# ニューラル系列変換のための Transformer の注意機構を活用した外部記憶融合

庵 愛，増村 亮，牧島 直輝，田中 智大，高島 瑛彦，折橋 翔太

日本電信電話株式会社，NTT メディアインテリジェンス研究所
mana.ihori.kx@hco.ntt.co.jp

## 1 はじめに

機械翻訳やキャプション生成，音声認識などの言語生成タスクでは，性能の高い Transformer [1] を用いることが主流となっている [2–9]．Transformer などのニューラル系列変換モデルを学習するためには，大量のパラレルデータが必要となる．そのため，テキストのみのデータを大量に保持していても，その対となるデータがなければ，学習に用いることはできない．また，パラレルデータの作成はコストが大きく，学習に十分な量のパラレルデータを用意することは難しい．

テキストのみのデータを活用するために，大量のテキストのみのデータから学習された外部言語モデルを参照すべき外部記憶として利用する方法が提案されている [10–12]．この方法は，外部言語モデルをニューラル系列変換モデルに統合することで実現され，ニューラル系列変換モデルの性能の改善に役立つことが確認されている．しかし，従来の外部言語モデルの統合方法は，recurrent neural network (RNN) に基づくニューラル系列変換モデルを対象としており，Transformer を対象としているわけではない．

Transformer と RNN に基づくニューラル系列変換モデルの異なる点として，multi-hop 注意機構 [13] を採用している点がある．multi-hop 注意機構では，デコーダの層数分，ソースターゲット注意を繰り返す．このように，Transformer ではソースターゲット注意を複数回繰り返すことによって，ソースが持つ系列変換に必要な情報をより柔軟に抽出していると考えられる．

以上より，本論文では Transformer の構造を活用した外部言語モデルの統合方法について提案する．Transformer が multi-hop 注意機構を採用する特徴的な構造であることを考慮し，外部言語モデルに対しても multi-hop 注意機構を適用する．ここでは，Transformer のデコーダの内部情報をキーとし，言語モデルに対するソースターゲット注意を多段に行うことで，言語モデルから系列変換に必要な情報を取捨選択することを期待している．評価実験には，話し言葉・書き言葉変換と方言変換の 2 種類のテキストスタイル変換タスクを採用する．各タスクにおいて，提案手法を用いた Transformer に対する外部言語モデルの統合が性能を向上させることを示す．

## 2 関連研究

ニューラル系列変換モデルに対する最も単純な外部言語モデルの統合方法として，系列変換モデルと言語モデルを対数線形補完する方法がある [10, 14, 15]．この方法は，shallow fusion と呼ばれる．しかし，shallow fusion は系列変換モデルと言語モデルを別々に学習しているため，系列変換モデルの学習中に外部言語モデルの情報を取り入れることができない．そこで，系列変換モデルの学習中に言語モデルの出力を統合する，deep fusion [11] や cold fusion [12] などの方法が提案されている．これらの方法では，系列変換モデルの学習にパラレルデータの情報だけでなく，テキストのみの情報も用いることが可能となった．図 1 に，テキストスタイル変換をタスクとし，Transformer に cold fusion を適用した場合のモデル構造を示す．しかし，これらの方法は，RNN に基づくニューラル系列変換モデルに適用することを前提として提案されたものであり，Transformer の構造を明示的に活かせていない．

## 3 提案手法

本論文では，Transformer の構造を明示的に活用した外部言語モデルの統合方法を提案する．ここで，提案手法は，cold fusion を拡張した方法となる．し



