用例利用型翻訳のための類似用例検索手法

池田成宏　内野一　古瀬蔵
ＮＴＴサイバーソリューション研究所
{ikeda, uchino, furuse}@soy.kecl.ntt.co.jp

1. はじめに

用例利用型翻訳は人間による翻訳例を利用して高品質な翻訳を行うことができることから、近年盛んに研究が行われている[1,2]。

現在、我々が究開発を行っている用例利用型翻訳では、入力文に類似した対訳用例文の目的言語文を自動加工して訳文を生成する[3]。訳文の加工は、入力文と用例文の単語対応に基づいて行われるため、入力文と用例文との正しい単語対応も考慮した類似文検索が必要となっている。

現在の用例利用型翻訳システムでは、入力文と用例文との単語対応は動的計画法に基づいて求められる。しかし、動的計画法では入力文と用例文とで語順が入れ替わった場合には、正しい対応づけを行うことはできず、誤訳の原因となっている。

そこで本稿では、入力文と用例文とで語順が異なる場合でも、正しい単語対応に基づいて類似文を検索する列島（Archipelago）アルゴリズムを提案し、その有効性を示す。

2. 動的計画法に基づく類似文検索手法

類似文検索は用例利用型翻訳において重要な機能の一つであり、これまでに様々な類似文検索手法が提案されている。その中で最も代表的な手法が、動的計画法に基づく手法である[4,5]。

図１に動的計画法による類似文検索手法の基本的な考え方を示す。動的計画法に基づく手法では、単語が対応しているセルを通り、始点から終点に至る最適な経路を求める。そして、終点におけるコストに基づいて類似度を計算する[4,5]。

しかし、図１の例のように入力文と用例文で語順が異なる場合には、経路に含まれない「５－１で」と「３－０で」の対応は無視される。そのため、対応づけられるべき語句が対応づけられなかったり、誤った対応づけが行われてしまう。そして、このような誤った対応関係に基づいて翻訳処理が行われると、誤った訳文が生成されるという問題が生じる。

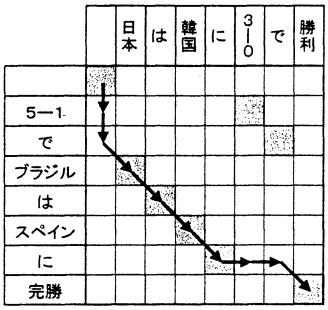

図 1 動的計画法に基づく検索手法の例

3. 列島（Archipelago）アルゴリズム

本稿で提案する新しい類似文検索手法であるArchipelago アルゴリズムでは、二文間で最適な一対一の単語対応の組み合せを求め、それに基づいて類似度を計算する。本手法では動的計画法のように経路を１本に限定しないため、語順が異なる場合でも正しい単語対応の組み合せを求めることが可能となる。

図２に Archipelago アルゴリズムによる単語対応づけの例を示す。Archipelago アルゴリズムでは一対の単語対応を「島」とみなし、島が右下方向に連続して並んだものを「列島」="Archipelago"と呼ぶ。列島は連続する単語同士が対応することを意味し、列島の大きさを利用して単語対応を最適化することができる。

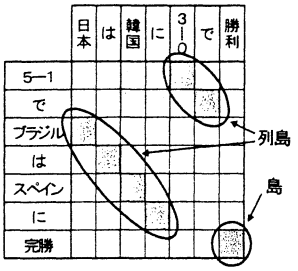

図 2 Archipelago アルゴリズムの概要

単語対応の組み合せを最適化するために、単語対応の並びの連続性を反映するマッチングスコアを導入する。そして、そのスコアを最大化するような単語対応の組み合せを求める。このときの単語対応の組み合せが最適な単語対応の組み合せであり、マッチングスコアが類似度となる。

3-1. 単語マッチングスコア

単語対応の最適化を行うために、島には表1のような単語マッチングスコアを与える。品詞、表記が完全に一致している場合に最もスコアが高く、同じカテゴリの固有名詞、数詞でマッチした場合がそれに続く。また、品詞や表記だけでなく意味情報も利用可能な場合には、意味の類似性に応じてマッチングスコアを与えることもできる。

3-2. マッチングスコアの計算

ここでは、ある単語対応の組み合せが与えられた場合の入力文と用例文のマッチングスコアの計算方法について説明する。

t、eをそれぞれ入力文T、用例文Eの先頭からの単語のインデックスとし、一対の単語対応を(t,e)と表す。図3では一部の単語対応についてのみスコアを記している。

n対の単語対応からなる単語対応の組み合せCを

$$C=\{(t_1,e_1),(t_2,e_2),\ldots,(t_n,e_n)\} \qquad (1)$$

$$t_i \neq t_j (i \neq j), e_i \neq e_k (i \neq k)$$

のように表す。ただし、入力文Tのt番目の単語$T(t)$、用例文Eのe番目の単語$E(e)$の単語マッチングスコアを$match(T(t),E(e))$としたとき、C中の単語対応(t,e)は$match(T(t),E(e))>0$となるようなものでなくてはならない。

