# 平成29年度AAMT/Japio特許翻訳研究会
# 報 告 書

## 機械翻訳及び機械翻訳評価に関する研究
## 並びに
## 特許・技術文書翻訳ワークショップ報告

平成 30 年 3 月

一般財団法人　日本特許情報機構

# 目　次

# AAMT/Japio 特許翻訳研究会委員名簿

（敬称略・順不同）

| | | |
|---|---|---|
| 委員長 | 辻井　潤一（※2） | 国立研究開発法人 産業技術総合研究所<br>人工知能研究センター 研究センター長 /<br>東京大学 名誉教授 |
| 副委員長 | 宇津呂武仁（※2） | 筑波大学大学院 教授 |
| | 須藤　克仁（※2） | 奈良先端科学技術大学院大学 情報科学研究科 准教授 |
| 委員 | 磯崎　秀樹（※1） | 岡山県立大学 教授 |
| | 今村　賢治 | 国立研究開発法人 情報通信研究機構<br>先進的音声翻訳研究開発推進センター 主任研究員 |
| | 越前谷　博（※2） | 北海学園大学大学院 教授 |
| | 江原　暉将（※2） | 元・山梨英和大学 教授 |
| | 熊野　　明 | 東芝デジタルソリューションズ株式会社 ICT インフラサービスセンター<br>ソフトウェア開発部 |
| | 黒橋　禎夫 | 京都大学大学院 教授 |
| | 後藤　功雄（※2） | NHK 放送技術研究所<br>ヒューマンインターフェース研究部 専任研究員 |
| | 下畑　さより | 沖電気工業株式会社 情報通信事業本部<br>ソフトウェアセンターサービス業務管理部 |
| | 綱川　隆司 | 静岡大学学術院 助教 |
| | 中澤　敏明（※2） | 国立研究開発法人 科学技術振興機構 情報企画部 研究員 /<br>京都大学 大学院情報学研究科 知能情報学専攻 研究員 |
| | 二宮　　崇 | 愛媛大学大学院 准教授 |
| | 横山　晶一 | 山形大学 名誉教授 |
| オブザーバ | 潮田　　明 | 国立研究開発法人 産業技術総合研究所 人工知能研究センター |
| | 高　　京徹 | 株式会社高電社 経営企画部 部長 |
| | 園尾　　聡（※2） | 東芝デジタルソリューションズ株式会社<br>RECAIUS 事業推進室 要素開発部 要素開発第二担当 |
| | 長瀬　友樹（※2） | 株式会社富士通研究所 メディア処理研究所 主管研究員 |
| | 王　　向莉（※2） | 株式会社ディープランゲージ |
| | 守屋　敏道 | 一般財団法人日本特許情報機構 顧問 |
| | 小林　　明 | 一般財団法人日本特許情報機構 専務理事 / 特許情報研究所 所長 |
| | 横井　巨人 | 一般財団法人日本特許情報機構 特許情報研究所 調査研究部 部長 |

| | | |
|---|---|---|
| | 大塩　只明 | 一般財団法人日本特許情報機構 特許情報研究所 調査研究部<br>総括研究主幹 |
| | 木下　　聡 | 一般財団法人日本特許情報機構 特許情報研究所 調査研究部<br>研究主幹 |
| | 三橋　朋晴 | 一般財団法人日本特許情報機構 特許情報研究所 調査研究部<br>研究管理課 課長 |
| | 白土　博之 | 一般財団法人日本特許情報機構 特許情報研究所 調査研究部<br>研究企画課 課長 |
| | 小川　直彦 | 一般財団法人日本特許情報機構 特許情報研究所 研究管理部<br>研究管理課 係長 |
| | 星山　直人 | 一般財団法人日本特許情報機構 情報運用部 情報整備課 係長 |
| | 土屋　雅史 | 一般財団法人日本特許情報機構 情報運用部 情報運用課 主任 |
| | 船戸　さやか | 一般財団法人日本特許情報機構 特許情報研究所 調査研究部<br>研究企画課 副主任 |

（※1：拡大評価部会部会長、※2：拡大評価部会メンバー）

| | | |
|---|---|---|
| 事務局 | 小松　浩平 | 株式会社インターグループ |
| | 佐藤　伶奈 | 株式会社インターグループ |

追悼

岡山県立大学 磯崎 秀樹 教授におかれましては、平成30年2月に急逝されました。

先生は、平成25年度より研究会オブザーバー及び拡大評価部会メンバーとなり、平成27年度以降は研究会委員及び拡大評価部会長を務められ、機械翻訳の評価に関する議論を主導されました。

ご生前のご功績を偲び、心からご冥福をお祈りいたします。

# 平成 29 年度 AAMT/Japio 特許翻訳研究会・活動履歴

平成 29(2017)年 5 月 12 日
第 1 回 AAMT/Japio 特許翻訳研究会、第 1 回拡大評価部会
（於キャンパス・イノベーションセンター東京）

平成 29(2017)年 6 月 16 日
第 2 回 AAMT/Japio 特許翻訳研究会
（於キャンパス・イノベーションセンター東京）

平成 29(2017)年 7 月 21 日
第 3 回 AAMT/Japio 特許翻訳研究会
（於キャンパス・イノベーションセンター東京）

平成 29 年(2017)年 9 月 18 日~9 月 22 日
第 16 回機械翻訳サミット、特許・技術文書翻訳ワークショップ（PSLT2017）（於日本（名古屋）名古屋大学）

平成 29(2017)年 10 月 13 日
第 4 回 AAMT/Japio 特許翻訳研究会、第 2 回拡大評価部会
（於キャンパス・イノベーションセンター東京）

平成 29(2017)年 12 月 8 日
第 5 回 AAMT/Japio 特許翻訳研究会
（於キャンパス・イノベーションセンター東京）

平成 30(2018)年 1 月 19 日
第 6 回 AAMT/Japio 特許翻訳研究会
（於キャンパス・イノベーションセンター東京）

平成 30(2018)年 3 月 2 日
第 7 回 AAMT/Japio 特許翻訳研究会、第 3 回拡大評価部会
（於キャンパス・イノベーションセンター東京）

平成 30(2018)年 3 月 31 日
『平成 29 年度 AAMT/Japio 特許翻訳研究会報告書 機械翻訳及び機械翻訳評価に関する研究並びに特許・技術文書翻訳ワークショップ報告』完成

# 1. はじめに

AAMT/Japio　特許翻訳研究会委員長
産業技術総合研究所人工知能研究センター　研究センター長
辻井　潤一

前年度の報告書でも述べたが、ニューラルネットワークを使った機械翻訳の実用化が急速に進展し、人間の翻訳家を含む翻訳業界に大きな影響を及ぼしつつある。これまでの規則による機械翻訳、統計モデルを使った機械翻訳などに比べると、はるかに流ちょうな翻訳が出力される。このために、専門分野の翻訳を行った場合、専門分野の知識を持たない翻訳家では、たとえ翻訳結果に誤訳が含まれていても、その誤りに気が付かずにスルーしてしまう可能性がある。機械翻訳に翻訳の第一稿を作らせ、それを人間が後修正するという、従来の機械援助型の翻訳ワークフローでも後修正を行う能力の高い翻訳家が必要であったが、ニューラル翻訳を使った翻訳ワークフローでは、この傾向がさらに顕著になっていくと考えられる。後修正を行う翻訳家には、言語的な能力だけでなく、テキストが取り扱っている専門分野に関する背景知識を有していることが要求されることになろう。

また、ニューラル翻訳を特許翻訳のような専門性の高いテキストに適用する場合の問題点も、徐々に明らかになってきている。特許翻訳など、専門性が高いテキストには専門用語が頻出するが、膨大な用語辞書を作ったとしても、未知語の出現は避けられない。また、ニューラル翻訳では、原言語テキストの専門用語が、翻訳結果のどの専門用語に翻訳されたのかが間接的にしか把握できない。このことは、大きな専門用語辞書を構築しても、その辞書をニューラル翻訳でどのように使うのかが自明ではなくなっている。専門用語辞書を、ニューラル翻訳で活用する技術の開発が望まれる。

以上のように、ニューラル翻訳という新たな技術が出てきたことで、これまでとは別の技術課題が生まれてきている。本委員会での活動が、今後ますます重要になると考えている次第である。1 年間の活動をまとめた本報告書が、激しく変貌を遂げつつある機械翻訳、特許翻訳の新たな技術的課題、また、本委員会がこれらの課題にどのように取り組もうとしているかを知っていただく一助になれば幸いである。

最後になりますが、本委員会の主要メンバーであった磯崎秀樹先生が、本年２月に急逝されました。先生には、本委員会での活動だけでなく、本委員会である翻訳評価部会の部会長として、大変なご尽力をいただきました。残されたご親族に、心よりお悔やみ申し上げます。

# 2. 機械翻訳および関連技術

# 2.1 CKY アルゴリズムに基づく畳み込みニューラルネットワークによるニューラル機械翻訳

愛媛大学　二宮 崇
愛媛大学　田村 晃裕
愛媛大学　渡邊 大貴

## 2.1.1 はじめに

近年，ニューラルネットワークに基づくニューラル機械翻訳（NMT）が単純な構造で，高い精度を実現し、人間に近い翻訳を行うことで知られている．NMT は，エンコーダデコーダモデルが盛んに研究されており，gated recurrent units（GRUs）(Cho, van Merrienboer, Gulcehre, Bahdanau, Bougares, Schwenk, and Bengio 2014b)や long short term memory（LSTM）(Hochreiter and Schmidhuber 1997; Gers, Schmidhuber, and Cummins 2000）といったリカレントニューラルネットワーク（RNNs）を用いて，原言語文を固定長のベクトルに変換し，その後そのベクトルから目的言語文を生成する（Sutskever, Vinyals, and Le 2014）．アテンションに基づく NMT (ANMT）は，エンコーダデコーダモデルを拡張したものであり，非常に正確な翻訳を行う機械翻訳の最先端技術の一つである（Luong, Pham, and Manning 2015; Dzmitry, KyungHyun, and Yoshua 2015）．ANMT は，デコーダがエンコーダの隠れ層の状態の履歴を参照しながら目的言語文を生成する翻訳手法である．エンコーダデコーダモデルは，さらに，原言語文，目的言語文，またはその両方の統語構造に基づく NMT に拡張されている．特に（Eriguchi, Hashimoto, and Tsuruoka 2016b）の手法は NMT の英日機械翻訳において原言語側の構造が有用であることを示している．しかし，統語構造に基づく NMT は事前に，構文解析を必要とする．本研究では，NMT において，構文解析を行わず原言語文の構造を活用する，畳み込みニューラルネットワーク（CNNs）に基づく新しいアテンション構造を提案する．CKY アルゴリズム(Kasami 1965; Younger 1967）では，CKY テーブルを介してボトムアップに文を解析する．これは動的計画法により効率的にすべての可能な単語の組み合わせを考慮し，文の構造を表現する．このアルゴリズムに基づき提案手法は CKY テーブルを模倣する CNNs を ANMT のアテンション構造へ組み込む．具体的には，提案手法のアテンション構造は CKY テーブルの計算手順と同様の順序で CNNs を構築した後，ANMT は CKY テーブルの各セルに格納された CNNs 隠れ層の状態を参照することにより目的言語文を生成する．提案手法のアテンション構造は ANMT モデルが事前に構文解析を行うことなく，目的言語の各単語を予測するために有用な原言語文の構造をとらえることを可能にする．実験では，ASPEC の英日翻訳タスクの評価(Nakazawa, Yaguchi, Uchimoto, Utiyama, Sumita, Kurohashi, and Isahara 2016）において，提案手法は 0.66 ポイント BLEU スコアが上昇することを示す．

CNNs を用いた NMT には様々な先行研究が存在する（Kalchbrenner and Blunsom 2013; Cho, Van Merrienboer, Bahdanau, and Bengio 2014a; Lamb and Xie 2016; Kalchbrenner,

Espeholt, Simonyan, Oord, Graves, and Kavukcuoglu 2016)．それらのモデルは一次元画像処理のための画像認識のCNNsと同様にエンコーダまたはデコーダに多層のCNNsを直列に接続することで構成されている．しかし，それらのモデルは長距離におけるフレーズ/単語間の接続を直接扱う構造を有していない．提案手法では多層の CNNs に対して CKY に基づく接続を採用しているため，NMT がエンコーダのフレーズ/単語間の接続を直接計算することを可能とする．また，アテンション構造は NMT がエンコーダとデコーダ間の構造的なアライメントをとらえることを可能とする．

### 2.1.2 アテンションに基づく機械翻訳(ANMT)

ANMT (Luong et al. 2015; Dzmitry et al. 2015) はエンコーダデコーダモデル (Sutskever et al. 2014; Cho et al. 2014b) の拡張である．このモデルは，RNN エンコーダを使用して原言語文を固定長のベクトルに変換し，その後 RNN デコーダがベクトルから目的言語文を生成する．

提案手法では，2 層の双方向 LSTM をエンコーダとして使用した．原言語文$x = x_1, x_2, \dots, x_t$が与えられたとき，エンコーダは$i$ 番目の単語$x_i$を単語埋め込み層によって$d$次元ベクトル$v_i$として表現する．その後，エンコーダは$v_i, h_i$の隠れ状態を次式により計算する．

$$\overrightarrow{h_i^{(1)}} = LSTM^{(1)}(v_i) \quad \cdots (1)$$

$$\overleftarrow{h_i^{(1)}} = LSTM^{(1)}(v_i) \quad \cdots (2)$$

$$\overrightarrow{h_i^{(2)}} = LSTM^{(2)}\left(\overrightarrow{h_1^{(1)}}\right) + \overrightarrow{h_1^{(1)}} \quad \cdots (3)$$

$$\overleftarrow{h_i^{(2)}} = LSTM^{(2)}\left(\overleftarrow{h_i^{(1)}}\right) + \overleftarrow{h_i^{(1)}} \quad \cdots (4)$$

$$h_i = \overrightarrow{h_i^{(2)}} + \overleftarrow{h_i^{(2)}} \quad \cdots (5)$$

→ と ← はそれぞれ順方向と逆方向を示す．$LSTM^{(1)}$と$LSTM^{(2)}$はそれぞれ第 1 層および第 2 層の LSTM エンコーダを表す．$\overrightarrow{h_i^{(1)}}, \overleftarrow{h_i^{(1)}}, \overrightarrow{h_i^{(2)}} \overleftarrow{h_i^{(2)}}, h_i$の次元は$d$である．

ANMT では，デコーダは，LSTM エンコーダの隠れ層の状態 $h_i$ を参照し，目的言語文を生成する．以下で説明するアテンション構造は $global\ attention\ (dot)$ と呼ばれる (Luong et al. 2015)．提案手法では，デコーダとして 2 層の LSTM を使用する．第 1 層と第 2 層の LSTM デコーダの

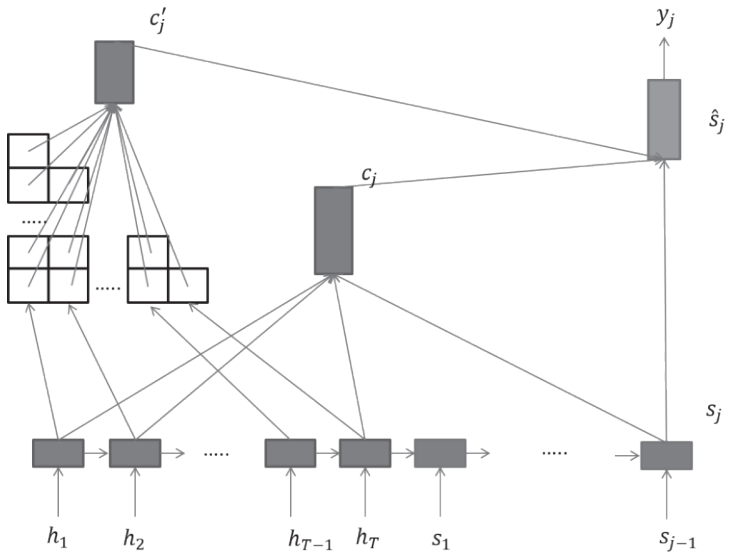



**図 1**　CKY に基づくアテンション構造の全体図

初期状態は，それぞれ逆方向の第 1 層および第 2 層の LSTM エンコーダの内部状態を用いて初期化される．各 LSTM デコーダの隠れ層の状態$s_j^{(1)}$と$s_j^{(2)}$は次式により計算される．

$$s_j^{(1)} = LSTM^{(1)}([w_{j-1}; \hat{s}_{j-1}]) \quad \cdots (6)$$

$$s_j^{(2)} = LSTM^{(2)}(s_j^{(1)}) \quad \cdots (7)$$

“

ここで$w_{j-1}$は出力単語$y_{j-1}$の単語埋め込み，‘;’ は行列の結合，$\hat{s}_{j-1}$は出力単語$y_{j-1}$を生成するために使用されたアテンションベクトルを表す．$w_{j-1}$と$\hat{s}_{j-1}$は$d$次元である．アテンションスコア$\alpha_j(i)$は次式を用いて計算される．

$$\alpha_j(i) = \frac{exp\left(h_i \cdot s_j^{(2)}\right)}{\sum_{k=1}^{T} exp(h_k \cdot s_j^{(2)})} \quad \cdots (8)$$

目的言語文を生成するコンテキストベクトル $c_j$ は次式により計算される．

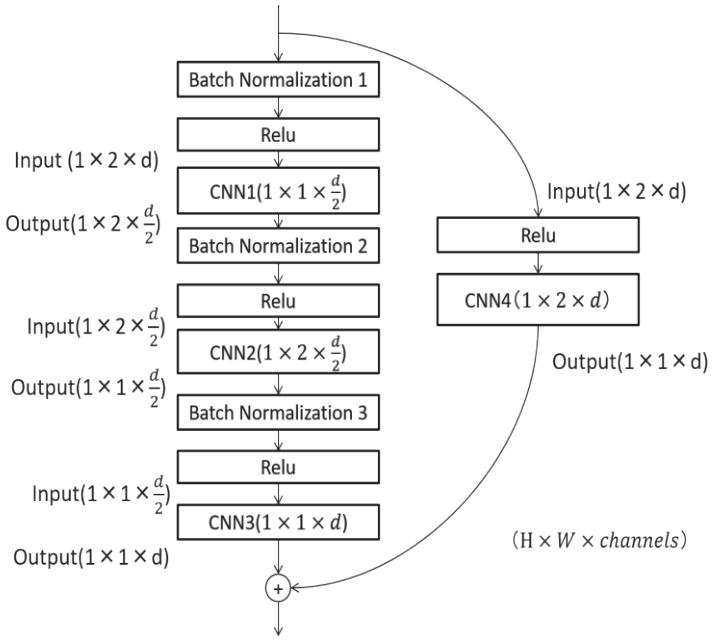



**図 2**　CKY に基づくアテンションの Deduction Unit の構造

$$c_j = \sum_{i=1}^{T} \alpha_j(i) h_i \qquad \cdots (9)$$

アテンションベクトル$\hat{s}_{j-1}$は，コンテキストベクトルを使用して次式により計算される．

$$\hat{s}_j = tanh(W_c \left[ s_j^{(2)}; c_j \right]) \qquad \cdots (10)$$

その後隠れ層の状態を使用して，出力単語$y_j$の確率は次式を用いて計算される．

$$p(y_j | y_{<j}, \boldsymbol{x}) = softmax(W_s \hat{s}_j) \qquad \cdots (11)$$

ここで，$W_c$と$W_S$は重み行列を表す．

### 2.1.3　CKY に基づく畳み込みアテンション構造による機械翻訳

図 1 に提案手法の全体の構造を示す．提案手法のアテンション構造において，CKY アルゴリズムによる生成規則は図 2 に示すネットワーク構造によって模倣される．このネットワークを Deduction Unit（DU）と呼ぶ．DU では，4 つのタイプの CNNs が residual connection によって接続されている．図 2 では各 CNN のフィルタサイズと出力チャネルの数が，括弧内に記され

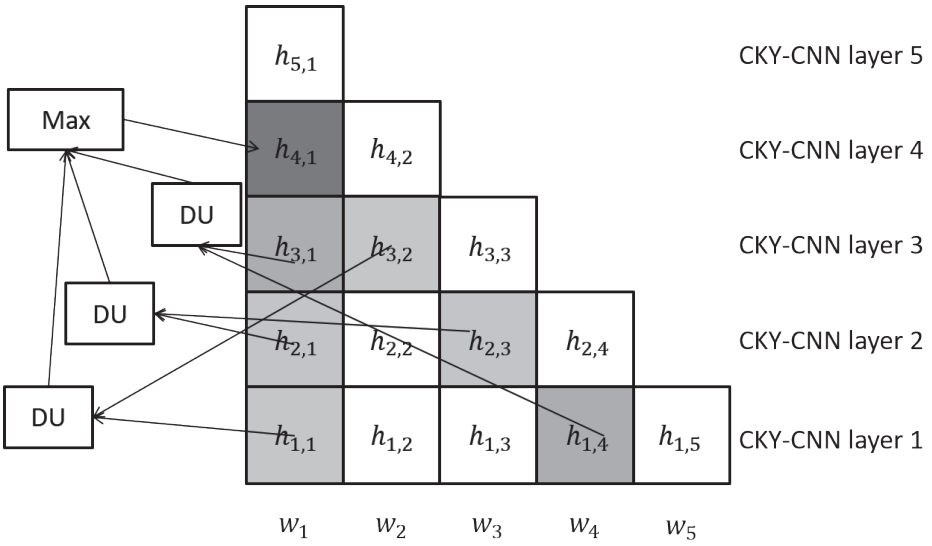



**図 3**　CKY-CNN の Max-pooling の例

ている．CNN1，CNN2，CNN3，CNN4 のフィルタサイズはそれぞれ$1 \times 1, 1 \times 2, 1 \times 1, 1 \times 2$であり，チャネル数は$d/2, d/2, d, d$である．各 DU は，CKY テーブル内の 2 つのセルの$d$次元のベクトルを受け取り，CKY アルゴリズムの生成ルールに対応する上位セルの$d$次元ベクトルを計算する．DU を使用することにより，CKY テーブルの各セルの状態は，CKY アルゴリズムの計算手順と同様の順序で下位セルの状態を畳み込むことにより導出される．この全体のネットワークを CKY-CNN と呼ぶ．以下，$i$番目の CKY-CNN 層の$j$番目のセルの状態を$h_{i,j}^{(cky)}$と定義する．ここで CKY-CNN の第 1 層の状態($\boldsymbol{h_1^{(cky)}} = \left(h_{1,1}^{cky}, \ldots, h_{1,T}^{(cky)}\right)$)は, LSTM エンコーダ($\boldsymbol{h} = (h_1, \ldots, h_T)$)の状態に設定される．CKY-CNN では，セルの状態は CKY アルゴリズムと同様に，DUs の複数の出力候補からセルの状態が導出される．具体的には，セルの状態はすべての次元の合計値が最大である出力ベクトルに設定される．これは次の式で計算される．

$$h_{i,j}^{(cky)} = Max_{1 \le k \le i-1} DU(h_{k,j}^{(cky)}, h_{i-k,j+k}^{(cky)}) \qquad \cdots (12)$$

図 3 は CKY-CNN における，図中の $h_{4,1}$ のセルを生成する畳み込みの例を示している．この生成過程では, 3 つの DU が, それぞれ 2 つのセル$\left(h_{3,1}, h_{1,4}\right)$, 2 つのセル$\left(h_{2,1}, h_{2,3}\right)$, 2 つのセル$\left(h_{1,1}, h_{3,2}\right)$の状態に基づいてベクトルを生成する. その後, ベクトルの要素和が最も大きいベクトルが $h_{4,1}$ のセルの状態に設定される．CKY-CNN を介して，CKY テーブル ($\boldsymbol{h^{(cky)}}$) 内のセルの状態が得られる．

CKY に基づく畳み込みアテンション構造による NMT では, LSTM エンコーダの隠れ層の状態に加えて，CKY-CNN の隠れ層の状態を参照しながら目的言語文を生成する．アライメントスコ

アは次式により計算される．

$$\alpha'(i,j) = \frac{exp\left(h_i \cdot s_j^{(2)}\right)}{\sum_{k=1}^{T} exp(h_k \cdot s_j^{(2)}) + \sum_{k=1}^{T}\sum_{l=1}^{T-k+1} exp(h_{k,l}^{(cky)} \cdot s_j^{(2)})} \quad \cdots (13)$$

$$\alpha''(i_1,i_2,j) = \frac{exp\left(h_{i_1,i_2}^{(cky)} \cdot s_j^{(2)}\right)}{\sum_{k=1}^{T} exp(h_k \cdot s_j^{(2)}) + \sum_{k=1}^{T}\sum_{l=1}^{T-k+1} exp(h_{k,l}^{(cky)} \cdot s_j^{(2)})} \quad \cdots (14)$$