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

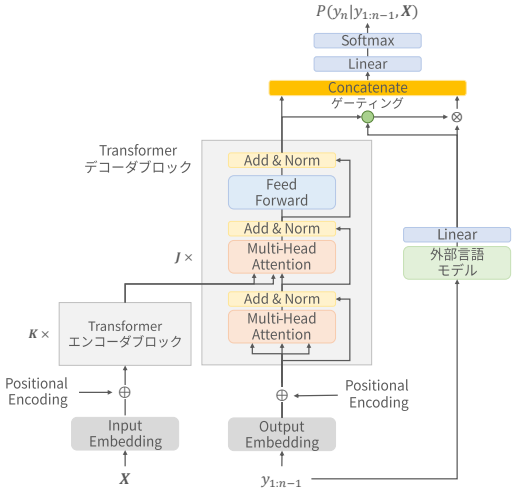


**図 1**　cold fusion を用いた Transformer

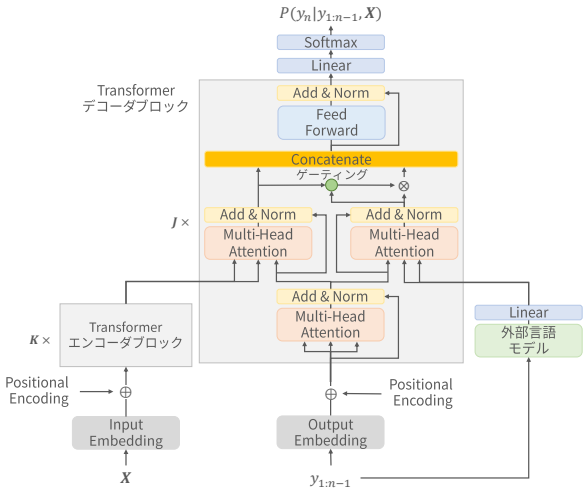


**図 2**　提案手法を用いた Transformer

かし，cold fusion が外部言語モデルの持つ知識を一度のみ使うのに対して，提案手法では，multi-hop 注意機構を用いることで外部言語モデルの持つ知識を繰り返し使用する．そのため，提案手法は，cold fusion よりも Transformer のデコーダが持つ multi-hop な構造を明示的に活かし，言語モデルの持つ知識を柔軟に抽出することが期待される．

本論文では，テキストスタイル変換をタスクとして，入力のトークン系列を $\boldsymbol{X} = \{x_1, \cdots, x_M\}$，出力のトークン系列を $\boldsymbol{Y} = \{y_1, \cdots, y_N\}$ と表す．ここで，$M$ は入力の，$N$ は出力のトークン数を表す．テキストスタイル変換では，変換前のトークン系列 $\boldsymbol{X}$ が与えられた時，変換後のトークン系列 $\boldsymbol{Y}$ の事後確率を式 (1) によって導出する．

$$P(\boldsymbol{Y}|\boldsymbol{X}; \boldsymbol{\Theta}) = \prod_{n=1}^{N} P(y_n|y_{1:n-1}, \boldsymbol{X}; \boldsymbol{\Theta}) \tag{1}$$

ここで，$y_{1:n-1} = \{y_1, \cdots, y_{n-1}\}$ であり，$\boldsymbol{\Theta} = \{\theta_{\text{enc}}, \theta_{\text{dec}}, \theta_{\text{lm}}\}$ はモデルパラメータを表す．$\theta_{\text{enc}}, \theta_{\text{dec}}$ は Transformer のエンコーダとデコーダ，$\theta_{\text{lm}}$ は外部言語モデルにおけるパラメータを表す．$P(y_n|y_{1:n-1}, \boldsymbol{X}; \boldsymbol{\Theta})$ は Transformer のエンコーダ，提案手法を用いて外部言語モデルを統合したデコーダによって算出される．図 2 に，テキストスタイル変換をタスクとし，Transformer に提案手法を適用した場合のモデル構造を示す．

