# 漸進的対応付けによる 対訳テキストからの翻訳表現の抽出

米沢恵司　　松本裕治
奈良先端科学技術大学院大学情報科学研究科

## 1 はじめに

我々は対訳テキストから翻訳表現を抽出する新しい方法を提案する。

まずあらゆる単語列のペアについて類似度を計算し、類似度付きの辞書を構築する。次に、この辞書を参照してコーパス中の単語列を対応付け、対応付けの回数によって類似度を再計算し、辞書を整形する。この処理を繰り返すことにより、辞書の誤ったエントリを取り除き、精度の高い辞書を得ることができる。

取引条件表現法辞典 [7] から辞書を構築する実験を行ない、さらに辞書の質を客観的に判断するために、機械的評価を行なった。その結果、過去の方法と比較してより質の高い辞書を構築できることを確認した。

## 2 過去の研究

対訳テキストから翻訳表現を抽出する研究は多数ある。

翻訳辞書の獲得は概して単語間の類似度 (距離) を計算し、類似度の高い (距離の小さい) 対を取り出すことによって得られる。単語間の類似度 (距離) は通常それぞれの語の出現頻度、二つの語が対訳文に同時に出現する頻度 (共起頻度) を数え上げることによって計算される。

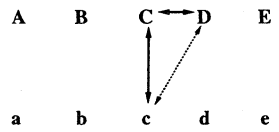
単純に類似度の高い対をすべて抽出したのでは、直接関係のない対まで翻訳対として抽出される。(図1)。これを避けるために、多くの研究では以下のどちらかの方法を採っている。

- 類似度の高い対から順次抽出して行き、それらを抽出の対象から除く。
- 類似度を再計算する処理を繰り返すことによって、徐々に翻訳辞書を整形して行く。

北村ら [8] は閾値を段階的に下げ、閾値を越える単語列のペアを順次抽出した。この手法により、ビジネス文書、科学論文等において高い精度で翻訳表現を抽出することに成功した。

Melamed[3] は単語の対応付けによって類似度を繰り返し再計算し、辞書をフィルタリングすることによって精度の高い結果を得た。

北村ら [8] は閾値を段階的に下げることによって類似度の高いエントリが先に決定されるようにしているが、各閾値においては、取り出される順は必ずしも類似度順にソートされていない。本研究の方法では、




cとCが互いに訳語で、かつCとDが互いに関連が深く似通った文脈に出現しやすいとき、直接関係のないcとDの共起頻度も大きくなってしまう。

図 1: 間接的関係

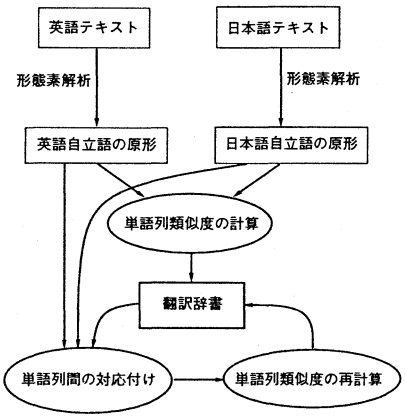



図 2: 翻訳辞書作成の過程

Melamed[3] と同様に対応付けを行なうことによって類似度の低いエントリをフィルタリングしていく。これによって常に類似度の高いエントリが優先されるので、よりきめ細かく優先順位を決めることが出来る。ただし、本研究では連語も扱うので、単語のみを扱った Melamed[3] の類似度再計算方法をそのまま取り入れることは出来ない。

## 3 翻訳辞書作成の過程

辞書の作成は以下の手順で行なう。

**1) 形態素解析・自立語抽出**

文対応の付いた対訳コーパスをそれぞれ形態素解析し、自立語[1]の原形を得る。

**2) 単語列の数え上げ**

各言語において、全ての可能な単語列[2]の出現頻度を数える。単語列を構成する語数の上限は $l_{max}$ とする。

**3) 単語列の共起頻度の数え上げ**

出現頻度が $T_f$ 以上の、ある英語単語列 $w_{ei}$ について、それが出現する文を選ぶ。それらの文の翻訳文について、その中に出現する日本語単語列 $w_{jt}(t = 1, ..., n)$ の出現頻度を数える。これによって、英語単語列 $w_{ei}$ と日本語単語列 $w_{jt}$ の共起頻度を数えていることになる。

