

同時通訳システムの現状と課題

松原茂樹¹

(¹名古屋大学大学院情報科学研究科)

This paper describes the current status of automatic simultaneous interpretation research. First, the configuration of a simultaneous interpretation systems and its technical issues are discussed. Next, as an example of automatic simultaneous interpretation research, the technologies for incremental parsing, transfer and generation are explained, and then the implementation of a prototype system is introduced. Finally, the future research tasks for advancement of simultaneous interpretation systems are shown.

1. はじめに

同時通訳はどのように遂行されるか。この原理的な問いに対して、これまで様々な説明が与えられてきた。典型的には、同時通訳で生じる現象や同時通訳に特有の過程などを対象に、それを捉えるモデルを提示することにより行われた。モデルの構築にあたり、言語学や心理学、認知科学、脳科学など、各分野の既存モデルを、対象に応じて拡張あるいは精緻化するという方法論などが採られてきた。

一方、著者らは、情報工学の分野から同時通訳にアプローチしている。これは、同時通訳機能を備えた計算機の実現が目的であり、同時通訳という情報処理に関するモデル、すなわち、同時通訳の計算アルゴリズムを与えることを目指している。アルゴリズムとして妥当であるためには少なくとも、入力から出力に至るまでの計算機の処理手順を、曖昧さを残すことなく記述する必要がある。他の分野のモデルと異なり、人間による同時通訳過程を正しく説明することがモデル化の目的ではない。ユーザにとって好ましい出力を与えることができれば、計算機がどのような処理過程を経たかに関わらず、そこで導入されたアルゴリズムは優れているとみなされる(ただし、一般的には、知的処理の計算アルゴリズムを開発する上で、人間の処理過程を模倣することは有効であり、人間の同時通訳過程を解明し、モデル化することは、情報工学の観点からも意義が大きい。)

MATSUBARA, Shigeki. "Current status and research issues of simultaneous interpretation technologies," *Interpretation and Translation Studies*, No. 12, 2012, Pages 21-30. © by the Japan Association for Interpreting and Translation Studies

本稿では、同時通訳システム研究の現状と課題について概説する。近年、音声認識や音声合成、機械翻訳の性能が格段に進化し、PC 向け通訳ソフトウェアの市販化に加え、スマートフォン用のアプリも登場するなど、自動通訳が身近な技術となりつつある(中村他 2008; 奥村 2008)。しかしながら、これらはいずれも逐次通訳として実現されている。一方、同時通訳技術については、近未来の自動通訳と位置付けられ、現状では世界的にも僅かに研究が行われているに過ぎず(Amtrup, 1999; Casacuberta et al., 2002; Ryu et al., 2006)、近い将来の活性化が待たれる状況にある。

本稿の構成は以下の通りである。第 2 章では、同時通訳システムの設計と実現のための課題について論じる。第 3 章以降では、英日同時通訳システムの研究開発の現状として著者らの試みを概説する。第 3 章で言語解析技術を、第 4 章で変換及び生成技術を説明し、第 5 章でシステムの実装と評価を述べる。第 6 章でまとめと今後の課題を示す。

2. 同時通訳システムの設計

本章では、同時通訳システムの設計について述べる。まず、自動通訳システムの標準的な構成と各要素技術の概要について説明し、次に、同時通訳機能を実現するための技術的課題として、各々の要素技術をどのように改良すべきかについて論じる。

2.1 自動通訳のシステム構成

これまで研究開発されてきた自動通訳システムの多くは、空港のチェックインやレストランでの注文など、海外旅行の典型的な場面で展開されるような会話の支援を想定している。システムには、1文単位で音声が入力されることを前提とし、文の入力が完了すると通訳処理を実行する。

自動通訳システムは、一般に、

- 音声認識:入力として1文の音声を受け取り、それを文字列に変換する。
- 言語翻訳:文字化された文を別の言語の文に変換する。
- 音声合成:翻訳後の文を音声に変換し、出力する。

