マルチヘッドニューラル N-gram による自己注意機構の代替

Mengsay Loem　高瀬 翔 *　金子 正弘　岡崎 直観
東京工業大学情報理工学院
{mengsay.loem, masahiro.kaneko}[at]nlp.c.titech.ac.jp,
sho.takase[at]linecorp.com, okazaki[at]c.titech.ac.jp

概要

Transformer の優れた性能は，入力系列内の全てのトークン間の依存関係を考慮する自己注意によるものであると考えられている．しかし，全ての依存関係を考慮する必要があるかは疑問が残る．本研究では，全トークン間の依存関係を考えるのではなく，各トークンの周辺のトークンのみを用いたニューラル N-gram モデルに着目し，マルチヘッド機構による改良を施したマルチヘッドニューラル N-gram モデル（multiNN）を提案する．系列変換タスクの実験を通して，提案手法は Transformer に匹敵する性能を示すことを報告する．また，様々な分析の結果から，マルチヘッドニューラル N-gram は自己注意とは異なる優位性があり，両者を組み合せることにより，Transformer を上回る性能を達成することも報告する．

1　はじめに

Vaswani ら [1] が Transformer を提案して以来，Transformer とその亜種のアーキテクチャは，機械翻訳，自動要約，画像認識，音声認識などの様々なタスクに適用されてきた [1, 2, 3, 4]．Transformer の主要なモジュールは自己注意（Self-attention）である．自己注意は，系列の各位置の入力を処理する際，全ての位置間の依存関係に基づいて出力を計算する．具体的には，入力系列の特徴表現 $x_1, ..., x_L$ が与えられたとき，位置 t における出力は x_t と $x_1, ..., x_L$ 間の注意重みによる重み付き和により計算される．例えば，入力文 *we focus on neighbor tokens* に対し，*on* の第 1 層の出力を計算するとき，図 1 (a) のように他の全ての位置の入力を用いる．

しかし，各位置において全ての入力との依存関係を考慮する必要があるのだろうか？　実際，自己注意は処理しているトークンの周辺のトークンのみに

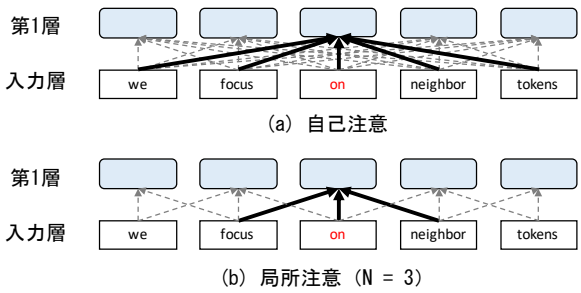


図 1　自己注意と局所注意の違い．自己注意はすべての入力の特徴表現を用いて次の層の特徴表現を計算するが，局所注意は周辺 N 個（この図では $N = 3$）の特徴表現のみを用いる．

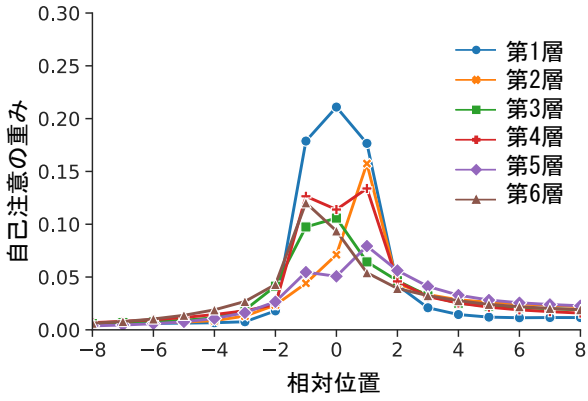
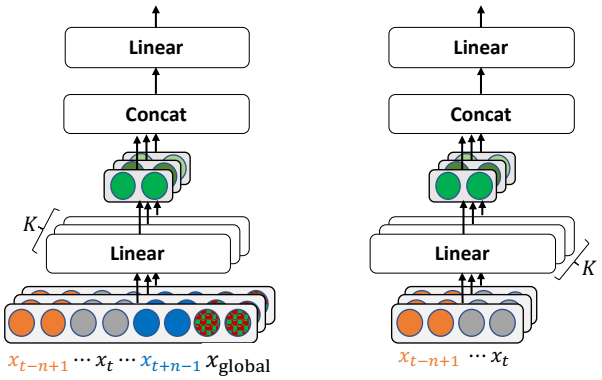