単語対応の組み合せCが与えられた場合の入力文Tと用例文EのマッチングスコアS_{TEC}は以下のようなアルゴリズムによって求められる。

表 1 単語マッチングスコアの例

単語対応のタイプ	スコア
完全一致(助動詞、記号以外)	8点
同じカテゴリの固有名詞、数詞	7点
助動詞、記号で完全一致	4点
意味カテゴリが類似	4～1点
自立語で品詞が一致	1点
不一致(その他)	0点

・ **step1 列島の形成 (I)**

右下方向に連続している単語対応（島）をまとめ、列島を形成する。ただし、単語の文法的な連続性を考慮し、この段階では列島は文節の境界を越えないようにする。しかし、文節のような文法的な区切りがない言語の場合には、このような制限はない。

列島B_{TECj}がp個の連続した島から形成される場合、列島B_{TECj}は次のように表される。

$$B_{TECj}=\{(t_{j1},e_{j1}),(t_{j2},e_{j2}),\ldots,(t_{jp},e_{jp})\} \quad (2)$$

そして、列島B_{TECj}のスコアU_{TECj}を以下のように定義する。

$$U_{TECj}=\{\sum_{(t,e)\in B_{TECj}} match(T(t),E(e))\}^2 \quad (3)$$

図3の例では、「ブラジル」と「日本」、「は」と「は」の二対の単語対応が、「ブラジルは」と「日本は」という一つの列島にまとめられ、そのスコアは$(7+8)^2$となる。

・ **step2 列島の形成 (II)**

連続する列島をまとめて、より大きな列島を

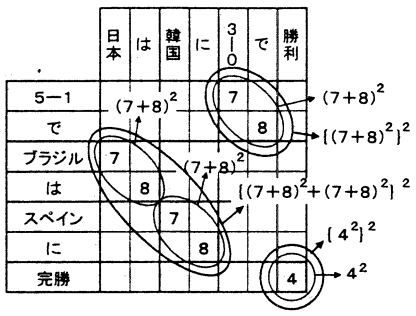

図3　Archipelago アルゴリズムのスコア計算例

形成する。**step1** では文節の境界で分断されていた列島は、この段階で結合される。q 個の連続する列島から形成される列島 P_{TECk} は、次のように表される。

$$P_{TECk} = \{B_{TECk1}, B_{TECk2}, \ldots, B_{TECkq}\} \quad (4)$$

そして、列島 P_{TECk} のスコア V_{TECk} は次のようになる。

$$V_{TECk} = \Big(\sum_{B_{TECj} \in P_{TECk}} U_{TECj} \Big)^2 \quad (5)$$

図 3 では、「ブラジルは」と「日本は」、「スペインに」と「韓国に」の 2 つの列島から、「ブラジルはスペインに」と「日本は韓国に」によるより大きな列島が形成され、そのスコアは $\{(7+8)^2+(7+8)^2\}^2$ となる。

・ **step3 全列島のスコアの計算**

全列島のスコアの総和をとり、それを W_{TEC} とする。

$$W_{TEC} = \sum_k V_{TECk} \quad (6)$$

図 3 の例では、全列島のスコアの総和 W_{TEC} は、$\{(7+8)^2+(7+8)^2\}^2+\{(7+8)^2\}^2+\{4^2\}^2=253381$ となる。

次に、スコアの最大値によって、スコアを正規化する。

$$N_{TEC} = \left(\frac{W_{TEC}}{W_{TT\max}^{1/2} W_{EE\max}^{1/2}} \right)^{1/4} \quad (7)$$

ここで、$W_{TT\max}$、$W_{EE\max}$ はそれぞれ入力文 T 同士、用例文 E 同士で列島のスコアの総和を計算した場合の最大値を表し、右辺全体の 1/4 乗はスケーリングのためである。図 3 の例では、$N_{TEC}=(253381/832 \cdot 832)^{1/4}=0.778$ となる。

