End-to-End 音声翻訳のためのデータ拡張の検討

高木景矢
豊橋技術科学大学
takagi.keiya.sk@tut.jp

秋葉友良
豊橋技術科学大学
akiba.tomoyoshi.tk@tut.jp

塚田元
豊橋技術科学大学
tsukada.hajime.hl@tut.jp

1 はじめに

音声翻訳とはソース言語の音声をターゲット言語のテキストに翻訳するタスクである．このタスクを実現させるために，先行研究では音声認識モデルと機械翻訳モデルを連結させたカスケードモデルや，単一のモデルで直接翻訳する End-to-End モデルが採用されている [1]．End-to-End 音声翻訳モデルは，誤り伝搬の軽減と低遅延化のために近年注目されている．しかし，End-to-End 音声翻訳において，ソース音声とターゲット文の大規模なペアデータが必要になることが問題として挙げられる．そこで，近年，音声合成精度が人間の発話に近くなっている [2] ことから対訳データのソーステキストから音声合成を使用して疑似ペアデータを作成しデータ拡張をする研究が行われている [3, 4]．これらの研究では，非公開データを使用していたり，英語講演音声翻訳コーパスに対してドメイン外の MT データを用いたデータ拡張をし，性能をあげるために音声翻訳データで fine-tuning を行っている．最近，MuST-C と呼ばれる英語講演音声翻訳コーパスが公開された [5]．これは以前からある英語講演音声翻訳コーパスである IWSLT18 [6] よりもサイズが大きく，品質が良いとされている．本論文では，IWSLT18 の音声翻訳コーパスと MuST-C の対訳データから End-to-End 音声合成モデルで作成した合成音声-ターゲットテキストの拡張データを用いて End-to-End 音声翻訳モデルを学習することの有効性を調査する．

また，我々は，音声認識タスクで単一話者の音声合成を用いたデータ拡張において，通常用いられる音声特徴量よりも DNN-HMM (Deep Neural Network Hidden Markov model) のボトルネック特徴量を用いることで音声認識精度が改善されることを示した [7]．音声認識・翻訳はともに sequence-to-sequence の問題であることから，我々の以前の研究が音声翻訳に応用可能かも調査する．

2 End-to-End 音声翻訳と End-to-End 音声合成

End-to-End 音声翻訳とはソース音声系列とターゲットテキスト系列の組を学習データとし，その変換を直接学習する手法である．従来の音声認識モデルと機械翻訳モデルを連結させるカスケード方式よりもモデルがシンプルで，誤り伝搬の軽減と低遅延化のために近年注目されている．End-to-End 音声翻訳は Berard ら [1] が最初に可能性を示した．

End-to-End 音声合成とは，文字系列とその音声の組を学習データとし，その変換を直接学習する手法である．近年の End-to-End 音声合成研究では，人間の発話に近い精度を達成している [2]．本研究では Tacotron2 [8] を用いて合成音声を作成する．Tacotron2 はテキストからメルスペクトログラムを生成するエンコーダデコーダモデルである．メルスペクトログラムから音声波形に変換する機構はボコーダと呼ばれ，本研究では WaveGlow [9] を使用する．WaveGlow は Glow [10] と WaveNet [11] の見識から開発されたもので，自己回帰を必要とせず高速で効率的かつ高品質の合成音声を出力する．

3 提案手法

提案するデータ拡張法は我々の以前の研究 [7] を音声翻訳に応用したものである．初期の音声翻訳データに追加のテキスト-テキスト対訳データのソーステキストを音声合成し，音声翻訳用の音声-テキスト対訳データで拡張する．

End-to-End 音声翻訳では通常，log-mel フィルタバンク特徴量などの音響特徴量が入力として用いられるが，音声に近い特徴量では多様な情報を含んでおり，多数の話者の学習データが必要になる．そこで，提案手法では音声とその書き起こしのペアデータで事前学習した DNN-HMM 音響モデルの DNN から抽出したボトルネック特徴量を End-to-End 音声翻訳の入力特徴量とすることで単一話者の音声合成によるデータ拡張で音声翻訳性能を改善する (図 1)．