ここで $s_j^{(2)}$は第 2 層の LSTM エンコーダの隠れ層の状態である．CKY-CNN のコンテキストベクトル$c_j'$は次式により計算される．

$$c_j' = \sum_{k=1}^{T} \alpha'(k,j)h_k + \sum_{k=1}^{T}\sum_{l=1}^{T-k+1} \alpha''(k,l,j)h_{k,l}^{(cky)} \quad \cdots (15)$$

$\hat{s}_j$は LSTM エンコーダのコンテキストベクトル$(c_j)$と CKY-CNN のコンテキストベクトル$(c_j')$を用いて次のように計算される．

$$\hat{s}_j = \tanh(\widehat{W}\left[s_j^{(2)}; c_j; c_j'\right]) \quad \cdots (16)$$

$\widehat{W} \in R^{d\times 3d}$は重み行列である．従来の ANMT と同様に softmax 関数を$\hat{s}_j$ に適用することで， デコーダは $j$番目の目的言語の単語を予測する．

### 2.1.4 実験

#### 2.1.4.1 実験設定

Asian Scientific Paper Excerpt Corpus (ASPEC)[1] の英日コーパスを用いて実験を行った．英語コーパスの単語セグメンテーションのために Moses decoder を使用し，日本語コーパスに対しては Kytea (Neubig, Nakata, and Mori 2011) を使用した．各コーパスについて，すべての文字を小文字に変換した．モデルの学習には 10000 文 (< 50)，テストには 1812 文を使用した．トレーニングデータ中に 2 回未満しか出現していない単語は，特殊文字 UNK に置き換えた．

単語ベクトルと隠れ層のベクトルの次元数は 256 とした．各パラメータの学習には

[1] http://orchid.kuee.kyoto-u.ac.jp/WAT/WAT2015/index.html

表 1 評価結果

| | BLEU(%) |
|---|---|
| ベースラインモデル | 26.09 |
| 提案手法 | 26.75 |

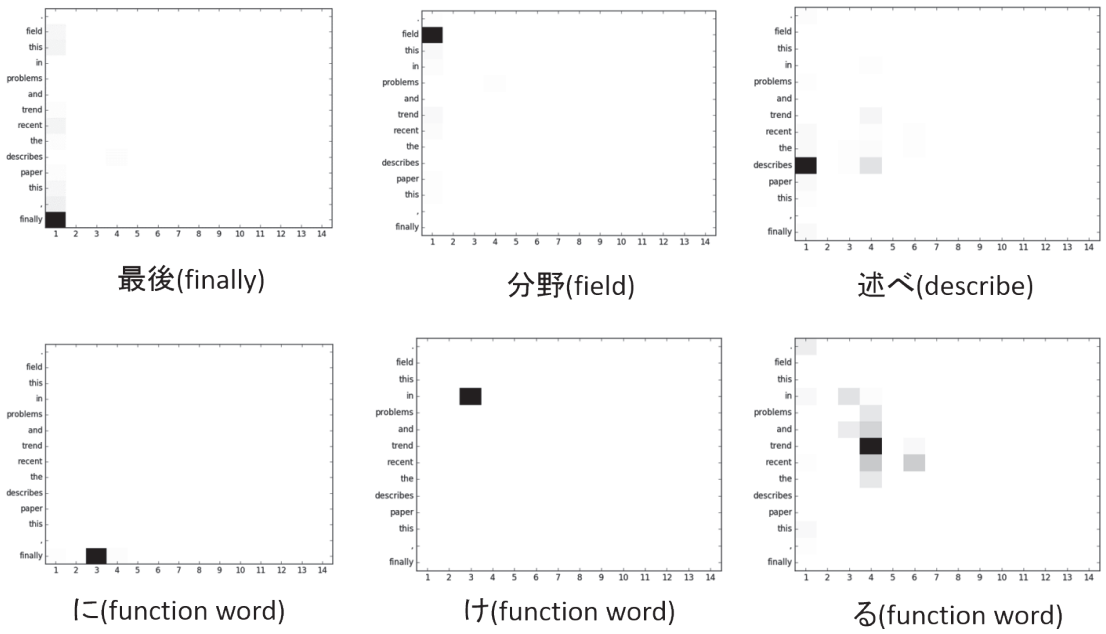



図 4 アテンションスコアの例

Adam (Kingsma and Ba 2014) を使用し，Adam のパラメータの初期値は，$\alpha = 0.01, \beta_1 = 0.9, \beta_2 = 0.99$ とした．学習率は 9 エポックと 12 エポックで半分にした．勾配は (Eriguchi, Hashimoto, and Tsuruoka 2016a) の手法に従い，3.0 でクリップした．過学習を防ぐために dropout (Srivastava, Hinton, Krizhevsky, Sutskever, and Salakhutdinov 2014) と weight decay を用いた．dropout の比率は，LSTMs を 0.2，CNN を 0.3 とし，weight decay の値は$10^{-6}$とした．

#### 2.1.4.2 実験結果

CKY に基づく畳み込みアテンションを用いた NMT と従来の NMT を比較し，CKY に基づくアテンション構造の有用性を確認した．ベースラインと提案手法の違いはアテンション構造である．表 1 に BLEU (Papineni, Roukos, Ward, and Zhu 2002) による翻訳精度を示す．参考のために，Mose フレーズベース統計的機械翻訳 (Koehn, Hoang, Birch, Callison-Burch, Federico, Bertoldi, Cowan, Shen, Moran, Zens, et al. 2007) をデフォルトの設定で使用したところ 18.69% の BLEU スコアが得られた．

表 1 は提案手法がベースラインモデルより優れていることを示し，提案手法のアテンション が

NMT に有用であることを示している．

図 4 はテストデータにおけるある文のアテンションスコアを示す．セルの深い色はより高いアテンションスコアを表している．縦軸は原言語文を示す．図 4 では，テスト中の文 “finally，this paper describes the recent trend and problems in this field .”である．横軸は，CKY-CNN の深さを示す．なお，CKY-CNN の第 1 層のアテンションスコアは，LSTM の隠れ層のアテンションスコアと一致する．図 4 は内容語 (“最後”，“分野”，“述べ”)のようなアライメントが明確に定義された単語は，高いアライメントスコアが第 1 層に位置している．一方，機能語 (“に”，“け”，“る”)のようなアライメントが明確に定義されていない単語は，高いアライメントスコアが深い層に位置している．“は”は格助詞を表し，“け”と“る”は前置詞 “おける”の一部である． これは，従来のアテンション構造が単語レベルでのアライメントを見つけるのに対して，提案手法のアテンション構造は構造的なアライメントを捕らえることを示している．

### 2.1.5 まとめ

本論文では，CKY アルゴリズムを模倣した CNNs に基づく NMT のアテンション構造を提案した. ASPEC 英日翻訳のタスクの評価では, 提案手法は 0.66 ポイント BLEU スコアが上昇し，また，提案手法は従来手法のアテンション構造では捕捉することのできない構造的なアライメントを捕らえることができた．提案モデルでは，CKY テーブルのすべてのセルの隠れ状態を保持するために，多くのメモリ量を使用する．将来，メモリ消費の問題について提案手法を改善し，大規模なデータセットに対する提案手法の有用性を検証したい．

### 参考文献

Cho, K., Van Merrienboer, B., Bahdanau, D., and Bengio, Y. (2014a). “On the properties of neural machine translation: Encoder-decoder approaches.” arXiv preprint arXiv:1409.1259.

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014b). “Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation.” In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, pp. 1724–1734.

Dzmitry, B., KyungHyun, C., and Yoshua, B. (2015). “Neural Machine Translation by Jointly Learning to Align and Translate.” In Proceedings of the 3rd International Conference on Learning Representations.

Eriguchi, A., Hashimoto, K., and Tsuruoka, Y. (2016a). “Character-based Decoding in Tree-to-Sequence Attention-based Neural Machine Translation.” In Proceedings of the 3rd workshop on Asian Translation, pp. 175–183.

Eriguchi, A., Hashimoto, K., and Tsuruoka, Y. (2016b). “Tree-to-Sequence Attentional Neural Machine Translation.” In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pp. 823–833.

Gers, F. A., Schmidhuber, J., and Cummins, F. (2000). “Learning to forget: Continual prediction with LSTM.” Neural computation, 12 (10), pp. 2451–2471.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). “Deep residual learning for image recognition.” In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778.

Hochreiter, S. and Schmidhuber, J. (1997). “Long short-term memory.” Neural computation, 9 (8), pp. 1735–1780.

Kalchbrenner, N. and Blunsom, P. (2013). “Recurrent Continuous Translation Models.” In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, p. 413.

Kalchbrenner, N., Espeholt, L., Simonyan, K., Oord, A. v. d., Graves, A., and Kavukcuoglu, K. (2016). “Neural machine translation in linear time.” arXiv preprint arXiv:1610.10099.

Kasami, T. (1965). “An Efficient recognition and syntax algorithm for context-free languages.” Tech. rep. AFCRL-65-758.

Kingsma, D. and Ba, J. (2014). “Adam: A method for stochastic optimization.” In 5th International Conference on Learning Representations.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al. (2007). “Moses: Open source toolkit for statistical machine translation.” In Proceedings of the 45th annual meeting of the Association for Computational Linguistics on interactive poster and demonstration sessions, pp. 177–180.

Lamb, A. and Xie, M. (2016). “Convolutional encoders for neural machine translation.” arXiv preprint arXiv:1611.02344.

Luong, T., Pham, H., and Manning, C. D. (2015). “Effective Approaches to Attention-based Neural Machine Translation.” In Proceedings of the 2015 Conference on Empirical Methods in

Natural Language Processing, pp. 1412–1421.

Nakazawa, T., Yaguchi, M., Uchimoto, K., Utiyama, M., Sumita, E., Kurohashi, S., and Isahara, H. (2016). “ASPEC: Asian Scientific Paper Excerpt Corpus.” In Proceedings of the Ninth International Conference on Language Resources and Evaluation, pp. 2204–2208.

Neubig, G., Nakata, Y., and Mori, S. (2011). “Pointwise prediction for robust, adaptable Japanese morphological analysis.” In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, pp. 529–533.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). “BLEU: a Method for Automatic Evaluation of Machine Translation.” In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, pp. 311–318.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). “Dropout: a simple way to prevent neural networks from overfitting.” Journal of Machine Learning Research, 15 (1), pp. 1929–1958.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). “Sequence to sequence learning with neural networks.” In Advances in neural information processing systems, pp. 3104–3112.

Younger, D. H. (1967). “Recognition and parsing of context-free languages in time n 3 .” Information and Control, 2 (10), pp. 189–208.

# 2.2 Neural Machine Translation Model with a Large Vocabulary Selected by Branching Entropy

Zi Long, Ryuichiro Kimura
(University of Tsukuba)
Shohei Iida
(Tokyo Denki University)
Takehito Utsuro
(University of Tsukuba)
Mikio Yamamoto
(University of Tsukuba)

## 2.2.1 Introduction

Neural machine translation (NMT), a new approach to solving machine translation, has achieved promising results [1][3][6][8][14][15][21]. An NMT system builds a simple large neural network that reads the entire input source sentence and generates an output translation. The entire neural network is jointly trained to maximize the conditional probability of the correct translation of a source sentence with a bilingual corpus. Although NMT offers many advantages over traditional phrase-based approaches, such as a small memory footprint and simple decoder implementation, conventional NMT is limited when it comes to larger vocabularies. This is because the training complexity and decoding complexity proportionally increase with the number of target words. Words that are out of vocabulary are represented by a single "<*unk*>" token in translations, as illustrated in Figure 1. The problem becomes more serious when translating patent documents, which contain several newly introduced technical terms.

There have been a number of related studies that address the vocabulary limitation of NMT systems. Jean el al. [6] provided an efficient approximation to the softmax to accommodate a very large vocabulary in an NMT system. Luong et al. [16] proposed annotating the occurrences of a target unknown word token with positional information to track its alignments, after which they replace the tokens with their translations using simple word dictionary lookup or identity copy. Li et al. [11] proposed to replace out-of-vocabulary words with similar in-vocabulary words based on a similarity model learnt from monolingual data. Sennrich et al.[20] introduced an effective approach based on encoding rare and unknown words as sequences of subword units. Luong and Manning [14] provided a character-level and word-level hybrid NMT model to achieve an open vocabulary, and Costa-jussà and Fonollosa[4] proposed a NMT system based on character-based embeddings.

However, these previous approaches have limitations when translating patent sentences. This is be- cause their methods only focus on addressing the problem of unknown words even though the words are parts of technical terms. It is obvious that a technical term should be considered as one word that comprises components that always have different meanings and translations when they are used alone. An example is shown in Figure 1, wherein Japanese word "ブリッジ"(bridge) should be translated to Chinese word "桥架" when included in technical term "bridge interface"; however, it is always translated as "桥".

To address this problem, Long et al.[13] proposed extracting compound nouns as technical terms and replacing them with tokens. Long et al.[12] proposed to select phrase pairs using the statistical approach of branching entropy; this allows the proposed technique to be applied to the translation task on any language pair without needing specific language knowledge to formulate the rules for technical term identification. In this paper, we apply the method proposed by Long et al.[12] to the WAT 2017 Japanese-Chinese and Japanese-English patent datasets. On the WAT 2017 Japanese-Chinese JPO patent dataset, the NMT model of Long et al. [12] achieves an improvement of 1.4 BLEU points over a baseline NMT model when translating Japanese sentences into Chinese, and an improvement of 0.8 BLEU points when translating Chinese sentences into Japanese. On the WAT 2017 Japanese-English

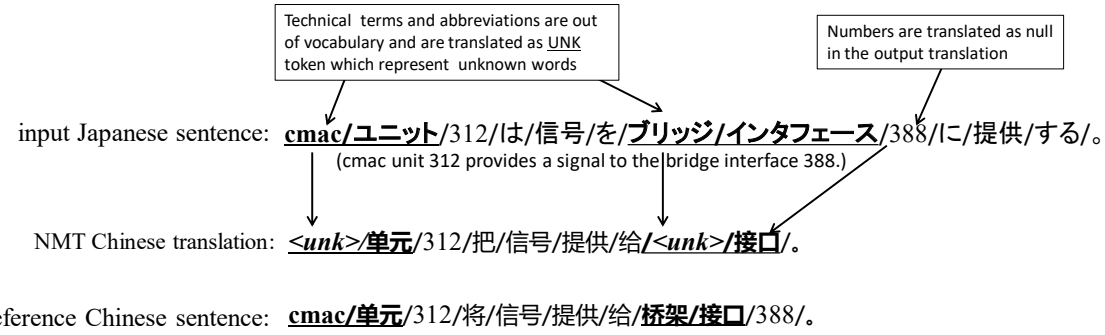



**Figure 1 Example of translation errors when translating patent sentences with technical terms using NMT**

JPO patent dataset, the NMT model of Long et al. [12] achieves an improvement of 0.8 BLEU points over a baseline NMT model when translating Japanese sentences into English, and an improvement of 0.7 BLEU points when translating English sentences into Japanese. Moreover, the number of translation error of under-translations[1] by PosUnk model proposed by Luong et al. [16] reduces to around 30% by the NMT model of Long et al. [12].

## 2.2.2 Neural Machine Translation (NMT)

NMT uses a single neural network trained jointly to maximize the translation performance [1][3][6][8] [14][15][21]. Given a source sentence $\boldsymbol{x} = (x_1, \dots, x_N)$ and target sentence $\boldsymbol{y} = (y_1, \dots, y_M)$, an NMT system uses a neural network to parameterize the conditional distributions

$$p(y_l | y_{<l}, \boldsymbol{x})$$

for $1 \leq l \leq M$. Consequently, it becomes possible to compute and maximize the log probability of the target sentence given the source sentence

$$\log p(\boldsymbol{y}|\boldsymbol{x}) = \sum_{l=1}^{M} \log p\,(y_l | y_{<l}, \boldsymbol{x})$$

In this paper, we use an NMT model similar to that used by Bahdanau el al.[1], which consists of an encoder of a bidirectional long short-term memory (LSTM)[5] and another LSTM as decoder. In the model of Bahdanau el al.[1], the encoder consists of forward and backward LSTMs. The forward LSTM reads the source sentence as it is ordered (from $x_1$ to $x_N$) and calculates a sequence of forward hidden states, while the backward LSTM reads the source sentence in the reverse order (from $x_N$ to $x_1$), resulting in a sequence of backward hidden states. The decoder then predicts target words using not only a recurrent hidden state and the previously predicted word but also a context vector as followings:

$$p(\boldsymbol{y}|\boldsymbol{x}) = g(y_{z-1}, s_{z-1}, c_Z)$$

where $s_{z-1}$ is an LSTM hidden state of decoder, and $c_Z$ is a context vector computed from both of the forward hidden states and backward hidden states, for $1 \leq z \leq M$.

## 2.2.3 Phrase Pair Selection using Branching Entropy

Branching entropy has been applied to the procedure of text segmentation (e.g., Jin and Tanaka-Ishii[7]) and key phrases extraction (e.g., Chen et al.[2]). In this work, we use the left/right branching entropy to detect the boundaries of phrases, and thus select phrase pairs automatically.

[1]It is known that NMT models tend to have the problem of the under-translation. Tu el al.[22] proposed coverage-based NMT which considers the problem of the under-translation.

#### 2.2.3.1 Branching Entropy

The left branching entropy and right branching entropy of a phrase $\boldsymbol{w}$ are respectively defined as

$$H_l(\boldsymbol{w}) = -\sum_{v \in \mathrm{V}_l(\boldsymbol{w})} p_l(v|\boldsymbol{w}) log_2 p_l(v|\boldsymbol{w})$$

$$H_r(\boldsymbol{w}) = -\sum_{v \in \mathrm{V}_r(\boldsymbol{w})} p_r(v|\boldsymbol{w}) log_2 p_r(v|\boldsymbol{w})$$

where $\boldsymbol{w}$ is the phrase of interest (e.g., "ブリッジインターフェイス" in the Japanese sentence shown in Figure 1, which means "bridge interface"), $\mathrm{V}_l(\boldsymbol{w})$ is a set of words that are adjacent to the left of $\boldsymbol{w}$ (e.g., "を" in, which is a Japanese particle) and $\mathrm{V}_r(\boldsymbol{w})$ is a set of words that are adjacent to the right of $\boldsymbol{w}$ (e.g., "388" in Figure 1). The probabilities $p_l(v|\boldsymbol{w})$ and $p_r(v|\boldsymbol{w})$ are respectively computed as

$$p_l(v|\boldsymbol{w}) = \frac{f(v, \boldsymbol{w})}{f(\boldsymbol{w})} \qquad p_r(v|\boldsymbol{w}) = \frac{f(\boldsymbol{w}, v)}{f(\boldsymbol{w})}$$

where $f(\boldsymbol{w})$ is the frequency count of phrase $\boldsymbol{w}$, and $f(v, w)$ and $f(\boldsymbol{w}, v)$ are the frequency counts of sequence "$v, \boldsymbol{w}$" and sequence "$\boldsymbol{w}, v$" respectively. According to the definition of branching entropy, when a phrase $\boldsymbol{w}$ is a technical term that is always used as a compound word, both its left branching entropy $H_l(\boldsymbol{w})$ and right branching entropy $H_r(\boldsymbol{w})$ have high values because many different words, such as particles and numbers, can be adjacent to the phrase. However, the left/right branching entropy of substrings of $\boldsymbol{w}$ have low values because words contained in $\boldsymbol{w}$ are always adjacent to each other.

#### 2.2.3.2 Selecting Phrase Pairs

Given a parallel sentence pair $\langle S_s, S_t \rangle$, all $n$-grams phrases of source sentence $S_s$ and target sentence $S_t$ are extracted and aligned using phrase translation table and word alignment of SMT according to the approaches described in Long et al. Next, phrase translation pair $\langle t_s, t_t \rangle$ obtained from $\langle S_s, S_t \rangle$ that satisfies all the following conditions is selected as a phrase pair and is extracted:

(1) Either $t_s$ or $t_t$ contains at least one out-of-vocabulary word.

(2) Neither $t_s$ nor $t_t$ contains predetermined stop words.

(3) Entropies $H_l(t_s)$, $H_l(t_t)$, $H_r(t_s)$ and $H_r(t_t)$ are larger than a lower bound, while the left/right branching entropy of the substrings of $t_s$ and $t_t$ are lower than or equal to the lower bound.

Here, the maximum length of a phrase as well as the lower bound of the branching entropy are tuned with the validation set.[2] All the selected source-target phrase pairs are then used in the next section as phrase pairs.

### 2.2.4 NMT with a Large Technical Term Vocabulary

In this work, the NMT model is trained on a bilingual corpus in which phrase pairs are replaced with tokens. The NMT system is then used as a decoder to translate the source sentences and replace the tokens with phrases translated using SMT.

[2] Throughout the evaluations on patent translation of both language pairs of Japanese-Chinese and Japanese-English, the maximum length of the extracted phrases is tuned as 7. The lower bounds of the branching entropy are tuned as 5 for patent translation of the language pair of Japanese-Chinese, and 8 for patent translation of the language pair of Japanese-English. We also tune the number of stop words using the validation set, and use the 200 most-frequent Japanese morphemes and Chinese words as stop words for the language pair of Japanese-Chinese, use the 100 most-frequent Japanese morphemes and English words as stop words for the language pair of Japanese-English.

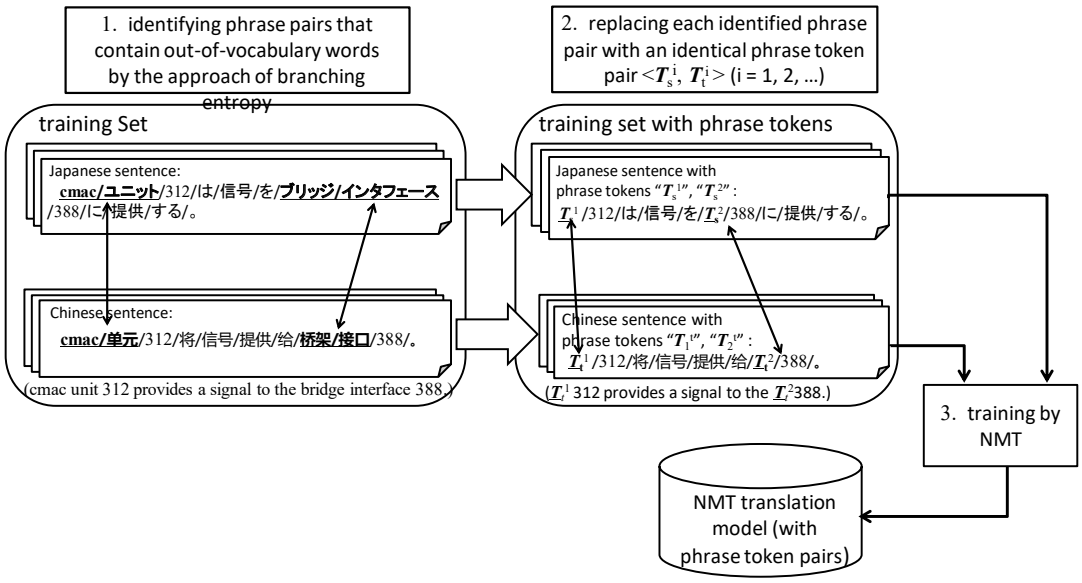



**Figure 2 NMT training after replacing phrase pairs with token pairs $\langle T_s^i, T_t^i \rangle$ $(i = 1, 2, \cdots, k)$**

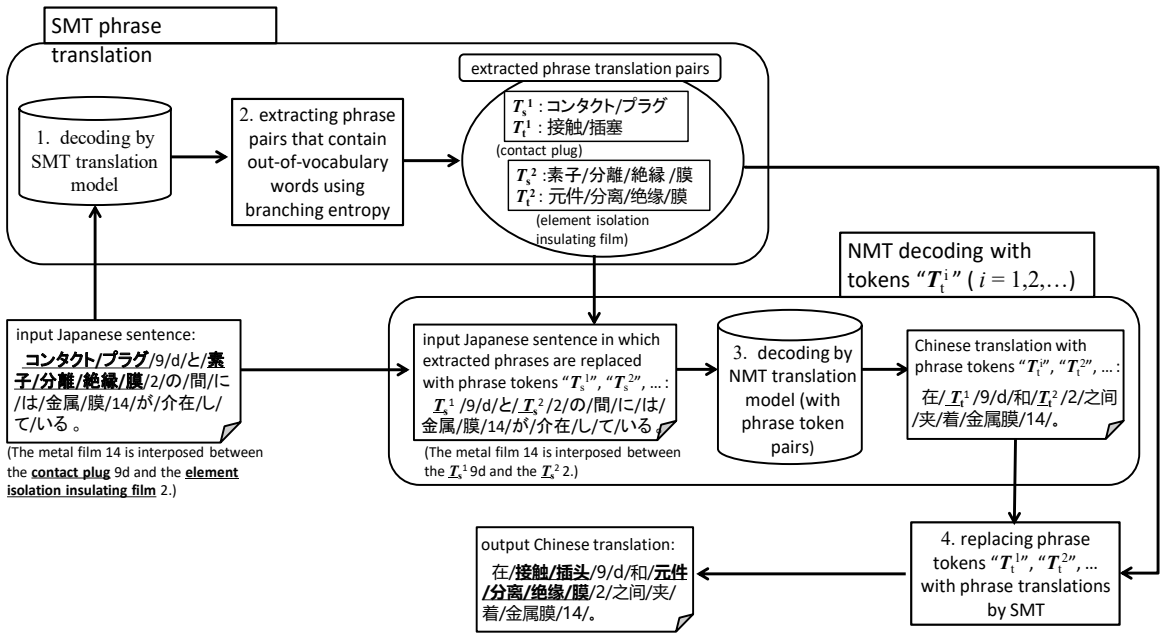



**Figure 3 NMT training after replacing phrase pairs with token pairs $\langle T_s^i, T_t^i \rangle$ $(i = 1, 2, \cdots, k)$**

### 2.2.4.1 NMT Training after Replacing Technical Term Pairs with Tokens

Figure 2 illustrates the procedure for training the model with parallel patent sentence pairs in which phrase pairs are replaced with phrase token pairs $\langle T_s^1, T_t^1 \rangle$, $\langle T_s^2, T_t^2 \rangle$, and so on.

