2020年度（令和2年度）　卒業論文

類似度を利用した
変換テーブルの精度向上

電気情報系学科 卒業論文検印	
学科長	

指導教員

村田　真樹
村上　仁一

鳥取大学工学部　電気情報系学科
自然言語処理研究室
B16T2117C 森本　世人

概要

相対的意味論に基づく変換主導型統計機械翻訳 TDSMT[1](Transfer Driven Statistical Machine Translation : 以下 TDSMT と記述する) が提案されている．TDSMT では変換テーブルを用いて翻訳を行う．変換テーブルは学習文対から自動作成する. 学習文対は英語文と日本語文の対である. 変換テーブルとは「A が B ならば C は D である」という A，B，C，D の相対性に基づいて関係を定義したテーブルである．ここで A と B は単語である．また，C と D は単語もしくは句である．しかし， TDSMT は自動で変換テーブルを作成するため, 誤った変換テーブルを作成する場合がある. そこで，誤った変換テーブルを削除するために変換テーブルを生成した後に，枝刈りを行う．従来行われている枝刈りの手法として，A と B，C と D の対訳単語確率を用いた枝刈りがある．しかし，枝刈りの精度は未だ不十分である．

本研究では，従来手法で行う枝刈りに加えて，A と C，B と D の類似度を用いて枝刈りを行うことを提案する．

実験では, 従来手法と提案手法を用いて作成した変換テーブルを枝刈りした. そして, 従来手法と提案手法の誤り率を比較した. 実験の結果, 提案手法によって変換テーブルの精度が大幅に向上した.

目 次

図 目 次

表 目 次

第1章　はじめに

機械翻訳において, 相対的意味論に基づく変換主導型統計機械翻訳 TDSMT が提案されている. TDSMT は変換テーブルを用いて, 学習文対を変換し, 出力文を作成する. 変換テーブルは「A が B」ならば「C は D である」という A，B，C，D の相対的関係に基づいて定義されたテーブルである. ここで A と B は単語である．また，C と D は単語もしくは句である．変換テーブルは学習文対 (パラレルコーパス) から自動作成する. 作成には IBM Model 1[3], パターンを利用する. TDSMT は学習文対 1 対から複数の変換テーブルを作成する. また, 出力の導出過程の解析もニューラル機械翻訳と比べ, 容易である. しかし，TDSMT において誤った変換テーブルを作成してしまうという問題点が存在する.

そこで，誤った変換テーブルを削除するために変換テーブルを生成した後に，枝刈りを行う．従来行われている枝刈りの手法として，A と B，C と D の対訳単語確率を用いた枝刈りがある．しかし，枝刈りの精度は未だ不十分である.

TDSMT は学習文対の単語を変換テーブルを利用し, 置き換えることによって出力文を得る. つまり, 変換テーブルは A と C, そして B と D が置き換え可能な関係が想定される. そこで, 本研究では従来手法で行う枝刈りに加えて，A と C，B と D の類似度を用いて枝刈りを行うことを提案する．本研究において，類似度とは注目単語の前後環境がどれだけ一致しているかと定義する．提案手法を用いることで変換テーブルの精度を向上できると考える.

第2章　従来研究

2.0.1　IBM翻訳モデル

IBM 翻訳モデルを以下に示す. 本節は力久ら [5] の抜粋である. 統計翻訳の代表的なモデルとして, IBM の Brown らによる仏英翻訳モデルがある.IMB 翻訳モデルは, 単語に基づく統計翻訳を想定して作成された, 単語対応の確率モデルである. この翻訳モデルは順に複雑な計算を行うモデル 1 から 5 の 5 つのモデルで構成される. また本研究ではモデル 1 を用いた.

本章では. 原言語であるフランス語文を F, 目的言語である英語文を E として定義する.

IBM モデルでは, フランス語文 E, 英語文 F の翻訳モデル $P(F|E)$ を計算するために, アライメント a を用いる. 以下に IBM モデルの基本式を示す.

$$P(F|E) = \sum_{a} P(F, a|E) \tag{2.1}$$

アライメントとは仏単語と英単語の対応を意味している.IBM モデルのアライメントでは, 各仏単語 f に対応する英単語 e は 1 つあり, 各英単語 e に対応する仏単語は 0 から n 個ある. また仏単語 f において適切な英単語と対応しない場合, 英語文の先頭に空単語 e_0 があると仮定し, その仏単語 f と空単語 e_0 を対応づける.

・モデル 1

(2.1) 式は以下の式に分解することができる.m はフランス語文の長さ, a_1^{j-1} はフランス語文における, 1 番目から $j-1$ 番目までのアライメント, f_1^{j-1} はフランス語文における, 1 番目から $j-1$ 番目まで単語を表している.

$$P(F, a|E) = P(m|E) \prod_{j=1}^{m} P(a_j|a_1^{j-1}, f_1^{j-1}, m, E) P(f_j|a_1^{j}, f_1^{j-1}, m, E) \tag{2.2}$$

(2.2) 式ではとても複雑であるので計算が困難である. そこで, モデル 1 では以下の仮定により, パラメータの簡略化を行う.

- フランス語文の長さの確率 ϵ は m, E に依存しない

$$P(m|E) = \epsilon$$

- アライメントの確率は英語文の長さ l に依存する

$$P(a_j|a_1^{j-1}, f_1^{j-1}, m, E) = (l+1)^{-1}$$

- フランス語の翻訳確率 $t(f_j|e_{a_j})$ は, 仏単語 f_j に対応する英単語 e_{a_j} に依存する

$$P(f_j|a_1^j, f_1^{j-1}, m, e) = t(f_j|e_{a_j})$$

パラメータの簡略化を行うことで, $P(F, a|E)$ と $P(F, E)$ は以下の式で表される.

$$P(F, a|E) = \frac{\epsilon}{(l+1)^m} \prod_{j=1}^{m} t(f_j|e_{a_j}) \tag{2.3}$$

$$P(F|E) = \frac{\epsilon}{(l+1)^m} \sum_{a_1=0}^{l} \cdots \sum_{a_m=0}^{l} \prod_{j=1}^{m} t(f_j|e_{a_j}) \tag{2.4}$$

$$= \frac{\epsilon}{(l+1)^m} \prod_{j=1}^{m} \sum_{i=0}^{l} t(f_j|e_{a_j}) \tag{2.5}$$

モデル 1 では翻訳確率 $t(f|e)$ の初期値が 0 以外の場合, Expectation-Maximization(EM) アルゴリズムを繰り返し行うことで得られる期待値を用いて最適解を推定する.EM アルゴリズムの手順を以下に示す.

