Multi-domain Adaptation for Statistical Machine Translation Based on Feature Augmentation

Kenji Imamura† and Eiichiro Sumita†

Domain adaptation is a major challenge when machine translation is applied to practical tasks. In this study, we present domain adaptation methods for machine translation that assume multiple domains. The proposed methods combine two types of models: a corpus-concatenated model covering multiple domains and single-domain models that are accurate but sparse in specific domains. We combine the advantages of both the models using feature augmentation for domain adaptation in machine learning; however, a conventional method of feature augmentation for machine translation uses a single model. Our experimental results show that the translation qualities of the proposed method improved or were at the same level as those of the single-domain models. The proposed method is extremely effective in low-resource domains. Even in domains having a million bilingual sentences, the translation quality was at least preserved and even improved in some domains. These results demonstrate that state-of-the-art domain adaptations can be realized with appropriate model selection and appropriate settings, even when standard log-linear models are used.

† National Institute of Information and Communications Technology
Key Words: Domain Adaptation, Phrase-based Statistical Machine Translation, Feature Augmentation, Corpus-concatenated Model, Empty Value

1 はじめに

さまざまな種類のテキストや、音声認識結果が機械翻訳されるようになってきている。しかし、すべてのドメインのデータにおいて、適切に翻訳できる機械翻訳器の実現はいまだ困難であり、翻訳対象ドメインを絞り込む必要がある。対象ドメインの翻訳品質を向上させるには、学習データ（対訳文）を大量に収集し、翻訳器を訓練するのが確実である。しかし、多数のドメインについて、対訳文を大量に収集することはコスト的に困難であるため、他のドメインの学習データを用いて対象ドメインの翻訳品質を向上させるドメイン適応技術が研究されている（Foster and Kuhn 2007; Foster, Goutte, and Kuhn 2010; Axelrod, He, and Gao 2011; Bisazza, Ruiz, and Federico 2011; Sennrich 2012; Sennrich, Schwenk, and Aransa 2013）。このドメイン適応は、機械翻訳を実用に供するときには非常に重要な技術である。

本稿では、複数ドメインを前提とした、統計翻訳の適応方式を提案する。本稿の提案方式は、複数のモデルを対数線形補間で組み合わせる方法である。シンプルな方法であるが、機械学習分野のドメイン適応方法である素性空間拡張法（Daumé 2007）の考え方を流用することで、複数ドメインの利点を活かす。具体的には、以下の2方式の提案を行う。

(1) 複数ドメインの同時最適化を行う方法。この場合、拡張された素性空間に対して、マルチドメイン対応に変更した最適化器で同時最適化を行う。

(2) 複数ドメインを一つ一つ個別に最適化する方法。この場合、素性空間を制限し、通常の対数線形モデルとして扱う。既存の翻訳システムへの改造が少なくても実現できる。

いずれの方法も、さまざまなドメインで未知語が少ないコーパス結合モデルと、ドメインを限定した際に翻訳品質が単独ドメインモデルを併用する。さらに、複数モデル組み合わせ時のハイパラメータをチューニングする。

素性空間拡張法を機械翻訳に適用した例には、Clark, Lavie, and Dyer (2012) がある。これは、翻訳文の尤度の算出に用いられる素性ベクトルの重みだけを適応させていて、素性関数は適応していない。本稿の新規性は、コーパス結合モデルと単独ドメインモデルを使って、素性関数を適応させること、および、複数モデル組み合わせ時のハイパラメータを適切に設定することの2点である。モデルの選択と設定を適切に行うことによって、最先端のドメイン適応と同等以上の精度が出せるすることを示す。

なお、本稿では、事前並べ替えを使ったフレーズベース統計翻訳方式 (PBSMT)(Koehn, Och, and Marcu 2003; Koehn, Hoang, Birch, Callison-Burch, Federico, Bertoldi, Cowan, Shen, Moran, 2005; Koehn 2007) を用いた。
Zens, Dyer, Bojar, Constantin, and Herbst 2007）を対象とする。以下、第2節では、統計翻訳のドメイン適応に関する関連研究を述べる。第3節では、提案方式を詳細に説明する。第4節では、実験を通じて本方式の特徴を議論し、第5節でまとめる。

2 統計翻訳のドメイン適応

機械翻訳のドメイン適応は、翻訳対象のドメイン（内ドメイン）データが少なく、他のドメイン（外ドメイン）データが大量にある場合、内外ドメインのデータ双方を使って、内ドメインの翻訳品質向上させる技術である。

ドメイン適応には、「ニュース」「Web」のように、あらかじめデータがどのドメインに属するか決まっている場合の他に、自動クラスタリングによって仮想的に作られる場合もある。自動で分割されたドメインでも、重みを最適化することによって、トータルの翻訳品質が向上するという報告もある（Finch and Sumita 2008; Sennrich et al. 2013）。本稿では、前者のあらかじめデータのドメインが決まっている場合で議論する。

2.1 コーパス結合

最もシンプルなベースラインとして用いられている方法は、内ドメインと外ドメインのデータを結合して学習し、1つのモデルを構築する方法である（本稿ではコーパス結合方式と呼ぶ）。学習された結合モデルは、各ドメインの開発セットで最適化される。

一般的な機械学習では、結合されたコーパスで学習したモデルは、内ドメイン、外ドメイン双方の中間的性質を持つため、その精度も内ドメインデータのみ、外ドメインデータのみで学習されたモデル（単独ドメインモデルと呼ぶ）の中間の精度になることが多い。一方、機械翻訳の場合、コーパスを結合することにより、カバーする語彙が増加するため未知語が減少し、単独ドメインモデルより翻訳品質が向上する場合もある。最終的に翻訳品質が向上するか否かは、未知語の減少とモデルパラメータの精度低下のトレードオフになる。

2.2 線形補間、対数線形補間

統計翻訳では、翻訳に使用するサブモデル（フレーズベース、言語モデル、並べ替えモデルなど）が定数値（素性関数数）を、線形または対数線形結合して、翻訳文の尤度を算出する。対数線形モデルでは、以下の式で尤度 \( \log P(e|f) \) を算出する。

\[
\log P(e|f) \propto w \cdot h(e, f) \tag{1}
\]

ただし、eは翻訳文、fは原文、\( h(e, f) \)は素性ベクトル、\( w \)は素性関数の重みベクトルである。

このとき、重みベクトル\( w \)をドメイン毎に切り替えることで、ドメイン依存訳を生成する方
法がある。たとえば，Foster and Kuhn (2007) は，PBSMT のサブモデルをドメイン毎に訓練し，線形補間，対数線形補間でドメイン毎の重みを変えて翻訳を行った。彼らはパープレキシティなどを目標関数にして，独自の重み推定を行ったが，近年は，重みの推定に誤り率最小訓練法 (MERT) (Och 2003) などの最適化方法が用いられている (Foster et al. 2010)。

素性空間拡張法 (Daumé 2007) は，翻訳に限らず，機械学習全般に使われるドメイン適応方式で，素性間数の重みをドメイン毎に最適化する (3.1 節参照)。Clark et al. (2012) は，これを対数線形補間方式の一種として翻訳に適用し，効果があったと報告している。なお，彼らは単一のモデルを用い，モデルの重みのみをドメイン適応させている。

