Transliterationの拡張としてのWikipediaからの意味的翻訳対の抽出

原田 泰佑† Andrew Finch‡ 田中 久美子† 隅田 英一郎‡
九州大学 システム情報科学府 † 独立行政法人 情報通信研究機構 ‡

{tai_harada, kumiko}@cl.ait.kyushu-u.ac.jp†
{andrew.finch,eiichiro.sumita}@nict.go.jp‡

1 はじめに

統計的機械翻訳では原文と対訳文の集合から成るパラレルコーパスから、文部分の翻訳対を大量に得て、そこから学習される統計的モデルに基づいて翻訳を行う。そのため、統計的翻訳の性能向上には、大量の翻訳対が必要になる。

翻訳対の中には音声的な対応をもつものがあり、そのような翻訳対をtransliterationと呼ぶ。transliterationの翻訳対の抽出は、音声的に対応する対としてのデータをもとに、文字対応を得ることにある。当然、抽出できるものは音声的な対応をもつ翻訳対のみで、例えば「九州大学」と「Kyushu University」のように意味的な対応の部分を含む翻訳対の抽出はその目的とされていなかった。

本稿では、transliterationの抽出方法を拡張させることで、音声的な対応と意味的な対応の両方を同時に得ることを目指した。この拡張により、先程の例の「九州大学」と「Kyushu University」のように音声的な対応をもつ部分と意味的な対応をもつ部分が混在する翻訳対から、「九州-Kyshu」「大学-University」のように単語同士の対応を一挙に得ることを目指す。

具体的には、確率的ブロック編集距離 [1] を利用することで、Wikipedia[1] の日本語と英語の題のペアから単語の翻訳対の抽出を行った。その際、ペアを日本語の文字の種類と英語の単語数によって分類し、予想される単語同士の対応の取りやすい順に抽出を行った。また、以前の抽出で得られた単語の翻訳対を次の抽出に利用した。その結果、transliterationの部分とそうでない部分が混在する句における単語の対応をとることができ、約10万対の単語の翻訳対を得ることができた。

[1] https://www.wikipedia.org/

2 関連研究

transliteration miningの研究として文献 [2] がある。この研究では、Wikipediaの題から音声的な翻訳対の抽出を行っており、その対象となる句の対の数が多いこと、句に含まれる単語の文字列の長さが短いこと等の性質において共通する点がある。しかし、文字の対応をとる手法として隠れマルコフモデルを利用しており、Bayes推定を基礎とする確率的ブロック編集距離を利用している点で本稿とは異なる。

Bayes推定を利用したtransliteration miningの研究としては文献 [3] がある。しかし、例えば「チ-Ch ョ-o ム-m ス-s キ-k ー-y」のように、対応する文字数が片方の言語で複数文字、もう一方で1文字単位としている。本稿では、これを「チョ-Cho ム-m ス-s キー-ky」のように両言語で複数文字同士の対応が得られるよう拡張した確率的ブロック編集距離を利用した。

同じくtransliteration miningの研究として文献 [4] がある。この研究では、本稿と同様に確率的ブロック編集距離を利用しtransliterationの文字列の対応を得ている。しかし、文献 [4] ではtransliterationのみの抽出を行っており、本稿ではtransliterationのみでなく、音声の対応のない意味的な翻訳対も含むデータを同時に抽出を行う点で異なる。

3 Transliteration miningの拡張

本稿ではtransliteration miningの手法を拡張することで意味的な翻訳対も獲得することを目指す。transliteration miningでは2章で述べたように確率的ブロック編集距離を利用することでtransliterationの文字列の対応をすでにとっている。本稿では必ずしもtransliterationではない文字列も学習データに加える。そして、transliteraionの部分とそうでない部分で区別せずに対応を取ることを目指している。

