単語分散表現に基づいた誤差によるニューラル機械翻訳の学習

帖佐 克己　　須藤 克仁　　中村 哲
奈良先端科学技術大学院大学 先端科学技術研究科 情報科学領域
{k-chousa, sudoh, s-nakamura}@is.naist.jp

1 はじめに

ニューラル機械翻訳（Neural Machine Translation; NMT）[1, 2] における問題点の一つとして，出力層における出力単語の生成確率を計算する際の計算コストが非常に大きいという問題が挙げられる．この計算コストは目的言語側の語彙サイズに比例するため，低頻度語を無視して語彙サイズを小さくした場合には語彙外の単語が増加してしまうために翻訳精度が大幅に減少してしまう．この語彙サイズと翻訳精度のトレードオフを改善するために，本研究では単語同士の距離を導入した新しい誤差関数を提案する．

従来の NMT の学習の際には誤差関数として *softmax cross-entropy* が使用され，この誤差関数は正解単語の生成確率 1 に近づくこと，その他の単語の確率については**その単語の意味によらず** 0 に近づくことを促す．例えば正解単語が *see* の時，*look* のような意味が近い単語にも意味が遠い単語と同様なペナルティが掛かってしまう．特に正解単語が語彙外の場合にこの問題は深刻となる．なぜならば，語彙外を示すシンボルのみが生成されやすくなるように最適化されてしまうことから，単語の生成確率が一様に減少し，人間にとって意味のある文章生成が行えなくなってしまう．

入力単位として subword[3, 4] を用いる方法は語彙外の単語を大幅に減少させることができ，単語単位のモデルと比べて高い精度が得られることが知られている．この方法を用いることで上記の語彙外の単語の最適化に関わる問題は回避することができる．しかし依然として softmax cross-entropy を使った最適化を行っているため，単語の意味を考慮することなく単語の生成確率が一様に減少する問題は解決することができていない．これらの問題に対して，単語の意味情報を考慮した誤差関数を NMT の学習に使用することによる改善が考えられる．この場合，単語の意味情報を何らかの手法で得ることが必要となる．

単語の意味情報を扱う研究として単語分散表現が知られている．この単語分散表現は単語を空間上の低次元の実数値ベクトルとして表現したものであり，word2vec[5] などの手法が提案されている．この表現は，単語同士の使用方法や意味が近ければ近いほどその単語のベクトル同士の距離も小さくなると言われている．本研究では，この特徴を使用して NMT の誤差関数に統語的・意味的な類似度に応じて誤差を与える誤差関数を提案する．誤差関数は正解単語と目的言語側の語彙に含まれる単語との間の単語分散表現での距離の重み付き平均で定義し，その重みにはモデルの softmax 層で得られる各単語の生成確率を使用する．この誤差関数を導入することで，正解単語に加えて似た意味の単語の生成確率が大きくなり，反対に似ていない単語の生成確率は小さくなることが促される．また，単語分散表現は NMT の学習に使用するパラレルコーパスとは独立に単言語コーパスから作成できるため，パラレルコーパス上で低頻度な語に関しても大規模な単言語コーパスを使用することでその情報を使用することができる．特に，語彙外の単語が多い場合には，表 1のように語彙外の単語の代わりに語彙内の似た単語が生成されることが期待できる．この提案手法の有効性を確認するため，複数の学習方法の比較（§4.3），目的言語側の語彙サイズを小さくした際の影響（§4.4）および異なる言語対での影響（§4.5）の 3 つの異なる実験を行う．

表 1: 日英翻訳における生成例．この場合では *moldings* が語彙外であるため，baseline では<unk>を出力している．一方，提案手法では *moldings* の代わりに意味が近い *extrusion* を使用した生成ができている．

source:	リザーバ内流動パターンと押出物形態
reference:	The flow pattern in the reservoir and the shape of <unk:moldings>.
baseline:	Flow patterns in reservoir and <unk> forms.
proposed:	The flow pattern in the reservoir and the extrusion form.