2.3 モデル適応

重みベクトルではなく，素性ベクトル \( h(e, f) \) (素性間数) を変更することによってドメイン適応する方法は，大きく 2 つに分けられる。一つは，訓練済みサブモデル自身を変更する方法である。もう一つは，ドメインに適応させたコーパスからモデルを訓練する方法である。このうち，サブモデル自身を変更する方法には，fill-up 法 (Bisazza et al. 2011)，翻訳モデル混合 (Sennrich 2012)，インスタンス重み付け (Foster et al. 2010; Matsoukas, Rosti, and Zhang 2009) が知られている。

fill-up 法は，フレーズテーブルから翻訳候補を取得する際，内ドメインの単独モデルにフレーズが存在する場合はその素性間数値を使用，存在しない場合は外ドメインの単独モデルからフレーズを取得し，その値を使用する。

翻訳モデル混合は，2 つのフレーズテーブルに記録された翻訳確率を重み付きで混合し，新たなフレーズテーブルを生成する。重みは開発セットのパープレキシティが最小になるように，素性間数毎に設定される。

インスタンス重み付けは，フレーズテーブルのモデルパラメーター一つ一つを，内ドメインと外ドメインを識別するように内ドメインの開発セットで混合する。

これらの方法は，素性間数値のみでなく，デコード時のフレーズ候補も 2 つのモデルのフレーズを使用するため，一般的には未知語は減少する。しかし，フレーズテーブル以外のサブモデルは，別の構築・混合法を使わなければならないというデメリットも存在する。

2.4 コーパスフィルタリング

素性ベクトル \( h(e, f) \) を変更するもう一つの方法は，モデル訓練用コーパスをドメイン適応させる方法である。最初に述べたコーパス結合も，この一種であるが，よりドメインに適応させると，外ドメインコーパスから対訳文を取捨選択した方がよい。この外ドメインコーパスから対訳文を取捨選択し，内ドメインコーパスとともにモデルを作成する方式を，コーパスフィルタリングと呼ぶ。Axelrod et al. (2011) は，内ドメインに適した対訳文を，内外ドメインの交
差エントロピーの差に基づき、外ドメインコーパスから取捨選択した。選択された文を内ドメインコーパスに追加してモデルを訓練することで、ドメイン適応を行った。

コーパスフィルタリングは、翻訳器が使用する全サブモデルを、その種類を問わず適応させることができる点がメリットであるが、最適な追加訓練文数は、あらかじめ予想できない点がデメリットである。

2.5 その他の方法

その他の方法としては、2つの翻訳器を直列に接続し、外ドメイン翻訳器による翻訳結果を、さらに内ドメイン翻訳器を使って訂正する方法もある（Jeblee, Feely, Bouamor, Lavie, Habash, and Oflazer 2014）。これは、ドメイン依存訳の生成を一種の誤り訂正ととらえていることに相当する。

3 マルチドメイン適応方式

3.1 素性空間拡張法

素性空間拡張法（Daumé 2007）は、一般的な機械学習におけるパラメータ（統計翻訳では素性の重みに対応）のドメイン適応に用いられる方式である。素性空間を共通、内ドメイン（ターゲットドメイン）、外ドメイン（ソースドメイン）に分割し、素性を、それが由来するドメインごとに異なる空間に配置する。内ドメインの素性は共通空間と内ドメイン空間に、外ドメインの素性は共通空間と外ドメイン空間にコピーして配置するのが特徴である。そして、全体を最適化することにより、適応された重みベクトルを得る。

素性が線形のとき、共通空間に格納される素性は、内ドメインデータから得られる素性と、外ドメインデータから得られる素性の和（OR）になる。そのため、共通空間にある素性が、内外ドメインのお互いに欠けた素性を補完し、精度が向上する。統計翻訳のような密な実数素性の場合、共通空間を介して、外ドメインの素性が内ドメインの事前分布として作用し、より内ドメインに適合した重みに調整される（Daumé (2007) の3.2 節参照）。

通常は、外ドメインを内ドメインに適応するために使用されるが、素性空間拡張法では、外ドメインと内ドメインを同等に扱っており、容易にDドメインに拡張することができる。その場合、素性空間は共通、ドメイン1、…、ドメインDのように、D+1空間に分割される（図1）。

すなわち、

\[ h(f, e) = \langle h_c, h_1, \ldots, h_i, \ldots, h_D \rangle \]  

ただし、\( h_c \)は共通空間の素性ベクトル、\( h_i \)はドメイン依存空間の素性ベクトルである。共通空
間には常に素性が配置されるが、ドメイン依存空間には、データの属するドメインが一致する場合だけ、素性が配置される。

\[
\begin{align*}
  h_c &= \Phi(f, e) \\
  h_i &= \begin{cases} 
    \Phi(f, e) & \text{if \( \text{domain}(f) = i \)} \\
    \emptyset & \text{otherwise}
  \end{cases}
\end{align*}
\]

ただし、\( \Phi(f, e) \) はモデルスコア等を格納した部分ベクトルで、素性空間を拡張しない場合は、\( h(f, e) \) と同じになる。この素性マトリクスを最適化し、重みベクトルを得る。

4節の実験では、Moses ツールキット (Koehn et al. 2007) のデフォルト素性 (15 次元) を使う。素性の一覧を表 1 に示す。これに素性空間拡張法を適用すると、共通空間では 15 次元、ドメイン依存空間では各 14 次元となる1。

Clark et al. (2012) は、アラビア語→英語翻訳（ドメインは News と Web）、およびチェコ語→英語翻訳（ドメインは Fiction など 6 ドメイン）について素性空間拡張法を適用し、効果が観測されたと報告している。ただし、彼らは、翻訳モデル、言語モデルなどのサブモデルは、コーパス結合モデル 1 種類だけを使用しており、純粋に素性関数の重みだけをドメイン最適化している。

1 未知語数を表す素性 UnknownWordPenalty は、重みの調整は不可とされているため、共通空間だけに配置する。
表1 本稿で用いる素性の一覧

| 素性関数名         | 内容                                | 次元数 | 備考                                                   |
|-------------------|-------------------------------------|--------|--------------------------------------------------------|
| UnknownWordPenalty| 原言語の未知語数                    | 1      | 重みは−100固定（調整不可）                             |
| WordPenalty       | 目的言語の単語数                    | 1      |                                                        |
| PhrasePenalty     | フレーズ数                           | 1      |                                                        |
| PhraseDictionary  | フレーズの翻訳確率                   | 4      | 原言語→目的言語及び方向のフレーズ翻訳対数確率、および単語翻訳対数確率の総和。フレーズテーブルから取得。 |
| LexicalReordering | 言語変化並び替えモデル確率            | 6      | 原言語→目的言語及び方向について、片側で隣接するフレーズの反対側での接続形式（単調、交換、分離）それぞれの対数確率、言語変化並び替えモデルファイルから取得。 |
| Distortion        | 歪モデルスコア                       | 1      | フレーズ間距離の総和                                    |
| LanguageModel     | 言語モデル確率                       | 1      | n-gram 言語モデルの対数確率                             |