**4) 類似度の計算**

以下の評価関数にしたがって類似度を計算し、翻訳辞書に登録する。この段階では北村ら [8] と同じ以下の評価式を用いる。

$$Sim(w_e, w_j) = \log(C_{ej}) * \frac{2 * C_{ej}}{C_e + C_j}$$

$w_e$: 英語単語列
$w_j$: 日本語単語列
$C_e$: 英語コーパスに $w_e$ が出現する頻度
$C_j$: 日本語コーパスに $w_j$ が出現する頻度
$C_{ej}$: 日英対訳コーパスに $w_e$ と $w_j$ が同じ対訳文に出現する頻度

あらゆる $w_{jt}$ に関して 4) を行い、さらにあらゆる $w_{ei}$ に関して 3) 4) を行う。これによって、すべての英語単語列、日本語単語列の組み合わせに付いて類似度を計算することになる。これによって類似度付きの翻訳辞書が得られる。

**5) 単語列の対応付け**

次に、この対訳辞書を元に、対訳コーパスに出現する英語、日本語それぞれの単語列を対応付ける。対応付けは翻訳辞書の類似度の高いエントリから順に考慮される。ある単語列がすでに対応付けをされている場合、その単語列を構成する語を含む単語列は対応付けされない。図3で、((B,C),b),(E,f) 間に対応付けが既になされている場合、((C,D),(c,d)),(F,f) 間に対応付けをすることは出来ない。

[1]自立語とは英語における形容詞、副詞、動詞等、日本語における形容詞、形容動詞、副詞、動詞等である。本論文におけるすべての実験で、自立語以外の語は無視した。

[2]単語列とは、連続した単語(本手法においては自立語の原形)の組である。たとえば「今日はいい天気だ。」という文では(今日), (いい), (天気), (今日, いい), (いい, 天気), (今日, いい, 天気)を指す。

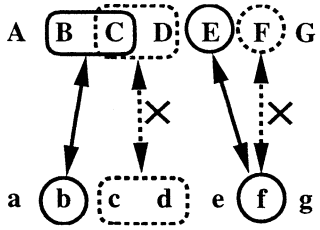



図 3: 対応付け

**6) 類似度の再計算**

そして以下の式によって類似度を再計算する。

$$Sim(w_e, w_j) = \log(L_{ej}) * \frac{2 * L_{ej}}{C_e + C_j}$$

$L_{ej}$: $w_e$ と $w_j$ が対応付けられた回数

**7) 辞書の更新**

翻訳辞書のエントリのうち、一度も対応付けがなされなかったものは取り除く。

5) ~ 7) を、変化が無くなるまで繰り返す。

この方法は、英日、日英の翻訳方法に依存しないため、英日、日英両方の翻訳辞書を作成することができる[3]。

## 4 実験

実験には取引条件表現法辞典 [7] の対訳文 9777 文を用いた。一つの文当りの平均単語数は英文約 23.7 語 / 文、日本語文約 29.0 語 / 文、自立語のみでは英文約 14.1 語 / 文、日本語文約 20.5 語 / 文だった。形態素解析は、英語文では Brill Tagger[1] の出力を元に辞書 (COMLEX Syntax) [5] を引き、原形を求めた。日本語文では日本語形態素解析システム『茶筌』 [6] を用いた。

単語列を構成する語数の上限 $l_{max}$ は 6 とし[4]、出現頻度の閾値 $T_f$ は 2 とした。

取引条件文 9777 文を 4 分割し、その 3/4 を辞書の作成に用い、残りの 1/4 を評価に用いる 4-fold のクロスバリデーションを行った。

客観的に辞書の質を判断するために、機械的評価を行なった。Melamed[4] の方法 (BiBLE) を一部修正した方法を用いた。E を原言語のコーパス、J を目標言語のコーパスとすると、機械的評価は以下の手順で行なわれる。

1. カウンタ a と c を 0 にする。
2. b を J における自立語の総出現数とする。

[3]ただし、4節で述べる通り、日英、英日で同一の類似度を用いるのではなく、翻訳方向によって類似度を変えた方が性能は良くなる。

[4]北村ら [8] は、単語列の構成単語数別に、抽出数と正解数を調べている。それによると構成単語数 7 以上の長い単語列を時間をかけて調べる利点はあまりなさそうである。