手順 1 翻訳確率 $t(f|e)$ の初期値を設定する.

手順 2 仏英対訳対 $(F^{(s)}, E^{(s)})$(但し, $1 \leq s \leq S$) において, 仏単語 f と英単語 e が対応する回数の期待値を以下の式により計算する.

$$c(f|e; F, E) = \frac{t(f|e)}{t(f|e_0) + \cdots + t(f|e_l)} \sum_{j=1}^{m} \delta(f, f_j) \sum_{i=0}^{l} \delta(e, e_i) \tag{2.6}$$

$\delta(f, f_j)$ はフランス語文 F 中で仏単語 f が出現する回数, $\delta(e, e_i)$ は英語文 E 中で英単語 e が出現する回数を表している.

手順 3 英語文 $E^{(s)}$ の中で 1 回以上出現する英単語 e に対して, 翻訳確率 $t(f|e)$ を計算する.

1. 定数 λ_e を以下の式により計算する.

$$\lambda_e = \sum_f \sum_{s=1}^{S} c(f|e; F^{(s)}, E^{(s)}) \tag{2.7}$$

2. (2.7) 式より求めた λ_e を用いて, 翻訳確率 $t(f|e)$ を再計算する.

$$\begin{aligned} t(f|e) &= \lambda_e^{-1} \sum_{s=1}^{S} c(f|e; F^{(s)}, E^{(s)}) \\ &= \frac{\sum_{s=1}^{S} c(f|e; F^{(s)}, E^{(s)})}{\sum_f \sum_{s=1}^{S} c(f|e; F^{(s)}, E^{(s)})} \end{aligned} \tag{2.8}$$

手順 4 翻訳確率 $t(f|e)$ が収束するまで手順 2 と手順 3 を繰り返す.

・モデル 2

モデル 1 では, 全ての単語の対応に対して, 英語文の長さ l にのみ依存し, 単語対応の確率を一定としている. そこで, モデル 2 では, j 番目の仏単語 f_j と対応する英単語の位置 a_j は英語文の長さ l に加えて, j と, フランス語文の長さ m に依存し, 以下のような関係とする.

$$a(a_j|j, m, l) \equiv P(a_j|a_1^{j-1}, f_1^{j-1}, m, l) \tag{2.9}$$

この関係からモデル 1 における (2.4) 式は, 以下の式に変換できる.

$$P(F|E) = \epsilon \sum_{a_1=0}^{l} \cdots \sum_{a_m=0}^{l} \prod_{j=1}^{m} t(f_j|e_{a_j}) a(a_j|j, m, l) \tag{2.10}$$

$$= \epsilon \prod_{j=1}^{m} \sum_{i=0}^{l} t(f_j|e_{a_j}) a(a_j|j, m, l) \tag{2.11}$$

モデル 2 では, 期待値は $c(f|e; F, e)$ と $c(i|j, m, l; F, E)$ の 2 つが存在する. 以下の式から求められる.

$$c(f|e; F, E) = \frac{t(f|e)}{t(f|e_0) + \cdots + t(f|e_l)} \sum_{j=1}^{m} \delta(f, f_j) \sum_{i=1}^{l} \delta(e, e_i) \tag{2.12}$$

$$= \sum_{j=1}^{m} \sum_{i=0}^{l} \frac{t(f|e) a(i|j, m, l) \delta(f, f_j) \delta(e, e_i)}{t(f|e_0) a(0|j, m, l) + \cdots + t(f|e_l) a(l|j, m, l)} \tag{2.13}$$

$$c(i|j, m, l; F, E) = \sum_a P(a|E, F) \delta(i, a_j) \tag{2.14}$$

$$= \frac{t(f_j|e_i) a(i|j, m, l)}{t(f_j|e_0) a(0|j, m, l) + \cdots + t(f_j|e_l) a(l|j, m, l)} \tag{2.15}$$

$c(f|e;F,E)$ は対訳文中の英単語 e と仏単語 f が対応付けされる回数の期待値, $c(i|j,m,l;F,E)$ は英単語の位置 i が仏単語の位置 j に対応付けされる回数の期待値を表している.

モデル 2 では, EM アルゴリズムで計算すると複数の極大値が算出され, 最適解が得られない可能性がある. モデル 1 では $a(i|j,m,l) = (l+1)^{-1}$ となるモデル 2 の特殊な場合であると考えられる. したがって, モデル 1 を用いることで最適解を得ることができる.

・モデル 3

モデル 3 は, モデル 1 とモデル 2 とは異なり, 1 つの単語が複数対応する単語の繁殖数や単語の翻訳位置の歪みについて考慮する. またモデル 3 では単語の位置を絶対位置として考える. モデル 3 では以下のパラメータを用いる.

- 翻訳確率 $P(f|e)$
  英単語 e が仏単語 f に翻訳される確率
- 繁殖確率 $n(\phi|e)$
  英単語 e が ϕ 個の仏単語と対応する確率
- 歪み確率 $d(j|i,m,l)$
  英語文の長さ l, フランス語文の長さ m のとき, i 番目の英単語 e_i が j 番目の仏単語 f_j に翻訳される確率

さらに, 英単語が仏単語に翻訳されない個数を ϕ_0 とし, その確率 p_0 を以下の式で求める. このとき, 歪み確率は $\frac{1}{\phi_0!}$ で, $p_0 + p_1 = 1$ で p_0, p_1 は 0 より大きいとする.

$$P(\phi_0|\phi_1^l, E) = \left(\begin{array}{c} \phi_1 + \cdots + \phi_l \\ \phi_0 \end{array} \right) p_0^{\phi_1+\cdots+\phi_l-\phi_0} p_1^{\phi_0} \tag{2.16}$$

したがって, モデル 3 は以下の式で求められる.

$$\begin{aligned} P(F|E) &= \sum_{a_1=0}^{l} \cdots \sum_{a_m=0}^{l} P(F,a|E) && (2.17) \\ &= \sum_{a_1=0}^{l} \cdots \sum_{a_m=0}^{l} \left(\begin{array}{c} m-\phi_0 \\ \phi_0 \end{array} \right) p_0^{m-2\phi_0} p_1^{\phi_0} \prod_{i=1}^{l} \phi_i! n(\phi_i|e_i) \\ &\times \prod_{j=1}^{m} t(f_j|e_{a_j}) d(j|a_j,m,l) && (2.18) \end{aligned}$$

モデル 3 では, 全てのアライメントを計算するため, 計算量が膨大となるので期待値を近似により求める.

