COLING2012 参加報告（その３）
– 木構造に基づく機械翻訳 –

中澤 敏明[†]

1　はじめに

本稿では COLING2012 で発表された機械翻訳に関する論文の中でも、特に木構造に基づく機械翻訳 (Tree-based Machine Translation) の論文を 2 つ紹介する。木構造に基づく機械翻訳というアイデア自体は新しいものではないが、近年の機械翻訳研究は文の構造的な情報を用いる方向に移行してきており、リッチな情報をいかに効率的・効果的に用いるかという点に注目が集まっている。紹介する論文は 3 章で説明する “Unsupervised Discriminative Induction of Synchronous Grammar for Machine Translation” (Xiao, Xiong, Liu, Liu, and Lin 2012) と 4 章で説明する “Tree-based Translation without Using Parse Trees” (Zhai, Zhang, Zhou, and Zong 2012) の 2 つである。これらの論文を紹介する前に、機械翻訳に馴染みのない読者のために、まずは次章で句に基づく機械翻訳および木構造に基づく機械翻訳について簡単に説明する。機械翻訳研究に携わっておられる方は、3 章から読み進めていただいて差し支えない。

2　句および木構造に基づく機械翻訳の簡単なおさらい

コーパスベースの機械翻訳手法の中で現在最も良く知られ、また広く用いられているのは、句に基づく機械翻訳 (Phrase-based Statistical Machine Translation: PSMT)(Koehn, Och, and Marcu 2003) であろう。図 1 に PSMT の概要を示す。PSMT では任意長の単語 n-gram を句 (phrase) という単位でまとめて扱っており、句単位での翻訳を行う。入力文の句への分割の仕方、各句の訳し方、翻訳後の句の順序には様々な可能性があるが、PSMT では全ての可能性の中から最良と思われる訳文を作り出す[1]。訳文の良さの定義には句の翻訳確率や句の並べ替えの確率、目的言語の言語モデルなど様々なものが考えらえる。PSMT ではこれら一つ一つを素性として取り入れ、対数線形モデル (log-linear model) により翻訳を実現している。

[†]京都大学, Kyoto University

[1]全探索は不可能なので、様々な方法を駆使して効率的に近似解を求めている。詳細は (Koehn 2010) などを参照。

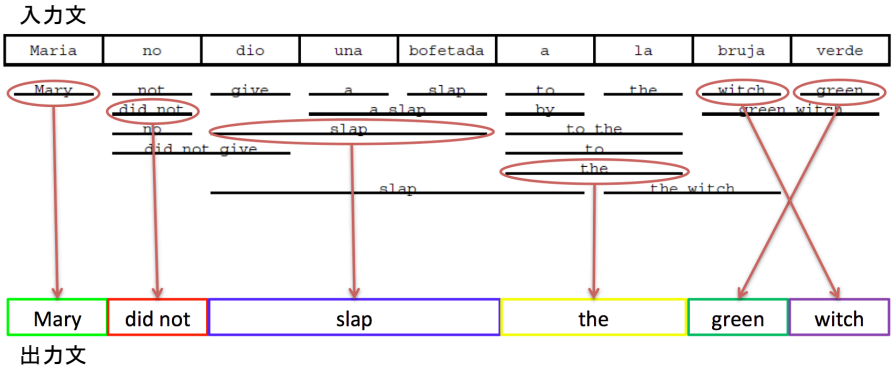

図 1　句に基づく機械翻訳

PSMT のように文を単語列として扱う手法は、英語 ↔ フランス語翻訳のように語彙や語順などが似た言語対ではかなりの精度で翻訳を行う事が可能で、すでに様々な場面で実用化されている。一方で、英語 ↔ 日本語など翻訳時に大きな語順の変更が伴うなど、言語的に遠い言語対における同手法の精度は受け入れがたいものである。試しにフランス語のニュースサイトを Google 翻訳などで英語、日本語それぞれに訳してみれば、その精度の違いは一目瞭然であろう[2]。このような状況から、単語列に基づく方法に限界を感じた研究者らは、文の構造的な情報を利用し始める。その一つが、木構造に基づく機械翻訳である。

