Transformer モデルにおける知識蒸留の適用

松永大希
岐阜大学大学院自然科学技術研究科
z4525080@edu.gifu-u.ac.jp

松本 忠博
岐阜大学工学部
tad@gifu-u.ac.jp

1 はじめに

近年，ニューラル機械翻訳の精度は飛躍的に向上している．一方で，精度が向上するにつれてモデルのサイズが増加することが知られている．この問題を解決するために近年では，モデルの軽量化の研究が広く行われている．知識蒸留は，モデルの軽量化手法の一つである．本研究では，知識蒸留を Transformer[1] に適用させ，機械翻訳モデルの軽量化を試みた．教師モデルに対してパラメータ数を約 4 割削減した生徒モデルに対して本手法を適用したところ，教師モデルよりも高い翻訳精度を得られることを確認した．

2 関連研究

2.1 モデルの軽量化

機械翻訳の軽量化手法として，枝刈り [2]，量子化 [3]，知識蒸留 [4] がある．これらの軽量化手法は互いに独立であり，併用することができる．

枝刈り ニューラルネットワークの重要ではない結合を取り除く．

量子化 訓練後に重みの浮動小数点をより少ないビット数に変換する．

知識蒸留 訓練済みの大きなモデルの知識を，軽量な別のモデルの学習に利用する．次項で詳しく説明する．

2.2 知識蒸留

知識蒸留は，Hinton ら [4] が提案した手法である．高精度で大きなモデルである教師モデルを作成し，その知識をより軽量なモデルである生徒モデルに継承する．知識はソフトラベルの誤差を最小化することで継承する．ソフトラベルは，教師モデルが出力する確率分布のことである．一方，訓練データの正解ラベルをハードラベルと呼ぶ．ソフトラベルとハードラベルの関係を図 1 に，知識蒸留の全体像を図 2 に示す．Hinton らの研究ではソフトラベルの損失関数として，教師及び生徒モデルの温度付きソフトマックス出力の交差エントロピーを用いているが，Zhang ら [5] は Kullback-Leibler divergence（KL Divergence）を用いている．KL Divergence は以下の数式で表される．

$$D_{KL}(q||p) = \sum_i p_i \log \frac{p_i}{q_i} \tag{1}$$

また，Romero ら [6] は，教師データよりも層の数を増やしつつ，中間層のパラメータを削減した生徒モデルを提案し，全体的なパラメータの削減を実現した．また，Romero らは提案した生徒モデルとソフトラベルとの比較のみではなく，教師モデルの中間層を比較することによって，過学習を防ぎ，モデルの精度を向上させた．この中間層をヒント層と呼ぶ．知識蒸留は自然言語処理の分野でも活用されている．Sanh ら [7] は，BERT[8] に知識蒸留を適用した．また，Kim ら [9] は機械翻訳のための LSTM モデルに知識蒸留を用いることを試みた．本研究では機械翻訳等に用いられる Transformer モデルを対象とした，知識蒸留手法を検討する．また，教師及び生徒モデルの Encoder と Decoder の出力層をヒント層として利用した知識蒸留も検討する．

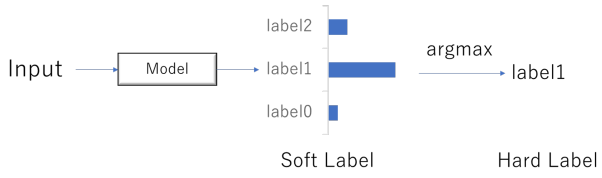

図 1 ソフトラベルとハードラベル

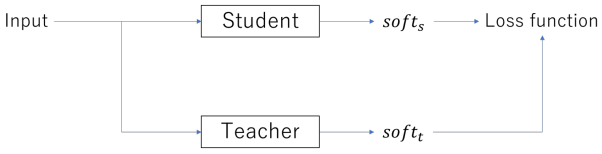

図 2 知識蒸留

3 提案手法

本研究では Transformer モデルを対象に，ヒント層を利用した合計 4 段階の学習手法を提案する．手法

は，Encoder 層の学習，Decoder 層の学習，ソフトラベルの学習，ハードラベルの学習で構成される．

Encoder 層の学習　Encoder の最終出力層の時点で誤差を計算し，学習する．誤差関数は平均二乗誤差を用いる．

Decoder 層の学習　Decoder の最終出力層の時点で誤差を計算し，学習する．誤差関数は Encoder 層と同様に，平均最小二乗誤差を用いる．

ソフトラベルの学習　本研究では，教師モデルが全結合層の後に出力した確率分布をソフトラベルとする．教師モデルのソフトラベルと教師モデルの出力との誤差を計算し，これを最小化する．

