構文の節結合強度に応じた書換え規則によるテクスト処理およびテクスト処理専用言語WRAPLの開発

佐良木昌　黒野健治
佐良木技術翻訳事務所・機械翻訳研究室　フリー研究者

I 英語表現の認識論的構造

英語においては、概括的抽象的表現からそれを補足する具体的再表現への叙述過程が基本である[1][1,2,3]。先に主節で、概念的な事柄や判断結論を述べ、続いて従属節で、具体的な事柄や理由条件を述べる。また英語においては、話者の判断・意思・態度は、be動詞や助動詞で表し[1]、語彙的には、直接的なものをゲルマン語で表現し、概念的なものをロマン語で表現する[2][4,5]（日本語の叙述順序は逆で、具体的な場面、時、条件の叙述が先になる[6,7]。[3]）逆に英語では、従属節を先に出すと、具体的な事柄や理由条件を強調することになる。また、これから述べる事柄・話題について、予め限定する場合には、主節の主部を関係節により限定するという叙述構造を採る（例えば冒頭で、これから論じる課題を明確にするとき、関係節を伴う長い主部が採用される[4]）。

関係節構文は、認識論的に、英語表現の中枢を成す。概念的な事柄から具体的な事柄へと認識内容を展開する認識＝叙述の基本構造は、関係節構文に典型的に示される。歴史的には、ラテン語的な関係代名詞構文[5]が、中英語期末から近代英語期前期に、英語へ取込まれることによって、疑問代名詞が関係代名詞として用いられるようになった[8,9]。同時に、上記の英語形成期において、文学表現および哲学が発達することによって、英語の認識＝叙述構造が確立した。

[1] 絵画的表現（文節文、貨物列車文、累積文等の認識＝叙述構造）等の検討は、今後の課題である。general picture から the picture in detail へという過程が基本にあると思われる。

[2] 句構造でも概念的な事を先述し具体的な事を後述する序列が基本である。ゲルマン語的配列では、名詞に対して形容詞前置、属格はA's Bであり、直接的な事柄が先行する。ロマン語的配列では、形容詞後置、B of A表現であり具体的説明は後述される構造をなす（詳細は別稿で行なう予定である）。[10,11,12]。

[3] 5W1Hということがらも、その叙述の優先順位は、認識＝叙述構造の違いから、英語では、Who, How, What; Where, When, 日本語では、When, Where; Who, What, Howとなる[6,7]。

[4] Another type of transistor that is particularly suitable for use in integrated circular is the field-effect transistor(FET), which is available in two types: … [ELECTRONIC CIRCUITS, McGRAW-HILL Electronic Engineering Series]

[5] 関係代名詞が語として独立しているのは、古代サンスクリット語と古代ギリシャ語のみであり[8]、古代ギリシャ人は、先に概念的な表現を主文にて叙述し、主文を限定する具体的な表現は関係代名詞節で叙述するという認識＝叙述構造を確立した[13]。SVO型の言語では、文の形式的構造上、話者・書き手の判断や意思・態度が先行することにも規定されている。ラテン語では、疑問代名詞を関係代名詞に充当するが、これが西ヨーロッパ諸語の関係節構文の基本となった[8,9,14,15]

II 分解的解析の適用限界と、関連性情報の明示化のためのテクスト処理

テクスト・文を木構造に分解する解析方法には、豊富な表層情報が失われるという問題が内在している。自然言語の工学的解析では、次の問題を考慮する必要がある。つまり、自然言語においては、統語構造が意味をもつ、言い換えれば統語構造と意味との有機的統一体が言語表現であるので、「統語構造のもつ意味を考えないで部分の意味から全体の意味を合成しようとする要素合成方式（原子論的方法）では、構造のもつ意味の欠落を防ぐことは困難と考えられる。」[16]

したがって、英語の統語構造の意味を、関連性情報として、あらかじめ抽出する必要がある。特に、節接続の機能をもつ語句の統語情報を抽出することが重要である。「自然言語システムが高度なものとなれば、構文的な制約や意味的な制約が絶対的なものではなく、制約によって強さに差があることを考慮する必要が生じてくる」[17]からである。

要素への分解という解析手法を取る前に、関連性情報の抽出という解析過程を採ることを、以下に提案したい。具体的には、関連性情報を抽出し、これを原文テクストに明示するためのテクスト処理を行なう。このテクスト処理は、同時に、解析木に沿った分析が可能なレベルにまで原文を簡素化・再編するので、機械翻訳の前編集としての意義をもつ。