2 Attention 機構付き Encoder-Decoder モデルによる NMT

はじめに，本稿のベースラインモデルとした Attention 機構付き Encoder-Decoder モデル [2] について説明する．

$X = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_I\}$ を入力文（入力系列），$Y =$

$\{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_J\}$ を出力文（出力系列）とする．ここで，$\mathbf{x}_i \in \mathbb{R}^{S\times 1}$ は i 番目の入力単語を表す one-hot ベクトル，I は入力文の長さ，$\mathbf{y}_j \in \mathbb{R}^{T\times 1}$ は j 番目の出力単語を表す one-hot ベクトル，J は出力文の長さを表す．

モデルは Encoder（§2.1）と Attention + Decoder（§2.2）の 2 つの機構から構成され，そのどちらも RNN（Recurrent Neural Network）を用いて構成される．

2.1 Encoder

Encoder は入力文 X を入力として受け取り，RNN を通じて順方向の隠れ状態ベクトル $\overrightarrow{\mathbf{h}_i}(1 \leq i \leq I)$ を返す．

$$\overrightarrow{\mathbf{h}_i} = RNN(\overrightarrow{\mathbf{h}_{i-1}}, \mathbf{x}_i). \tag{1}$$

同様に，逆順に並べた入力文を入力することで逆方向の隠れ状態ベクトル $\overleftarrow{\mathbf{h}_i}(1 \leq i \leq I)$ が得られる．これらの 2 つの方向の隠れ状態ベクトルを結合することで以下のように入力文の隠れ状態ベクトルを得る．これにより全てのタイムステップにおいて前後の文脈を考慮した隠れ状態ベクトルを得ることができる．

$$\mathbf{h}_i = [\overrightarrow{\mathbf{h}_i}; \overleftarrow{\mathbf{h}_i}]. \tag{2}$$

2.2 Attention + Decoder

Attention + Decoder では Encoder で計算された入力文の隠れ状態ベクトルから翻訳文の単語を 1 つずつ生成する．Decoder の RNN は初期隠れ状態ベクトル $\mathbf{h}_I$ から始まり，隠れ状態と過去の出力系列から再帰的に単語を生成する．出力単語 $\mathbf{y}_i$ の条件付き確率は以下のように定義される．

$$p_\theta(\mathbf{y}_j|\mathbf{y}_{<j}, X) = softmax(\mathbf{W}_s\tilde{\mathbf{d}_j}), \tag{3}$$

$$\tilde{\mathbf{d}_j} = tanh(\mathbf{W}_c[\mathbf{c}_j; \mathbf{d}_j]), \tag{4}$$

$$\mathbf{d}_j = RNN(\mathbf{d}_{j-1}, \mathbf{y}_{j-1}). \tag{5}$$

ここで，$\mathbf{W}_c, \mathbf{W}_p$ は学習されるパラメータである．また，$\mathbf{c}_j$ は文脈ベクトルである．この $\mathbf{c}_j$ を求めるために Attention と呼ばれる機構を用いる．Attention 機構では，入力文の隠れ状態ベクトル $\mathbf{h}_i$ をその各ベクトルに対応する時間ステップ j における重み α_{ij} を計算し，その重みと隠れ状態ベクトルの重み付き平均を取ることで $\mathbf{c}_j$ が以下のように求められる．

$$\mathbf{c}_j = \sum_{i=1}^{I} \alpha_{ij}\mathbf{h}_i, \tag{6}$$

$$\alpha_{ij} = \frac{exp(\mathbf{d}_j^T\mathbf{h}_i)}{\sum_{i'=1}^{I} exp(\mathbf{d}_j^T\mathbf{h}_{i'})} \tag{7}$$

3 誤差関数

この節では NMT における一般的な誤差関数である Softmax Cross-Entropy について振り返った後（§3.1），提案手法である Word Embedding-based loss について説明する（§3.2）．