In the step 1 of Figure 2, source-target phrase pairs that contain at least one out-of-vocabulary word are selected from the training set using the branching entropy approach described in Section 2.2.3. As shown in the step 2 of Figure 2, in each of the parallel patent sentence pairs, occurrences of phrase pairs $(t_s^1, t_t^1)$, $(t_s^2, t_t^2), \cdots, (t_s^k, t_t^k)$ are then replaced with token pairs $\langle T_s^1, T_t^1 \rangle$, $\langle T_s^2, T_t^2 \rangle$, $\cdots$, $\langle T_s^k, T_t^k \rangle$. Phrase pairs $(t_s^1, t_t^1)$, $(t_s^2, t_t^2), \cdots, (t_s^k, t_t^k)$ are numbered in the order of occurrence of the source phrases $t^i$ $(i = 1,2, \cdots, k)$ in each source sentence $S_s$. Here note that in all the parallel sentence pairs $\langle S_s, S_t \rangle$, the tokens pairs $\langle T_s^1, T_t^1 \rangle$, $\langle T_s^2, T_t^2 \rangle$, $\cdots$ that are identical throughout all the parallel sentence pairs are used in this procedure. Therefore, for example, in all the source patent sentences $S_s$, the phrase $t_s^1$ which appears

earlier than other phrases in $S_s$ is replaced with $T_s^1$. We then train the NMT model on a bilingual corpus, in which the phrase pairs are replaced by token pairs $\langle T_s^i, T_t^i \rangle$ $(i = 1,2,\cdots,k)$, and obtain an NMT model in which the phrases are represented as tokens.

#### 2.2.4.2 NMT Decoding and SMT Technical Term Translation

Figure 3 illustrates the procedure for producing target translations by decoding the input source sentence using the NMT model of Long et al.[12].

In the step 1 of Figure 3, when given an input source sentence, we first generate its translation by decoding of SMT translation model. Next, as shown in the step 2 of Figure 3, we automatically extract the phrase pairs by branching entropy according to the procedure of Section 2.2.3, where the input sentence and its SMT translation are considered as a pair of parallel sentence. Phrase pairs that contains at least one out-of-vocabulary word are extracted and are replaced with phrase token pairs $\langle T_s^i, T_t^i \rangle$ $(i = 1,2,\cdots,k)$. Consequently, we have an input sentence in which the tokens "$T_s^i$"$(i = 1,2,\cdots,k)$ represent the positions of the phrases and a list of SMT phrase translations of extracted Japanese phrases. Next, as shown in the step 3 of Figure 3, the source Japanese sentence with tokens is translated using the NMT model trained according to the procedure described in Section 2.2.4.1. Finally, in the step 4, we replace the tokens "$T_t^i$"$(i = 1,2,\cdots,k)$ of the target sentence translation with the phrase translations of the SMT.

## 2.2.5 Evaluation

#### 2.2.5.1 DataSets

We evaluated the effectiveness of the NMT model of Long et al.[12] on the WAT 2017 Japanese-Chinese and Japanese-English JPO dataset.[3] Out of the training set of the WAT 2017 Japanese-Chinese JPO dataset, we used 998,954 patent sentence pairs, whose Japanese sentences contain fewer than 100 morphemes, Chinese sentences contain fewer than 100 words. Out of the training set of the WAT 2017 Japanese-English JPO dataset, we used 999,636 sentence pairs whose Japanese sentences contain fewer than 100 morphemes and English sentences contain fewer than 100 words. In both cases, we used all of the sentence pairs contained in the development sets of the WAT 2017 JPO datasets as development sets, and we used all of the sentence pairs contained in the test sets of the WAT 2017 JPO datasets as test sets. Table 1shows the statistics of the dataset.

**Table 1 Statistics of datasets**

| | training set | validation set | test set |
|---|---|---|---|
| Japanese-Chinese | 998,054 | 2,000 | 2,000 |
| Japanese-English | 999,636 | 2,000 | 2,000 |

According to the procedure of Section 2.2.3, from the Japanese-Chinese sentence pairs of the training set, we collected 102,630 occurrences of Japanese-Chinese phrase pairs, which are 69,387 types of phrase pairs with 52,786 unique types of Japanese phrases and 67,456 unique types of Chinese phrases. Within the total 2,000 Japanese patent sentences in the Japanese-Chinese test set, 266 occurrences of Japanese phrases were extracted, which correspond to 247 types. With the total 2,000 Chinese patent sentences in the Japanese-Chinese test set, 417 occurrences of Chinese phrases were extracted, which correspond to 382 types.

From the Japanese-English sentence pairs of the training set, we collected 38,457 occurrences of Japanese-English phrase pairs, which are 35,544 types of phrase pairs with unique 34,569 types of Japanese phrases and 35,087 unique types of English phrases. Within the total 2,000 Japanese patent

[3] http://lotus.kuee.kyoto-u.ac.jp/WAT/patent/index.html

sentences in the Japanese-English test set, 249 occurrences of Japanese phrases were extracted, which correspond to 221 types. With the total 2,000 English patent sentences in the Japanese-English test set, 246 occurrences of English phrases were extracted, which correspond to 230 types.

#### 2.2.5.2 Training Details

For the training of the SMT model, including the word alignment and the phrase translation table, we used Moses[10], a toolkit for phrase-based SMT models. We trained the SMT model on the training set and tuned it with the validation set.

For the training of the NMT model, our training procedure and hyperparameter choices were similar to those of Bahdanau et al.[1]. The encoder consists of forward and backward deep LSTM neural networks each consisting of three layers, with 512 cells in each layer. The decoder is a three-layer deep LSTM with 512 cells in each layer. Both the source vocabulary and the target vocabulary are limited to the 40K most-frequently used morphemes / words in the training set. The size of the word embedding was set to 512. We ensured that all sentences in a minibatch were roughly the same length. Further training details are given below:

(1) We set the size of a minibatch to 128.

(2) All of the LSTM's parameter were initialized with a uniform distribution ranging between -0.06 and 0.06.

(3) We used the stochastic gradient descent, beginning at a fixed learning rate of 1. We trained our model for a total of 10 epochs, and we began to halve the learning rate every epoch after the first seven epochs.

(4) Similar to Sutskever et al.[21], we rescaled the normalized gradient to ensure that its norm does not exceed 5.

We trained the NMT model on the training set. The training time was around two days when using the described parameters on a 1-GPU machine.

We compute the branching entropy using the frequency statistics from the training set.

#### 2.2.5.3 Evaluation Results

In this work, we calculated automatic evaluation scores for the translation results using a popular metrics called BLEU[19]. As shown in Table 2, we report the evaluation scores, using the translations by Moses[10] as the baseline SMT and the scores using the translations produced by the baseline NMT system without the approach proposed by Long et al.[12] as the baseline NMT. As shown in Table 2, the BLEU score obtained by the NMT model of Long et al. [12] is clearly higher than those of the baselines. Here, as described in Section 2.2.3, the lower bounds of branching entropy for phrase pair selection are tuned as 5 throughout the evaluation of language pair of Japanese-Chinese, and tuned as 8 throughout the evaluation of language pair of Japanese-English, respectively. On the WAT 2017 Japanese-Chinese JPO patent dataset, when compared with the baseline SMT, the performance gains of the NMT model of Long et al. [12] are approximately 5.6 BLEU points when translating Japanese into Chinese and 5.4 BLEU when translating Chinese into Japanese. On the WAT 2017 Japanese-English JPO patent dataset, when compared with the baseline SMT, the performance gains of the NMT model of Long et al. [12] are approximately 15.9 BLEU points when translating Japanese into English and 13.1 BLEU when translating English into Japanese. When compared with the result of the baseline NMT, the NMT model of Long et al. [12] achieved performance gains of 1.4 BLEU points on the task of translating Japanese into Chinese and 0.8 BLEU points on the task of translating Chinese into Japanese. When compared with the result of the baseline NMT, the NMT model of Long et al. [12] achieved performance gains of 0.8 BLEU points on the task of translating Japanese into English and 1.4 BLEU points on the task of translating English into Japanese.

**Table 2 Automatic evaluation results (BLEU)**

| System | ja → ch | ch → ja | ja → en | en → ja |
|---|---|---|---|---|
| Baseline SMT [10] | 30.0 | 36.2 | 28.0 | 29.4 |
| Baseline NMT | 34.2 | 40.8 | 43.1 | 41.8 |
| NMT with PosUnk model [16] | 34.5 | 41.0 | 43.5 | 42.0 |
| NMT with technical term translation by SMT[12] | **35.6** | **41.6** | **43.9** | **42.5** |

**Table 3 Human evaluation results of pairwise evaluation**

| System | ja → ch | ch → ja | ja → en | en → ja |
|---|---|---|---|---|
| NMT with PosUnk model [16] | 13 | 12.5 | 9.5 | 14.5 |
| NMT with technical term translation by SMT[12] | **23.5** | **22.5** | **15.5** | **19** |

**Table 4 Human evaluation results of JPO adequacy evaluation**

| System | ja → ch | ch → ja | ja → en | en → ja |
|---|---|---|---|---|
| Baseline SMT [10] | 3.1 | 3.2 | 2.9 | 3.0 |
| Baseline NMT | 3.6 | 3.6 | 3.7 | 3.7 |
| NMT with PosUnk model [16] | 3.8 | 3.9 | 3.9 | 3.9 |
| NMT with technical term translation by SMT[12] | **4.1** | **4.1** | **4.2** | **4.1** |

Furthermore, we quantitatively compared our study with the work of Luong et al. [16]. Table 2 compares the NMT model with the PosUnk model, which is the best model proposed by Luong et al. [16]. The NMT model of Long et al. [12] achieves performance gains of 0.9 BLEU points when translating Japanese into Chinese, and performance gains of 0.6 BLEU points when translating Chinese into Japanese. The NMT model of Long et al. [12] achieves performance gains of 0.4 BLEU points when translating Japanese into English, and performance gains of 0.5 BLEU points when translating English into Japanese.

In this study, we also conducted two types of human evaluations according to the work of Nakazawa et al.[18]: pairwise evaluation and JPO adequacy evaluation. In the pairwise evaluation, we compared each translation produced by the baseline NMT with that produced by the NMT model of Long et al. [12] as well as the NMT model with PosUnk model, and judged which translation is better or whether they have comparable quality. In contrast to the study conducted by Nakazawa et al. [18], we randomly selected 200 sentence pairs from the test set for human evaluation, and both human evaluations were conducted using only one judgement.Table 3 and Table 4 show the results of the human evaluation for the baseline SMT, baseline NMT, NMT model with PosUnk model, and the NMT model of Long et al. [12]. We observe that the NMT model of Long et al. [12] achieves the best performance for both the pairwise and JPO adequacy evaluations when we replace the tokens with SMT phrase translations after decoding the source sentence with the tokens.

Moreover, Table 5 shows the results of automatic evaluation, pairwise evaluation and JPO adequacy evaluation from the WAT 2017[17].[4] We observe that the NMT model of Long et al. [12] achieves a substantial improvement over the WAT 2017 baseline. For the test sets, we also counted the numbers of the untranslated words of input sentences. As shown in Table 6, the number of untranslated words by the baseline NMT reduced to around 65% by the NMT model of Long et al. [12]. This is mainly

[4] http://lotus.kuee.kyoto-u.ac.jp/WAT/evaluation/

**Table 5 Evaluation results from WAT 2017**

| Evaluation | System | ja → ch | ch → ja | ja → en | en → ja |
|---|---|---|---|---|---|
| Automatic evaluation BLEU | Baseline SMT [10] | 32.1 | 38.5 | 30.8 | 34.3 |
| | Baseline NMT | **33.2** | **40.5** | **37.3** | **41.1** |
| Pairwise Evaluation | NMT with PosUnk model [16] | 21.8 | 40.1 | 51.5 | 49.5 |
| JPO adequacy evaluation | NMT with technical term translation by SMT[12] | 4.1 | 3.9 | 4.2 | 4.3 |

**Table 6 Numbers of untranslated morphemes / words of input sentences**

| System | ja → ch | ch → ja | ja → en | en → ja |
|---|---|---|---|---|
| NMT with PosUnk model [16] | 1,112 | 846 | 1,031 | 794 |
| NMT with technical term translation by SMT[12] | **736** | **581** | **655** | **571** |

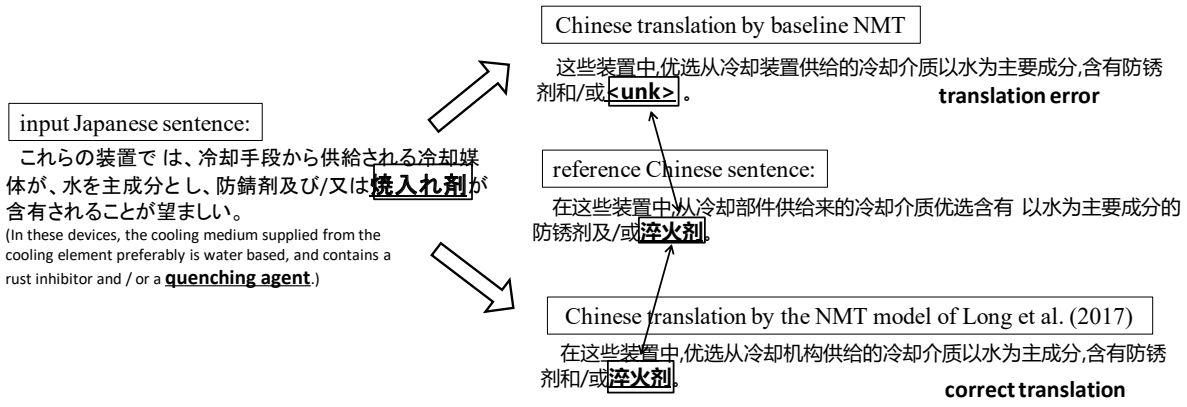



**Figure 4 An example of correct translations produced by the MT model of Long et al.[12] when addressing the problem of out-of-vocabulary words (Japanese-to-Chinese)**

because part of untranslated source words are out-of-vocabulary, and thus are untranslated by the baseline NMT. The NMT model of Long et al. [12] extracts those out-of-vocabulary words as a part of phrases and replaces those phrases with tokens before the decoding of NMT. Those phrases are then translated by SMT and inserted in the output translation, which ensures that those out-of-vocabulary words are translated.

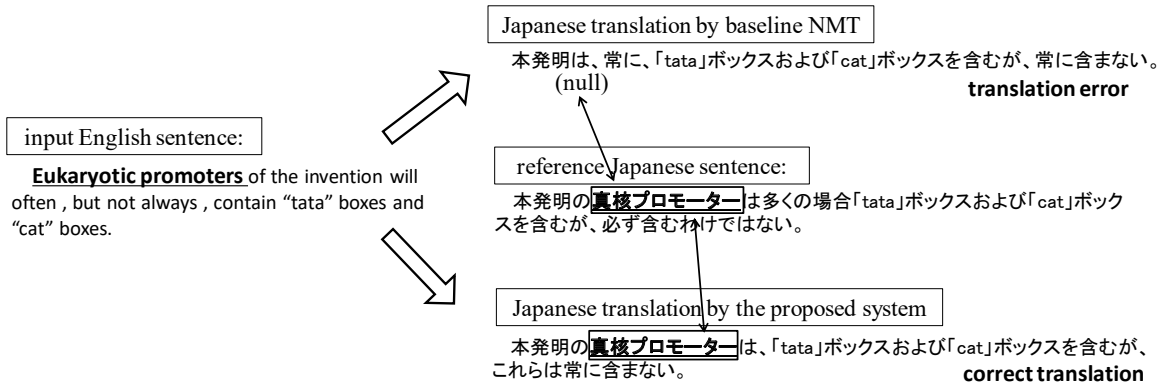



**Figure 5 An example of correct translations produced by the NMT model of Long et al.[12] when addressing the problem of under-translation (English-to-Japanese)**

Figure 4 compares an example of correct translation produced by the NMT model of Long et al. [12] with one produced by the baseline NMT. In this example, the translation is a translation error because the Japanese word "焼入れ (quenching)" is an out-of-vocabulary word and is erroneously translated into the "<*unk*>" token. The NMT model of Long et al. [12] correctly translated the Japanese sentenceinto Chinese, where the out-of-vocabulary word "焼入れ" is correctly selected by the approach of branching entropy as a part of the Japanese phrase "焼入れ剤 (quenching agent)". The selected Japanese phrase is then translated by the phrase translation table of SMT. Figure 5 shows another example of correct translation produced by the NMT model of Long et al.[12] with one produced by the baseline NMT. As shown in Figure 5, the translation produced by baseline NMT is a translation error because the out-of-vocabulary English words "eukaryotic" and "promoters" are untranslated words and their translations are not contained in the output translation of the baseline NMT. The NMT model of Long et al. [12] correctly translated those English words into Japanese because those English words "eukaryotic" and "promoters" are selected as an English phrase "eukaryotic promoters" with branching entropy and then are translated by SMT.

## 2.2.6 Conclusion

Long et al. [12] proposed selecting phrases that contain out-of-vocabulary words using the branching entropy. These selected phrases are then replaced with tokens and post-translated using an SMT phrase translation. In this paper, we apply the method proposed by Long et al. [12] to the WAT 2017 Japanese-Chinese and Japanese-English patent datasets. We observed that the NMT model of Long et al. [12] performed much better than the baseline NMT system in all of the language pairs: Japanese-to-Chinese/Chinese-to-Japanese and Japanese-to-English/English-to-Japanese. One of our important future tasks is to compare the translation performance of the NMT model of Long et al. [12] with that based on subword units (e.g. Sennrich et al.[20]). Another future work is to integrate the reranking framework for minimizing untranslated content [5] into the NMT model of Long et al. [12], which is expected to further reduce the number of untranslated words. This future work is roughly based on the observation reported in Kimura et al.[9], where the NMT model of Long et al.[12] is not only effective in reducing the untranslated content without any specific framework of minimizing the untranslated content, but also successfully reduced the estimated volumes of the untranslated content, which was proposed by Goto and Tanaka [5].

# References

[1] D. Bahdanau, K. Cho, and Y. Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proc. 3rd ICLR*.

[2] Y. Chen, Y. Huang, S. Kong, and L. Lee. 2010. Automatic key term extraction from spoken course lectures using branching entropy and prosodic/semantic features. In *Proc. 2010 IEEE SLT Workshop*, pages 265–

270.

[3] K. Cho, B. van Merriënboer, C. Gulcehre, F. Bougares, H. Schwenk, and Y. Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *Proc. EMNLP*.

[4] M. R. Costa-jussà and J. A. R. Fonollosa. 2016. Character-based neural machine translation. In *Proc. 54th ACL*, pages 357–361.

[5] I. Goto and H. Tanaka. 2017. Detecting untranslated content for neural machine translation. In *Proc. 1st NMT*, pages 47–55. S. Hochreiter and J. Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

[6] S. Jean, K. Cho, Y. Bengio, and R. Memisevic. 2014. On using very large target vocabulary for neural machine translation. In *Proc. 28th NIPS*, pages 1–10.

[7] Z. Jin and K. Tanaka-Ishii. 2006. Unsupervised segmentation of Chinese text by use of branching entropy. In *Proc. COLING/ACL 2006*, pages 428–435.

[8] N. Kalchbrenner and P. Blunsom. 2013. Recurrent continous translation models. In *Proc. EMNLP*, pages 1700–1709.

[9] R. Kimura, Z. Long, T. Utsuro, T. Mitsuhashi, and M. Yamamoto. 2017. Effect on reducing untranslated content by neural machine translation with a large vocabulary of technical terms. In *Proc. 7th PSLT*, pages 9–20.

[10] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proc. 45th ACL, Companion Volume*, pages 177–180.

[11] X. Li, J. Zhang, and C. Zong. 2016. Towards zero unknown word in neural machine translation. In *Proc. 25th IJCAI*, pages 2852–2858.

[12] Z. Long, R. Kimura, T. Utsuro, T. Mitsuhashi, and M. Yamamoto. 2017. Neural machine translation model with a large vocabulary selected by branching entropy. In *Proc. MT Summit XVI*, pages 227–240.

[13] Z. Long, T. Utsuro, T. Mitsuhashi, and M. Yamamoto. 2016. Translation of patent sentences with a large vocabulary of technical terms using neural machine translation. In *Proc. 3rd WAT*, pages 47–57.

[14] M. Luong and C. D. Manning. 2016. Achieving open vocabulary neural machine translation with hybrid word- character models. In *Proc. 54th ACL*, pages 1054–1063.

[15] M. Luong, H. Pham, and C. D. Manning. 2015a. Effective approaches to attention-based neural machine translation. In *Proc. EMNLP*.

[16] M. Luong, I. Sutskever, O. Vinyals, Q. V. Le, and W. Zaremba. 2015b. Addressing the rare word problem in neural machine translation. In *Proc. 53rd ACL*, pages 11–19.

[17] T. Nakazawa, S. Higashiyama, C. Ding, H. Mino, I. Goto, G. Neubig, H. Kazawa, Y. Oda, J. Harashima, and S. Kurohashi. 2017. Overview of the 4th workshop on Asian translation. In *Proc. 4th WAT*.

[18] T. Nakazawa, H. Mino, I. Goto, G. Neubig, S. Kurohashi, and E. Sumita. 2015. Overview of the 2nd workshop on asian translation. In *Proc. 2nd WAT*, pages 1–28.

[19] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proc. 40th ACL*, pages 311–318.

[20] R. Sennrich, B. Haddow, and A. Birch. 2016. Neural machine translation of rare words with subword units. In *Proc. 54th ACL*, pages 1715–1725.

[21] I. Sutskever, O. Vinyals, and Q. V. Le. 2014. Sequence to sequence learning with neural machine translation. In *Proc. 28th NIPS*.