原文に統語情報を付加するには、パターン・マッチングと書換え規則を用いる。このために、テクストを処理する専用言語として、WRAPLを黒野が開発した。

III 構文の節結合度に応じた書換え規則

英語の認識＝叙述構造に踏まえ、個々の構文のもつ統語＝意味情報を抽出して明示するために、段階化された構文の節結合度に応じた書換え規則を設定する。

A. 関係詞節（限定用法）
B. 分詞構文
C. that節構文
D. 相関対[6]
E. 関係詞節（継続用法）

結合度は最強をAとし最低をEとするレベルで表される。レベルAは二つの節が不可分の結合度であることを示し、レベルEは完全分離が可能であるほどの結合度であることを示す。書換えられた編集文は、節結合度に応

[6] 相関接続副詞と相関従位接続詞との対や、特殊な形式での、従属接続詞の対などを、相関対と呼び、相関対が生成する情報を相関情報と呼ぶこととする。

じて、特殊な文構造に再編する場合（A～C）と、二文に分割される場合（D,E）とに別れる。

A 限定用法の関係節

次の二つの場合に整理して書換え規則を設ける。主語を限定する関係節（先行詞とは不可分離の統語構造）、目的語を限定する関係節（照応関係を明示することによって、概念的な記述と、具体的な記述とに分割可能な統語構造）。**関係代名詞節**の書換え規則の具体例を示す。

[1] 特殊的な書換え規則

① 主節の主語を関係節が修飾しているとき[7]

＜NP which VP_2～(,) VP_1・・・＞ → NP=Subject, VP_1=Predicate

＜NP which VP_2～ : The NP above＋VP_1・・・＞

② 目的語を限定する関係節であっても、限定詞が目的語（NP）を限定しているとき

＜NP_1+VP_1+ only|even NP_2 which VP_2～＞→

＜NP_2 which VP_2～: NP_1+VP_1+ only|even the NP_2 above.＞

NP_1=Subject, VP_1=Predicate

[2]一般書換え規則

＜NP_1+VP_1+ NP_2 which VP_2～.＞→

＜NP_1+VP_1+ NP_2: This NP_2+VP_2～.＞

＜NP_1+VP_1+ NP_2 + $prep + which NP_3 + VP_2～.＞→

＜NP_1+VP_1+ NP_2: $prep this NP_2, NP_3 + VP_2～.＞

指示語（the, above, this）により先行詞（NP1）を明確に指定し、かつ前置詞句を文頭に配置することで、二文の強固な結合関係が、関連性情報として記述・保持される。

B 分詞構文

分詞節に従属接続詞を付加することで副詞節としての従属節とすることができる。この接続詞により結合された主節・従属節となるので、結合度はやや強い。接続詞により、統語情報と接続の意味（時・条件など）が与えられる。分詞構文の諸形態に応じて書換え規則を設ける。以下に、分詞構文の諸形態と書換え規則例を示す。

[α] 分詞構文が現在分詞で始まる場合

＜Ving ～, X-clause＞→＜$Conj＋the below+Vs, X-clause＞

[β] 分詞構文が前置詞＋分詞で始まる場合

＜Conj ＋Ving ～, X-clause＞→

＜Conj+the below+Vs ～, X-clause＞

[γ] 独立分詞構文（主文と主語を異にする）が主節の後に来る場合

＜X-clause, NP+Ving ～＞→＜X-clause, $conj NP+V ～＞

[θ] 主節の一部に挿入された分詞構文の場合

＜NP(Subject), Ving～, VP…＞→ ＜$Conj ＋ that+Vs, NP+VP…＞ Vs は動詞現在形

C. 相関対

さしあたり次の三形態に分類する（個々の統語の認識論的構造については、別稿に回したい）。

①先行副詞および相関従位接続詞

＜NP + VP + so adj that X-clause.＞→

＜NP + VP + so adj. Then, X-clause. ＞／

＜NP + VP + so adj : X-clause, in this degree.＞

②従属接続詞の対

＜X-clause partly becauseY-clause, partly because Z-clause.＞ → ＜X-clause. A part of the reason is that Y-clause. Another part of the reason is that Z-clause.＞

