

# 頑健な多言語音声翻訳のための不適格入力分割処理

古瀬 蔵<sup>†, 1</sup> 山田 節夫<sup>†, 2</sup> 山本 和英<sup>†, 3</sup>

複数の文から成る長い発話、文法規範から逸脱した表現などを包含する話し言葉の翻訳入力に対して分割を行い頑健な多言語音声翻訳を実現する手法を提案する。提案手法は、構文構造のスコアを使って全体的に最も整合性のとれた部分へ翻訳入力を分割する。そして、分割されたそれぞれの部分の翻訳結果を連結させることにより、翻訳入力全体の翻訳結果を作る。分割は、逐次的構文解析の枠組みの中で行うため、翻訳時間に重大な影響を与えることはない。提案手法には、翻訳出力率や翻訳成功率など翻訳性能を向上させることに加えて、複数の文から成る発話や誤った音声認識結果などの翻訳入力に対しても頑健な翻訳を行う、という効果がある。また、特定の言語に依存しない手法のため、多言語処理に有効である。提案手法を導入した変換主導型機械翻訳システムの翻訳実験から、提案手法が多言語音声翻訳の頑健性向上に有効であることを実証した。

## Splitting Ill-formed Input for Robust Multi-lingual Speech Translation

OSAMU FURUSE,<sup>†, 1</sup> SETSUO YAMADA<sup>†, 2</sup>  
and KAZUHIDE YAMAMOTO<sup>†, 3</sup>

This paper proposes a method for robust speech translation by splitting spoken-language input which may include many ill-formed expressions. The proposed method splits input into well-balanced translation units based on a score for a substructure. The complete translation result is formed by concatenating the partial translation results of each split unit. The splitting is performed during left-to-right parsing, and does not degrade translation efficiency. The proposed method gives the following advantages: elimination of null outputs, improvement of the understandability of the output, splitting of utterances into sentences, robust translation of erroneous speech recognition results, and a universal framework effective for multi-lingual translation. Experimental results of the Transfer-Driven Machine Translation system's performance have shown that the proposed method is effective in multi-lingual speech translation.

### 1. はじめに

話し言葉を対象とする機械翻訳システムの入力は文法的に整った1文で発話されるとは限らない。そのため、システムの持つ文法規則では解析できずにシステムが不適格<sup>4</sup>と判断する入力が増える恐れがある。とりわけ、音声翻訳システムでは、音声認識結果が機械翻訳の入力となるので、システムに入力された発話が文法的に整っていたとしても、音声認識の誤りが原

因で機械翻訳の入力に不適格表現が送られることもある。さらに、書き言葉の翻訳で構文解析の有効な手掛かりとなる句読点が、機械翻訳の入力となる音声認識結果で記述されるとは保証されない<sup>5</sup>。また、同時通訳のような音声翻訳システムを目指す場合は複数の文などが長々と連続する入力にも対応する必要が出てくる。このように、音声翻訳システムには、話し言葉の多様な翻訳入力を扱うことができる頑健性(robustness)が要求される。

頑健な音声翻訳の実現のため、誤った音声認識結果のうち信頼性が高いと思われる部分だけ機械翻訳結果を出力する手法<sup>14)</sup>や、音声認識された発話を言語処理しやすい単位へ変換しようとする手法<sup>13)</sup>が提案されている。しかし、前者の手法は音声認識誤り以外の不適

<sup>†</sup> ATR 音声翻訳通信研究所

ATR Interpreting Telecommunications Research Laboratories

<sup>1</sup> 現在、NTT サイバースリユーション研究所

Presently with NTT Cyber Solutions Laboratories

<sup>2</sup> 現在、NTT コミュニケーション科学基礎研究所

Presently with NTT Communication Science Laboratories

<sup>3</sup> 現在、ATR 音声言語通信研究所

Presently with ATR Spoken Language Translation Research Laboratories

<sup>4</sup> 本論文では、システムの持つ文法規則で解析が可能な場合を適格(well-formed)、そうでない場合を不適格(ill-formed)と定義する。

<sup>5</sup> 本論文で示す翻訳入力に句読点は含まない。

格表現，複数文から成る長い発話なども含めた話し言葉全般を対象とはしていない．後者の手法は日本語のみを対象としており，多言語翻訳に有効とはいえない．

本論文は，翻訳入力を分割することにより文法的な逸脱表現，複数文より成る長い発話，などの話し言葉全般の翻訳入力について，統一的な枠組みで構文解析と翻訳を可能にする手法を提案する．提案手法は，特定の言語に依存しない手法であり多言語翻訳に適用できる．また，構文解析の枠組みを大きく変更したり，複雑な，あるいは言語依存的な後処理を用意したりせずに分割処理を導入できるので，翻訳時間に重大な影響を与えることはない．

構文構造が複雑な長文を対象として，分割により構文解析を成功させようとする手法が提案されている<sup>7)~9),11)</sup>．しかし，これらの手法は，人手で作られた多くの分割規則を用いたり，並列句など特定の表現や日本語のみを対象としたりする．そのため，日本語以外の原言語，文法的な逸脱表現を含む幅広い口語表現なども対象とする多言語音声翻訳に有効とはいえない．