・モデル 4

モデル 4 では, モデル 3 と異なり, 単語の位置を絶対位置ではなく, 相対位置で考える. またモデル 3 では考慮されていない各単語の位置, 例えば形容詞と名詞の関係を考慮する. モデル 4 では歪み確率 $d(j|i.m,l)$ を 2 つの場合で考える.

- 繁殖数が 1 以上である英単語に対応する仏単語の中で, 最も文頭に近い場合

$$P(\Pi_{[i]1} = j|\pi_1^{[i]-1}, \tau_0^l, \phi_0^l, E) = d_1(j - \odot_{i-1}|\mathcal{A}(e_{[i-1]}), \mathcal{B}(f_j)) \tag{2.19}$$

  $\odot_{i-1}$ は i-1 番目の英単語が対応する仏単語の位置を表している.

- それ以外の場合

$$P(\Pi_{[i]k} = j|\pi_{[i]1}{}^{k-1}, \pi_1^{[i]-1}, \tau_0^l, \phi_0^l, E) = d_{>1}(j - \pi_{[i]k-1}|\mathcal{B}(f_j)) \tag{2.20}$$

  $\pi_{[i]k-1}$ は同じ英単語に対応している直前の仏単語を表している.

・モデル 5

モデル 4 では, 単語の位置に関して直前の単語以外は考慮されていない. したがって, 複数の単語が同じ位置に生じたり, 単語の存在しない位置が生成される. モデル 5 では, この問題を避けるために, 単語を空白部分に配置するよう改善が施されている.

- 繁殖数が 1 以上である英単語に対応する仏単語の中で, 最も文頭に近い場合

$$\begin{aligned} P(\Pi_{[i]1} &= j|\pi_1^{[i]-1}, \tau_0^l, \phi_o^l, E) \\ &= d_1(v_j|\mathcal{B}(f_j), v_{\odot_{i-1}}, v_m - \phi_{[i]} + 1)(1 - \delta(v_j, v_{j-1})) \end{aligned}$$

  v_j は j 番目までの空白数, $\mathcal{A}$ は英語の単語クラス $\mathcal{B}$ はフランス語の単語クラスを表している.

- それ以外の場合

$$\begin{aligned} P(\Pi_{[i]k} &= j|\pi_{[i]1}{}^{k-1}, \pi_1^{[i]-1}, \tau_0^l, \phi_0^l, E) \\ &= d_{>1}(v_j - v_{\pi_{[i]k-1}}|\mathcal{B}(f_j), v_m - v_{\pi_{[i]k-1}} - \phi_{[i]} + k)(1 - \delta(v_j, v_{j-1})) \end{aligned}$$

2.0.2 GIZA++

GIZA++[8] とは, 統計翻訳で用いることを前提に作られたツールである.IBM 翻訳モデルを用いて, 対訳文 (原言語文と目的言語文の対) から対訳単語と単語翻訳確率を自動的に得る.

2.1 相対的意味論に基づく変換主導統計機械翻訳 (TDSMT)

本章は中村 [7] らの抜粋である. “相対的意味論に基づく変換主導型統計機械翻訳 (TDSMT)” は, 安場らが提案した機械翻訳の手法である. TDSMT は, 学習文対と, 変換テーブルを用いて, 原言語文を入力とし, 目的言語文を出力する. 変換テーブルは “*A* が *B* ならば *C* は *D*” で表現する. *A* は学習文対中の原言語句, *B* は学習文対中の目的言語句, *C* は入力文中の原言語句, *D* は出力文中の目的言語句である.

原言語入力文が, 学習文対の原言語側と一致するまで, 入力文と変換テーブル中の *AC* を照合する. 次に, 一致した学習文対の目的言語側を, 照合した変換テーブルの *BD* に従って変換し, 目的言語翻訳文を出力する.

TDSMT は変換テーブルを学習文対から自動作成する. しかし, 問題点として, 誤った対訳を含む変換テーブルを作成することがあげられる.

2.1.1 TDSMT の手順

TDSMT の手順を示す. 手順は “学習” と “翻訳” の二部からなる.

2.1.2 学習の手順

TDSMT における学習は “変換テーブルの作成” のみである. 本節で作成手順を示す.

手順 1 対訳単語の作成

学習文対と対訳単語確率 (IBM Model 1) を利用して, 対訳単語を作成する. このとき付与される対訳単語確率を P_w とする. 例として, 表 2.1 に示す学習文対を使用して, 表 2.2 に示す対訳単語を作成する. 表 2.2 の値は例であり, 実際の数値とは異なる.

表 2.1: 対訳単語作成に用いる学習文対

学習文対 (日本語側)	彼の弟は学生だ。
学習文対 (英語側)	His brother is a student.

表 2.2: 作成される対訳単語

	日本語単語	英語単語	p_w
対訳単語 1	彼	His	0.4
対訳単語 2	弟	brother	0.7
対訳単語 3	学生	student	0.6

手順 2　パターンの作成

学習文対内で対訳単語に当たる部分を変数化し, パターンを作成する. 例を表 2.3 に示す.

表 2.3: パターンの作成例

学習文対 (日本語側)	彼の兄は医者だ。
学習文対 (英語側)	His brother is a doctor.
パターン (日本語側)	*X0* の *X1* は *X2* だ
パターン (英語側)	*X0 X1* is a *X2*

手順 3　変換テーブルの作成

学習文対とパターンを照合する. 変数化した対訳単語と, 変数に当たる対訳句を変換テーブルとする. 表 2.4 では変数 *N2* の部分から変換テーブル “「学生」が「student」ならば「教師」は「teacher」” が得られる.

