理解容易性を指向した訳語/統語構造選択規範に基づく文生成

吉村 裕美子† 平川 秀樹†

従来，言語間で異なる個別言語現象をいかに文法の枠内で処理するかという点が機械翻訳における文生成の研究の中心であった．しかし，機械翻訳システムの実運用における入力文は複雑で多岐に渡るため，個々には正確な文法的処理でも，単純に組み合わせただけでは訳文全体の理解容易性を高めることに必ずしもつながらず，全体構造がわかりにくいという事例によく遭遇する．一方，テキスト生成の研究において，出力される表層文あるいは生成前の中間構造を評価し，それをもとに推敲を行う機構に関するものがある．これらの機構では，生成の対象に対する想定が広いため，比較的自由にダイナミックな語・統語構造の変換を適用できるが，これを翻訳に応用する際には，翻訳の評価尺度の一つである忠実度を考慮する必要がある．本論文は，翻訳の忠実度を踏まえながら訳文の理解容易性の向上に焦点を置いた，訳語・統語構造の選択を制御する方式について述べる．統語関係・修飾関係の曖昧性を回避し，統語的バランスを取りつつ，意図する意味を最大限反映するために，本方式においては，概念構造に関する重さ・構造の情報，原文中の語順情報を主要なキーとして利用し，適切な表層表現の選択，語順・パンクチュエーションの制御を遂行する．本方式の効果を見るため，計算機操作説明書331文に対して実験を行ったところ，9.4%の文において訳文の理解容易性の向上を見た．

Control of Lexical and Syntactic Choices for Improving Translation Comprehensibility

YUMIKO YOSHIMURA† and HIDEKI HIRAKAWA†

Research on sentence generation in machine translation has focused on how to treat language specific phenomena within the framework of that particular grammar. It is often difficult to grasp the global structure of complex sentences found in practical documents even though all the local sub-structures satisfy the grammatical requirements. Therefore, merely combining individual grammatical treatments would not improve the comprehensibility of output sentences. Meanwhile, in text generation, some has proposed models which revise output sentences by evaluating their intermediate representation or their surface sentences. This leads to dynamic changes in words and syntactic structures. For application to MT, the fidelity of output sentences need to be considered. This paper proposes a method to control lexical and syntactic choices considering fidelity, to achieve higher comprehensibility. To generate well-balanced sentences and convey the original meanings with the least ambiguity in syntactic relations, our generation system employs information on intermediate structures including their weight as well as word order of the original sentences so as to control syntactic patterns, word order and punctuation of output sentences. We have conducted experiments using 331 sentences from a computer operation manual. The result shows that this method will improve the comprehensibility of these sentences by 9.4%.

1. はじめに

自然言語生成は「what to say」と「how to say」の二つの過程からなる．機械翻訳においては，「what to say」は原文の概念自体として与えられ，また「how to say」のうち語用論的側面の決定もすでに原文において行われていると見ることができる．そこで機械翻訳システムに残るタスクは，「how to say」の残る選択余地の中で，原文の中の情報を目的言語の自然で理解容易な表層文字列として翻訳することにある．

ビジネス利用の機械翻訳システムが対象とする実際の翻訳原文書は，長文や構造の複雑な文が多くその表現も非常に多彩である．一般に，言語族が異なる言語間の翻訳では，文法体系の差が顕著なことから，一見単純な文の翻訳においても曖昧性が生じることがある．その差の一例が修飾句と被修飾句の語順の違いで

† (株)東芝 研究・開発センター
Research & Development Center, TOSHIBA Corp.

ある．この違いは日英翻訳における曖昧性の原因の一つとなっている．英語では前置修飾と後置修飾が可能だが，日本語では前置修飾のみが可能である．原文が複雑な場合はさらに曖昧性が増し，これはしばしば解釈不能な訳文に繋がったり文意の取り違いを招くことになる．このように，訳文がいかに目的言語の文法の観点からは正しくても，原文の意味が正しくとらえられないような文ならば翻訳としての価値はない．そこで，我々は訳文の理解容易性をよりよい翻訳のための重要な規範としてとらえる．

機械翻訳における生成に関する従来の研究の多くは，言語間で異なるある言語現象をいかに文法的に処理するかという点に焦点を置いていた[1)~3)]．このような個々の文法的処理は翻訳には欠かせないものである．しかし，個々の処理が正確でも，実際の文章の翻訳においては必ずしも訳文の理解容易性を高めることに繋がっていない．局所的には文法的な処理を各所に施していても，実際の入力文は複雑で多岐に渡るため，係り関係を含めた訳文全体としての統語構造がわかりにくいという例によく出会う．

一方，機械翻訳に限定しない一般的な生成の問題として，表層文における語および統語構造の選択に関する研究が多数報告されている[4)~14)]．このうち，特に統語構造の大局的変換制御をも可能にするものとして，出力される表層文あるいは生成前の中間表現を評価しそれをもとに出力文の推敲を行う機構に関する研究がある[9)~14)]．これらの評価機構が有する文の評価の規範は，機械翻訳の訳文を評価する上でも有効な尺度となる．しかし，検出された要対処事項を回避するための手段としての語・統語構造の変換許容性という点においては，機械翻訳は翻訳という性質上大きな制約を担っている．すなわち，翻訳においては，原文書に表れている話題の展開の仕方，叙述内容に対する筆者・話者の態度などを可能な限り訳文中にも反映することが求められる．訳文の忠実度は翻訳の重要な評価基準の一つである．上記推敲機構は，このような翻訳特有の生成自由度の制約を前提とするものでないため，ダイナミックな語・統語構造の変換を比較的自由に適用できるが，機械翻訳に応用する際には，新たな尺度として翻訳の忠実度を考慮する必要がある．