筆者らは，日英双方向や日韓双方向の多言語翻訳を実現している変換主導型機械翻訳 (Transfer-Driven Machine Translation, 以下, TDMT と呼ぶ<sup>3),6)</sup> に提案手法を組み込み，TDMT システムを用いた翻訳実験により提案手法が多言語音声翻訳の頑健性向上に有効であることを確認した．

以下，2章で本論文で提案する分割処理の枠組みについて，3章で TDMT の翻訳処理の概要について，4章で TDMT への分割処理の導入について，それぞれ日英と英日翻訳を使って説明する．そして，5章で，TDMT システムの性能について分割処理の有無で比較した評価実験結果に基づき，提案手法が多言語音声翻訳の頑健性向上に有効であることを示す．

なお，本論文は，文献 5) の内容を検討し，新たにまとめたものである．

## 2. 分割処理の枠組み

文法的な逸脱表現，および句読点なしで連続する複数の文は，機械翻訳システムにとって，蓄積された文法規則だけでは扱えない未知の入力である可能性がある．そのため，図 1 (a) のように入力全体の構文構造を作れずに，システムが翻訳結果を出力できない場合もある．このような場合，図 1 (b) のように構文解析および翻訳が可能な部分に入力を分割して，構文解析を成功させる．そして，分割されたそれぞれの部分の翻訳結果を連結させて入力全体の翻訳結果とする．

入力に不適格表現を含むことが原因で，構文解析と

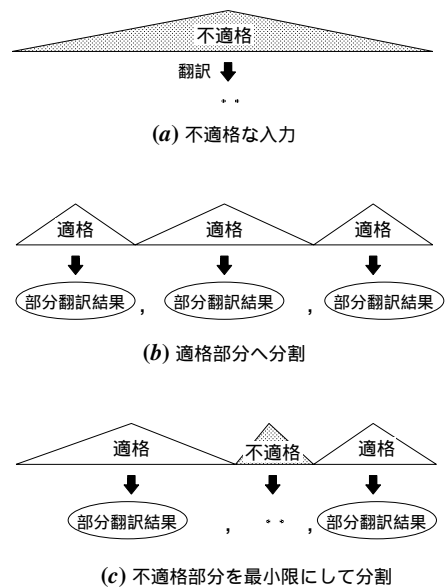


図 1 分割についての考え方  
Fig. 1 Idea of splitting.

翻訳が可能な部分だけへ分割することができない場合は，図 1 (c) のように，システムにとって既知の構文構造である部分によって入力ができるだけ広範囲にカバーされるよう入力の分割を行う．

### 2.1 分割の戦略

本論文で提案する分割処理は，以下の戦略 (A)~(F) により規定される．

- (A) 隣接する構文構造を連結させることにより，さらに大きな構文構造を作る．
- (B) 構文構造を連結させることで作られた構文構造は，右に隣接する構文構造と連結して，さらに大きな構文構造を作る．
- (C) 内容語だけでなく，機能語や未知語も 1 語単位で構文構造を構成できる．
- (D) 構文構造の曖昧性を解消する場合は，分割を用いた構文構造の優先度を低くする．
- (E) 全体の翻訳結果は，分割された部分ごとに翻訳された結果を並べることにより形成される．
- (F) 目的言語表現を持たない語は，理解不能な部分であることを示す“..”のような文字列に変換する

本論文では，構成要素であるすべての葉を具体的な語句により充足しているものを構文構造と呼ぶ．チャート法では不活性弧が構文構造に対応する．

本論文の図においては，システムの持つ規則では構文構造および翻訳結果を作れない部分に対して，背景パターンに陰影をつける．また，翻訳結果における分割の箇所を分かりやすくするために，分割された部分の翻訳結果の間に読点“.”を挿入する．

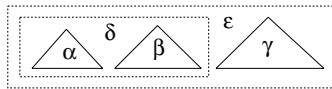


図2 分割された部分構文構造から成る構文構造  
Fig. 2 Structure from split substructures.

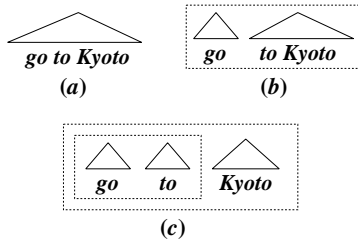


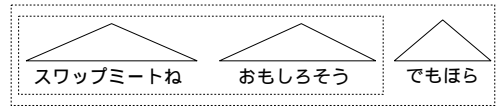
図3 “go to Kyoto” の構文構造  
Fig. 3 Structures for “go to Kyoto”.