表 2.4: 変換テーブルの作成例

学習文対 (日本語側)	彼の弟は学生だ。
学習文対 (英語側)	His brother is a student.
パターン (日本語側)	*X0* の *X1* は *X2* だ。
パターン (英語側)	*X0 X1* is a *X2* .
照合する学習文対 (日本語側)	私の母は教師だ。
照合する学習文対 (英語側)	My mother is a teacher.
変換テーブル *(X2)*	*A*:学生　*B*:student *C*:教師　*D*:teacher

手順 4　変換テーブルに確率を付与

対訳単語確率 P_w を利用し, 変換テーブルに確率を付与する. この確率を変換テーブル確率 P_v とする.

1. 変換テーブルの *CD* に存在する全ての日英単語の組み合わせを確認する.
2. 日本語単語に対応する英語単語の中で, 対訳単語確率 P_w の最大値を得る.
3. 各日本語単語について得られた値と, 変換テーブルの *AB* の対訳単語確率 P_w について, 対数の総和を求める.

2.1.3 翻訳の手順

本節で TDSMT における翻訳の手順を示す. 入力文を「私の姉は教師だ。」とする.

手順 1　入力文に日本語側の変換テーブルを適用

変換テーブルの C と A を利用して, 入力文を学習文対の日本語側と一致させる. 表 2.5 では入力文中の「教師」を「生徒」に変換する.

表 2.5: 日本語側変換テーブルの適用例

入力文	私の姉は教師だ。
変換テーブル:C	教師
変換テーブル:A	生徒
一致する学習文対 (日本語側)	私の姉は生徒だ。

手順 2　学習文対に英語側の変換テーブルを適用

手順 1 と同じ変換テーブルの B と D を学習文対の英語側に適用し, 出力候補文を作成する. 表 2.6 では学習文対中の「student」を「teacher」に変換している.

表 2.6: 英語変換テーブルの適用例

一致した学習文対 (日本語側)	私の姉は生徒だ。
一致した学習文対 (英語側)	My sister is a student.
変換テーブル:B	student
変換テーブル:D	teacher
出力候補文	My sister is a teacher.

手順 3　最終的な出力文の決定

複数の出力候補文が得られた場合, 計算式 (2.21) に従って, 最終的な出力文を決定する. ここで P_m は言語モデルの確率である.

$$logP = logP_v + logP_m \tag{2.21}$$

図 2.1 に TDSMT の流れ図を示す.

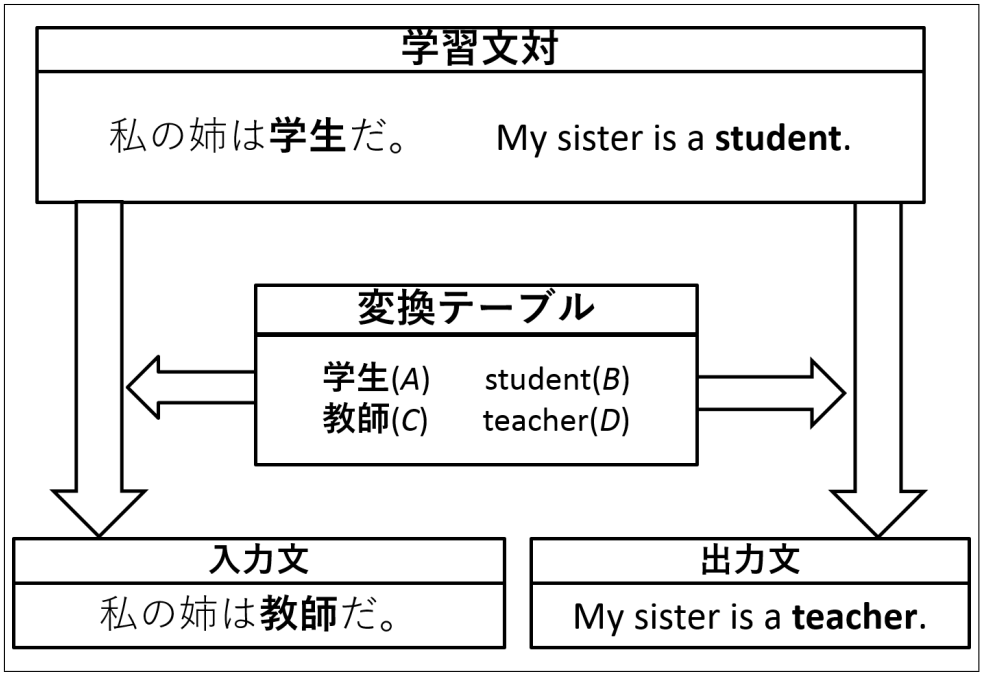

図 2.1: TDSMT の流れ図

2.1.4 変換テーブルの問題点

TDSMT の問題点は, 誤った変換テーブルが存在する点である. 表 2.7 に誤った変換テーブルの作成過程を示す. 表 2.7 において, *X2* の変換テーブルの「C」は「髪」である. しかし,「D」 は 「dyed」である. ゆえに, TDSMT は *C*,*D* の訳が誤っているため, 置き換え不可能である変換テーブルが存在する.

表 2.7: 誤った変換テーブルの作成過程

学習文対 (1)	日本語	私 は 英語 を 勉強し た 。		
	英語	I studied English .		
単語レベル 文パターン (手順 2)	日本語	*X1* は *X2* を *X3* た 。		
	英語	*X1 X3 X2* .		
学習文対 (2)	日本語	彼女 は 髪 を 染め た 。		
	英語	She had her hair dyed .		
X3 の 変換テーブル (手順 3)	*A*	英語	*B*	English
	C	髪	*D*	dyed

第3章　本論

3.1　提案手法(類似度の利用)

本研究は誤った変換テーブルの削除を目的とする. 変換テーブルは *A* と *C* や *B* と *D* の置き換えを想定している. 誤った変換テーブルは *A* と *C* や *B* と *D* の文中での置き換えが可能でない場合が多い. さらに類似度が高い単語及び句は置き換え可能性が高いと考える．そこで本手法は,*A* と *C* や *B* と *D* の類似度を閾値として誤った変換テーブルを削除する.