図 2　WMT 英独データセットの検証データにおける各層の自己注意の重み分布を処理している位置からの相対位置でプロットしたもの．WMT データセットで学習した 6 層の Transformer のエンコーダ側を使用した．

大きな注意重みを割り当てる傾向があることが経験的に知られている．例として，図 2 に機械翻訳タスクで用いられる系列変換モデルのエンコーダ側の各層の自己注意の注意重み分布を示した．この図から，着目している位置の周辺の数トークンに大きな重みが割り当てられていることが分かる．また，近年の研究では，自己注意で行われる入力間の依存関係の計算において，処理している位置の周囲の N 個のトークンのみに制限した局所注意の成功が報告さ

* 現在の所属は LINE 株式会社である．



（a）エンコーダ側のmultiNN　　（b）デコーダ側のmultiNN

図 3　エンコーダ側とデコーダ側におけるマルチヘッドニューラル N-gram のアーキテクチャ

れている [5, 6]．図 1 (b) に，$N = 3$ の場合の局所注意の例を示した．

ある位置の周辺の入力に着目する手法として，N-gram ベースのモデルがよく用いられている．Sun ら [7] は，残差結合 [8] などの最新の手法を導入することにより，ニューラル N-gram 言語モデル [9] の性能を著しく向上させた．Sun らの手法は，従来のニューラル N-gram モデルより優れているが，Transformer を上回ることはできなかった．それでは，N-gram ベースのアプローチで自己注意よりも優れた性能を示すモデルを作ることは本当に不可能なのであろうか？

本研究では，Transformer [1] の自己注意で用いられているマルチヘッド機構を導入することにより，ニューラル N-gram モデル [7] を改良する手法を提案する．この改良手法をマルチヘッドニューラル N-gram (multiNN: Multi-head Neural N-gram) と呼ぶ．この改良手法の性能を調べるため，Transformer の自己注意を提案手法の multiNN に置き換え，既存の Transformer との性能比較を行う．機械翻訳（4 節），自動要約（付録 A），音声認識（付録 B）を含む様々な系列変換タスクにおいて実験を行ったところ，提案手法の multiNN は自己注意と同等かそれ以上の性能を達成することが示された．また，multiNN に関する分析を通して，既存の Transformer の代替となるアーキテクチャを模索するために有用な知見が得られた（5 節）．

2　マルチヘッド自己注意

マルチヘッド自己注意は，広く使われている Transformer のコア・モジュールである．マルチヘッド自己注意は，入力系列の全体に対する注意関数で入力間の依存関係を捉える．また，マルチヘッド自己注意は，単一の注意関数を用いるのではなく，異なる線形写像で変換された入力の特徴表現に対し，複数の注意関数を適用する．具体的には，長さ L の D 次元の入力の特徴表現の列を $x_1, ..., x_L \in \mathbb{R}^D$ に対し，K 個のヘッドからなる注意機構は位置 t における出力 $z_t \in \mathbb{R}^D$ を次式で計算する（$[\cdot;\cdot]$ はベクトルの連結，$k \in \{1, \ldots, K\}$ である）．