戦略 (A) と (B) により、隣接する構文構造を再帰的に連結させて、新たな構文構造を次々に作っていくことができる。すなわち、入力をいくつかの部分に分割し、それぞれの部分に構文構造が作れるならそれらを連結して全体の構文構造を作ることができる。図2は、戦略 (A) と (B) に基づく構文構造の構成を示す。 $\alpha, \beta, \gamma$  は隣接する構文構造である。三角形は、蓄積された文法規則のみから構成された構文構造を表す。点線の四角形は、隣接する構文構造を連結することにより作られた構文構造を表す。 $\delta$  は、隣接する  $\alpha$  と  $\beta$  の連結により構成される。さらに、 $\epsilon$  は、隣接する  $\delta$  と  $\gamma$  の連結により構成される。

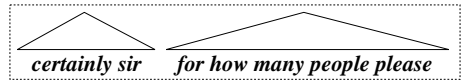
戦略 (C) は、不適格な表現部分でも1語単位なら構文構造を構成することを認める。戦略 (C) を加えることにより、不適格な表現部分が入力に含まれていても、入力を適格な表現部分と不適格な表現部分に分割し、適格な表現部分の構文構造と不適格な表現部分の構文構造とを連結して全体の構文構造を作ることが可能になる。

戦略 (D) は、入力を分割して構成された構文構造は、最尤の構文構造としては選ばれにくくなるようにする。分割の戦略は構文解析の頑健性を向上させるが、断片的な表現を並べた構造より、広域的な構造のほうが、構文的あるいは意味的な依存関係を多く含むので、より良い翻訳結果が得られると考えられる。それゆえ、分割に頼りすぎる構文解析は避けて、分割は補完機能として位置づける。

図3は、“go to Kyoto” に対する3つの可能な構文構造を示す。(a)では、分割を行っていない。(b)では、“go” と “to Kyoto” の2つに分割、(c)では、“go”、“to”、“Kyoto” の3つに分割、を行っている。この中



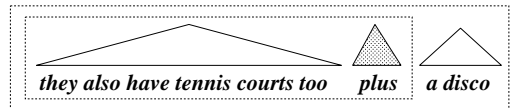
the swap meet , sounds interesting , but look



はい , 何人ですか

図4 例(1)と(2)に対する翻訳

Fig. 4 Translations of (1) and (2).



同様に テニスコートがまたあります , .. , ディスコ

図5 例(3)に対する英日翻訳

Fig. 5 English-to-Japanese translation of (3).

では、分割なしの (a) を最尤の構造に選ぶべきである。

戦略 (E) と (F) は、入力の分割により構成された構文構造からの翻訳出力を規定する。

### 2.2 分割を用いた翻訳の例

前節の戦略 (A)~(F) に基づいた分割によって、例 (1) や (2) のような口語表現にも、図4に示すように、構文構造と日英、英日それぞれの翻訳結果を与えることができる。

- (1) 「スワップミートねおもしろそうでもほら」
- (2) “certainly sir for how many people please”

また、例 (3) を入力とする英日翻訳において、“they also have tennis courts too” と “a disco” を “plus” で連結して構文構造を作る規則がシステムにまだないと仮定する。

- (3) “they also have tennis courts too plus a disco”

このような場合にも、図5のように翻訳結果を得ることができる。

### 2.3 音声認識誤りへの対応

音声入力した発話に対して音声認識は誤った結果を与える場合がある。誤認識部分はしばしば不適格な翻訳入力を与える。分割処理は、このような誤認識の翻訳入力に対しても何らかの翻訳結果を出力し、その翻訳結果から発話の意味内容を伝える割合も向上させることができる。

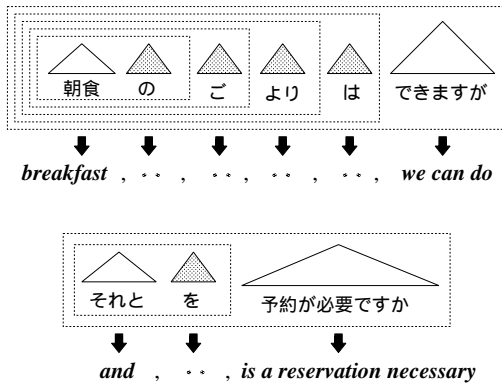


図6 例(4')と(5')に対する日英翻訳

Fig. 6 Japanese-to-English translations of (4') and (5').

音声入力された日本語発話(4)と(5)の誤認識結果の例が、それぞれ、(4')と(5')である。

- (4) 「朝食のご用意はできますが」  
 (4') 「朝食のごよりはできますが」  
 (5) 「それと予約が必要ですか」  
 (5') 「それとを予約が必要ですか」

(4)の認識では「用意」が「より」となる置換誤りが、(5)の認識では「を」の挿入誤りが起きている。分割を用いた(4')と(5')の日英翻訳を図6に示す。翻訳結果は、品質に多少問題があるものの、音声入力された発話の意味を伝えることができる。

このように、提案手法は、複数文より成る長い発話、文法規範から逸脱した口語表現などの処理と同じ枠組みで、音声認識誤りについても対応することができる。

音声認識誤りを含む表現を対象として、意味距離や連続形態素数の閾値に基づいて信頼性の高い部分だけを抽出して部分的に翻訳結果を生成する手法も提案されている<sup>14)</sup>。しかし、この手法では、信頼性の高い部分が見つからないために翻訳結果を出力しない場合がある。一方、本論文の提案手法は、翻訳入力全体での整合性を維持してできるだけ多くの部分を翻訳出力させようとする。音声認識の信頼性が低い場合は文献14)の手法のほうが有効であるが、信頼性が高い場合は本論文の提案手法のほうが有効と考えられる。