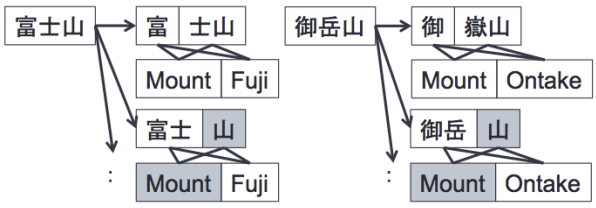

図 1: Transliteration mining の拡張としての意味的翻訳の抽出の一例

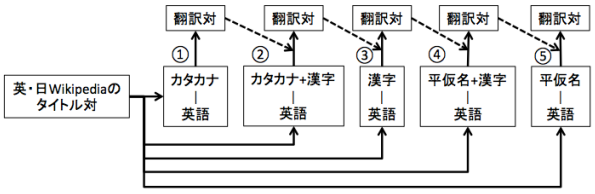

図 2: Wikipedia のタイトル対からの抽出の流れ

具体的には、複数の意味的に対応するペアを与え、まず各ペアの中の transliteration となっている部分の対応を確定させる。その上で、残った文字列部分の対応する可能性を、統計的に吟味する。たとえば、図 1 のように、「富士山」と「御嶽山」の二つの対応ペアを考える。まず、各ペアのそれぞれの音声的な対応としての「富士」と「御岳」の部分の対応が確定する。残った部分として「山」と「Mount」同士が対応が考えられるが、この二つのペア（また他の事例）があることから、この意味的な対応がとれるというわけである。

4 確率的ブロック編集距離

本稿では翻訳対の文字の対応を得る手法として文献 [1] の確率的ブロック編集距離を利用した。これは文献 [5] における Bayes 推定を用いた手法を転用したものである。

編集距離は 2 つの文字列がどの程度異なっているかを示し、具体的にはある文字列を別の文字列に変形する時に必要な編集操作 (挿入・削除・置換) の最小回数、または最小コストで表される。また、通常は編集操作は主に 1 文字単位を対象とするものであった。

これに対して、ブロック編集距離では、編集操作の対象は複数文字を含むブロック単位であり、文字列を 1 つの単位として編集操作を行うことができる。このブロック編集操作については文献 [6] で初めて触れられているが、具体的な編集操作のコストの求め方についての言及はなされていなかった。

編集操作のコストの推定においては EM アルゴリズムを利用したものが考えられるが、過学習が起きてしまう可能性がある。そのため、文献 [1] では文献 [5] での文字列の対応を Bayes 推定によって求める方法を転用し、コスト推定を実現している。

具体的には、入力された文字列ペア $X = x_1^m = x_1, x_2, ..., x_m$、$Y = y_1^n = y_1, y_2, ..., y_n$ に対して、式 (1) で確率的ブロック編集距離を再帰的に定義している。

$$d(x_1^u, y_1^v) = \sum_{i=0}^{u} \sum_{j=0}^{v} (c((x_{u-i+1}^u, y_{v-j+1}^v) + d(x_1^{u-i}, y_1^{v-j}))) \quad (1)$$

この式における c はコスト関数であり、編集操作が入力された時に、そのコストとして確率の対数の負の値を返す。m, n はそれぞれ文字列 X, Y の長さを示し、$1 \leq u \leq m, 1 \leq v \leq n$ である。また、$x_{u+1}^u := \epsilon, y_{v+1}^v := \epsilon, d(\epsilon, \epsilon) := 0$ であり、$d(\epsilon, \epsilon) := 0$ により再帰式は終了する。

5 Wikipedia からの翻訳対の抽出

本節では 4 節で述べた確率的ブロック編集距離を利用し翻訳対の抽出を行い、得られた翻訳対について報告する。

5.1 抽出の概要

本稿で利用する Wikipedia の日本語の題においては平仮名やカタカナ、漢字等の様々な文字が含まれる。日本語と英語の文字の対応を考えると、音声・意味を扱う上では日本語の文字の種類を利用できる。例えば、カタカナは主に外来語であるため、transliteration が多く含まれる。そのため、カタカナを含む翻訳対のみで文字の対応をとると、より音声的な対応を取ることができる。

そのため、今回の翻訳対の抽出はすべてのデータに対して一度に行うのではなく、入力するペアの日本語の文字の種類、英語の句に含まれる単語数によってデータを分け、その分類ごとに抽出が易しいと予想される順、図 2 の番号順に行った。この図 2 で、まず日本語と英語の Wikipedia の題の対からカタカナと英語の対を抜き出し、翻訳の関係になっている単語のペアの抽出を行う。そして、Wikipedia の題の対からカタカナと漢字を含む日本語の句と英語の句の対を抜き出し、事前データとして先ほど得たカタカナと英語の単語の翻訳対を与え、抽出を行う。それ以降も同様の手順で行っていった。

