文字レベルの日中・中日ニューラル機械翻訳における文字分解による低頻度文字の削減

張 津一　　松本 忠博
岐阜大学 大学院 工学研究科
{zhang, tad}@mat.info.gifu-u.ac.jp

1 はじめに

機械翻訳の精度は，ニューラル機械翻訳（Neural Machine Translation, NMT）の登場によって従来の統計的機械翻訳から大きく向上した．NMT における問題の一つとして低頻度語の扱いがあげられる．NMT モデルでは語彙サイズが大きくなると計算量が膨大になるため，一般的な単語レベルの NMT では通常数万語程度に語彙サイズを制限し，残りの低頻度語は一様に未知語として扱う．未知語の増加は翻訳精度の低下に繋がるため，低頻度語の扱いが問題となっている．その対策として，常用される文字が少なく比較的長い単語の多い欧州の言語を対象に，バイト対符号化（BPE）などによって単語を出現頻度の高い部分文字列に分割する部分単語レベルの NMT が提案され [1]，利用されるようになった．

しかし，表語文字である漢字を主に使用する中国語では 1〜2 文字の単語が多く，文字の種類も多いため，単語を高頻度の部分文字列に分割するのは難しい．同じく漢字を利用する日本語との間の翻訳では文字レベルの NMT が適していると考えられる．文字レベルの NMT では，文を単語に分割する過程での誤りや揺れが生じないという利点もある．

単語レベルに比べ，文字レベルの学習では語彙サイズ（文字の異なり数）は低く抑えられるが，それでも訓練データ中の出現頻度が極端に低い文字が数多く存在する．単語レベルの学習では，統計的信頼性の低い低頻度語を，関連する高頻度の別の語で置き換える手法が考えられているが，文字ではそのような置き換えは難しい．そこで本研究では，低頻度文字をその文字の構成要素（偏旁：形成文字や会意文字では意味や音を表す）と擬似的な部分文字に分解し，それらを他の文字と共有することで低頻度文字を削減する方法を考案し，実験により効果を調べた．

NMT モデルとして Luong ら [2] のものを，実験用データとしてアジア学術論文抜粋コーパス（ASPEC）[3] を用いて評価実験を行った．

2 NMT システムと ASPEC-JC コーパス

本研究で使用した NMT システムと ASPEC-JC コーパスについて簡単に述べる．

2.1 NMT システム

本研究では Luong ら [2] によるグローバル注意機構付きエンコーダ・デコーダモデルを実装した NMT システムを文字レベルで使用する．エンコーダは，双方向 LSTM リカレントニューラルネットワークであり，入力列 $x = (x_1, \ldots, x_m)$ を読み取って，順方向の隠れ状態列 $(\overrightarrow{h}_1, \ldots, \overrightarrow{h}_m)$ と逆方向の隠れ状態列 $(\overleftarrow{h}_1, \ldots, \overleftarrow{h}_m)$ を求める．隠れ状態 $\overrightarrow{h_j}$ と $\overleftarrow{h}_j$ は連結され，アノテーションベクトル h_j が作られる．デコーダは，目的言語文 $y = (y_1, \ldots, y_n)$ を予測する LSTM リカレントニューラルネットワークである．各単語（本研究では文字）y_i は，一つ過去の隠れ層の状態 s_i と，前回予測された単語（文字）y_{i-1}，文脈ベクトル c_i を元に予測される．文脈ベクトル c_i は，アノテーション h_j の加重和として計算される．各 h_j の重みは，y_i と x_j のアラインメントモデル α_{ij} によって決められる．

NMT システムの実装としては OpenNMT[4] を用いた．

2.2 ASPEC-JC コーパス

本研究では日中対訳コーパスとして，Asian Scientific Paper Excerpt Corpus（ASPEC）[3] の日中学術論文抜粋コーパス（ASPEC-JC）を使用した．

このコーパスは医学，情報，生物，環境，化学，材料，農業，エネルギー分野の論文の和文妙録とその中国語訳から成る．コーパスは train（672,315 文対），dev

(2,090 文対)，dev-test（2,148 文対），test（2,107 文対）の 4 つデータセットで構成されており[*1]，互いに同じ論文に属する文は含まれていない．

本研究では train を学習データ，dev をバリデーションデータ，test をテストデータとして使用した．

3　文字の分解による低頻度文字の削減

学習データの中には極めて出現頻度の低い文字が多く含まれており，それが翻訳精度に影響を与えている可能性がある．本研究では低頻度の文字（主に漢字を想定）を高頻度の文字と擬似的な文字を使って分解し，擬似的な文字を複数の低頻度文字間で共有することで，一定以下の頻度の文字をなくす方法を考案し，翻訳精度への影響について実験により調べた．以下，低頻度文字の分解方法について述べる．