3. それぞれの対訳文 (e, j) ∈ (E, J) について、(a) ~ (d) を行う。

   (a) $\hat{j} = \phi$ とする[5]。

   (b) 文 e の中のそれぞれの自立語 s について、最も類似度の高い辞書のエントリを用いて s の翻訳を $\hat{j}$ に加える。単語列として考えた方が、その各々の構成要素を単語として考えた場合より類似度が高いエントリがある場合は、それらを単語列として扱う。翻訳が単語列である場合は、それを単語に分割して $\hat{j}$ に加える。もし辞書にエントリが存在しなかったら、それは無視される[6]。

   (c) $a = a + |\hat{j}|$

   (d) $\hat{j}$ のそれぞれの語について、それが文 j の中にあるかどうかを調べる。もしあれば、カウンタ c を 1 増やし、j からその語を除く[7]。

4. $Precision := c/a, Recall := c/b$ とする。

単語列の置き換えを行なう際に用いる類似度として、以下の二通りを用意した。

- Non-directed

  翻訳方向にかかわらず同一の類似度を用いる。

- Directed

  翻訳方向によって違う類似度を用意する。英語から日本語への翻訳を行うときの英語単語列 $w_e$ と日本語単語列 $w_j$ の類似度を $Sim(w_e \rightarrow w_j)$ とし、日本語から英語への翻訳を行うときの日本語単語列 $w_j$ と英語単語列 $w_e$ の類似度を $Sim(w_j \rightarrow w_e)$ とする。

$$Sim(w_e \rightarrow w_j) = \log(C_{ej}) * \frac{C_{ej}}{C_e}$$

$$Sim(w_j \rightarrow w_e) = \log(C_{ej}) * \frac{C_{ej}}{C_j}$$

### 4.1 結果、考察

適合率、再現率の値は絶対的には低いが、これは翻訳辞書の質が悪いことを意味しない。現実の対訳文においてはすべての文が逐語訳的であるとは限らない。またここで用いた機械的評価では同義語も不正解として排除される[9]。

[5]ここに Bag-of-word 翻訳機による訳語が加えられる。つまり、$\hat{j}$ と j がより似通っていれば辞書の性能が良いということになる。

[6]Melamed[4] はそのまま加えているが、これでは「取り得」、つまり正解する確率の低いエントリでも無いよりはましということになってしまう。また、英仏と英日では事情も違うだろう。そのままで正解の物が多いとは考えにくい。

[7]同じ文で複数回同じ訳語が出たときに、目標言語の文にはそれが一つしかないというときに重複してカウントするのを避ける。

[8]ただし、(北村 97) の手法では文の区切り (カンマ、読点) を超える単語列も考慮しているが、実験では考慮しないようにした。

[9]Melamed[2] はこの評価方式 (ただし辞書にないエントリも Precision の分母にカウントする) の最高点を見積もっている。それによると Precision = 0.62, Recall = 0.60 である。

| | 英語 | 日本語 |
|---|---|---|
| | company | 会社 |
| | distributor | 販売 店 |
| ※ | product | 契約 品 |
| | seller | 売り手 |
| | buyer | 買い手 |
| | article | 条 |
| | party | 当事者 |
| | territory | 地域 |
| | agent | 代理店 |
| | write | 書面 |

表 2: 辞書の抽出例 (1)

| | 英語 | 日本語 |
|---|---|---|
| | hereinafter | 本 契約 中 以下 |
| × | term condition | 諸 条件 |
| | individual contract | 個々 契約 |
| × | japan commercial | 国際 商事 |
| | organize exist law | 現存 する 法人 |
| | bill lade | 船荷 証券 |
| × | difference arise | 意見 相違 |
| × | govern law | 準拠 法 |
| | hereby | 本 契約 より |
| × | in witness whereof | 上記 証拠 する |

表 3: 辞書の抽出例 (2)

実験 1a より 実験 1b、実験 2a より 実験 2b、の方がスコアが良かったのは当然である。実験 1a と 実験 1b の差に比べて 実験 2a の 実験 2b 差が小さいのも止むをえない。北村ら [8] の手法では抽出した翻訳表現を構成する語を含む単語列について、その出現頻度を数え上げの対象から除く。これは一方または両方の単語列が多義語である場合に類似度が不当に低くなってしまうのを防ぐためだが、非対称な類似度を持ち込むのを難しくしているという不利な点もある。また、後から抽出された翻訳表現の中に、先に抽出されたものより高い類似度を持つものが出てくる場合もあるので、損得は微妙である。北村ら [8] の方法に非対称な類似度を持ち込むには無理があるので、実験 1b と 実験 2b の結果を比べるのはフェアでないとの見方も出来るが、最終的には翻訳精度の向上を目指すわけであり、翻訳時には翻訳方向を考慮した類似度を用いた方が有利なので、その非対称な類似度を持ち込みやすいという利点があると言える。