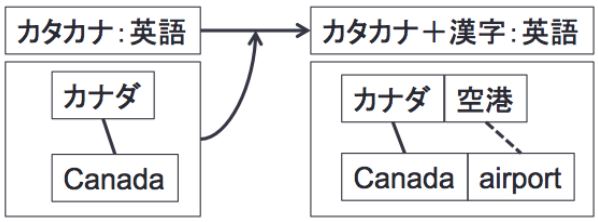

図 3: 事前データの利用

これにより、図 3 の例のようにカタカナと英語の翻訳対の抽出で得られた「カナダ-Canada」を次のカタカナと漢字が含まれる日本語と英語の翻訳対の抽出で利用することで「カナダ空港-Canada airport」の「カナダ-Canada」の対応の確率を事前に上げることができる。それによって、残りの「空港-airport」の対応の確率を上げることができ、より正確な対応が取ることができると予想される。

5.2 Wikipedia のデータ

Wikipedia の日本語と英語の題で対応するものを抽出して利用する。合計で 53 万対の題のペアが得られた。さらに、図 2 のように日本語の種類（例えば平仮名、カタカナ等）によって分類を行った。英語については日本語と対応が得られやすくするため、日本語と英語の句に含まれる単語数を揃えることによって分類を行った。結果として得られた約 12 万対を実験データとして利用した。実際に利用した各データの説明と Wikipedia の題のペア数について表 1 にまとめる。データの内実については以下に説明する。

表 1: 各データの題のペア数

データ	題のペア数
カタカナ：英語	49,364
漢字, カタカナ：英語	10,686
漢字：英語	54,309
平仮名：英語	963
平仮名, 漢字：英語	2,317
	計 117,639

5.2.1 カタカナ

カタカナと英単語の翻訳対のデータにおいては、英語の句に含まれる単語の数によって分類を行った。カタカナの中でも 1 つの句を複数の単語に分けれるもの（例えば「マイケル・ジャクソン」等）においては、更に分けて抽出を行った（図 2-①）。

5.2.2 漢字

漢字と英語の翻訳対の抽出においては、まず、事前データとしてカタカナと英語の翻訳対のデータを利用し、漢字とカタカナを含む語（例えば「筆ペン」等）と英単語の対応を取り、翻訳対を抽出した（図 2-②）。その後、それまでに得た漢字と英語の翻訳対を利用し、漢字のみからなる日本語の句と英語の句の翻訳対の抽出をおこなった（図 2-③）。

5.2.3 平仮名

平仮名と英語の翻訳対の抽出においては、漢字と英語の際と同様に、まず、漢字と平仮名を含む語（例えば「冬休み」等）と英語の対応をとった（図 2-④）。この際、事前データとして漢字と英語の翻訳対のデータを利用した。その後、それまでに得た平仮名と英語の翻訳対を利用して平仮名のみからなる日本語の句と英語の句の翻訳対の抽出を行った（図 2-⑤）。

6 結果

6.1 抽出された翻訳対

5.2 節での各データから翻訳対の抽出を行った。また、実際に得られたものが正確な翻訳対であるかどうかを確認するために、得られた翻訳対から 1000 セットを無作為に選び文字の対応の精度 p を式 (2) から求めた。この式 (2) において、c は正確に対応が取れていた単語の翻訳対の数を表す。

$$p = \frac{c}{1000} \times 100 \tag{2}$$

得られた単語の翻訳対の数とその精度を表 2 にまとめる。結果、合計で約 10 万対の日本語と英語の単語の翻訳対を抽出することができた。そして、各データの翻訳対から 1,000 対を無作為に選択し、対応関係の正確さを調べたところ、平均で 92.4%の翻訳対が正しく対応付けられていた。さらに、全データに対して意味的な対応が含まれている割合を調べるために、得られた全翻訳対から 1000 対を無作為に抽出した。このうち正しく対応関係が取られている翻訳対の中から、意味的な翻訳対を含む割合を計算した。その結果、全体の約 40%に意味的な対応関係が含まれていることが確認された。