という 3 つの要素技術から構成される。この順で処理を実行することにより、原言語(source language)の音声を目的言語(target language)の音声に変換する。

これらの要素技術はいずれも、近年、格段に性能が向上しており、すでに実用化の域に達している。音声認識ソフトウェアは、多くのスマートフォンに標準搭載されるなど、急速に普及しており、雑音環境下でも高い認識性能を発揮するようになった。機械翻訳は Web サービスとして定着しつつあり、Web サイト翻訳などの用途で広く利用されている。また、音声合成が組み込まれた機器や社会サービスも増加し、街中で合成音を聴く機会も多い。このような技術的進展には、大規模な音声・言語データの利用が容易になったことが大きく寄与している。最近の音声認識、機械翻訳、音声合成は、いずれも統計に基づく確率的アプローチを採用しており、

使用するデータが大規模化するほど、品質の高い確率モデルを獲得できるためである。自動通訳システムは、これらの3つのソフトウェアを結合すれば実現でき、要素技術のレベルが上がれば、システムの通訳性能も向上する。

2.2 同時通訳機能を実現するための技術的課題

前節のシステム構成で実現される通訳では、1文ごとに処理が実行されるため、通訳様式は逐次となる。このような自動通訳に、新たに同時通訳機能を付加することを目指すとき、各構成要素の技術的課題は以下のようになる。

- 音声認識: 現行の音声認識は統計的手法で実現されており、この方式では、入力された音声信号に対して、音響や言語に関する確率モデルを適用し、確率が最大となる文字列を認識結果として出力する。認識結果を確定し出力するタイミングは、通常、休止(pause)を検出したときとなる。一方、同時通訳のための音声認識では、より早い段階で認識結果を出力することが望ましい。休止の検出に依存することなく認識結果を確定する方式の開発が進められており(今井他 2001)、そのような技術を導入することが考えられる。
- 言語翻訳: これまで構文変換、用例主導、統計翻訳など様々な翻訳方式が開発されている。いずれの方式でも、まずは入力文全体の構造を捉えることが先決であり、このことは、文全体が確定してからでないと翻訳処理を実行できないことを意味する。同時通訳のための言語翻訳では、文全体が入力される前の時点でも、すでに入力された部分的な単語列に対してその構造を解析し、目的言語に変換する仕組みが必要となる。
- 音声合成: 自然な音声を生成するために、出力する文の構造に関する情報に基づき、アクセントやイントネーション、休止位置などが決定される。ただし、多くの場合、それらは局所的な情報から定めることができるため、必ずしも訳出する文の全体が確定していなくても、自然な合成音を作り出すことはできる。一方で、聴衆にとって聞きやすい同時通訳音声に関して分析が試みられており(遠山他 2005; 遠山他 2007)、同時通訳のための音声合成では、そのような知見を考慮した音声出力が求められる。

本稿の以下では、自動通訳を構成するこれらの要素技術のうち、特に言語翻訳に焦点をあてて論じる。

2.3 同時通訳のための言語翻訳

自動通訳システムの言語翻訳は、一般的には、さらに以下の3つの要素技術から構成される。

- 解析: 文字化された原言語文の構造を作成する。
- 変換: 原言語文の構造を目的言語文の構造に変換する。
- 生成: 目的言語文の構造をもとに、訳出する文を作り上げる。

現状の言語処理では、これらの要素技術はいずれも、文を単位とする処理方式が一般的である。同時通訳システムに導入するためには、文の入力途中の段階で順次処理を実行できるように、それぞれ新たな技術の開発が必要である。

次章及び次々章では、同時通訳のための言語解析・変換・生成の技術として、著者らが開発した手法について概説する。

3. 漸進的な構文解析

自然言語の構文解析(parsing)とは、文の構成要素間の構文的関係を明らかにすることである。既存の方法の多くは、文が入力されていることを前提とし、文全体に対する構文構造を出力する。一方、同時通訳のための構文解析では、音声入力途中の段階でそれまでに文字化された部分に対してその解析結果を作成し、それを次の変換処理に渡すことが必要であり、これは漸進的な構文解析(incremental parsing)と呼ばれる。