本稿では，翻訳における忠実度も踏まえながら訳文の理解容易性を高めるという視点から，訳語および統語構造の選択の扱いについて論じる．実用の翻訳において訳文の理解容易性は翻訳の品質を大きく左右する．日本語と英語のような言語族の異なる言語間の翻訳ではなおさらその度合いが大きい．

本稿は，ビジネス利用の機械翻訳システムを通して，多くの実文書の翻訳に触れてきた結果蓄積された[15)~18)]より良い生成のための規範について述べる．従来，このような実システムの利用を経た議論は見られなかった．まず，一般的な問題を例を用いて述べる．次に，訳文に対する意味の誤認識や文の読みにくさ・不自然さを回避するために，曖昧性を抑え原文の意味を最大限反映した理解容易な文の生成をどのように制御するかについて論じる．

2. 理解容易性を上げるためのアプローチ

構文・意味解析の結果として得られる概念依存構造は，通常ネットワークや木構造など，文法的制約のない二次元構造で表現される．一方，出力すべき表層文は一次元の単語列であり，満たさなくてはならない文法を持っている．また，単純に統語的制約では説明できない認知的側面からの実際的制約もある（例えば，中央埋め込みのネストの深さに関する制約）．この生成過程で留意すべき点は，これらの制約内で，概念構造のもつ意味を失わないまま，自然で理解容易な文を生成することである．

以下，訳文の品質に関して問題となっている事例を紹介するとともに，これらに対処し理解容易性を上げるために必要となるアプローチについて述べる．

2.1 統語関係の曖昧性の回避

英語の文章において，ある動詞句の中に別の動詞句が埋め込まれると，その後に位置する副詞句の統語構造上の位置づけは一般に曖昧になる．すなわち，直前の動詞句を修飾するものか，あるいは，その句を飛び越えたより上位の動詞句を修飾するものかという曖昧性である．ところが，人間による文生成は，文脈から明らかでないかぎり，実際の発話であれば，発話のスピードを変えたり，空ける間の長さを調節したり，イントネーションを調節したり，語順を変換したりすることにより，この問題を解消することができる．一方，文字として書かれる文章となると，間を調節する代りにパンクチュエーションが効果的に用いられたり，語順の操作が意図的に行われる．次に挙げる(1)-(2)は，以上のような工夫なしに生成された文章である．

(1) Distribute the data processed through the network at this stage effectively.

(2) Print out the data processed through the network at this stage using the command "ymk".

これらは，シンタックス上それぞれ "effectively", "using…" の部分が直前の動詞句 "processed…", "processed…" を修飾するのか，その上位の動詞句である "Distribute…", "Print out…" を修飾するのかは厳密には明確でない．しかし，直前に存在するという点で，前者の解釈のほうがより自然である．後者の修飾先を暗示させるためには次のような語順操作やコンマの挿入が求められる．

(1)′ Distribute effectively the data processed through the network at this stage.

(2)′ Using the command "ymk", print out the data processed through the network at this stage.

(2)″ Print out the date processed through the network at this stage, using the command "ymk".

また逆に，後置される従属副詞節の前に単純にコンマを挿入するという規則に従うと，(3)のように修飾先が埋め込まれた文であると，コンマの存在により直前の動詞句を飛び越えてその上位の動詞句を修飾しているようにとらえられてしまう．(3)において "if…" が直前の "is…" を修飾しているものとして生成するにはコンマの挿入を抑える必要がある．

(3) You have to know that the current directory immediately after logging in is a root directory, if there is no specification about it.

(3)′ You have to know that the current directory immediately after logging in is a root directory if there is no specification about it.

このような操作は，ある句とある句との統語的関係のみを考慮するだけでは十分に処理しきれない．ある修飾句とその被修飾句との間にどんな構造のどの程度複雑な句が生成されるか，また修飾句の構造はどんなものかという観点に立って初めて可能となるからである．概念依存構造からの生成の場合，部分的な構造は同じでも，それがどういう構造下で生成されるかで要求される語順の制御は大きく異なってくる．よって，生成規則中で状況に応じて処理を調整できるような柔軟なコントロール機構が必要である．

2.2 読みにくさ・不自然さの回避

英語では，end-weight の原理により，重い句，構造の複雑な句は文尾の位置が好まれる．また，新情報は重い構造を持つことが多く，end-weight の原理が end-focus の原理と協同して句の外置を促す[19]．

このような原理を考慮せずに生成した文は，前節でも述べたように，重い句，構造の複雑な句の後に置かれた句と真の修飾先の句との関係が不明確になり，文全体の統語構造が理解しにくく，文のバランスも悪い．また，複数の語が共起して新たな意味を構成するような場合，それらの語の間に長い句が生成されると，最終的には統語上の曖昧性はなくても文を読み進める上でなかなか文全体の意味が確定されず読みにくい，という問題が生じる．

