人手書き起こしの知識を用いた音声認識誤りに頑健な機械翻訳

福田 りょう　須藤 克仁　中村 哲
奈良先端科学技術大学院大学
{fukuda.ryo.fo3, sudoh, s-nakamura}@is.naist.jp

1 はじめに

一般的な音声翻訳システムは, 発話をテキスト化する音声認識モデル (ASR) と, テキストを他言語へ翻訳する機械翻訳モデル (MT) を持つ. こうしたシステムにおいて, MT が入力として受け取る ASR 出力には音声認識誤りが含まれることがあり, 翻訳精度低下の原因となる.

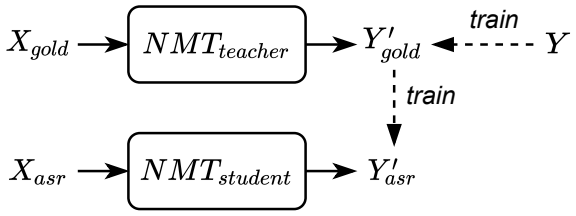
本研究では, 高精度かつ音声認識誤りに対して頑健な音声翻訳システムの実現を目的とし, 人手の書き起こしと ASR の出力を効果的に組み合わせた機械翻訳モデルの学習手法を提案する. 具体的には, 書き起こしを入力として学習した MT の出力を教師信号として, ASR 出力を入力とする MT を学習する. これは, 高品質な対訳で学習した教師モデルの知識を実環境用のモデルへ引き継ぐ知識蒸留であり, 高い翻訳精度を維持しつつ音声認識誤りに対して頑健になることを期待するものである.

実験では, 書き起こしと音声認識誤りを混合して学習させたモデルに対して, およそ 0.5pt の BLEU スコア向上を示した. また, ドメイン適応学習の手法である Fine-tuning との併用を検討し, より高い翻訳精度が得られることを確認した.

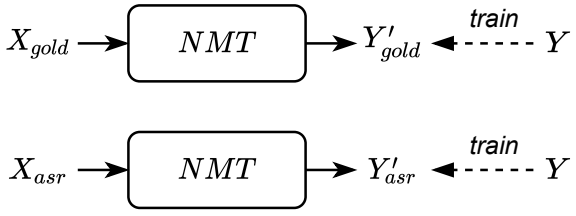
2 関連研究

音声翻訳システムにおける, ASR 出力の曖昧性や音声認識誤りを考慮した機械翻訳の研究がこれまで多く行われている. Sperber ら [1] は ASR のラティス構造を, Osamura ら [2] は ASR の出力分布をニューラル機械翻訳 (NMT) の入力として用いることで, ASR 出力の曖昧性を考慮する翻訳手法を提案した. Sperber ら [3], Xue ら [4] は, 対訳の原言語文に擬似的な音声認識誤りを含ませることで, 音声認識誤りに対する翻訳精度が向上することを示した.

知識蒸留 (Knowledge Distillation) [5, 6] は, 機械学習において教師モデルの出力を生徒モデルに模倣させる学習手法であり, よりパラメータの多い複雑なモデルから軽量なモデルへ, あるいは高機能で低速なモデルから高速なモデルへといった形で利用されてきた.



(a) Knowledge Distillation



(b) Fine-tuning

図 1 本稿の学習方式.

また, 低資源下における機械翻訳の学習手法としてドメイン適応 (Domain Adaptation) [7] が知られている. ドメイン適応は, 対訳データの少ない対象ドメインの翻訳学習に, 外部ドメインの対訳データや対象ドメインの単言語データを利用する手法である.

Di Gangi ら [8] は, 話し言葉機械翻訳において, 書き起こしと ASR 出力を併用したドメイン適応学習を行うことで, 一方のみを用いた場合と比較して翻訳精度が向上することを示した. 本研究では, 書き起こしと ASR 出力のより効果的な活用を目指し, 知識蒸留に基づく学習, 及びドメイン適応との併用を検討した.

3 提案手法

3.1 知識蒸留

知識蒸留を用いた学習の概略を図 1(a) に示す. まず教師モデル ($NMT_{teacher}$) として, 人手による書き起こし (X_{gold}) を入力とした機械翻訳を学習する.