3.2 提案法

3.2.1 コーパス結合モデルと単独ドメインモデルの導入

機械翻訳では、素性の重みより素性関数の方が翻訳品質に対する影響が大きいため、素性空間によってモデルを切り替えるのは自然な拡張である。

本稿で用いる素性関数は、実数値を返す関数であるが、その実態は、モデルファイルに記録されているインスタンスを読み込み、それに応答する値を返している。もし、訓練データ不足などの理由でモデルファイル中にインスタンスが存在しない場合、素性関数値は極小値になる。つまり、実数値の素性関数であっても、実際には二値素性と同様に、スベースネスの問題がある。そこで本稿では、二値素性の素性空間拡張法の考え方を素性関数に適用し、共通空間に対応するモデルとして、全ドメインのコーパスから作成したコーパス結合モデルを使用する。ドメイン依存空間には、それぞれの単独ドメインモデルを使用する。具体的には、

- フレーズテーブル、言語化並び替えモデル、言語モデルなどのサブモデルについて、単独ドメインモデルとコーパス結合モデルを作成しておく。
- 素性空間拡張では、共通空間にはコーパス結合モデルのスコアを素性関数値として配置し、各ドメインの空間には、単独ドメインモデルのスコアを配置し、最適化する（図1）。

式 (3)(4) は以下の式に置換される。

\[
    h_c = \Phi_c(f, e) \tag{5}
\]

\[
    h_i = \begin{cases} 
    \Phi_i(f, e) & \text{if } \text{domain}(f) = i \\
    \emptyset & \text{otherwise}
    \end{cases} \tag{6}
\]
ただし、\( \Phi_i(f, e) \) はコーパス結合モデルから得られた素性ベクトル、\( \Phi_i(f, e) \) は、単独ドメインモデル \( i \) から得られた素性ベクトルである。

・ デコーディングの際は、まず、単独ドメインモデルとコーパス結合モデルのフレーズテーブルをすべて検索し、翻訳仮説を生成する。探索の際には、共通空間と対象ドメインの空の素性だけを使って尤度計算をする。

翻訳仮説生成にコーパス結合フレーズテーブルを使用することにより、他のドメインで出現した翻訳フレーズも利用でき、未知語の減少が期待できる。また、翻訳仮説が単独ドメインモデルに存在している場合、高い精度の素性関数値が得られると期待される。

本方式は、拡張素性空間を最適化することによって、重みベクトル \( w \) を適応し、コーパス結合モデルと単独ドメインモデルを併用することで素性関数 \( h(e, f) \) の適応を行っている。どちらもモデルの種類に依存しないため、翻訳に使用する全サブモデルについて、適応させることができる。

なお、機械翻訳では、言語モデルを大規模な単言語コーパスから作成する場合がある。このモデルは、いわば非常に多くのドメインを含むコーパス結合モデルに相当するため、共通空間に配置するとよい。このように、外部知識から得られたモデル（素性関数）を追加する場合には、共通空間の次元を増加させる。

3.2.2 empty 値

本方式では、コーパス結合モデル、単独ドメインモデルのどちらか一方にのみ出現するフレーズ対が多数存在する。これらフレーズなどのインスタンスに関しても素性関数は値を返す必要がある。この値を本稿では empty 値と呼ぶ。これはいわば n-gram 言語モデルにおける未知語確率に相当するものであるので、フレーズの翻訳確率分布から算出されるべきものであるが、本稿では、バイパーサラメータとして扱い、開発コーパスにおける BLEU スコアが最高になるよう、実験的に設定する。

3.3 最適化

3.3.1 同時最適化

一般的な機械学習における素性空間拡張法の利点の一つは、素性空間を操作しているだけで、最適化アルゴリズムが既存の方法が使えるという点である。

機械翻訳の場合、最適化方法には、誤り率最小訓練 (MERT)(Och 2003)、ベアランク最適化 (PRO)(Hopkins and May 2011)、k ベストパッチ MIRA (KBMIRA) (Cherry and Foster 2012) が知られている。本稿では、高次元の最適化に適した KBMIRA を使用する。

2 Moses では、empty 値として一律に –100 を割り当てている (Koehn and Schroeder 2007; Birch, Osborne, and Koehn 2007)。4 節で述べるように、これは小さすぎるように BLEU スコアが下がる原因となる。
3 もう一つの理由は、予備実験において、ベースラインシステムの BLEU スコアが最も高かったためである。
通常の機械学習における最適化と、機械翻訳の最適化の大きな相違点は、多くの機械学習の損失関数が、尤度などデコーダが出力するスコアを使用しているのに対して、機械翻訳は BLEU (Papineni, Roukos, Ward, and Zhu 2002) のような、翻訳文の自動評価値を使用する点である。この自動評価値は、翻訳文と参照文との比較によって算出され、コーパス単位に計算される場合が多い。実際、MERT, KBMIRA は開発セットの BLEU スコアを損失関数の一部に使用している。つまり、複数ドメインを同時に最適化する場合は、ドメイン毎に BLEU スコアを算出しないと、結果がドメイン最適にならないことを意味している。

上記問題を解決するため、本稿では KBMIRA を変更する。Cherry and Foster (2012) のアルゴリズム 1 に対する変更点は、以下のとおりである。

1. 処理済み翻訳文の BLEU 統計量 (n-gram 一致数など) を保存する変数 $BG$ を、1 つからドメイン数 $D$ 個に拡張する。
2. 各翻訳文の BLEU スコアは、その翻訳文のドメイン $i$ の $BG_i$ から算出する。
3. 素性重みを更新後、その翻訳文の BLEU 統計量を $BG_i$ に追加する。

この変更によって、各ドメイン空間の素性重みは、そのドメインの開発セットに最適化される。

### 3.3.2 個別最適化

同時最適化は、適応させたいドメインが限られている場合にも、すべてのドメインを最適化しなければならないので、少々非効率である。そこで、全 $D$ ドメインのうち、ドメイン $i$ だけ適応させたい場合、ドメイン $i$ に関連する空間だけに限り、最適化を行う。これを本稿では個別最適化と呼ぶ。

個別最適化は、素性空間を共通空間とドメイン $i$ 空間に制限し、チューニングデータもドメイン $i$ に関するものだけにする。すなわち、式 (2) は式 (7) に置き換える。

$$ h(f, e) = \langle h_c, h_i \rangle \tag{7} $$

$$ h_c = \Phi_c(f, e) \tag{8} $$

$$ h_i = \Phi_i(f, e) \tag{9} $$

これは、一般的な対数線形モデルであるので、同時最適化を行わなくても、既存の最適化器をそのまま使うことができる。また、デコーダも、(1) 複数モデルを同時に使用すること、(2) empty 値を設定できること、の 2 点を満たすものであればよいため、既存のものを少し修正するだけで利用可能となる。

同時最適化に比べると、共通空間の最適化が弱くなる恐れがあるが、もともと機械翻訳は素性の重みよりも素性関数の影響の方が大きいため、実用上は問題は少ないと考えられる。ただし、