3.1 Softmax Cross-Entropy

Softmax Cross-Entropy は言語生成のような多クラス分類問題の最適化に一般的に用いられる誤差関数であり，以下のように定義される．

$$\ell_{ent} = -\sum_{j=1}^{J}\sum_{k=1}^{K} \mathbf{y}_{jk} \log p_\theta(\mathbf{y}_{jk}|\mathbf{y}_{<j}, X). \tag{8}$$

ここで $\mathbf{y}_{jk}$ は翻訳文の j 番目の単語に対応する one-hot ベクトルの k 番目の要素，K は目的言語側の語彙サイズを表す．

先に述べたように，Softmax Cross-Entropy は正解単語以外のすべての単語に，その単語の意味に関わらず等しくペナルティを与える．

3.2 Word Embedding-based Loss

本研究では，正解単語との意味の近さに応じて誤差を与える以下の誤差関数を導入する．この誤差関数を本稿では *Word Embedding-based Loss* と呼び，以下のように正解単語との距離の重み付き平均として定義する。距離の重みは式（3）で定義された出力単語の生成確率を用いる．

$$\ell_{emb} = -\sum_{j=0}^{J}\sum_{k=0}^{K} p_\theta(\mathbf{y}_{jk}|\mathbf{y}_{<j}, X) d(E(V_k), E(\mathbf{y}_j)). \tag{9}$$

ここで，V_k は目的言語側の語彙に含まれる k 番目の単語，$E(w)$ は単語 w の単語分散表現を表す．また，d は 2 つの単語ベクトルの間の距離を計算する関数で，本稿ではユークリッド距離を用いた．

$$d(\mathbf{s}, \mathbf{t}) = \|\mathbf{s} - \mathbf{t}\|. \tag{10}$$

4 実験

提案手法の影響を確認するために，複数の学習方法の比較（§4.3），目的言語側の語彙サイズを小さくした際の影響（§4.4）および異なる言語対での影響（§4.5）の 3 つの異なる実験を行った．

4.1 実験設定

モデルの実装には primitiv[*1] を用いた．また，Encoder と Decoder の RNN はそれぞれ 2 層の LSTM とし，input feeding[2] を行った．単語埋め込みベクトルや隠れ状態ベクトルの次元数はどちらも 512，ミニバッチのサイズは 64 とした．原言語側の語彙は訓練用データ中に出現する頻度の高い単語上位 20,000 語を使用した．最適化アルゴリズムには Adam[6] を使用し，gradient clipping は 5，weight decay は 10^{-6} に設定して学習を行った．ドロップアウトの確率 p は 0.3 とし，learning rate は各 epoch ごとに validation loss が低下

[*1] https://github.com/primitiv/primitiv

表 2: 実験に用いたコーパス.

Corpus	Lang.	Number of Sentence		
		Train	Valid.	Test
ASPEC	Ja-En	964k	1790	1812
IWSLT17	En-Fr	226k	890	1210

しない場合にのみ $1/\sqrt{2}$ を掛けることで減衰を行った．また，テストは最も小さい validation loss を記録したモデルによって行った．評価尺度には，機械翻訳の自動評価として一般的な BLEU[7] と同義語への言い換えにも対応している METEOR[8] を使用した．

4.2 データセット

実験には 2 つのパラレルコーパスを使用した．1 つは日本語から英語へのタスクに用いた ASPEC[9] で，3 つ全ての実験で使用した．もう 1 つは英語からフランス語へのタスクに用いた IWSLT17[*2] で，3 つ目の実験（§4.5）で用いた．表 2 にコーパスの詳細を示す．

英語及びフランス語のトークナイズには Moses tokenizer[*3]，日本語には KyTea[10] を使用した．また，60 トークンを超える文対を学習データから削除するフィルタリングを行った．