## 2.4 分割の前提条件

2.1節で述べた分割の戦略を実現するためには、以下の前提条件が必要である。

- (i) 最尤の構文構造を決定するためのスコアを、構文構造に対して定義できる。
- (ii) システムの持つ文法規則で構成された構文構造に対しては、変換および生成を行うため

の規則が用意されている。

(i)の条件は、構文構造のスコアを元にして最適な分割箇所を決めるため、(ii)の条件は、分割されたそれぞれの部分から翻訳結果を得るためである。3章で述べるTDMTはこれらの前提条件を満たしている。

## 3. TDMTの概要

TDMTは、用例利用型(example-based)の手法を用いて翻訳を行う。用例利用型の手法では、あらかじめ収集した翻訳例の中から翻訳入力に最も類似する原言語表現を求め、その原言語表現に対応する目的言語表現を模倣することにより翻訳結果を作る。しかし、翻訳入力が文の場合、あらゆる入力文に対し類似する文をすべて用意することは不可能である。そこで、TDMTでは、いろいろな構文構造単位の翻訳例を組み合わせることで翻訳結果を作る。

### 3.1 構文構造

構文構造の基本単位を表すのは構成素境界パターンである<sup>6)</sup>。構成素境界パターン(以下、パターンと呼ぶ)は、変項と構成素境界より成る。変項は、XやYなどの記号により表し、構成素に対応する。構成素境界になるのは、構成素を関係づけたり修飾したりする助詞や助動詞など機能語である。たとえば「京都に行く」の構文構造は、格助詞「に」を構成素境界とするパターン「XにY」を用いて表す。また「こちら事務局」や「I go」などの表現のように、構成素の間に機能語がない場合などには、品詞バイグラムマーカを2つの構成素の間に挿入する。前方の構成素の最後の語の品詞をA、後続する構成素の先頭の語の品詞をBとすると、<A-B>を品詞バイグラムマーカと定義する。品詞バイグラムマーカの挿入により、「こちら事務局」と「I go」に対して、図7のような構文構造を作ることができる。

TDMTでは、構成素境界解析(constituent boundary parsing)という構文解析の手法により、チャート法に基づくアルゴリズムで逐次的(left-to-right)に翻訳入力の語を読み込んで、パターンを組み合わせた構文構造をボトムアップに作り上げる<sup>6)</sup>。たとえば、「京都に行きます」や「I go to Kyoto」の構文構造は、図8のようなパターンの組合せで表す。

### 3.2 変換知識

TDMTでは、パターンごとに、照合する翻訳例を収集、編集することにより、原言語表現と目的言語表現を対応づける変換知識を作る(この作業を以下、翻訳訓練と呼ぶ)。TDMTの翻訳処理の中心である構文解析と変換は、変換知識の情報を用いることにより行

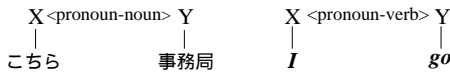


図7 品詞バイグラムマーカを用いた構文構造の記述

Fig. 7 Structure description using part-of-speech bigram marker.

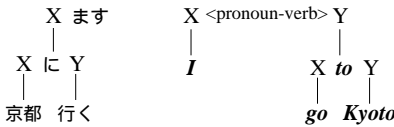


図8 構成素境界パターンの組合せ

Fig. 8 Combination of constituent boundary patterns.

われる。たとえば、日本語表現「X の Y」について、「ホテルの住所」と“the address of the hotel”、「英語のパンフレット」と“the pamphlet in English”などの翻訳例を収集し、次のような日英の変換知識を作る。

- X の Y ⇒
- Y' of X' ((ホテル, 住所), (新幹線, 切符), ...),
  - Y' in X' ((英語, パンフレット), ...),
  - Y' for X' ((明日, 天候), ...),
  - :

X' は X の対訳を示す。この変換知識は「X の Y」が“Y' of X'”や“Y' in X'”などさまざまな目的言語表現の候補を持つことを表しており、それぞれの目的言語表現が選ばれる場合の変項 X と Y の具体的な語を“(ホテル, 住所)”のように併記する。たとえば「日本語のパンフレット」という入力に対して、パターン「X の Y」に関する変換知識の翻訳例の中で、最小の意味距離（この意味距離を「類似用例との意味距離」と定義する）を与える原言語表現が「英語のパンフレット」であるとすると、“Y' in X'”を「X の Y」の目的言語表現に決定し、“the pamphlet in Japanese”という翻訳結果を得る。現在、TDMT では、シソーラス上での意味属性の位置関係により単語間の意味距離を定義し<sup>12)</sup>、パターンの変項についての単語間の意味距離の合計値を、パターンに関する意味距離としている<sup>6)</sup>。

翻訳例を大量に学習することが、変換知識における訳し分けの条件を詳細に記述し、目的言語表現の決定の精度を高めることになる。

### 3.3 構文構造のスコア

TDMT では、現在、構文構造を構成するそれぞれのパターンに関する類似用例との意味距離をすべて合計した値を、その構文構造のスコアと定義する<sup>3), 6)</sup>。構文構造の曖昧性がある場合は、このスコアの最小の



パターン	類似用例との意味距離	パターン	類似用例との意味距離
X と Y (動詞句)	0.17	X に Y	0.01
X に Y	0.01	X と Y (名詞句)	0.50

○ 意味距離の合計: 0.18    ✕ 意味距離の合計: 0.51

図9 「友人と東京に行く」に対する構文構造の候補

Fig. 9 Candidate structures for “Yuujin to Tokyo ni iku”.