---

4 Clark et al. (2012) が素性空間拡張法の最適化に使用した PRO は文単位に近似した BLEU スコアを用いている。
以下の2点に関しては、同時最適化と共通に満たす必要がある。
(1) 共通空間の素性にコーパス結合モデルを使うこと。
(2) empty 値を適切に設定すること。

4 実験

4.1 実験設定

4.1.1 ドメイン／コーパス

本稿では、英日／日英翻訳を対象に、以下の4つのドメインの最適化を行う。各ドメインのコーパスサイズを表2に示す。MED コーパスは比較的小規模で、それ以外のドメインは100万文規模である。なお、訓練文は80単語以下のものだけを使用した。

- MED: 病院等における医師（スタッフ）と患者の疑似対話のコーパス。内部開発。
- LIVING: 外国人が日本に旅行や在留する際の疑似対話コーパス、内部開発。
- NTCIR: 特許コーパス。訓練コーパスと開発コーパスは国際ワークショップ NTCIR-8、テストコーパスは NTCIR-9 のものを使用5。
- ASPEC: 科學技術文献コーパス (Nakazawa, Yaguchi, Uchimoto, Utiyama, Sumita, Kurohashi, and Isahara 2016) 6. ASPEC-JE のうち、対訳信頼度の高い100万文を使用。

4.1.2 翻訳システム

各コーパスの対訳文は、内部開発の事前並べ替え (Goto, Utiyama, Sumita, and Kurohashi (2015) の4.5節) を適用したのちに使用した。これは、新聞、Wikipedia、旅行会話、生活会話、科学技術文、ブログなどを混合した対話コーパスから、ランダムに500万文程度を抽出して訓練した。特定のドメインに依存しない事前並べ替えである。今回は、全ドメイン、全方式について同一の事前並べ替えを適用した。

| ドメイン | 文数 | 訓練 | 開発 | テスト | 訓練単語数 | 英語 | 日本語 |
|----------|------|------|------|-------|------------|------|-------|
| MED      | 222,945 | 1,000 | 1,000 |       | 3.1 M      | 3.3 M |
| LIVING   | 986,946 | 1,800 | 1,800 |       | 14.3 M     | 16.5 M|
| NTCIR    | 1,387,713 | 2,000 | 2,000 |       | 48.7 M     | 52.3 M|
| ASPEC    | 1,000,000 | 1,790 | 1,784 |       | 25.9 M     | 28.7 M|

5 http://research.nii.ac.jp/ntcir/index-ja.html
6 http://lotus.kuee.kyoto-u.ac.jp/ASPEC/
また、日本語に関しては、訓練、テスト等すべての文について、あらかじめ形態素解析器MeCab(Kudo, Yamamoto, and Matsumoto 2004)で単語分割を行ってから使用した。英語に関しては、Mosesツールキット(Koehn et al. 2007)付属のtokenizer、truecaserで前処理した。

翻訳システムの訓練のうち、フレーズテーブル、語義化並び替えモデルの学習にはMosesツールキットをデフォルト設定で使用した。言語モデルはKenLM(Heafield, Pouzyrevsky, Clark, and Koehn 2013)を用いて、訓練セットの目的語言語側から5グラムモデルを構築した。最適化は3.3.1節で述べたマルチドメインKBMIRAを使用した。

デコーディングには、内部開発のMosesのクローンデコーダを使用した。デコーダの設定値はMosesのデフォルト値と同じphrase_table_limit = 20, distortion_limit = 6。ビーム幅200とした。なお、フレーズ候補取得の際には、複数のフレーズテーブルをすべて検索して候補を取得後、拡張素性空間で算出された尤度に従って、20個に絞り込んだ。

4.1.3 empty 値

3.2.1節で述べたempty値は、−3から−20の間の整数値のうち、全ドメインの開発セット翻訳結果を1ドキュメント扱いしたときのBLEUスコアが最高値になる値を採用した。結果、英日翻訳に関しては−7、日英翻訳は−6となった。これを確率値とみなしの場合、英日はexp(−7) ≈ 0.0009、日英はexp(−6) ≈ 0.0025となる。

4.1.4 評価指標

評価指標にはBLEU、翻訳集計率(TER)(Snover, Dorr, Schwartz, Micciulla, and Makhoul 2006), Meteor(Denkowski and Lavie 2014)(英語のみ), RIBES(Isozaki, Hirao, Duh, Sudo, and Tsukada 2010)(日本語のみ)を使用し、MultEvalツール(Clark, Dyer, Lavie, and Smith 2011)7で有意差検定を行った8。危険率はp < 0.05とした。最適化の揺れを吸収するため、5回最適化を実施し、その平均値を使用した。なお、単純化のため、本文ではBLEUで説明する。

4.1.5 比較方式

各ドメインコーパスだけでモデルを構築、最適化、テストする単独ドメインモデル方式をベースラインにし、他の方式と比較する。従来法としては、2節で述べた以下の方式を使用する。

- コーパス結合：全ドメインのコーパス結合モデルを使用し、各ドメインの開発セットで最適化、テストした場合。

4.1.5 素性空間拡張法(Clarck)：共通空間、ドメイン空間共に、コーパス結合モデルの素性関

---

7 https://github.com/jhclark/multeval
8 RIBESに関しては、http://www.kecl.ntt.co.jp/icl/lirg/ribes/index.htmlのスクリプトと同等なものをMultEvalツールに組み込んで測定した。

607
数を使った素性空間拡張法。Clark et al. (2012) の設定と同じだが、最適化にはマルチドメイン KBMIRA を使用した。

- **Fill-up 法**: ドメイン適応方式に fill-up 法 (Bisazza et al. 2011) を用いた場合。
- **翻訳モデル混合**: ドメイン適応方式に翻訳モデル混合 (Sennrich 2012) を用いた場合。Moses 付属の tmcombine プログラムで混合した。
- **コーパスフィルタリング**: Axelrod et al. (2011) が提案した修正 Moore-Lewis フィルタリングで、自分以外のドメインのコーパスから対訳文を選択し、ドメインの訓練文に加えた場合。追加文数は、10 万文単位に変数を変え、各ドメインの開発セットの BLEU スコアが最高になった文数を採用した。

提案法は、以下のバリエーションをテストする。

- **提案法（同時最適化）**: 提案法のうち、3.3.1 節で述べた同時最適化を使用した場合。
- **提案法（個別最適化）**: 提案法のうち、3.3.2 節で述べた個別最適化を使用した場合。
- **提案法 (empty = -100)**: 提案法の empty 値を、Moses と同じ -100 に設定した場合。なお、最適化は個別最適化を使用したが、同時に最適化でも同じ傾向が観察された。
- **提案法（外ドメイン）**: 提案法のうち、共通空間に対応するモデルとして、コーパス結合モデルではなく、内ドメインデータを取り除いた外ドメインコーパス（つまり、内ドメイン以外の 3 ドメインのコーパス）だけを学習したモデルを使用した場合。これも個別最適化を使用した。