③相関節（形容詞・副詞の比較級が主節・従属節の先頭に位置する）

＜The ad-1 X-clause , the ad-2 Y-clause＞→

＜By how much ad-1 X-clause, by so much ad-2 Y-clause.＞

D. that節構文

① that節を目的語とする構文

make sure that節など。

＜・・・ V+C+that $clause ～＞

→＜・・・ V+C+the following matter: $Sentence ～＞

② It_that 構文は、強調点を文頭で明示するという構文上の意味をもつ。仮主語と that の間に おかれた語句を、文頭に配置し、かつコンマで区切ることで、強調点として明示して単文にするという形で、構文の編集を行なう。

＜it is $adj that X-clause.＞→＜$adv, X-clause.＞

＜it is AdP that X-clause.＞→＜AdP, X-clause.＞AdP は副詞句

E 継続用法の関係節

関係代名詞節の書換え規則を例示する。

①＜X-clause, QP of which～＞→＜X-clause. QP of them ～＞

② ＜X-clause, $prep which ～＞→＜X-clause. There$, ～＞

Qp は数量代名詞 、$prep は前置詞、There$は接続副詞を表す。この接続副詞は、二文を意味的に関連付ける語句であり、前置詞＋関係代名詞や、関係副詞と意味的に同等である。なお、統語情報を抽出し訳文に反映するためには、原文に、統語情報を付加すると共に、結合度に応じた形態にて単文へと再編するが、そのための**記述シンボルとして接続副詞を活用する**ことを提案したい[8]。接続副詞を活用することで、接続詞のもつ意味情報を保持しながら原文を単文に編集することが可能となる。

---

[7] 限定用法関係代名詞節の場合、主節の主部が関係節により修飾されるときには、最も関係節による意味限定が強くなる。したがって、関係代名詞節の訳出が先に行なわれ、日本語訳文において先に記述される必要がある。そのためには、英文を分離・編集するときに、必ず関係代名詞節を含む主部を単独で先に記述しなければならない。意味的には、関係代名詞節の意味内容が先にあってはじめて当該文の意味が論理的に成立する（A restricted by relatives is B. →A restricted by relatives: This A is B.）。なお、関係節の重層構造のパターン化と書換え規則については別項にて行なう予定である。

[8] 接続副詞の機能分類を例示する。回帰(return to the point) 追加(Addition) 比較と類似(Comparison& Similarity) 譲歩条件(Concession, Qualification) 結論(Conclusion) 後述(Consequence) 反論(Contradiction) 対照(Contrast) 余談(Digression) 例示(Exemplification) 理由(Reason) 反復(Repetition) 連続(Series) 話題転換(Shift of subject) 特定(Specification) 概括(Summation) 時間的継起関係(Temporal relationships) 不確実性(Uncertainty)

Ⅳ テクスト処理専用言語 WRAPL

1．WRAPL(Ver1.1)の特徴的機能[17]

①非膠着語文のテキストファイルを対象とする。

スペースにて単語が区切られた英文等の非膠着語文からなるテキストファイルを、置換えルールを記述したルールファイルに基づいて自動的に異なる並びに置換え、テキストファイルとして出力する。ただし、スペースを適当に配置すれば日本語等の膠着語文にも適用可能である。

②置換えルールは単語の並びを記述する。

キーとなる単語以外の単語は、単語の文字列自体を意識せずに、置換えルールが記述できる。

③置換えルールは基本的に１種類の置換え命令で記述する。

補助的にジャンプ命令が存在するが、実質的には１種類の置換え命令を組み合わせて置換えルールを記述する。

④処理対象テキストファイルから１論理行ずつ読み込んで置換えルールで処理する。
各論理行を先頭から順に、論理行の単語配列をルールファイルの置換え命令とパターンマッチさせる。

⑤単語配列のパターンマッチが成功すると、マッチング処理により各単語または単語群に付与された「単語番号」を用いて、自由に配列を並べ替えまたは削除して出力できる。同じ単語番号を重複して使用できる。