本研究において，類似度とは前後単語の一致度と定義する．類似度の計算は学習文中における単語や句の前後の単語を利用する．以下は提案手法の手順である. 手順 3 までは従来行われていた枝刈り手法である．なお, 変換テーブルの枝刈りは 2.3.4 節で述べた *ABCD* テーブルを対象とし, 提案手法を行なう.

手順 1 *A* と *B*，*C* と *D* の対訳単語確率を計算する

手順 2 得られた対訳単語確率をもとに順位をつける

手順 3 任意の対訳単語確率の順位を閾値として閾値を超えた変換テーブルを削除する

手順 4 *A* と *C*，*B* と *D* の類似度を計算する

手順 5 任意の類似度の値値を閾値として閾値を超えた変換テーブルを削除する

3.1.1 類似度の計算

二つの日本語単語 A，C の類似度の値を *sim(AC)* とすると *sim(AC)* は以下の式 3.1.1 で計算する．

$$sim(AC) = \log_2\{\frac{count(A_{context} \cap C_{context})}{count(A_{context})} \times \frac{count(A_{context} \cap C_{context})}{count(C_{context})}\}$$

count(X):集合 X の単語の総数 $A_{context}$,$C_{context}$:単語 A,C の前後単語の集合

計算は One-hot の word2vec に類似している．なお類似度の計算においては 2 単語連続を 1 単語として用いる．式 3.1.1 を英語単語 B と D でも同様の計算を行い *sim(BD)* とする．

3.1.2 類似度計算例 (日本語)

「犬」と「猫」という単語の類似度を表 3.1 の学習文を利用し計算する例を説明する.

表 3.1: 日本語学習文

日本語文 (1)	文頭 私 は **犬** を 飼っ て いる 。
日本語文 (2)	文頭 戸口 に **犬** が 見える 。
日本語文 (3)	文頭 私 は **猫** を 一 匹 飼っ て いる 。
日本語文 (4)	文頭 塀 の 上 に **猫** が いる 。

$$
\begin{aligned}
sim(\text{犬}, \text{猫}) &= \log_2\{\frac{count(\text{私は})}{count(\text{私は, を飼っ, 戸口に, が見える})} \\
&\times \frac{count(\text{私は})}{count(\text{私は, を一, 上に, がいる})}\} \\
&= \log_2\{\tfrac{1}{4} \times \tfrac{1}{4}\} \\
&= \log_2 \tfrac{1}{16} = -4
\end{aligned}
$$

第4章　実験

4.1　実験目的と方法

本実験の目的は誤った変換テーブルの削除である. 実験方法はTDSMTで作成した変換テーブルに3.1節を利用する. 評価は3.1で示した手順3まで行ったものを従来手法とし, 手順５まで行ったものを提案手法として比較する. 具体的には, 残った変換テーブルの精度によって提案手法を評価する. また, 精度は人手評価により評価する.

4.2 実験データ

表 4.1 に変換テーブル作成と選択に用いた学習文対の総数を示す.

表 4.1: 変換テーブル作成と選択に使用した学習文対の総数

学習文対　159,998 対

本実験は学習文対 15,9998 対を変換テーブル作成に使用する. また, 本実験で使用する学習文対は日本語文と英語文の対である. 使用する学習文対は電子辞書などの例文より抽出した単文データ [10] である. 学習文対の例を表 4.2 に示す.

表 4.2: 学習文対の例

学習文対例 (1)	
日本語文	ピアノ の 勉強 に ヨーロッパ に 行く 。
英語文	Go to Europe to study the piano .
学習文対例 (2)	
日本語文	公園 は 川 まで 広がっ て いる 。
英語文	The park reaches to the river .
学習文対例 (3)	
日本語文	きょう は 時折 小雪 の ちらつく 寒い 一 日 だっ た 。
英語文	It was a cold day today with occasional light snowfall .

4.2.1 評価方法

従来手法と提案手法を利用後の変換テーブルをランダムに 100 個抽出する．抽出した変換テーブルに対して人手で正解もしくは誤りで 2 値評価する．誤りと評価した数を誤り率とする．誤り率の低さを精度の高さとして評価する．

変換テーブルの評価は置き換え可能性の評価が難しい．よって，翻訳に用いる際最も重要となる *CD* の対訳関係の正誤を評価する．

評価基準を以下に示す．

正解 *C* と *D* の訳において助詞，冠詞を除いた単語の過不足がほぼ見られない

誤り *C* と *D* の訳において助詞，冠詞を除いた単語の過不足がある

4.2.2 閾値

今実験で用いた閾値を表 4.3 に示す．

表 4.3: 実験で用いた閾値

	対訳単語確率	類似度
従来手法	16 位	用いない
提案手法	16 位	-39.0

4.3 実験結果

4.3.1 変換テーブル精度

評価結果を表 4.4 に示す.

表 4.4: 従来手法と提案手法の誤り率

	提案手法	従来手法
変換テーブルの数	701,828	3,698,524
誤り率	3%	28%

提案手法を用いることで誤り率が 28%から 3%になった．よって, 提案手法を行なうことで変換テーブルの精度が向上することが確認できた. 一方で変換テーブルの数が約 81%減少した.

4.3.2 出力例

正解と評価した変換テーブルの例を 4.5,4.6 に示す．誤りと評価した変換テーブルの例を 4.7,4.8 に示す.