## 3.1　エンコーダ

エンコーダでは，$K$ 層の Transformer エンコーダブロックを用いて，入力系列 $\boldsymbol{X}$ を隠れ状態ベクトル $\boldsymbol{S}^{(K)}$ に変換する．まず，最初の Transformer エンコーダブロックの入力 $\boldsymbol{S}^{(0)} = \{s_1^{(0)}, \cdots, s_M^{(0)}\}$ は，式 (2) によって算出される．

$$\boldsymbol{s}_m^{(0)} = \texttt{AddPositionalEncoding}(\boldsymbol{x}_m) \tag{2}$$

$$\boldsymbol{x}_m = \texttt{Embedding}(x_m; \theta_{\text{enc}}) \tag{3}$$

ここで，$\texttt{AddPositionalEncoding}(\cdot)$ は，位置エンコーディングの関数を，$\texttt{Embedding}(\cdot)$ は，トークンを連続ベクトルに変換する関数を表す．次に，$k$ 層目の隠れ状態ベクトルである $\boldsymbol{S}^{(k)}$ は，下層の出力 $\boldsymbol{S}^{(k-1)}$ を入力として式 (4) によって算出される．

$$\boldsymbol{S}^{(k)} = \texttt{TransformerEncBlock}(\boldsymbol{S}^{(k-1)}; \theta_{\text{enc}}) \tag{4}$$

ここで，$\texttt{TransformerEncBlock}(\cdot)$ は，Transformer エンコーダブロックを表しており，複数ヘッドの自己注意層と位置単位の順伝播ネットワーク [1] によって構成される．

## 3.2　提案手法を用いたデコーダ

提案手法を用いたデコーダでは，$J$ 層の Transformer デコーダブロックを用いて算出された隠れ状態ベクトル $\boldsymbol{u}_{n-1}^{(J)}$ から，トークン $y_n$ の事後確率を式 (5) を用いて算出する．

$$P(y_n|y_{1:n-1}, \boldsymbol{X}; \boldsymbol{\Theta}) = \texttt{softmax}(\boldsymbol{u}_{n-1}^{(J)}; \theta_{\text{dec}}) \tag{5}$$

ここで，$\texttt{softmax}(\cdot)$ は，活性化関数 softmax を用いた線形変換を表す．

Transformer デコーダブロックの入力の隠れ状態ベクトルである $\boldsymbol{u}_{n-1}^{(0)}$ は式 (6) によって算出される．

$$\boldsymbol{u}_{n-1}^{(0)} = \texttt{AddPositionalEncoding}(\boldsymbol{y}_{n-1}) \tag{6}$$

$$\boldsymbol{y}_{n-1} = \texttt{Embedding}(y_{n-1}; \theta_{\text{dec}}) \tag{7}$$



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

$j$ 番目の隠れ状態ベクトルである $\boldsymbol{u}_{n-1}^{(j)}$ は，入力系列の情報を持つ隠れ状態ベクトル $\boldsymbol{c}_{n-1}^{(j)}$ と，外部言語モデルの情報を持つ隠れ状態ベクトル $\boldsymbol{b}_{n-1}^{(j)}$ から算出される．まず，$\boldsymbol{c}_{n-1}^{(j)}$ は，下層の出力 $\boldsymbol{u}_1^{(j-1)}, \cdots, \boldsymbol{u}_{n-1}^{(j-1)}$ と，エンコーダの出力 $\boldsymbol{S}^{(k)}$ を入力として以下のように算出される．

$$\bar{\boldsymbol{v}}_{n-1}^{(j)} = \mathtt{SourceTarget}(\boldsymbol{u}_{1:n-1}^{(j-1)}, \boldsymbol{u}_{n-1}^{(j-1)}; \theta_{\mathrm{dec}}) \tag{8}$$

$$\boldsymbol{v}_{n-1}^{(j)} = \mathtt{LayerNorm}(\boldsymbol{u}_{n-1}^{(j-1)} + \bar{\boldsymbol{v}}_{n-1}^{(j)}) \tag{9}$$

$$\bar{\boldsymbol{c}}_{n-1}^{(j)} = \mathtt{SourceTarget}(\boldsymbol{S}^{(K)}, \boldsymbol{v}_{n-1}^{(j)}; \theta_{\mathrm{dec}}) \tag{10}$$

$$\boldsymbol{c}_{n-1}^{(j)} = \mathtt{LayerNorm}(\boldsymbol{u}_{n-1}^{(j)} + \bar{\boldsymbol{c}}_{n-1}^{(j)}) \tag{11}$$

