

# 概要

現在，機械翻訳の分野において，対訳文対から自動的に翻訳規則を生成し翻訳を行う統計翻訳が注目され，研究が盛んに行われている．統計翻訳では英語-イタリア語など文法構造が類似する言語対において翻訳精度が高くなり，日本語-英語などの文法構造が異なる言語対においては翻訳精度が低くなる傾向がある．別の翻訳手法にパターン翻訳がある．パターン翻訳では文パターン辞書と単語辞書を用いて翻訳を行う．文パターンが有する大局的な文法情報を用いることで翻訳文全体の構造を保持した翻訳精度の高い翻訳文を生成できる利点がある．しかし，従来，文パターン辞書の作成は人手で行うため，開発にコストがかかる欠点がある．

そこで本研究では，文パターン辞書を対訳文対から自動的に作成する手法を検討する．文パターン辞書の自動作成により，開発にかかるコストの削減が可能となる．また，パターン翻訳を統計翻訳の前処理に用いることで，日本語の文法構造を英語に類似し翻訳精度が向上すると考えられる．

従来の日英統計翻訳をベースラインとして，ベースラインと提案手法の翻訳精度の調査を行い，結果を比較する．実験は，単文を用いた翻訳実験と，重文複文を用いた翻訳実験の2種類を行う．評価方法には自動評価法と人手による対比較評価を行う．自動評価法にはBLEU，NIST，METEORの3種類を用いる．人手による対比較評価については，提案手法の翻訳結果がベースラインの翻訳結果よりも優れていた場合，提案手法の翻訳結果がベースラインの翻訳結果よりも劣っていた場合，どちらの翻訳結果も文質に差がなかった場合，同一出力であった場合の4つの基準で評価を行う．

実験の結果，単文の実験で，提案手法はベースラインと比較して，自動評価でBLEU値が0.29%，NIST値が0.008低下した．METEOR値は0.15%向上した．人手による対比較評価では，提案手法の翻訳結果がベースラインの翻訳結果よりも優れていた文が30文であり，反対に劣っていた文では9文であった．また，重文複文の実験では，提案手法はベースラインと比較して自動評価でBLEU値が0.3%，NIST値が0.0526，METEOR値が0.0.28%と全ての評価基準で向上した．また対比較評価においても，提案手法の翻訳結果がベースラインの翻訳結果よりも優れていた文が33文であり，反対に劣っていた文では3文であった．

以上の結果から，単文では，自動評価法において提案手法はベースラインと比較して性能に差がない結果となっている，しかし，対比較評価においては，ベースラインよりも優れていた文が明らかに多い．また，重文複文では，自動評価法と対比較評価のどち

らにおいても提案手法はベースラインよりも優れた結果となっている。したがって、本研究の提案手法の有効性が確認できた。

# 目次

第1章	はじめに	1
第2章	日英パターン翻訳システム	2
2.1	パターン翻訳の概要	2
2.2	パターン翻訳の手順	2
2.3	日英文パターン辞書	3
2.4	意味属性制約	3
第3章	日英統計翻訳システム	5
3.1	統計翻訳の概要	5
3.2	統計翻訳の手順	5
3.3	翻訳モデル	6
3.3.1	翻訳モデルの概要	6
3.3.2	IBM 翻訳モデル	6
3.3.2.1	モデル1	7
3.3.2.2	モデル2	9
3.3.2.3	モデル3	10
3.3.2.4	モデル4	11
3.3.2.5	モデル5	12
3.4	GIZA++	12
3.5	フレーズテーブル作成法	13
3.6	言語モデル	17
3.6.1	言語モデルの概要	17
3.6.2	$N$ -gram モデル	18
3.7	デコーダ	20
3.7.1	ビームサーチ法	20
3.7.2	マルチスタック法	21