表 2: 得られた単語の翻訳対数と精度

データ	単語の翻訳対数	精度 (%)
カタカナ：英語	31,407	98.1
漢字, カタカナ：英語	12,481	92.4
漢字：英語	53,277	91.2
平仮名：英語	988	97.3
平仮名, 漢字：英語	1,221	82.4
	計 99,374	平均 92.4

6.2 得られた翻訳対の一例

得られた結果の中で正しく対応がとれていた例を図 4 で示す。この例において、「ボイデン-boyden」「カゲラ-kagera」の部分は音声的な対応であり、「天文台-observatory」「川-river」は意味的な対応である。これは、「ボイデン-boyden」「カゲラ-kagera」の音声的な部分の対応が強くとれているために、「天文台-observatory」「川-river」の意味的な対応も正確にとれていると考えられる。

図 4: 正確な対応がとれた例

また、得られた単語の翻訳対で誤った対応が起きた例を図 5 で示す。この例において本来なら「ザカスピ-⟨NULL⟩ ⟨NULL⟩-transcaspian 州-oblast」となるべきところ、「ザ-trans カ-ca スピ-spi ⟨NULL⟩-an 州-oblast」となっている。これは、「カ-ca スピ-spi 」の部分が他のタイトルのペアでも多く得られたことで強く対応したため、「ザ-trans」が対応してしまったと考えられる。このように、文字列の一部が対応してしまうことで正確でない翻訳対が取れてしまう場合があった。

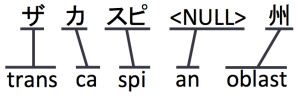

図 5: 誤った対応がとれた例

「平仮名, 漢字-英語」の対応をとった際に、「カタカナ-英語」や「漢字-英語」などの他の文字の組み合わせに比べて対応の精度が下がった。この原因として、誤った翻訳対として得られた一例の「森のようちえん-forest kindergarten」のように「の」などの助詞が入ることで、単語の切れ目が曖昧になってしまっている事が考えられる。また、他の文字の組み合わせに比べて「平仮名, 漢字-英語」のセット数は少なく、文字の対応の十分な学習ができていなかったことも原因の一つと考えられる。

7 まとめ

本稿では確率的ブロック編集距離を、音声的な対応を含む対だけでなく意味的な対応を含む対に適用することで、日本語と英語の Wikipedia の題の対から音声的な対応と意味的な対応を同時に抽出することを行った。その結果、日本語と英語の題のペア約 12 万対から約 10 万対の単語の翻訳対を抽出することができた。また、得られた単語の翻訳対に対して正しく対応関係がとれているかを評価したところ、平均で 92.4%が正しく対応関係がとれていることが分かった。さらに、意味的な対応関係が含まれている割合を調べたところ、全体の約 40%が意味的な対応関係であることが確認できた。

今後は日本語と英語だけでなく、ロシア語や中国語等の他の言語においてもこの手法で翻訳対の抽出が行えるか試みていく必要があると考える。その際には、今回の日本語のように文字の種類によっての分類ができないため、単語の品詞によって分類を行うなどの新たな工夫を考える必要がある。

参考文献

[1] 中谷 洸樹 , Andrew Finch , 田中 久美子 , 隅田 英一郎, “確率的ブロック編集距離” 言語処理学会第 20 回年次大会, March 2014.

[2] Nabende, Peter. Mining Transliterations from Wikipedia Using Pair HMMs. Proceedings of the 2010 Named Entities Workshop. pp. 76-80, 2010.

[3] Hassan Sajjad, Alexander Fraser, and Helmut Schmid. A Statistical Model for Unsupervised and Semi-supervised Transliteration Mining. Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 469-477, 2014.

[4] Takaaki Fukunishi, Andrew Finch, Eiichiro Sumita,and Seiichi Yamamoto. A Bayesian Alignment Approach to Transliteration Mining , ACM Transactions on Asian Language Information Processing (TALIP), 2013.

[5] Takaaki Fukunishi, Andrew Finch, Seiichi Yamamoto,and Eiichiro Sumita. A bayesian alignment approach to transliteration mining. Vol. 12, No. 3, pp. 9:1-9:22, August 2013.

[6] Daniel Lopresti and Andrew Tomkins. Block edit models for approximate string matching. Theoretical Computer Science, Vol. 181, No. 1, pp. 159-179, 1997.