自然言語の文法として様々な形式が存在するが、各言語に備わる性質によって、計算に適した文法形式は異なる。英語では、文脈自由文法(context free grammar)を用いることが多いが、目的言語のタイプによっては、依存文法(dependency grammar)が適する場合もある。本章では、英語を対象に、漸進的な構文解析の技法について概説する。

3.1 文脈自由文法による解析

文脈自由文法は、形式文法の一つであり、文法規則の集合から構成される。例えば、図 1 のような文脈自由文法が存在するとき、英語文

(1) I want to fly from San Francisco to Denver next Monday.

の構文構造は図 2 に示すような木構造(tree structure)で表現される。

$$\begin{aligned} S &\rightarrow NP VP \mid VP \\ VP &\rightarrow VBP S \mid TO VP \mid VP PP PP NP \\ PP &\rightarrow IN NP \end{aligned}$$

図 1 文脈自由文法の例

文脈自由文法を用いた構文解析では、文の入力に対して木構造を出力する。そのような解析アルゴリズムは多数存在する。入力文に対して可能な文法規則を適用することにより構文木を成長させ、最終的に文(S)を根とする構文木が文全体に対して生成できれば解析成功となる。

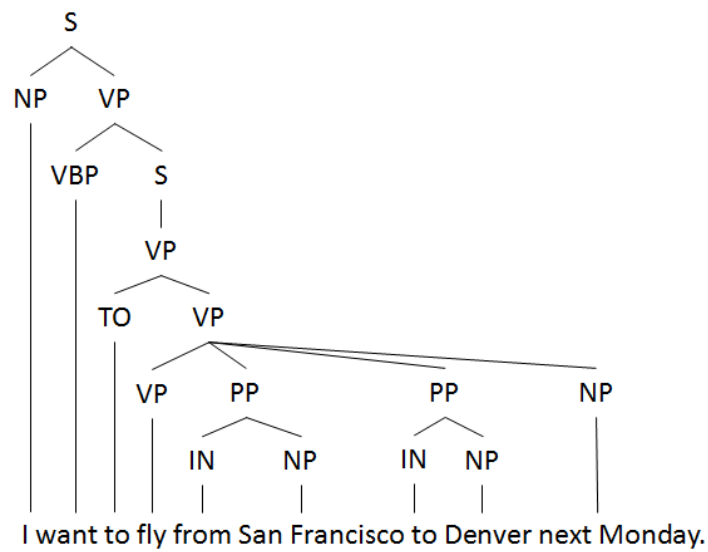


図 2 “I want to fly from San Francisco to Denver next Monday.”の構文木

一方、漸進的な構文解析では、文の入力途中の段階でそれまでの入力に対する構文木を作成する必要がある。これは、新たな単語が入力されるたびに、その単語に対する構文木を作成し、作成した構文木をすでに入力済の単語列に対する構文木に結合するという方式によって可能となる(加藤他 2003)。例えば、文(1)の部分である“I want to”の入力に対して図3の構文木を作成できる。なお、図中の“?”の部分は未決定であることを示す。

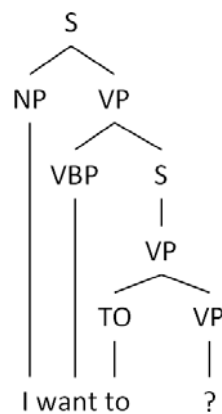


図 3 “I want to”の構文木

3.2 依存構造解析

英語の構文解析では依存文法もよく利用される。特に、日本語の係り受け文法との親和性が高く、英日機械翻訳において日本語に変換することを考えると、依存文法を用いると都合がよい。依存文法では、単語間の修飾・非修飾関係を表現する。依存文法に基づく構文構造は依存構造(dependency structure)と呼ばれる。

