同時通訳データを利用した自動同時通訳システムの構築*

☆清水宏晃，Graham Neubig，Sakriani Sakti，戸田智基，中村哲（奈良先端大）

1　はじめに

音声翻訳は，自動的にある言語の音声を異なる言語の音声に翻訳する技術であり，言語の壁を越えて人間の意思疎通を可能にする．しかし，音声翻訳にはいくつか問題点がある．例えば，音声翻訳が日常会話のようなユーザーの発話が短い場面で使用されるとき，ユーザーの発話が翻訳されるまでの処理時間は短いため，実用的に使用できる．一方，音声翻訳が講演のような発話が長い場面に使用されるとき，ユーザーの発話が翻訳されるまでの処理時間は長くなる．それゆえ，発話と翻訳の間に発生する時間のずれ（以降，遅延時間）が原因となり，ユーザーは講演の内容をリアルタイムで理解することが困難となる．

実際，講演を通訳する際は，多くの同時通訳者が，話者の発話が開始されるとまもなく，通訳を開始する．その際，同時通訳者は遅延時間を短縮するために，長い発話を短いチャンクに分割して通訳をしている [1]．音声翻訳における遅延時間の問題を解決するために，同時通訳者のように，自動的に長い発話を短いチャンクに分割し翻訳を行う手法 [2][3] が提案されている．しかし，システムの学習には，翻訳者によって作成された翻訳データが利用されているため，同時通訳者のような翻訳結果にならない．そこで，同時通訳者が同時通訳したデータ（以降，同時通訳データ）を利用すれば，同時通訳者の訳出に近づき，かつ遅延時間の短縮が期待できる．

本稿では，講演を自動的に通訳する同時通訳システムの性能を改善するために，学習に同時通訳データを利用する．実験では，TED 講演に対して同時通訳システムの性能を調査し，遅延時間と翻訳精度の観点から評価を行う．

2　同時通訳データ

本節では，同時通訳システムの学習に利用する同時通訳データについて記述する．

著者ら [4] は，TED 講演[1]を収録材料とし，英日方向の同時通訳音声を収録および書き起こしを行い，同時通訳データベースを構築している．この同時通訳データベースにおける特徴の一つとして，通訳経験年数の異なる複数の同時通訳者が同じ講演を同時通訳している点である．そのため，通訳経験年数による通訳の特徴を比較できる．Fig. 1 は，同時通訳データの書き起こし例を示す．通訳音声に対して文を定めることは困難なため，通訳音声を 0.5sec 以上の無音区間によって分割したものを発話単位として定めている．また，書き起こしデータには，フィラーや言いよどみなどの言語情報を表すタグだけでなく，各発話単位に対して発話開始時間や発話終了時間も付与している．

```
0001 - 00:44:107 - 00:45:043
本日は<H>
0002 - 00:45:552 - 00:49:206
みなさまに(F え)難しい話題についてお話したいと思います。
0003 - 00:49:995 - 00:52:792
(F え)みなさんにとっても意外と身近な話題です。
```

Fig. 1　同時通訳データの書き起こし例．F はフィラー，H は言い伸ばしのタグを示す

3　分割手法

本節では，同時通訳システムに利用する発話の分割手法について記述する．

言語情報を利用した発話の分割手法が Fujita ら [3] によって提案されている．この手法は，原言語を目的言語に翻訳する際に，両言語の語順の並び替えが起きにくい確率 （以降，右確率）を利用して，翻訳単位を決定する．

この手法のアルゴリズムについて記述する．まずフレーズテーブルと呼ばれる対訳データから抽出されたフレーズペアの集合を利用し，フレーズテーブルに存在する最長フレーズを仮の翻訳単位として決定する．具体的には，原言語の入力文を先頭から 1 単語ずつ単語列 H へ追加する途中で，$H = h_1, \ldots, h_j$ がフレーズテーブルの原言語に存在しなくなった場合，$h_1, \ldots, h_{j-1}$ を仮の翻訳単位として選択する．次に，語順の並び替えが起きにくい位置で分割するために，右確率を利用し，最終的な翻訳単位として決定する．具体的には，仮の翻訳単位の右確率と閾値を比較する．右確率が閾値を上回った場合は，最終的な翻訳単位として決定する．一方，右確率が閾値を下回った場合は，次に存在する仮の翻訳単位と結合し，閾値と比較する．この手法の利点は，発話の途中で翻訳を開始できることである．そのため，発話がすべて終了してから翻訳する場合と比較すると，遅延時間が短縮される．