のがシステムの持つ知識と最も合致すると判断され、最尤の構文構造として選ばれる。

「X と Y」というパターンの構造レベルには、動詞句や名詞句など複数の可能性がある。図9は、「友人と東京に行く」に対する構文構造の候補を示す。構文構造を構成するそれぞれのパターンについて、類似用例との意味距離が図9のように与えられるとすると、意味距離の合計値は、「X と Y」が動詞句の場合では0.18であり、名詞句の場合では0.51である。よって、「X と Y」が動詞句である方が、意味距離の合計値が小さいので、「友人と東京に行く」の構文構造として選ばれる。

## 4. TDMT への分割処理の導入

本章では、2章で述べた分割処理を TDMT へ導入する手法について説明する。

### 4.1 構成素境界から成る構文構造

TDMT でチャート法の不活性弧に相当するのは、パターンの組合せで変項に未確定部分がない状態のもの、あるいは1個の内容語であり、これらは構文構造と認定される。さらに、分割処理の導入で、2.1節の戦略(C)により、1個の機能語でも、不活性弧を張り構文構造を作ることができる。TDMT では、機能語と品詞バイグラムマーカを構成素境界として使う。したがって、戦略(C)を以下のように拡張する。

(C') 内容語に加えて、構成素境界(機能語、または品詞バイグラムマーカ)も不活性弧を張り、構文構造を構成できる。

例(6)では、“please”と“two”の間に品詞バイグラムマーカ<adverb-numeral>を挿入する。

(6) “four please two children two adults”

“four please”のような文と、“two children two

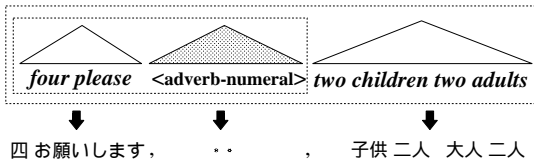


図 10 例 (6) の英日翻訳

Fig. 10 English-to-Japanese translation of (6).

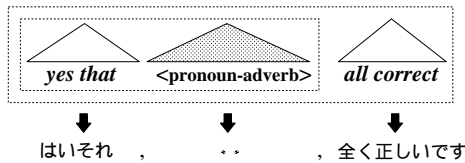


図 11 例 (7') に対する英日翻訳

Fig. 11 English-to-Japanese translations of (7').

adults” のような名詞句を関係づけるパターン “X <adverb-numeral> Y” がシステムにまだない場合でも、戦略 (C') により、構文解析と英日翻訳が可能になる。分割を用いた (6) の英日翻訳の結果を図 10 に示す。

また、(7') は、音声入力された英語発話 (7) の誤認識結果の例である。

(7) “yes that 's all correct”

(7') “yes that all correct”

発話された “'s” は音声認識で脱落している。戦略 (C') により、(7') のような翻訳入力も構文解析と英日翻訳に成功することができる。(7') について、分割を用いた英日翻訳の結果を図 11 に示す。

#### 4.2 分割に対する意味距離の付与

TDMT では意味距離計算に基づいて構文スコアを与えているので、2.1 節の戦略 (D) を以下のように拡張する。

(D') 分割により構文構造を作るときは、大きな意味距離を付与する。

分割に対して与える意味距離とパターンに与える意味距離との相関関係によって構文解析における分割の優先度は変わってくる。現在の TDMT の意味距離計算の体系では、それぞれの変項について与える意味距離の変域は 0~1 としている。変項を 2 個持つパターン「X の Y」の意味距離の変域は 0~2 となる。1 つの形態素または品詞バイグラムマーカより成る構文構

例 (6) には図 10 で表示されたもの以外にも品詞バイグラムマーカが挿入されるが、図を単純化するため、本論文では、戦略 (C') に従って構文構造を構成する場合のみ品詞バイグラムマーカを明示することにする。

造の意味距離には 0.00 を与えている。

分割に与える最適なスコアは、構文スコアの計算方法、翻訳訓練の量、翻訳対象言語の種類などに影響される。現在は、5.00 という暫定値を分割に対して与えている。これは、(1) 当面、パターンとの組合せによって構成された構文構造を優先し、構文構造を何とか出力するための補完機能として分割を位置づけた、(2) TDMT システムの日英翻訳において、現時点では、他の値と比較して、ほぼ最善の翻訳品質をもたらしていた、などの理由による。このような意味距離の設定のうえで、5 章で述べる評価実験を行った。

例 (8) は複数の文から成り、構文解析および翻訳を成功させるには分割が必要である。

(8) “we'd like to arrive around eleven is that all right”