次の例(4)-(7)はその典型例である．(4)は文の中の位置に重い目的語をかかえている．文の読みやすさを考えれば，表現自体を例えば(4)′のように変換する必要がある．(4)′では重い句が末尾に置かれておりバランスもよく読みやすい．(5)では "make" と "available" とで一つの意味を構成しているのだが，目的語が長いため "available" まで読み進めるまでの語列が長く，なかなか "make" の意味が確定できない．これに対しては，(5)′のように "available" を目的語の前に置くと自然で理解しやすい文になる (SVOC から SVCO への転換)．(6)の "set" と "aside" も同様で，距離があるため "set" の意味がとらえにくく (6)′への転換が望まれる (SVOA から SVAO への転換)．(7)は(5)と同じ "make…available" の構造を持っている．(7)の目的語は(5)のような複雑な構造の修飾句を従えてはおらず，単純な名詞句の並列からなっている．そのため，その後に位置する句の統語関係を特に曖昧にしているということはないが，(5)と同様に "make" と "available" を遠ざけてしまうことから，不自然で読みにくい文となっている．

(4) The research planning center provides this national network including some computers in Europe with funds.

(4)′ The research planning center provides funds for this national network including some computers in Europe.

(5) We have made a real-time on-line translating communication system connected via communication satellite available.

(5)′ We have made available a real-time on-line translating communication system connected via communication satellite.

(6) We cannot set a whole system of rules devised by Congress itself aside.

(6)′ We cannot set aside a whole system of rules devised by Congress itself.

(7) We have made a voice recognition server, a translation server, and a phonetic synthesizing server available.

(7)′ We have made available a voice recognition server, a translation server, and a phonetic synthesizing server.

以上のような統語構造・語順の制御には，概念依存構造中の部分構造の重さや構造に関する情報を参照できる機能を導入する必要がある．

2.3 原文中の修飾のスコープ情報の利用

一般に，被修飾句から離れた修飾句ほど修飾のスコープは広いといえる．日本語では修飾句は常に被修飾句より前に置かれる．英語では，一般的な制約のもとで前置修飾と後置修飾とがある．翻訳においては，原文中の修飾のスコープを訳文に最大限反映することが，訳文の理解度を上げる重要な鍵となる．

日英翻訳では，修飾句の各々について前置修飾とするか後置修飾とするかを構文生成時に入力の構造に応じて適切に判断している．生成処理の前段階では，各部分構造がどんな句としてどんな生成位置制御を受けるべきか，十分に予測されない．その決定には，各修飾句の構造・重さ，句の持つ属性など種々の情報が参照され判断の元になっている．この決定に基づいて修飾句の生成順を制御する必要がある．すなわち，前置修飾と決定された句については原文中の生起順に従い，後置修飾と決定された句については原文中の生起順とは逆の順に生成を行う．

例えば，(8)のような単文においては “in the transfer method” と “at the transfer stage” は共に前置修飾とすることも後置修飾とすることも可能である．一方，同じ概念依存構造が(9)のように名詞句として表現される場合には，いずれも後置修飾のほうが望ましい．それは，この前置修飾句の被修飾句の構造の形態（関係節）によっている．(10)では前置詞句は共に後置されているが適切ではない．より修飾のスコープの広い “in the transfer method” が “at the transfer stage” よりも前に生成されているからである．

(8) In the transfer method$_1$, conceptual structures of the source language are converted into those of the target language at the transfer stage$_2$.

(SL) トランスファ方式では$_1$ トランスファ過程で$_2$原語の概念構造が目的言語の概念構造に変換される．

(9) conceptual structures of the source language which are converted into those of the target language at the transfer stage$_2$ in the transfer method$_1$

(10)* conceptual structures of the source language which are converted into those of the target language in the transfer method$_1$ at the transfer stage$_2$

上記の点より，生成過程で，各修飾句について前置修飾とするか後置修飾とするかを決定するのに応じて，前置修飾句同士間，後置修飾句同士間の生成順を制御できるような柔軟なメカニズムが必要である．

3. 生成処理の概要

インプリメントについて述べる前に，上記アプローチが対象とする機械翻訳システムの生成処理の概要について説明する．

本機械翻訳システムはトランスファ方式を採用している．入力文に対する構文・意味解析処理の後，トランスファ過程において，解析結果である原言語の概念依存構造（木構造で表現される）から目的言語の概念依存構造および疑似統語構造への変換が行われる．意味解析処理が終わった段階で，原文中に並列句や従属節構造に伴って格要素の共有（省略）がある場合には，概念依存構造中に共有（省略）要素の補完が行われている．これは，例えば埋め込み表現などで，動詞句部分に修飾先である格要素が欠けていても，修飾先のノードを痕跡ノードとして概念依存構造中にコピーして補うことにより，埋め込み表現でない通常の動詞表現と，後に適用される語彙トランスファ規則を共有化することにつながる．

トランスファ過程の前半をなす語彙トランスファ過程では，基本的に，各概念に対応する目的言語の表層語（訳語）の決定とその語の使用に伴う統語情報の付与までを行う．従って，ここで得られる概念依存構造は純粋に目的言語の概念の関係だけを表すものではなく，目的言語の統語情報で色づけを施された疑似概念依存構造である．ここで用いる知識はすべて個々の概

念ごとに規則として記述されている．各概念に対する訳語の決定・統語情報の付与は，概念依存構造中の依存関係の下位にある概念から順番に行っていく．すなわち，依存関係の上位の概念に対して訳語を決定する際には，下方の部分構造の訳語の選択結果を参照することができる．