表 4.5: 提案手法における正解の出力

A	ドア	*B*	door
C	犬 の 首輪	*D*	collar of the dog
A 原文	ドア が 外れた。		
B 原文	The door has got unhinged.		
C 原文	犬 の 首輪 が 外れ た。		
D 原文	The collar of the dog has got loose.		
AB(順位)	1	*BA*(順位)	1
CD(順位)	2	*DC*(順位)	2
AC	-17.5	*BD*	-23.0

表 4.6: 提案手法における正解の出力

A	走る	*B*	run
C	猫 の 手 も かり たい	*D*	am extremely busy and short of hands
A 原文	私 は 走る。		
B 原文	I run.		
C 原文	私 は 猫 の 手 も かり たい。		
D 原文	I am extremely busy and short of hands.		
AB(順位)	2	*BA*(順位)	2
CD(順位)	1	*DC*(順位)	1
AC	-11.2	*BD*	-14.7

表 4.7: 提案手法における誤った出力

A	声	*B*	voice
C	鼻水	*D*	nose
A 原文	彼 は 声 が 詰まっ た 。		
B 原文	His voice choked .		
C 原文	彼 は 鼻水 が 出 た 。		
D 原文	His nose watered .		
AB(順位)	1	*BA*(順位)	1
CD(順位)	1	*DC*(順位)	7
AC	-23.2	*BD*	-25.3

表 4.8: 提案手法における誤った出力

A	走る	*B*	run
C	足 が 立た ない	*D*	cannot stand up in this swimming pool
A 原文	私 は 走る 。		
B 原文	I run .		
C 原文	この プール は 足 が 立た ない 。		
D 原文	I cannot stand up in this swimming pool .		
AB(順位)	2	*BA*(順位)	2
CD(順位)	2	*DC*(順位)	1
AC	-14.2	*BD*	-14.7

4.3.3 改善例

誤りと評価した変換テーブルを提案手法によって削除した例を 4.9,4.12 に示す.

表 4.9: 改善例

A	英語	*B*	English
C	あひる は 水かき	*D*	its webbed feet
A 原文	英語 の 勉強 を 怠け て いる 。		
B 原文	He is lazy in the study of English .		
C 原文	あひる は 水かき で 水 を 掻い て いる 。		
D 原文	The duck is paddling in the water with its webbed feet .		
AB(順位)	1	*BA*(順位)	1
CD(順位)	2	*DC*(順位)	2
AC	-47.2	*BD*	-47.9

表 4.10: 改善例

A	書く	*B*	Write
C	鳩 に 変え た	*D*	The magician changed
A 原文	文書 に 日付 を 書く 。		
B 原文	Write the date on a document .		
C 原文	魔術 師 は ハンカチ を 鳩 に 変え た 。		
D 原文	The magician changed the handkerchief into a dove .		
AB(順位)	3	*BA*(順位)	2
CD(順位)	1	*DC*(順位)	2
AC	-45.2	*BD*	-45.6

4.3.4 改悪例

正解と評価した変換テーブルを提案手法によって削除した例を 4.11,**??**に示す.

表 4.11: 改悪例

A	使える	*B*	useful
C	右腕 を ひどく 傷つけ られ た	*D*	badly injured on the right arm
A 原文	彼 は 使える 。		
B 原文	He is useful .		
C 原文	彼 は 右腕 を ひどく 傷つけ られ た 。		
D 原文	He was badly injured on the right arm .		
AB(順位)	1	*BA*(順位)	3
CD(順位)	1	*DC*(順位)	1
AC	-10.6	*BD*	-43.96

表 4.12: 提案手法における誤った出力

A	痛み	*B*	pain
C	左腕 の 痛み	*D*	pain in my left arm
A 原文	痛み が 和らい だ 。		
B 原文	The pain has eased .		
C 原文	左腕 の 痛み は ひどく なる 一方 だ 。		
D 原文	The pain in my left arm is getting worse .		
AB(順位)	1	*BA*(順位)	1
CD(順位)	2	*DC*(順位)	1
AC	-14.1	*BD*	-45.7

第5章　考察

5.1　単語と句に分けた調査

枝刈りした変換テーブルにおける単語と句の内訳を調査する．方法として，4.3 章で枝刈りした変換テーブルの CD に当たる部分が単語-単語，単語-句，句-単語，句-句でグループ分けする．

5.1.1 変換テーブルの数

グループごとの変換テーブルの数を表 5.1 に示す.

表 5.1: グループごとの変換テーブルの数

	枝刈り前	従来手法	提案手法
単語-単語	9,443,152	1,969,265	597,256
単語-句	1,960,034	147,918	25,787
句-単語	4,126,333	239,873	49,616
句-句	5,115,214	1,341,468	29,169

5.1.2 変換テーブルの精度

表 5.1 の各グループを評価した．グループごとにランダムに 100 個ずつ抽出して評価した．評価結果を表 5.11 に示す.

表 5.2: グループごとの誤り率

	枝刈り前	従来手法	従来手法
単語-単語	73%	9%	2%
単語-句	59%	10%	3%
句-単語	62%	5%	2%
句-句	55%	38%	11%

表 5.11 より，句-句のグループでは誤り率が 38%から 11%に向上した.

5.1.3 出力例

5.1 節における句の出力例を表 5.3,5.4 に示す．表 5.3,5.4 に示す変換テーブルはすべて従来手法で出力された変換テーブルである．表 5.3,5.4 においては変換テーブルの *CD* の部分のみ示す．提案手法の項は出力されたものを「出」，出力されないものを「無」と表記する．

表 5.3: 句の出力例

評価	*C*	*B*	提案手法
正解	犬 に ほえられ た	was barked at by the dog	出
正解	犬 の 首輪	collar of the dog	出
正解	犬 に かまれ た	was bitten by a dog	無
誤り	犬 が 吠え	hear a dog barking	無
誤り	犬 の 足跡	left its tracks	無
誤り	犬 や 猫	and cats	無

表 5.4: 句の出力例

評価	*C*	*B*	提案手法
正解	猫 の しっぽ を つかん だ	grasped the tail of a cat	出
正解	猫 を 締め出し た	shut the cat out	出
正解	猫 の 目 が 光っ た	The eyes of a cat flashed	無
誤り	猫 が 鼠	cat chasing a mouse	無
誤り	猫 は 餌 を 求め て 路地	alleys in search of food	無
誤り	猫 が ブロック 塀	concrete block wall	出

表 5.3,5.4 の例より，提案手法を用いることで誤った変換テーブルを削除できることを示した．

しかし，表 5.3 の 3 番目の例と，表 5.4 の 3 番目の例で示すように正解の変換テーブルを削除してしまう問題もある．また，表 5.4 の 6 番目の例で示すように誤った変換テーブルが一部残る問題もある．