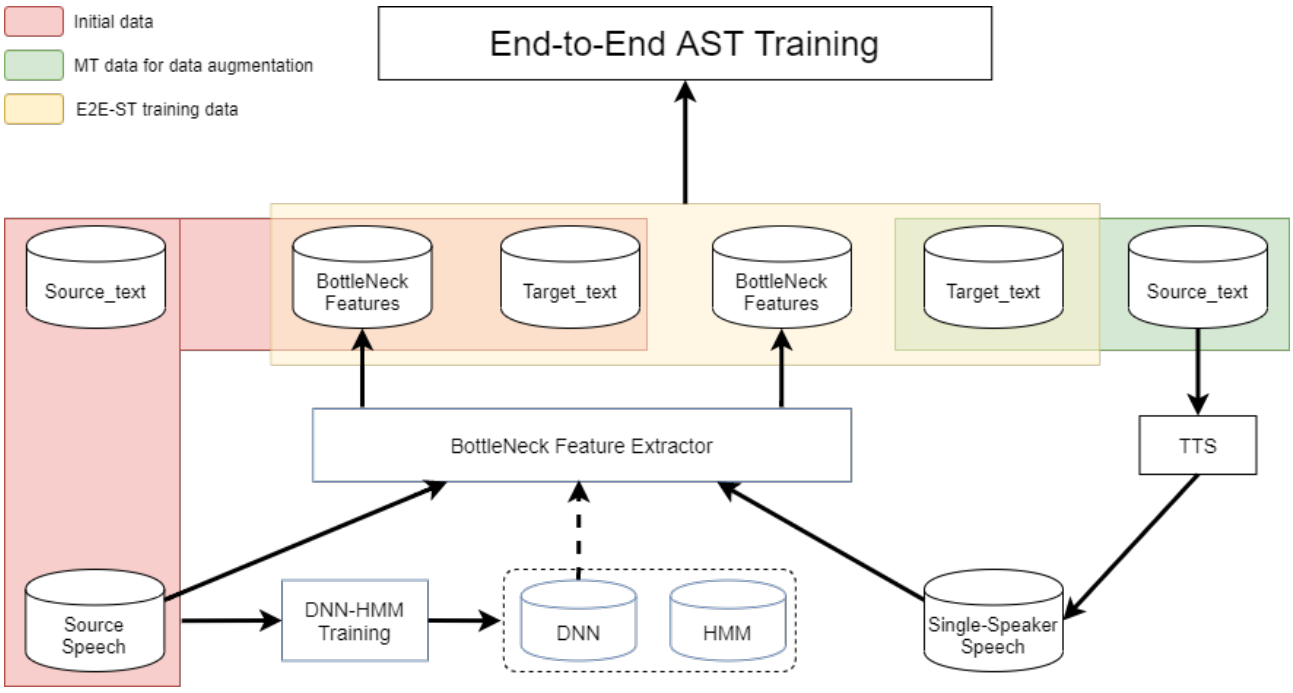

図 1　提案手法の概略図

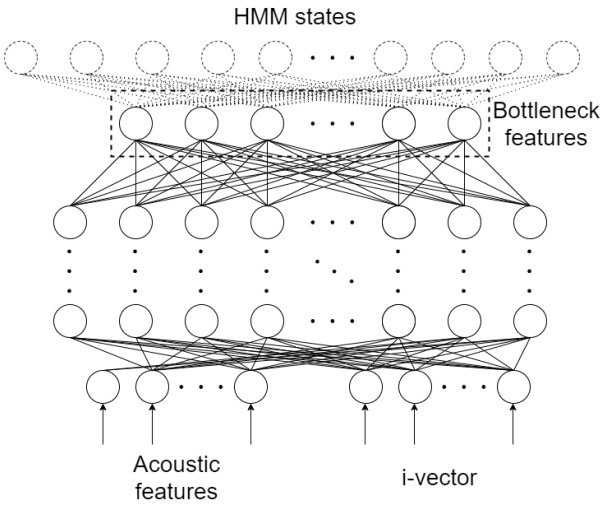

図 2　ボトルネック特徴量抽出器の概略図

ボトルネック特徴量の合成には，図 2 のように DNN-HMM の出力に近く比較的ノード数の少ない層を用いる．音素の事後確率は話者に依存しないので，DNN-HMM の中間特徴量は話者に依存しない表現として有用であると考える．DNN-HMM の学習は従来の手法と同様で，GMM-HMM (Gaussian Mixture Model Hidden Markov model) を使用してフレームごとに HMM の状態をアライメントし，その HMM の状態を予測するように DNN を学習する．