(8) の構文構造の候補として図 12 の (a), (b), (c) に示す構文構造を仮定する。各構文構造の最上部の “●” の右に示した数値は、その構文構造全体の意味距離の合計値、すなわち、構文スコアとする。分割で与える意味距離を S とすると、意味距離の合計値は以下のようになる。

$$(a) 0.83+0.00+0.02+S \times 2=2S+0.85$$

$$(b) 0.17+0.01+0.00+0.02+S \times 3=3S+0.20$$

$$(c) 0.17+0.50+0.00+0.00+S \times 3=3S+0.67$$

TDMT システムの現在の S の値 5.00 を代入すると、(a) が 10.85、(b) が 15.20、(c) が 15.67 となる。ゆえに、合計値が最小の (a) が最尤の構文構造に選ばれる。

#### 5. 評価実験

本章では、TDMT システムの日英、英日、日韓、韓日の翻訳についての実験結果を用いて分割の効果を示す。

実験した TDMT システムは、旅行会話の多言語翻訳を対象としている。形態素辞書の語彙数は、日本語 (日英と日韓で使用) が約 13,000 語、英語 (英日で使用) が約 7,000 語、韓国語 (韓日で使用) が約 4,000 語である。翻訳訓練の文数は、日英と英日が約 2,900 文、日韓が約 1,400 文、韓日が約 600 文である。

分割に与える最適なスコアについては、TDMT における最善の意味距離と構文スコアという研究課題の中で、今後、検討していく必要がある。

本章で示す実験結果において、システムへの入力、文字列でなく正解形態素列とした。これは、形態素解析誤りの場合を除くことにより、分割処理に絞って評価できるようにするためである。

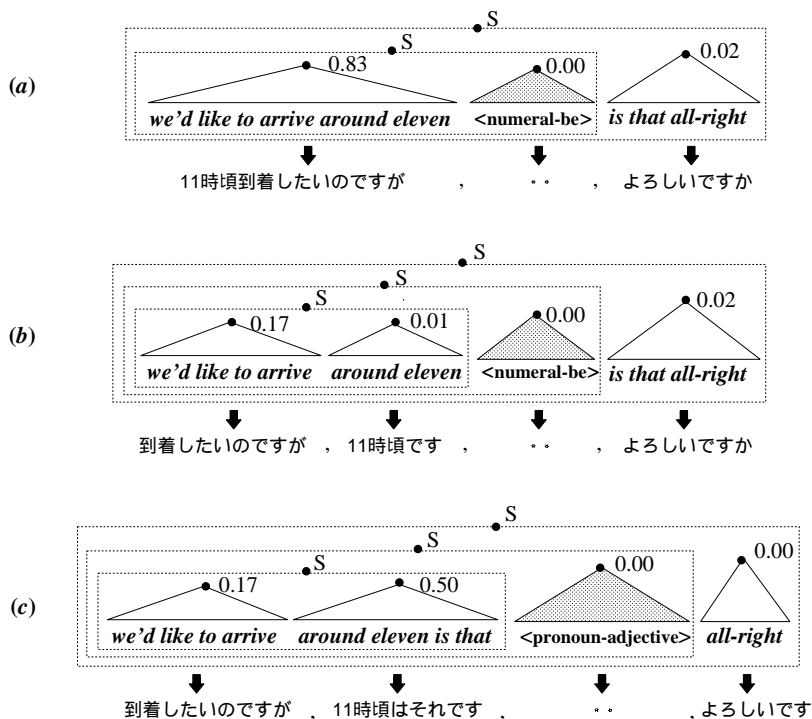


図 12 例 (8) に対する英日翻訳  
 Fig. 12 English-to-Japanese translations of (8).

表 1 翻訳性能における分割の効果  
 Table 1 Effect of splitting on translation performance.

	翻訳出力率 (%)		構文解析成功率 (%)		翻訳成功率 (%)	
	分割処理なし	分割処理あり	分割処理なし	分割処理あり	分割処理なし	分割処理あり
日英	95.8	100	77.6	77.8	77.1	78.5
英日	94.2	100	73.5	74.6	80.8	83.8
日韓	87.2	100	67.9	70.5	82.6	93.0
韓日	70.4	100	57.3	60.0	67.5	92.2

5.1 翻訳性能の向上

旅行会話に関するバイリンガル模擬会話の言語データベース<sup>4)</sup>から評価文を無作為抽出して、翻訳性能のブラインドテストを行った。各言語ペアの翻訳それぞれについて異なり1,000文以上となるよう評価文を抽出した結果、のべ文数は、日本語(日英と日韓の評価用)が1,225、英語(英日の評価用)が1,341、韓国語(韓日の評価用)が1,174となった。翻訳性能に関して評価した項目は以下のとおりであり、のべ文数に対して数値を計算した。

翻訳出力率： 何らかの翻訳結果を出力できた文の割合

構文解析成功率： すべての部分木が誤りでないような構文構造が得られた文の割合

翻訳成功率： 話し手の言いたいことの最低限必

要な内容が翻訳結果に現れていると判断される文の割合

表1に分割処理の有無でシステムの性能を比較した結果を示す。分割処理はすべての項目について翻訳性能を向上させている。

分割処理によって、すべての種類の翻訳において100%の翻訳出力率が得られるようになった。内容語とパターンには何らかの変換結果が与えられるので、分割処理を用いることにより、入力的一部分にでも翻訳可能な部分があれば、翻訳結果が何か得られ、翻訳結果なしという事態を避けることができた。機械翻訳システムでは、どのような入力に対しても、利用段