$$z_t = [h_t^{(1)}; \ldots; h_t^{(K)}]W \tag{1}$$

$$h_t^{(k)} = \sum_{i=1}^{L} \alpha_{ti}^{(k)} x_i^{(k)} \tag{2}$$

$$x_i^{(k)} = x_i W_v^{(k)} \tag{3}$$

ここで，$W \in \mathbb{R}^{Kd \times D}$, $W_v^{(k)} \in \mathbb{R}^{D \times d}$ は線形写像の重み行列である（d は各ヘッドに写像された特徴表現 $x_i^{(k)}$ の次元数であり，通常は $d = D/K$ である）．また，ヘッド k における注意重み $\alpha_{ti}^{(k)} \in \mathbb{R}$ は入力の位置 t と i の間のスケーリングされた内積である [1]．式 (2) は，線形写像された入力表現 $x_1^{(k)}, \ldots, x_L^{(k)}$ に対し，注意関数を適用して各ヘッドの出力 $h_t^{(k)}$ を計算する．そして，式 (1) で，全ヘッドの出力 $h_t^{(1)}, \ldots, h_t^{(K)}$ を連結し，線形写像 W を適用することにより，マルチヘッド自己注意の出力 z_t を得る．

3　マルチヘッドニューラル N-gram

本研究では，ニューラル N-gram 言語モデルのアーキテクチャを踏襲しつつ，Transformer のマルチヘッド自己注意と同様にマルチヘッド機構を導入し，ニューラル N-gram モデルを改良する．マルチヘッドニューラル N-gram（multiNN）は，入力系列の各位置において，その位置のトークンの特徴表現とその周囲の N 個のトークンの特徴表現を連結したものを入力として受け取る．連結された特徴表現は，図 3 のように複数の線形写像を独立に適用する．具体的には，D 次元の特徴表現 $x_1, ..., x_L$ の系列が与えられた際，エンコーダ側で K 個のヘッドを持つ multiNN は，位置 t の出力の特徴表現 z_t を，以下

の式で計算する．

$$z_t = [h_t^{(1)}; \ldots; h_t^{(K)}]W \tag{4}$$

$$h_t^{(k)} = \mathrm{ReLU}([x_{t-n+1}^{(k)}; \ldots; x_{t+n-1}^{(k)}]W^{(k)}) \tag{5}$$

$$x_t^{(k)} = x_t W_x^{(k)} \tag{6}$$

ここで，$W \in \mathbb{R}^{Kd \times D}, W^{(k)} \in \mathbb{R}^{(2n-1)d \times D}$ と $W_x^{(k)} \in \mathbb{R}^{D \times d}$ は線形写像の重み行列である．multiNN の各ヘッド k で使われる入力表現は，式 (6) の異なる重み行列 $W_x^{(k)}$ で線形写像される．式 (5) で n 個の周辺の入力表現を連結し，線形写像 $W^{(k)}$ と活性化関数 ReLU を適用することで各ヘッドにおける位置 t の出力 $h_t^{(k)}$ を計算する．最後に，式 (4) で全 K 個のヘッドの出力 $h_t^{(1)}, \ldots, h_t^{(K)}$ を連結し，線形写像 W を施すことで，位置 t における multiNN の出力 z_t を計算する．

multiNN では n トークン以上離れた距離にある依存関係を考慮することができない．この弱点を緩和するため，入力系列を各ヘッド k に写像した後の全特徴表現に最大値プーリングを適用し，グローバルな特徴表現

$$x_{\mathrm{global}}^{(k)} = \max(x_1^{(k)}, \ldots, x_L^{(k)}) \tag{7}$$

を計算したのち，式 (5) を次式で置き換える選択肢も検討する．

$$h_t^{(k)} = \mathrm{ReLU}([x_{t-n+1}^{(k)}; \ldots; x_{t+n-1}^{(k)}; x_{\mathrm{global}}^{(k)}]W^{(k)}) \tag{8}$$

このとき，重み行列 $W^{(k)}$ のサイズを $W^{(k)} \in \mathbb{R}^{2nd \times D}$ に変更する．

なお，系列変換モデルのデコーダ側に multiNN を採用するときは，マスク付き自己注意のアイディアと同様に，着目している位置 t の左側の特徴表現 $x_{t-n+1}^{(k)}, \ldots, x_t^{(k)}$ のみを用いることとし，右側（将来）の特徴表現は参照しないこととする．