4　同時通訳データのモデル適応

同時通訳システムの性能を改善するために，同時通訳データを学習に利用する．本節では，翻訳精度の改善及び遅延時間の短縮を目指し，同時通訳データを利用する過程について記述する．

4.1　翻訳精度の改善

統計的機械翻訳システムを構築する際に行われる３つの過程に同時通訳データを使用し，翻訳精度の改善を試みる．

チューニング：チューニングとは，テストのドメインに特化するため，統計モデルのパラメータを最適化する方法である．その最適化に用いるデー

---

*Constructing a Automatic Simultaneous Interpretation System using Simultaneous Interpretation Data. by SHIMIZU, Hiroaki, NEUBIG, Graham, SAKTI, Sakriani, TODA, Tomoki, NAKAMURA, Satoshi (NAIST)

Table 1　実験で用いた翻訳モデルの学習データ (TM), 言語モデルの学習データ (LM), チューニングデータ (tune) および テストデータ (test) に使用した TED 講演の翻訳データ (TED-T), 同時通訳データ (TED-I) および英次郎辞書と付属の例文 (DICT) の形態素数

	TED-T	TED-I	DICT
TM (en)	1.57M	29.7k	13.2M
TM (ja)	2.24M	33.9k	19.1M
LM (en)	1.57M	29.7k	13.2M
LM (ja)	2.24M	33.9k	19.1M
tune (en)	12.9k	12.9k	—
tune (ja)	19.1k	16.1k	—
test (en)	—	11.5k	—
test (ja)	—	14.9k	—

タに対し，翻訳データの代わりに同時通訳データを使用する．

言語モデルの学習：言語モデルの学習に同時通訳データを利用する理由は，機械翻訳において，言語モデルは出力文のスタイルに大きな影響を与えるため，同時通訳のような文を生成するには有効と考えられるためである． 翻訳データに比べて同時通訳データを大量に確保することが困難であるため，翻訳データおよび同時通訳データそれぞれに対して言語モデルを構築し，二つのモデルを線形補間で組み合わせる．その際，パラメータは同時通訳データに対して最適化する．

翻訳モデルの学習：翻訳モデルの学習に同時通訳データを利用する理由は，翻訳モデルは翻訳するフレーズに影響を与えるため，同時通訳が使用するフレーズを生成するのに有効と考えられるためである．言語モデルの学習と同様，翻訳データに比べて同時通訳データを大量に確保することが困難であるため，Fill-up 法 [5] を使用して，フレーズテーブルを作成する．Fill-up 法とは，他のフレーズテーブルには存在するが，優先度が最も高いフレーズテーブルにないフレーズおよびスコアを，優先度が最も高いフレーズテーブルに追加する手法である．翻訳データと同時通訳データから作成した 2 つのフレーズテーブルの内，同時通訳データを優先度が高いフレーズテーブルと設定する．

4.2　遅延時間の短縮

同時通訳データには，遅延時間を短縮される工夫がされており，翻訳データと比べて，並び替えが起きにくい．そのため，3 節で紹介した右確率の学習に，同時通訳データを利用することで，分割される翻訳単位の数が変化すると考えられる．そのため，翻訳データに同時通訳データを加え，右確率を学習する．

5　実験

本節では，同時通訳システムの性能を評価する実験について記述する．

5.1　実験データ

タスクは，TED 講演に対して英日方向の通訳である．そのため，データには，TED 講演の翻訳データと同時通訳データを使用する．さらに，辞書として英辞郎辞書と付属の例文[2]を使用する．

各データの詳細は Table 1 に示す．テストの正解文には，同時通訳データを使用している．なぜなら，翻訳データでは必ずしも通訳者に近い訳出を実現できているとは限らないことが著者ら [4] によって報告されているからである．

また，TED 講演の同時通訳データは，文アライメントが取れていないことである． そのため，TM および LM の学習に使用した同時通訳データには，Champollion ToolKit[6] (CTK) により，自動的に文アライメントを取る．一方，tune と test には人手で文アライメントを取る．

5.2　ツールキットと評価手法

同時通訳システムには，Moses [7] のフレーズベース機械翻訳を使用する．なお，デコーディングにおける並び替えの制限 (distortion limit) は 12 と設定する．トークン化には，英語では Moses に含まれるスクリプト，日本語では KyTea [8] を使用する．学習データ間の単語アライメントを取るツールには GIZA++ [9]，目的言語である日本語に対して言語モデルを作成するツールには SRILM [10] を使用し，5-gram で学習する．各素性の重みは MERT [11] を用いて BLEU が最大になるように最適化を行う．