ここで，$\mathtt{SourceTarget}(\cdot)$ は，複数ヘッドのソースターゲット注意層を，$\mathtt{LayerNorm}(\cdot)$ は，layer normalization を表す．次に，$\boldsymbol{b}_{n-1}^{(j)}$ は，隠れ状態ベクトル $\boldsymbol{v}_{n-1}^{(j)}$ と，外部言語モデルの出力を入力として以下のように算出される．

$$\boldsymbol{l}_{n-1}^{\mathrm{LM}} = \mathtt{LanguageModel}(y_{1:n-1}; \theta_{\mathrm{dec}}) \tag{12}$$

$$\boldsymbol{h}_{n-1}^{\mathrm{LM}} = \mathtt{linear}(\boldsymbol{l}_{n-1}^{\mathrm{LM}}; \theta_{\mathrm{dec}}) \tag{13}$$

$$\bar{\boldsymbol{b}}_{n-1}^{(j)} = \mathtt{SourceTarget}(\boldsymbol{h}_{1:n-1}^{\mathrm{LM}}, \boldsymbol{v}_{n-1}^{(j)}; \theta_{\mathrm{dec}}) \tag{14}$$

$$\boldsymbol{b}_{n-1}^{(j)} = \mathtt{LayerNorm}(\boldsymbol{v}_{n-1}^{(j)} + \bar{\boldsymbol{b}}_{n-1}^{(j)}) \tag{15}$$

ここで，$\mathtt{LanguageModel}(\cdot)$ は，外部言語モデルを表す．さらに，隠れ状態ベクトル $\boldsymbol{c}_{n-1}^{(j)}$ と $\boldsymbol{b}_{n-1}^{(j)}$ をゲーティング機構 [16] を用いて式 (16) によって連結させる．

$$\boldsymbol{q}_{n-1}^{(j)} = [\boldsymbol{c}_{n-1}^{(j)\,\mathrm{T}}, \boldsymbol{g}_{n-1}^{(j)} \odot \boldsymbol{b}_{n-1}^{(j)\,\mathrm{T}}]^{\mathrm{T}} \tag{16}$$

$$\boldsymbol{g}_{n-1}^{(j)} = \mathtt{sigmoid}([\boldsymbol{c}_{n-1}^{(j)\,\mathrm{T}}, \boldsymbol{b}_{n-1}^{(j)\,\mathrm{T}}]^{\mathrm{T}}; \theta_{\mathrm{dec}}) \tag{17}$$

ここで，$\mathtt{sigomid}(\cdot)$ は，活性化関数 sigmoid を用いた線形変換を表す．最終的に，$\boldsymbol{u}_{n-1}^{(j)}$ は $\boldsymbol{q}_{n-1}^{(j)}$ を入力として以下のように算出される．

$$\bar{\boldsymbol{u}}_{n-1}^{(j)} = \mathtt{FeedForward}(\boldsymbol{q}_{n-1}^{(j)}; \theta_{\mathrm{dec}}) \tag{18}$$

$$\boldsymbol{u}_{n-1}^{(j)} = \mathtt{LayerNorm}(\boldsymbol{q}_{n-1}^{(j)} + \bar{\boldsymbol{u}}_{n-1}^{(j)}) \tag{19}$$

ここで，$\mathtt{FeedForward}(\cdot)$ は，位置単位の順伝播ネットワークを表す．

### 3.3 学習

Transformer のモデルパラメータは，学習データ $\mathcal{D} = \{(\boldsymbol{X}^1, \boldsymbol{Y}^1), \cdots, (\boldsymbol{X}^{|\mathcal{D}|}, \boldsymbol{Y}^{|\mathcal{D}|})\}$ を用い，以下の損失関数で最適化される．

$$\mathcal{L} = -\frac{1}{|\mathcal{D}|} \sum_{d=1}^{|\mathcal{D}|} \log P(\boldsymbol{Y}^d | \boldsymbol{X}^d; \boldsymbol{\Theta}) \tag{20}$$