トランスファ過程の後半は，個々の語に依存しない一般的な構造の変換を行う構造トランスファ過程である．ここで用いる知識は一般規則として記述され，手続き的に適用が行われる．この過程では，語彙トランスファ過程で局所的に付与された種々の情報を，関与する構造全体に伝搬させる処理も行う．例えば，目的言語が英語の場合の例をあげると，ある動詞の目的格に動詞表現を伴う際に表層形態（that 節，to 不定詞など）を指定する情報が語彙トランスファ過程で当該動詞表現相当概念に付与されると，その概念を構成する並列動詞概念全部にその表層形態情報が伝搬される．また，続く構文生成の前処理として，疑似概念依存構造をより表層構造に近い構造，より統語的な構造（疑似統語構造）へと変換を行う．例えば，受動表現における動詞の目的格は，受動態表現が可能な部分木の場合は，動詞表現の主語という統語的な関係へと書き換えられる．この統語構造を意識する段階で，疑似統語構造中のどの語（概念）を表層文中では代名詞として生成するか，省略するかについてもおおよそ決定される．最終的な決定は，続く構文生成の過程で個々の条件判断に基づき行われる．

続いて，上記トランスファ過程を経て出力される疑似統語構造から，構文生成規則を用いて表層の単語列を生成する構文生成処理が起動される．ここでは，語順の決定，省略語句の決定，疑似概念構造に表現されない語句の補完（冠詞，等位接続詞など），パンクチュエーションの決定，語の表層文における活用変化形（形態素生成時に参照する）の指定などが主な役割をなす．

最後に，構文生成処理の結果生成された単語列に対して形態素の生成を行い，表層文を生成する．

4. インプリメント

4.1 各句の重さ・構造情報の利用

各句の重さ（生成後の語列の長さ）や句の構造を生成過程で考慮することは，文の自然さ，読みやすさを高める上には欠かせない事項である．このための手段として，生成規則中の条件記述部に，任意の部分構造を構成する単語数参照，任意の部分構造におけるアーク存在チェック（構造参照）を行う記述を可能にした．以下，その規則記述の例を挙げながら，各句の重さ・構造を考慮した訳語の選択，統語構造の選択について説明する．

4.1.1 適切な表層表現の選択

一つの概念が，表層形態として構文的に異なる複数の表現候補を持つことがある．(11)と(11)′は動詞のセットフレーズの選択肢の例である．（既出の(4)，(4)′と同一）

(11) The research planning center provides this national network including some computers in Europe with funds.

(11)′ The research planning center provides funds for this national network including some computers in Europe.

(11)は“provide A with B”のパターンを用い，(11)′は“provide B for A”のパターンを用いている．(11)では，“A”の句が動詞句を含み長く複雑であるため，“with B”の句がヘッドの動詞“provide”から遠くなり“provide”との関係が弱まっている．これと対比的に(11)′は，“B”の句を“including…”のスコープから外しており，“provide”と“for”の関係も弱めることなく読みやすい文となっている．

これらは，統語構造を決定するフェーズの一般規則では処理できず，訳語を決定するフェーズにおいて個々の概念ごとに個別に処理されなくてはならない．そこで，辞書中の各項目に，入力である概念依存構造の任意の部分構造の重さ(表層構成単語数)を参照した語彙トランスファ規則の記述を行うことを可能にした．(12)は(11)に訳し分けるための語彙トランスファ規則の記述例である．ここでは“[provide]”は“provide”に相当する概念を表している．

(12) [provide]

(a) MP : 1(AGENT_2 OBJECT_3 RECIPIENT_4)
TP : 1(SUBJECT_2 OBJECT_3 PREP_5 (NP_4))
Conditions :
3. pw<5|
(4. pw>3. pw&!(4__REL. |…))
Actions : Node 5→ “for”

(b) MP : 1(AGENT_2 OBJECT_3 RECIPIENT_4)

TP : 1(SUBJECT_2 OBJECT_4 PREP_5 (NP_3))

Conditions : null

Actions : Node 5→ "with"

語彙トランスファ規則は tree-to-tree 変換の形態をとっている．各規則は，マッチングパターン（MP），ターゲットパターン（TP），条件部，アクションから構成される．MP，TP 中のノード"1"が辞書中のエントリー（ここでは"[provide]"）に対応する．(12 a) と (12 b) は互いに排反しており (12 a) が適用されたら (12 b) の適用にはいかない．

(12 a) の条件部中の"pw"は，概念依存構造中の指定された部分構造を構成する単語数を参照するための予約語である．前章で述べたように，語彙トランスファ規則は，概念依存構造中の依存関係の下位の概念から順に適用されるため，"[provide]" の規則適用時には，"3"，"4"に相当するノード（概念）以下の部分構造に対する訳語の選択が終了しているため，将来生成される表層単語数がカウントできる．意味解析を経て復元された痕跡ノードのカウントにあたっては，複数語からなる表層語句が与えられていても，後に代名詞化される可能性の高いものは'1'（例：節内の並列句の先頭句の格要素が痕跡ノードの場合），その他は'0'とカウントする．この段階では統語構造が正確に予測できないため，痕跡ノードのカウントはあくまで概算にとどまっている．

(12)によれば，軽い OBJECT を持つ概念依存構造に対しては (12 a) が適用され"provide B for A"のパターンを選択し，他の場合は (12 b) が適用され"provide A with B"のパターンを選択する．"4__REL."はノード"4"が（直下に限らず）下位構造として"REL"（関係節）を従えることを指定する．