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

表 1 Fisher データセットのデータ統計. 対訳文の数と, 前処理後のファイルのトークン数. Dev, Test データにおいては 4 通りの Disfluent translation, 2 通りの Fluent translation がそれぞれ用意されている.

	Sentences	Gold transcripts	ASR outputs	Disfluent translations	Fluent translations
Fisher/Train	138,720	1,810,271	1,540,782	1,846,992	1,460,117
Dev	3,977	50,741	41,787	50,538 / 50,765 / 50,485 / 51,141	37,143 / 37,132
Test	3,641	47,899	41,544	49,244 / 48,961 / 47,744 / 48,502	36,189 / 35,546

続いて, 生徒モデル ($NMT_{student}$) として ASR 出力 (X_{asr}) を入力とした機械翻訳を学習する. この時, 出力 (Y'_{asr}) が教師モデル (Y'_{gold}) と同一になるように交差エントロピー損失の最小化が行われる. 知識蒸留の手法として sequence-level knowledge distillation [9] を用いる. これは, 教師モデルの出力分布を学習する word-level knowledge distillation と対照的に, 教師モデルでビーム探索を行い, 決定した出力トークンの系列を学習する方式である. また, 教師と生徒のモデル構造は同一とする. つまり一般的な知識蒸留が「"強い"モデルの知識を"弱い"モデルへ蒸留する」ことに対し, ここでは「"強いデータ"で学習したモデルの知識を"弱いデータ"で学習するモデルへ蒸留する」ことを行う.

3.2 ドメイン適応

今回, ドメイン適応の典型的な手法である Multi-domain 学習 [7] と Fine-tuning [10] を検討した. Multi-domain 学習はドメイン外データとドメイン内データを混合して学習に用いる手法である. Fine-tuning は, ドメイン外データでモデルを事前学習後, ドメイン内データで追加学習を行う手法である. 図 1(b) に Fine-tuning による学習の概略を示す. 上段では, 人手による書き起こしを入力としてニューラル機械翻訳 (NMT) の事前学習を行い, 後段では, ASR 出力を入力として NMT を追加学習する.

4 実験

4.1 実験設定

4.1.1 データセット

実験には, IWSLT 2020 Conversational Speech Translation [11] の Fisher データセットを使用して, スペイン語から英語へのテキスト翻訳を学習する. これは, 160 時間のスペイン語会話音声とその書き起こし (Gold transcript) である LDC Fisher Spanish speech コーパス [12] に ASR 出力 (ASR output) と英語の翻訳テキスト (Disfluent translation) [13], フィラーや言い淀みなどを除去した流暢な英語の翻訳テキスト (Fluent translation) [14] を加えたマルチウェイ対訳データセットである. その中で, 今回は Gold transcript, ASR output を入力として, Fluent translation を出力として用いた. データ統計を表 1 に示す.

4.1.2 モデル

機械翻訳モデルは Fairseq (v0.6.2)[1] を用いて Transformer [15] を構築した. Transformer の構成およびハイパーパラメータは *transformer_base* [15] に準じる. Encoder, Decoder はそれぞれ 6 層とし, トークンの埋め込みベクトル, 各層の隠れ状態ベクトル, フィードフォーワードネットワークの次元数をそれぞれ 512, 512, 2048 とした. サブレイヤの dropout は 0.1 の確率で行い, Multi-head attention のヘッド数は 8 とした. 最適化アルゴリズムは Adam を使用し, そのパラメータを $\beta_1 = 0.9$, $\beta_2 = 0.997$ に設定の上, 学習率の初期値を 0.0007 として Vaswani ら [15] の方法で学習率を変化させた. ミニバッチのサイズは 4096 トークンとして, 8 個積み重ねて 1 回更新することを学習が収束するまで行った. モデルは 5000 回更新する毎に Dev セットで損失を計算し, 最終的に最も小さい値を獲得したモデルでテストを行った. またデータは, subword-nmt[2] を用いて Byte Pair Encoding (BPE) によるサブワード分割 [16] を行った. 語彙数はそれぞれ最大およそ 8000 に設定している.