# 2.3 局所類似性指標に基づく類似文を利用したニューラル機械翻訳

京都大学　中尾　亮太
中澤　敏明
黒橋　禎夫

## 2.3.1 はじめに

ニューラル機械翻訳(NMT)は近年登場した手法で、従来の手法を大きく上回る成果を出している。従来手法では言語モデルや単語アライメントの学習や翻訳の実行、翻訳のチューニングを別々に行う必要があったが、NMT では単一のモデルによって学習から翻訳まで可能で、さらに入力文全体を連続空間上で表現可能であることが特徴である。

Bahdanau ら(2014)が提案した注目型ニューラルネットワークは、入力文の長さに比例した情報を保持し、そのうち必要な部分に注目し値を取り出すことを可能にする。これによって長い入力文でも翻訳精度が落ちにくくなり、またニューラル機械翻訳システムが翻訳中に入力文のどこに着目しているかの可視化も容易になった。

Gu らはこの注目型ニューラルネットワークをベースに「翻訳メモリ」に類似した機構を取り入れることでさらなる翻訳精度の改善を果たした(2017)。翻訳メモリとは、単語やフレーズ、文などの単位で原文と翻訳文が対となり格納されたデータベース、及びそれを内包し翻訳者の仕事を支援するツールやソフトウェアのことである。翻訳したい文やフレーズと一致または類似した文やフレーズをデータベースから検索して引用することで、翻訳作業の高速化や精度の向上、表現の統一化を図る。

Gu らの手法では、トレーニングコーパスをモデルのトレーニング時に使うのみではなく、翻訳時にトレーニングコーパスから対訳対を取り出して翻訳メモリのように参考にして翻訳する。しかし、Gu らの手法には問題がある。Gu らは翻訳したい文に類似した文をトレーニングコーパスから取り出すために、文と文の編集距離を類似度の指標として使用しているが、これでは類似文を翻訳に利用するというアイデアを活かしきれない。なぜなら Gu らが基とした注目型ニューラルネットワークは、必要な部分に注目して翻訳を行うことが出来ることが特徴であるゆえに、フレーズや文節レベルの局所的な意味のまとまりの類似性こそが重要であるにも関わらず、Gu らの提案した編集距離を用いた類似度では、文全体の類似性は計算できるものの、フレーズ単位といった局所的な部分での類似性は考慮されていないからである。

そこで本稿では局所的な指標を利用して、フレーズ単位で類似した文を取得できる、注目型ニューラルネットワークにより適した類似度の計算法を用いて類似文を選択する手法を提案する。実験の結果、この手法を用いることで BLEU スコアを改善することができた。

## 2.3.2　類似文を利用したニューラル機械翻訳

本研究は Gu らの提案した Search Engine Guided non-Parametric Neural Machine Translation (以下 SEGNMT)という手法を改善するものである。SEGNMT の概要を図 1 に示す。

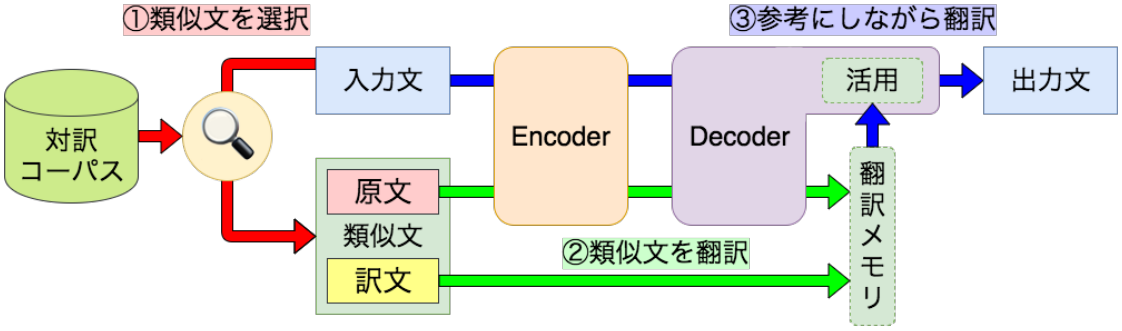



図 1: SEGNMT の概要

図中の番号と対応付けて順に説明すると以下のようになる。

1. Retrieval Stage：類似文の検索
   (ア) 翻訳したい文に類似している文とその訳文を汎用の検索エンジンを用いていくつかトレーニングコーパスから取り出す(①)
   (イ) 翻訳したい文との類似度によって取り出した対訳対を順位付けする(①)
   (ウ) 不要な対訳対を取り除く(①)
2. Translation Stage：翻訳の実行
   (ア) 選択した対訳対の元言語の文を翻訳し、翻訳中のニューラルネットワーク内部状態を保存する(②)
   (イ) 入力文の翻訳時に、保存しておいた内部状態を適切に利用することで精度の向上を図る(③)

**Retrieval Stage**: Retrieval Stage ではトレーニングコーパスから翻訳したい文 $X$ に類似した文 $X'$とその翻訳文 $Y'$の組$(X', Y')$を複数選択する。しかし、一般にニューラルネットワークのトレーニング用データは数十万から数百万もの対訳対からなり、それら全ての文に対して類似度を計算するのは現実的ではない。そこでまず汎用の検索エンジンを利用して類似度の計算対象を絞り込む。

検索エンジンから類似文を取得する際、ニューラルネットワークのトレーニング中は必ず入力文と等しい類似文及びその訳文が検索結果として得られる。しかしトレーニング中に正解となる訳文を参照してしまってはそれをそのまま出力する恒等写像を学習するのみで、トレーニングコーパスにない文を翻訳できなくなる。そこで、この段階で入力文と等しい類似文は取り除いておく。

次に検索エンジンを用いて取得した $X'$に対して $X$ との類似度を類似度関数によって計算し、$(X', Y')$を類似度の高い順に並び替える。Gu らの提案した類似度関数は以下のようなものである。なお $D_{edit}(S_1,S_2)$は二つの文 $S_1$、$S_2$の編集距離を表す。

$$S_{ed} = 1 - \frac{D_{edit}(X, X')}{\max(|X|, |X'|)}$$

最後に、類似度の高い順に見て、翻訳したい文に含まれる単語のカバー率が高くなる対訳対のみを選択する。

以上のようにして選択した類似文の集合を翻訳メモリ$M$とする。

**Translation Stage**: Translation Stage では Retrieval Stage で得た翻訳メモリ$M$を利用して翻訳を実行する。まず、$M$内の翻訳元の言語の文を注目型ニューラルネットワークに入力して翻訳する。この時、Decoder の各時点における RNN の状態ベクトル$z_t'$、コンテキストベクトル$c_t'$及び$M$から取り出した、正解となるトークン$y_t'$を保存しておく。また、Decoder の RNN と注目機構には直前の Decoder の出力$y_{t-1}$の代わりに$M$から取り出した$y_{t-1}'$を渡す。

次に、入力文 $X$を翻訳する。Encoder は通常の注目型ニューラルネットワークと同じだが、Decoder で各単語の確率 $p(y_t|y_{<t},X)$を計算する際は、保存しておいた$(z_\tau',c_\tau',y_\tau')$を利用する($y_{<t}$は$\{y_1,y_2,\cdots,y_{t-1}\}$を表す)。$f_{gate}$は 1 層の隠れ層を持つニューラルネットワークで $v_g$、$W_g$は重み行列、$M$及び$\lambda$はそれぞれ訓練可能な重み行列及びスカラ値のパラメータである。$\beta_{t,\tau}$は各時点で翻訳メモリ中のどの単語が既に利用されたかをあらわす。

$$
\begin{aligned}
p(y_t|y_{<t},X,\mathcal{M}) &= \zeta_t\ p_{copy}(y_t|c_t,\mathcal{M}) + (1-\zeta_t)p(y_t|y_{<t},X)\\
\zeta_t &= f_{\text{gate}}([c_t;\ z_t;\ \tilde{z}_t])\\
f_{\text{gate}}(v) &= \sigma(v_g^{\mathrm{T}}\}\tanh(W_g\ v))\\
\tilde{z}_t &= \sum_v q_{t,v}z_v'\\
q_{t,\tau} &= \frac{\exp\big(E(c_t,c_\tau')\big)}{\sum_v \exp\big(E(c_t,c_v')\big)}\\
E(c_t,c_\tau') &= c_t^{\mathrm{T}}\ M\ c_\tau' - \lambda\,\beta_{t-1,\tau}\\
\beta_{t,\tau} &= \sum_{v=1}^{t} q_{v,\tau}\cdot\zeta_v\\
p_{\text{copy}}(y_\tau|c_t,\mathcal{M}) &= q_{t,\tau}
\end{aligned}
$$

このようにして各単語の出力確率$p(y_t|y_{<t},X)$を$M$を利用して補正した確率$p(y_t\mid y_{<t},X,M)$を各時点の Decoder の出力とする。

### 2.3.3 類似文を利用したニューラル機械翻訳

Guらの手法における類似度関数は、複数の点で類似度の計算法として最適ではない。例えば “I like this pen I bought yesterday at that store” という文を翻訳したいとき、“I bought a pen at that store yesterday and I like it” と “I gave a pen to Tom yesterday and today he broke it” の２つの文の類似度は Gu らの手法では等しくなってしまう。明らかに “I bought” や “at that store” といったフレーズが一致している前者のほうが類似度が高いが、関係詞節の独立による文全体の順序の入れ替えが考慮されておず、“I”、“pen”、“yesterday”などの一致に引きづられて同じ類似度と判定してしまっている。もちろんこれは極端な例ではあるが、このように局所的には一致しているが文全体の並びが変化している場合に弱い手法であることは確かである。また、類似文は複数取得することが出来るから、後者のように文全体で飛び飛びに幾

つかの単語が一致しているような文を幾つも取得するよりも、人間が翻訳する時のように、フレーズ単位で一致している文を、各フレーズごとに取得できるような手法、すなわち文全体ではなくフレーズレベルでの類似度による手法が最適である。
そこで、本研究では Gu らの提案した類似度関数に代替する、局所的な類似性に基づく類似度関数を提案する。

**N-gram のカバー率による類似度:** Papineni ら(2002)の提案した BLEU は機械翻訳タスクの精度の指標として近年重要な役割を果たしている。よってこれを類似度の指標として使用することを考える。

BLEU は最大 4 の N-gram までしか使用しないため局所的な類似度たりえる。しかし、BLEU をそのまま文の類似度として使用するにはいくつか問題がある。

1. **高次の N-gram が 1 つも共有されず、BLEU スコアが 0 になる**: BLEU は本来翻訳タスクにおける使用を想定されているため、２つの文を比べる際に両方が文全体で同じ意味を表していることが期待される。一方、我々のタスクではトレーニングコーパスから類似している文を取ってきているだけなので、文全体で意味が同じであることはむしろめずらしい。そのため、同じ単語列が現れる確率も低くなってしまう。
2. **短い文に対してペナルティがある**: BLEU では評価対象文が短い場合に N-gram の共有率が不当に高くなってしまい正しく評価できないことから、参照訳より短い翻訳機の出力文に対して簡潔ペナルティを設けている。我々のタスクでは局所的な類似度を測りたいため、２つの文の長短は関係ない。

これらの問題を解消した類似度指標が n-gram coverage である。前者の問題に対する解決策として Lin ら(2004)の提案した BLEU+1 をベースにし、後者の問題に対する解決策として BLEU+1 から簡潔ペナルティを取り除いた。以下がその式である。なお $c(X, \nu_N)$ は文 X 中に N-gram $\nu_N$ が何回現れるかを表す。

$$s_{\mathrm{nc}}(X, X') = \exp\left(\sum_{n=1}^{4} \frac{1}{4} log\, \alpha_n(X, X')\right)$$

$$\alpha_n(X, X') = \begin{cases} \dfrac{m_n(X, X')}{|X| - n + 1} & (\text{if } n = 1) \\ \dfrac{m_n(X, X') + 1}{|X| - n + 2} & (\text{otherwise}) \end{cases}$$

$$m_n(X, X') = \sum_{\nu_n \in \{X \cup X'\}} \min\big(c(X, \nu_n), c(X', \nu_n)\big)$$

match_length は文の長さによらないため、単語数の多い文のほうが有利となるが、注目機構により適切に必要な部分のみに注目できることから問題ないと考えられる。また、文の長さによる正規化が出来ないため、異なる X に対する $s_{ml}(X, X')$ は比較不可能である。よって、他の用途、例えば機械翻訳システムの評価などには転用することが難しい。

### 2.3.4 実験

#### 2.3.4.1 実験設定

実験にはNakazawaら(2016)によるASPECデータセットを使用し、中国語→日本語の翻訳についてトレーニングを行った。トレーニングおよびバリデーション時は70文字以下の文のみを使用した。その結果、トレーニング、バリデーション、テスト用のデータ数はそれぞれ表 1のようになった。

**表 1　実験に使用した中国語・日本語の対訳文数**

| データセット | 対訳文数 |
|---|---|
| トレーニング | 650176 |
| バリデーション | 2022 |
| テスト | 2107 |

ニューラルネットワークに入力するデータに関してはバイト対符号化を施すことであまり現れない単語の翻訳の改善を図った(Senrich, et al., 2016)が、類似文の検索時には符号化されていない平文のテキストで類似度を計測した。

類似文の絞り込みに用いる検索エンジンにはApache Luceneを使用し、アルゴリズムはデフォルトのBM25を用いた。検索結果の最大数を100に制限して翻訳したい文をクエリとして検索を実行した。

実験に使用したニューラル機械翻訳モデルは以下のように設定したものを4つ別々にトレーニングし、それらをアンサンブルしてビーム幅20のビームサーチで探索を行いテストデータセットでBLEUスコアの計算を行った。

トレーニングにはGuらの手法による類似文のみを使用して共用のモデルをトレーニングし、テストの際には各種法による類似文を使用した。トレーニング、バリデーション時には類似文の最大数をそれぞれ2、1に制限し高速化・省メモリ化したが、テスト時には類似文の数に制限は設定しなかった。

Encoderが2層のBiLSTMで隠れ層の次元数は1024、忘却率50%、Decoderは1層のLSTMで隠れ層の次元数は1024、Maxoutの隠れ層の次元数は2176、プールサイズは2、単語の埋め込み次元数は入力、出力ともに640、注目機構の隠れ層の次元数は1024、翻訳メモリのゲート$f_{gate}$の隠れ層の次元数は512、ミニバッチのサイズは最大20、入力と出力の語彙数はそれぞれ18558, 15461を指定した。Decoderの隠れ状態ベクトル、セルベクトルの初期値それぞれをゼロベクトル、訓練可能なパラメータで初期化した。また文末を表す特殊なトークン<EOS>を時刻0における直前の出力として利用した。勾配降下法の最適化アルゴリズムとしてKingmaら(2014)が提案したAdamを利用し、各種パラメータはKingmaらと同様に設定した。match_lengthの$N$には100を設定した。

#### 2.3.4.2 結果

Guらの手法edit_distance、提案手法1 n-gram_coverage、提案手法2 match_lengthによって類似文を選択した時の翻訳精度をBLEUスコアで評価すると表 2のようになった。n-gram_coverage、match_lengthともに既存手法であるedit_distanceを0.1ほど上回っている。n-gram_coverage

のほうが若干ながら良いスコアが出ている。

**表 2　手法ごとの BLEU スコア**

| 手法 | BLEU スコア |
|---|---|
| edit_distance | 43.62 |
| n-gram_coverage | 43.77 |
| match_length | 43.73 |

#### 2.3.4.3　考察

**1 クエリあたりの類似文の数**: テストデータセットについて、各種法により選択された類似文の、1 クエリあたりの数を図 2に示した。n-gram_coverage < match_length < edit_distance の順により多くの類似文を取得している。これは実験結果で得られた BLEU スコアの大小と相関関係にある。このことから、より優れた類似文の選択手法であるほど必要な類似文のみを選択し、1 類似文あたりの翻訳への貢献度を大きくしていると考えられる。

**各種法の手法間の類似度**: テストデータセットについて、各種法により選択された類似文集合の、手法間の類似度を式

$$f_{method-similarity}(S_1,S_2) = \frac{|S_1 \cap S_2|}{|S_1 \cup S_2|}$$

によって調査すると表 3 のようになった。n-gram_coverage と match_length は七割以上一致しており非常に似通った性質を持つことがわかる。一方　edit_distance　と他２つの手法の一致度は共に 25%を下回っており、これら２つの手法が edit_distance と大きく異なることを示している。

**表 3　各手法間の類似文集合の類似度**

| 手法 - 手法 | 一致度(%) |
|---|---|
| edit_distance - n-gram_coverage | 24.90 |
| edit_distance - match_length | 23.66 |
| n-gram_coverage - match_length | 71.10 |

各手法の定性的評価を行ったところ、n-gram_coverage と match_length を比較すると、match_length は一つの文中に複数回現れる単語(副詞など)の一致に引きずられて類似度が高いと判定されていた。これは N-gram の共有数について n-gram coverage ではクエリと類似文とで出現回数の最小値を取るのに対し match_length ではクエリの N-gram 数について最小値を取らないからである。このことから、語の一致を全て数えるのではなくクエリ中のその n-gram の出現数で抑えたほうが良いと考えられる。

また、edit_distance と比較して n-gram_coverage と match_length では 2-gram の一致が多数見られたことから、局所的な類似度が高いことがわかる。かわりに、n-gram_coverage、match_length ともに全体的に文長が長くなっており不要な文字列の割合が増えているが、注目機構によりこれらが翻訳に影響することはないと想定しているので問題ない。

加えて、データセット内での頻出のフレーズが存在することがわかった。このようなフレーズのみの文は、わざわざ重要視して選択しなくても複数の類似文を取得できればその中に 1 つ以上含

まれていることが期待できる。加えて、頻出のフレーズであるなら注目型ニューラルネットワークの学習のみで十分に精度を出すことが期待できるので、あえて翻訳メモリを利用して精度を上げようとする必要性も薄い。よって、こういった頻出フレーズについて類似性の計算時に無視するか、あるいは影響度を低くするために重み付けすることでさらに有用な類似文の選択が可能になると考えられる。重み付けの方法としては、トレーニングコーパスに出現する各単語について逆文書頻度によって重みを計算しておくことが考えられる。

ただしこれ以上に類似文選択の精度を高めたとしても、本研究による BLEU スコアの改善が0.1程度に留まったことから、それに見合う翻訳精度の改善は期待できない。

なお、トレーニングコーパスから類似文を選別するのに使用している検索エンジンのアルゴリズム BM25 は TF-IDF 法と同じく単語の出現頻度 TF と逆文書頻度 IDF を用いて類似度を計算するため、類似度によって並べ替えることなく、検索エンジンによるランキングをそのまま利用すれば上記の頻出フレーズによる問題が解決され、精度が向上する可能性はある。ただし BM25 はフレーズ単位より細かい単語単位での類似度となるため、イディオムやことわざ等逐語訳の適さない翻訳において不利になることが考えられる。

### 2.3.5 おわりに

本研究では注目機構によって入力文の必要な部分に注目して翻訳が可能であることに着目し、局所的な類似性による、より優れた類似文の選択方法を提案した。実験を通して、次の 2 つの結論が得られた。1 つ目は、局所的な類似性に基づく類似文の選択により SEGNMT の精度を向上させる事ができるということである。2 つ目は、本研究で提案した 2 つの手法は多くの場合で似通った類似度の順位付けを行うということである。

今後の課題として、match_length の計算においてクエリ中の N-gram の数を考慮することで頻出単語の一致に過剰に影響されないようにすること、Retrieval stage の最終的な選択で単語のカバー率を用いているのを、N-gram や連続して一致している単語列を考慮したものにすること、検索エンジンによるランキングをそのまま利用する手法を試すことが挙げられる。

また、SEGNMT を転移学習に適用することも考えられる。例えば Chu ら(2017)の研究では転移学習と複数ドメイン学習を組み合わせ、大規模ドメイン外コーパスによるトレーニングの後、小規模ドメイン固有コーパスを混合してニューラルネットワークを転移学習させることで、ドメイン固有コーパスのみでの転移学習に比べ過学習を防ぐことを可能にしている。この際、どのコーパスの文であるかの情報をタグ付けしておくことによって、翻訳精度が改善することが示されている。 SEGNMT を用いれば、タグ付けされていない場合でも、混合コーパスからドメイン固有コーパスの類似文を選択することで、タグ付けと同等の効果が得られると考えられる。類似度の計算においてドメイン固有のフレーズの一致を加味できるという点で、本研究の提案手法は有効性が期待される。

### 参考文献


- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, Neural machine translation by

jointly learning to align and translate, *CoRR*, Vol. abs/1409.0473, (2014).

- Jiatao Gu, Yong Wang, Kyunghyun Cho, and Victor O. K. Li, Search engine guided non-parametric neural machine translation, *CoRR*, Vol. abs/1705.07267, (2017).
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu, Bleu: A method for automatic evaluation of machine translation, *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, (Association for Computational Linguistics, Stroudsburg, PA, USA, 2002), pp. 311–318.
- Chin-Yew Lin and Franz Josef Och, Orange: a method for evaluating automatic evaluation metrics for machine translation, (2004).
- Toshiaki Nakazawa, Manabu Yaguchi, Kiyotaka Uchimoto, Masao Utiyama, Eiichiro Sumita, Sadao Kurohashi, and Hitoshi Isahara, Aspec: Asian scientific paper excerpt corpus, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2016)* (eds. Nicoletta Calzo- lari (Conference Chair), Khalid Choukri, Thierry Declerck, Marko Grobel- nik, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis), (European Language Resources Association (ELRA), Portoroˇz, Slovenia, 2016), pp. 2204–2208.
- Rico Sennrich, Barry Haddow, and Alexandra Birch, Neural machine translation of rare words with subword units, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, (Association for Computational Linguistics, Berlin, Germany, 2016), pp. 1715–1725.
- Diederik P. Kingma and Jimmy Ba, Adam: A method for stochastic optimization, *CoRR*, Vol. abs/1412.6980, (2014).
- Raj Dabre Chenhui Chu and Sadao Kurohashi, An empirical comparison of domain adaptation methods for neural machine translation, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL2017 short)*, (Vancouver, Canada, 2017.7).