3.8	パラメータチューニング	21
3.9	評価方法	23
3.9.1	BLEU	23
3.9.2	NIST	24
3.9.3	METEOR	25
<b>第4章</b>	<b>翻訳システムの自動作成</b>	<b>26</b>
4.1	本研究の翻訳システムの概要	26
4.2	本研究の翻訳システムの手順	27
4.3	ベースラインシステム	31
<b>第5章</b>	<b>実験環境</b>	<b>32</b>
5.1	実験データ	32
5.2	翻訳モデルの学習	33
5.3	言語モデルの学習	33
5.4	デコーダのパラメータ	33
<b>第6章</b>	<b>翻訳実験</b>	<b>34</b>
6.1	提案手法におけるパターン翻訳	34
6.2	パターン翻訳部の実験結果	34
6.3	提案手法における統計翻訳	35
6.4	ベースラインシステムにおける実験	35
<b>第7章</b>	<b>実験結果</b>	<b>36</b>
7.1	自動評価結果	36
7.2	人手による評価	37
7.2.1	判断基準	37
7.2.2	評価結果	38
<b>第8章</b>	<b>考察</b>	<b>39</b>
8.1	翻訳精度が高い出力の解析	39
8.2	誤り解析	41
8.2.1	適合したパターンが不適切であった場合	42
8.2.2	フレーズ対に問題があった場合	43

8.2.3	統計翻訳が悪影響を及ぼしている場合 . . . . .	44
8.2.4	単語の意味の違いによって誤った翻訳をしている場合 . . . . .	45
第9章	おわりに	46

# 目次

2.1	日英パターン翻訳の手順 . . . . .	2
2.2	意味属性制約の例 . . . . .	4
3.1	日英統計翻訳の手順 . . . . .	6
3.2	アライメントの例 . . . . .	7
3.3	デコーダの探索例 . . . . .	20
3.4	マルチスタック・ビームサーチ法の適用例 . . . . .	21
4.1	提案手法の枠組み . . . . .	26
4.2	日英対訳フレーズ辞書の作成手順 . . . . .	27
4.3	日英文パターン辞書の作成手順 . . . . .	28
4.4	日英パターン翻訳の流れ . . . . .	29
4.5	モデル学習の例 . . . . .	30
4.6	英' 英統計翻訳の例 . . . . .	30

# 表 目 次

2.1	日英文パターンの例 . . . . .	3
3.1	フレーズテーブルの例 . . . . .	6
3.2	日英方向の単語対応 . . . . .	13
3.3	英日方向の単語対応 . . . . .	13
3.4	intersection の例 . . . . .	14
3.5	union の例 . . . . .	14
3.6	grow の例 . . . . .	15
3.7	grow-diag の例 . . . . .	15
3.8	grow-diag-final の例 . . . . .	16
3.9	grow-diag-final-and の例 . . . . .	16
3.10	grow-diag-final-and で作成されたフレーズテーブルの例 . . . . .	16
3.11	言語モデルの例 . . . . .	17
5.1	単文の例 . . . . .	32
5.2	重文複文の例 . . . . .	32
6.1	各辞書のエントリ数 . . . . .	34
6.2	パターン翻訳の出力文数 . . . . .	34
7.1	10,000 文での自動評価結果 . . . . .	36
7.2	パターン翻訳を通じた出力文での自動評価結果 . . . . .	37
7.3	対比較評価の結果 . . . . .	38
8.1	短文での提案手法が優れている翻訳例 . . . . .	39
8.2	重文複文での提案手法が優れている翻訳例 . . . . .	39
8.3	対比較評価の結果 . . . . .	41
8.4	適合した文パターンが不適切であった場合の翻訳例 . . . . .	42

8.5	フレーズ対に問題があった場合の翻訳例 . . . . .	43
8.6	統計翻訳が悪影響を及ぼした場合の翻訳例 . . . . .	44
8.7	単語の意味の違いによって誤った翻訳をしていた場合の翻訳例 . . . . .	45