性能の評価は，遅延時間および翻訳精度の観点から行う．翻訳精度には，機械翻訳の自動評価尺度である BLEU [12] および RIBES [13] を用いる．遅延時間 D は，$D = A + T$ で計算を行った [3]．A は一文あたりの音声認識に要した時間を表す．T は一文あたりのデコーディングにかかる平均時間を表す．

5.3　同時通訳データと翻訳精度に関する実験

4.1 節で記載した 3 つの過程に同時通訳データを使用し，翻訳精度の改善について調査した．実験のテストデータは，人手の書き起こしデータを使用した．分割法は，3 節に紹介した右確率を用いた手法を使用した．右確率の学習には，TED-T のみを使用した．閾値は，0.0，0.2，0.4，0.6，0.8，1.0 の場合を実験した．比較対象として，翻訳データのみ (Baseline) の場合と，同時通訳データをチューニング (Tu)，チューニング及び言語モデル (LM+Tu)，チューニング，言語モデル及び翻訳モデル (TM+LM+Tu) に追加した場合を実験した．実験結果を Fig 2 に示す．

Fig 2 より，BLEU と遅延時間の観点から評価したグラフと見ると，Tu は Baseline と比べ，翻訳精度に統計的有意差が確認できなかった．しかし，LM+Tu 及び TM+LM+Tu は Baseline と比べ，右確率が 0.8 及び 1.0 では，翻訳精度に関して統計的有意差が確認できた．例えば，Baseline は BLEU が 7.81 の翻訳結果を得るのに 5.23 秒要している．しかし，TM+LM+Tu はわずか 2.08 秒で BLEU が 8.39 の翻訳結果を得ることができた．

次に，RIBES と遅延時間の観点から評価したグラフを見ると，Tu，LM+Tu 及び TM+LM+Tu は Baseline と比べ，翻訳精度に統計的有意差が確認できなかった．考え得る原因の一つは，チューニングであ

[1] http://www.ted.com
[2] http://eijiro.jp

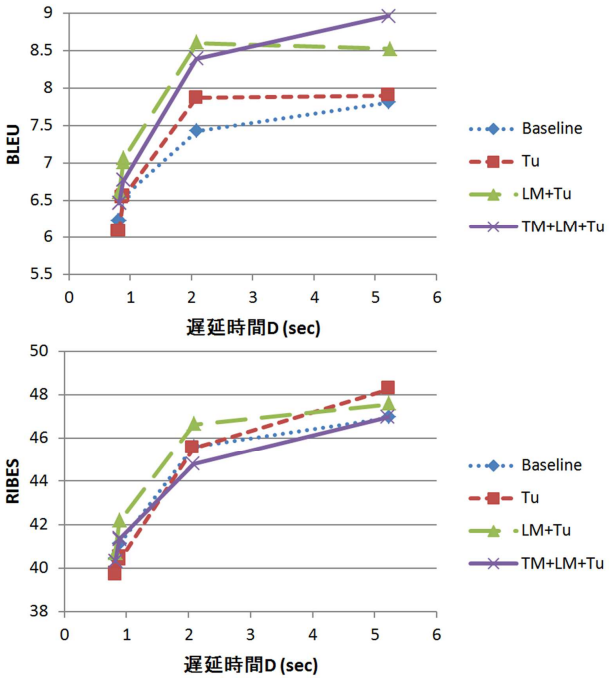

Fig. 2　同時通訳データと翻訳精度の実験結果

Table 2　テストセットにおける右確率を用いて分割された原言語側の翻訳単位数．右確率を学習したデータがそれぞれ TED の翻訳データのみ (W/O I)，及び TED の同時通訳データを追加したデータ (With I)

右確率	W/O I	With I
0.0	6150	6118
0.2	6140	6105
0.4	5155	5097
0.6	5123	5060
0.8	2070	1977
1.0	560	560

る．今回，チューニングは BLEU を最適化するようにパラメータを決定したため，RIBES が改善しなかったと考えられる．

Table 3 に翻訳例を記載する．TM+LM+Tu の翻訳文の長さは，Baseline に比べ短縮されており，正解文に近づいた．

5.4　同時通訳データと遅延時間に関する実験

次に，右確率の学習データに同時通訳データを使用し，遅延時間の短縮について調査した．右確率の学習データには，翻訳データのみ (W/O I) の場合と翻訳データに同時通訳データを追加した場合 (With I) を比較した．システムの学習データには，節の TM+LM+Tu を使用した．

