潜在変数の精緻化による非自己回帰型ニューラル機械翻訳

朱 中元[1]　Jason Lee[2]　Kyunghyun Cho[2]　中山 英樹[1]
[1] 東京大学大学院 情報理工学系研究科
{shu, nakayama}@nlab.ci.i.u-tokyo.ac.jp
[2] ニューヨーク大学
{jasonlee.inf, kyunghyun.cho}@nyu.edu

1 はじめに

近年，ニューラル機械翻訳の進歩により翻訳文の質は著しく向上した．通常のニューラル機械翻訳モデルは，文の生成確率 $p(y \mid x)$ を下記のように各単語の生成確率に分解する自己回帰モデルとして記述される．

$$p(y \mid x) = \prod_{t=1}^{|y|} p(y_t \mid y_{<t}, x). \tag{1}$$

ここで，$|y|$ は出力文 y の長さであり，$y_{<t}$ は単語 y_t より手前の単語列 $y_1, \ldots, y_{t-1}$ を表す．右辺の単語確率を計算するために，LSTM [1] や Transformer [9] が一般的に用いられる．

しかし，自己回帰モデルでは，各単語の生成確率がそれより前の単語 $y_{<t}$ に依存するため，文中の単語を並列で予測することができない．Transformer を使う場合，Teacher Forcing により学習時の単語予測は容易に並列化できるものの，テスト時には単語を逐次的に出力することになる．そのため，自己回帰型ニューラル機械翻訳では，長文を翻訳する際に大きな遅延が生じるという欠点がある

本研究では，ニューラル機械翻訳を潜在変数モデルとして定式化し，連続潜在空間で潜在変数を精緻化（更新）するアプローチを提案する．潜在変数が出力文の単語間の依存性を捉えられるように学習することで，デコーダはすべての単語を同時に生成することができる．さらに高品質な翻訳文を得るために，潜在変数を更新する方法を検討する．提案手法は，WMT'14 英独翻訳タスクにおいて，ベースラインモデルの BLEU 値より 1.0 ポイント下回るが，6.8 倍速く翻訳できることを示した．ASPEC 日英翻訳タスクにおいては，ベースラインの翻訳精度を保ったまま，8.6 倍高速に翻訳できることを確認した[1]．

[1]本研究のソースコードは https://github.com/zomux/lanmt にて公開している．

2 関連研究

2018 年以降，出力文の長さを利用し，単語間の依存関係を捉える非自己回帰型ニューラル機械翻訳が提案された [3, 6]．Gu ら [3] の手法では，まず出力文の単語数 l_y を予測する．続いて，出力文の長さを条件にした翻訳モデル $p(y \mid x, l_y)$ を使って翻訳を行う．この手法では，l_y が離散潜在変数の役割を担っていると考えることができる．しかし，このモデルの翻訳精度は WMT'14 英独翻訳タスクにおいて，Transformer モデルよりも著しく下回っている．出力単語の整合性を向上させるために，Lee ら [6] と Ghazvininejad ら [2] は出力した単語を逐次的に再予測 (更新) する手法を提案した．

これらの関連研究では，出力文長を潜在変数とした．一方，本研究では，出力文長を含む，出力文の内容を表現するような潜在変数を自動的に学習することに焦点を当てる．

3 LaNMT:潜在変数に基づく非自己回帰型ニューラル機械翻訳

本研究では，潜在変数モデルとしてニューラル機械翻訳を定式化する方法を提案する．

提案手法では，潜在変数の数を入力単語数に合わせる．すなわち，入力文の各単語 x_i に対して潜在変数 z_i を学習する．以降，全入力単語に対応する潜在変数の系列 $\{z_1, \ldots, z_{|x|}\}$ を z と表記する．直観的には，各潜在変数は入力文のある単語に対応する部分的な翻訳を捉えているとみなせる．この場合，式 (1) の代わりに，下記の周辺確率を最大化するようにモデルを学習する．

$$\log p(y|x) = \log \int p(y|z, x) p(z|x) dz. \tag{2}$$

潜在変数 z の事前分布を $p_\omega(z|x)$ とした場合，この式の変分下限は

$$\mathcal{L}(\omega,\phi,\theta) = \mathbb{E}_{z\sim q_\phi}\big[\log p_\theta(y|x,z)\big] - \mathrm{KL}\big[q_\phi(z|x,y)||p_\omega(z|x)\big] \quad (3)$$

のようになる．ここで，θ と ϕ,ω はそれぞれ確率分布のパラメータを表す．$q_\phi(z|x,y)$ は潜在変数 z の事後分布を近似する近似分布であり，reparameterization trick [5] によって学習できる．