# 第1章 はじめに

機械翻訳の歴史は文法規則や変換規則などを用いて翻訳を行うルールベース翻訳から始まる。この手法では、日本語と英語のルールを定義する必要がある。例えば、日本語のルール  $[S \rightarrow NP \text{ は } VP]$  がある場合には、英語のルール  $[S \rightarrow NP VP]$  といった、ルールの対が大量に必要となる。したがって、翻訳するためのルールを作成するには、多言語化が困難であり、開発に時間がかかる。そして1960年代半ばに、大量の翻訳対から作成した文パターン辞書を用いて翻訳を行うパターン翻訳 [1] が提案される。ルールと同様に、パターン辞書も人手で作成するので、開発に時間がかかる [2] が、文パターンに適合した場合に翻訳精度の高い翻訳文が得られる。1990年代前半に「語に基づく統計翻訳」が提案されたが、翻訳精度が低くあまり研究がされなかった。しかし、2000年の初めに「句に基づく統計翻訳 [3]」が提案され、「語に基づく統計翻訳」と比べて翻訳精度が高いことから、現在、機械翻訳において統計翻訳が主流となっている。統計翻訳では英語-イタリア語など文法構造が類似する言語対において翻訳精度が高くなり、日本語-英語などの文法構造が異なる言語対においては翻訳精度が低くなる傾向がある [4]。過去に、英語の文法構造を日本語の文法構造に類似させる研究 [5] がされており、翻訳精度の向上が報告されている。

そこで本研究では、パターン翻訳の利点と統計翻訳の利点を活かすことを考え、文パターン辞書を対訳文対から自動的に作成する手法を検討する。文パターン辞書の自動作成により、開発にかかるコストの削減が可能となる。また、パターン翻訳を統計翻訳の前処理に用いることで、日本語の文法構造を英語に類似し翻訳精度が向上すると考えられる。

本論文の構成は以下の通りである。第2章で従来の日英パターン翻訳システムについて説明し、第3章で従来の日英統計翻訳システムについて説明し、第4章では本研究で提案する翻訳システムについて説明する。第5章で実験環境を説明し、第6章で翻訳実験について述べ、第7章で実験結果を示し、第8章で本研究の考察を述べる。

## 第2章 日英パターン翻訳システム

### 2.1 パターン翻訳の概要

パターン翻訳とは、文パターン辞書と単語辞書を用いて、日本語文パターンに適合した情報を対応する英語文パターンに用いて適切な翻訳文を生成する、機械翻訳手法の1つである。パターン翻訳には文パターンが有する大局的な文法情報を用いることで英語文全体の構造を保持した翻訳精度の高い翻訳文を生成できるという利点がある。しかし、従来の文パターン辞書の作成は人手で行うため、開発に時間がかかるという欠点がある。また、文パターン辞書に適合しない入力文については出力を行うことができないため、入力文に対するカバー率が低い問題もある。

### 2.2 パターン翻訳の手順

日英パターン翻訳は、まず日本語入力文  $j$  が与えられたとき、日英文パターン辞書と日英単語辞書を参照する。次に、日本語入力文に適合した日本語文パターンに対応する英語文パターンの変数部を単語に置き換えることで翻訳を行う。図 2.1 に日英パターン翻訳の手順を示す。