# 2.4 パターンを用いた特許文請求項の構造解析

山形大学名誉教授　横山晶一

## 2.4.1 はじめに

特許文において、課題や解決手段、請求項 1 の部分が、長い文で、複雑な係り受け構造を持つことが多いということは言うまでもない[1]。

統計的機械翻訳(SMT)においては、請求項の構造を sublanguage と捉えて組み込むことによって、翻訳品質が著しく改善されることが報告されている[2, 3]。Sublanguage としては、並列助詞「と」や機能語が一部扱われているが、意味構造を的確に把握するためには、なお解析が必要と考えられる。

また、ニューラルネットを用いた自然言語処理では、大量のデータを学習に用いた後の処理の修正が難しいという側面が指摘されているが、当初の学習時にある程度バイアスのかかったデータを学習に用いることによって、方向性の異なるデータを同時に学習することによる不明確化を避けられるかもしれないという期待がある。実際にうまくゆくかどうかはきちんと確認する必要があるが、やってみる価値はありそうである。ただし、そのためには、データの言語学的な性質を、人間の眼でもう少し深く解明しておく必要がある。

以下では、請求項 1 に含まれる構造にどのようなものがあるかを引き続き考察し、さらに詳細な係り受け構造や意味構造を得るための可能性について言及する。すでに、これまでそれほど調べてこなかった一定の動詞や機能語に着目して、それが請求項の中でどのような役割を果たし、また、係り受け等にどのような影響をもたらすかを考察した結果については、簡単な発表を行っている[4]が、本稿では、その後に追加したデータを加えて新たに考察した結果について述べる。

## 2.4.2 資料・調査方法

2003 年の特許公開公報（特開 2003-180493～180625）の中から、請求項 1 の一文が 120 字を超えるものを 100　選び出した（二文以上から成るが、一文が 120　字を超えるものも含む)。このうち前半の 50　文は、前回まで用いたデータで、すでに一部の結果を発表している[4]が、今回改めて見直しを行った。後半の 50 文は、新たに調査したものである。文字数の内訳を表 1 に示す。

表 1　調査特許文請求項 1 の文字数の内訳

| 文字数 | 120～199 | 200～299 | 300～399 | 400 以上 |
|---|---|---|---|---|
| 特許数 | 45 | 40 | 12 | 3 |

この表から分かるように、約半数が 200 字未満である。最も短い文は 125 文字、最も長い文は 625 文字、平均で約 226 字であった。この比率や平均値はこれまで調査した別の公開特許公報での分布[1]とほぼ同じである。

これらについて、次のような観点から分析を行った。

(1) 機能語、改行による文の分割

機能語とは、以前の報告でも述べたように、ここでは、日本語の複数の形態素から成る複合語の中で、いわゆる「つなぎ言葉」的な役割をになうものと定義し、「～において」、「～であって」、「～に関して」、「特徴とする」、「特徴として」などを示す。

長い文からなる請求項 1 には、文自体に改行を入れることによって、文の構造や係り受けを明確化しているものもある。ここではそれについても調査した。

(2) 並立助詞による並列構造

典型的には「と」による並列構造の構築があげられる。すでに述べたことと重複する部分もあるが、それらについて調査した。

(3) 動詞「備える」、「有する」の役割

助詞ではなく、動詞を並列させることによる並列構造も多く見られるが、特に設備等に言及した特許文の請求項には、「備える」、「有する」などの特定の動詞が頻出する。これまで、これらの動詞については余り調査を行ってこなかったが、今回改めてその役割について調べる。

以下では、これらの調査結果について述べる。

### 2.4.3 機能語・改行による文分割と係り受けの明確化

(a) 機能語「であって」、「において」

機能語については、すでに述べたように、典型的な形として、いくつかの機能語で、「～[名詞句 A] [機能語]、…した[名詞句 A]」という形をとった場合には、多くが機能語を境として前後に分割できることが分かっている。前後に分割できれば、各々の名詞句の長さが短くなって、より詳細な解析をすることが可能になる。

今回も、「であって」、「において」（1 文のみは「に於いて」という表現になっている）について調査した。

今回調査した特許文中に出現した機能語の内訳を表 2 に示す。

表 2　特許文中の機能語の内訳

| | 典型 | 非典型 | その他 | |
|---|---|---|---|---|
| | | | 読点あり | 読点なし |
| であって | 20 | 3 | 1 | 2 |
| において | 26 | 3 | 1 | 3 |

100 文のうち、「であって、」と読点を含む形で典型的に名詞句が前後に分離されるものは 20、「において、」では 26 例あり、分離できるが典型例ではない（つまり名詞句の形が機能語の直後と句の末尾で異なる）ものが、それぞれ 3 ずつあることを示している。その他は、入れ子構造の下部、すなわち修飾句の一部になっていて、分離できないもので、たとえば、「であって、」の場合には、もう一つの「であって、」の非典型例とともに出現したものが 1 例のみある（図 2 に示す）。この場合には読点を含んでいるが、含まないものも存在する。

次に、改行であるが、長文をなるべく分かりやすくするために、文中に改行を入れているものが、今回の調査では 48 文あった。すべての文が、係り受け関係の明確化に役立つ改行構成になっている。

図 1 に、「において、」の典型例、図 2 に「であって、」の非典型例と入れ子、改行をともに含む例を示す。図 2 の文中の後半の「であって、」は、前述のように、修飾句下部の修飾にのみ寄与しており、文の分割にはかかわっていない。

食器洗い機本体の内部に収容した被洗浄物を洗浄するための洗浄室と、この洗浄室に洗浄水を導くための給水路とを設けた食器洗い機**において、**
食器洗い機本体に引き出し可能な給水タンクを設け、その給水タンクを食器洗い機本体から引き出した状態で給水タンクに給水可能なことを**特徴とする**食器洗い機。

図 1　「において、」の典型例と改行を含む例（特開 2003-180596）

所定の照明光により照射された部位を撮像する撮像素子を先端部に備える電子スコープ**であって、**
前記撮像素子を含む前記電子スコープ全体の電気的駆動に関する回路が形成された、第一の基板および第二の基板を有し、
前記第一の基板は、所定位置に少なくとも一つの第一基板側ケーブル接続部を有し、
前記第二の基板は、所定の一端近傍に、所定値以上の高さを有する少なくとも一つの部品と少なくとも一つの第二基板側ケーブル接続部とを有し、
前記第一の基板と前記第二の基板とは、前記少なくとも一つの部品と前記少なくとも一つの第二基板側ケーブル接続部とが前記第一の基板に対して露出する状態**であって、**かつ該少なくとも一つの部品と該少なくとも一つの第二基板側ケーブル接続部と前記少なくとも一つの第一基板側ケーブル接続部とが近接して配設される状態で、重ね合わせて固定されることを**特徴とする**電子スコープの基板構造。

図 2　「において」の非典型例と入れ子、改行を含む例（特開 2003-180624）

(b) 「特徴とした」、「特徴とする」を含む用例

図 1, 2 でも見られるが、最後の名詞句の直前に、「特徴とする」あるいは「特徴とした」という修飾語句を伴った例は非常に多く、「特徴とする」を含む文は 67（「特徴する」というのが別に 1 つあったが、これは「特徴とする」のミスプリントと思われる)、「特徴とした」を含む文は 8、合わせて実に 76 の文がこの句を含んでいることが分かる。

つまり、より正確に述べると、前節で述べた名詞句の多くは、「～[名詞句 A] [機能語]、…[特徴とする] （または[特徴とした] ）[名詞句 A]」と言う形になっている。この構造を持つ特許文請求項がすべてこのような形をとるわけではないが、図 1 に明らかなように、非常に多くの例でこの形が見られている。

表 2 の典型例で調査したところによると、「であって」の 20 例のうち 15 例が「特徴とする」（うち 1 例は上記の「特徴する」）を伴っており、1 例のみが「特徴とした」を伴っている。すなわち、80%は「特徴とする」ないしは「特徴とした」を伴っていることになる。

また、「において」では、26 例すべてが、「特徴とする」と共起している。「において」で名詞句が分割できる場合に、常に「特徴とする」を伴うかどうかは、調査をもっと拡大しないと確実なことは言えないが、多くの例で上記のようなパターンが出現していることは言える。

### 2.4.4 並立助詞「と」による並列構造

特許文請求項では、全体として長い名詞句を作るために、助詞「と」で並列構造を作る場合が多く見られる。今回もこのような例が 39 見出された。また、動詞や形容詞の連用形を繰り返すことによる並列も 61 あった。このうち 21 例は、両方が用いられており、階層構造を形成する場合もあるので解析には注意が必要である。

並立助詞「と」と、動詞の連用形による並列構造が用いられた例を図 3 に示す。

| |
|---|
| 枠体内に進退可能に設置した洗浄槽と、この洗浄槽内を給排水する手段と、前記洗浄槽内に乾燥空気を送り込む手段と、洗浄槽内の洗浄液および乾燥空気を加熱するヒータと、これらを制御する制御装置を**備え**た食器洗い機**において、**<br>前記ヒータは、前後方向に傾斜するように設置し、高い方の部位から温度ヒューズに熱伝達するように構成したことを**特徴とする**食器洗い機。 |

図 3　助詞、動詞等による並列構造の例（特開 2003-180602）

図に示した例では、「と」による名詞の並列が、まとめて「これら」にかかり、それがさらに「制御する制御装置を備えた」という修飾語句を通じて、句末にかかるという典型的な並列構造になっていることがわかる。動詞「備える」については、この図に見られるような形式ではなく、「～を備え､」といったパターンで、修飾構造を比較的容易に捉えること

ができる。これについては次節で触れる。

### 2.4.5 動詞「備える」を用いた並列構造

今回調査した事例では、動詞「備える」が比較的多く見られ、中でも「～を備え､」といったパターンを持つものが 18 例見つかった。典型的な例を図 4 に示す。

| |
|---|
| 洗浄槽と、前記洗浄槽内に配置し食器類を収納する食器かごと、小物食器類を収納する小物入れと、食器類に洗浄水を噴射する洗浄ノズルと、前記洗浄ノズルに洗浄水を供給する洗浄ポンプと**を備え、**前記小物入れは、前記食器かごに設けた複数の支持ピンで保持し、前記複数の支持ピンの一部は略U字形状として前記小物入れの一側面を軸支し、他の複数の支持ピンは前記小物入れの他の側面を上下方向に保持した食器洗浄機。 |

図 4　「～を備え､」という形を含む例（特開 2003-180607）

この例では、いくつかの名詞が「と」による並列構造で「備える」にかかっており、これが「保持し」などの動詞と並列になって最後の「食器洗浄機」にかかる構造となっている。上記のような例では、いずれも最後に来る名詞の直下か、機能語「であって」､「において」によって分離された名詞の直下にこの構造を有していることが確認された。

そこで、同様の知見が別のデータでも得られるかどうかを確認するために、以前調査した 150 のデータ[1]を再度確認したところ、「～を備え､」と言う例は 22 例あったが、このうち、係り受けがかなり下部の修飾句になっていると考えられる 2 例を除き、やはり最後に来る（最上位の）名詞の直下か、機能語の直下の修飾句となることが確認できた。図 5 は､以前の文献[1]の図 1 と同じものであるが､これが典型的な例の一つである。ここには、機能語、改行といった請求項の特徴がいくつか含まれている。

ここでは、動詞「有する」についても調査したが、この語はさまざまな位置に出現し、ある特定の階層での修飾句と言う形を取らないため、今回明確な結論を出すことはできなかった。

| |
|---|
| バンド駒をピンによって連結し構成されるバンドのバンド構造**であって、**<br>ピンを固着せず、ピンを挿通させるためのバンド駒のピン穴は、ピンを被覆するパイプを**備え、**<br>ピンを固着するバンド駒のピン穴は、ピンの表面又は／及びピン穴の内壁に施されたメッキによって当該ピンと溶接されていることを特徴とするバンド構造。 |

図 5　「～を備え､」を含む以前の典型例（特開 2003-180414）

### 2.4.6 問題点と今後の検討

機能語については、前に主張した文の分割の可能性が、今回も裏付けられた。今後はこの観点からさらに調査対象を広げていきたい。すでに述べたように、sublanguage という形で組み込むことによって、SMT での翻訳精度が上がることが示されている[2, 3]が、パターンを用いることによって、NMT においてもさらに精度を上げられることが考えられる。

今回、並列や機能語などの複数要因が絡み合った諸相については、やや解析が不十分なところがある。今後、これら複数要因がどのように関係しているかについてさらに調査を進め、知見を深めていく予定である。公開特許には、類似のものを同時期にまとめて出願するために、ある程度の偏りも見られるので、幅広い調査を行った上で知見を深める必要がある。

動詞については、「～を備え、」というパターンは確認できたが、その他の動詞において、明確なパターンがあるかどうかの確認が必要である。今後さらにデータを増加させて調査する予定である。

また、前回調査の対象とした「該」、「前記」といった照応的な語については、今回もサンプル数が少なく、十分な調査が行えなかった。これらについても今後さらに検討していく予定である。

### 参考文献


[1] 横山晶一：特許文請求項の構造に関する調査、平成 28 年度 AAMT/Japio 特許翻訳研究会報告書 (2017) pp.31-36,
http://aamtjapio.com/kenkyu/files/kenkyu05/AAMT_Japio_20170324.pdf

[2] Masaru Fuji, Atsushi Fujita, Masao Utiyama, Eiichiro Sumita, Yuji Matsumoto: Patent Claim Translation based on Sublanguage-specific Sentence Structure, Proceedings of MT Summit XV, vol.1, (2015) pp.1-16

[3] 富士秀、藤田篤、内山将夫、隅田英一郎、松本裕治：特許請求項に特有の文構造に基づく英中日特許請求項翻訳、自然言語処理 Vol.23, No.5 (2016) pp.407-435

[4] 横山晶一：パターンによる特許文請求項の構造解析、Japio Yearbook (2017) pp.298-301

# 2.5 外国特許文献調査のためのFタームの利用

静岡大学　綱川　隆司

## 2.5.1　はじめに

特許の出願・利用に伴う先行技術を検索する際には IPC（国際特許分類）等の特許分類が用いられる．日本国内の特許を調べる場合，FI および F タームによる詳細な分類による検索が可能であるが，日本国外の特許には付与されておらず，これらを用いて直接検索することはできない．また，任意の特許文献に対し F ターム等の分類を機械学習により自動付与する方法が検討されているが，付与例が少ない F タームについては学習が困難である．本研究では，外国特許文献を F タームの観点から直接検索するための方法を検討する．

## 2.5.2　CPC（共同特許分類）と FI・F ターム

国際的に用いられている IPC は最も細かい分類であるサブグループを含めて約 7 万項目からなっている．また各国の特許庁は独自の分類体系を用いてより細かい分類を付与している．国や地域によって特定の分野に対する適切な分類粒度が異なるという事情はあるものの，国際的に通用可能な分類体系をより詳細化する動きがあり，IPC 自体を詳細化するプロジェクトも稼働していた（太田, 2013）．一方，2013 年には EPO（欧州特許庁）及び USPTO（米国特許商標庁）が IPC の下位分類に位置づけられる CPC（共同特許分類）の運用を開始している．CPC は EPO の内部分類であった ECLA および ICO をベースとしており，EPO が管理する特許および EPO が ECLA および ICO の付与を行っていた米国特許については過去の文献に遡ってアクセスが可能と考えられる．さらに，中国の SIPO および韓国特許庁（KIPO）でも CPC 付与に向けた動きが見られる（塩澤, 2016）．

日本特許庁（JPO）では IPC の下位分類に当たるものとして FI を用いており，FI が持つ分類について CPC との調和を図る取り組みがなされている一方で，現在のところ直接 CPC を付与する予定はない．CPC と FI の対応関係を調べる方法の一つとして，特許庁では分類対照ツール[1]を用意しており，IPC，FI（F タームのテーマコードを含む）および CPC 間の関係を検索できる．

FI は CPC と同様に IPC よりさらに細かい分類としての性質を持つが，F タームは FI を所定分野ごとに複数の技術的観点から細分類したものであり，複数の観点を組み合わせることで関連特許をより効率的に絞り込むことを目的に定められている．F タームの例を表 1 に示す．まずあらかじめ区分された技術範囲を「テーマ」として英数字 5 桁のテーマコードで表し，テーマごとに定められる複数の「観点」（英字 2 桁）に対して数字 2 桁を付加した F タームが割り当てられる．名称の前のドットは階層の深さを表しており，表 1 に対応する階層構造は図 1 のようになる．

[1] http://www.jpo.go.jp/cgi/cgi-bin/search-portal/narabe_tool/narabe.cgi

表 1　F タームリストの例

| 5B034 | ハードウェアの冗長性 | | | | | | |
|---|---|---|---|---|---|---|---|
| **観点** | **F ターム** | | | | | | |
| AA | AA00 | AA01 | AA02 | AA03 | AA04 | AA05 | … |
| | 受動的冗長 | ・二重化 | ・・照合 | ・・・圧縮照合 | ・多重化 | ・・多数決 | … |
| BB | BB00 | BB01 | BB02 | BB03 | BB04 | BB05 | … |
| | 能動的冗長 | ・切替 | ・・予備切替 | ・・・共通予備 | ・・選択 | ・・・信頼度 | … |
| | | BB11 | BB12 | BB13 | | BB15 | … |
| | | ・再構成 | ・・緊急制御回路 | ・・機番変更 | | ・切離し | … |
| … | … | … | … | … | … | … | … |

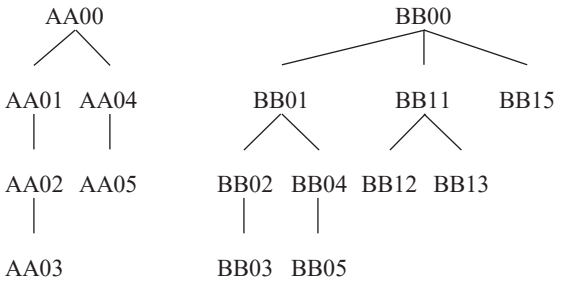



図 1　F タームリスト階層の例

表 2　特許分類体系と項目数

| 分類体系 | 項目数 |
|---|---|
| IPC | 73915（2018 年 1 月） |
| CPC | 260741（2018 年 2 月） |
| FI | 195017（2017 年 12 月） |
| F ターム | 401477（2018 年 2 月） |

F タームはそれ自体が階層構造を持つと同時に当該特許のキーワードを表すような性質を持っている．これにより F タームを利用した特許検索では特定の観点を共有する特許を横断的に検索できることが期待される．また，FI や CPC と比べても項目数は多い（表 2）．

特許庁は，2011～2013 年度公開の中国特許文献の一部の重要分野に対し，FI および F タームを付与したデータを公開している[2]．対象となる分野はプリンターや光学関連等を含む 79 のテーマが選ばれており，付与対象となった特許について該当するすべての F タームの付与がなされている．これにより対象分野の F タームについては付与対象となっている中国特許文献について直接検索が可能である．本データは 2015 年の更新を最後に付与作業は終了している．外国特許文献への人手による F タームの付与には大きなコストがかかることが見込まれるため，本データのような付与作業を行う上では外国特許に付与され得る F タームを自動的に絞り込むといった効率化が一層図られる必要があると考えられる．

### 2.5.3　F タームによる外国特許検索方法

F タームによって外国特許文献を検索する方法は，大きくわけて二つの方式が考えられる．一つは多くの国で付与が進む CPC と F タームとの対応関係を明らかにすることで，新たな分類付与や推定をすることなく F タームに対応する CPC を持つ特許を検索する方法である．もう一つは，検索対象とする F タームと対応付く特許文献集合を得ることで，それらに類似する外国特許文献を探索する方法である．以下それぞれについて述べる．

[2] https://www.jpo.go.jp/shiryou/s_sonota/china_patent.htm

表 3　CPC 分類の例

| | | |
|---|---|---|
| G06K9/4614 | ・・・・・{filtering with Haar-like subimages, e.g. computation thereof with the integral image technique (biologically-inspired filters such as Gabor wavelets or local ICA kernels G06K9/4619; local approaches in face detection or representation G06K9/00248, G06K9/00281)} | ・・・・・　{ハール状サブイメージによりフィルタリングするもの，例．一体型画像技術によるその演算（ガボールウェーブレットまたは局部的ＩＣＡカーネル等の生物学的に示唆されたフィルタＧ０６Ｋ９／４６１９;顔検出または表現における局部的アプローチＧ０６Ｋ９／００２４８，Ｇ０６Ｋ９／００Ｆ２Ｌ）} |

### 2.5.3.1　F タームと CPC の対応付けによる検索方法

2.5.2 で述べたように F タームは独自の多角的観点に基づく分類であり，CPC と F タームが単純な一対一の関係で対応付くことは期待されない．例えば，CPC の分類 G06K9/4614 には表 3 のようなタイトルが付与されている．この分類の上位分類で FI と直接対応づくのは G06K9/46（・・画像の特徴の抽出）であり，分類対象ツールによれば F タームのテーマコード 5B064（文字認識）が対応しているが，このテーマに含まれる F タームに対応するものは見当たらない．対応する可能性のある F タームとして 5K041HH07（信号処理＞・線形変換，直交変換）があり得るが，それぞれの分類が付与されるかどうかは特許文献ごとに判断されることになると考えられる．

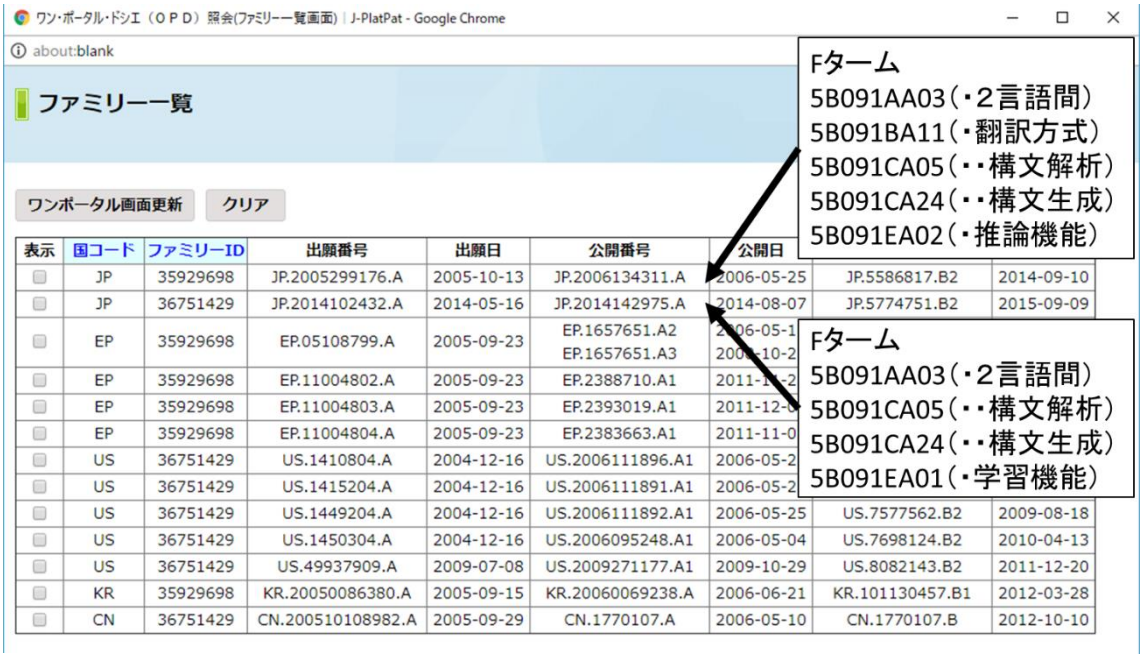
現状では F タームと CPC の直接の対応付けは難しいが，CPC の一部について対応付く F タームを発見しておくことは有用と考えられる．

### 2.5.3.2　F タームが付与された特許文献集合による検索方法

特定の F タームが付与された日本の特許文献集合は当該 F タームに関連する何らかの共通部分を持っているはずである．そこで，日本の特許文献と外国の特許文献を文書として直接比較することで，当該 F タームが付与された特許文献集合と類似する外国特許文献を検索することを考える．

日本の特許文献と外国の特許文献の類似性を測るクロスリンガルな方法として，要約の英訳や機械翻訳を用いる方法のほか，パテントファミリーを用いる方法が考えられる．以下，それぞれについて順に述べる．

日本の特許文献に付与されている要約の英訳（PAJ）や明細書を英語に機械翻訳した結果から，当該 F タームに対応する英文が得られる．これにより，同一言語内での文書間類似度による文書検索手法が適用できることとなる．本手法はどの特許文書でも適用可能でありコストも低いという利点を持つが，要約の英訳を用いる場合は特許の持つ情報が制限される．また，機械翻訳を用いる場合は翻訳精度に依存するほか，特定の専門用語がキーワードとなって当該 F タームを構成している場合に正しく翻訳できないおそれがあり，検索精度を下げる可能性があることが短所である．




| 表示 | 国コード | ファミリーID | 出願番号 | 出願日 | 公開番号 | 公開日 | | |
|---|---|---|---|---|---|---|---|---|
| | JP | 35929698 | JP.2005299176.A | 2005-10-13 | JP.2006134311.A | 2006-05-25 | JP.5586817.B2 | 2014-09-10 |
| | JP | 36751429 | JP.2014102432.A | 2014-05-16 | JP.2014142975.A | 2014-08-07 | JP.5774751.B2 | 2015-09-09 |
| | EP | 35929698 | EP.05108799.A | 2005-09-23 | EP.1657651.A2<br>EP.1657651.A3 | 2006-05-1<br>2006-10-2 | | |
| | EP | 35929698 | EP.11004802.A | 2005-09-23 | EP.2388710.A1 | 2011-1 | | |
| | EP | 35929698 | EP.11004803.A | 2005-09-23 | EP.2393019.A1 | 2011-12- | | |
| | EP | 35929698 | EP.11004804.A | 2005-09-23 | EP.2383663.A1 | 2011-11-0 | | |
| | US | 36751429 | US.1410804.A | 2004-12-16 | US.2006111896.A1 | 2006-05-2 | | |
| | US | 36751429 | US.1415204.A | 2004-12-16 | US.2006111891.A1 | 2006-05-2 | | |
| | US | 36751429 | US.1449204.A | 2004-12-16 | US.2006111892.A1 | 2006-05-25 | US.7577562.B2 | 2009-08-18 |
| | US | 36751429 | US.1450304.A | 2004-12-16 | US.2006095248.A1 | 2006-05-04 | US.7698124.B2 | 2010-04-13 |
| | US | 36751429 | US.49937909.A | 2009-07-08 | US.2009271177.A1 | 2009-10-29 | US.8082143.B2 | 2011-12-20 |
| | KR | 35929698 | KR.20050086380.A | 2005-09-15 | KR.20060069238.A | 2006-06-21 | KR.101130457.B1 | 2012-03-28 |
| | CN | 36751429 | CN.200510108982.A | 2005-09-29 | CN.1770107.A | 2006-05-10 | CN.1770107.B | 2012-10-10 |

図 2　パテントファミリーの例（ワン・ポータル・ドシエ照会）

パテントファミリーは，同じ発明について複数の国に出願した特許のまとまりであり，“同一の優先権またはその優先権の組み合わせを持つすべての文献を含むもの”と定義されている．一例として公開番号特開 2014-142975「ツリーレット翻訳対の抽出」の特許文献から同一のファミリー ID を持つ各国の特許を特許情報プラットフォームのワン・ポータル・ドシエ(OPD)照会[3]により検索した例を図 2 に示す．

検索対象 F タームが付与された特許のうち，パテントファミリーの関係から得られる英語で記述された特許文献を用いることで，機械翻訳等による方法と比べてより正確な外国特許文献を得ることができる．しかし，パテントファミリーを持つ日本の特許文献は限られるため，当該 F タームにパテントファミリーがなければこの方法は適用できず，存在する場合でも限られた数の特許文献から外国特許を検索しなければならない．また，パテントファミリーの定義上，パテントファミリーに含まれる各国の特許は同一の内容とは限らず，互いに関連した内容に過ぎない場合もあり，パテントファミリーに含まれる特許が当該 F タームを付与すべきものかどうかの保証はない．

### 2.5.4　外国特許文献への F ターム付与における階層的分類手法の適用について

本節では，外国特許文献の F タームによる検索を可能にするため，事前に外国特許文献に対して F タームを自動付与する方法について考える．2.5.3.2 で挙げた方法により，ある F タームが付与される可能性の高い外国特許文献の集合が得られる．それを訓練データとした教師あり学習を行うことで，一般の外国特許文献に適用できる分類器が構築できる．しかし，実際には F タームによる分類が非常に細かいために，特定の F タームが付与された特許文献数が限られることから教師あり学習が実用的に適用可能な F タームは少数に留まる．本稿では特許分類体系の階層構

[3] https://www10.j-platpat.inpit.go.jp/pop/all/popd/POPD_GM101_Top.action

造を考慮した方法を検討する．

特許分類体系はすべて階層的に整理されており，一般にはある分類が付与された特許文献はその上位の分類にも属すると解することができる．最上位層の分類を除き，中間層の分類も付与可能となっている．F タームの場合は，各テーマコードの観点のレベルが付与可能な最も上位の分類であり，通常はある F タームが付与された特許文献にその上位の F タームを付与することはしない．このとき，中間層の F タームが付与される特許文献の条件は，その下位にあるどの F タームにも該当しないことであると考えられる．ここで，中間層の F タームによる特許文献検索を行う際，以下の二つの場合が考えられる．

(a) 当該 F タームおよびその下位分類のいずれかが付与された特許文献を検索したい場合
(b) 当該 F タームの下位分類に含まれるものを除き，純粋に当該 F タームが付与された特許文献を検索したい場合

(b) には，例えば当該 F タームが示す分野を全体的に含む内容の文書に検索対象を絞りたいような場合が含まれる．

適用可能な階層的分類手法には以下の三つが挙げられる（Silla and Freitas, 2011）.