文長予測モデル　提案モデルでは，潜在変数の数は入力単語数と一致するので，翻訳文の単語を同時に生成するためには出力文の長さ l_y を予測する必要がある．ここで，デコータ分布を次のように式変形する．

$$\begin{aligned} p_\theta(y|x,z) &= \sum_l p_\theta(y,l|x,z) \\ &= p_\theta(y,l=l_y|x,z) \\ &= p_\theta(y|x,z,l_y)p_\theta(l_y|z). \end{aligned} \quad (4)$$

この結果を式 (3) に代入すると，提案モデルの目的関数は

$$\mathbb{E}_{z\sim q_\phi}\big[\sum_{t=1}^{|y|}\log p_\theta(y_t|x,z,l_y) + \log p_\theta(l_y|z)\big] - \sum_{i=1}^{|x|}\mathrm{KL}\big[q_\phi(z_i|x,y)||p_\omega(z_i|x)\big] \quad (5)$$

のようになる．

本研究では，事前分布 $p_\omega(z|x)$ と近似分布 $q_\phi(z|x,y)$ を共に多変量正規分布とし，式 (5) の変分下限を最大化するようにモデルを学習する．

3.1　ニューラルネットワーク構造

提案モデルは式 (5) に示した 4 つの確率分布を計算する必要がある．それぞれの確率分布は図 1に示すニューラルネットワークによって計算する．各モジュールは Transformer モデルと類似した構造を用い，$p_\omega(z|x)$ と $q_\phi(z|x,y)$ を予測する際に予め隠れ状態に対して線形変換を行う．

潜在変数の長さの変換　近似分布 $q_\phi(z|x,y)$ からサンプリングする潜在変数の長さは $|x|$ であるが，出力単語を予測するためには，出力側の長さ $|y|$ に変換する必要がある[2]．変換した潜在変数を $\bar{z} = \bar{z}_1,...,\bar{z}_{|y|}$ とすると，提案モデルは各 $\bar{z}_t$ を $z_1,\ldots,z_{|x|}$ の重み付き平均によって計算する．

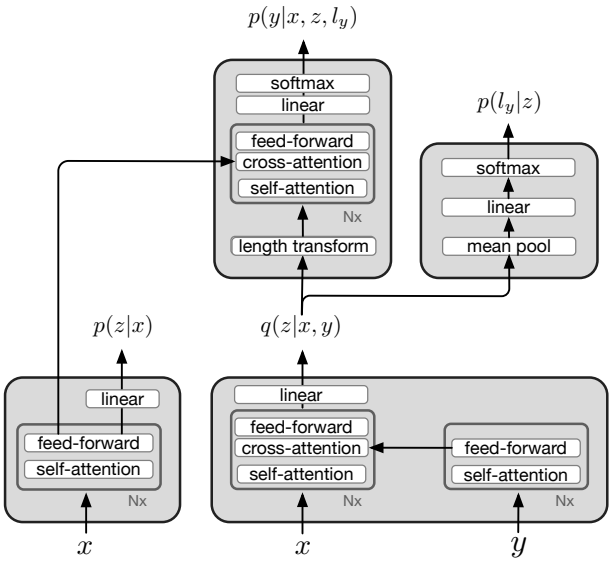

図 1: 提案する非自己回帰型ニューラル機械翻訳モデルのネットワーク構造.

$$\bar{z}_t = \sum_{k=1}^{|x|} w_i^t z_i, \qquad w_i^t = \frac{\exp(a_i^t)}{\sum_{i'=1}^{|x|}\exp(a_{i'}^t)}, \quad (6)$$

$$a_i^t = -\frac{1}{2\sigma^2}(i - \frac{|x|}{|y|}t)^2. \quad (7)$$

ここで，$\bar{z}_t$ を計算する際に，入力側で位置が $\frac{|x|}{|y|}t$ に近い変数に対して高い重みを付与する．重みの広がりを制御するパラメータ σ は自動的に学習される．

3.2　潜在変数の精緻化

翻訳文の品質を一定に保つため，決定的に翻訳文を出力する必要がある．そのために，事前分布 $p_\omega(z|x)$ から平均ベクトル z_{prior} を取得し, デコータ $p_\theta(y|x,z)$ を用いて出力文を予測する方法がまず考えられる．しかし，予備実験では，WMT'14 英独タスクにおいてこの方法による翻訳結果の BLEU 値はベースラインから約 4.0 下回ることが分かった．一方，近似分布 $q_\phi(z|x,y)$ に正解出力文 y^* を入力することで得た z^* を用いて翻訳した場合，はるかに高い BLEU 値を達成できた．また，z_{prior} と z^* の線形補間を行った場合，BLEU 値は滑らかに変化することを確認した．