3.1　文字の分解

漢字はその成り立ちから，物の形をかたどって造られた象形文字（例：月，山），抽象的概念を字形で表した指事文字（例：一，上），既存の文字の意味を組み合わせて造られた会意文字（例：休，森），意味的なカテゴリを表す「意符」と音を表す「音符」から成る形声文字（例：銅，江）などに分類される．漢字全体の 80% 以上が形声文字と言われており，会意文字と形声文字の特徴を併せ持つ場合も多い．このように漢字の多くは複数の部分文字（偏旁）で構成される．

低頻度の文字であってもそれを構成する部分文字の一部は高頻度文字であることがある．例えば，ASPEC-JC の訓練データ（日本語）では「楡」の出現頻度は 16 だが，その構成要素の「木」は 7,780，「俞」は 11 である．ほかにも木偏を持つ低頻度の漢字があれば，「木」の出現頻度はさらに高くなる．高頻度の部分文字は部首である可能性が高く，一般に形成文字において部首はその漢字の意味と関連していると考えられる．

そこで，本研究では後述の漢字構成表（3.2 節）により低頻度文字を 2 つの部分文字に分解（構成要素が 3 つ以上の場合は，1 番目とそれ以外に二分）し，訓練データの中で出現頻度の高い方の部分文字（部首の可能性が高い）はそのまま，低い方の部分文字は擬似的な部分文字（s_1, s_2, ..., s_n）で置き換える．次のように低頻度文字間で擬似文字を共有することでその出現頻度を高める．

例）楡 → (木, s_1)，桝 → (木, s_2)
　　炒 → (火, s_1)，焔 → (火, s_2)

文字の構成要素にも頻度の大きな差がある．したがって，対になる擬似文字の添字の最大値もまちまちになり，頻度の低い擬似文字が現れうる．それを避けるため，擬似文字の添字の上限を定め，上限に収まらない場合は，高頻度側の部分文字も擬似文字で置き換える．また，擬似文字は出現頻度が偏らないように添字の開始番号を変化させる．

例）榊 → (s_{13}, s_{16}), 枷 → (s_{19}, s_{22})

なお，漢字構成表に分解方法が記載されていない漢字については，「漢」と擬似文字の対に置き換える．また，漢字以外の低頻度文字については，仮名は「仮」，その他記号などは「符」と擬似文字の対に置換する．

このように頻度 k 以下の低頻度文字を高頻度文字と擬似文字の対，または擬似文字の対に置き換え，訓練データから頻度 k 以下の文字をなくす．

学習と翻訳は次の手順で行う．低頻度文字から部分文字の対へのマッピングを，訓練データの日本語側と中国語側とで独立に設定し，分解後のデータで学習を行う．テスト時には，原言語側のテストデータを分解してから翻訳し，翻訳結果中の分解された文字を復元する．復元できなかった文字は空白文字に置き換える．

3.2　漢字構成表の作成

漢字からその構成要素を求めるための表を，Python 漢字ライブラリ cjklib[5] の Chinese character decomposition table を参考に作成した．このほか，CHISE プロジェクト [6] の漢字構造情報データベース，字源 [7]（漢字データベースプロジェクト [8] の配布データを基盤としたオンライン漢字辞典）も補助的に利用した．同じ構成要素を持つ漢字が複数存在する場合は，次のように番号を付けて区別する．

例）　暈 → 日軍 1，　暉 → 日軍 2
　　　杰 → 木示 1，　标 → 木示 2

次の例のように，構成要素が単純な形で分解すると意味が希薄になる場合は構成情報から除外して，分解を行わないようにした．

[*1] 中国訳の内容が “。。” のみの文が，train データと test データに少量含まれるが，本研究ではそれらを除外して使用した．

例） 七 → 乚 一， 么 → 丿 厶

4 翻訳実験

4.1 NMT システムの設定と翻訳結果の評価法

翻訳システムの実装には OpenNMT を用いた．モデルのパラメータは (− 0.1, 0.1) の範囲の一様乱数で初期化し，最適化にはデフォルトの確率的勾配降下法を用いた．学習率はエポック 6 までは 1.0 とし，それ以降はエポックごとに 0.5 倍する．最大勾配ノルムは 1，最大バッチサイズは 100 とした．また，リカレント層は 1 層で，単語ベクトルと隠れ層の次元は 512 とした．dropout 確率は 0.5 に設定し，デコード時のビームサイズは 5 とした．文の最大長は，デフォルトでは 250 だが，文字レベルでは長くなるため 500 に設定した．