文法規則上に句の間の依存関係に関する情報があらかじめ付与された文脈自由文法を用

いて構文解析すると、生成された構文木を依存構造に変換できる。例えば、図 2 の構文木では、文法規則上の依存関係情報から各部分木の主辞(head)を求めることができ、図 4 に示す依存構造を作成できる(矢印は、修飾する単語から修飾される単語への関係を示す。)。この仕組みは漸進的な構文解析にもそのまま応用できるため、文の入力途中の段階でそれまでの入力に対する依存構造を獲得できる(加藤他 2003)。例えば、“I want to”に対する構文木(図 3)から得られる依存構造は図 5 のようになる。



図 4 “I want to fly from San Francisco to Denver next Monday.”の依存構造

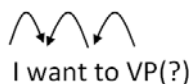


図 5 “I want to”の依存構造

3.3 確率文法を用いた解析

上述の方法により、文の入力途中で構文構造を生成でき、それをその時点で変換処理に渡すことができる。同時通訳の変換処理は、一つの正しい構文木を受け取ることができればよい。しかし一般に、文脈自由文法による解析には曖昧性があり、一文に対して膨大な数の構文木が生成されることになる。あまりに膨大な数の構文木の作成は解析処理時間の増大という問題を引き起こす。

この問題を避けるための一つの方法として、確率文脈自由文法(probabilistic context free grammar)を使用することが考えられる(加藤他 2002)。これは、文脈自由文法の各文法規則に、それが使用される確率が付与されたものである。この確率は、構文木が付与された大規模コーパスから自動獲得できる。

確率文脈自由文法を用いれば、構文木の生成確率を計算できる。このため、一定の確率値に満たない構文木は作成しないなど、確率に応じて構文木の作成を抑制することができる。また、変換処理には、作成した構文木の中から確率が最大となる構文木のみを渡せばよい。

4. 漸進的な変換・生成

4.1 同時通訳のための変換処理

漸進的な変換では、英語の依存構造を日本語の係り受け構造に変換する。いずれも、単語や文節などの構成素間の修飾・非修飾関係を表すものであり、構造上自然に対応する。このため、助動詞や前置詞などの扱いに関する変換規則を用意さえすれば、基本的には対訳辞書を用いた語彙的な変換により実現できる。例えば、文(1)の“Denver”まで入力された時

点で作成されている依存構造(図 6)を日本語の係り受け構造に変換すると図 7 のようになる。



図 6 “I want to fly from San Francisco to Denver”の依存構造

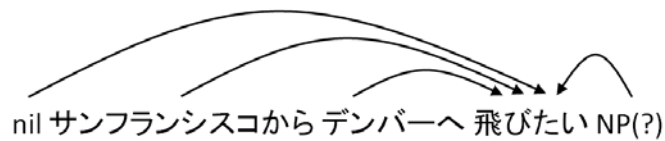


図 7 “I want to fly from San Francisco to Denver”の依存構造から変換した結果

4.2 同時通訳のための生成処理

英語と日本語では語の正規順序が異なるため、英語の入力に対して標準的な日本語を出力すると、訳出に大きな遅延が生じることになる。一方で、英語の語順に従って対応する日本語を生成すると、訳出の遅延は小さくなるものの、当然、生成される日本語は不自然となる。この不自然さは、この方法で生成された日本語が後方修飾性を満たしていないからであるとして説明できる。係り受けの後方修飾性とは、係り受け関係にある 2 つの文節は、必ず前方の文節が後方の文節に係ることをいい、標準的な日本語文はこの性質を充足する。

漸進的な生成では、作成された日本語構造に基づき、どの日本語を訳出するかを制御することが目的の一つであり、この制御を、日本語の係り受け文法における後方修飾性の概念に基づき実行する(Ryu et al. 2006)。

この生成法は、同時通訳者による訳出事例でも観察される。例えば、同時通訳データベース(松原他 2001)に収録されたデータからは以下のような対訳例が見つかる。

(2) You can pick your ticket up at the counter today.