実験結果を Fig 3 に示す．このグラフから，翻訳精度及び遅延時間は同時通訳データを追加した場合がそうでない場合と比較して，劣化した．この原因を考察するため，右確率を用いて分割した翻訳単位の数を調べた (Table 2)．全ての右確率における翻訳単位数が，並び替えモデルに同時通訳データを追加した場合が，追加しなかった場合に比べ減少した．つまり，

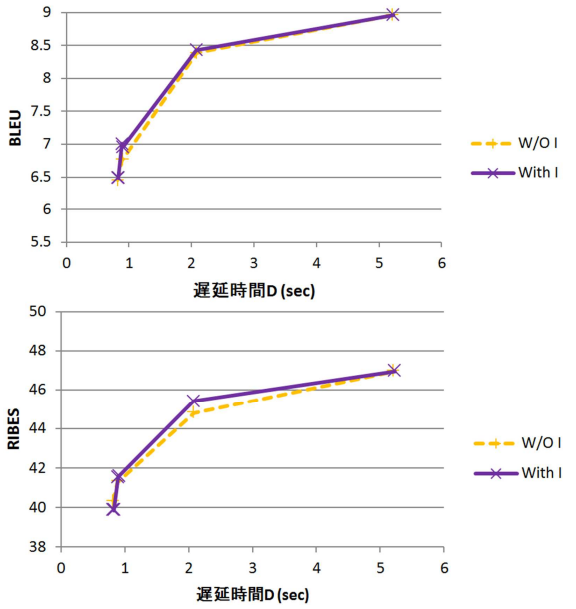

Fig. 3　右確率の学習データを変更した場合の実験結果

本実験の設定では，同時通訳データを右確率の学習に用いたが，大きな変化は見られなかった．

5.5　音声認識部を考慮した実験

次に，5.4 節の実験では，テストに人手で書き起こしたデータが使用されたが，本節では音声認識したデータを使用し再度実験を行った．テストデータの単語誤り率は 19.36%である．

実験結果を Fig 4 に示す．まず確認できることが，人手で書き起こされた結果 (Fig 2) の場合と比較すると翻訳精度が劣化している点である．これは音声認識の誤りが原因であると考えられる．しかし，性能の傾向は，人手書き起こされたデータを用いた場合と同様であることが分かる．

また，構築した同時通訳システムの性能と実際の通訳者の性能を比較するために，Fig 4 に通訳経験年数 4 年 (A ランク) と 1 年 (B ランク) の通訳者の翻訳精度と遅延時間をプロットした．BLEU の観点からは，構築したシステムの性能は通訳者に及ばなかった．しかし，RIBES と遅延時間の観点から評価したグラフを見ると，TM+LM+Tu は RIBES が 44.59 の翻訳結果を得るのに 2.06 秒要しており，B ランクも 2.17 秒要して RIBES が 45.59 の訳出をしている．RIBES は日本語と英語のような並び替えが起きやすい言語対の評価に適しているため，構築した同時通訳システムは B ランクの通訳者とほぼ同等の性能であると考えられる．

6　関連研究

同時通訳データを利用したシステムの構築に関する研究はいくつかなされている．

Ryu らは，構文ルールを利用した同時通訳システムを構築している [14]．この研究は，学習データに SIDB の同時通訳コーパス [15][16][17] を利用して同時通訳システムを構築している．しかし，ルールベー

Table 3　翻訳文の一例

	翻訳文
原言語文	the next slide i show you will be a rapid fast forward of what's happened over the last 25 years
同時通訳データの正解文	この 25 年間に何が起こったかというのを早送りで見せたいと思います
Baseline (右確率 1.0)	次のスライドをお見せしますが急速に進んで何が起こったのです過去 25 年間
TM+LM+Tu (右確率 1.0)	次のスライドをお見せしますがこの 25 年間に起こったのです