- フラット分類
- トップダウン分類
- 大域的分類

フラット分類は階層構造を考慮しない分類方法であり，上位と下位の分類が同一の特許文献に同時に付与されることがあり得る．上記（a）の場合のみを扱うのであれば，上位下位関係にある F タームが共に付与された場合に下位の F タームのみを残すという方法が採用できるが，もともと下位の分類のみが付与された特許文献との差がなくなる分だけ得られる情報が少なくなる．一方, (b) の場合を扱うには上位と下位の F タームのいずれを優先すべきか判断する必要が生じる．

階層構造を考慮し，上位の分類から順に付与すべきかどうか判定するトップダウン分類を取ることで上記の問題は生じなくなる．例えば図 1 においては，まずテーマコード 5B034 の各観点 AA00, BB00, … に属するかどうかをそれぞれ判定し，属すると判定したものの下位分類のみを再帰的に判定していく．最終的に属するとされた最下層の分類のみを付与する．このとき，中間層の分類の判定においては，その下位分類が付与されたものも全て含んでいるものとして扱う．

トップダウン分類では，中間層での判定で属さないと判定されてしまうと下位分類の判定を行わないため，下位層の分類が付与されにくくなるブロッキング問題が発生する．これを緩和するために判定基準を緩めると上位層で付与される分類が多くなり適合率が低下するおそれがある．日本の特許文献への F ターム付与タスクにおいて階層的 SVM を適用した結果, フラットな SVM に比べ精度が低下したという報告がある（Li et al., 2007）.

上記のいずれにも属さず，階層構造を学習して多値分類を行うようなものを大域的分類と呼ぶが，このような分類器を設計するのは難しい．一つの可能性として，ニューラルネットワークの多層構造をそのまま階層的分類に応用した手法（Wu and Saito, 2017）が挙げられる．図 3 のように層間のニューロンを階層構造と等価になるように結びつけ，各層にストップニューロン（右端）を置く．分類推定時には上から下に向かって確率値が最適になるようなパスを探索し，スト

ップニューロンの直前のニューロンに対応付く分類を出力することで階層構造を考慮した分類を実現している．Wu and Saito (2017) はウェブサイトのカテゴリ分類である Open Directory Project の 11497 カテゴリを用いた実験において精度向上およびパラメータの大幅な削減を示している．

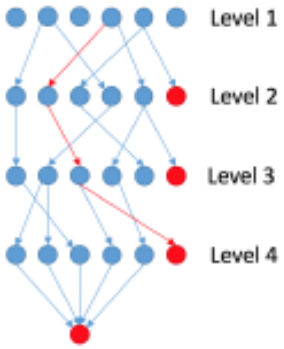



図 3　ニューラルネットワークによる階層的分類（Wu and Saito (2017) より引用）

### 2.5.5　おわりに

本稿では F タームによる外国特許文献の検索を可能にするための方法として，CPC との対応付け，機械翻訳等による文書間検索，および特許分類の階層構造を考慮した階層的分類手法について検討した．今後の方針として，特定の F タームを表す embeddings のような知識表現を得る方法や，ニューラルネットワークによる文書分類手法の適用が挙げられる．

### 参考文献

Li, Y., Bontcheva, K., and Cunningham, H. (2007). SVM based learning system for F-term patent classification. In *Proc. of NTCIR-6 Workshop Meeting*, pp. 396-402.

Silla, Jr., C. N. and Freitas, A. A. (2011). A survey of hierarchical classification across different application domains. *Data Mining and Knowledge Discovery*, Vol. 22, No. 1-2, pp. 31-72.

Wu, Z. and Saito, S. (2017). HiNet: Hierarchical classification with neural network. ICLR 2017 workshop. arXiv:1705.11105, pp. 1-6.

太田良隆. (2013). CHC プロジェクトの現状およびその行く末について. 情報の科学と技術, Vol. 63, No.7, pp. 277-281.

塩澤正和. (2016). 特許分類に関する最新動向. *Japio YEAR BOOK 2016*, pp. 38-43.

# 3. 機械翻訳評価手法

# 3.1 拡大評価部会の活動概要

奈良先端科学技術大学院大学　須藤　克仁

AAMT/Japio 特許翻訳研究会では、2012 年度より機械翻訳の評価に関する議論を深めるための下部組織として「拡大評価部会」を設置し、研究会委員以外の識者も加えた活動を展開している。

拡大評価部会では、翻訳評価を様々な視点から捉えるため、以下の３つのサブグループに分かれ個別の課題についての検討を行い、部会会合において部会全体で議論している。

・自動評価サブグループ（磯崎 秀樹、越前谷 博、須藤 克仁）
　自動評価尺度の改善の検討、およびメタ評価に関する議論を行う
・テストセットサブグループ（江原 暉将、長瀬 友樹、王 向莉）
　機械翻訳における典型的な課題を含む評価用データセットの設計・作成・評価を行う
・人手評価サブグループ（中澤 敏明、園尾 聡、後藤 功雄）
　種々の人手評価方法の検討、および実際の評価データの分析・議論を行う

2017 年度は 3 回の部会会合を開催した。

・2017 年 5 月 12 日　　年度活動計画の策定
・2017 年 10 月 13 日　　中間報告および今後の活動内容についての議論
・2018 年 3 月 2 日　　年度活動報告および報告書内容の確認

自動評価サブグループは、近年のニューラル機械翻訳の進展に伴う機械翻訳の質の向上と表層レベルの評価の限界を鑑みた新たな自動評価手法について議論した。3.2 で自動評価サブグループとしての課題認識について、3.3 で単語分散表現を用いる自動評価法について報告する。

テストセットサブグループでは、中国語特許文献において頻出する分離表現に注目したテストセットの拡充について議論した。3.4 で中国語の分離表現とその日本語への翻訳パターンの収集および評価について報告する。

人手評価サブグループでは、国際ワークショップ WAT での機械翻訳共通タスクにおける人手評価結果の分析、自動評価との関係の分析を行った。3.5 でその結果について報告する。

# 3.2 機械翻訳の自動評価の現状

岡山県立大学　磯崎　秀樹
北海学園大学　越前谷　博
奈良先端科学技術大学院大学　須藤　克仁

機械翻訳における自動評価は、機械翻訳システムあるいは機械翻訳結果の最終的な良し悪しを測る尺度として重要であるとともに、機械翻訳システムの改善の指針あるいは学習時の目的関数として、この二十年近くに渡る統計的機械翻訳の発展を支えてきた存在と言える。特に 2002 年に IBM が提案した BLEU は現在でもデファクトスタンダードとして用いられ続けている。

BLEU およびその後多く提案された自動評価手法は単語の表層情報のみを利用するものが主流で、参照訳との比較のみで簡便に評価が可能である反面、意味は正しくとも訳語選択が参照訳と異なるような場合に誤って低い評価をしてしまったり、文意が異なっていても表層的な一致度が高ければ比較的高い評価をしてしまったりする問題が長く指摘されてきたことも事実である。しかしながら、フレーズベース翻訳を初めとする統計的機械翻訳は対訳データから得られる対訳素片を組み合わせて翻訳を実現しており、意味には踏み込まない表層的な共起関係に基づく訳語選択が行われることが多かったために、その問題が深刻化していなかったと言える。

ところが、ニューラルネットワークを利用した深層学習技術の急速な発展によって、2014 年に発表されたニューラル機械翻訳を皮切りに機械翻訳のパラダイムは大きく変化した。ニューラル機械翻訳において、単語や文は離散的な記号ではなく連続空間上の点として扱われるようになり、単語の意味的な近さを考慮しつつ強力な言語モデルによる非常に流暢な文を生成するような訳語選択が可能となった。また、注視(attention)と呼ばれる状態参照機構によって、長く機械翻訳技術の深刻な問題であった語順の誤りが大幅に軽減された。

こうした変化を受けて、表層の一致という制約から脱却し、意味的な類似性に着目した自動評価を行う手法が提案されている。例えば、単語 n-gram の分散表現を利用した MEANT 2.0[1] は国際会議 WMT の評価尺度共通タスクにおいて、文単位メタ評価で人手評価との高い相関(相関係数 0.6 前後)が得られることが示された[2]。システム単位メタ評価においては文字レベルで表層を比較する手法 (chrF や CharacTER 等)が人手評価と非常に高い相関(相関係数 0.95 以上)を示した[2]が、MEANT 2.0 の結果も大きく劣るものではなかった。

システム単位メタ評価は翻訳システムの平均的な優劣を予測できることを示すものでしかなく、各文を正しく評価できていることを保証するものではない。文単位の評価は小さな変化に敏感で評価の分散が大きくなりがちなこと、比較対象となる人手評価自体の評価揺れの影響を受けやすいこともあって性能検証が非常に難しいという問題があるが、今後は文単位での評価の確度をさらに高めることが重要となることは疑いない。

## 参考文献

[1] C.-K. Lo, MEANT 2.0: Accurate Semantic MT Evaluation for Any Output Language, Proc. of the Second Conference on Machine Translation (WMT 2017), pp. 589-597, 2017.

[2] O. Bojar, Y. Graham, A. Kamran, Results of the WMT17 Metrics Shared Task, Proc. of the Second Conference on Machine Translation (WMT 2017), pp. 489-513, 2017.

# 3.3 単語の分散表現と語順情報を用いた自動評価法の提案

北海学園大学　越前谷　博

## 3.3.1　はじめに

近年、ニューラル翻訳[1][2]が急速な進展を遂げている。従来の機械翻訳手法に比べ、ニューラル翻訳ではより意味を考慮した翻訳が可能となった。そのためニューラル翻訳が出力する翻訳文に対して、より高い精度で評価するためには、自動評価法もまたそれに追随したものであることが望ましいと考えられる。そこで、本報告では、意味を考慮した新たな自動評価法を提案する。提案手法では、単語の意味を word2vec[3]による分散表現を用いて現す。更に、意味的に近い単語間に基づいたアライメント及び語順情報を用いた評価を行う。性能評価実験では WAT（Workshop on Asian Translation）2017[4]データにおける日本文と英文の翻訳文を用い、人手評価との相関を求めた。WAT2017 データにはニューラル翻訳が出力した翻訳文が数多く含まれており、提案手法の有効性を確認するために最適である。性能評価実験の結果、提案手法はシステム単位において他の自動評価法よりも高い相関を示すことを確認した。

## 3.3.2 関連研究

機械翻訳システムを評価するための自動評価法においては、デファクトスタンダードな評価法として BLEU[5]がある。しかし、英日のような文法構造が異なる言語間を対象とした際、そこで得られる翻訳文は参照訳との間で語順が大きく変わる場合も考えられる[6]。その場合、BLEU は局所的な語順の相違には追随できるが、大局的な語順の違いを考慮するこが困難となり、十分な評価精度が得られない可能性がある。また、BLEU は単語間の表層レベルでの n-gram 一致率に基づいているため、単語の意味を反映したものとはなっていない。更に、語順を考慮した自動評価法としては IMPACT[7]や RIBES[8]がある。IMPACT は翻訳文と参照訳間の一致した単語列であるチャンクの出現順を考慮し、評価スコアに反映している。RIBES は一致単語の語順に対して順位相関を適用することにより評価スコアを算出している。しかし、これらの手法は BLEU と同様に表層レベルでの単語間の一致に基づいており、単語の意味を反映したものとはなっていない。一方、単語の意味に基づいて文間の類似度を得る手法はいくつか存在する。その中でも代表的な手法である WMD（Word Mover's Distance）[9]は EMD（Earth Mover's Distance）[10]を自然言語文に対応させた手法であり、word2vec による単語分散表現を用いて単語の意味を考慮したうえで類似度を算出している。具体的には、単語のアライメントの際に単語を単語分散表現にマッピングし、意味的に近い単語同士の対応付けを行っている。しかし、語順は考慮しておらず、翻訳文と参照訳間で語順が大きく異なったとしても類似度には反映されない。提案手法も WMD と同様に EMD に基づいた自動評価法であるが、単語間の距離計算の際に語順の情報も評価スコアに反映している。したがって、翻訳文と参照訳文との間で全て同じ単語で構成されている翻訳文であっても語順が変化した場合、評価スコアにその違いが反映される。

### 3.3.3 提案手法

#### 3.3.3.1 EMD の利用

提案手法では、EMD を用いて単語の分散表現からなる文間の類似度を求める。EMD は需要地と供給地、需要量と供給量、そして、距離からなる最適化問題の一つである輸送問題の考え方に基づいている。文間の類似度計算に適用する際には、需要地と供給地、即ち、特徴量には単語の分散表現である word2vec を用いる。需要量と供給量、即ち、重みには *tf・idf* を用いる。そして、距離計算にはコサインを用いる。しかし、EMD はもともと画像検索のために提案された手法であり、言語をタスクとした場合にはこのようなパラメータの定義だけでは不十分である。そこでより言語に対応した類似度を求めるために 2 つの観点より新たな処理を導入する。一つは単語アライメントである。単語アライメントを行い、その結果を距離行列に反映する。そのことにより対応関係にある単語のみに着目した精度の高い類似度を得ることができる。二つ目は語順情報の利用である。翻訳文と参照訳間で出現順が異なる場合、負の重みとして語順情報を距離計算に付与する。そのことにより語順が近いほど類似度が高くなり、語順を反映したものとなる。これらの 2 つの処理については **3.3.3.3** と **3.3.3.4** で詳細を述べる。

また、EMD は類似しているほど 0.0 に近くなり、値は小さくなる。しかし、それでは一般的な自動評価法とは異なる。そこで、EMD の出力値を 1.0 以下に正規化し、1.0 から引くことで類似しているほど 1.0 に近くなるように補正する。

#### 3.3.3.2 単語の重み付け

EMD のパラメータの一つである重みには *tf・idf* を用いる。その場合、*tf* は任意の単語の文中の出現頻度とする。また、*df* は任意の単語が出現する文の数とする。したがって、多くの単語の *tf* は 1 や 2 といった小さな値となる。提案手法では、*tf・idf* の導入はあくまでも内容語と機能語を区別することを目的としている。出現する多くの単語の *tf* は小さいが、機能語については数多くの文に出現するため *idf* が小さくなり、*tf・idf* の値は小さくなると考えられる。その結果、内容語と機能語が区別できると考えられる。

また、EMD の値が 0.0 から 1.0 の範囲となるように、文中の全単語の *tf・idf* の総和を 1.0 に正規化する。

#### 3.3.3.3 単語アライメント

EMD を用いる場合、特徴量、重み、そして距離を定義することで利用可能である。しかし、言語に適用する場合、言語特有の情報を利用することが有効である。特に単語アライメントを行うことは有効である。単語アライメントは WMD を始め、他の先行研究[11]でも取り入れられている。

提案手法では、翻訳文と参照訳間において全ての単語を word2vec で得られる単語ベクトルにマッピングし、全単語間に対してコサインを用いて意味的な類似度を求める。そして、翻訳文の１つ１つの単語を基準に参照訳中の単語と最もコサインの値が大きな単語との対応付けを行う。最大となるコサインの値が複数存在する場合には、一意に対応関係を決定できないとして、対応

付けは行わない。

以下の図 1 に翻訳文“重油中の有害物質が障害の原因である。”と参照訳“重油中に含まれる有害物質が障害の原因となる。”における単語アライメントの具体例を示す。

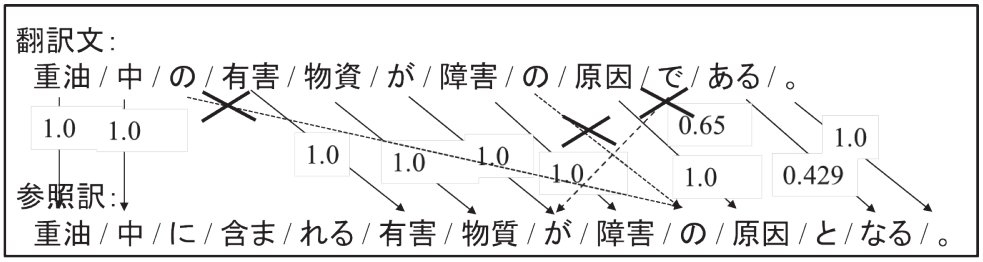



図 1　単語アライメントの具体例

図 1 の数値は単語間のコサインの値である。そして、実線は単語間の対応付けが行われたことを示している。図 1 では、翻訳文中の単語“ある”と参照訳中の“なる”は表層レベルでは一致していないが、参照訳文中の単語の中では最も高いコサインを示したため対応付けがされている。一方、単語“の”については翻訳文中に 2 回出現するため、参照訳中の単語の“の”と対応関係を一意に決定できないとして対応付けは行わない。また、翻訳文中の単語“で”については、参照訳文中の単語の中では“が”とのコサインが 0.65 と最も高い値を示したが、翻訳文中の単語“が”とのコサインが 1.0 と最も高いため、単語“が”との対応関係を優先する。その結果、翻訳文中の単語“で”は対応する単語は参照訳中には存在しないことになる。

#### 3.3.3.4 語順情報の利用に基づく距離計算

単語アライメントの結果は EMD の距離計算に利用する。その際、対応関係が得られた単語間と得られなかった単語間とで区別し、かつ、対応関係が得られた単語間においても意味的に近く、表層的にも一致した単語間と意味的に近いものだけの単語間を区別する。更に、そのようにして定義した距離に対して語順情報を重みとして付与する。以下の式(1)に距離計算の式を示す。

$$d = \begin{cases} 1.0 - cosine \times poss\_diff & \text{(意味的に近くかつ表層が一致)} \\ 1.0 - cosine^2 \times poss\_diff & \text{(意味的に近い)} \\ 1.0 & \text{(対応関係なし)} \end{cases} \quad (1)$$

*cosine* は単語分散表現間の距離であり、単語同士の意味的な類似度を示している。また、意味的には近いが、表層的には一致していない場合には、*cosine* の値を二乗することでより小さな値に変換する。そして、意味的にも表層的にも対応関係が存在しない単語間の距離は 1.0 とする。*pos_diff* は対応する単語における翻訳文中及び参照訳中のそれぞれの出現位置の相対的なずれを示しており、負の重みとして用いている。以下の式(2)に *pos_diff* の式を示す。

$$pos\_diff = 1.0 - \left| \frac{pos(w_c)}{len(c)} - \frac{pos(w_r)}{len(r)} \right| \quad (2)$$

式(2)の $pos(w_c)$は翻訳文における単語の出現位置である。先頭の単語を 1 として何番目に位置するかを示している。$pos(w_r)$は参照訳における単語の出現位置である。そして、これらを翻訳文の単語数 *len(c)*と参照訳の単語数 *len(r)*でそれぞれ正規化することで最大値を 1.0 にする。更に、$|pos(w_c)/len(c)-pos(w_r)/len(r)|$は相対位置のずれが大きいほど大きな値となるため、1.0 から引くことで、*poss_diff* は相対位置のずれが大きいほど小さくなるようにする。そうすることにより、*pos_diff* は負の重みとなる。

これらの式(1)と式(2)を用いて、翻訳文と参照訳間のすべての単語間の距離計算を行い、距離行列を生成する。図 2 に図 1 で用いた翻訳文“重油中の有害物質が障害の原因である。”と参照訳“重油中に含まれる有害物質が障害の原因となる。”における距離行列の例を示す。

| | 重油 | 中 | に | 含ま | れる | 有害 | 物質 | が | 障害 | の | 原因 | と | なる | 。 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 重油 | 0.012 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 中 | 1.0 | 0.024 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| の | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 有害 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 0.095 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 物質 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 0.083 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| が | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 0.071 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 障害 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 0.060 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| の | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 原因 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 0.036 | 1.0 | 1.0 | 1.0 |
| で | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| ある | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 0.788 | 1.0 |
| 。 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 0.0 |

図 2　距離行列の具体例

距離行列の要素は小さいほど単語間の類似度が高いことを示している。翻訳文中の単語“なる”と参照訳中の単語“ある”においては表層レベルでは一致していないため、距離としては最も 1.0 に近い 0.788 となっている。また、“。”以外の単語間が表層レベルで一致しているのにもかかわらず 0.0 となっていないのは語順情報である *pos_diff* の値が 1.0 未満となるためである。

### 3.3.4 性能評価実験

#### 3.3.4.1 実験データ

本報告では提案手法の有効性を検証するために WAT2017 データの中で翻訳文が英語及び日本語のものを用いた。データのカテゴリーとしては ASPEC、JPC、IITB、JIJI、RECIPE の 5 種類である。人手評価は全翻訳文に対して 2 名が adequacy の観点で 5 段階評価を行ったものが付与されている。5 種類のデータにおける言語間の内訳を以下の表 1 に示す。表中の“〇”は今回の性能評価実験で使用したデータである。“〇”が記入されていないデータは翻訳文が英語でも日本語でもないため評価の対象としなかった。いずれのデータも 200 文の翻訳文に対して参照訳と人手評価が付与されている。表中の“ヒ”はヒンディー語を示している。

表 1　WAT2017 データの内訳

| ASPEC | | | | JPC | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 日英 | 英日 | 日中 | 中日 | 日英 | 英日 | 日中 | 中日 | 日韓 | 韓日 |
| ○ | ○ | | ○ | ○ | ○ | | ○ | | ○ |

| IITB | | JIJI | | RECIPE | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | TTL | | ING | | STE | |
| ヒ英 | 英ヒ | 日英 | 英日 | 日英 | 英日 | 日英 | 英日 | 日英 | 英日 |
| ○ | | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |

#### 3.3.4.2 実験方法

提案手法の評価はシステムが算出したスコアと人手評価のスコアの間の Pearson と Kendall の相関係数を求め、それぞれシステム単位と文単位の評価結果とした。また、比較のため、他の自動評価として IMPACT、RIBES、WMD を用いた。全ての自動評価法を使用する際には、英文に対しては tokenizer.perl と lowercase.perl[12]による前処理を行った。また、日本文に対しては MeCab[13]を用いて分かち書きを行った。そして、提案手法と WMD においては単語分散表現を得るための word2vec のモデルとして、英語については GoogleNews-vectors-negative300.bin（300 次元、語彙数：3,000,000）を、日本語については日本語エンティティの entity_vector.model.bin（200 次元、語彙数：1,015,474）をそれぞれ用いた。GoogleNews-vectors-negative300.bin は新聞記事を学習データとしており、ストップワードは含まない。entity_vector.model.bin は日本語 Wikipedia を学習データとしており、ストップワードも含んでいる。