ここで，外部言語モデルのパラメータ $\theta_{\mathrm{lm}}$ は，freezing したものを使用する．

## 4 評価実験

評価実験では，話し言葉・書き言葉変換と方言変換の 2 種類のテキストスタイル変換タスクを用いた．話し言葉・書き言葉変換は，音声認識結果である話し言葉テキストを，フィラーや言い淀みなどを削除した書き言葉テキストに変換するタスクである [17]．方言変換は，日本各地の方言で話されているテキストを標準語のテキストへ変換するタスクである．

### 4.1 データ

**話し言葉・書き言葉変換：** 話し言葉・書き言葉変換では，日本語話し言葉コーパス（CSJ）[18] と日本語文章のための話し言葉・書き言葉変換コーパス（CJSW）[17] を使用した．ここで，CSJ を 46,847 文の学習セット，13,510 文の開発セット，3,949 文のテストセットに分割した．また，CJSW を 36,749 文の学習セット，2,627 文の開発セット，6,415 文のテストセットに分割した．すべてのデータは，音声認識の書き起こしである話し言葉テキストと，クラウドソーシングによって作成された書き言葉テキストのパラレルデータから成る．

**方言変換：** 東北弁，大阪弁，九州弁で話されたテキストを標準語のテキストに変換したパラレルデータをクラウドソーシングによって作成した．それらのデータを，15,506 文の学習セット，3,924 文の開発セット，2,160 文のテストセットに分割した．これらのデータは，3 種類の方言を混合したものであり，テストセットは，東北弁が 700 文，大阪弁が 862 文，九州弁が 598 文で構成した．

**テキストのみのデータ：** 様々なトピックのウェブページから我々が実装したクローラで収集した 100 万文の書き言葉テキストを用いた．

### 4.2 実験条件

テキストスタイル変換モデルには Transformer を採用し，shallow fusion [10] と cold fusion [12]，提案手法を用いて外部言語モデルの統合を行った．ここで，外部言語モデルの統合を用いない Transformer をベースラインとして用いた．また，外部言語モデルには任意の言語モデルを用いることができるが，本論文では Open AI GPT [19] を採用した．


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

Transformer のエンコーダには 256 ユニットを持つ 4 層の Transformer エンコーダブロックを，デコーダには 256 ユニットを持つ 2 層の Transformer デコーダブロックを採用した．また，GPT には 256 ユニットを持つ 4 層の Transformer ブロックを採用した．このとき，すべての Transformer ブロックにおける複数ヘッドの注意機構はヘッド数を 8 に設定した．出力層のユニットサイズは，外部言語モデルの学習データに含まれる文字数である 5,640 に設定した．shallow fusion の $\gamma$ は 0.1 に設定した．学習には adam を用い，学習率の warmup を 100,000 に設定した．また，ミニバッチサイズは 64 に設定し，各発話は 200 で truncate した．さらに，トークンの分割には文字区切りを採用した．