[2]例えば，各潜在変数 z_i の次元数が 8 である場合, 近似分布からサンプリングした z は $|x|\times 8$ の行列である．

	ASPEC 日英翻訳			WMT'14 英独翻訳		
	BLEU(%)	速度	実時間	BLEU(%)	速度	実時間
Transformer ベースライン, beam 幅=3	27.1	1x	415ms	26.1	1x	602ms
Transformer ベースライン, beam 幅=1	24.6	1.1x	375ms	25.6	1.3x	461ms
非自己回帰型翻訳モデル（提案モデル）	13.3	17.0x	24ms	11.8	22.2x	27ms
+ 知識蒸留法	25.2	17.0x	24ms	22.2	22.2x	27ms
+ 潜在変数の精緻化	27.5	8.6x	48ms	24.1	12.5x	48ms
+ 複数の潜在変数を探索	28.3	4.8x	86ms	25.1	6.8x	88ms

表 1: 提案モデルと Transformer ベースラインの比較．WMT'14 英独翻訳タスクにおいて，本実験での Transformer ベースライン実装は元論文 [9] の BLEU 値より 1.0 低くなっている．

以上の予備実験の結果は，事前分布から得た潜在変数を適切に更新することにより，翻訳文の質を向上させることが可能であることを示唆するものであると言える．

デルタ分布による決定的な更新アルゴリズムの導出 本研究では，近似分布を介して決定的に潜在変数を更新するアルゴリズムを考案し，実験で検証する．ここで，決定的デルタ分布 $r(z)$ を $r(z=\mu)=1, r(z\neq\mu)=0$ と定義する．すなわち，この分布の確率質量をすべて μ に置く．次に，近似分布との KL ダイバージェンス $\mathrm{KL}(r(z)||q_\phi(z|x,y))$ を最小化することで，$\mu=\mathbb{E}_{q_\phi}[z]$ を得ることができる．

式 (3) の変分下限の中の $q_\phi(z|x,y)$ を $r(z)$ に置き換えると $\mathcal{L}=\log p_\theta(y|x,z=\mu)-\log p_\omega(\mu|x)$ と式変形することができる．これにより，変分下限を最大化するためには，各出力単語を $\hat{y}_t=\mathrm{argmax}_{y_t}\log p_\theta(y_t|x,z=\mu)$ によって得ればよいことが分かる．

出力単語を更新した場合，近似分布 $q_\phi(z|x,y)$ も変化するので，デルタ分布の μ を改めて計算する必要がある．そのため，μ と $\hat{y}$ を交互に更新するアルゴリズムを導出した．実験では，μ の初期値を z_{prior} にすることで，一回目の更新で BLEU 値がほぼ収束することが観測された．

4 実験

4.1 実験設定

データと評価 提案モデルの翻訳精度を考察するために，ASPEC 日英翻訳データ [8] 及び WMT'14 英独翻訳データを用いた．それぞれのデータは 300 万と 450 万の文ペアを含む．ASPEC 日英データの前処理は英語側を Moses，日本語側は Kytea で処理し，最終的に BPE を用いて語彙数が 4 万になるようにサブワード分割を行った．WMT'14 データセットの前処理は sentencepiece で語彙数が 3.2 万になるように処理した．

翻訳精度を評価するために，日英翻訳では tokenized BLEU，英独翻訳では SacreBLEU ツールを用いて BLEU 値を計算した．翻訳速度は NVIDIA V100 GPU 一枚を用いて測定した．

ハイパーパラメータ 本実験では可能な限り Base Transformer[9] のハイパーパラメータを用いた。隠れ層の次元数を 512 にし，$p(z|x)$ と $p(y|x,z,l_y)$ を計算するネットワークはそれぞれ 6 層，$q(z|x,y)$ を計算するネットワークは 3 層にした．各入力単語に対する潜在変数 z_i の次元数は 8 次元に設定した．

モデル学習 生成モデルの変分下限をそのまま最適化すると，近似分布と事前分布の KL ダイバージェンスが学習の初期に 0 になり，学習が進まない場合がある．この現象は Posterior Collapse と呼ばれる．本実験では，Kingma ら [4] の手法を用いて，各潜在変数の KL の目標値を 3 に設定し，KL 項の値が目標値よりも低くなった場合，KL 項の最適化を行わないように損失関数を変更した．モデルの学習時には，KL の目標値を徐々に 0 まで減らした．