### 4.2 翻訳品質

各方式について、英日翻訳および日英翻訳における自動評価スコアを、それぞれ表 3、表 4 に示す。なお、表中太字は方式間最高値、(+) は、単独ドメインモデル方式をベースラインとしたとき有意に向上したもの、(−) は有意に悪化したものを表す (p < 0.05)。

今回用いたデータは、ドメイン同士の関連性が比較的薄かったため、従来法では、適応によって翻訳品質は軒並み悪化した。たとえば、コーパス結合方式と素性空間拡張法 (Clark) は、単独モデルより翻訳品質が低下する傾向が強かった。コーパス結合方式は各ドメインが平均化されたモデルが作成され、素性関数の精度が落ちたためと考えられる。

Fill-up 法は、コーパス結合方式に比べると翻訳品質は向上する場合が多かったが、単独ドメインモデルより悪化した。翻訳モデル混合は、ドメインによって有意に向上する場合と悪化する場合があり、この実験での有効性は確認できなかった。コーパスフィルタリングは、ASPEC の英日翻訳を除き、単独ドメインモデルより有意に向上、または同等品質となった。ASPEC 英日翻訳は 10 万文を加えただけだったが、これが悪影響しており、コーパスフィルタリング是有数の影響を及ぼした。

---

9 数十万文単位で追加文数を変えながら開発セット BLEU スコアを測定し、その後、最適文数近傍で 10 万文単位の調整を行った。
表 3 方式別の自動評価スコア（英日翻訳）

| 方式 | ドメイン | MED | LIVING |
|------|---------|-----|--------|
|      |         | BLEU | TER | RIBES | BLEU | TER | RIBES |
| 単独ドメインモデル | 23.23 | 62.46 | 78.34 | 24.56 | 61.33 | 78.78 |
| コーパス結合 | 22.65 (−) | 64.08 (−) | 77.53 (−) | 22.99 (−) | 63.41 (−) | 77.52 (−) |
| 素性空間拡張法 (Clark) | 22.49 (−) | 63.78 (−) | 77.70 (−) | 22.97 (−) | 63.40 (−) | 77.41 (−) |
| Fill-up | 22.42 (−) | 63.63 (−) | 77.46 (−) | 23.38 (−) | 63.16 (−) | 77.80 (−) |
| 翻訳モデル混合 | 23.81 (+) | 61.94 (+) | 78.37 | 24.05 (−) | 62.05 (−) | 78.62 (−) |
| コーパスフィルタリング | 24.02 (+) | 61.61 (+) | 78.43 | 24.50 | 62.08 (−) | 78.51 (−) |
| 提案法（同時最適化） | 23.69 (+) | 61.79 (+) | 78.67 (+) | 24.43 | 61.51 | 78.72 |
| 提案法（個別最適化） | 23.75 (+) | 61.78 (+) | 78.56 | 24.43 | 61.09 (+) | 78.80 |
| 提案法 (empty = −100) | 23.66 (+) | 62.37 | 78.12 | 23.91 (−) | 61.84 (−) | 78.39 (−) |
| 提案法（外ドメイン） | 23.79 (+) | 62.19 | 78.46 | 24.29 (−) | 61.77 (−) | 78.75 |

| 方式 | ドメイン | NTCIR | ASPEC |
|------|---------|--------|-------|
|      |         | BLEU | TER | RIBES | BLEU | TER | RIBES |
| 単独ドメインモデル | 38.62 | 47.85 | 80.16 | 32.69 | 54.12 | 78.16 |
| コーパス結合 | 38.09 (−) | 48.54 (−) | 79.88 (−) | 30.59 (−) | 55.95 (−) | 77.22 (−) |
| 素性空間拡張法 (Clark) | 38.09 (−) | 48.54 (−) | 79.90 (−) | 30.65 (−) | 55.91 (−) | 77.28 (−) |
| Fill-up | 38.37 (−) | 48.12 (−) | 80.03 (−) | 31.50 (−) | 55.06 (−) | 77.62 (−) |
| 翻訳モデル混合 | 38.32 (−) | 48.11 (−) | 80.09 (−) | 31.97 (−) | 54.77 (−) | 77.84 (−) |
| コーパスフィルタリング | 38.77 (+) | 47.82 | 80.17 | 32.57 (−) | 54.14 | 78.14 |
| 提案法（同時最適化） | 38.72 (+) | 47.77 | 80.22 (+) | 32.69 | 54.20 | 78.10 |
| 提案法（個別最適化） | 38.83 (+) | 47.64 (+) | 80.22 | 32.76 | 54.13 | 78.15 |
| 提案法 (empty = −100) | 38.56 | 47.90 | 80.10 (−) | 32.62 | 54.17 | 78.13 |
| 提案法（外ドメイン） | 38.65 | 47.78 | 80.28 (+) | 32.72 | 54.02 | 78.19 |

効だが、最適な追加文数の決定は難しいことを示している。

一方、提案法は、同時最適化、個別最適化ともに、すべてのドメインにおいて単独ドメインモデルより向上あるいは同等品質となり、適切に適応できた。同時最適化に比べ、個別最適化の方が、BLEU スコアが高い傾向がある。なお、empty 値を −100 にすると、最適化時に BLEU スコアが振動して、最適化できない場合があった（表中の N/A）。また、提案法でも、共通空間に使用するモデルを外ドメインモデルにすると、大部分のケースでは提案法に比べ品質が悪化した。共通空間に使用するモデルは、内ドメインを含むコーパス結合モデルの方が望ましいことを示している。

まとめると、提案法はドメイン適応方式の中ではほぼ最高品質を確保できた。特に、個別最適化方式のような標準的な対数線形モデルであっても、適切な設定をすれば、最先端方式と同等以上のドメイン適応が実現できることを示している。
表 4 方式別の自動評価スコア（日英翻訳）

| 方式             | コーパス結合 | 素性空間拡張法 (Clark) | Fill-up | 翻訳モデル混合 | コーパスフィルタリング | 提案法（同時最適化） | 提案法（個別最適化） | 提案法（empty = −100） | 提案法（外ドメイン） |
|-----------------|-------------|-----------------------|---------|--------------|---------------------|---------------------|---------------------|----------------------|---------------------|
|                 | 17.07       | 16.75 (−)             | 16.56 (−) | 17.55        | 18.14 (+)           | 18.14 (+)           | 18.43 (+)           | 17.32              | 17.32              |
|                 | 70.72       | 71.39 (−)             | 71.98 (−) | 71.17        | 70.14 (+)           | 69.29 (+)           | 69.85 (+)           | 70.93              | 70.93              |
|                 | 25.26 (−)   | 25.13 (−)             | 25.11 (−) | 25.65 (+)    | 25.90 (+)           | 26.90 (+)           | 26.00 (+)           | 25.34              | 25.34              |
|                 | 19.71       | 18.95 (−)             | 19.06 (−) | 19.99 (+)    | 19.66              | 20.16 (+)           | 20.17 (+)           | 19.66              | 19.66              |
|                 | 67.08       | 68.31 (−)             | 68.45 (−) | 67.22        | 67.31 (+)           | 68.50 (+)           | 66.94 (+)           | 67.31              | 67.31              |
|                 | 27.58       | 26.75 (−)             | 27.58 (−) | 27.31 (−)    | 27.33 (+)           | 27.49 (−)           | 27.67 (−)           | 27.58              | 27.58              |