実験では, ベースラインとして

- Single_{gold}: Gold transcript で学習したモデル
- Single_{asr}: ASR output で学習したモデル

を用意した. また, ドメイン適応手法として

- $\text{Multi}_{gold\&asr}$: Gold transcript と ASR output のデータを混合した Multi-domain 学習モデル
- $\text{FT}_{gold \to asr}$: Single_{gold} に対し ASR output で追加学習を行った Fine-tuning モデル

を作成した. また知識蒸留は ASR output を入力とし,

1) https://github.com/pytorch/fairseq
2) https://github.com/rsennrich/subword-nmt

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

表 2　BLEU スコアによる翻訳精度の比較. † はベースラインモデル Single_{asr}, ‡ は Fine-tuning モデル $\text{FT}_{gold\to asr}$ より有意に高いことを示す (いずれも $p < 0.05$).

System	Fisher/Test0		Fisher/Test1	
	ASR output	Gold transcript	ASR output	Gold transcript
Single_{gold}	17.45	**26.75**	16.98	**26.14**
Single_{asr}	17.49	17.62	16.87	17.15
$\text{Multi}_{gold\&asr}$	17.97†	25.99	17.27	25.57
$\text{FT}_{gold\to asr}$	18.31†	24.89	17.5†	24.52
KD_{gold}	18.48†	16.52	17.87†	16.24
$\text{KD}_{gold\&asr}$	16.59	16.12	13.1	13.05
$\text{FT} + \text{KD}_{gold}$	**18.76**†‡	25.24	**17.96**†‡	24.86

- KD_{gold}: Single_{gold} の出力を教師信号として学習したモデル
- $\text{KD}_{gold\&asr}$: $\text{Multi}_{gold\&asr}$ の出力を教師信号として学習したモデル
- $\text{FT} + \text{KD}_{gold}$: Single_{gold} に KD_{gold} の方式で追加学習を行う, Fine-tuning と知識蒸留を組み合わせたモデル

を作成した. Dev セットで検証する際には, ASR output の入力に対して行い, 出力を 2 通りの Fluent translation で評価した. Test セットによる評価時は, Gold transcript と ASR output を入力とし, 2 通りの Fluent translation (Fisher/Test0, Fisher/Test1) それぞれに対して BLEU スコアを測定した. また, ベースラインと提案手法間でブートストラップ再サンプリング方式による BLEU スコアの有意差検定 [17] を行った. 実装は util-scripts の paired-bootstrap.py[3] を使用し, 有意水準を 5%とした.

4.2　実験結果

実験結果を表 2 に示す. ベースライン同士を比較すると, Single_{gold} は Gold transcript に対して高い翻訳精度である一方, ASR output を入力した場合におよそ 9pt の精度低下がある. Single_{asr} は, 入力形式によって結果が大きく変化せず, いずれも Single_{gold} に Gold transcript を入力した場合を 9pt 程度下回った. 書き起こしを入力として学習した MT は, 高い翻訳能力を獲得し得るが, 音声認識誤りに対する頑健性が低い. 反対に, 音声認識誤りを含む入力を用いて学習した MT は, 音声認識誤りに頑健であるものの基本となる翻訳能力が低い. そのため実用の場面で ASR output を入力することを想定した場合, 両モデルはおよそ対等であると言える.

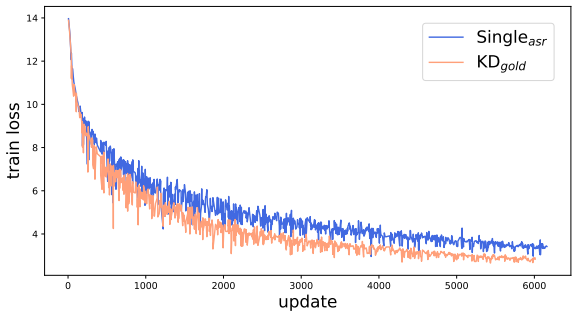


図 2　ベースラインモデル Single_{asr} と, 知識蒸留を行った提案手法 KD_{gold} の学習曲線の比較.