(3) チケットをカウンターで今日お取りいただけます。

訳出された文(3)を構成する表現「チケットを」「カウンターで」「今日」の正規順序は、標準的な日本語とは異なるものの、了解性は備えている。この日本語文は後方修飾性を充足している。すなわち、後方修飾性を満たすという条件のもとで、英語文の生起順序を対応させることにより、より同時性の高い訳出が可能となる。

その一方で、以下のような訳出事例も観察される。

(4) There are three Sushi restaurants near this hotel.

(5) お寿司屋さんが三軒あります、ホテルの近くに。

この訳文には倒置が含まれる。「ホテルの近くに」は「あります」に係っており、後方修飾性を満たさない。しかし、このような倒置により訳出の同時性が高められており、倒置を生起可能な状況下では、それを活用することが考えられる。

これらのアイデアは、後方修飾性を満たすための出力制御アルゴリズム「ある文節について、それを係り先とするすべての文節が出力済であれば、その文節を出力する」、及び、倒置を活用するための出力制御アルゴリズム「ある述部文節について、それを係り先とする文節が複数個存在し、それらが出力済であれば、その文節を出力する」を組み合わせることにより実現できる。

例として、文(1)の“to”が入力された段階で、変換処理によって生成された日本語係り受け構造は図 8 のようであり、そのとき「サンフランシスコから」を出力する。その後、文の入力が完了する前までに「デンバーへ飛びたい」を訳出し、文末の検出後に、「来週の月曜日に」を出力する。出力された訳文は、

(6) サンフランシスコからデンバーへ飛びたい、来週の月曜日に。

となる。訳文には倒置が含まれているが、システムが出力する訳文としては容認できなくはない。

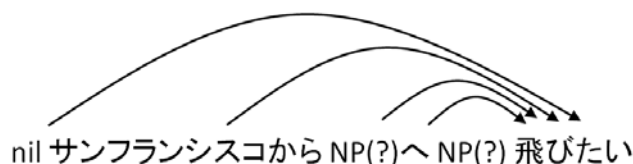


図 8 “I want to fly from San Francisco to”の依存構造から変換した結果

5. 英日同時通訳システムとその評価

前章までに記したアルゴリズムに基づき、英日同時通訳システムを Java で実装している(笠他 2009)。システムは、Windows PC 上で動作する。ただし、現状のシステムでは、文法や語彙などの規模は実験レベルに留まっており、システムとしての汎用性を高めるための研究開発を今後進める必要がある。

Penn Treebank (Marcus et al., 1993) の英語会話文データである ATIS コーパスの全 578 文を使用して通訳実験を実施したところ、71.1%の正解率を得ており、話し言葉の自動通訳としては悪くない程度の品質を備えていることを確認している。また、訳出の遅延も平均で 2.08 文節に留まっており、本稿で記したアイデアを導入することにより、高い同時性が実現されることが示されている。

6. まとめ

本稿では、同時通訳システムの現状と課題について述べた。特に、著者らが研究開発を進めている英日同時通訳システムを取り上げ概説した。計算機による同時通訳の研究は、ここで紹介した著者らの事例を含め世界的にも僅かであるが、次世代自動通訳技術の中核的対象として、今後は同時通訳システムの研究開発が活発になることが予想される(松原 2008)。

本稿では、同時通訳のための言語翻訳に焦点をあてたが、同時性の高い音声認識技術も同様に重要である。この技術は、同時通訳に限らず、音声対話や字幕生成などの応用システ

ムにおいても鍵となる技術であり、今後の技術的進展が期待される状況にある。

また、本稿では、言語翻訳のための方式として、規則に基づく変換を紹介したが、基盤となる手法は単純なものである。近年は、統計翻訳方式が高い翻訳性能を示しており、同時通訳のためのデータが十分に整備されれば、今後、同時通訳システムへの導入が進む可能性がある。

また、同時通訳システムを高度化するにあたり、同時通訳者による通訳テクニックを活用することが有望であり(遠山他 2006)、そのようなテクニックのアルゴリズム化を積極的に進める必要がある。この点で、通訳研究で示されてきたモデルを数理的に再構成することに加え、同時通訳データベースなどの大規模通訳事例を多角的観点から丹念に分析することが極めて重要となる。

.....