木構造と一口に言っても様々な表現があるが、ここでは文脈自由文法 (Context-free Grammar: CFG) によって表現される木構造を考える。単言語の構文解析などで使われる CFG とは若干異なり、翻訳での CFG は原言語と目的言語の 2 言語を同時に考慮する必要があるため、同期的文脈自由文法 (Synchronous CFG: SCFG) と呼ばれる。つまり非終端記号の書き換え規則の右側に、2 つの言語間の対応関係が含まれているのである。なお、実際には規則の右側に部分木を含む木置換文法 (Tree Substitution Grammar: TSG) が使われる事が多いが、部分木の内部構造を無視して葉のノードのみに注目すれば、CFG と同様に計算できる。

さて、SCFG に基づいた機械翻訳において、非終端記号をたった 2 種類 (S と X、S は CFG と同様に文全体を表す) だけを用いるのが階層的な句に基づく機械翻訳 (Hierarchical PSMT: HPSMT)(Chiang 2005) である。HPSMT は PSMT の拡張であり、句の中に非終端記号を含める事ができる。この非終端記号は、大きな句のペアに含まれる小さな句のペアを取り除くことで生成される。これらは単語列上で行われる操作であり、構文解析器などは用いていない。図 2 の左側に HPSMT の概要を示す。3 章で紹介する論文は、この HPSMT に関する論文である。

これに対して、非終端記号のラベルに句構造解析のカテゴリーなどを使う方法が統語情報に基づく機械翻訳 (Syntax-based SMT: SSMT)(Galley, Hopkins, Knight, and Marcu 2004) である。

[2]Google 翻訳が PSMT を用いているかどうかは定かではないが、参考にはなるであろう。

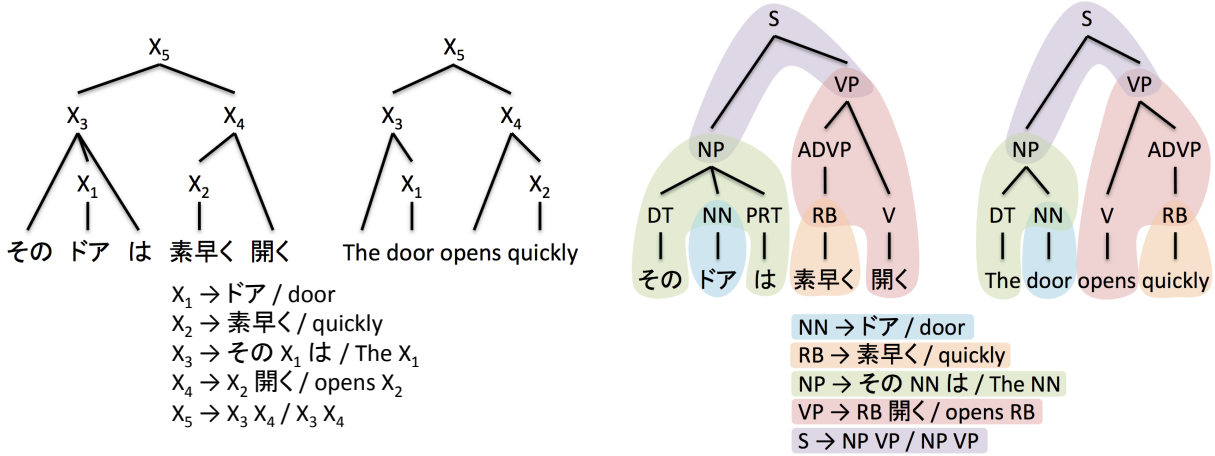

図 2　左：階層的な句に基づく機械翻訳 (HPSMT)、右：統語情報に基づく機械翻訳 (SSMT)。上部に木構造を、下部に適用されたルールを示す。HPSMT の例では全ての X を区別しているが、実際には X_5 のルールのように、ルールの右側に複数の X があるときのみその区別をつけ、それ以外の X に区別はない。また S は省略している。SSMT の例では非終端記号のラベルは説明の都合上付与してある。

図 2 の右側に SSMT の概要を示す。SSMT では構文解析結果を利用している点で HPSMT とは異なる。また図 2 の例では両言語ともに構文情報を利用しているが、多くの場合はどちらか一方の言語のみで構文情報が利用される事が多い。4 章で紹介する論文は、この SSMT に関する論文である。