#### 3.3.4.3　実験結果

表 2 に 4 つの自動評価法による評価と人手評価との Pearson の相関係数を示す。また、表 3 には 4 つの自動評価法による評価と人手評価との Kendall の相関係数を示す。そして、表 4 にはそれぞれのデータの種類ごとの相関係数を示す。例えば、ASPEC であれば JE、EJ、CJ の相関係数の平均を意味する。表中の太字及び下線の数値は各データにおいて 4 つの自動評価法の中で最も高い相関係数であることを示している。表 2 と表 3 の各データにおいて翻訳文を得るために使用された MT システムは次の通りである。ASPEC-JE は AIAYN、Kyoto-U、NTT の 3 つ、ASPEC-EJ は AIAYN、Kyoto-U、NAIST-NICT、NICT-2、NTT の 5 つ、ASPEC-CJ は AIAYN、Kyoto-U、NICT-2 の 3 つ、JPC-JE は CUNI、JAPIO、u-tkb の 3 つ、JPC-EJ は EHR、JAPIO、u-tkb の 3 つ、JPC-CJ は EHR、JAPIO、u-tkb の 3 つ、JPC-KJ は EHR、JAPIO の 2 つ、IITB-HE は XMUNLP、IITB-MTG の 2 つ、JIJI-JE は AIAYN、NTT、XMUNLP の 3 つ、JIJI-EJ は AIAYN、NTT、XMUNLP の 3 つ、RECIPE については全て AIAYN、XMUNLP の 2 つである。

表 2 自動評価法による評価と人手評価との Pearson の相関係数

| | 提案手法 | IMPACT | RIBES | WMD |
|---|---|---|---|---|
| ASPEC-JE | -0.627 | 0.725 | **0.787** | 0.710 |
| ASPEC-EJ | **0.354** | 0.042 | 0.178 | 0.077 |
| ASPEC-CJ | -0.818 | -0.739 | **-0.683** | -0.760 |
| JPC-JE | **0.997** | 0.991 | 0.985 | 0.988 |
| JPC-EJ | **0.782** | 0.547 | 0.595 | 0.507 |
| JPC-CJ | **0.949** | 0.606 | 0.651 | 0.615 |
| JPC-KJ | -1.000 | -1.000 | -1.000 | -1.000 |
| IITB-HE | 1.000 | 1.000 | 1.000 | 1.000 |
| JIJI-JE | -1.000 | **-0.996** | -1.000 | -1.000 |
| JIJI-EJ | **-0.962** | -1.000 | -0.999 | -1.000 |
| RECIPE-TTL-JE | 1.000 | 1.000 | 1.000 | 1.000 |
| RECIPE-TTL-EJ | **1.000** | -1.000 | -1.000 | -1.000 |
| RECIPE-ING-JE | 1.000 | 1.000 | 1.000 | 1.000 |
| RECIPE-ING-EJ | 1.000 | 1.000 | 1.000 | 1.000 |
| RECIPE-STE-EJ | 1.000 | 1.000 | 1.000 | 1.000 |
| RECIPE-STE-JE | 1.000 | 1.000 | 1.000 | 1.000 |

表 3 自動評価法による評価と人手評価との Kendall の相関係数

| | 提案手法 | IMPACT | RIBES | WMD |
|---|---|---|---|---|
| ASPEC-JE | 0.304 | **0.342** | 0.271 | 0.277 |
| ASPEC-EJ | 0.301 | **0.374** | 0.354 | 0.327 |
| ASPEC-CJ | 0.171 | **0.202** | 0.166 | 0.151 |
| JPC-JE | 0.255 | **0.280** | 0.271 | 0.252 |
| JPC-EJ | 0.195 | **0.232** | 0.213 | 0.200 |
| JPC-CJ | 0.331 | **0.439** | 0.436 | 0.340 |
| JPC-KJ | 0.115 | 0.180 | **0.182** | 0.154 |
| IITB-HE | **0.428** | 0.391 | 0.270 | 0.405 |
| JIJI-JE | **0.219** | 0.085 | 0.086 | 0.176 |
| JIJI-EJ | **0.220** | 0.073 | 0.095 | 0.085 |
| RECIPE-TTL-JE | 0.470 | **0.490** | 0.452 | 0.447 |
| RECIPE-TTL-EJ | **0.274** | 0.271 | 0.259 | 0.212 |
| RECIPE-ING-JE | 0.469 | **0.520** | 0.497 | 0.106 |
| RECIPE-ING-EJ | **0.370** | 0.340 | 0.337 | 0.026 |
| RECIPE-STE-EJ | 0.379 | **0.468** | 0.411 | 0.438 |
| RECIPE-STE-JE | 0.392 | **0.407** | 0.378 | 0.303 |

表 4 データの種類毎の相関係数

| | 提案手法 | | IMPACT | | RIBES | | WMD | |
|---|---|---|---|---|---|---|---|---|
| | Pearson | Kendall | Pearson | Kendall | Pearson | Kendall | Pearson | Kendall |
| ASPEC | -0.364 | 0.259 | 0.009 | 0.306 | 0.094 | 0.264 | 0.009 | 0.252 |
| JPC | 0.432 | 0.224 | 0.286 | 0.283 | 0.308 | 0.276 | 0.278 | 0.237 |
| IITB | 1.000 | 0.428 | 1.000 | 0.391 | 1.000 | 0.270 | 1.000 | 0.405 |
| JIJI | -0.981 | 0.220 | -0.998 | 0.079 | -1.000 | 0.091 | -1.000 | 0.131 |
| RECIPE | 1.000 | 0.392 | 0.667 | 0.416 | 0.667 | 0.389 | 0.667 | 0.255 |
| Avg. | **0.355** | 0.306 | 0.261 | **0.318** | 0.282 | 0.292 | 0.259 | 0.244 |

#### 3.3.4.4 考察

表 2 よりシステム単位の評価精度である Pearson の相関係数においては提案手法が最も高いことがわかる。特に JPC と RECIPE データにおいて提案手法の相関が高かった。RECIPE-TTL-EJ では提案手法のみが相関係数 1.0 を示し、他の自動評価法では-1.0 であった。実際のデータを参照すると、人手評価は AYAIN が 4.2、XMUNLP が 4.075 であるのに対して、提案手法は AYAIN が 0.339、XMUNLP が 0.332 と共に AYAIN の方が高いスコアとなっている。それに対して、IMPACT では AYAIN が 0.365、XMUNLP が 0.402 と XMUNLP の方が高い評価スコアとなっている。全データの相関係数の平均を求めると、表 4 に示すように提案手法のみが 0.3 を超えて 0.355 となり、他の自動評価法よりも高い相関係数を示した。

しかし、表 3 より文単位の評価精度である Kendall の相関係数においては IMPACT に比べて低い相関係数であった。IITB と JIJI データにおいては提案手法の方が IMPACT の相関係数を上回っているが、ASPEC と JPC データにおいては全て IMAPCT の方が高い相関係数を示している。表 4 の相関係数の平均を見ても、IMPACT の 0.318 に対して、提案手法は 0.306 であった。しかし、RIBES や同じ単語分散表現に基づく WMD との比較においては提案手法は高い相関係数を示した。表 3 と表 4 の結果より、提案手法においては文単位での評価精度向上が今後の大きな課題となる。

### 3.3.5 まとめ

本報告では、単語の意味を考慮した自動評価法を提案し、性能評価実験に基づきその有効性を検証した。提案手法は EMD を用いて単語の分散表現からなる文同士の類似度を求める。更に、EMD を用いる際に言語タスクに対応させるために単語アライメントを行い、その結果に対して語順情報も用いて単語間の距離を算出する。性能評価実験の結果、提案手法はシステム単位において人手評価との相関が他の自動評価法に比べ最も高いものとなった。しかし、文単位においては IMPACT よりも低い相関となった。

今後は文単位においても高い相関が得られるように改良を行う予定である。その方法としては、他の自動評価法と組み合わせることなどが考えられる。

**謝辞**

本研究を行うにあたり、WAT2017 データをご提供いただいた科学技術振興機構、情報通信研究機構、そして、京都大学に感謝致します。

**参考文献**

[1] Ilya Sutskever, Oriol Vinyals, and Quoc V.L.E (2014) “Sequence to Sequence Learning with Neural Networks,” Advances in Neural Information Processing Systems, pp. 3104-3112.

[2] Minh-Thang Luong, Hieu Pham, and Christopher D. Manning (2015) “Effective Approaches to Attention-based Neural Machine Translation,” arXiv preprint arXiv:1508.04025.

[3] Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean (2013) “Efficient Estimation of Word Representations in Vector Space,” Proceedings of Workshop at International Conference on Learning Representations 2013.

[4] Toshiaki Nakazawa, Shohei Higashiyama, Chenchen Ding, Hideya Mino, Isao Goto, Hideto Kazawa, Yusuke Oda, Graham Neubig and Sadao Kurohashi (2017) “Overview of the 4th Workshop on Asian Translation,” Proceedings of the 4th Workshop on Asian Translation, pp.1-54.

[5] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu (2002) “BLEU: a Method for Automatic Evaluation of Machine Translation,” Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL 2002), pp. 311-318.

[6] Hideki Isozaki, Katsutoshi Sudoh, Hajime Tsukada and Kevin Duh (2010) “Head Finalization: A Simple Reordering Rule for SOV Languages”, Proceedings of the Joint 5th Workshop on Statistical Machine Translation and MetricsMATR, pp244-251.

[7] Hiroshi Echizen-ya, and Kenji Araki (2007) “Automatic Evaluation of Machine Translation based on Recursive Acquisition of an Intuitive Common Parts Continuum,” Proceedings of the Eleventh Machine Translation Summit, pp.151-158.

[8] Hideki Isozaki, Tsutomu Hirao, Kevin Duh, Katsuhito Sudoh, and Hajime Tsukada (2010) “Automatic Evaluation of Translation Quality for Distant Language Pairs,” Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pp. 944–952.

[9] Matt J. Kusner, Yu Sun, Nicholas I. Kolkin, and Kilian Q. Weinberger (2015) “From Word Embeddings To Document Distances,” Proceedings of the 32nd International Conference on Machine Learning, pp.957-966.

[10] Yossi Rubner, Carlo Tomasi, and Leonidas J. Guibas (2000) “The Earth Mover's Distance as a Metric for Image Retrieval,” International Journal of Computer Vision, 40(2), pp.99-121.

[11] 松尾潤樹，小町守，須藤克仁 (2016) “単語分散表現を用いた単語アライメントによる日英機械翻訳の自動評価尺度,” 情報処理学会第 227 回自然言語処理研究会, Vol.2016-NL-229 No.20, pp.1-7.

[12] “Welcome to Moses!,”, http://www.statmt.org/moses/
[13] “MeCab: Yet Another Part-of-Speech and Morphological Analyzer,”
http://taku910.github.io/mecab/

# 3.4　中日テストセットを用いた特許文献の翻訳評価

## －中国語分離パターンの拡充および評価の実施－

元・山梨英和大学　　江原 暉将
(株)富士通研究所　　長瀬 友樹
(株)ディープランゲージ　王　向莉

### 3.4.1　はじめに

機械翻訳評価の一手法として、表現パターン別に評価用例文を用意しておき、翻訳結果に対して対応する表現パターンがうまく訳されていることをピンポイントでチェックする「テストセット評価」が提案されている[1)2)3)]。

筆者らは、中国語特許文献の中日機械翻訳評価のためにテストセットの構築を行い、昨年度までに以下のことを実施した[4)5)6)7)]。

・中日特許文平行コーパスの収集
・テストセットの作成
・評価用サイトの整備[1]

昨年度までのテストセットの作成において、1064 個の中国語表現パターンとそれを含む中国語特許文の収集および中国語表現パターンに対する日本語翻訳パターン設問の作成を行った。

### 3.4.2　中国語表現パターンの追加収集

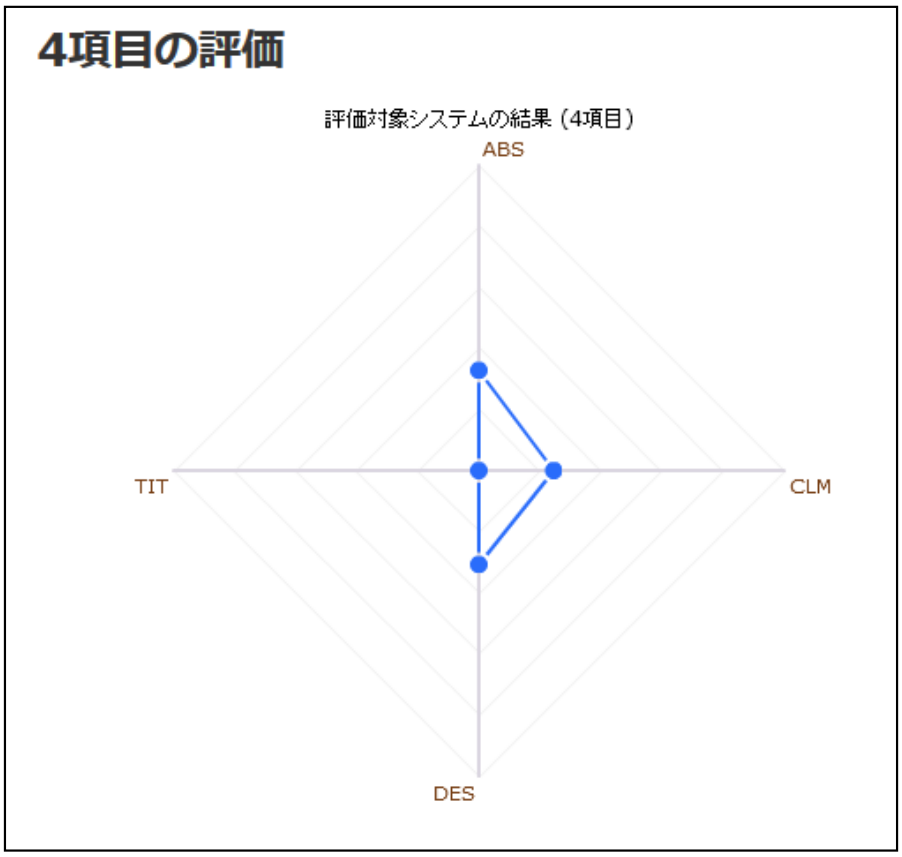
100 万文対からなる中日特許文平行コーパス[2]から中国語部分を抜き出し、昨年度から中国語分離表現パターンを抽出を行っている。今年度は追加収集として 69 個の中国語分離表現パターンを収集した。さらに不足していたタイトル部分のパターンを 18 パターン収集した。その結果、昨年度までの収集と合わせてで 1151 個の中国語表現パターンとそれを含む中国語特許文を収集し、日本語翻訳パターン設問を作成した。

### 3.4.3　AAMT 自動評価サイトでの試験

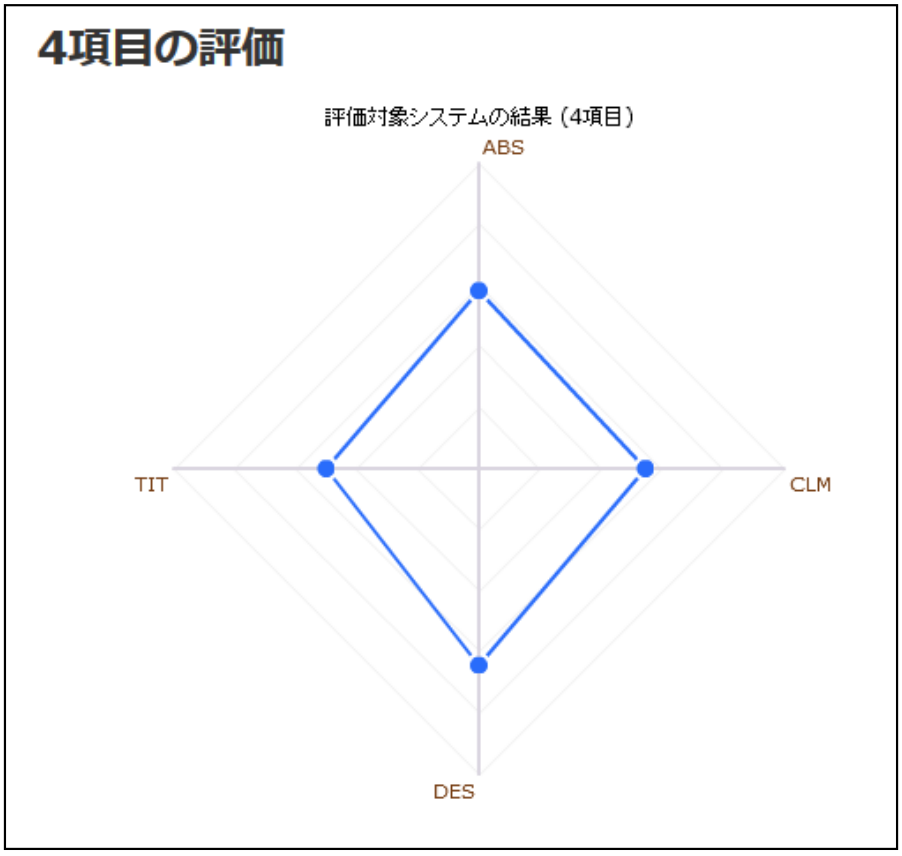
昨年度構築した AAMT 自動評価サイトを用いて、いくつかの翻訳システムについて評価を行った[3]。結果を図 1 に示す。RBMT、SMT、オンラインサイト(2016 年版および 2018 年版)、NMT での翻訳である。評価結果は、「発明の名称(TIT)」、「要約(ABS)」、「請求範囲(CLM)」、「詳細説明(DES)」の 4 種類の出典別に 0～100%の範囲で設問への正解率が表示される。表示された四辺形の面積が大きいほうが評価値が高い。評価結果は、評価値の高い順に NMT、オンラインサイト(2018 年版)、SMT、オンラインサイト(2016 年版)、RBMT の順である。

---

[1] 本部分は、AAMT 課題調査委員会で整備したサイトを利用させてもらっている。
[2] 本コーパスは WAT2016 の JPCzh-ja task において開示されたコーパスを利用している[8)]。
[3] テストセットは 2015 年度までに作成したものを用いている。発明の名称(TIT)が 4、要約(ABS)が 67、請求範囲(CLM)が 53、詳細説明(DES)が 720 の合計 844 設問である。

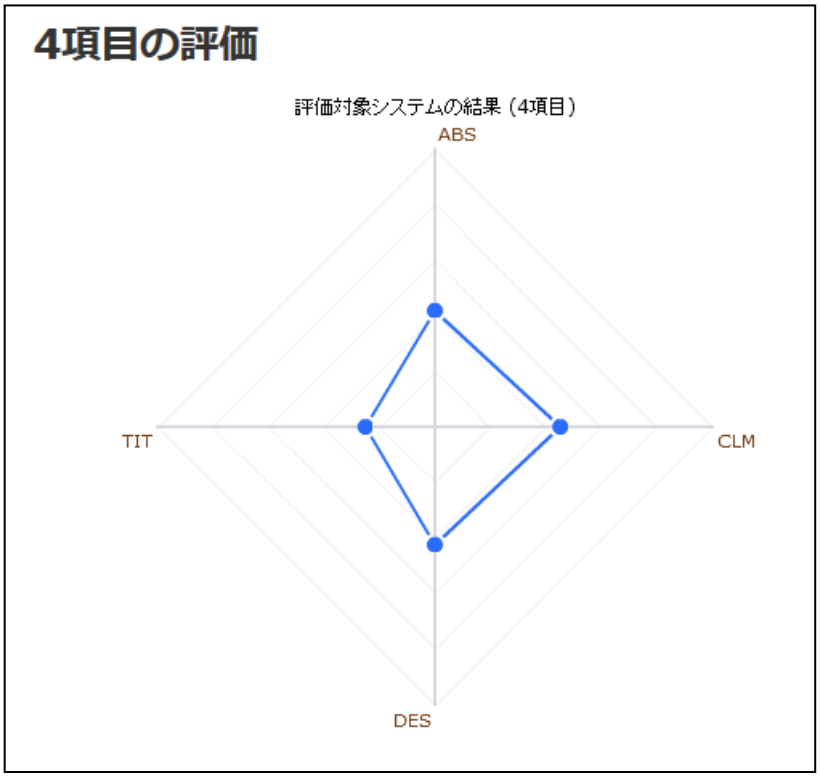



(a) RBMT[4]

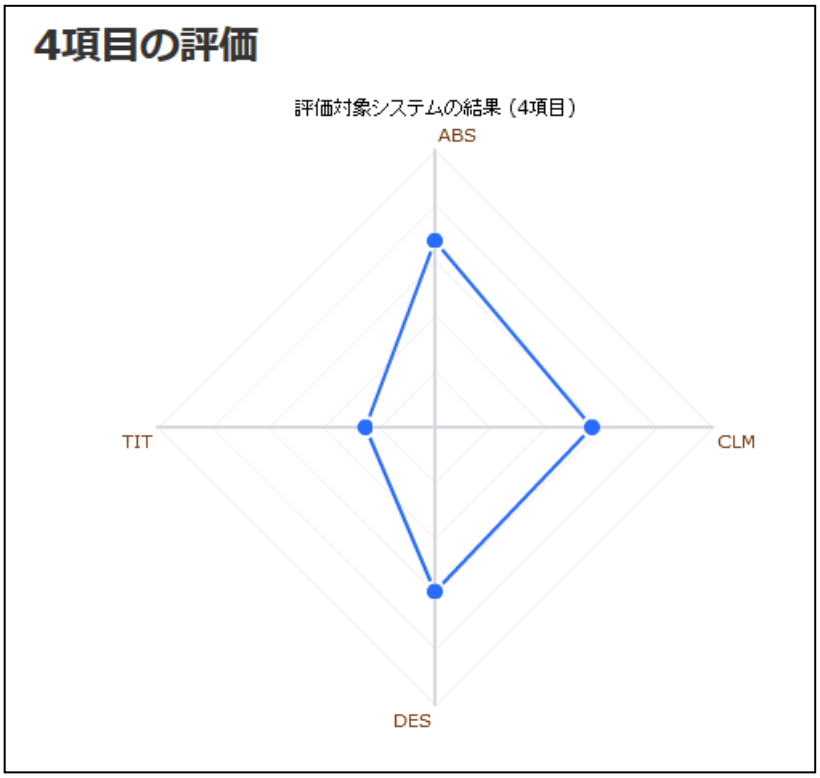



(b) SMT[5]

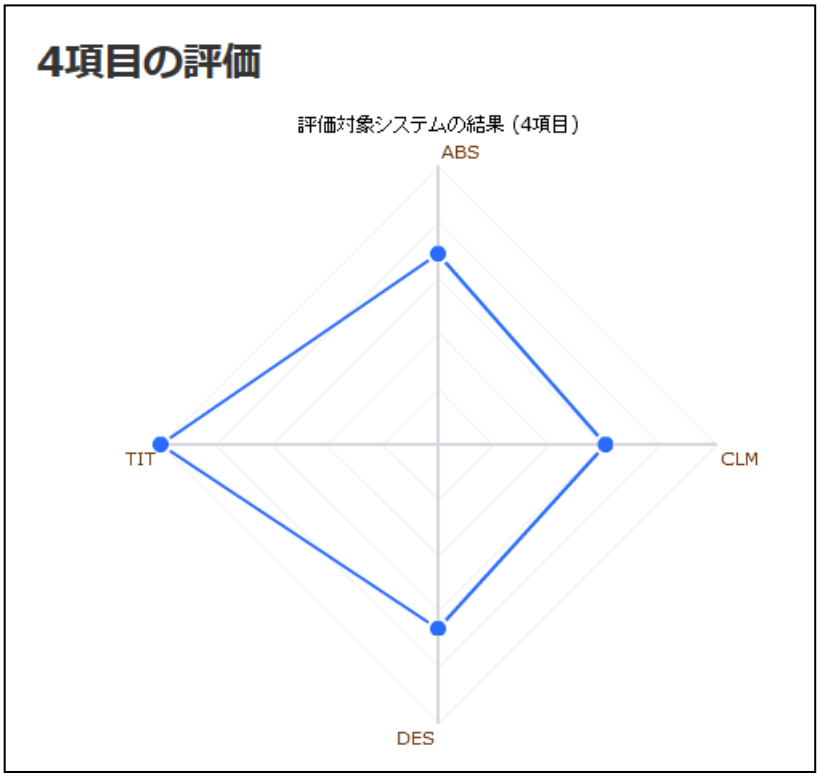
[4] 市販の RBMT システムである。
[5] 文献 9)に示す SMT システム(2)である。




(c) online (2016 年版)




(d) online (2018 年版)




(e) NMT[6]

図 1 テストセットを用いた評価例

### 3.4.4 まとめと今後の課題

昨年度までのデータ作成と今年度の作成を合わせて 1116 の設問設定ができた。今後の課題としては以下のことがあげられる。

・日本語翻訳パターンのバラエティが不足している部分があり、より適切な設問とすることが必要である。
・数式や化学式、数量表現など特許に特有な表現パターンが不足している。
・自動評価や人手評価とテストセット評価との比較を行い、双方のメリット・デメリットを明らかにする。

今後、これらの課題を解決して、より良い中日特許文テストセットとしていきたい。

### 参考文献

1) Isahara, H. 1995. JEIDA's Test-Sets for Quality Evaluation of MT Systems –Technical Evaluation from the Developer's Point of View–. *Proc. of MT Summit V.*
2) Uchimoto, K., K. Kotani, Y. Zhang and H. Isahara. 2007. Automatic Evaluation of Machine Translation Based on Rate of Accomplishment of Sub-goals. *Proc. of NAACL HLT*, pages 33-40.