【謝辞】

共同研究者である名古屋大学大学院国際開発研究科の笠浩一朗助教に感謝します。本研究は一部、科学研究費補助金(基盤研究(B))「同時的な発話理解のための話し言葉処理に関する研究」(No. 22300051)によります。

.....

【著者紹介】

松原 茂樹 (MATSUBARA Shigeki) 名古屋大学大学院工学研究科博士後期課程修了。博士(工学)。現在、名古屋大学大学院情報科学研究科准教授。自然言語処理、音声言語処理、情報検索、デジタル図書館の研究に従事。

.....

【参考文献】

- 今井亨・田中英輝・安藤彰男・磯野春雄 (2001) 「最ゆう単語列逐次比較による音声認識結果の早期確定」『電子情報通信学会論文誌』第 J84-D-II 巻, 第 9 号, pp. 1942-1949.
- 加藤芳秀・松原茂樹・外山勝彦・稲垣康善 (2003) 「主辞情報付き文脈自由文法に基づく漸進的な依存構造解析」『電子情報通信学会論文誌』第 J86-D-II 巻, 第 1 号, pp. 84-97.
- 加藤芳秀・松原茂樹・外山勝彦・稲垣康善 (2002) 「確率文脈自由文法に基づく漸進的構文解析」『電気学会論文誌』第 C-122 巻, 第 12 号, pp. 2109-2119.
- 遠山仁美・松原茂樹 (2005) 「同時通訳における聴きやすさとポーズの関係, 一同時通訳コーパスを用いた被験者実験による分析一」『通訳研究』第 5 号, pp. 137-155.
- 遠山仁美・松原茂樹 (2006) 「訳出遅延時間と訳出開始タイミングに着目した同時通訳者の原発話追従戦略に関する分析」『通訳研究』第 6 号, pp. 113-128.
- 遠山仁美・松原茂樹 (2007) 「英日同時通訳者発話におけるフィラーの出現と聴きやすさとの関係」『通訳研究』第 7 号, pp. 39-49.
- 奥村明俊 (2008) 「携帯端末用多言語自動通訳システムの実用化に向けて」『情報処理』第 49

- 卷, 第 6 号, pp. 611-616.
- 中村哲・隅田英一郎・清水徹(2008)「ここまでの音声翻訳技術」『情報処理』第 49 卷, 第 6 号, pp. 606-610.
- 松原茂樹・相澤靖之・河口信夫・外山勝彦・稲垣康善 (2001)「同時通訳コーパスの設計と構築」『通訳研究』第 1 号, pp. 85-102.
- 松原茂樹 (2008)「同時通訳の工学と科学 一次世代自動通訳技術の実現に向けて」『情報処理』第 49 卷, 第 6 号, pp. 617-623.
- 笠浩一朗・松原茂樹・稲垣康善 (2009)「英日同時翻訳のための依存構造に基づく訳文生成手法」『電子情報通信学会論文誌』第 J92-D 卷, 第 6 号, pp. 921-933.
- Amtrup, J.W. (1999). Incremental Speech Translation, *Lecture Note in Artificial Intelligence*, Vol. 1735.
- Casacuberta, F., Vidal, E., and Vilar, J.W. (2002). Architectures for Speech-to-Speech Translation using Finite-State Models, *Proceedings of Workshop on Speech-to-Speech Translation: Algorithms and Systems*, pp. 39-44.
- Marcus, M.P., Santorini, B., and Marcinkiewicz, M.A. (1993). Building a Large Annotated Corpus of English: the Penn Treebank, *Computational Linguistics*, Vol. 19, No. 2, pp. 310-330.
- Ryu, K., Matsubara, S., and Inagaki, Y. (2006). Simultaneous English-Japanese Spoken Language Translation Based on Incremental Dependency Parsing and Transfer, *Proceedings of Joint 21th International Conference on Computational Linguistics and 44th Annual Meeting of Association for Computational Linguistics*, pp. 683-690.