表 2は構築された辞書の上位 10 語を取り出したものである。表 3は辞書のエントリのうち、日本語、英語共に 2 語以上か、片方が 3 語以上の単語列であるものの上位 10 語を取り出したものである。間違いの中には、

表 1: 機械的評価

| | 英語 → 日本語 | | | 日本語 → 英語 | | | F- 値 |
|---|---|---|---|---|---|---|---|
| | 適合率 | 再現率 | F- 値 | 適合率 | 再現率 | F- 値 | 平均 |
| 実験 1a | 0.713 | 0.469 | 0.566 | 0.550 | 0.548 | 0.549 | **0.557** |
| 実験 2b | 0.719 | 0.461 | 0.562 | 0.537 | 0.541 | 0.539 | **0.550** |
| 実験 1b | 0.745 | 0.484 | 0.587 | 0.564 | 0.556 | 0.560 | **0.573** |
| 実験 2b | 0.740 | 0.470 | 0.575 | 0.545 | 0.544 | 0.544 | **0.559** |

1 本研究の方法
2 (北村 97) の方法[8]
a non-directed
b directed

$$F = \frac{2 * precision * recall}{precision + recall}$$

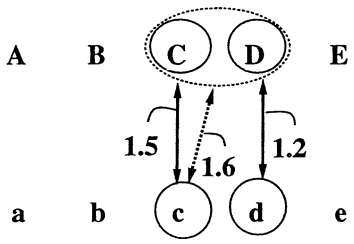



図 4: 文全体の対応付けの最適化

それぞれを構成している一部同士が訳語であるものが多く見受けられる。これは間接的関係によって直接関係のない語が結びつけられてしまったためだと考えられる。これを避ける方法としては、対応付けを単純に類似度の高いものから順に行うのではなく、文全体の類似度の総和を大きくするなどの方法で、文全体の対応付けを最適化することが考えられる。たとえば図 4で、(C,c),(D,d) より (CD,c) の類似度が高いが、(C,c),(D,d) の類似度の和が (CD,c) の類似度より高いので、前者を優先する。

## 5 おわりに

対訳テキストから翻訳表現を抽出する新しい方法を提案し、辞書の構築、機械的評価の実験を行った。その結果、過去の研究の方法より質の良い辞書を構築できることがわかった。一方で、長い単語列では適合率が低いという問題も残されている。

また、今回の実験では最初に構築される辞書のサイズが 200Mbyte 近くになった。より大きいサイズの対訳テキストを扱うためには、最初の類似度計算と対応付けの処理を直結させるなどの工夫が必要になる。

今回は連続の単語列のみを扱ったが、上記の問題を解決すれば、離散型の翻訳表現についても同様の処理を行なうことが考えられる。

## 参考文献


[1] Eric Brill. Some advances in rule-based part of speech tagging. In *AAAI*, 1994.

[2] I. Dan Melamed. Automatic evaluation and uniform filter cascades for inducing n-best translation lexicons. In *Proceedings of the Third Workshop on Very Large Corpora (WVLC3)*, pp. 184–198, 1995.

[3] I. Dan Melamed. Automatic construction of clean broad-coverage translation lexicons. In *2nd Conference of the Association for Machine Translation in the Americas (AMTA'96), Montreal, Canada*, 1996.

[4] I. Dan Melamed. Automatic discovery of non-compositional compounds in parallel data. In *proceeding of EMNLP-2*, 1997.

[5] Meyers, Adam, Catherine Macleod and Ralph Grishma. *COMLEX Syntax 2.0 Manual for Tagged Entries*. Proteus Project, New York University, 1995. http://cs.nyu.edu/cs/faculty/grishman/comlex.html.

[6] 松本裕治, 北内啓, 山下達雄, 今一修, 今村友明. 日本語形態素解析システム『茶筌』 version 1.0 使用説明書, 1997. NAIST Technical Report, NAIST-IS-TR97007.

[7] 石上進. 取引条件表現法辞典 電子ブック版 第 1 巻 物品取引. 国際事業開発株式会社, 1992.

[8] 北村美穂子、松本裕治. 対訳コーパスを利用した対訳表現の自動抽出. 情報処理学会論文誌, Vol. 38, No. 4, pp. 727–736, 1997.