また，実験中の目的言語である英語とフランス語に対する一般的なドメインの単語分散表現を使用した．英語の単語分散表現には Google News dataset によって学習済みのもの[*4] を使用した．フランス語では，Wikipedia のダンプデータ[*5] を使用して gensim[*6] で学習したものを使用した．単語分散表現を得るためのアルゴリズムには word2vec の CBOW モデルを用いた．ウィンドウサイズおよびネガティブサンプリングの数はそれぞれ 5，単語分散表現の次元数は 300 として使用した．

4.3 学習方法による影響

まずはじめに，cross-entropy（ℓ_{ent}）と提案手法（ℓ_{emb}）をそれぞれどのように使用すれば良い結果が得られるのかを確認した．具体的には，ℓ_{ent} のみ (baseline)，ℓ_{emb} のみ，$\ell_{ent} + \ell_{emb}$ の 3 つの組み合わせについて比較を行った．また，ℓ_{emb} のみで事前学習を行った場合についても同様に比較を行った．また，目的言語側の語彙サイズは 10,000 とした．

表 3の中段に実験結果を示す．ただし，事前学習なしの ℓ_{emb} のみを使用する方法はうまく学習できず，誤差が下がらなかったため掲載していない．結果としては，ℓ_{emb} を使用したすべての方法でベースラインよりも BLEU および METEOR が向上する結果となった．特に，ℓ_{ent} で事前学習を行った後に ℓ_{emb} で最適化を行ったものが最も高い精度となり，語彙サイズを 2 倍にした際の精度（表 3上段）と同等の精度を達成した．これらの結果から，提案した誤差関数は比較的小さな語彙サイズにおいて有効であると考えられる．

4.4 語彙サイズによる影響

次に，非常に語彙サイズが限られた環境における提案手法の有効性を確認するため，目的言語側の語彙サイズを 1,000 単語に制限して実験を行った．

表 3の下段に実験結果を示す．こちらでも ℓ_{emb} を用いたすべての手法の精度がベースラインを上回る結果となった．特に，事前学習後に ℓ_{emb} のみで最適化を行う方法では METEOR でのスコアが +1.72 ポイントと大きな改善を示していることから，同義語にうまく言い換える事によって未知語を回避した翻訳が行えている事がわかる．これらの結果より，提案手法は非常に限られた環境でも機能すると考えられる．

4.5 言語対による影響

最後に，提案手法が特定の言語対に依存したものであるかを確認するために，日本語-英語とは異なる言語対による実験を行った．実験は IWSLT17 の英語-フランス語データで行い，§4.3の結果と比較するために目的言語側の語彙サイズは 10,000 単語とした．

表 4に実験結果を示す．今までの実験と同様に．ℓ_{emb} を使用した方法の精度がベースラインをすべて上回っており，特に事前学習後に ℓ_{emb} のみを用いたものが最も高い精度を記録した．また，英語-フランス語での結果は日本語-英語での精度向上よりも更に大きな改善を示している．これらの結果から，提案手法は特定の言語対に依存した手法ではないと考えられる．

4.6 考察

以上の実験結果より NMT における提案手法の有効性が示された．特に誤差関数が平滑化されることによって似た意味の単語が生成されやすくなっていることがわかった．

表 1に ASPEC での日本語-英語翻訳の生成例を示す．この例では参照訳に含まれる *moldings* が未知語になっている．そのため，ベースラインによる翻訳文には<unk>が含まれる結果となった．一方で提案手法では未知語が含まれる *shape of moldings* を *extrusion form* というフレーズへと言いかえることができている．

実験では単語分散表現として一般的なドメインの単語分散表現を使用し，提案手法の有効性を確認した．この単語分散表現のドメインをより学習データに適応させる事によってさらなる精度改善が期待できるが，これは今後の課題としたい．

[*2] http://workshop2017.iwslt.org/
[*3] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/tokenizer/tokenizer.perl
[*4] https://code.google.com/archive/p/word2vec/
[*5] https://dumps.wikimedia.org
[*6] https://radimrehurek.com/gensim/