データ拡張では，機械翻訳のソーステキストデータから単一話者の音声合成モデルを用いてテキストと対応する音声を合成する．合成した音声に対し，ペアデータで学習しておいた DNN-HMM を用いてフレームごとにボトルネック特徴量を抽出する．抽出したボトルネック特徴量と元のターゲットテキストを対にして疑似ペアデータを作成する．元のペアデータに疑似ペアデータを加え End-to-End 音声翻訳モデルを学習する．

4　実験

音響特徴量と提案するボトルネック特徴量を用いて End-to-End 音声翻訳モデルを学習しその翻訳精度を比較する．

4.1　データセット

本実験では，コーパスとして IWSLT18 と MuST-C を使用した．翻訳タスクとしては英語-ドイツ語を選択した．IWSLT18 コーパスは 271 時間分の英語講演音声とその書き起こし，ドイツ語の翻訳が含まれている．先行研究 [12] に倣い，アライメント品質の低い発話を削除し約 137k の発話を使用した．MuST-C は英語講演音声とその書き起こしとドイツ語やフランス語を含む 8 言語の翻訳が含まれるコーパスである．そのうち，ドイツ語翻訳のある 408 時間分のデータを使用した．実験では，IWSLT18 を初期ペアデータとし，データ拡張用テキストデータとしては，MuST-C のテキストを使用した．ただし，MuST-C については 40 単語以内のものを使用し

表 1　音声翻訳データの統計情報

データセット	講演数	文数	発話時間
IWSLT18	1528	136617	209h
MuST-C	2003	207893	319h
dev2010	7	653	1.2h
tst2010	11	1656	2.9h
tst2013	16	1234	2.2h
tst2014	15	1405	2.5h
tst2015	12	1373	2.5h
tst-COMMON	27	2641	4.2h
tst-HE	12	600	1.2h

た．開発セットには IWSLT18 の dev2010 を使用し，テストセットには tst2010，tst2013，tst2014，tst2015 と MuST-C のテストセット tst-COMMON，tst-HE を用いた．

4.2　データの前処理

まず，テストセットに含まれる講演と学習データに含まれる講演が重複しないように学習データを削除した．また，学習データと開発セットについては 3000 フレームを超えるものや，400 文字を超える発話は削除した．各データの最終的な情報を表 1 に示す．

テキストデータについてはトークナイズと小文字化は Moses [1] のスクリプトを用いて行った．句読点などの非発話記号は削除し，文字レベルのボキャブラリを使用した．

4.3　ボトルネック特徴量抽出

ボトルネック特徴量を合成する DNN-HMM を学習するために Kaldi [13] を用いた．学習データは初期データである IWSLT18 を用いた．入力は 40 次元の MFCC と 100 次元の i-vector で出力は HMM の共有状態 (6056 次元) として学習を行った．出力に近く比較的次元の少ない層の出力 (レシピでいうと prefinal-xent の batchnor 層) をボトルネック特徴量 (256 次元) として合成した．

4.4　データ拡張

疑似ペアデータを作成するために単一話者の音声合成には Tacotron2 [8] と WaveGlow [9] を用いた．モデルは，公開されているモデルを使用した [2]．これらのモデルは，単一女性話者によって収録された 24 時間程度の英語音声コーパスである LJSpeech を学習データとしている．

4.5　End-to-End 音声翻訳

ESPnet [3] [14] のレシピに倣い音声データの特徴量抽出を行った．従来使用される音響特徴量として窓幅 25ms，フレームシフト 10ms で 80 次元の log-mel フィルタバンク特徴量と 3 次元の pitch 情報を得た．また，この特徴量は学習データの平均と標準偏差で平均化した．簡単のために学習データセットを次のように表記する．

IWSLT18
発話速度を 0.9，1.0，1.1 倍にしデータを拡張した IWSLT18 のソース音声とターゲットテキストのペアデータ．

TTS_MuST-C
MuST-C のターゲットテキストと対応するソーステキストの合成音声の疑似ペアデータ．

RAW_MuST-C
性能の上限を調査するための MuST-C のソース音声とターゲットテキストのペアデータ．

これらのデータの組み合わせで実験を行った．

モデルアーキテクチャは ESPnet に従ったもので注意機構を有する VGG-like のモデルである．エンコーダに 4 層の CNN と 1024 次元，5 層の BiLSTM，デコーダに 1024 次元，2 層の LSTM が採用されている．バッチサイズは 16 でオプティマイザは adadelta を使用し，15 Epoch の学習を行い，開発セットで最も Accuracy が良かったモデルを最終的なモデルとして選択した．