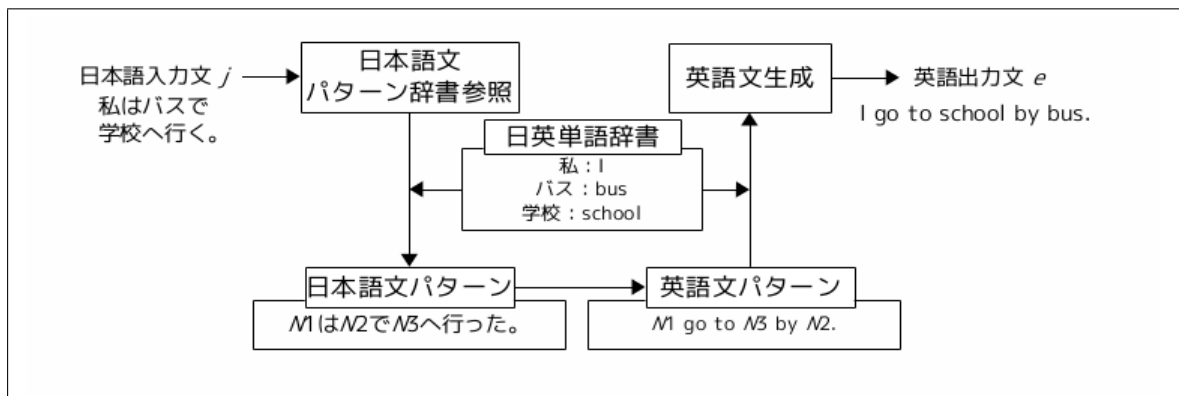


図 2.1 日英パターン翻訳の手順

## 2.3 日英文パターン辞書

日英文パターン辞書とは，大量の対訳コーパスから対応する単語を変数により置き換えることで得られる文パターンのデータベースである．

過去の研究に，単文での文パターン辞書作成が行われている [6]．作成方法としては，大量の対訳コーパスと対訳単語辞書を用いて，対となる名詞などを変数化する．また翻訳時の変数部を単語に置き換える際，置き換える単語の候補をしぼるために，品詞や字面などの意味属性制約を変数に付与する．この過程により，文パターン辞書を作成する．表 2.1 に [6] で作成された文パターンの例を示す．

表 2.1 日英文パターンの例

日本語原文	彼はその戦闘で傷を受けた。
英語原文	He was wounded in that battle.
日本語文パターン	<i>PRO1</i> は <i>REN2N3</i> で傷を受けた。
英語文パターン	<i>PRO1</i> was wounded in <i>REN2 N3</i> .

表 2.1 の例では，*PRO* は代名詞，*REN* は連体詞，*N* は名詞を表しており，翻訳時に置き換える単語が限定される．しかし，第 4 章で詳しく説明する，本研究で作成する文パターンではカバー率を向上するために，意味属性制約などの制約は用いない．

## 2.4 意味属性制約

[2] では，日本語の用言 (約 6,000 語) の文パターン (約 14,000 件) が定義されている．文パターンの格要素は，一般名詞意味属性によって制限された名詞と格助詞を組み合わせで構成されている．格要素の使い方により，用言の意味に違いが生じるので，一つの用言に対して複数の文パターンが存在している．そのため，格要素を構成する名詞は変数化されており，一般名詞意味属性による制約条件が付与されている．

一般名詞意味属性は，単語を意味的用法により階層的に分類，体系化したものである．約 40 万単語を 12 段の木構造によって，2,710 の意味属性に分類している．一般名詞意味属性は，上位の意味属性の性質を下位の意味属性に継承している．したがって，制約条件として中位の意味属性が与えられている場合でも，その意味属性の下位にあたる名詞は制約を満たす．また，用言意味属性は「用言が持つ動的属性の種類」の観点から分類した 36 種類の属性体系である．このことから，文パターンによる意味解析ができるとい

える．図 2.2 に「愛する」についての文パターンとその文パターンにおける意味属性制約を示す．

愛する	
(1)	$N1$ が $N2$ を 愛する $N1$ love $N2$ $N1\{4 \text{ 人}\}$ $N2\{*\}$ 31 感情動作
(2)	$N1$ が $N2$ を 愛する $N1$ love $N1$ 's $N2$ $N1\{4 \text{ 人}\}$ $N2\{72 \text{ 人間 (親族関係)}\}$ 31 感情動作

図 2.2 意味属性制約の例

(1) と (2) は同じ日本語文パターンであるが，名詞変数  $N2$  には意味属性制約によって異なる名詞が適合する．(1) の文パターンの  $N2$  には全ての名詞に対して適合可能である．一方，(2) の  $N2$  では，親族関係にある名詞のみに適合可能である．