### 4.3 実験結果

表 1 に話し言葉・書き言葉変換の結果を，表 2 に方言変換の結果を示す．ここで，自動評価指標には，3-gram の BLEU [20] と ROUGE-L [21]，METEOR [22] を採用した．表 1，2 より，2 つのスタイル変換タスクにおいて提案手法が最も高い性能を示した．また，各手法におけるテキストスタイル変換の例を図 3 に示す．まず，CSJ の変換結果に着目すると，提案手法では"新鮮"という単語を正しく生成できているが，他の手法では生成に失敗していることがわかる．ここで，"新鮮"という単語は，Transformer の学習データには含まれていなかったが，外部言語モデルの学習データには含まれていた．次に，東北弁の変換結果に着目する．shallow fusion や cold fusion では，ベースラインの生成文の流暢性を改善することはできているものの，文頭の誤変換は修正できていなかった．一方，提案手法では，他手法の"皆"という誤変換を"何"と正しく修正できていることが確認された．以上より，提案手法では，Transformer の持つ多段な構造を明示的に活かすことにより，外部言語モデルの情報をより柔軟に抽出できるようになったと考えられる．また，表 1 と表 2 の結果を比べると，方言変換タスクの方が提案手法による効果が大きいことが確認できる．これは，Transformer の学習データ量に依存したものだと考えられる．実際に，学習データ量は，CSJ が最も多く，次に CSJW，方言変換データとなっていた．以上より，外部言語モデルの統合は，ニューラル系列変換モデルの学習データ量が少ない時ほど，より有効に働くことが示唆された．

**表 1** 話し言葉・書き言葉変換タスクにおける結果

| | | BLEU-3 | ROUGE-L | METEOR |
|---|---|---|---|---|
| CSJ | a). | 0.667 | 0.855 | 0.853 |
| | b). | 0.667 | 0.850 | 0.853 |
| | c). | 0.657 | 0.852 | 0.847 |
| | d). | **0.669** | **0.860** | **0.856** |
| CJSW | a). | 0.706 | 0.763 | 0.866 |
| | b). | 0.690 | 0.752 | 0.855 |
| | c). | 0.709 | 0.764 | 0.867 |
| | d). | **0.718** | **0.769** | **0.874** |

a). ベースライン　b). Shallow fusion
c). Cold fusion　d). 提案手法

**表 2** 方言変換タスクにおける結果

| | | BLEU-3 | ROUGE-L | METEOR |
|---|---|---|---|---|
| 大阪弁 | a). | 0.649 | 0.784 | 0.790 |
| | b). | 0.638 | 0.774 | 0.780 |
| | c). | 0.648 | 0.784 | 0.787 |
| | d). | **0.663** | **0.795** | **0.802** |
| 九州弁 | a). | 0.741 | 0.857 | 0.872 |
| | b). | 0.729 | 0.849 | 0.859 |
| | c). | 0.738 | 0.855 | 0.867 |
| | d). | **0.752** | **0.864** | **0.880** |
| 東北弁 | a). | 0.619 | 0.767 | 0.742 |
| | b). | 0.603 | 0.755 | 0.721 |
| | c). | 0.610 | 0.761 | 0.730 |
| | d). | **0.630** | **0.772** | **0.752** |

a). ベースライン　b). Shallow fusion
c). Cold fusion　d). 提案手法

## 5 おわりに

本論文では，Transformer の構造を明示的に活用した外部言語モデルの統合方法について提案した．従来のニューラル系列変換モデルに対する外部言語モデルの統合方法は，最終層の直前に言語モデルの情報を統合するのみであった．それに対して，提案手法では外部言語モデルの出力に対して，多段に注意機構を設けることにより，系列変換に有効な情報を柔軟に取捨選択できるようにした．評価実験の結果，既存手法と比べて提案手法では，2 種類のテキストスタイル変換タスクにおいて従来手法よりも性能が改善され，有効性が示唆された．

| | CSJの変換結果 | 東北弁の変換結果 |
|---|---|---|
| 入力 | 新鮮で海のもの | なにみ、みにいぐんだべ |
| 正解 | 新鮮で海のもの | 何を見に行くのでしょう。 |
| ベースライン | 新朝海でのもの | 皆、見にいくらいでしょう。 |
| shallow fusion | 新海でのもの | 皆見に行くのでしょうか。 |
| cold fusion | 新朝で海のもの | 皆、見に行くのでしょうか。 |
| 提案手法 | 新鮮で海のもの | 何を見に行くのでしょうか。 |

**図 3** 各手法におけるスタイル変換の例


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

[2] Qiang Wang, Bei Li, Tong Xiao, Jingbo Zhu, Changliang Li, Derek F. Wong, and Lidia S. Chao. Learning deep transformer models for machine translation. In *Proc. Association for Computational Linguistics(ACL)*, pp. 1810–1822, 2019.