4.1.2 各句の重さ・構造を考慮した語順操作

2章で述べたように，一般に，動詞句を従える関係節・前置詞句などの長い複雑な修飾句を下位構造に持つ重い句を文尾以外に置くと，修飾のスコープの問題が絡み，その後に位置する句とのシンタックス上の関係を曖昧にしがちである．また，"make" と "available", "set" と "aside" のように，複数の語が組み合わされて一つの意味をなすような語句の間に長い句が生成されると，語の意味を認識しながら文を読み進められないため，読みにくく不自然な文となる．

このような問題に対処し生成する句の順序を適切に制御するために，各句を構成する単語数・各句が従える構造を参照する機能を構文生成規則に導入した．(1)′，(5)′を生成しわけるためのルールの記述例を (13)-(14) に示す．

(13) $VP → v,
ADV{pw=1}$_1$
[*_OBJ{pw>5|
__REL.|
__NPP{pw>4}.|
…}.],
OBJ,
ADV_2, …

(14) $VP → v,
OBJ{(pw<6|
!(__REL.|__NPP{pw>4}.|…)
&…}$_1$,
COMP{pw<3}$_1$,
OBJ_2,
$COMP_2$, …

本構文生成規則は書き換え規則を変形・応用したものである．上記2ルールの構成と機能を簡単に説明する．

右辺の各項目（(13)では v，ADV，OBJ，(14)では v，COMP，OBJ）の記述順が各部分構造の生成順を表している．指定の部分構造が満たすべき属性条件は"{…}"の中に記述される．"[…]" には生成しようとしている句に関する条件以外の諸条件が記述される．すなわち，共起する句およびその下位・上位の特定構造の存在の参照やその句の属性の参照のための記述を行う．"*"は動詞句 $VP のヘッドノード，"_" は基準ノードの直下の構造の参照，"--" は直下に限らない下位構造の参照をする．これらの条件記述を伴う (13) 中の"ADV{…}$_1$ […]"は，"[…]" 内の条件を満たすとき"{…}"の属性を満たすような"ADV"の生成を行うことを意味する．ここで生成された副詞句が"ADV_2" によって重複して生成されることはない．"ADV_2" では"ADV{…}$_1$[…]"で条件的に排除された句のみが生成対象とされる．(14) においても"OBJ", "COMP" が二つずつあるが，同様に "OBJ{…}$_1$", "COMP{…}$_1$" の属性条件を満たさないもののみが"OBJ_2" "$COMP_2$" により生成される．

各条件記述部中の"pw"は，前記語彙トランスファ規則 (12 a) の"pw"と同様，疑似統語構造中の指定された部分構造を構成する単語数を参照するための予約語である．構文生成段階における"pw"は，前構造

トランスファ過程までに代名詞化されると予測されたノードについては‘1’，生成を省略されると予測されたノードについては‘0’とカウントする．前にもふれたように，厳密な判定は構文生成で順次行われるため，このカウントも概算の意味合いは排除されていない．

(13)によれば，単語数6以上の目的語（OBJ）あるいは関係節（REL）か単語数5以上の前置詞句（NPP）他を従える目的語が存在する時，単語数1の副詞句（ADV）は目的語の前に生成される．また(14)によれば，単語数2以下の目的格補語（COMP）は目的語の前に生成される．

同様に，句の重さ・構造の情報を用いれば，外置変形なども選択できる．(15)はその例である．

(15)′は文法的には可能だが，主語が重くバランスが悪い．この類いの統語的な選択は実用レベルでは非常に有効である．

(15) It is necessary to select the appropriate external device to be connected with the PRT/FDD connecter.

(15)′ Selecting the appropriate external device to be connected with the PRT/FDD connecter is necessary.

本稿では主に英文を生成する場合を例にあげているが，このような機能は日本語を生成する上でも有効である．日本語では，長く複雑な句は他の句に先行する傾向があり，文頭の位置に生起しやすい．これは，主語や他の句と述語の間に構造の複雑な句が置かれると，同様に修飾のスコープが絡んで，複雑な句に先行する句の統語上の係り関係が分かりにくくなり，かつ文のバランス面から見てもぎこちなくなるのを回避するためである．(16)はその例である．同じ格の要素を持つ(17)では「…について」の句が重くないため文中の位置に置かれても不自然でない．

(16) 英語 MS-DOS と併用する時に日本語 MS-DOS 専用のパーティションを作成するための指定についてユーザに図を用いて説明を行う．

(17) ユーザに指定内容について図を用いて説明を行う．

(16)の語順調整も上で述べたのと同類の条件記述を用いることにより可能である．日本語生成においては，単語数，文節数，文字数などをキーにすることが候補に挙げられる．しかし，合成語，補助用言などを見るとわかるように，どこまでの範囲を「単語」と捉えれば，句の重さを適切に表現できるかという非常に難しい問題がある．例えば，「作業机」は1語か2語か，「走ってくる」は1語か2語か，などの基準作りである．文節数については，これらの配慮も欠く分，さらに句の重さ反映の信頼度合が低下する．そこで，日本語生成規則では，“pw”として表層文中の語列の長さである文字数をあてるのが処理も容易な上に効果も期待できる．