## 第3章 日英統計翻訳システム

### 3.1 統計翻訳の概要

統計翻訳とは、翻訳する言語と目的言語の対訳文を大量に収集した対訳データを用いて、自動的に翻訳規則を獲得し翻訳を行う、機械翻訳手法の1つである。統計翻訳では、統計的モデルを学習するアルゴリズムは言語に依存しない。したがって、大量の対訳データがあれば多言語化が容易であり、開発時間も抑えることが可能である。統計翻訳には、最初に提案された「語に基づく統計翻訳」と近年提案された「句に基づく統計翻訳」があり、現在は「句に基づく統計翻訳」が主流となっている。これは、語順の並び替えや文脈における訳語の選択や翻訳精度において、「句に基づく統計翻訳」の方が「語に基づく統計翻訳」よりも優れているためである。また統計翻訳の特徴として、文法構造が似ている言語間では翻訳精度が高くなる傾向があり、文法構造の異なる言語間では翻訳精度が低くなる傾向がある。

### 3.2 統計翻訳の手順

日英統計翻訳は、日本語入力文  $j$  が与えられたとき、全ての組合せから確率が最大となる英語文  $\hat{e}$  を探索して翻訳を行う。

$$\hat{e} = \arg \max_e P(e|j) \quad (3.1)$$

$$\approx \arg \max_e P(j|e)P(e) \quad (3.2)$$

$P(j|e)$  は翻訳モデル、 $P(e)$  は言語モデルと呼ぶ。図 3.1 に統計翻訳の手順を示す。

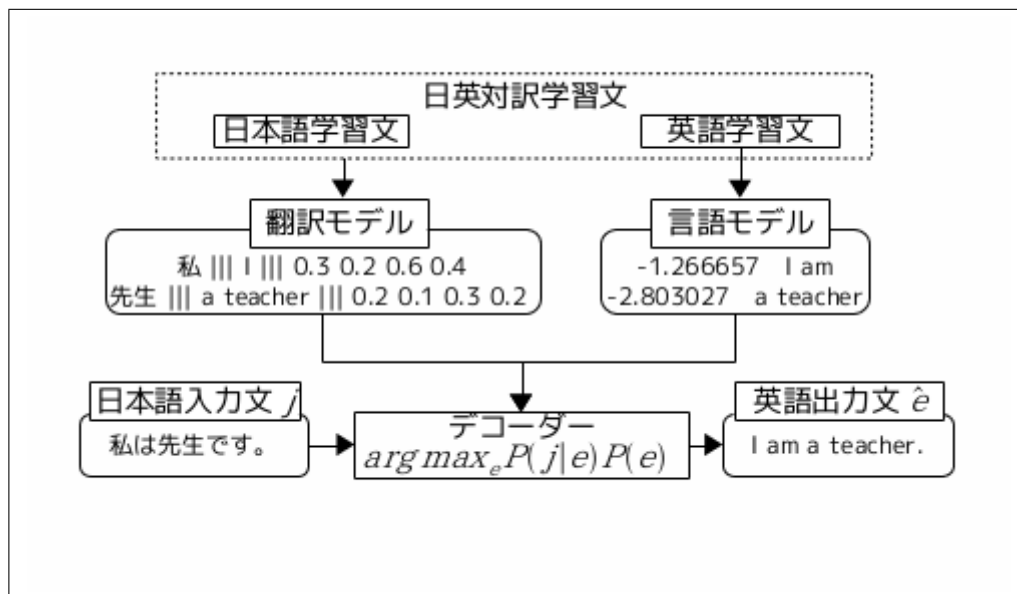


図 3.1 日英統計翻訳の手順

### 3.3 翻訳モデル

#### 3.3.1 翻訳モデルの概要

翻訳モデルは英語から日本語の単語列へ確率的に翻訳を行うためのモデルである。翻訳モデルはフレーズテーブルで管理されている。フレーズテーブルの例を表 3.1 に示す。

表 3.1 フレーズテーブルの例

ここ	から		from	here		0.7778	0.1407	0.9545	0.2571
すぐに			at	once		0.5	0.0111	0.1964	0.003