ハードラベルの学習　最後に生徒モデルと，実際のコーパスのターゲット文を比較し，誤差を計算する．この時の誤差関数は，教師モデル作成時に使用した誤差関数と同様のものを用いる．

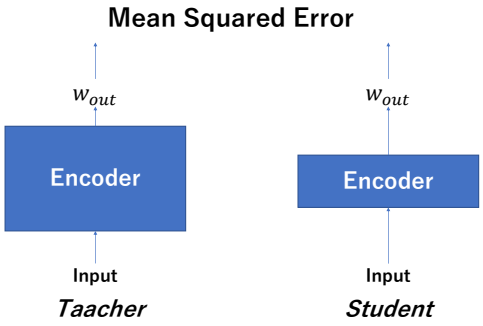

図 3　Encoder 層の学習

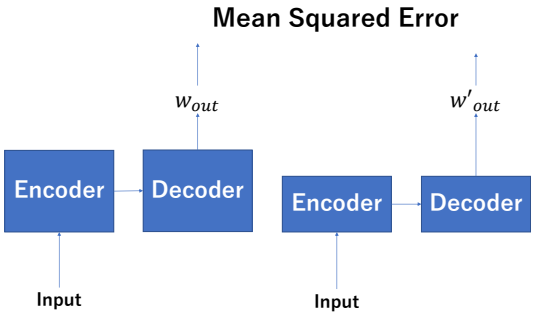

図 4　Decoder 層の学習

4　実験

4.1　実験設定

実験設定を以下に示す．なお，全ての実験で PyTorch を用いた[1)]．また，学習，推論ともに計算資源として RTX2070super を 1 枚用いた．

1）　https://github.com/harvardnlp/annotated-transformer を改良したものを用いた

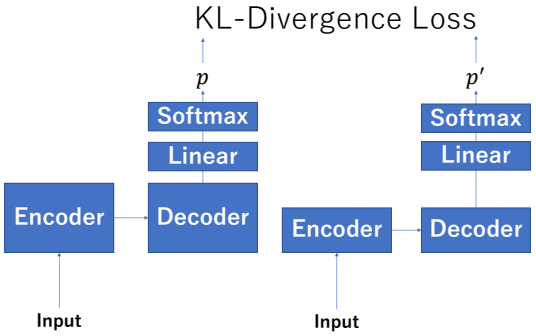

図 5　ソフトラベルの学習

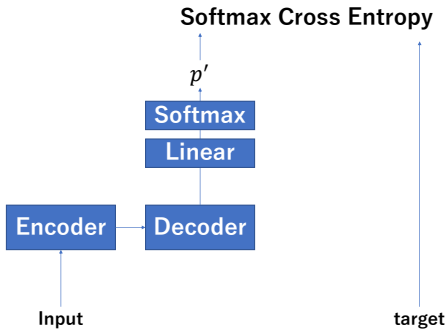

図 6　ハードラベルの学習

対訳コーパス　実験用データとしてアジア学術論文抜粋コーパス（ASPEC-JE）[10] を使用した．データセットのうち訓練データは先頭の 100 万文対のみを使った．コーパスの文対数の詳細を表 1 に示す．単語分割の際に Sentencepiece[11] を用いた．Sentencepiece は訓練データの英日文を連結させて，語彙サイズ 16000 で学習したものを用いた．また，最大系列長を 50 とした．

表 1　実験用対訳コーパスの文対数（ASPEC-JE より）

訓練データ	100M
評価データ	1,790
テストデータ	1,812

モデルと学習　実験用の Transformer モデルは教師モデルを 1 つ，生徒モデルを 5 つ構築した．学習時間によって精度は上昇している可能性を懸念して，すべてのモデルの学習回数を合計 30 回に統一した．表 2 にモデルの詳細と各ステップの学習エポック数を示す．すべてのモデルにおいて隠れ層の次元は 512 次元でフィードフォワード層の次元は 2048 とした．また，単語埋め込み層のサイズは 512 次元，ドロップアウトを 0.1 とした．最適化手法として Adam[12] を用いた．ウォームアップステップは 2000 とした．教師モデルの損失関数は，Label Smoothing Entoropy を用い

た．誤差関数は，hint layer は平均最小二乗誤差関数，ソフトラベルの誤差関数は KL Divergence，教師モデルの誤差関数及び，生徒モデルのハードラベルの誤差関数は Cross Entoropy を用いた．この際，$\varepsilon = 0.1$ の Label Smoothing を行った．

