

英日対話文翻訳における漸進的な日本語文生成法

赤峯 享 古瀬 蔵

A T R 音声翻訳通信研究所

1 はじめに

従来、言語翻訳や対話システム等で文を生成する場合、文単位で処理を行うことがほとんどであった。しかしながら、自動通訳システムのように入力の時系列に与えられる場合、文単位の処理では、発話が終了するまで処理を開始することができず、発話の伝達に遅延が生じてしまう。その結果、1) 発話間の結束性が壊れてしまう、2) 処理時間が増加してしまう等の問題が生じる。

ユーザーが自動通訳システムを利用する理由は、通常の対話を行うのと同程度の負担で、自分の目的を達成することが可能になるからである。したがって、本来の目的以外の大きな負担をユーザに与えてしまう上記の問題を解決することは、実用的な自動通訳システムの構築にとって非常に重要である。本研究の目的は、文単位で翻訳を行った場合と同等の了解性のある文を、短時間で出力することである。

本稿では、「ホテルを予約する際の対話」のように目的が明確な対話における英語から日本語への翻訳を例に、同時通訳的な処理を行った際の漸進的な文生成方式を示す。すなわち、接続詞のような区切りとなる表現をトリガーとして、入力された発話を句や節といった単位で英日変換し、その部分的な変換結果の日本語表現を順次確定していきながら、文全体として意味の通る自然な日本語文を生成する方式を提案する。

2 漸進的な処理

自動通訳における、文単位の処理と漸進的な処理との処理時間を図1に示す。発話の結束性を保ち、処理時間を短縮させるためには、出力開始までの時間を短縮し、かつ、出力終了までの時間を短縮する必要がある。図1に示すように、文単位の処理では、「出力開始までの時間 = 発話時間 + 翻訳処理時間」である。また、現状の多くの翻訳システムでは、処理時間は高速になって

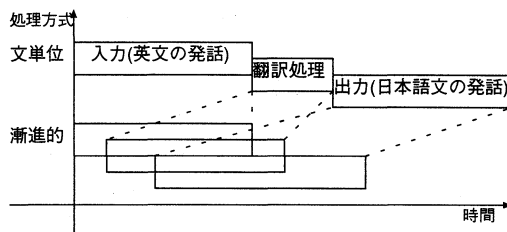


図1: 処理時間

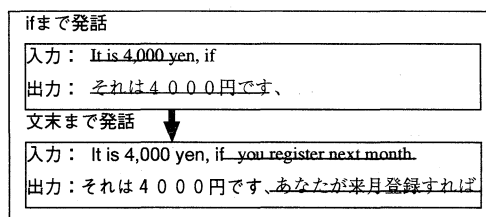


図2: 漸進的な翻訳の例

きており、「発話時間 >> 翻訳処理時間」が成り立つ。したがって、翻訳処理時間を短縮することは、出力を開始するまでの時間を短縮させることにあまり寄与しない。一方、漸進的な処理を行えば、入力の発話が終了する前に、翻訳処理や出力を行うことで、出力開始までの時間、及び、出力終了までの時間を大幅に短縮することが可能である。

例えば、「It is 4000yen, if you register next month.」という文を翻訳する場合、図2に示すように、「if」が発話された段階で、「それは4000円です」という翻訳を出力することで、発話開始までの時間を短縮させることが可能である。また、最終的な翻訳結果「それは4000円です、あなたが来月登録すれば」を出力するまでの時間も短縮できる。

しかしながら、日本語と英語のように、語順や構造が大きく異なる言語では、単純に順送りに翻訳結果を出力するのでは、意味の通らない文を生成する可能性が生じる。また、自動通訳のように最終的な出力を音声とした場合、一旦出力したものを消去することができないため、出力した日本語表現を自己修復する機構が生成部に必要となる。

3 日本語対話文の特徴

英文から漸進的に日本語対話文を生成するためには、従来の書き言葉の生成が対象としてきた「文法的に適切な文」を生成するだけの能力では不十分である。書き言葉では「不適格文」とされるが、日本語の対話では、頻繁に用いられるものとして、「繰り返し」、「倒置」等がある [1, 2, 3]。これらを使うことで生成能力を高めることを考える。以下に、「繰り返し」と「倒置」について、簡単に説明する。

3.1 繰り返し

人間の発話では、繰り返しを用いることで、すでに言った語句に新たな情報を追加したり、不的確な表現を訂正することがある [2]。例えば、「電話番号を、あなたに連絡がとれる電話番号を教えてください」というように、「電話番号」を繰り返すことで、すでに言った語句（「電話番号」）に対して新たな情報（「あなたに連絡がとれる」）を追加することができる。