5.1.4 変換テーブルの評価

今回は簡易的に評価を行うために *CD* の部分の訳が適切であるかで評価を行った．変換テーブルの構成を考えて *AB* の訳の適切さ，*AC*，*BD* の置き換え可能性も考慮に入れて評価する．そこで，4.3 章で用いた変換テーブルに注目して再度評価を行う．

5.1.5 評価結果

評価結果を表 5.5 に示す．

表 5.5: 従来手法と提案手法の誤り率

	従来手法	提案手法
誤り率	34%	3%

評価結果より，置き換え可能性を考慮した評価でも提案手法を用いた際の誤り率は従来手法を用いた際の誤り率より低い事が分かる．

5.1.6 評価例

5.1.4 章で評価した変換テーブルの例を示す．表 5.6 は正解と評価した例である．

表 5.6: 評価例

A	少し	*B*	little
C	調査	*D*	reseach
A 原文	その会社は調査部を設けた。		
B 原文	The firm has instituted a research department.		
C 原文	英語は少し話せます。		
D 原文	I speak a little English.		

表 5.6 において対訳関係は正しいと考える．ここで置き換え可能性を考える.「少し」は副詞であり,「調査」は名詞である．副詞は後ろに何を伴っても良い．よって *A* と *C* は置き換え可能性があると考えられる．

表 5.7 は正解と評価した例である．

表 5.7: 評価例

A	限り	*B*	limit
C	では	*D*	at
A 原文	地下資源には限りがある。		
B 原文	There is a limit to underground resources.		
C 原文	成功へはめったに一足飛びでは届かない。		
D 原文	Success is rarely reached at a single leap.		

表 5.6 と同様に表 5.7 も対訳関係は正しいと考える．同様に，置き換え可能性を考える．「で は」は主に接続助詞と係助詞の連語であり,「限り」は名詞である．接続助詞の前単語は名詞を取る．名詞は複合名詞の形で名詞につながる可能性がある．よって *A* と *C* は置き換え可能性があると考える．

表 5.6 と表 5.7 で示した例のように品詞が異なった変換テーブルの置き換え可能性を評価するのが難しい．よって安定的な評価を行うためには対訳関係のみで評価する方が良い．

5.1.7 パターンについて

変換テーブルの作成において，複数の変数によるパターンが用いられている．表5.8と5.9に変換テーブルとパターンの例を示す.

表5.8は正解と評価される変換テーブルである.

表5.8: パターン例

A	落ちる	*B*	fall
C	犬を激しくひっかいた	*D*	scratched wildly at the dog
A 原文	葉 | が | 落ちる。		
B 原文	The |leaves |fall.		
C 原文	猫 | が | 犬を激しくひっかいた。		
D 原文	The |cat |scratched wildly at the dog.		
日本語パターン	X2 X1 X3		
英語パターン	X1 X2 X3		

表5.8では「が」と「The」のように訳されない単語を仮定した上で，SVの形でパターンが生成され上手く対応されている．(ここで*D*の原文はSVO形ではあるがVとOを一つの句として扱っている)

表5.9は誤りと評価される変換テーブルである.

表5.9: パターン例

A	本	*B*	book
C	犬をひも	*D*	a leash
A 原文	本 | を | 買っ | た。		
B 原文	I |bought |a |book.		
C 原文	犬をひも | に | つないで | ください。		
D 原文	Put |your dog |on |a leash.		
日本語パターン	X4 X2 X3 X1		
英語パターン	X1 X3 X2 X4		

表5.9では*D*原文が命令文のため，動詞が先頭に来る．しかし*B*原文は通常のSVO形を持っているため同言語間で動詞の位置が異なってしまう．よって，英語のパターン作成において動詞が対応ついていない.

パターン作成において対応がついていない変換テーブルは誤った変換テーブルが多く見られる.

5.2 閾値を順位とした実験

4.3 章の実験では，閾値を類似度の値に設定した．5.2 章の実験では，閾値を類似度の値を昇順に順位付けした順位に設定する．

5.2.1 閾値

5.2 の実験では求められた類似度を昇順に順位付けする．さらに，類似度の順位を閾値とする．実験で用いた閾値を表 5.10 に示す.

表 5.10: 実験で用いた閾値

	対訳単語確率	類似度
従来手法	16 位	用いない
提案手法	16 位	16 位

5.2.2 実験結果

評価結果を表 5.11 に示す．表において手法 1 は 4.3 章における実験結果を示す．手法 2 は 5.3 章における実験結果を示す

表 5.11: 誤り率

	手法 1(値)	手法 2(順位)	従来手法
変換テーブルの数	701,828	1,101,173	3,698,524
誤り率	3%	25%	28%

5.11 より閾値を類似度の順位とする手法は従来手法と比べて誤り率が 28%から 25%へ向上している．一方で，閾値を類似度の値とする手法と比べると誤り率が高い．よって精度向上を目的とする場合，閾値は値に設定したほうが良い．しかし，この手法の利点として，変換テーブルの数の調整が比較的容易であることが考えられる．

5.2.3 出力例

閾値を順位としたとき誤った出力例を示す．を表 5.13 と**??**に示す.

表 5.12: 誤り例

A	恵まれ	*B*	favored
C	うそ つき の 汚名 を 着せ	*D*	branded him a
A 原文	彼 は 運 に 恵まれ た。		
B 原文	Luck favored him.		
C 原文	彼ら は 彼 に うそ つき の 汚名 を 着せ た。		
D 原文	They branded him a liar.		
AB(順位)	2	*BA*(順位)	1
CD(順位)	2	*DC*(順位)	1
AC(値)	-47.0	*BD*(値)	-79.7
AC(順位)	5	*BD*(順位)	1

表 5.13: 誤り例

A	壊れ	*B*	breaks
C	英語 は もちろん スペイン 語 さえ	*D*	can speak Spanish besides
A 原文	ガラス は 壊れ やすい。		
B 原文	Glass breaks easily.		
C 原文	彼女 は 英語 は もちろん スペイン 語 さえ 話せる。		
D 原文	She can speak Spanish besides English.		
AB(順位)	2	*BA*(順位)	4
CD(順位)	1	*DC*(順位)	2
AC(値)	-79.7	*BD*(値)	-79.7
AC(順位)	1	*BD*(順位)	3

5.3 変換テーブルの数を増加させた実験

4.3 章の実験では，枝刈り後の変換テーブルの数が従来手法と提案手法で大きく異なる問題があった．5.3 章では，変換テーブルの数を提案手法と従来手法で同程度に揃えた上で精度を調査する．閾値の条件を変更することで変換テーブルの数を増加させた．そして，提案手法と従来手法を比較する．

5.3.1 閾値

5.3 の実験では求められた類似度を昇順に順位付けする．さらに，類似度の順位を閾値とする．実験で用いた閾値を表 5.14 に示す．

表 5.14: 実験で用いた閾値

	対訳単語確率	類似度
従来手法	16 位	用いない
提案手法	512 位	2048 位

5.3.2 実験結果

評価結果を表 5.15 に示す.