左から、日本語フレーズ、英語フレーズ、フレーズの日英方向の翻訳確率、日英の単語翻訳確率の積、フレーズの英日方向の翻訳確率、英日の単語翻訳確率の積である。

#### 3.3.2 IBM 翻訳モデル

統計翻訳の翻訳モデルとして代表的なものに、Brown らが提案した IBM の仏英翻訳モデル [7] がある。この翻訳モデルは提案者である Brown らが全員 IBM の社員であったため、IBM 翻訳モデルと呼ばれている。この翻訳モデルは順に複雑な計算を行うモデル 1

からモデル5の5つのモデルから成る．IBM 翻訳モデルではフランス語から英語への翻訳を想定しているため，フランス語文を  $F$ ，英語文を  $E$  として説明を行う．

IBM モデルでは英語文  $E$  とフランス語文  $F$  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 モデルの基本的な計算式を示す．

$$P(F|E) = \sum_a P(F, a|E) \quad (3.3)$$

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．図 3.2 にアライメントの例を示す．

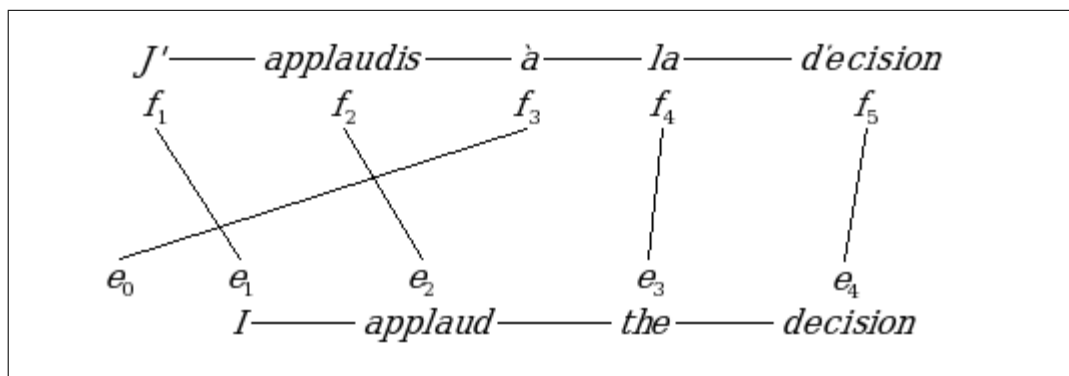


図 3.2 アライメントの例

### 3.3.2.1 モデル1

(3.3) 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．

$$P(F, a|E) = P(m|E) \prod_{j=1}^m P(a_j | a_1^{j-1}, f_1^{j-1}, m, E) P(f_j | a_1^j, f_1^{j-1}, m, E) \quad (3.4)$$

(3.4) 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  $\epsilon$  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。

$$P(F, a|E) = \frac{\epsilon}{(l+1)^m} \prod_{j=1}^m t(f_j|e_{a_j}) \quad (3.5)$$

$$P(F|E) = \frac{\epsilon}{(l+1)^m} \sum_{a_1=0}^l \cdots \sum_{a_m=0}^l \prod_{j=1}^m t(f_j|e_{a_j}) \quad (3.6)$$

$$= \frac{\epsilon}{(l+1)^m} \prod_{j=1}^m \sum_{i=0}^l t(f_j|e_{a_j}) \quad (3.7)$$

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  $e$  に対応する回数の期待値を以下の式により計算する。

$$c(f|e; F, E) = \frac{t(f|e)}{t(f|e_0) + \cdots + t(f|e_l)} \sum_{j=1}^m \delta(f, f_j) \sum_{i=0}^l \delta(e, e_i) \quad (3.8)$$



$\delta(f, f_j)$  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  $\lambda_e$  を以下の式により計算する.

$$\lambda_e = \sum_f \sum_{s=1}^S c(f|e; F^{(s)}, E^{(s)}) \quad (3.9)$$

2. (3.9) 式より求めた  $\lambda_e$  を用いて, 翻訳確率  $t(f|e)$  を再計算する.