4.1.3 各句の重さ・構造を考慮したパンクチュエーション操作

各句の修飾のスコープを明確にし，読みやすい文章を得るためのもう一つの重要な手段として，パンクチュエーション操作，特にコンマの挿入がある．(1)のように，副詞句が複雑でなく簡単な構造であれば目的語より前に生成することが可能であるが，(2)のように副詞句がある程度の構造を持っていると可能ではない．また，(2)′のように文頭に移動させるのも構文的に可能なケースは制限される．例えば，“print…”に並列句が続く場合は，文頭に生成された“using…”は“print…”だけでなく後続する動詞句をも修飾するように読めてしまう．そこで，コンマを有効的に挿入することが要求されてくる．

(18)は(2)″を生成しわけるための構文生成規則の記述例である．

```
(18)  $VP → v, OBJ,
            "," [*_OBJ{(pos=v|
                       __REL.|
                       __NP{pos=v}.|
                       …)}.&
                 *_%ADV{pos=v}.],
            ADV{pos=v};
```

(18)によれば，関係節・動名詞句（NP{pos=v}）などの動詞句を従える目的語と動詞句からなる副詞句（ADV{pos=v}）の間にコンマが挿入される．“*_%ADV.”は未生成の“ADV”が直下に存在することを指定する．

一方，(3)と(3)′のように，被修飾句の構造によって，積極的にコンマの挿入を回避する規則は次のように記述できる．“#embeded”は条件判定値レジスタ“embeded”に納められた真偽値を参照する．ここでは埋め込み節かどうかの判定に用いている．指定条件の判定および真偽値のレジスタへの登録は構文生成規則の任意の箇所で自由に記述できる．例えば“embeded”

は埋め込み節を処理するサブルーチン規則の右辺で真の値を設定する．

(19) $S → SUBJ, $VP,
",", [*_%ADVCLS. &!#embedded],
ADVCLS;

他の適用例として，並列関係の曖昧性除去があげられる．どの句とどの句が並列されているかの解釈が複数ありえる文でも，コンマを挿入することによってどちらの解釈が正しいかを明示することができる．(20), (20)′, (20)″ はコンマの有無を除けば同じ構成要素からなる文である．コンマのない(20)においては並列句の捉え方に曖昧性がある．"The sorter receives…" と "the user guide messages are…" が並列であることを示したいのなら (20)′ のようにコンマを挿入すれば明確になる．一方，"the second copy is…" と "the user guide messages are…" が並列であるなら(20)″ のように生成するのが望まれる．

(20) The sorter receives a signal after the second copy is made and the user guide messages are displayed.

(20)′ The sorter receives a signal after the second copy is made, and the user guide messages are displayed.

(20)″ The sorter receives a signal, after the second copy is made and the user guide messages are displayed.

これらに対する規則の具体的な記述例は省略するが，"after" の前のコンマの制御には(19)に示した規則の条件部に，ADVCLS の従える節中の並列句の有無，および非修飾句中の並列句の有無を参照した制約を追加することになる．"and" の前のコンマの有無の制御については，同様に節の並列表現のための等位接続詞を生成するルーチンで，並列されている句の一方に動詞節を従える後置修飾節の有無を参照し，曖昧性を回避するようコンマの生成の是非を制御する．

4.2 修飾句の生成位置に応じた語順制御

生成過程の前の段階では，概念依存構造中の各部分構造が将来どんな種類の句としてどんな生起位置制御を受けるべきか，十分予測されない．そこで，各修飾句の生成位置（前置修飾か後置修飾かなど）を決定しながらそれに応じて修飾語同士の語順を制御するために，構文生成規則中の任意の箇所で原文の語順に基づいて生成順序を制御できる，メタ制御メカニズムを導入した．これは，一般に被修飾句から離れた修飾句ほど修飾のスコープが広いという原理を利用して，原文における各句のスコープを最大限保持しようとするものであり，構文生成のフェーズでダイナミックに生成順序を制御するには欠かせない機能である．

次に挙げるのは，日英翻訳で，原文の日本語の語順により文頭に生成する副詞句と文尾に生成する副詞句の語順を制御するための，規則の記述例である．一文中で，前置修飾するものについては原文中の語順の先のものから後のものへ，後置修飾するものについては後のものから先のものへと，順に生成することを意図している．

(21) $S=$PREADV, SUBJ, $VP, $POSTADV,
$PREADV=ascend,
ADV{前置修飾条件},
ADV{前置修飾条件},
ADV{前置修飾条件};
$POSTADV=descend,
ADV, ADV, ADV;

"ascend" の記述により，以下の処理は，同じアーク名を持つ部分構造については原文中の語順の先のものから後のものへと生成を進める．逆に "descend" の記述により，語順の後のものから先のものへと生成処理を行う．この枠組を用いた結果，例えば次の(22)は，各副詞句が前置修飾条件に適合すれば(23)のように生成され，適合しなければ(24)のように生成される．

(22) 気温の急騰により都心地域では電力の供給が不十分である．

(23) Owing to the sudden rise in temperature, in the midtown area, the supply of electric power is insufficient.

(24) The supply of electric power is insufficient in the midtown area, owing to the sudden rise in temperature.