⑥出力時に並べ替えと同時に、各単語の文字列の変換や新たな単語または単語群の挿入が可能である。

2．正規表現または正規表現を用いた処理系との比較

①単語の文字列を考慮せずに処理できる。

正規表現は文字単位のパターンマッチであるため、単語の文字列および単語間のスペースを意識した表現が必要である。正規表現は単語数を考慮した表現が困難である。

②常に１論理行全体へのマッチングであり、部分的マッチングはしない。

正規表現を用いた処理系は、１論理行の部分に対するパターンが記述できる。WRAPL(Ver1.1)は、必ず１論理行全体に対するパターンを記述しなくてはならないが、ほとんど複雑になることはない。

③置換え命令は、単語番号で単語または単語群を特定している。

正規表現を用いた処理系では、表現の一部を囲っている『(　)』の左からの順番に基づいて特定の変数と対応させているが、変数と正規表現との対応が完全ではなく、常に同じ変数が同じ表現部分を示すとは限らない。このため、正規表現を用いた処理系では望まない置き換えがなされる場合がある。

④出力時の単語の変換処理

単なる、単語の並び替えのみでなく、置換え命令の中で同時に単語毎の変換処理、単語または単語群の挿入が可能である。正規表現を用いた処理系では、置換えの後に別の命令でしなくてはならず、複雑化する。

⑤前述のごとく命令が少ない。

⑥キーとなる単語のパターンの記述が簡便。

3．ルールファイルに記述できる命令

次の２命令を基本単位とする。

①置換え命令：

『行番号＝パターンエリア/出力エリア
＝肯定ジャンプ先？否定ジャンプ先』

②ジャンプ命令：

『行番号=>ジャンプ先』

実行は、ファイル先頭の命令から開始される。その後は、パターンマッチの結果やジャンプ命令に応じる。

4．パターンエリアの記述・機能

4－1．単語単位のパターンの記述

①論理行単位で単語数にてパターンを記述する。

・文字列の代りに、『{ }』および『( )』を用いて、１論理行全体のパターンを記述する。

『{ }』,『( )』の記述例：『(5018)』,『(215)?』,『(120)+』,『(31)*』,『(467)3-5』,『(111)2-』,『(9843)-10』,『(674)6』,『{20 such}』,『{40 that}?』

・『(』,『{』の直後に単語番号(以下、nn で表す)を記述する。

・『)』の後の『?,+,*,2-,3-5,-10,6』、『}』の後の『?』が単語の数を表している。省略は『1』を表す。

②『{ }』と『( )』との使用上の区別

・『{ }』は、単語番号の後にスペースを挟んで主にキーとなる単語を指定する。具体的に指定できる単語が適する。

・『( )』は単語番号のみで、単語は指定しなくてもよい。具体的に特定できない単語または単語群の指定に適する。『{ }』と同様に単語番号の後に単語を記載して該当する単語を指定することもできる。

③『{ }』,『( )』内における単語のパターン表現

・単語をそのまま記述：『{nn that}』,『(nn that)』と記載する。ただし、

・『¯such』：『such』以外の単語。

・『*ing』：末尾に『ing』が存在する単語。

・『un*』：先頭に『un』が存在する単語。

・『*tion*』：先頭から末尾までのいずれかの位置に『tion』が存在する単語。

・『un*ing』,『un*tion*ing』,『un*tion*』,『*tion*ing』,『¯*ing』,『¯un*ing』,『¯un*』といった組合わせも可能である。ただし、『*』は２つまでである。後述する出力エリアの『*』とは機能が異なる点に注意すべきである。

・パターンを『|』で接続すると、『¯』がないパターンの間は論理和、『¯』があるパターンは『¯』がないパターンに対する論理積となる。

4－2．パターンマッチ処理の特徴

①『{ }』と『( )』との処理順序の特徴

『{ }』が優先的に処理される。

パターンエリアの最左位置の『{ }』からパターンマッチされ、1論理行内での位置が決定される。1つの『{ }』の位置が決定される毎にその『{ }』の前のすべての『( )』のパターンマッチ処理が行われる。

②複数の『( )』は前方最多一致

『{ }』の前、あるいは『{ }』と『{ }』との間の単語群に対して、複数の『( )』が対応している場合には、その内で前にある『( )』ほど多数の単語が対応するようにパターンマッチされる。