4 実験

4.1 設定

multiNN の性能を評価するために，WMT の英語—ドイツ語と IWSLT ドイツ語—英語の 2 つの機械翻訳ベンチマークで実験を行った．WMT では，Vaswani ら [1] の実験と同様に 450 万文のペアを含む WMT データセットを使用した．検証セットとテストセットとして，newstest2013 と newstest2014 をそれぞれ使用した．IWSLT では，Cettolo ら [10] と同様に，16 万文ペアが含まれるデータセットを使用した．各データセットの語彙の構築には，ソース側とターゲット側で語彙を共有する Byte Pairing Encoding [11] を用いた．語彙のサイズは WMT で 32K，IWSLT で 10K に設定した．ベースライン手法として，Transformer-base [1] を採用し，マルチヘッド自己注意を multiNN に置き換えた場合の性能を比較した．また，その他のベースライン手法として，提案手法の multiNN と同様に周囲の N 表現のみを使用した局所注意 [5, 6] とも性能の比較を行った．

表 1 機械翻訳タスクの BLEU スコア

モデル	IWSLT	WMT
自己注意	35.34 ± 0.10	**27.20** ± 0.09
局所注意	34.77 ± 0.08	26.71 ± 0.15
multiNN	**35.49** ± 0.08	27.15 ± 0.09

表 2 WMT の検証セットにおけるベースライン手法の性能および multiNN モデルのアブレーション結果

モデル	パラメータ数	BLEU
自己注意	61M	26.02
局所注意	61M	25.50
Sun ら [7]	62M	25.41
multiNN	62M	26.00
− マルチヘッド	62M	25.33
− グローバル	62M	25.80

4.2 結果

機械翻訳タスクの結果を表 1 に示した．各手法のランダムシードを変化させ，3 回の実行結果の平均スコアと分散を報告した．WMT と IWSL の両方のデータセットでは，multiNN が自己注意と同等の BLEU スコアを達成した．この結果から，multiNN は，すべての入力表現の位置依存関係を考慮することなく，注意機構に基づく手法と同等の性能が達成する能力があることが推測される．この結果は，multiNN は単純なアーキテクチャを採用し，扱える文脈の範囲を近傍のトークンに限定したにも関わらず，注意機構に匹敵する能力を有することを示唆している．また，multiNN は両方のデータセットで局所注意を 0.40 ポイント以上上回る性能を示した．

5 分析

5.1 各構成要素の貢献

ここでは，ニューラル N-gram ベースのモデルに関する知見を得るために，multiNN における各構成要素の貢献度合いを調査した．この実験では，全て

のモデルがほぼ同じパラメータ数になるようにして，モデル間の性能を比較した．

まず，マルチヘッド機構の効果について議論したい．表 2 の「multiNN」と「− マルチヘッド」の比較から分かるように，マルチヘッド機構を用いない N-gram モデルでは，BLEU スコアが 0.67 ポイントも低下した．このことから，ニューラル N-gram ベースの手法の性能向上において，本稿で提案したマルチヘッド機構が効果的であることが示唆される．

続いて，エンコーダ側のグローバル特徴表現の効果について議論したい．表 2「multiNN」と「− グローバル」の比較から，multiNN からグローバル特徴表現を取り除くと，モデルの性能は低下するものの，その低下は僅かであった．したがって，グローバル特徴表現も multiNN モデルの性能向上に貢献しているものの，マルチヘッド機構の方がより効果が大きいと推察される．

5.2 multiNN と自己注意との組合せ

Sun ら [7] はニューラル N グラムモデルと自己注意を組み合わせた場合の実験結果を報告していた．本稿でもそれに倣い，multiNN と自己注意を組合せたモデルでの実験結果を報告する．multiNN と自己注意は，入力間の異なる種類の依存関係を捉えると考えられるため，エンコーダとデコーダの両方で性能の向上が期待できる．本研究では，各層で両者を組み合わせた場合と，エンコーダ側およびデコーダ側で両者を切り替えて用いた場合の二種類の実験を行った．