3　Unsupervised Discriminative Induction of Synchronous Grammar for Machine Translation

同期文法、つまり HPSMT の翻訳ルールを教師なしで識別モデルにより学習するという内容である (Xiao et al. 2012)。多くのコーパスベース機械翻訳システムでは、まず与えられた対訳コーパスに単語アライメントを自動で付与し、アライメント済み対訳コーパスから翻訳ルールの抽出を行うという 2 つのステップに分かれている。この方法の欠点としては、アライメントモデルと翻訳モデルの関連性が薄い、単語アライメント誤りが翻訳に伝播する、翻訳ルール抽出などに様々なヒューリスティクスが入っており理論的基盤が弱い、などが挙げられる。これを克服するために、統一的なモデルで対訳コーパスから直接翻訳ルールを学習しようという研究が多くなされた。しかしそれらは全て生成モデルを用いており、様々な有効な素性を考慮する事が難しいため、識別モデルによる教師なし学習を行ったというのが、この論文の貢献である。

モデル自体はよくある log-linear モデルを用いており、確率的勾配降下法で MAP 推定により

翻訳ルールを学習している。素性として用いたのは各ルールの対訳コーパス中の出現回数、ルール内部の単語対応、目的言語側の単語数および原言語側の句境界である。ここでの句とは、ある非終端記号を根とする部分木がカバーする原言語側の単語列 (スパンとも呼ばれる) であり、句境界素性として句の先頭のバイグラムとその一つ前の単語、および句の末尾のバイグラムとその一つ後ろの単語を用いている。この素性により、より意味的にまとまりのある句が作られていくようだ。

実験では NIST 翻訳評価ワークショップのデータ (MT03-05) を使って、中 → 英翻訳での翻訳精度がベースラインである 2 ステップ学習による HPSMT と比べて向上したことを BLEU 値で示している。また Oracle の精度 (作り出す事が可能な全ての翻訳の中で、正解と最も近いものだけを選んで測定した精度) も大きく向上しており、より翻訳に適したルール抽出が行えているようである。また識別モデルであるため、他の素性が入れやすいのも評価できる点である。例えば単語の品詞を使ったり、チャンキングして境界情報を与えるなど、いろいろ考えられる。

4　Tree-based Translation without Using Parse Trees

構文解析器を用いずに、木構造 (統語情報) に基づく機械翻訳を行おうという内容である (Zhai et al. 2012)。SSMT では基本的には構文解析器が必要であるが、どの言語でも用意できるわけではない。また構文解析と単語アライメントは独立に行われるため、お互いの結果の親和性が低く、抽出できる翻訳ルールが少なくなったり、不適切な翻訳ルールが抽出されるなどの問題がある。そこで、構文解析器は使わず、与えられた単語アライメントに親和的な構文木を作り出すという方法を提案している。構文木の作り方は一意ではないため、多くの構文木を詰め込んだ表現である構文森 (packed forest) を利用し、構文森から同期的木置換文法を EM アルゴリズムにより学習する。なお提案しているモデルは目的言語側のみ構文木を作り出す、String-to-Tree の翻訳モデルであり、簡単のため各構文木は二分木としている。

非終端記号のラベルには、スパンが 1 単語の場合は各単語の品詞を、スパンが 2 単語の場合は各単語の品詞を組み合わせたものを、スパンが 3 単語以上の場合はスパンの先頭及び末尾の単語の品詞を組み合わせたものをそれぞれ用いる。例えば “we(PRP) meet(VBP) again(RB)” の 3 単語を考えると、“meet again” をカバーする非終端記号のラベルは “VBP+RB” となり、3 単語全体をカバーする非終端記号のラベルは “PRP...RB” となる。

構文森を構築する際に全てのスパンを考慮すると、構文森が巨大になり計算量が爆発してしまうため、1. Bilingual sentence segmentation と 2. Frontier node assumption という 2 つの方法で探索空間を削減している。前者は、対訳文を句読点などをたよりに分割し、各部分で構文森を構築して、最後に森同士を組み合わせるという方法である。これにより各部分をまたぐようなスパンは考慮されなくなる。後者はあるスパンに対する森を構築するときは、フロンティ

アノードが最も多い木だけを保存するという方法である。フロンティアノードの定義は (Galley et al. 2004) などを参照していただきたいが、簡単に言うと、カバーするスパンが単語アライメントと整合的な非終端記号のことを言う。つまりフロンティアノードが多い木というのはそれだけ単語アライメントと親和的であり、適切な翻訳ルールが多く抽出できる可能性があるということになる。

実験では同じく NIST 翻訳評価ワークショップのデータ (MT04-05) を使って、中 → 英翻訳での翻訳精度向上を BLEU 値で示している。String-to-Tree モデルなので、英語側だけ構文木を作り出している。HPSMT や構文解析器を使うノーマルな SSMT などと比較しても、提案手法の方が精度が良かったようだ。