3.2 倒置

日本語文では、「節(文)の主動詞は、文末に表れなければならない」と言われる。しかしながら、実際の対話では、図2の翻訳結果のような語順が倒置したものが頻繁に見られる。ここでは、この現象を前記の繰り返しと同様の枠組で扱うために、語順が倒置したのではなく、文の繰り返しと省略によって、生じたものであると考える [4]。例えば、「それは4000円です、あなたが来月登録すれば」という文は、まず、「それは4000円です」が条件を省略した文として生成され、次に、この文の条件をはっきりさせるために、条件を文末で繰り返すことによってこの文が生成されたと考える。

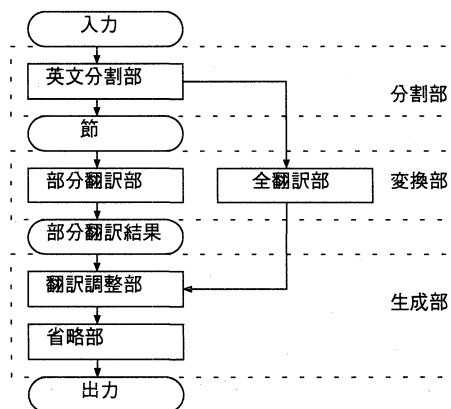


図3: 翻訳処理

4 生成方針

上記の日本語対話文の特徴を生成に取り入れ、かつ、生成文の了解性の低下を避けるために、以下のような方針で生成を行う。

- 文の了解性の確保を確保するために、係り受けの語順は、必ず係り元から係り先にする。
- 係り受けの語順が逆転する場合は、その部分を繰り返すことで文を自己修復する。
- 出力時間を短縮するために、繰り返しにおける冗長な表現を省略する。

5 翻訳方式

前節の生成方針を取り入れた翻訳処理の概略を図3に示す。英文分割部は、入力文を適切な単位に分割する。部分翻訳部は、分割した部分を順次翻訳し、部分的な翻訳結果を得る。全翻訳部は、入力文の最大の構造に対して、翻訳を行う。翻訳調整部は、全翻訳部より送られた翻訳結果と、これまでの翻訳結果を比較し、必要ならば繰り返しを行うことで、適切な語順で訳文を出力する。省略部は、繰り返しにおける冗長な表現を省略する。

以下で、今回、評価用に用いた具体例を用いて、各々の処理部の説明を行う。

表 1: 英文の節のパターン

節の種類	英文パターン	日本語パターン	レベル
文頭	By the way / X	ところで / X'	1
	no / X	いえ / X'	1
並列節	X / but Y	X' / しかし Y'	2
従属節	X / if Y	Y' ならば / X'	3
	if X / [CN-PRON] Y	Y' ならば / X'	3
引用節	X think [V-PRON] / Y	X' は Y' と 思う	4
関係節	X / where Y	Y' [連体] / X'	5

5.1 英文分割部

今回、英文を分割する単位を 1) 意味的なまとまりがあり、英語と日本語の間で対応が取れやすい、2) 英文に分割のための比較的明確なマーカー(接続詞等)が現れる、という理由から、節とした。

また、英文を節に分割するためのパターンとして、用例ベースの翻訳システムで使用しているパターン[5]を利用した。代表的な英文のパターンを表 1 に示す。この英文パターンは、表層表現(“but”、“if”等)及び品詞列の並び([CN-PRON]等)をキーとして、文を節に分割する。なお、表中のレベルの数字は分割の強さを、X と Y は英語表現を、X' と Y' は対応する日本語の訳を示す。

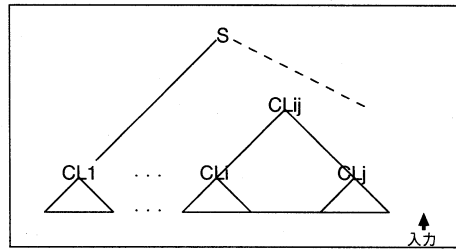


図 4: 節の処理

以下の処理を発話の終了まで繰り返す。

変換 1 ・部分翻訳結果 $trans(CL)$ を求める。

変換 2 ・全翻訳結果 $trans(CL_i, \dots, CL_j)$ を求める

生成 1 $trans(CL_j)$ が $trans(CL_i, \dots, CL_j)$ と等しい場合 [$i = j$ の場合]

- $trans(CL_i, \dots, CL_j)$ を翻訳結果として出力する。

生成 2 $trans(CL_i, \dots, CL_{j-1}) trans(CL_j)$ が $trans(CL_i, \dots, CL_j)$ と等しい場合 [語順が等しい場合]

- $trans(CL_j)$ を翻訳結果として出力する。

生成 3 $trans(CL_j) trans(CL_i, \dots, CL_{j-1})$ が $trans(CL_i, \dots, CL_j)$ と等しい場合 [語順が逆の場合]

- $trans(CL_i, \dots, CL_j)$ を翻訳結果とし、繰り返しの共通部分(冗長な表現)を省略して出力する。

生成 4 [その他の場合]