4.6　実験結果

音声翻訳結果を表 2，図 3 に示す．図 3 は表 2 を各コーパスのテストセットについて平均した数値をグラフ化したもので，左が fbank+pitch を入力特徴量とする音声翻訳結果で，右が BNF を入力特徴量とする音声翻訳結果である．評価指標には Moses の multi-bleu-detok.perl スクリプトを用いた case insensitive BLEU を使用した．

まず，TTS によるデータ拡張の効果を見るために，IWSLT18 と IWSLT18+TTS_MuST-C の行を比較する．どちらの特徴量を用いた場合でも，データ拡

1）https://github.com/moses-smt/mosesdecoder.git
2）https://github.com/NVIDIA/tacotron2
3）https://github.com/espnet/espnet

表 2　IWSLT18 と MuST-C のテストセットの音声翻訳結果 (BLEU)

データセット	入力特徴量	dev2010	tst2010	tst2013	tst2014	tst2015	tst-COMMON	tst-HE
IWSLT18	fbank+pitch	14.34	10.88	9.45	9.11	9.51	10.04	9.69
IWSLT18+TTS_MuST-C		17.71	16.19	15.01	14.01	14.67	16.56	15.68
IWSLT18+RAW_MuST-C		17.25	14.93	14.92	13.18	14.18	19.82	18.50
IWSLT18	BNF	14.48	12.21	10.54	10.58	10.85	11.79	11.57
IWSLT18+TTS_MuST-C		17.38	14.83	12.81	12.87	13.71	14.32	14.78
IWSLT18+RAW_MuST-C		17.76	14.65	13.56	12.35	13.89	16.90	16.35

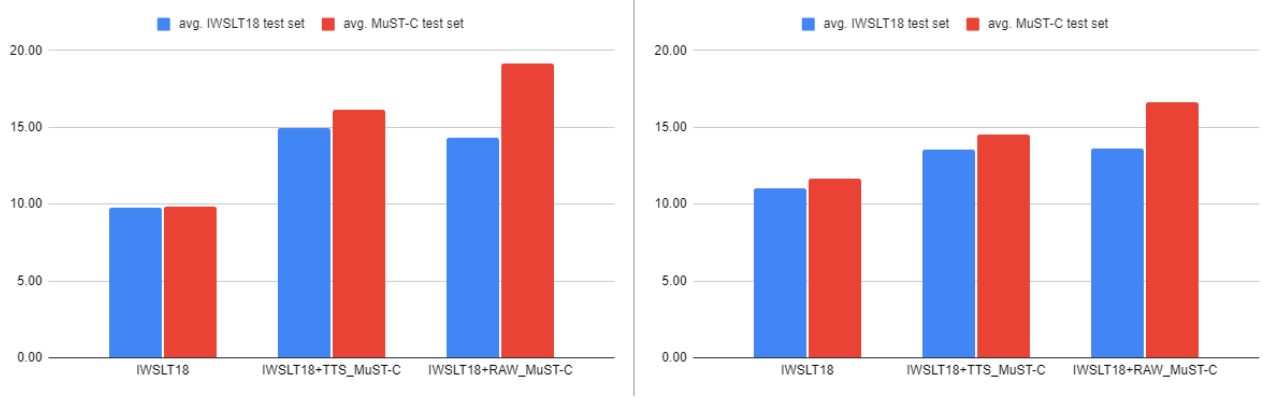

図 3　fbank+pitch を入力特徴量する IWSLT18 と MuST-C のテストセットの音声翻訳結果の平均値 (左)，BNF を入力特徴量する IWSLT18 と MuST-C のテストセットの音声翻訳結果の平均値 (右)

張 (+TTS_MuST-C) によって音声翻訳精度が改善されていることがわかり，シングルスピーカー TTS でデータ拡張する効果が確認できた．次に，その効果を合成ではない実際の音声との対訳でのデータ拡張 (IWSLT18+RAW_MuST-C) と比較してみる．MuST-C テストデータでは音声合成データ (TTS_MuST-C) は実際の音声 (RAW_MuST-C) の性能には至っていないが，IWSLT テストデータでは同等の効果が得られている．拡張データと評価データに同じコーパスを使った場合 (MuST-C テストデータ) に対し，それぞれが異なるコーパスとなる IWSLT テストデータでは，音声合成でも実音声と同じような効果となった可能性がある．