| 方式             | NTCIR       | ASPEC                 |
|-----------------|-------------|-----------------------|
|                 | BLEU        | TER                   | Meteor   | BLEU        | TER                   | Meteor   |
|                 | 33.63       | 52.67                 | 35.68    | 21.75       | 64.95                 | 31.01    |
|                 | 33.21 (−)   | 52.94 (−)             | 35.33 (−) | 20.41 (−)   | 66.00 (−)             | 30.36 (−) |
|                 | 33.24 (−)   | 53.00 (−)             | 35.38 (−) | 20.39 (−)   | 66.18 (−)             | 30.33 (−) |
|                 | 33.14 (−)   | 53.06 (−)             | 35.48 (−) | 20.98 (−)   | 65.41 (−)             | 30.58 (−) |
|                 | 33.32 (−)   | 52.78 (−)             | 35.54 (−) | 21.16 (−)   | 65.17 (−)             | 30.77 (−) |
|                 | **33.73**   | **52.45 (+)**         | **35.71** | **21.72**   | **64.71 (+)**         | **31.03 (+)** |
|                 | 33.68       | 52.42 (+)             | 35.70    | 21.75       | 64.79 (+)             | 31.20 (+) |
|                 | 33.70       | **52.33 (+)**         | 35.67    | **21.81**   | **64.76 (+)**         | **31.19 (+)** |
|                 | N/A         | N/A                   | N/A      | N/A         | N/A                   | N/A      |
|                 | 33.52 (−)   | 52.70                 | 35.62    | 21.73       | **64.72 (+)**         | 31.06    |

4.3 シングルドメイン適応としての効果

ドメイン適応が必要となる場面は、新たなドメインデータの翻訳を行わなければならないにも関わらず、十分な量の訓練文が集まらない場合である。本節では、MED 英日翻訳に絞って、訓練コーパスのサイズを変えて翻訳品質を測定する。なお、他のドメインについては変更せず、全訓練文を使用する。

表 5 は、単独ドメインモデル、コーパス結合、コーパスフィルタリングと提案法（個別最適化）を比較した結果である。表中の (+) は単独ドメインモデルと比較して有意に高く、 (−) は有意に低いことを表す。(*) はコーパス結合と比較して有意に高く、(†) はコーパスフィルタリングと比較して有意に高いことを表している (p < 0.05)。

訓練コーパスが 1,000 文 (1 k) しかない場合は、提案法は単独ドメインモデルに比べて非常に
高い品質となっているが、コーパス結合とはほぼ同じである。訓練コーパースサイズが増えるにしたがい、全方式ともにBLEUスコアが向上するが、コーパス結合の品質向上は単独ドメインモデルより緩やかで、約10万文（100 k）で単独ドメインモデルの品質が逆転する。提案法は、3,000文（3 k）以上では常に単独ドメインモデル、コーパス結合の品質を上回っており、両者の利点をうまく融合させた方式であることを示している。

コーパスフィルタリングは、提案方式とほぼ同様に、3万文（30 k）以上では、単独ドメインモデル、コーパス結合を有心に上回ったが、提案法を有心に上回ることはなかった。4.2節でも述べたように、コーパスフィルタリングは、最適な追加文数の決定が難しい。実際、この実験では、10万文単位での最適な追加文数を決定するため、1試行あたり10回以上の訓練、最適化テストを繰り返した。提案法の訓練回数は、個別最適化の場合、コーパス結合モデルと単独ドメインモデルの2回に固定されており、使いやすさの観点では提案法がコーパスフィルタリングより優れていると考えられる。

ここで、表3、4に戻る。表3、4では、MEDコーパスにおける提案法（同時最適化、個別最適化）の翻訳品質が英日、日英ともに向上した。この理由は、MEDコーパスが約22万文と、他のコーパスに比べて小規模で、翻訳品質向上の余地を残していたためと推測できる。一方、MED以外のドメインは、100万文規模の大規模コーパスから学習しているため、必ずしも翻訳品質が向上するわけではない。しかし、特筆したいのは、大規模コーパスに提案方式を適用しても、翻訳品質が下がることがなく、データによっては向上する場合もあるという点である。その点で、提案方式はコーパスサイズに対して非常にロバストな方式といえる。

### 4.4 未知語

本稿の提案方式の特徴は、コーパス結合モデルと単独ドメインモデルの利点を融合させたもののとまとめられる。本節では各モデルの未知語の観点から分析する。

本稿では、Irvine, Morgan, Carpuat, Daumé, and Munteanu (2013) に準じて、未知語を原言
表 6 未知語を含む文の割合（英日翻訳）

| 未知語種別 | 方式                    | ドメイン       |
|-----------|------------------------|--------------|
|           |                        | MED | LIVING | NTCIR | ASPEC |
| 原言語未知 | 単独ドメインモデル       | 9.1%| 4.1%   | 7.9%  | 22.5% |
|           | コーパス結合             | 1.0%| 2.8%   | 6.5%  | 21.6% |
|           | 提案法（個別最適化）    | 0.9%| 2.4%   | 6.4%  | 21.2% |
| 目的言語未知| 単独ドメインモデル       | 38.5%| 26.4% | 21.3% | 26.5% |
|           | コーパス結合             | 18.1%| 20.1% | 17.1% | 21.6% |
|           | 提案法（個別最適化）    | 16.1%| 17.0% | 17.2% | 20.0% |

語未知と目的言語未知に分けて考える。原言語未知は入力の単語あるいはフレーズがフレーズテーブルに存在しない場合である。目的言語未知は、原言語のフレーズは存在するが、目的言語の単語あるいはフレーズが存在していないために、参照訳が生成できない場合である。これは、強制デコーディング (Yu, Huang, Mi, and Zhao 2013) を行い、参照訳が生成できるかどうかで判定できる。

英日翻訳におけるドメイン毎の未知語率を表 6 に示す。これは、5 回の最適化のうちの 1 試行を取り出した結果である。ドメイン毎に割合の差はあるが、単独ドメインモデルに比べ、コーパス結合は原言語未知、目的言語未知ともに減少する。たとえば、MED コーパスでは、原言語未知は 9.1% → 0.9%、目的言語未知は 38.5% → 16.1% と減少し、他のドメインの単語が利用可能になっている。しかし、最終的な翻訳品質は単独ドメインモデルの方がよかったことを考えると、未知語の減少が直接品質向上に寄与したわけではない。最適化が重要であることがわかる。

提案法は、NTCIR を除き、さらに未知語が減少した。提案法はコーパス結合モデルと単独ドメインモデルのフレーズを OR 検索している。2 つのフレーズを OR で検索することで、重複する訓練コーパスを使用しているにも関わらず、訓練によって得られるフレーズは異なるため、結果的にカバレッジが向上したものと考えられる。