表 3: ASPEC による実験結果．括弧内の数値は ℓ_{ent} からの差分を表す．また，太字で表されている BLEU スコアは ℓ_{ent} との差分が統計的に有意であることを示す（$p < 0.01$）．

target vocab.	loss	pre-train	BLEU	METEOR
20,000	ℓ_{ent}	None	24.91	30.71
10,000	ℓ_{ent}	None	23.78	29.39
	$\ell_{ent} + \ell_{emb}$	None	**24.75** (+0.97)	29.93 (+0.54)
	$\ell_{ent} + \ell_{emb}$	ℓ_{ent}	**24.60** (+0.82)	29.52 (+0.13)
	ℓ_{emb}	ℓ_{ent}	**24.85** (+1.07)	29.81 (+0.41)
1,000	ℓ_{ent}	None	14.21	18.43
	$\ell_{ent} + \ell_{emb}$	None	14.35 (+0.14)	18.66 (+0.23)
	$\ell_{ent} + \ell_{emb}$	ℓ_{ent}	**14.72** (+0.51)	18.88 (+0.45)
	ℓ_{emb}	ℓ_{ent}	**14.74** (+0.53)	20.15 (+1.72)

表 4: IWSLT17 による実験結果．括弧内の数値は ℓ_{ent} からの差分を表す．また，太字で表されている BLEU スコアは ℓ_{ent} との差分が統計的に有意であることを示す（$p < 0.01$）．

target vocab.	loss	pre-train	BLEU	METEOR
10,000	ℓ_{ent}	None	33.89	56.37
	$\ell_{ent} + \ell_{emb}$	None	33.94 (+0.05)	57.20 (+0.83)
	$\ell_{ent} + \ell_{emb}$	ℓ_{ent}	**35.46** (+1.57)	58.35 (+1.98)
	ℓ_{emb}	ℓ_{ent}	**35.60** (+1.72)	58.35 (+1.99)

5 まとめ

本論文では，機械翻訳における正解単語との単語分散表現での距離を生成確率による重み付き平均したものを誤差とする誤差関数を提案した．実験において提案手法は BLEU と METEOR の 2 つの評価尺度による翻訳精度の改善を示し，語彙サイズが限られた環境においても未知語を同義語に言い換えられていることが確認できた．

今後の課題としては，誤差関数の計算の高速化や word2vec 以外の単語分散表現の使用，人間による主観評価を含む更に詳細な評価，subword の利用などが挙げられる．

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．

参考文献

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

[2] Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. In Proceedings of EMNLP, pp. 1412–1421, September 2015.

[3] Taku Kudo. Subword regularization: Improving neural network translation models with multiple subword candidates. In Proceedings of ACL, pp. 66–75, 2018.

[4] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural Machine Translation of Rare Words with Subword Units. In Proceedings of ACL, pp. 1715–1725, Berlin, Germany, August 2016.

[5] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In ICLR Workshop, 2013.

[6] Diederik P. Kingma and Jimmy Lei Ba. Adam: a method for stochastic optimization. In Proceedings of ICLR2016, 2015.

[7] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In Proceedings of ACL, pp. 311–318, Philadelphia, Pennsylvania, USA, July 2002.

[8] Michael Denkowski and Alon Lavie. Meteor universal: Language specific translation evaluation for any target language. In Proceedings of the EACL 2014 Workshop on Statistical Machine Translation, 2014.

[9] Toshiaki Nakazawa, Manabu Yaguchi, Kiyotaka Uchimoto, Masao Utiyama, Eiichiro Sumita, Sadao Kurohashi, and Hitoshi Isahara. Aspec: Asian scientific paper excerpt corpus. In Proceedings of LREC 2016, pp. 2204–2208, Portoro, Slovenia, may 2016.

[10] Graham Neubig, Yosuke Nakata, and Shinsuke Mori. Pointwise prediction for robust, adaptable japanese morphological analysis. In Proceedings of ACL-HLT, pp. 529–533, Portland, Oregon, USA, June 2011.