形態素解析の出力がない、入力に内容語がないという場合には、翻訳結果なしという状況もありうる。

階での後編集，開発段階でのデバッグ，などのために，何らかの翻訳結果を出力することが望ましい．特に，音声翻訳システムを利用している状況では，何も出力がないと会話の進行を妨げる事態も起こりうる．

日韓と韓日の翻訳出力率が「分割処理なし」において低いのは，日英と英日に比べて翻訳訓練の文数が少なく，システムにとっての未知の構文構造を評価文に含んでいる場合が多かったためである．しかし，分割処理は，翻訳規則の不足を補い，翻訳出力率 100% を達成させている．分割による翻訳成功率の向上は，日韓と韓日が，日英と英日に比べて大きい．日本語と韓国語は語順，構文構造，省略表現などで類似するため，断片的にでも翻訳結果を与えれば言われている内容が何とか推察できる場合が多かったためである．

また，2 文以上から成る日本語 277 発話と英語 368 発話を用いて，文単位への分割についての成功率を調べた結果，長い翻訳入力を文単位へ正しく分割する提案手法の効果により，日本語 83.8%，英語 69.3% の成功率で文の切れ目を正しく検知することができた．

## 5.2 音声認識後の翻訳

日英の音声翻訳における分割の効果を誤認識された 47 の日本語入力を用いて検証した．これらの入力は翻訳訓練されており，正しく音声認識されれば，TDMT システムによる翻訳は成功する．誤認識結果は文献 10) の方法を用いた実験データから収集した．

翻訳結果から発話の意味がどのくらい理解できるかというレベル分けし，それぞれのレベルに属する文数を示したのが表 2 である．発話の全体または一部が理解可能だったのは，「分割処理なし」でわずか 19.1% だが「分割処理あり」で 57.4% に増加した．このことは分割が音声翻訳に有効であることを示す．分割処理を使ったにもかかわらず音声翻訳に失敗した原因が多かったのは，述部のような文の意味的に中心な部分を誤認識したことであった．

## 5.3 翻訳時間

提案手法は逐次的構文解析の枠組みの中で実行されるので，翻訳時間に重大な影響を与えることはない．TDMT システムの日英と英日の翻訳時間を，分割処理のあるなし 2 つの場合について比較した．表 3 は，1 文，2 文以上より成る 1 発話，それぞれを入力とした日英と英日の翻訳時間を示す．日英翻訳の日本語入力の平均形態素数は，1 文 9.7 語，1 発話 15.2 語である．英日翻訳の英語入力の平均形態素数は，1 文 7.1 語，1 発話 11.4 語である．処理時間は，Common Lisp で記述したプログラムをコンパイルした TDMT システムについて，SPARCstation10 上で計測した．

表 2 誤った音声認識結果に対する翻訳  
Table 2 Translation after erroneous recognition.

	文数	
	分割処理なし	分割処理あり
全体を理解可能	6	15
	(12.8%)	→ (31.9%)
部分的に理解可能	3	12
	(6.3%)	→ (25.5%)
誤って理解， または，理解不能	38	20
	(80.9%)	→ (42.6%)

表 3 翻訳時間

Table 3 Translation time.

入力 単位	日本語 (日英)		英語 (英日)	
	分割処理 なし	分割処理 あり	分割処理 なし	分割処理 あり
1 文	0.35 sec	→ 0.36 sec	0.55 sec	→ 0.57 sec
1 発話	0.78 sec	→ 0.80 sec	0.60 sec	→ 0.61 sec

分割処理の有無で翻訳時間に大きな差はないことが示されたことから，分割処理を導入しても TDMT システムの処理の効率性が維持されると期待できる．

## 6. おわりに

頑健な音声翻訳を実現するために多様な話し言葉の翻訳入力を分割する手法を提案した．多言語翻訳を行う変換主導型機械翻訳システムに提案手法を導入して，処理時間に悪影響を与えず翻訳性能を向上させたことを実験結果により示した．提案手法は，言語に依存しない汎用的な枠組みで，文法規範から逸脱した表現や，複数の文などにより成る長い発話などの構文解析や翻訳を成功させることができるので，多言語音声翻訳に有効である．

分割により音声翻訳の精度をさらに向上させるための主な課題としては，補完処理と構文スコアがあげられる．特に日英翻訳では，主語や目的語を補完しないと翻訳結果により正しい意味内容が得られない場合が多い．省略が多い話し言葉を対象とする補完処理<sup>15)</sup>と分割処理とを組み合わせる必要がある．また，“and” を多く含む英語発話など正しい分割が困難な翻訳入力に対して，構文解析が成功するよう構文スコアを改良していく必要がある．

実用的な音声翻訳システムに向けた同時通訳機構の実現も今後の課題である．同時通訳機構は，入力が終了してから処理を開始するのではなく入力の途中でも適当なタイミングで部分的に出力を行っていく．提案手法は逐次的な構文解析の枠組みの中での分割により，構文構造の候補を漸進的に作っていくことができる．それゆえ，本手法による構文解析と漸進的生成処