表 5.15: 従来手法と提案手法の誤り率

	提案手法	従来手法
変換テーブルの数	2,993,145	3,698,524
誤り率	20%	28%

表 5.15 の結果より，枝刈り条件を大きくした際も，提案手法による変換テーブルの精度向上が確認できた．提案手法を用いることにより誤り率が 28%から 20%に向上する．

5.3.3 出力例

誤りと評価した変換テーブルを提案手法によって削除した例を表 5.16 と 5.17 に示す.

表 5.16: 提案手法による変換テーブルの改善例

A	炒っ	*B*	roasted
C	し	*D*	with
A 原文	豆をこんがりと炒った。		
B 原文	He roasted the beans brown .		
C 原文	平手でその子を強く打とうとした。		
D 原文	He swiped at the child with his open hand .		
AB(順位)	1	*BA*(順位)	1
CD(順位)	14	*DC*(順位)	11
AC(順位)	91,907	*BD*(順位)	38,529

表 5.17: 提案手法における変換テーブルの改善例

A	あけ	*B*	Open
C	し	*D*	He
A 原文	口 を 大きく あけ なさい 。		
B 原文	Open your mouth wide .		
C 原文	会議 を 中座 し た 。		
D 原文	He left the meeting halfway through .		
AB(順位)	2	*BA*(順位)	4
CD(順位)	2	*DC*(順位)	10
AC(順位)	67522	*BD*(順位)	51169

この節の例については 5.3.4 章で考察する

5.3.4 品詞分解の問題

5.3 章で得た出力例を考察する.

例と同様に，*C* が「し」となる変換テーブルは従来手法で枝刈りをした場合 6482 個存在する．一方提案手法で枝刈りをすると 44 個となる．中身として従来手法では「し」の訳として「with」や「to」などが多くあるのに対し，提案手法では「did」や「has」,「made」などが列挙される.

日本語において「し」は動詞に付属して「〜 し た」という 3 単語の形で用いられることが多い．しかし，英語では「〜 し た」を 1 単語の動詞の過去形で訳される場合が多い.

また，文において「し」の対訳単語はない場合も多い．よって「し」の様な単語において間違った変換テーブルを作成してしまう場合がある．提案手法では C と D の類似度を用いて枝刈りを行うので多くの間違った変換テーブルを削除できると考える．

第6章　おわりに

機械翻訳において, 相対的意味論に基づく変換主導型統計機械翻訳 TDSMT が提案されている. TDSMT は変換テーブルを用いて, 学習文対を変換し, 出力文を作成する. 変換テーブルは学習文対 (パラレルコーパス) から自動作成する. しかし, 自動作成のため誤った変換テーブルが存在する. そこで, 本研究は誤った変換テーブルの削除を目的とした.

本研究では提案手法として, 同言語間の類似度を利用して誤った変換テーブルを削除する手法を提案した. 提案手法によって変換テーブルの精度の向上を確認した.

今後の課題として, 適切な閾値の策定がある.

謝辞

本研究を進めるにあたり, 研究の説明や論文の書き方など様々なご指導を頂きました鳥取大学工学部電気情報系工学科自然言語処理研究室の村上仁一准教授に心から御礼申し上げます. また, 本研究を進めるにあたり, 御指導, 御助言を頂きました, 村田真樹教授に心から御礼申し上げます. また, 同じ班に所属されていた金子先輩, 中村先輩をはじめとする自然言語処理研究室の皆様へ心から感謝の気持ちと御礼を申し上げたく謝辞にかえさせて頂きます.

参考文献

[1] 安場裕人, 村上仁一 (2018). “変換主導型翻訳の提案”. 自然言語処理学会第 24 回年次大会.

[2] Philipp Koehn (2005). “Europarl: A Parallel Corpus for Statistical Machine Translation”. *MT Summit*, pp.79?86.

[3] Peter F.Brown, Stephen A.Della Pietra, Vincent J.Della Pietra, Robert L.Mercer (1993). “The mathematics of statistical machine translation: Parameter Estimation”. *Computational Linguistics*.

[4] 西尾聡一郎 (2016). “パターンに基づく統計翻訳における文パターン確率の考察”. 平成 27 年度 卒業論文, pp.3-16.

[5] 力久 剛士 (2015). “レーベンシュタイン距離を用いた翻訳精度の向上”. 平成 26 年度 卒業論文, pp.3-15

[6] Philipp Koehn, Marcello Federico, Brooke Cowan, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, Evan Herbst (2007). “Moses: Open Source Toolkit for Statistical Machine Translation”. *Proceedings of the ACL 2007 Demo and Poster Sessions*, pp.177-180.

[7] 中村 勇太 (2019). “ 相対的意味論に基づく変換主導型パターンベース統計機械翻訳（TDPBSMT）の提起”. 平成 30 年度 卒業論文.

[8] Franz Josef Och, Hermann Ney (1996). “A Systematic Comparison of Various Statistical Alignment Models”. *Computational Linguistics*, 29(1), pp.299-314.

[9] Zellig S.Harris (1954). “Distributional structure”. *Word*, Vol. 10, No.23, pp.146-162.

[10] 村上仁一, 藤波進 “日本語と英語の対訳文対の収集と著作権の考察 ”, 第一回コーパス日本語学ワークショップ, pp.119-130. 2012.