次に，2 つの特徴量 fbank+pitch と BNF の効果を比較する．学習データのサイズが小規模な場合 (IWSLT18 の行) を比較すると，BNF の方が高い翻訳性能となった．一方，学習データのサイズが増えた場合 (+TTS_MuST-C および+RAW_MuST-C の行) を比較すると，fbank+pitch の方が翻訳性能が高い．この理由として，BNF は音素表現に近い特徴量を持つがゆえに多様性が小さいため，データリソースが少ない場合は翻訳モデルの学習に有利であったと考えられる．データリソースが増えると，多様性の大きい特徴量 (fbank+pitch) でも学習が改善され，音声の表現能力の差から性能が逆転した可能性がある．

5　結論

本論文では，英語講演音声翻訳コーパスである IWSLT18 を初期データとして，MuST-C のソーステキストの音声合成とターゲットテキストで作成した疑似ペアデータを用いてデータ拡張を行った．その結果，翻訳性能が大幅に上昇することを確認した．IWSLT18 のテストセットに関しては，実際の音声との対訳データで拡張した場合と同等の効果が得られた．

また，以前の研究で有用であったボトルネック特徴量を音声翻訳に応用した．学習データのサイズが小さい場合は従来の音響特徴量を用いるよりも提案手法が翻訳性能を上回ることを示した．

今後の課題としては複数話者の音声合成モデルを用いたデータ拡張実験との比較と，End-to-End 音声翻訳モデルを Transformer ベースのモデルにすることが挙げられる．

謝辞

本研究は JSPS 科研費 19K11980 および 18H01062 の助成を受けた．

参考文献

[1] Alexandre Bérard, Olivier Pietquin, Christophe Servan, and Laurent Besacier. Listen and translate: A proof of concept for end-to-end speech-to-text translation. *arXiv preprint arXiv:1612.01744*, 2016.

[2] Jonathan Shen, Ruoming Pang, Ron J Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, Rj Skerrv-Ryan, et al. Natural TTS synthesis by conditioning wavenet on mel spectrogram predictions. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4779–4783. IEEE, 2018.

[3] Ye Jia, Melvin Johnson, Wolfgang Macherey, Ron J Weiss, Yuan Cao, Chung-Cheng Chiu, Naveen Ari, Stella Laurenzo, and Yonghui Wu. Leveraging weakly supervised data to improve end-to-end speech-to-text translation. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7180–7184. IEEE, 2019.

[4] Juan Pino, Liezl Puzon, Jiatao Gu, Xutai Ma, Arya D McCarthy, and Deepak Gopinath. Harnessing indirect training data for end-to-end automatic speech translation: Tricks of the trade. *arXiv*, pp. arXiv–1909, 2019.

[5] Mattia A. Di Gangi, Roldano Cattoni, Luisa Bentivogli, Matteo Negri, and Marco Turchi. MuST-C: a Multilingual Speech Translation Corpus. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2012–2017, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[6] Jan Niehues, Ronaldo Cattoni, Sebastian Stuker, Mauro Cettolo, Marco Turchi, and Marcello Federico. The IWSLT 2018 evaluation campaign. *In Proceedings of IWSLT*, 2018.

[7] 高木景矢, 秋葉友良, 塚田元. ボトルネック特徴量の合成に基づく音声認識のためのデータ拡張の検討. 日本音響学会 2020 年春季研究発表会 (ASJ), 3 2020.

[8] Jonathan Shen, et al. Natural TTS synthesis by conditioning wavenet on mel spectrogram predictions. *CoRR*, Vol. abs/1712.05884, , 2017.

[9] Ryan Prenger, et al. Waveglow: A flow-based generative network for speech synthesis. *CoRR*, Vol. abs/1811.00002, , 2018.

[10] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In *Advances in neural information processing systems*, pp. 10215–10224, 2018.

[11] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

[12] Hirofumi Inaguma, Kevin Duh, Tatsuya Kawahara, and Shinji Watanabe. Multilingual end-to-end speech translation. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 570–577. IEEE, 2019.

[13] Daniel Povey, et al. The kaldi speech recognition toolkit. In *In IEEE 2011 workshop*, 2011.

[14] Shinji Watanabe, et al. Espnet: End-to-end speech processing toolkit. *CoRR*, Vol. abs/1804.00015, , 2018.