また Single_{gold} と比較して, ドメイン適応を行った 2 手法 ($\text{Multi}_{gold\&asr}$, $\text{FT}_{gold\to asr}$) は, Gold transcript に対する精度が 1~2pt 低下する代わりに ASR output に対する精度が 0.7~0.9pt 向上した. これは人手の書き起こしと ASR 出力を併用した学習が, 翻訳精度向上に効果的であることを示している.

4.2.1　知識蒸留の効果

KD_{gold} は, ASR output に対する翻訳精度がベースラインやドメイン適応のモデルより有意に高く, 知識蒸留の有用性を示している. Single_{asr} と比較した時, 学習の差は教師信号を人手による参照訳から Single_{gold} の出力に変更した点である. この変更で, ASR output に対しおよそ 1pt の向上が観測された. その理由として, MT を通すことで自然発話の多様性または複雑性が失われ, 学習の難しさが緩和されたことが考えられる. 実際に, 学習時の損失の変化を示した図 2 では, KD_{gold} がより低い位置で推移していることが見て取れる. しかし, 反対に Gold transcript に対しては 1pt の低下があった. 教師として与えた MT 出力と Test セットの人手による参照訳との間で

3)　https://github.com/neubig/util-scripts/blob/master/paired-bootstrap.py

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

表 3 ASR output に対するベースラインモデル Single_{asr} と, 知識蒸留と Fine-tuning を併用した提案手法 $\text{FT} + \text{KD}_{gold}$ の生成例. 上の例で, Single_{asr} は音声認識誤り (“sur”) を直訳 (“South”) したが, $\text{FT} + \text{KD}_{gold}$ はこれを無視した. また ASR output が記号や大文字を含まないことも翻訳精度低下の原因になりうる. 下の例で, 記号 (“¿”, “?”) を含まない疑問文に対し, Single_{asr} は平叙文を訳出したが, $\text{FT} + \text{KD}_{gold}$ では疑問符を回復した.

ASR output	en un sur super nuevo que salió
Gold transcript	uno super, super nuevo que salió
Fluent translation	One super new that came out
Single_{asr}	In the South, it came out
$\text{FT} + \text{KD}_{gold}$	In a super new one that came out
ASR output	y hace tiempo ya que está en esta cosa de llamar por teléfono
Gold transcript	¿Y hace tiempo que ya estás en ésta cosa? ¿De llamar por teléfono?
Fluent translation	And have you been in this thing of calling on the phone long time?
Single_{asr}	It’s been a long time since you’re calling on the phone
$\text{FT} + \text{KD}_{gold}$	How long have you been in this thing to call on the phone?

表 4 知識蒸留における教師モデルの出力の BLEU スコア.

System	Fisher/Train
Single_{gold}	48.03
$\text{Multi}_{gold\&asr}$	37.26

生じた分布の差異がこの原因として考えられる.

他方, $\text{KD}_{gold\&asr}$ は, 大きく精度が低下した. 知識蒸留を行ったそれぞれの教師モデル (Single_{gold}, $\text{Multi}_{gold\&asr}$) の Train データにおける BLEU スコアを表 4 に示す. Train データに対し, 両者には 10pt 近いスコアの差がある. $\text{Multi}_{gold\&asr}$ は Single_{gold} と比べて, ASR output に頑健であるが, Gold transcript の翻訳精度は低い. このことから, 生徒モデルの教師となるデータにはある程度の高品質さが求められることが分かる.

4.2.2 知識蒸留とドメイン適応の併用

知識蒸留と Fine-tuning を組み合わせた $\text{FT}+\text{KD}_{gold}$ は, ASR output に対し最も高い翻訳精度を達成し, ベースラインと比較して 1pt 程度の向上が確認された. 生成例の比較を表 3 に示す. また知識蒸留のみ行ったモデルと比較して, Gold transcript に対する翻訳精度が大きく向上しており, Single_{gold} の基本翻訳性能の高さを引き継ぎながら音声認識誤りに対する頑健性を獲得できたと言える. Fine-tuning はパラメータを, 知識蒸留は出力の知識を, それぞれ事前学習モデルや教師モデルから引き継ぐ. このように両者は異なる情報を継承するため, 併用することでそれぞれを単体で用いたモデルより高い評価値を得た.