構文解析器を使わずに構文木を作り出した方が精度が良いというのは面白いが、これは必ずしも構文解析器の精度が悪いのが原因であるとか、構文解析は不要であるという結論には至らないということに注意が必要である。正しい構文解析結果を使っても、単語アライメントとの相性が良くない、もしくは単語アライメントの精度が低いために翻訳精度が上がらない可能性が十分にあり、構文情報を取り込んだアライメントモデルを使うなどすれば、ノーマルな SSMT の方が精度が高くなると思われる。一方で、動機付けの一つとなっているように、構文解析器のない言語でも適用可能であり、その精度が構文解析を使った場合と遜色ないという結果は評価できる。

5 まとめ

本稿で紹介した論文の他には対訳語彙獲得が 3 件、述語項構造を使った翻訳が 2 件、パラメータチューニングが 2 件のほか、翻訳精度推定、語順などに関する発表があった。またここでは詳細は説明しないが、我々のグループからは機能語のアライメント誤りに注目した、単語依存構造木上でのアライメント手法を提案した (Nakazawa and Kurohashi 2012)。興味のある方はぜひ目を通していただきたい。

COLING2012 では機械翻訳分野のブレークスルー的な論文は残念ながら見当たらなかった。他の多くの機械翻訳論文と同様、今回紹介した論文も評価実験で BLEU 値の多少の向上を報告しているが、よく言われるように、BLEU 値の向上が実際にどの程度の訳質向上なのかは不透明である。提案した手法が翻訳の改善にどのように貢献したのかなど、実例を挙げている論文は極めて少ない。

ここからは個人的な所感なのだが、研究的機械翻訳と実用的機械翻訳との間に大きなギャップがあるように思われる。研究的機械翻訳を否定するわけではなく、もちろん積み重ねが重要なのだが、どうも最近の機械翻訳研究は積み重ねではなく敷き詰めている感じがするものが多い。例えば今回紹介した 1 つ目の論文も、生成モデルとの比較はなされておらず、使い古され

たベースラインである HPSMT との比較に留まっている。ひょっとしたら生成モデルの方が精度が高いかもしれないのである。

機械翻訳を社会に役立てるという目標に向けては、もっと別の視点からの洞察も必要ではないか (自戒の意味も込めて)。BLEU 値だけをみるのではなく、実際の訳文をもっと時間をかけて分析したり、機械翻訳ユーザーのニーズは何なのか、使ってもらうためにはどうしたらいいのかなどを考える必要があると思う。その上で、機械翻訳の本質的な問題を解決する手法を提案するべきである。東北観光博[3]のような問題を繰り返さないためにも。

参考文献

Chiang, D. (2005). "A Hierarchical Phrase-Based Model for Statistical Machine Translation." In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pp. 263–270.

Galley, M., Hopkins, M., Knight, K., and Marcu, D. (2004). "What's in a translation rule?." In Susan Dumais, D. M. and Roukos, S. (Eds.), *HLT-NAACL 2004: Main Proceedings*, pp. 273–280 Boston, Massachusetts, USA. Association for Computational Linguistics.

Koehn, P. (2010). *Statistical Machine Translation* (1st edition). Cambridge University Press, New York, NY, USA.

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

Nakazawa, T. and Kurohashi, S. (2012). "Alignment by Bilingual Generation and Monolingual Derivation." In *Proceedings of COLING 2012*, pp. 1963–1978 Mumbai, India. The COLING 2012 Organizing Committee.

Xiao, X., Xiong, D., Liu, Y., Liu, Q., and Lin, S. (2012). "Unsupervised Discriminative Induction of Synchronous Grammar for Machine Translation." In *Proceedings of COLING 2012*, pp. 2883–2898 Mumbai, India. The COLING 2012 Organizing Committee.

Zhai, F., Zhang, J., Zhou, Y., and Zong, C. (2012). "Tree-based Translation without using Parse Trees." In *Proceedings of COLING 2012*, pp. 3037–3054 Mumbai, India. The COLING 2012 Organizing Committee.

略歴

中澤 敏明（正会員）：

[3]「東北観光博 誤訳」で検索

2010 年京都大学大学院情報学研究科知能情報学専攻博士後期課程修了。博士（情報学）。機械翻訳の研究に従事。

（2012 年 11 月 30 日依頼）
（2013 年 1 月 21 日受付）