4.5 ドメイン適応による翻訳の変化例

統計翻訳のモデルを変更した場合、翻訳文の変化を適切に分類するのは非常に難しい。本提案のドメイン適応に関しても、正確な分類は困難だと判断したが、翻訳品質が向上している場合、典型的には以下の 3 パターンが観察された。ここでは表 7 に示す翻訳例を参照しながら説明する。この例は MED ドメインの英日翻訳から抜き出したものである。

- 単語の翻訳がドメイン依存訳に変化したものがある。例 1 の “first visit” の翻訳に関しては、コーパス結合では「初めて」と一般的な訳になっているが、単独ドメインモデルと

---

10 原言語未知は、Irvine et al. (2013) の SEEN エラー、目的言語未知は SENSE エラーに相当する。
| 番号 | 方式 | 原文／参照訳／翻訳文 |
|------|------|-----------------------|
| 1    | 原文 | What should I tell the receptionist on my first visit? |
|      | 参照詰 | 初診のとき、受付で最初に何を言えばいいんですか？ |
|      | 単独ドメインモデル | 私は初診の受付を伝えればいいのですか？ |
|      | コーパス結合 | 私は、初めての人に伝えたらいでしょうか？ |
|      | 提案法（個別最適化） | 私の初診の受付を伝えればいいのですか？ |
| 2    | 原文 | You make an appointment within the hours for Ningen Dock during the opening hours. |
|      | 参照詰 | 診療時間内で人間ドックを行っている時間に、予約を入れるのですね。 |
|      | 単独ドメインモデル | あなたは、診療時間内に人間ドックの時間内に予約をしてください。 |
|      | コーパス結合 | 開店時間に人間ドックの時間で予約してください。 |
|      | 提案法（個別最適化） | 診療時間内に、人間ドックの時間内に予約をしてください。 |
| 3    | 原文 | On the contrary, nonprescription drugs are generally effective for various symptoms. |
|      | 参照詰 | それに対して市販薬はいろいろな症状に効くものが多いんです。 |
|      | 単独ドメインモデル | 逆に、市販の薬は、一般的にはいろんな症状に効果があります。 |
|      | コーパス結合 | 逆に、市販の薬は、様々な症状に有効である。 |
|      | 提案法（個別最適化） | 逆に、市販の薬は、様々な症状に効果があります。 |
| 4    | 原文 | Of course, they use ingredients approved as foods, however, there are stuff you need to avoid eating. |
|      | 参照詰 | もちろん食品として認められた成分の材料を使ってはいますが、たくさん食べるのはさけたほうがいい材料もあります。 |
|      | 単独ドメインモデル | もちろん。食品として認可成分を使用し、ただ、食べていけないものがあります。 |
|      | コーパス結合 | もちろん、食材を使った料理には、食べるのは避けなければならないものである。 |
|      | 提案法（個別最適化） | もちろん、食品として認可されている食材を使っていましたが、食べていけないものがあります。 |
| 5    | 原文 | When the body goes into starvation state, you sometimes get an abnormal appetite. |
|      | 参照詰 | 体が飢餓状態になると、時に異常な食欲が出たりします。 |
|      | 単独ドメインモデル | 体がstarvation状態になったときは、異常な食欲が出ます。 |
|      | コーパス結合 | 体が飢餓状態になった場合には、異常は食欲が出ます。 |
|      | 提案法（個別最適化） | 体が飢餓状態になった場合には、異常な食欲が出ます。 |
| 6    | 原文 | I'm not sure where this is, but it is close to Sangenjaya station, a short walk in the direction of Shimokitazawa. |
|      | 参照詰 | わかりませんが、三軒茶屋駅から下北沢方面に歩いたところです。 |
|      | 単独ドメインモデル | この場所はわかりませんが、Sangenjaya 駅、Shimokitazawa の方向に少し歩いてください。 |
|      | コーパス結合 | どこか、わかりませんが、三軒茶屋下北沢駅の方向に歩いてすぐのところに近い。 |
|      | 提案法（個別最適化） | これはどこにあるのかわからないのですが、下北沢、三軒茶屋駅のすぐ近くです。 |
提案法では、「初診」となり、病院における対話に適応した訳となっている。同様に、例
2 では “opening hours” の訳が「診療時間」と翻訳されている。
- 例 3, 4 のように、コーパス結合では常体の翻訳文になっているが、単独ドメインモデル、
提案法では敬体の翻訳文となっている。コーパス結合モデルでは、特許や論文のコーパス
を用いているため、常体が多数派を占めているためだと考えられる。提案法は、コー
パース結合モデルを併用しているにも関わらず、MED の単独ドメインモデルを適切に利用
して、敬体の翻訳文を生成した。これも一種のドメイン依存訳であると考えられる。
- 例 5, 6 では、単独ドメインモデルでは原語語未知になる単語があった (starvation,
Sangenjaya, Shimokitazawa) が、コーパス結合、提案法ではこれが解消されている。

5 まとめ

本稿では、複数ドメインを前提とした、統計翻訳の適応方式を提案した。本稿の方式は、カ
パレッジが広い（未知語が少ない）コーパス結合モデルと、素性関数の精度がよい単独ドメイ
ンモデルを併用し、機械学習分野のドメイン適応方法である、素性空間拡張法の考え方を利用
して両者を結合した。また、empty 値をチューニング対象に追加した。

実験では同時最適化を行った場合、個別最適化を行った場合ともに、単独ドメインモデルに
比べ、翻訳品質が向上または同等を保持した。提案法は、当該ドメインの訓練コーパスが小規
模である場合に高い効果を持ち、100 万文規模の大規模コーパスを持つドメインへの適応に使
用しても、翻訳品質を下げることなく、ドメインによっては品質向上の効果がある。基本的な
対数線形モデルでも、モデルの選択とチューニングを慎重に行うことで、最先端方式と同等以
上の適応方式になることを示した。

提案方法は、対数線形モデルに基づく統計翻訳にはすべて適用可能であるので、木構造変換
のような翻訳方式でも効果を確認したいと考えている。ただし、木構造変換の翻訳方式は、構
文解析を利用しているため、翻訳品質は構文解析自身の精度にも影響を受ける。本稿で用いた
事前並び替えも、構文解析を利用している。構文解析などの翻訳に使われているコンポーネン
トについては、別途ドメイン適応させる必要がある (森下、赤部、波多野、Neubig、吉野、中村
2016) が、本稿の提案方式は、コンポーネントの適応とは独立に併用可能である。

謝 辞

本研究は総務省の情報通信技術の研究開発「グローバルコミュニケーション計画の推進—多
言語音声翻訳技術の研究開発及び社会実証—I. 多言語音声翻訳技術の研究開発」の一環として
行われました。
参考文献

Axelrod, A., He, X., and Gao, J. (2011). “Domain Adaptation via Pseudo In-Domain Data Selection.” In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, pp. 355–362, Edinburgh, Scotland, UK.

Birch, A., Osborne, M., and Koehn, P. (2007). “CCG Supertags in Factored Statistical Machine Translation.” In Proceedings of the 2nd Workshop on Statistical Machine Translation, pp. 9–16, Prague, Czech Republic.