表 2　モデルと学習エポック数

層数	-	Encoder	Decoder	soft	hard
6	teacher	-	-	-	30
3	baseline	0	0	0	30
	soft + hard	0	0	15	15
	hint + soft + hard	3	3	12	12
	soft	0	0	30	0
	hint + soft	3	3	24	0

推論　テストデータの翻訳の際，ビーム幅 5 のビームサーチを用いた．評価指標として BLEU[13] と RIBES[14] を用いた．評価は一度文を生文に戻した後，MeCab を用いて再分割してから行った．BLEU の計算は `multi-bleu.perl`[2]，RIBES の計算は `RIBES.py`[3] を用いた．

4.2　実験結果と考察

全ての実験において教師モデルと生徒モデルの 3 層，6 層におけるパラメータ数は，表 3 のようになった．

表 3　パラメータ数

	3 層	6 層
英日	40,260,507	62,329,755
日英	37,352,823	59,422,071

また，実験結果は表 4（英→日），表 5（日→英）のようになった．

表 4　実験結果 英日

	英→日		
	BLEU	RIBES	推論時間
teacher	36.77	82.38	741
baseline	36.62	82.09	571
soft	36.76	82.29	567
hint + soft	36.61	82.57	570
soft + hard	**39.40**	**83.81**	573
hint + soft + hard	38.97	83.65	**560**

実験より，Transformer の層を 6 層から 3 層に減らすことによって，パラメータ数がそれぞれ 35%，37%、推論時間が 23% 程度削減されることを確認した．また，同じ 30 エポックでも学習の比率によって結果が異なることを確認した．ソフトラベル，ハードラベルのみの学習だけの場合よりも，双方の学習を行ったモデルの方が高精度であった．ただし，ヒント層の学習の効果は本研究では見られなかった．これは，ヒント層の誤差の計算方法や学習回数に問題があると考えられ，さらなる改善が必要である．

表 5　実験結果 日英

	日→英		
	BLEU	RIBES	推論時間
teacher	26.57	76.15	745
baseline	25.89	75.78	571
soft	26.91	76.64	**565**
hint + soft	26.82	76.02	567
soft + hard	**27.14**	**76.30**	**574**
hint + soft + hard	27.00	76.27	576

5　おわりに

本研究では，ヒント層を利用した機械翻訳の知識蒸留手法を提案した．実験の結果，ハードラベルの学習が生徒モデルの精度向上に有効であることが分かった．今後は，ヒント層を利用した手法をさらに詳しく検討する必要がある．例えば，ヒント層の学習エポック数をすべて 3 回と設定したが，今後はその数を増減させた場合の効果の変化も検証する必要がある．また，Encoder および Decoder の最終出力だけではなく中間の出力の誤差も計算する必要もある．さらに，層数だけでなく，各層の出力次元数を減らしたモデルでの実験も行う必要がある．

参考文献

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, Vol. 30, pp. 5998–6008, 2017.

[2] Song Han, Jeff Pool, John Tran, and William Dally. Learning both weights and connections for efficient neural network. *Advances in neural information processing systems*, Vol. 28, pp. 1135–1143, 2015.

[3] Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew Howard, Hartwig Adam, and Dmitry Kalenichenko. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2704–2713, 2018.

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

[5] Ying Zhang, Tao Xiang, Timothy M Hospedales, and Huchuan Lu. Deep mutual learning. In *Proceedings of the*

2）https://github.com/moses-smt/mosesdecoder/blob/master/scripts/generic/multi-bleu.perl

3）https://github.com/nttcslab-nlp/RIBES/blob/master/RIBES.py

IEEE Conference on Computer Vision and Pattern Recognition, pp. 4320–4328, 2018.

[6] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014.

[7] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

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

[10] Toshiaki Nakazawa, Manabu Yaguchi, Kiyotaka Uchimoto, Masao Uchiyama, Eiichiro Sumita, Sadao Kurohashi, and Hitoshi Isahara. ASPEC: Asian scientific paper excerpt corpus. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2016)*, pp. 2204–2208, Portorož, Slovenia, may 2016. European Language Resources Association (ELRA).

[11] Taku Kudo and John Richardson. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*, 2018.

[12] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[13] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pp. 311–318, 2002.

[14] Hideki Isozaki, Tsutomu Hirao, Kevin Duh, Katsuhito Sudoh, and Hajime Tsukada. Automatic evaluation of translation quality for distant language pairs. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pp. 944–952, Cambridge, MA, October 2010. Association for Computational Linguistics.