[3] Loïc Barrault, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, Christof Monz, Mathias Müller, Santanu Pal, Matt Post, and Marcos Zampieri. Findings of the 2019 conference on machine translation (WMT19). In *Proc. Conference on Machine Translation (WMT)*, pp. 1–61, 2019.

[4] Jie Li, Xiaorui Wang, Yan Li, et al. The speechtransformer for large-scale Mandarin Chinese speech recognition. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7095–7099, 2019.

[5] Jun Yu, Jing Li, Zhou Yu, and Qingming Huang. Multimodal transformer with multi-view visual representation for image captioning. *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2019.

[6] Guang Li, Linchao Zhu, Ping Liu, and Yi Yang. Entangled transformer for image captioning. In *Proc. International Conference on Computer Vision (ICCV)*, pp. 8928–8937, 2019.

[7] Linhao Dong, Shuang Xu, and Bo Xu. Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5884–5888, 2018.

[8] Shigeki Karita, Nanxin Chen, Tomoki Hayashi, Takaaki Hori, Hirofumi Inaguma, Ziyan Jiang, Masao Someki, Nelson Enrique Yalta Soplin, Ryuichi Yamamoto, Xiaofei Wang, et al. A comparative study on transformer vs RNN in speech applications. In *Proc. Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 449–456, 2019.

[9] Julian Salazar, Katrin Kirchhoff, and Zhiheng Huang. Self-attention networks for connectionist temporal classification in speech recognition. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7115–7119, 2019.

[10] Anjuli Kannan, Yonghui Wu, Patrick Nguyen, Tara N Sainath, Zhijeng Chen, and Rohit Prabhavalkar. An analysis of incorporating an external language model into a sequence-to-sequence model. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5824–5828, 2018.

[11] Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loic Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. On using monolingual corpora in neural machine translation. *arXiv preprint arXiv:1503.03535*, 2015.

[12] Anuroop Sriram, Heewoo Jun, Sanjeev Satheesh, and Adam Coates. Cold fusion: Training seq2seq models together with language models. In *Proc. International Speech Communication Association (INTERSPEECH)*, pp. 387–391, 2018.

[13] Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. End-to-end memory networks. In *Proc. Advances in neural information processing systems (NIPS)*, pp. 2440–2448, 2015.

[14] Jan Chorowski and Navdeep Jaitly. Towards better decoding and language model integration in sequence to sequence models. In *Proc. International Speech Communication Association (INTERSPEECH)*, pp. 523–527, 2017.

[15] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Proc. Advances in neural information processing systems(NIPS)*, pp. 3104–3112, 2014.

[16] Zhilin Yang, Bhuwan Dhingra, Ye Yuan, Junjie Hu, William W Cohen, and Ruslan Salakhutdinov. Words or characters? fine-grained gating for reading comprehension. *arXiv preprint arXiv:1611.01724*, 2016.

[17] Mana Ihori, Akihiko Takashima, and Ryo Masumura. Parallel corpus for Japanese spoken-to-written style conversion. In *Proc. Language Resources and Evaluation Conference (LREC)*, pp. 6346–6353, 2020.

[18] Kikuo Maekawa, Hanae Koiso, Sadaoki Furui, and Hitoshi Isahara. Spontaneous speech corpus of Japanese. In *Proc. International Conference on Language Resources and Evaluation (LREC)*, pp. 947–952, 2000.

[19] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, p. 9, 2019.

[20] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proc. Annual Meeting on Association for Computational Linguistics (ACL)*, pp. 311–318, 2002.

[21] Chin-Yew Lin and Franz Josef Och. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proc. Annual Meeting on Association for Computational Linguistics (ACL)*, pp. 605–612, 2004.

[22] Satanjeev Banerjee and Alon Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proc. the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pp. 65–72, 2005.




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