Bisazza, A., Ruiz, N., and Federico, M. (2011). “Fill-up versus Interpolation Methods for Phrase-based SMT Adaptation.” In Proceedings of the International Workshop on Spoken Language Translation (IWSLT), San Francisco, California, USA.

Cherry, C. and Foster, G. (2012). “Batch Tuning Strategies for Statistical Machine Translation.” In Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 427–436, Montréal, Canada.

Clark, J. H., Dyer, C., Lavie, A., and Smith, N. A. (2011). “Better Hypothesis Testing for Statistical Machine Translation: Controlling for Optimizer Instability.” In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 176–181, Portland, Oregon, USA.

Clark, J. H., Lavie, A., and Dyer, C. (2012). “One System, Many Domains: Open-Domain Statistical Machine Translation via Feature Augmentation.” In Proceedings of the 10th biennial conference of the Association for Machine Translation in the Americas (AMTA 2012), San Diego, California, USA.

Daumé, III, H. (2007). “Frustratingly Easy Domain Adaptation.” In Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pp. 256–263, Prague, Czech Republic.

Denkowski, M. and Lavie, A. (2014). “Meteor Universal: Language Specific Translation Evaluation for Any Target Language.” In Proceedings of the 9th Workshop on Statistical Machine Translation, pp. 376–380, Baltimore, Maryland, USA.

Finch, A. and Sumita, E. (2008). “Dynamic Model Interpolation for Statistical Machine Translation.” In Proceedings of the 3rd Workshop on Statistical Machine Translation, pp. 208–215, Columbus, Ohio, USA.

Foster, G., Goutte, C., and Kuhn, R. (2010). “Discriminative Instance Weighting for Domain Adaptation in Statistical Machine Translation.” In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pp. 451–459, Cambridge, Massachusetts,
USA.

Foster, G. and Kuhn, R. (2007). “Mixture-Model Adaptation for SMT.” In Proceedings of the 2nd Workshop on Statistical Machine Translation, pp. 128–135, Prague, Czech Republic.

Goto, I., Utiyama, M., Sumita, E., and Kurohashi, S. (2015). “Preordering Using a Target-Language Parser via Cross-Language Syntactic Projection for Statistical Machine Translation.” ACM Transactions on Asian and Low-Resource Language Information Processing, 14(3), pp. 13:1–13:23.

Heafield, K., Pouzyrevsky, I., Clark, J. H., and Koehn, P. (2013). “Scalable Modified Kneser-Ney Language Model Estimation.” In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 690–696, Sofia, Bulgaria.

Hopkins, M. and May, J. (2011). “Tuning as Ranking.” In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, pp. 1352–1362, Edinburgh, Scotland, UK.

Irvine, A., Morgan, J., Carpuat, M., Daumé, III, H., and Munteanu, D. (2013). “Measuring Machine Translation Errors in New Domains.” Transactions of the Association for Computational Linguistics, 1, pp. 429–440.

Isozaki, H., Hirao, T., Duh, K., Sudoh, K., and Tsukada, H. (2010). “Automatic Evaluation of Translation Quality for Distant Language Pairs.” In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pp. 944–952, Cambridge, Massachusetts, USA.

Jeblee, S., Feely, W., Bouamor, H., Lavie, A., Habash, N., and Oflazer, K. (2014). “Domain and Dialect Adaptation for Machine Translation into Egyptian Arabic.” In Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP), pp. 196–206, Doha, Qatar.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). “Moses: Open Source Toolkit for Statistical Machine Translation.” In Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions, pp. 177–180, Prague, Czech Republic.

Koehn, P., Och, F. J., and Marcu, D. (2003). “Statistical Phrase-Based Translation.” In HLT-NAACL 2003: Main Proceedings, pp. 127–133, Edmonton, Alberta, Canada.

Koehn, P. and Schroeder, J. (2007). “Experiments in Domain Adaptation for Statistical Machine Translation.” In Proceedings of the 2nd Workshop on Statistical Machine Translation, pp. 224–227, Prague, Czech Republic.
Kudo, T., Yamamoto, K., and Matsumoto, Y. (2004). “Applying Conditional Random Fields to Japanese Morphological Analysis.” In Lin, D. and Wu, D. (Eds.), Proceedings of EMNLP 2004, pp. 230–237, Barcelona, Spain.

Matsoukas, S., Rosti, A.-V. I., and Zhang, B. (2009). “Discriminative Corpus Weight Estimation for Machine Translation.” In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, pp. 708–717, Singapore.

Nakazawa, T., Yaguchi, M., Uchimoto, K., Utiyama, M., Sumita, E., Kurohashi, S., and Isahara, H. (2016). “ASEPC: Asian Scientific Paper Excerpt Corpus.” In Proceedings of the 10th Edition of the Language Resources and Evaluation Conference (LREC-2016), Portoroz, Slovenia.

Och, F. J. (2003). “Minimum Error Rate Training in Statistical Machine Translation.” In Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics, pp. 160–167, Sapporo, Japan.

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 (ACL), pp. 311–318, Philadelphia, Pennsylvania, USA.

Sennrich, R. (2012). “Perplexity Minimization for Translation Model Domain Adaptation in Statistical Machine Translation.” In Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, pp. 539–549, Avignon, France.

Sennrich, R., Schwenk, H., and Aransa, W. (2013). “A Multi-Domain Translation Model Framework for Statistical Machine Translation.” In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 832–840, Sofia, Bulgaria.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., and Makhoul, J. (2006). “A Study of Translation Edit Rate with Targeted Human Annotation.” In Proceedings of the 7th Biennial Conference of the Association for Machine Translation in the Americas (AMTA-2006), pp. 223–231, Cambridge, Massachusetts, USA.

Yu, H., Huang, L., Mi, H., and Zhao, K. (2013). “Max-Violation Perceptron and Forced Decoding for Scalable MT Training.” In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp. 1112–1123, Seattle, Washington, USA.
略歴

今村 賢治：1985 年千葉大学工学部電気工学科卒業。同年～2014 年日本電信電話株式会社。1995 年～1998 年NTT ソフトウェア株式会社。2000 年～2006 年 ATR 音声言語コミュニケーション研究所。2014 年より（株）ATR-Trek 所属として、国立研究開発法人情報通信研究機構 (NICT) へ出向。現在 NICT 先進的音声翻訳研究開発推進センター主任研究員。主として自然言語処理技術の研究・開発に従事。博士（工学）、電子情報通信学会、情報処理学会、言語処理学会、ACL 各会員。

隅田英一郎：1982 年電気通信大学大学院修士課程修了。1999 年京都大学大学院博士（工学）取得。1982 年～1991 年（株）日本アイ・ビー・エム東京基礎研究所研究員。1992 年～2009 年国際電気通信基礎技術研究所研究員。主幹研究員。室長。2007 年～現在 国立研究開発法人情報通信研究機構 (NICT) 先進的音声翻訳研究開発推進センター (ASTREC) 副センター長。2016 年 NICT フェロー。機械翻訳の研究に従事。

（2017 年 1 月 16 日 受付）
（2017 年 4 月 9 日 再受付）
（2017 年 5 月 18 日 採録）