図 4 に，自己注意を持つ 6 層の Transformer の下位の層から，自己注意をグローバル特徴表現を用いない multiNN に置き換えたときの，WMT データセットの検証セットにおける性能を示した．図 4 に示すように，下位から 3 層までに multiNN を用いることにより，既存の Transformer に比べて BLEU スコアを最大 0.50 ポイント改善できることが分かった．ところが，この改善効果は multiNN をさらに上の層まで置き換えてしまうと，減少してしまった．この結果から，Transformer の下位の層において狭い文脈に着目するように誘導することが重要であると考えられる．

WMT の検証セットにおいて，エンコーダ側とデコーダ側で multiNN と自己注意を切り替えた場合の実験結果を表 3 に示した．デコーダ側の自己注意を multiNN に置き換えると，元々の Transformer よりも

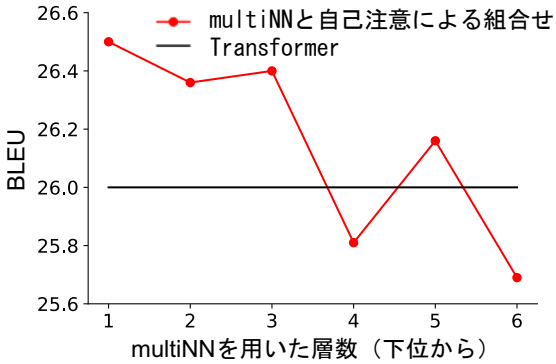


図 4 multiNN と自己注意の層別組合わせ．

表 3 エンコーダ側とデコーダ側における multiNN と自己注意の組合せ

エンコーダ側	デコーダ側	BLEU
自己注意	multiNN	26.46
自己注意	自己注意	26.02
multiNN	multiNN	26.00
multiNN	自己注意	25.95

高い性能を示すことが分かった．一方で，エンコーダ側の自己注意を multiNN に置き換えると，元々の Transformer よりもわずかに性能が低下する．以上の結果から，この機械翻訳タスクにおいてはエンコーダ側で長距離の依存を考慮できるようにしておき，デコーダ側では短距離の依存関係を考慮するように誘導する方が有利であることが示唆される．このように，multiNN は自己注意とは異なる効果を持っており，これらを併用することでタスクの性能を改善できる可能性がある．

6 おわりに

本研究では，入力の全体ではなく周辺の特徴表現のみを用いたニューラル N-gram ベースモデルが自己注意の代替になり得るかという問いを探求した．その問いを検証するため，ニューラル N-gram モデルを改良し，マルチヘッド機構を導入した multiNN を提案した．文脈の範囲を高々前後 n トークンに限定したにも関わらず，系列変換タスクにおいて，multiNN が自己注意や局所注意などの既存手法と同等かそれ以上の性能を示すことを実証した．また，multiNN と自己注意を組み合わせることより，既存の Transformer を上回る性能を発揮し得ることも報告した．今後の課題として，深いモデルや事前学習での性能を検証することで，multiNN のスケーラビリティを評価することを予定する．

謝辞　本研究成果は，国立研究開発法人情報通信研究機構（NICT）の委託研究「自動翻訳の精度向上のためのマルチモーダル情報の外部制御可能なモデリングの研究開発」（課題 225）により得られたものです.

参考文献

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, L ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In **Advances in Neural Information Processing Systems**, 2017.

[2] Sho Takase and Naoaki Okazaki. Positional encoding to control output sequence length. In **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, pp. 3999–4004, 2019.

[3] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In **International Conference on Learning Representations**, 2021.

[4] Linhao Dong, Shuang Xu, and Bo Xu. Speech-transformer: A no-recurrence sequence-to-sequence model for speech recognition. In **2018 IEEE International Conference on Acoustics, Speech and Signal Processing**, pp. 5884–5888, 2018.

[5] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. **CoRR**, Vol. abs/1904.10509, , 2019.

[6] Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer. **CoRR**, Vol. abs/2004.05150, , 2020.

[7] Simeng Sun and Mohit Iyyer. Revisiting simple neural probabilistic language models. In **Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, pp. 5181–5188, Online, 2021.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In **2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**, pp. 770–778, 2016.

[9] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. A neural probabilistic language model. **Journal of Machine Learning Research**, Vol. 3, p. 1137–1155, 2003.