理<sup>1),2)</sup>などを組み合わせた同時通訳機構のモデルの構築について検討していく予定である。

謝辞 音声認識データおよびその翻訳結果を提供していただいた ATR 音声翻訳通信研究所の脇田由実氏〔現在、松下電器産業(株)〕に感謝いたします。

### 参考文献

- 1) 赤峯 享, 古瀬 蔵: 英日対話文翻訳における漸進的な日本語文生成法, 言語処理学会第 1 回年次大会発表論文集, pp.281-284 (1995).
- 2) Finkler, W. and Schauder, A.: Effects of Incremental Output on Incremental Natural Language Generation, *Proc. 10th European Conference on Artificial Intelligence*, pp.505-507 (1992).
- 3) 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, Vol.35, No.3, pp.414-425 (1994).
- 4) Furuse, O., Sobashima, Y., Takezawa, T. and Uratani, N.: Bilingual Corpus for Speech Translation, *Proc. AAAI-94 Workshop on Integration of Natural Language and Speech Processing*, pp.84-91 (1994).
- 5) Furuse, O., Yamada, S. and Yamamoto, K.: Splitting Long or Ill-formed Input for Robust Spoken-language Translation, *Proc. COLING-ACL'98*, pp.421-427 (1998).
- 6) 古瀬 蔵, 山本和英, 山田節夫: 構成素境界解析を用いた多言語話し言葉翻訳, 自然言語処理, Vol.6, No.5, pp.63-92 (1999).
- 7) 兵藤安昭, 池田尚志: 表層的情報と N 近傍ブロック化手法による日本語長文の骨格構造解析, 情報処理学会論文誌, Vol.36, No.9, pp.2091-2101 (1995).
- 8) 金 淵培, 江原暉将: 日英機械翻訳のための日本語長文自動短文分割と主語の補完, 情報処理学会論文誌, Vol.35, No.6, pp.1018-1028 (1994).
- 9) 黒橋禎夫, 長尾 眞: 長い日本語文における並列構造の推定, 情報処理学会論文誌, Vol.33, No.8, pp.1022-1031 (1992).
- 10) Shimizu, T., Yamamoto, H., Masataki, H., Matsunaga, S. and Sagisaka, Y.: Spontaneous Dialogue Speech Recognition using Crossword Context Constrained Word Graphs, *Proc. ICASSP '96*, pp.145-148 (1996).
- 11) 白井 諭, 瀬下貴加子, 木村淳子, 横尾昭男, 池原 悟: 従属節の階層構造に基づく日本語長文の自動分割とその効果, 第 53 回情報処理学会全国大会論文集(分冊 2), pp.67-68 (1996).
- 12) Sumita, E. and Iida, H.: Example-based Transfer of Japanese Adnominal Particles into English, *IEICE Trans. Information and Sys-*

*tems*, Vol.E75-D, No.4, pp.585-594 (1992).

- 13) 竹沢寿幸, 森元 暉: 発話単位分割または接合による言語処理単位への変換手法, 自然言語処理, Vol.6, No.2, pp.83-95 (1999).
- 14) 脇田由実, 河井 淳, 飯田 仁: 意味的類似性を用いた音声認識正解部分の特定法と正解部分のみ翻訳する音声翻訳手法, 自然言語処理, Vol.5, No.4, pp.111-125 (1998).
- 15) 山本和英, 隅田英一郎: 決定木学習による日本語対話文の格要素省略補完, 自然言語処理, Vol.6, No.1, pp.3-28 (1999).

(平成 11 年 12 月 3 日受付)

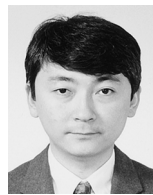
(平成 13 年 2 月 1 日採録)



古瀬 蔵 (正会員)

1982 年九州大学工学部情報工学科卒業。1984 年同大学大学院情報システム学専攻修士課程修了。同年日本電信電話公社入社。1990 年より ATR 自動翻訳電話研究所, ATR

音声翻訳通信研究所へ出向。1997 年日本電信電話株式会社へ復帰。現在, NTT サイバーソリューション研究所。自然言語処理, 特に機械翻訳の研究に従事。言語処理学会, 電子情報通信学会各会員。



山田 節夫 (正会員)

1990 年東京電機大学理工学部情報科学科卒業。1992 年同大学大学院情報科学専攻修士課程修了。同年日本電信電話(株)入社。1997 年より ATR 音声翻訳通信研究所, ATR

音声言語通信研究所へ出向。2000 年日本電信電話株式会社へ復帰。現在 NTT コミュニケーション科学基礎研究所。自然言語処理, 特に機械翻訳の研究に従事。言語処理学会会員。



山本 和英 (正会員)

1996 年豊橋技術科学大学大学院博士後期課程システム情報工学専攻修了。博士(工学)。同年 ATR 音声翻訳通信研究所客員研究員, 2000 年 ATR 音声言語通信研究所客員研究員, 現在に至る。1998 年中国科学院自動化研究所国外訪問学者。自然言語処理, 特に換言処理, 機械翻訳, 要約処理の研究に従事。言語処理学会, The Association for Computational Linguistics 各会員。