学習と翻訳は文字レベルで行うが，評価は単語レベルのシステムと同じ条件で行うため，日本語文は MeCab[*2]，中国語文は jieba[*3]で単語ごとに分割した後，OpenNMT 付属の multi-bleu.perl で BLEU スコアを算出した．

多くの場合，エポック 10 前後で validation perplexity（dev データでの perplexity）が下げ止まった．その時点からエポック 16 までの BLEU スコアの平均を評価値とした．ベースラインは何も加工しない訓練データによる文字レベルの翻訳である．

4.2 実験結果

■BLEU スコアの変化 3 節で述べた文字分解手法により訓練データから低頻度文字をなくしたときの BLEU スコアの変化を図 1 に示す．ベースラインの最低文字出現頻度は 1 である．擬似文字の添字の上限は両言語とも基本的に 55 に設定したが，中国語データで最低頻度 7000 以上に設定すると擬似文字が不足したため 60 に設定した．

目的言語が中国語となる日中翻訳では，最低頻度を 10 から 120 の間のとき，ベースラインを上回る結果になることが多かった（平均 0.08%）．最低頻度を 20 に設定したときに 0.29% 向上した．一方，目的言語が日本語となる中日翻訳では，ベースラインを上回る結果になることが少なかったが，最低頻度を 150 に設定したときだけ 0.23% 向上した．

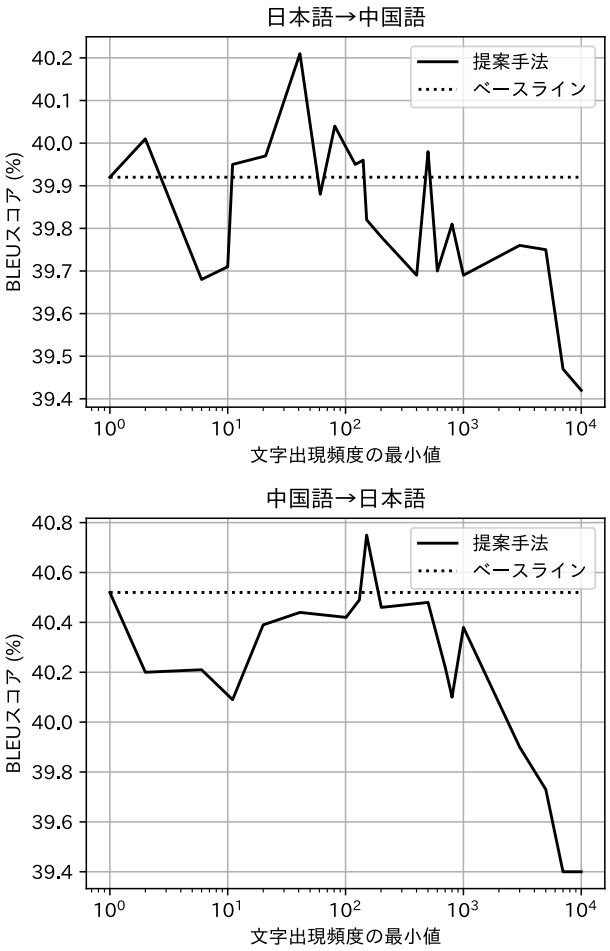

図 1 低頻度文字の削減による BLEU スコアの変化

訓練データにおける文字種の構成は，中国語では漢字が 78% を占めるのに対し，日本語では仮名文字が 47% で，漢字は 36% にすぎない．この違いが結果に影響していることが考えられる．また，訓練データに含まれる文字の異なり数（NMT システムにとっての語彙サイズ）にも，中国語 6,088 に対して，日本語 4,249 と大きな差があることも関係している可能性がある．

■学習時間の変化 低頻度文字を減らすことで語彙サイズ（文字の異なり数）が減少するため，学習するパラメータの数も減少する．これにより学習時の使用メモリ量や学習時間の減少が期待できる．学習時のログから得られたパラメータ数と 1 エポック当りの平均学習時間の変化を図 2 に示す．CPU や GPU の構成の異なる複数のシステム上で実験を行なったため，結果は各システムでのベースライン学習時の値を 1 とした相対値で表している．