[10] Mauro Cettolo, Jan Niehues, Sebastian Stüker, Luisa Bentivogli, and Marcello Federico. Report on the 11th IWSLT evaluation campaign. In **Proceedings of the 11th International Workshop on Spoken Language Translation: Evaluation Campaign**, pp. 2–17, 2014.

[11] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In **Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics**, pp. 1715–1725, 2016.

[12] Alexander M. Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. In **Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing**, pp. 379–389, 2015.

[13] Abigail See, Peter J. Liu, and Christopher D. Manning. Get to the point: Summarization with pointer-generator networks. In **Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics**, pp. 1073–1083, 2017.

[14] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: An asr corpus based on public domain audio books. In **2015 IEEE International Conference on Acoustics, Speech and Signal Processing**, pp. 5206–5210, 2015.

[15] Taku Kudo and John Richardson. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In **Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations**, pp. 66–71, 2018.

表 4　見出し生成タスクにおける ROUGE-1, 2, L（R-1, R-2, R-L）の F1 スコア

手法	パラメータ数	R-1	R-2	R-L
自己注意	61M	**37.80**	18.91	34.91
局所注意	61M	34.72	17.34	31.75
multiNN	62M	37.71	**19.02**	**35.17**

表 5　文書要約タスクにおける ROUGE-1, 2, L（R-1, R-2, R-L）の F1 スコア

手法	パラメータ数	R-1	R-2	R-L
自己注意	61M	**39.15**	17.21	**36.43**
局所注意	61M	34.27	14.57	32.24
multiNN	62M	39.04	**17.38**	36.01

A　自動要約タスクの実験

本研究では，見出し生成と文書要約の二つの自動要約タスクについても実験を行った．見出し生成は文レベルの自動要約タスクであり，ニュース記事の先頭の文から対応する見出しを生成するタスクである．このタスクの実験では，Gigaword データセット [12] を使用した．文書要約タスクは，文書レベルの要約タスクであり，モデルは原文書から複数の文からなる要約を生成する．このタスクの実験では，CNN/DailyMail データセット [13] を用いた．また，両タスクにおいて，ソース側とターゲット側で語彙を共有した Byte Pairing Encoding [11]（語彙サイズは 32K）を用いた．モデルのサイズは機械翻訳の実験と同等になるように設定した．

見出し生成タスクと文書要約タスクの結果を表 4 と表 5 に示す．見出し生成タスク，文書要約タスクのいずれにおいても，multiNN は自己注意に匹敵する性能を示した．また，局所注意と比較すると，提案手法の multiNN は大幅な性能向上を達成した．これらの知見は，機械翻訳タスクにおけるものと同様である．

B　音声認識タスクの実験

テキストの系列変換タスクの他に，LibriSpeech データセット [14] を用い，自動音声認識タスクの実験を行った．LibriSpeech には，約 960 時間に及ぶオーディオブックの音声が収録されている．このデータセットには，話者の単語誤り率に基づき，クリーンとその他の 2 種類のデータプールが存在する．実験では，両方のプールを使用した．デコーダ側の語彙は SentencePiece [15] を用い，語彙サイズは 10K とした．

表 6　LibraSpeech の検証セットにおける単語の誤り率

手法	パラメータ数	クリーン	その他
自己注意	52M	3.58	8.86
局所注意	52M	5.91	10.02
multiNN	52M	**3.27**	**8.83**

表 7　LibraSpeech のテストセットにおける単語の誤り率

手法	パラメータ数	クリーン	その他
自己注意	52M	4.06	**8.67**
局所注意	52M	6.43	10.46
multiNN	52M	**3.52**	8.70

LibriSpeech データセットの検証セットとテストセットにおける自動音声認識タスクにおける単語の誤り率を表 6 と表 7 に示した．提案手法の multiNN は，ほぼ同じモデルサイズで自己注意と同程度の性能を発揮した．また，multiNN の単語誤り率は局所注意よりも 1 ポイント以上の低かった．この実験結果も，機械翻訳タスクと自動要約タスクの実験結果と整合している．