③『{ }』を用いずに、すべて『( )』で記述することも可能であるが、『{ }』を用いた場合とは、処理の順序の特徴から、パターンマッチの結果が異なることがある。

5．出力エリアの機能

5－1．単語単位の置換え

①パターンエリアで割振られた単語番号単位による出力

例：『100,20,1,50,1000』

②単語単位の文字列変更処理

例：『20(*ing(2)ed)(un*(1))|(*ed(2)ing)』

＜詳細はWRAPLマニュアル参照のこと＞

③文字列の挿入。

単語単位に限らない、複数の単語や改行も可能。

例：『100,20,{¥n　It is},1,50,1000』

6．開発中のWRAPLの新たな機能およびJWRAPL

6－1．処理中の論理行の前後の論理行に対しても、パターンマッチ可能とする。

処理対象論理行の前後の記述関係も出力に含められるようにする。文章のパターンマッチも可能とする。

6－2．出力エリアにおいて、異なる単語番号のパターンマッチを可能とする。

例：　20(10 he|she)(*(1-)s)

(ただし、これは複数行の置換え命令を用いれば、同様な処理は現状でも可能である。)

6－3．サブルーチンの記述を可能にする。

①処理内容の整理の容易化

各サブルーチンで行番号が独立する。

②個別に開発したルールファイルの統合容易化

6－4．JWRAPL

日本語についても、所定のルール、たとえば、字種切り法で分離する等により、字種単位またはその組合せでの置換え処理を可能にする。

【参考文献】

[1]宮下眞二[1982]『英語文法批判』日本翻訳家養成センター

[2]宮下眞二[1985]『英語はどういう言語か』季節社

[3]江原暉将[1995]「多次元尺度法を用いた語順パラメータの間の関係付け」言語処理学会第1回年次大会発表論文集 pp.173-176

[4]佐良木昌[1995]「シノニムに隠されたメタファーの意味－英語表現構造の一般モデル」自然言語処理シンポジウム、自然言語処理における文脈

[5]佐良木昌[1995]「シノニムに隠されたメタファーの意味」英語表現学会第2回地区研究発表会

worth, work と value, labour とはシノニムであるが、直接的な表現では worth, work を使い、概念的な表現では value, labour を使用する。前者はゲルマン起源語であり、後者はラテン借入語である。

[6]中村元[1989]『日本人の思惟方法』中村元選集第3巻、春秋社

[7]中村元[1995]「推理についての考察2 ／第6章民族によって異なる推理の表現様式」「現代思想」1995,7、青土社

[8]karl Brunner [1960,1962] "Die englische Sprache: Ihre geschichtliche Entwicklung" 邦訳書名『英語発達史』大修館書店

[9]中尾俊夫『英語史Ⅰ』英語学体系8、大修館

[11]河原俊昭、春田勝久[1992]「形容詞の配列順序の事例研究」「*telos*」第8号、金沢経済大学人間科学研究所

[12]Henry Bradley [1904] "THE MAKING OF ENGLISH" 邦訳書名『英語発達小史』岩波書店

[13]ARISTOTLE『詩学 20章』アリストテレス全集17巻 岩波書店

[14]池上岑夫編『日本語小文典（ロドリゲス）』の解説 岩波書店

[15]Edward Gibon『ローマ帝国衰亡史1』筑摩書房

[16]池原・宮崎・白井・林[1987]「言語における話者の認識と多段翻訳方式」情報処理学会論文誌 vol.28, no.12, pp.1269-1279

[17]Ralph Grishman [1986] "Computational Linguistics: An Introduction", 邦訳書名「計算言語学（コンピュータの自然言語理解）」山梨訳、サイエンス社

[10]Donald E.Knuth, et al. [1989] "Mathematical Writing" MAA Notes Number 14,The Mathematical Association of America

クヌース先生によると、具体的な事柄を概括する語、または上位の概念 ***two conditions*** を表わす語を用いて概念を先に述べ、次に具体的な事柄<grunt> and <snort> を記述する。こうすると左から右へと文章の流れに沿って叙述された筆者の認識内容を、読者が理解しやすい。

"Try to make sentences easily comprehensible from left to right. For example, 'We prove that <grunt> and <snort> implies <blah>. ' It would be better to write. 'We prove that ***two conditions*** <grunt> and <sort> imply <blath>.' Otherwise it seems as first that <grunt> and <snort> are being proved."

[17]WRAPLマニュアル

ルール記述の例を示す。

;継続用法の関係代名詞の書換え規則

120=(1)(2)+{3 ,}{4 about|at|between|by}{5 which}(6)(7)+/1,2,{;},4{There*},{;},6,7=121?121