5 おわりに

本研究では, 書き起こしテキストと ASR 出力の併用による機械翻訳モデルの学習手法を検討した. 実験では, 知識蒸留とドメイン適応が音声認識誤りへの頑健性獲得のために有効であることを示し, 更にこれらを併用することでより高い翻訳精度を達成できることを確認した.

今後の課題として, まず ASR 出力の精度による提案手法の有効性の変化を検証する. 今回用いた ASR 出力は, HMM に基づく音声認識モデルによるものであり, Fisher/Test に対し WER 36.5pt [13] と比較的多くの誤りを含んでいる. より音声認識誤りの少ない ASR 出力に対しても本研究の手法が有効であるかどうかを確認したい. また, 学習手法についても広範に検証を行っていく必要がある. 例えばドメイン適応の一手法である Multi-domain 学習にも, 文頭にドメインタグを付加して区別させる方法やドメイン間のデータ数を揃える upsampling や downsampling など数多くの変種が存在する. 知識蒸留では, word-level knowledge distillation を用いてより豊富な知識を継承することで更なる向上が期待できる.

謝辞

本研究の一部は JSPS 科研費 JP17H06101 の助成を受けたものである.

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

参考文献

[1] Matthias Sperber, Graham Neubig, Jan Niehues, and Alex Waibel. Neural lattice-to-sequence models for uncertain inputs. *arXiv preprint arXiv:1704.00559*, 2017.

[2] Kaho Osamura, Takatomo Kano, Sakriani Sakti, Katsuhito Sudoh, and Satoshi Nakamura. Using spoken word posterior features in neural machine translation. *architecture*, Vol. 21, p. 22, 2018.

[3] Matthias Sperber, Jan Niehues, and Alex Waibel. Toward robust neural machine translation for noisy input sequences. In *International Workshop on Spoken Language Translation (IWSLT)*, 2017.

[4] Haiyang Xue, Yang Feng, Shuhao Gu, and Wei Chen. Robust neural machine translation with asr errors. In *Proceedings of the First Workshop on Automatic Simultaneous Translation*, pp. 15–23, 2020.

[5] Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 535–541, 2006.

[6] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[7] Chenhui Chu, Raj Dabre, and Sadao Kurohashi. An empirical comparison of domain adaptation methods for neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 385–391, 2017.

[8] Mattia Antonino Di Gangi, Robert Enyedi, Alessandra Brusadin, and Marcello Federico. Robust neural machine translation for clean and noisy speech transcripts. *arXiv preprint arXiv:1910.10238*, 2019.

[9] Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. *arXiv preprint arXiv:1606.07947*, 2016.

[10] Rico Sennrich, Barry Haddow, and Alexandra Birch. Improving neural machine translation models with monolingual data. *arXiv preprint arXiv:1511.06709*, 2015.

[11] Ebrahim Ansari, Nguyen Bach, Ondřej Bojar, Roldano Cattoni, Fahim Dalvi, Nadir Durrani, Marcello Federico, Christian Federmann, Jiatao Gu, Fei Huang, et al. Findings of the iwslt 2020 evaluation campaign. In *Proceedings of the 17th International Conference on Spoken Language Translation*, pp. 1–34, 2020.

[12] David Graff, Shudong Huang, Ingrid Cartagena, Kevin Walker, and Christopher Cieri. Fisher spanish speech (ldc2010s01). *Web Download. Philadelphia: Linguistic Data Consortium*, 2010.

[13] Matt Post, Gaurav Kumar, Adam Lopez, Damianos Karakos, Chris Callison-Burch, and Sanjeev Khudanpur. Improved speech-to-text translation with the fisher and callhome spanish–english speech translation corpus. In *Proc. IWSLT*, 2013.

[14] Elizabeth Salesky, Matthias Sperber, and Alex Waibel. Fluent translations from disfluent speech in end-to-end speech translation. *arXiv preprint arXiv:1906.00556*, 2019.

[15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

[16] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*, 2015.

[17] Philipp Koehn. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pp. 388–395, 2004.

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).