・ **step4 最終的なスコアの計算**

step3 で計算された N_{TEC} だけでは語順によるスコアへの影響が大きい。そこで、単純な単語対応の割合 R_{TEC} も考慮して、二文間のマッチングスコア S_{TEC} を式(8)、(9)により計算する。

$$S_{TEC} = N_{TEC} R_{TEC} \quad (8)$$

$$R_{TEC} = \frac{\sum_{(t,e) \in C} match(T(t), E(e))}{(\sum_t match(T(t), T(t)))^{1/2} (\sum_e match(E(e), E(e)))^{1/2}} \quad (9)$$

図 3 の例では、$R_{TEC}=0.875$ となるため、$S_{TEC}=0.778 \times 0.875=0.681$ となる。

3-2. 単語対応の最適化

2 文間の類似度は、単語対応の組み合せが最適となったときのマッチングスコアである。最適な単語対応の組み合せ C_{opt} は以下のような手続きによって求められる。

・ **step1 初期化**

単語対応の組み合せ C を空にする

・ **step2 スコア計算**

$match(T(t), E(e)) > 0$ かつ $(t,e) \notin C$ の各単語対応候補 (t,e) について、C と (t,e) からなる単語対応集合で文マッチングスコアを計算する。ただし、一対一の単語対応の制約に反する場合には、この制約に反する単語対応を C から削除してマッチングスコアを計算する。

・ **step3 単語対応候補選択**

step2 で最大のマッチングスコアを与える (t,e) を C に追加する。

以上のstep2, 3をマッチングスコアが変化しなくなるまで繰り返す。すると、最終的に最適な単語対応の組み合せ C_{opt} が得られる。図3の例は、最適な単語対応の組み合せのスコア計算例であり、類似度は0.68となる。

4. 実験

本手法の有効性を確認するために、我々が作成したスポーツ情報コーパスに対して検索実験を行った。

このコーパス中の34文について177文の中から類似文を検索し、最も類似している文が検索された順位によって比較を行った。Archipelagoアルゴリズムと文献[3]の従来手法の類似文検索結果は表2のようになった。

表2から、Archipelagoアルゴリズムは従来手法と同等以上の性能を示していることがわかる。

次に、単語対応の正確性を調べるための実験を行った。上記の実験と同様にスポーツ情報コーパス中の12文の各文について、以下に示すように語順を入れ替えることによって3文を作成した。ただし、3文のうち2文（文1、文2）は原文と意味が異なり、残りの1文（文3）のみが原文と同じ意味の文である。

原文：中央から中田が、左にいたMF名波にパス。
文1：中央から中田に、左にいたMF名波がパス。
文2：左にいたMF名波が、中央から中田にパス。
文3：左にいたMF名波に、中央から中田がパス。

このようにして作成された3文の中から原文と同じ意味を持つ文を検索し、原文と同じ意味を持つ文が検索された順位の比較を行った。検索結果を表3に示す。なお、上記の文1−3については、原文との類似度はそれぞれ0.430、0.427、0.615となった。

また、原文と同じ意味を持つ文に対して正しい単語対応づけが行われた割合は、Archipelagoアルゴリズムでは、260語／260語=1.0、動的計画法では228語／260語=0.88であった。

以上のことから、Archipelagoアルゴリズムでは語順が異なる場合でも正しい単語対応を検出し、前後の単語との連続性を反映して意味的に同じ文を検索可能なことがわかる。

表2　類似文検索結果の比較(1)

検索手法	1位	2位	3位	4位以下
Archipelago	18	7	2	7
従来手法	16	5	5	8

表3　類似文検索結果の比較(2)

検索手法	1位	2位	3位
Archipelago	12	0	0
従来手法	2	2	8

5. まとめ

本稿では、入力文と用例文とで語順が入れ替わった場合でも、正しい単語対応の組み合せに基づいて類似文を検索するArchipelagoアルゴリズムを提案した。実験により、入力文とは語順が入れ替わった用例文でも正しい単語対応の組み合せを求め、単語対応の連続性を反映した類似度を計算できることを確認した。

今後は、単語と複合語のマッチングの問題などについても検討し、さらに評価をすすめる予定である。

参考文献

[1]古瀬他: “経験的知識を活用する変換主導型機械翻訳”, 情報処理学会論文誌, Vol.35, No.3, pp.414-425, 1994.

[2]渡辺他: “用例ベース処理を用いたパターンベース翻訳システム”, 言語処理学会第4回年次大会論文集, pp.488-491, 1998.

[3]高橋他: “用例利用型日英機械翻訳の基本設計”, 言語処理学会第3回年次大会論文集, pp.145-148, 1997.

[4]Stephen, A.G.: “STRING SEARCHING ALGORITHMS”, World Scientific, 1994.

[5]Planas, E. et al.: “Formalizing Translation Memories”, *proc. of MT Summit VII*, pp.331-339, 1999.