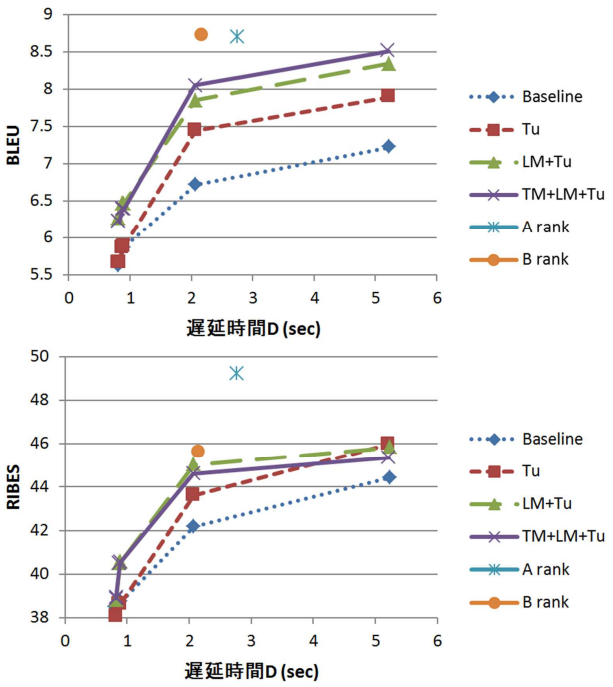

Fig. 4　音声認識部を考慮した実験結果

ス機械翻訳を用いているため，多言語に対応することが容易ではない．

著者らは，学習データに同時通訳データを利用した機械翻訳システムを構築している [4]．しかし，翻訳精度の観点からのみ機械翻訳システムの性能を評価しており，遅延時間の観点から性能の評価が行えていない．

7　おわりに

本稿は，同時通訳データを利用することによって，同時通訳システムを構築し，遅延時間と翻訳精度から評価を行った．実験の結果，同時通訳データを学習データに使用することで，人手で書き起こしたデータ，音声認識の認識結果共に性能が改善した．今後の課題としては，同時通訳システムの主観評価が考えられる．

謝辞　本研究の一部は，JSPS 科研費 24240032 の助成を受け実施したものである．

参考文献

[1] Roderick Jones. *Conference Interpreting Explained (Translation Practices Explained)*. St. Jerome Publishing, 2002.

[2] Srinivas Bangalore et al. Real-time incremental speech-to-speech translation of dialogs. In *Proc. NAACL12*, 2012.

[3] Tomoki Fujita et al. Simple, lexicalized choice of translation timing for simultaneous speech translation. In *Proc. 14th InterSpeech*, 2013.

[4] 清水 宏晃 他. 同時通訳データを利用した同時通訳用機械翻訳システムの構築. 情報処理学会 第 212 回自然言語処理研究会 (SIG-NL), 北海道, 7 2013.

[5] Arianna Bisazza, Nick Ruiz, Marcello Federico. Fill-up versus interpolation methods for phrase-based smt adaptation. In *Proc. IWSLT*, pp. 136–143, 2011.

[6] Xiaoyi Ma. Champollion: A robust parallel text sentence aligner. In *Proc. LREC*, 2006.

[7] Philipp Koehn et al. Moses: Open source toolkit for statistical machine translation. In *Proc. ACL*, pp. 177–180, Prague, Czech Republic, 2007.

[8] Graham Neubig, Yosuke Nakata, Shinsuke Mori. Pointwise prediction for robust, adaptable Japanese morphological analysis. In *Proc. ACL*, pp. 529–533, Portland, USA, June 2011.

[9] Franz Josef Och, Hermann Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, Vol. 29, No. 1, pp. 19–51, 2003.

[10] Andreas Stolcke. SRILM - an extensible language modeling toolkit. In *Proc. 7th International Conference on Speech and Language Processing (ICSLP)*, 2002.

[11] Franz Josef Och. Minimum error rate training in statistical machine translation. In *Proc. ACL*, 2003.

[12] Kishore Papineni et al. BLEU: a method for automatic evaluation of machine translation. In *Proc. ACL*, pp. 311–318, Philadelphia, USA, 2002.

[13] Hideki Isozaki et al. Automatic evaluation of translation quality for distant language pairs. In *Proc. EMNLP*, pp. 944–952, 2010.

[14] Koichiro Ryu et al. Incremental japanese spoken language generation in simultaneous machine interpretation. In *Proc. Asian Symposium on Natural Language Processing to Overcome language Barriers*, 2004.

[15] Yasuyuki Aizawa et al. Spoken language corpus for machine interpretation research. In *International Conference on Speech and Language Processing (ICSLP)*, pp. vol. III, page 398–401, 2000.

[16] Shigeki Matsubara et al. Bilingual spoken language corpus for simultaneous machine interpretation research. In *Proc. LREC*, pp. vol. I, page 153–159, 2002.

[17] Hitomi Toyama et al. Ciair simultaneous interpretation corpus. In *Proc. Oriental COCOSDA*, 2004.