概念構造上では，一つの動詞句を二つの副詞句が修飾している場合の副詞句同士のように，互いに並列する部分構造間の関係は表されず，差を持たない．この枠組によれば，これら要素に対する語順操作が容易である．

5. 評　　価

以上述べてきた，(a)語彙レベルの表層表現の選択，(b)語句の生成位置，(c)コンマ挿入の制御機能の実用上の効果を検証するために，本生成機能をインプリメントした日英機械翻訳システムと，インプリメ

ントなしのシステムの両者を用いて日本語計算機操作説明書の文章を翻訳し，訳文の異なる文に対して効果の有無をチェックした．検証を簡単にするため，(A)重さ・構造情報の利用に関する実験と，(B)修飾句の位置情報の利用に関する実験は別個に行った．効果の有無に関する判断の基準は以下のとおりである．

(A)-(a)，(A)-(b)

風斗 (1981)[20] の線条化変換における「能率」の概念に照らし合わせ，「能率」の向上のある場合，効果ありとする．

(A)-(c)

原文中の依存関係を照らし合わせ，訳文における依存関係の曖昧性が減少した場合，効果ありとする．

(B)-(b)

訳文中の各修飾句のスコープと原文中の対応する句とを照らし合わせ，保存度が高まった場合，効果ありとする．

上記判定により効果ありとされた件数を**表1**に挙げる．テスト対象の機能が効果として現れた文の割合は9.4% であった．表1中の「有意差なし」に計上した事例は，コンマの有無に関して曖昧性の増減が見られなかった事例である．上記の判定と同時に訳文の容認性に対する主観的評価も行ったところ，効果ありとされた文についてはすべて容認性の低下は見られていない．本実験文書は比較的一文の長さが短く効果の出にくい性質のものであったが，本機能の有効性を検証することができた．

表1 実験結果
Table 1 Results of the experiment.

有効原文数	331 文
一文平均文字数	63 字
≪改良件数≫	
(A) 重さ・構造情報の利用	
(a) 語彙レベルの表層表現の選択	1 件
(b) 語順制御	6 件
(c) コンマ制御	7 件
(B) 修飾句の位置情報の利用	
(b) 語順制御	17 件
≪悪化件数≫	
	0 件
≪有意差なし≫	
(A) 重さ・構造情報の利用	
(b) コンマ制御	2 件

6. 議　論

本稿では，特に機械翻訳における訳文の生成に焦点をあて，出力文の品質の改善について述べてきた．1章でもふれたように，翻訳においては，Hovy (1988)[21]，McKeown ら (1991)[22]，Inui ら (1992)[13] などが着目している「what to say」と「how to say」の間の相互依存性は考慮する余地がない．さらに，翻訳の忠実度の観点から，生成途中あるいは生成後に検出される問題を回避する手段の幅は狭い．すなわち，従来のテキスト生成・推敲の分野で取られているようなダイナミックな生成戦略のバックトラックの枠組み[9)~13)]が，そのまま訳文の改善に同様の効果を及ぼせるものではなく，翻訳という固有の視点に着目する必要がある．このようなことから，本稿で述べる生成方式は，生成途中で中間的構造を評価し，既に選択された規則の処理内容も適宜参照しながら，忠実度も考慮して可能な限り予想される問題を回避しようという戦略をとっている．

語順やパンクチュエーションの制御は，日本語のようにこれらの制約の弱い言語の文の推敲には効果が高い．Hayashi (1992)[14] は，日本文の統語上の曖昧性解消，語順と読点の調整を導入している．しかし，実際には，語順は統語上の曖昧性解消だけの目的で制御されるべきものではなく，新情報・旧情報，焦点など，考慮すべき要因が複数存在する[13]．Hayashi (1992)[14] の手法は純粋に統語上の曖昧性解消を目的としている．一方 Inui ら (1992)[13] の WEIVER は，これらの種々の要因を考慮するために必要であれば「what to say」のレベルまでバックトラックできる機構を持っている．翻訳においては，基本的に「what to say」に関する情報はすべて原文の中に含まれると考えることができるので，このレベルへのバックトラック能力は実質的に必要とならない．そこで，本稿で述べる生成方式の語順調整は，規則の中で，句の統語上の役割を意識し，原文中の語順を参照するなどして他の句との関係を意識しながらよりふさわしい語順の決定をすることを趣旨としている．

機械翻訳においては，これまで，いかに自然言語の持つ曖昧性をうまく解析するか，特異な文法現象をいかにうまく処理するかに関する研究に焦点があたり，解析結果をいかにうまく生成するかという問題については系統立てて議論される機会が少なかった．Somers ら (1988)[23] は，英日機械翻訳における複雑な

英語名詞句の denominalization について論じている．これは，日本語では体言よりも用言を用いた表現の方が好まれるという英語・日本語間の言語構造の差異に基づいたもので，精度の高い翻訳を得るための価値ある議論である．しかし，denominalization と訳語の割り当て・選択との関連については触れていない．実際には，英語構造の denominalization を初めとする統語構造の変換と共起すべき訳語の選択もある．denominalizationは訳語および統語構造の選択と非常に関係の深い問題であると言える．今後は．nominalization と denominalization を生成過程に導入し，訳語の決定にこれらの処理を活性化させ，より精密な選択制御ができるよう拡張を検討していく．これは，機械翻訳システムの利用者の立場から見れば，より広範囲な訳出表現の選択ができ，かつそれを知識として学習させることができることにつながる．

7. おわりに

以上，本稿では，訳文の理解容易性という観点からより良い生成候補を選択する必要のあることを示し，また，我々の生成機能でそれをどのように鑑みて処理しているかを示した．すなわち，生成過程への入力データである概念依存構造および中途段階で生成される疑似概念依存構造・疑似統語構造に関する重さ・構造の情報，原文中の語順情報を利用して，適切な表層表現の選択，語順・パンクチュエーションの制御という３種類の制御を行うアプローチについて述べた．