文字分解によりパラメータ数は減るが，処理する文

[*2] http://taku910.github.io/mecab/
[*3] https://github.com/fxsjy/jieba

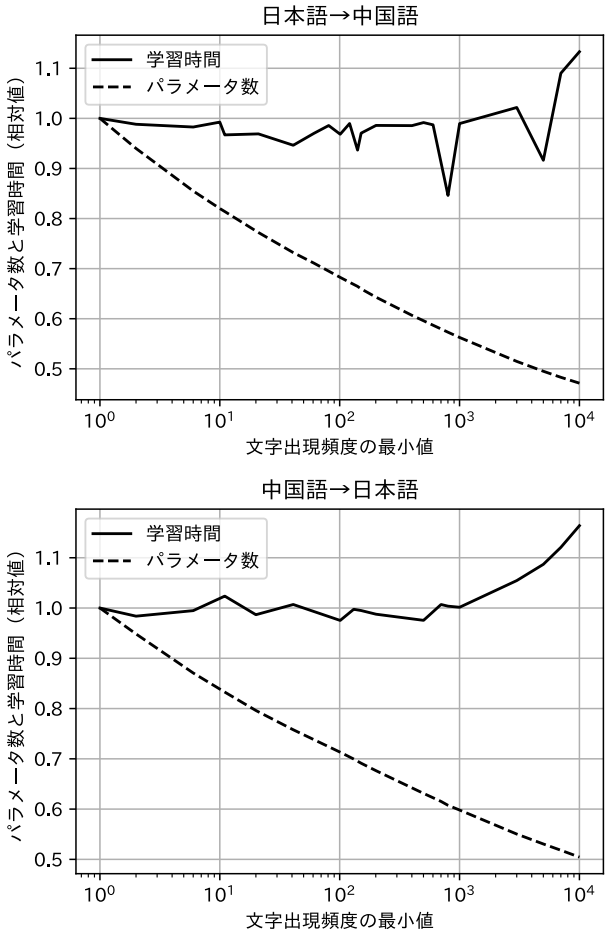

図 2　低頻度文字の削減によるパラメータ数と 1 エポック当りの平均学習時間の変化

字の数は増える．日中翻訳では，最低頻度が 1000 までは常にベースラインより学習時間が短くなり，10 から 1000 の範囲では平均 3.56% 短くなった．一方，中日翻訳では学習時間短縮の効果はあまり見られなかった．

5　おわりに

本研究では漢字構成表を作成し，低頻度文字を漢字構成要素と擬似的な文字に分解することで低頻度文字を削減する手法を考案した．日中・中日 NMT モデルを用いた実験により，文字の最低頻度による BLEU スコアと学習時間の変化を調べたところ，ベースラインの NMT と比較して，BLEU 値は日→中で最大約 0.3%，中→日で 0.2% 上回ることがあった．しかし，とくに中→日ではほとんどの場合，ベースラインを下回る結果となった．学習時間は，日→中では最小出現回数が 1000 以下のとき，概ねベースラインより短くなった．もし翻訳精度を改善するための文字分解方法の条件が見出せれば，中国語から他の言語へ，あるいは，日本語から他の言語への文字レベルの翻訳においても，漢字構成要素の利用が翻訳精度の向上につながる可能性があると考えられる．

謝辞

著者の一人である張津一は中国国家留学基金委の助成 (No.201708050078) を受けている．ここで感謝の意を表する．

参考文献

[1] R. Sennrich, B. Haddow, and A. Birch, “Neural machine translation of rare words with subword units,” Proc. 54th Annual Meeting of the Assoc. for Computational Linguistics, pp.1715–1725, Berlin, Germany, Aug. 2016.

[2] M.T. Luong, H. Pham, and C.D. Manning, “Effective approaches to attention-based neural machine translation,” Proc. 2015 Conf. on Empirical Methods in Natural Language Processing, pp.1412–1421, ACL, Lisbon, Portugal, 2015.

[3] T. Nakazawa, M. Yaguchi, K. Uchimoto, M. Utiyama, E. Sumita, S. Kurohashi, and H. Isahara, “ASPEC: Asian scientific paper excerpt corpus,” Proc. 9th Int. Conf. on Language Resources and Evaluation (LREC 2016), pp.2204–2208, European Language Resources Association (ELRA), Portorož, Slovenia, 2016.

[4] G. Klein, Y. Kim, Y. Deng, J. Senellart, and A.M. Rush, “OpenNMT: Open-source toolkit for neural machine translation,” ArXiv e-prints, p.1, 2017.

[5] cjklibdevelopers, “cjklib — han character library,” http://cjklib.org/, Aug. 2010.

[6] CHISE プロジェクト，“文字情報サービス環境 CHISE,” http://www.chise.org/，Aug. 2018.

[7] jigen.net, “漢字と古典の総合サイト，字源,” http://jigen.net/，July 2018.

[8] 漢字データベースプロジェクト，“漢字データベース,” http://kanji-database.sourceforge.net/． 参照 July 9, 2018.