- $trans(CL_i, \dots, CL_j)$ を翻訳結果として出力する。

5.2 変換部

翻訳部は、対訳用例を利用した対話文翻訳システム[5, 6, 7]を利用した。このシステムでは、(「I would like to ~」→「～たいのですが」)、(「~ for ...」→「... に対する ~」)のような比較的細かな単位で用例を用意し、それを組み合わせることで1文の変換を行うことを可能としている。

5.3 翻訳・生成方式

入力文 S が $CL_1, \dots, CL_i, \dots, CL_j (1 \leq i \leq j)$ の節に分割された場合、以下の方式で文を変換・生成する。ただし、 CL_j は、現時点の入力での右端の節であり、 CL_i は、 CL_j を含む最大の節の開始の節である(図 4 に略図を示す)。また、英語の節 CL の翻訳結果を $trans(CL)$ とする。

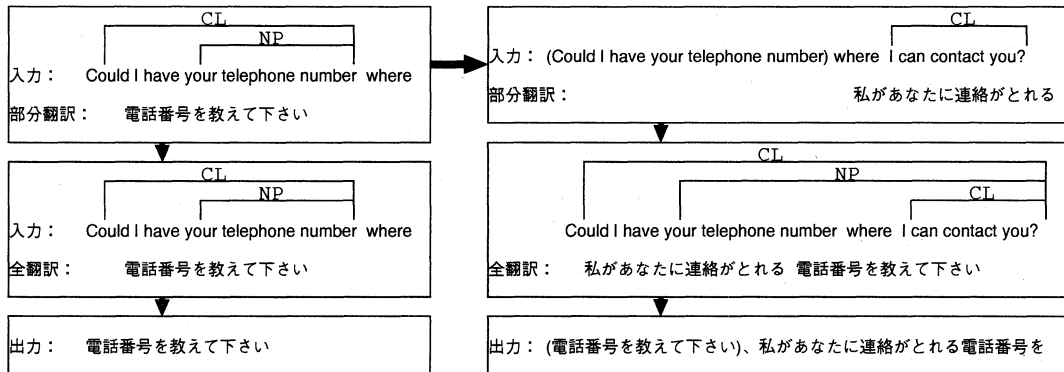


図 5: 関係節の翻訳過程

5.4 翻訳処理例

「Could I have your telephone number where I can contact you?」という文を例にとり、翻訳処理の流れを示す。

まず、「Could I have your telephone number where」と入力された段階で、「X where Y」という節のパターンにマッチし、英文分割部により節に分割される。部分翻訳および全翻訳が行われ(図5の左部分)、部分翻訳結果と全翻訳結果が等しいので、全翻訳結果の「電話番号を教えてください」が出力される。

次に、文末まで入力されると、部分翻訳部が「I can contact you」の部分翻訳し、全翻訳部がその節を含む最大の節(つまり、全文)を翻訳する(図5の右部分)。語順が逆になっているため、全翻訳結果を翻訳結果とし、共通部分を省略することで、「私があなに連絡がとれる電話番号を」が出力される。

6 おわりに

英日翻訳を例に、同時通訳的な処理を行った際の漸進的な文生成方式を示した。この方式を用いることで、翻訳結果の出力開始までの時間を大幅に短縮することができることを示した。

今回、評価を行ったのは、容易に節に分解できる英文の代表的な構文のみである。今後は、1) 英文を分割するための適切な単位(パターン)の吟味、2) 左から順に分割する方式では正しく分割することが困難なパターン

(“and”で結ばれる並列等)の解消の検討などを行う予定である。

参考文献

- [1] 竹沢寿幸、田代敏久、森元暉: “音声言語データベースを用いた自然発話の言語現象の調査”, 人工知能学会 SIG-SLUD-94-3-3 (1995).
- [2] 佐川雄二、大西昇、杉江昇: “大規模コーパスに基づいた日本語自己修復文の分析”, 情報処理学会 自然言語処理研究会 100-10 (1994).
- [3] 山本幹雄、小林聡、中川聖一: “音声対話文における助詞落ち・倒置の分析と解析手法”, 情報処理学会論文誌 Vol.33 No.11 (1992).
- [4] 久野すすむ: “談話の文法”, 大修館書店(1978).
- [5] Osamu FURUSE and Hitoshi IIDA: “Constituent Boundary Parsing for Example-Based Machine Translation”, COLING 94 (1994).
- [6] 古瀬蔵、隅田英一郎、飯田仁: “経験的知識を活用する変換主導型機械翻訳”, 情報処理学会論文誌(1994).
- [7] Susumu AKAMINE, Osamu FURUSE and Hitoshi IIDA: “Integration of Example-Based Transfer and Rule-Based Generation”, ANLP 94 (1994).