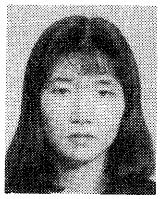
これらのアプローチの実現により，生成文中の各句相互の統語関係が明確になり，自然な文，理解容易な文の生成に近づくことができた．なお，本アプローチは，実用の機械翻訳システムに既にインプリメントされ実用面でも効果を見せている．

参考文献

1) Munster, E.: The Treatment of Scope and Negation, *Proceedings of COLING '88*, pp. 442-447 (1988).
2) Gailly, P. J.: Expressing Quantifier Scope in French Generation, *Proceedings of COLING '88*, pp. 182-184 (1988).
3) Kohl, D.: Generation from Under- and Over-specified Structures, *Proceedings of COLING '92*, pp. 686-692 (1992).
4) McDonald, D.: On the Place of Words in Generation Process, *Natural Language Generation in Artificial Intelligence and Computational Linguistics*, pp. 229-247, Kluwer Academic Publishers (1991).
5) Nirenburg, S.: A Framework for Lexical Selection in Natural Language Generation, *Proceedings of COLING '88*, pp. 471-475 (1988)
6) Ward, N.: Issues in Word Choice, *Proceedings of COLING '88*, pp. 726-731 (1988).
7) Fedder, L.: Generating Sentences from Different Perspectives, *Proceedings of the 28th Annual Meeting of ACL*, pp. 125-130 (1990).
8) Yang, G. et al.: From Functional Specification to Syntactic Structures: Systemic Grammar and Tree Adjoining Grammar, *Computational Intelligence*, Vol. 7, pp. 207-219 (1991).
9) Mann, W.C. and Moore, J. A.: Computer Generation of Multiparagraph English Text, *American Journal of Computational Linguistics*, Vol. 7, No. 1, pp. 17-29 (1981).
10) Gabriel, R. P.: Deliberate Writing, *Natural Language Generation Systems*, pp. 1-46, Springer-Verlag (1988).
11) Mann, W.C.: An Overview of the Penman Text Generation System, *Proceedings of National Conference on Artificial Intelligence*, pp. 261-265 (1983).
12) Meteer, M. M. and Moore, D. D.: A Model of Revision in Natural Language Generation, *Proceedings of the 24th Annual Meeting of ACL*, pp. 90-96 (1986).
13) Inui, K. et al.: Text Revision: A Model and Its Implementation, *Aspects of Automated Natural Language Generation*, pp. 215-230, Springer-Verlag (1992).
14) Hayashi, Y.: A Three-level Revision Model for Improving Japanese Bad-styled Expressions, *Proceedigs of COLING '92*, pp. 665-671 (1992).
15) Amano, S. et al.: The Toshiba Machine Translation System, *Future Generations Computer Systems*, Vol. 2, No. 2, pp. 121-123, North-Holland (1986).
16) Nogami, H. et al.: Parsing with Look-ahead in Real-time On-line Translation System, *Proceedings of COLING '88*, pp. 488-493 (1988).
17) Miike, S. et al.: Experiences with an On-line Translating Dialogue System, *Proceedings of the 26th Annual Meeting of ACL*, pp. 155-162 (1988).
18) Hirakawa, H. et al.: EJ/JE Machine Translation System ASTRANSAC—Extensions toward Personalization, *Proceedings of MT Summit III*, pp. 73-80 (1991).
19) Quirk, R. et al.: *A Comprehensive Grammar of the English Language*, Longman (1985).
20) 風斗博之：依存関係表示と線条化変換，言語の科学，第8号，pp. 57-84，東京言語研究所

(1981).
21) Hovy, E. H.: *Generating Natural Language under Pragmatic Constraints*, Lawrence Erlbaum Associates (1988).
22) McKeown, E. H. and Elhadad, M.: A Contrastive Evaluation of Functional Unification Grammar for Surface Language Generation: A Case Study in Choice of Connectives, *Natural Language Generation in Artificial Intelligence and Computational Linguistics*, pp. 352-396, Kluwer Academic Publishers (1991).
23) Somers, H. et al.: The Treatment of Complex English Nominalization in Machine Translation, *Computers and Translation 3*, pp. 3-21, Kluwer Academic Publishers (1988).
24) Brogan, J. A.: *Clear Technical Writing*, McGraw-Hill (1973).

(平成6年4月5日受付)
(平成6年11月17日採録)



吉村裕美子（正会員）

1962年生．1985年京都大学文学部文学科言語学専攻卒業．同年，(株)東芝入社．現在，同社研究・開発センター情報・通信システム研究所に所属し，機械翻訳を主とする自然言語処理システムの研究開発に従事．

平川 秀樹（正会員）

1956年生．1980年京都大学大学院工学研究科電気工学専攻修士課程修了．同年東京芝浦電気(株)（現，(株)東芝）入社．以来．機械翻訳，談話解析などの自然言語処理システムの研究開発に従事．1982-1985年(財)新世代コンピュータ技術開発機構研究員．1993年より(株)日本電子化辞書研究所第五研究室室長．現在(株)東芝研究・開発センター情報・通信システム研究所主任研究員．人工知能学会，言語処理学会，ACL 各会員．