[6] 文献 10)に示す文字ベースの NMT システムである。リランキングは行っていない。

3) Nagase, T., H. Tsukada, K. Kotani, N. Hatanaka and Y. Sakamoto. 2011. Automatic Error Analysis Based on Grammatical Questions. *Proc. of PACLIC*.
4) 長瀬友樹, 江原暉将, 王向莉. 2014. 中日特許文評価用テストセットの作成, *平成 25 年度 AAMT/Japio 特許翻訳研究会報告書*, pages 78-82.
5) 長瀬友樹, 江原暉将, 王向莉. 2015. 中国語特許文献の中日翻訳評価のためのテストセットの改良と評価サイトの作成, *平成 26 年度 AAMT/Japio 特許翻訳研究会報告書,* pages 104-109.
6) 江原暉将, 長瀬友樹, 王向莉. 2016. 中国語特許文献の中日翻訳評価のためのテストセットの拡充, *平成 27 年度 AAMT/Japio 特許翻訳研究会報告書,* pages 40-42.
7) 江原暉将, 長瀬友樹, 王向莉. 2017. 中日テストセットを用いた特許文献の翻訳評価―中国語分離パターンの利用―, *平成 28 年度 AAMT/Japio 特許翻訳研究会報告書,* pages 62-66.
8) Toshiaki Nakazawa, Hideya Mino, Chenchen Ding, Isao Goto, Graham Neubig and Sadao Kurohashi. 2016. Overview of the 3rd Workshop on Asian Translation. *Proceedings of the 3rd Workshop on Asian Translation (WAT2016)*, pages 1-46.
9) Terumasa Ehara. 2016. Translation systems and experimental results of the EHR group for WAT2016 tasks, *Proceedings of the 3rd Workshop on Asian Translation (WAT2016)*, pages 111-118.
10) Terumasa Ehara. 2017. SMT reranked NMT, *Proceedings of the 4th Workshop on Asian Translation (WAT2017)*, pages 119-126.

# 3.5 WAT2017 人手評価結果の分析

科学技術振興機構　中澤　敏明
NHK 放送技術研究所　後藤　功雄
東芝デジタルソリューションズ株式会社　園尾　　聡

## 3.5.1　はじめに

今年度の拡大評価部会人手評価グループでは、WAT2017[1]で行われた JPO Adequacy Evaluation（特許庁が提案している「特許文献機械翻訳の品質評価手順」のうち「内容の伝達レベルの評価」[2]）の結果の分析を行った。JPO Adequacy Evaluation は、テストセットのうちの200 文を対象に、2 名の評価者が以下の基準での絶対評価を行う。

**表 1：JPO Adequacy Evaluation の評価基準**

| 評価 | 基準 |
|---|---|
| 5 | すべての重要情報が正確に伝達されている。(100%) |
| 4 | ほとんどの重要情報は正確に伝達されている。(80%～) |
| 3 | 半分以上の重要情報は正確に伝達されている。(50%～) |
| 2 | いくつかの重要情報は正確に伝達されている。(20%～) |
| 1 | 文意がわからない、もしくは正確に伝達されている重要情報はほとんどない。(～20%) |

WAT で行われた全ての翻訳タスクについてこの基準での評価が行われたが、今回は特許の日英・英日翻訳結果についての分析を行ったので、報告する。

## 3.5.2　成績トップのシステムの κ 係数が小さい理由

英日翻訳の評価結果上位 3 システムは順に SYSTEM A（平均 4.75）、SYSTEM B（4.63）、SYSTEM C（4.40）であったが、評価者間の一致度を示すκ係数（Cohen's Kappa）は順に 0.32、0.42、0.43 であり、トップの SYSTEM A に対するκ係数が最も小さい値となった（WAT の方針に習って、システム名を匿名化している）。これは日英翻訳においても同様で、トップのシステムのκ係数が最も小さくなった。この理由を、英日翻訳の SYSTEM A と SYSTEM B を例にとり説明する。

表 2 に SYSTEM A の評価結果の詳細を、表 3 に SYSTEM B の評価結果の詳細を示す。

**表 2：SYSTEM A の評価結果詳細**

| 評価 | 5 | 4 | 3 | 2 | 1 | 計 |
|---|---|---|---|---|---|---|
| 5 | 154 | 11 | 2 | 1 | 0 | 168 |
| 4 | 11 | 3 | 3 | 0 | 0 | 17 |
| 3 | 4 | 3 | 4 | 0 | 0 | 11 |
| 2 | 0 | 0 | 2 | 1 | 0 | 3 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 |
| 計 | 169 | 17 | 11 | 3 | 0 | 200 |

表 3：SYSTEM B の評価結果詳細

| 評価 | 5 | 4 | 3 | 2 | 1 | 計 |
|---|---|---|---|---|---|---|
| 5 | 141 | 15 | 1 | 0 | 0 | 157 |
| 4 | 8 | 7 | 6 | 1 | 0 | 22 |
| 3 | 4 | 2 | 6 | 2 | 0 | 14 |
| 2 | 0 | 0 | 4 | 1 | 1 | 6 |
| 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| 計 | 153 | 24 | 17 | 4 | 2 | 200 |

この表から純粋に二人の評価者の評価が一致しているものの割合$p_o$を計算すると、SYSTEM A では対角線上の数字を足した 162(=154+3+4+1+0)を 200 で割って 0.81 であり、SYSTEM B では 0.78 となるため、SYSTEM A の方が一致していることになる。一方で κ 係数は二人の評価者の評価が**偶然に一致する可能性**も考慮している。偶然に一致する可能性$p_e$は、評価者 X が評価 N をつけた割合を$p_{XN}$とすると、以下の式で計算される（2 名の評価者を A と B とする）。

$$p_e = \sum_{N=1}^{5} p_{AN} * p_{BN}$$

この式に則って SYSTEM A と SYSTEM B の$p_e$を計算すると、それぞれ 0.72 と 0.62 となり、SYSTEM A の方が偶然に一致する可能性が高いことになる。κ 係数は上記$p_o$と$p_e$を使って、以下のように計算される。

$$\kappa = \frac{p_o - p_e}{1 - p_e}$$

ざっくり言うと、偶然に一致する可能性を差し引いて一致率を計算していることになり、偶然に一致する可能性が高い場合ほど、κ 係数は小さく見積もられることになる。上記の場合、2 つのシステムにおいて$p_o$の値はさほど変わらないが、$p_e$の値に大きな違いがあるため、SYSTEM A の方が κ 係数が小さくなっているのである。

SYSTEM A の$p_e$の値が大きくなる要因の一つは、SYSTEM A の翻訳結果が良く、どちらの評価者も 5 をつける割合が高くなっているからだと考えられる。このように、非常に良いシステム（もしくは非常に悪いシステム）においては、κ 係数が不当に低く見積もられる可能性があることに注意が必要である。一般的に、評価結果がなんらかの値に偏る傾向が強い場合には、κ 係数の解釈には注意が必要であると考えられる。つまり κ 係数が低いからと言って、評価結果がばらついていると短絡的に考えることは危険なのである。

### 3.5.3　2 名の評価者間で評価が割れている文の検討

次に日英・英日の特許翻訳結果において、2 名の評価者間で評価が 2 以上離れている文について、どのような傾向があるのかを検討した。

#### 3.5.3.1　評価者のミス

単純に評価者の評価ミスと思われるものがいくつか見受けられた。以下に例を示す（各翻訳文

の後ろにある１組の数字が評価値である)。

| | |
|---|---|
| 入力 | Accordingly, the present invention relates to a rod-like crystal of CZTS. |
| 正解 | それ故、本発明は、CZTS のロッド状結晶体に関する。 |
| 出力 | したがって、本発明は、<unk>の棒状結晶に関する。(3, 5) |

この例では NMT 特有の<unk>がそのまま出力されているにも関わらず、一方の評価者が評価 5 としている。

| | |
|---|---|
| 入力 | However no long text and even only pictures may be included in pages for many book images. |
| 正解 | しかし、多くの本画像のページには、長いテキストは含まれず、ピクチャしか含まれないことがある。 |
| 出力 | しかし、多くの書籍画像のページには、長いテキストや絵も含まれなくてもよい。(2, 5) |

この例は入力の英語文の解釈が難しいが、正解と比較すると出力は明らかに誤っている。しかしながら一方の評価者が評価 5 としている。

| | |
|---|---|
| 入力 | A plug 72 provided on the rear end of the ink pack 7 faces the plug opening 335 . |
| 正解 | 口栓用開口部３３５には、インクパック７の後端に設けられた口栓７２が臨んでいる。 |
| 出力 | プラグ開口 335 には、インクパック 7 の後端に設けられたプラグ 72 が設けられている。(2, 5) |

出力では“faces”が正しく訳出されていないため誤訳であるが、一方の評価者は評価 5 としている。

| | |
|---|---|
| 入力 | 第１熱媒体経路２３２はその内部に熱媒体としての水を通流させる。 |
| 正解 | Water as the heat medium flows through the inside of the first heat medium passage 232 . |
| 出力 | The first thermal medium path 232 flows through the water as a heat medium .(1, 4) |

この例では入力と正解とで主語が変化しており、それゆえ入力が使役文であるのに対し、正解は能動態の文になっている。一方の出力は主語は入力と一致しているが、能動態で訳してしまっているため翻訳としては誤りとなっている。

| | |
|---|---|
| 入力 | 同軸芯線５２ａは、導線で構成される。 |
| 正解 | The coaxial core 52 a is constituted by a conducting wire. |
| 出力 | The coaxial core 52 a is constituted by a conductor.(2, 5) |

conductor だけでも導線という意味があるため、出力も正しいと考えられるが、評価者の一方が評価 2 としている。

#### 3.5.3.2 「重要情報」の定義の違いによる評価の差

特許庁の基準では、「重要情報」がどれだけ正確に伝達されているかが評価ポイントになるが、「重要情報」の定義は評価者の主観に委ねられている。以下に示す例ではこの定義の評価者間の

ズレが影響している可能性がある。

| | |
|---|---|
| 入力 | For example, FIG. 1A shows a plan view of a videoconference room with a typical arrangement. |
| 正解 | 例えば、図 1A は、典型的な構成からなるビデオ会議室の平面図である。 |
| 出力 1 | 例えば、図１Ａは、典型的な構成を有する会議室の平面図を示す。(3, 5) |
| 出力 2 | 例えば、図 1(a)は、一般的な配置の会議室の平面図を示す。(3, 5) |
| 出力 3 | 例えば、図 1a は、典型的な構成を有する会議室の平面図を示す。(3, 5) |

この例ではすべてのシステムにおいて「ビデオ」という部分が抜けているが、一方の評価者は評価 5 としている。これは「ビデオ」の部分がさほど重要ではないと判断し、ほかの重要情報が全て含まれているため評価 5 としている可能性もある。

| | |
|---|---|
| 入力 | In this example, assume that jobs have been deleted and A′ =12 is obtained. |
| 正解 | 今回の例ではジョブが削除されA'＝１２となったとする。 |
| 出力 1 | この例では、ジョブが削除され、A'=12 であると仮定する。(3, 5) |
| 出力 2 | この例では、ジョブが削除され、a=12 が得られると仮定する。(3, 5) |

この例では情報の不足はないように見えるが、一方の評価者は「仮定する」と訳していることが誤りと判断している可能性がある。

| | |
|---|---|
| 入力 | The recording medium 212 is a memory card freely removable from, for example, the camera body 200. |
| 正解 | 記録媒体２１２は、例えばカメラ本体２００に着脱自在になされたメモリカードである。 |
| 出力 | 記録媒体 212 は、例えばカメラ本体 200 から着脱自在に着脱自在のメモリカードである。(3, 5) |

この例では出力に重複が見られるが、評価者の一方が評価 5 としている。確かに重複があっても重要情報が正しく伝達されていると考えられなくもない。特許庁基準では過剰に出力された情報をどう扱うかの基準が設けられていないため、このような評価割れが起こる可能性がある。

| | |
|---|---|
| 入力 | 第１熱媒体経路２３２はその内部に熱媒体としての水を通流させる。 |
| 正解 | Water as the heat medium flows through the inside of the first heat medium passage 232 . |
| 出力 | The first thermal medium path 232 flows through the water as a heat medium .(1, 4) |

この例では入力と正解とで主語が変化しており、それゆえ入力が使役文であるのに対し、正解は能動態の文になっている。一方の出力は主語は入力と一致しているが、能動態で訳してしまっているため、文全体の意味としては誤りとなっている。ここで、文全体の意味が誤っているから評価 1 とするか、個々の名詞は正しく訳出されているから評価 4 とするかで差が生まれていると考えられる。次に示す例も同様である。

| | |
|---|---|
| 入力 | また、その固体撮像装置を用いた電子機器を提供することを目的とする。 |
| 正解 | Also, it is desirable to provide an electronic system using the solid-state imaging |

| | |
|---|---|
| | device. |
| 出力 | It is also an object of providing an electronic device using the solid state imaging device.(1, 4) |

この例でも述部の訳が少しズレているが、必要な部品は全て正しく訳出されている。このような差は、例えば否定の有無といった場合にも起こりえる。

#### 3.5.3.3 訳語の不統一

専門用語などが入力文中で複数回出現した際にその訳が統一されていない場合や、より適切な訳がある場合などに、評価が割れる傾向があるようだ。

| | |
|---|---|
| 入力 | For information, new items are not limited to added items but include changed items. |
| 正解 | なお、新規項目は、追加された項目に限らず、変更された項目を含む。 |
| 出力 1 | 情報のために、新しいアイテムは、追加項目に限定されず、変更項目を含む。(3, 5) |
| 出力 2 | 情報については、新しいアイテムは、追加項目に限定されず、変更されたアイテムを含む。(3, 5) |

“item”という単語の訳が「アイテム」であったり「項目」であったりと訳が揺れている。

| | |
|---|---|
| 入力 | In a particular embodiment, the photo sensor is a photo-multiplier tube. |
| 正解 | 具体的な一実施形態ではそのフォトセンサは光電子増倍管である。 |
| 出力 1 | 特定の実施形態では、光センサは光マルチプライヤー管である。(3, 5) |
| 出力 2 | 特定の実施形態では、フォトセンサーは photo-multiplier チューブである。(1, 4) |

“photo”が「フォト」や「光」と訳されていたり、photo-multiplier tube が様々に訳されている。決まった訳語がなさそうな場合には、一般的には専門用語全体で統一して和名もしくは洋名（カタカナ）で訳すべきであり、和名と洋名が混ざって訳されることは好ましくないと考えられる。

| | |
|---|---|
| 入力 | FIG. 15 is a representation of one unit of a carboxymethyl cellulose molecule. |
| 正解 | 図１５は、カルボキシメチルセルロース分子の１つの単位の模式図である。 |
| 出力 | ｃａｒｂｏｘｙｍｅｔｈｙｌセルロース分子の一単位の表示である。 (4, 2) |

この例では英単語がそのまま日本語文に訳出されている。頭字語など英単語をそのまま訳出すれば良い場合ももちろん存在するが、日本語訳が存在する場合には日本語にするべきであると考えられる。しかしながらこの例のように、英単語のまま訳出されても意味はわかると考えることもでき、基準を決めないと評価が割れる要因になる。

| | |
|---|---|
| 入力 | 糖タンパク質Dは、いくつかの宿主受容体のうちの１つを認識し、結合し得る。 |
| 正解 | Glycoprotein D recognizes and can bind to one of several host receptors. |
| 出力 | Sugar proteins D may recognize and couple one of several host receptors. (3, 5) |

この例では「糖タンパク質」が”Sugar proteins”と訳出されているが、これは誤訳で”Glycoprotein”が正しい訳である。この例からは、専門用語の正しい訳を知っていないと正しく評価できないということと、現在の翻訳システムが専門用語を構成的に翻訳してしまうことの弊害が見て取れる。

#### 3.5.3.4 一文だけでは訳が確定しない

稀なケースではあるが、分脈がないと訳が確定できないものもあった。

| | |
|---|---|
| 入力 | The encryption method in this case is AES as mentioned above. |
| 正解 | この際の暗号化方式は、上述のようにＡＥＳである。 |
| 出力 1 | この場合の暗号化方法は、上述のようなＡＥＳである。(3, 5) |
| 出力 2 | この場合の暗号化方法は、上述したような AES である。(3, 5) |

この例では、AES というものが 1 つしかないならば（前の文脈でそのような記述がされているならば）正解訳とする必要があるが、AES には何種類かあり、そのうちの一つが前の文脈で記載されているならば、出力が正しい訳となる。このように、文脈情報がないと正しく翻訳できないものも、割合は少ないが存在する。

### 3.5.4 まとめ

本稿では WAT2017 の特許日英・英日翻訳タスクの JPO Adequacy Evaluation に対して、2 つの観点からの分析を行った。質の良い翻訳システムの評価において κ 係数が不当に低く算出されることを明らかにし、κ 係数だけを見ても評価の信頼性は測れないことを指摘した。今後は κ 係数だけでなく、評価値の詳細も明らかにする必要があると思われる。

また評価結果が 2 名の評価者間で乖離している例を分析したところ、その要因は評価者の単純なミスによるもの、評価尺度の定義の解釈の違いによるもの、訳語の不統一によるものなどがあった。評価者のミスはある程度は仕方がないが、評価尺度の定義や訳語の不統一の扱いなどは事前に基準を決めておくなどすれば、より一致度の高い評価が行えると思われる。

### 参考文献

[1] Overview of the 4th Workshop on Asian Translation, In Proceedings of the 4th Workshop on Asian Translation (WAT2017).

[2] https://www.jpo.go.jp/shiryou/toushin/chousa/tokkyohonyaku_hyouka.htm

# 4. The 7th Workshop on Patent and Scientific Literature Translation (PSLT 2017) 報告

# 4. The 7th Workshop on Patent and Scientific Literature Translation (PSLT 2017) 報告

奈良先端科学技術大学院大学　須藤　克仁

## 4.1 開催概要

本研究会の活動の一環として、2005 年の第 1 回から数えて 7 回目となるワークショップを機械翻訳サミットの会議最終日である 9 月 22 日（金）に開催した。前回 2015 年に引き続き特許・技術文書翻訳ワークショップ（Workshop on Patent and Scientific Literature Translation）と題している。Co-chair は宇津呂・須藤の両名が務め、綱川先生（静岡大）に Publication Chair をご担当いただいた。プログラム委員として研究会委員及び 6 名の国内外の研究者にご協力いただき、技術講演論文の査読を依頼した。

招待講演者の選定は昨年度末から研究会での議論を通じて行い、知財関係公的機関、翻訳者／翻訳会社関係者、機械翻訳研究者から各 1 名ずつに依頼した。

技術講演に関しては、本会議研究論文トラックの投稿期限及び採否発表スケジュールが延長されたこともあってか、投稿数が伸び悩み査読を行った論文は 4 件にとどまり、査読の結果 3 件を採択した。

また、ワークショップ当日の参加者は 30 数名程度であり、前回 2015 年マイアミで開催されたときと同程度であったが、多くが日本からの参加者であったと見受けられた。

## 4.2 ワークショップ報告

午前のセッションは招待講演 2 件であった。

1 件目の招待講演は WIPO の Bruno Pouliquen 氏で、前回の PSLT 2015 に引き続いて WIPO の機械翻訳システムについてご紹介いただいた。前回はフレーズベース統計的機械翻訳(PBMT)を基本として、データ選択、多言語化、事前並べ替え等の前処理等についてのものであったが、今回はその後のニューラル機械翻訳(NMT)への技術動向の変化を捉えた、特許翻訳向け NMT についての講演であった。NMT は PBMT に比べモデルのサイズが非常に小さくできること、非常に流暢な訳文が生成できること等特許翻訳においても大きなアドバンテージがあり、実際 WIPO の新しい特許向け NMT システムにおいても従来の PBMT システムを大きく上回る翻訳精度を達成しており、同じく NMT を採用する Google 翻訳よりも優れていること等が示された。

2 件目の招待講演は MK 翻訳事務所の梶木正紀氏で、今年 2 月の JTF ジャーナルの Google NMT 特集において翻訳者の視点から見た NMT について寄稿をされている等我々にとって有益なご見解が伺えると考えて講演を依頼した。梶木氏はすでに機械翻訳を使った翻訳ワークフローの転換を近々に進めようとされていて、そのために Google や Microsoft の NMT を用いた翻訳後編集作業のテストを行い、担当した翻訳者からも 20-30%の作業効率の改善への効果が期待できるという反応を得たことについてご紹介いただいた。

午後のセッションは招待講演 1 件と技術講演 3 件であった。

招待講演は Apple の Andrew Finch 氏で、前職の NICT 在籍時の統計的翻字に関する技術の進展についてご紹介いただいた。翻字は音訳とも言い、日本語では computer という英語をコンピュータという表記に変換されるような過程を指す。問題としては機械翻訳を簡略化したものと捉えることもできるため、統計的機械翻訳と同様の技術によって進化している。近年のニューラルネットワーク翻訳の技術は翻字においても同様に有効であり、従来の統計的機械翻訳を応用したものを上回る精度が達成できていることが示された。

技術講演 1 件目は筑波大の木村氏らによる、ニューラル機械翻訳における訳抜けの影響に関する研究の発表であった。著者らの研究グループで取り組んでいる専門用語等を統計翻訳のフレーズテーブルを利用して翻訳した結果をニューラル機械翻訳結果の該当箇所に埋め込む翻訳方式の利点として訳抜けが少ないことが確認できたことが報告された。

技術講演 2 件目は東芝ソリューションズ／静岡大の熊野氏らによる、対訳辞書エントリの派生語を自動的に生成して対訳辞書を拡張する研究の発表であった。既存の対訳エントリから派生変化パターンを抽出し、品詞や接辞辞書の情報から付与した素性と合わせた派生変化規則の構築と、構築された派生変化規則を適用して得られる派生候補を Web テキストデータでの検証を行い、動詞的名詞のエントリ拡張で有効性を確認したことが報告された。

技術講演 3 件目は北京師範大学の李氏らによる、特許に関する言語リソースを活用した中英特許機械翻訳の改善に関する研究の発表であった。著者らが SIPO の特許コーパスを利用して構築している CTKB（中国特許知識ベース）や対訳辞書、翻訳規則等を活用して NTCIR9 共通タスクにおけるルールベース機械翻訳の性能を上回ったことが報告された。

### 4.3 所感

今回ワークショップの企画当初より、特に注目したかった内容はニューラル機械翻訳技術が特許・技術文書翻訳に与える影響や、そこでの従来型の言語リソースの活用可能性であった。Pouliquen 氏の講演は近年の技術動向の大きな変化を如実に感じさせるものであり、梶木氏の講演ではそれが人手の翻訳にまで明確に波及しつつあることが語られた。Finch 氏の講演はその変化を翻字というタスクの視点から捉えたものであったし、技術講演でもニューラル機械翻訳と対訳言語リソースの関係について議論があった。

技術文書特有の問題のうち、多数の固有名詞の問題と非常に長い文の翻訳については広く活発に研究されていると言えるが、実際に難解な技術文書での実験評価を行っているケースはあまり多くなく（アジア言語ワークショップ WAT は科学技術論文や特許翻訳の共通タスクを行っているおそらく現在唯一のものと言える）、産業界の需要に学術界が応えられていないのではないかと感じる面もある。今回は例年以上に技術講演論文の投稿が少なく、本ワークショップの visibility や位置づけの明確化等不十分な面があったことは否めない。単に学術一辺倒でなく、人手の翻訳との関係や実際の産業応用等、昨今の技術進展に合わせた展開を見据えたワークショップ（あるいはシンポジウム）の企画が重要であると強く感じる。

平成 29 年度 AAMT/Japio 特許翻訳研究会報告書

発 行 日　平成 30 年 3 月

発　　行　一般財団法人　日本特許情報機構（Japio）
〒135−0016　東京都江東区東陽町 4 丁目 1 番 7 号
佐藤ダイヤビルディング
TEL：(03) 3615−5511　FAX：(03) 3615−5521

編　　集　アジア太平洋機械翻訳協会（AAMT）

印　　刷　株式会社インターグループ