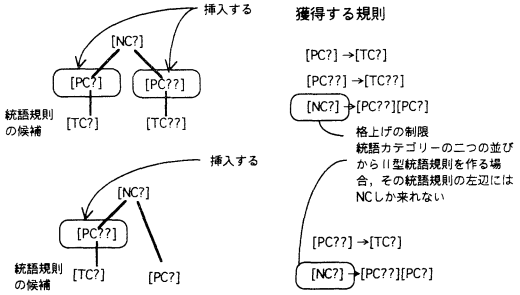

図 5 統語規則の候補の統語規則への格上げ．

統語カテゴリーのペアが，[TC1][TC2] である場合，それら TC をそれぞれ PC に置き換える規則とそれら PC を NC に書き換える規則の合計 3 つの統語規則を学習する．また，[PC1][TC1]， [TC1][PC1] のいずれかの場合には，TC を PC に置き換える規則とその PC を NC に書き換える規則の計 2 つの統語規則を学習する．

なお，これは文法学習ではなく語彙学習であるが，“複合語の候補” の複合語への格上げも “統語規則の候補” の統語規則への格上げと同じやり方で行う．たとえば，＜無＞＜関心＞という複合語の候補があり，これが一定の頻度に達した場合，＜無関心＞という複合語として登録されるようになる．

3.3 連鎖的な文法学習・語彙学習

3.1， 3.2 で述べたようにして統語知識を獲得すると，それが引き金となって，新たな統語知識の獲得を引き起こすことがある．その統語規則の獲得は，さらに別の統語知識の獲得に連続的につながる．

統語規則を一つ獲得すれば，それに伴い，未知語とその統語カテゴリーを新たに登録することになる．それゆえ，連鎖的な文法学習が起これば，同時に，二つの統語カテゴリーの同一化も連鎖的に生じる．また，複合語の新規登録，単語からの接辞の分離といった語彙レベルの連鎖的な学習も生じることになる．

4. まとめ

事前情報のない談話コーパスを処理し，統語規則を自動的に獲得するシステムを提案した．本システムは，人間の統語処理の仕組みを考慮しており，また，談話コーパス内の文字列の形態情報を最大限利用している．なお，本システムの概要はワークショップの中で発表されている．