知識蒸留法 既存研究と同じように，ベースラインとなる自己回帰型モデルを用いて一回学習データを翻訳し，翻訳した学習データを提案モデルの教師データにした．

	BLEU(%)	速度
Transformer (元論文) [9]	27.1	-
Gu らのベースライン [3]	23.4	1x
NAT モデル (NPD S=100)	19.1 (-4.3)	2.3x
Lee らのベースライン [6]	24.5	1x
Adaptive NAT モデル	21.5 (-3.0)	1.9x
Wang らのベースライン [10]	27.3	1x
NAT-REG モデル	20.6 (-6.7)	27.6x*
+ リランキング	24.6 (-2.7)	15.1x*
Ghaz らのベースライン [2]	27.8	1x
CMLM モデル (4 iterations)	26.0 (-1.8)	-
CMLM モデル (10 iterations)	26.9 (-0.9)	2∼3x
Ma らのベースライン [7]	27.1	-
FlowSeq-large (NPD S=30)	25.3 (-1.8)	-
本実験のベースライン	26.1	1x
提案モデル+潜在変数の精緻化	24.1 (-2.0)	12.5x
+ 複数の潜在変数を探索	25.1 (-1.0)	6.8x

表 2: WMT'14 英独翻訳タスクにおける関連研究との比較．$\star$ は IWSLT'14 独英タスクにおける速度である

4.2 結果

自己回帰型モデルとの比較結果を表 1にまとめる．知識蒸留法を用いない場合，非自己回帰モデルの性能が大きく低下することを確認した．一方，知識蒸溜法を用いた場合は，ベースラインの BLEU 値より約 2 ∼ 4 ポイント下回る程度まで改善した．さらに，提案する更新アルゴリズムを 1 ステップ適用した場合，BLEU 値がさらに約 2 ポイント向上することが確認できた．

複数の潜在変数の探索　これまでの実験では，潜在変数の初期値を事前分布の平均 z_{prior} としたが，事前分布から複数の潜在変数を K 個サンプリングし，異なる翻訳結果 $\hat{y}_1, ..., \hat{y}_K$ を同時に生成して，ベースラインモデルを用いてリランキングする方法も考えられる．$K = 50$ にとった場合，ASPEC 日英翻訳では，翻訳精度がベースラインを超え，WMT'14 英独翻訳においてベースラインの精度に近づくことができた．この設定では，一文を翻訳する平均速度はベースラインよりそれぞれ 4.8 倍，6.8 倍高速であった．表 2に関連研究との比較をまとめる．

5 おわりに

本研究では，潜在変数モデルとして定式化した非自己回帰型ニューラル機械翻訳モデルを提案し，変分下限の近似分布をデルタ分布に置換することで，潜在変数の更新アルゴリズムを導出し，その有効性を検証した．

事前分布で取得した潜在変数を適切に更新することによって，非自己回帰型ニューラル機械翻訳モデルの精度を著しく改善できることを実験的に確認した．すなわち，単語の予測精度は提案モデルにおいて潜在変数の更新関数 $\text{update}(z, x)$ の性能によって左右される．今後は，優れた更新関数の提案によって非自己回帰型モデルの翻訳性能を更に向上できると考える．

謝辞

本研究は，独立行政法人情報通信研究機構 (NICT) の委託研究「多言語音声翻訳高度化のためのディープラーニング技術の研究開発」および JSPS 科研費 JP19H04166 の成果として得られたものです．

参考文献

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *ICML*, 2015.

[2] Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke S. Zettlemoyer. Constant-time machine translation with conditional masked language models. *CoRR*, 2019.

[3] Jiatao Gu, James Bradbury, Caiming Xiong, Victor O. K. Li, and Richard Socher. Non-autoregressive neural machine translation. *CoRR*, Vol. abs/1711.02281, , 2018.

[4] Diederik P. Kingma, Tim Salimans, and Max Welling. Improving variational inference with inverse autoregressive flow. *CoRR*, Vol. abs/1606.04934, , 2016.

[5] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. *CoRR*, Vol. abs/1312.6114, , 2014.

[6] Jason Lee, Elman Mansimov, and Kyunghyun Cho. Deterministic non-autoregressive neural sequence modeling by iterative refinement. In *EMNLP*, 2018.

[7] Xuezhe Ma, Chunting Zhou, Xian Li, Graham Neubig, and Eduard H. Hovy. Flowseq: Non-autoregressive conditional sequence generation with generative flow. *EMNLP*, 2019.

[8] Toshiaki Nakazawa, Manabu Yaguchi, Kiyotaka Uchimoto, Masao Utiyama, Eiichiro Sumita, Sadao Kurohashi, and Hitoshi Isahara. Aspec: Asian scientific paper excerpt corpus. In *LREC*, 2016.

[9] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, 2017.

[10] Yiren Wang, Fei Tian, Di He, Tao Qin, ChengXiang Zhai, and Tie-Yan Liu. Non-autoregressive machine translation with auxiliary regularization. *CoRR*, Vol. abs/1902.10245, , 2019.