$$\begin{aligned} t(f|e) &= \lambda_e^{-1} \sum_{s=1}^S c(f|e; F^{(s)}, E^{(s)}) \\ &= \frac{\sum_{s=1}^S c(f|e; F^{(s)}, E^{(s)})}{\sum_f \sum_{s=1}^S c(f|e; F^{(s)}, E^{(s)})} \end{aligned} \quad (3.10)$$

### 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.

#### 3.3.2.2 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.

$$a(a_j|j, m, l) \equiv P(a_j|a_1^{j-1}, f_1^{j-1}, m, l) \quad (3.11)$$

この関係からモデル 1 における (3.6) 式は, 以下の式に変換できる.

$$P(F|E) = \epsilon \sum_{a_1=0}^l \cdots \sum_{a_m=0}^l \prod_{j=1}^m t(f_j|e_{a_j}) a(a_j|j, m, l) \quad (3.12)$$

$$= \epsilon \prod_{j=1}^m \sum_{i=0}^l t(f_j|e_{a_j}) a(a_j|j, m, l) \quad (3.13)$$

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。

$$c(f|e; F, E) = \frac{t(f|e)}{t(f|e_0) + \cdots + t(f|e_l)} \sum_{j=1}^m \delta(f, f_j) \sum_{i=1}^l \delta(e, e_i) \quad (3.14)$$

$$= \sum_{j=1}^m \sum_{i=0}^l \frac{t(f|e) a(i|j, m, l) \delta(f, f_j) \delta(e, e_i)}{t(f|e_0) a(0|j, m, l) + \cdots + t(f|e_l) a(l|j, m, l)} \quad (3.15)$$

$$c(i|j, m, l; F, E) = \sum_a P(a|E, F) \delta(i, a_j) \quad (3.16)$$

$$= \frac{t(f_j|e_i) a(i|j, m, l)}{t(f_j|e_0) a(0|j, m, l) + \cdots + t(f_j|e_l) a(l|j, m, l)} \quad (3.17)$$

$c(f|e; F, E)$  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。

### 3.3.2.3 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  $\phi$  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  $\phi_0$  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 .

$$P(\phi_0|\phi_1^l, E) = \binom{\phi_1 + \dots + \phi_l}{\phi_0} p_0^{\phi_1 + \dots + \phi_l - \phi_0} p_1^{\phi_0} \quad (3.18)$$

したがって , モデル 3 は以下の式で求められる .

$$P(F|E) = \sum_{a_1=0}^l \dots \sum_{a_m=0}^l P(F, a|E) \quad (3.19)$$

$$= \sum_{a_1=0}^l \dots \sum_{a_m=0}^l \binom{m - \phi_0}{\phi_0} p_0^{m - 2\phi_0} p_1^{\phi_0} \prod_{i=1}^l \phi_i! n(\phi_i|e_i) \\ \times \prod_{j=1}^m t(f_j|e_{a_j}) d(j|a_j, m, l) \quad (3.20)$$

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 .

### 3.3.2.4 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

$$P(\Pi_{[i]1} = j | \pi_1^{[i]-1}, \tau_0^l, \phi_0^l, E) = d_1(j - \odot_{i-1} | \mathcal{A}(e_{[i-1]}), \mathcal{B}(f_j)) \quad (3.21)$$

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

- それ以外の場合

$$P(\Pi_{[i]k} = j | \pi_{[i]1}^{k-1}, \pi_1^{[i]-1}, \tau_0^l, \phi_0^l, E) = d_{>1}(j - \pi_{[i]k-1} | \mathcal{B}(f_j)) \quad (3.22)$$

$\pi_{[i]k-1}$  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 .

### 3.3.2.5 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

$$\begin{aligned} P(\Pi_{[i]1} = j | \pi_1^{[i]-1}, \tau_0^l, \phi_0^l, E) \\ = d_1(v_j | \mathcal{B}(f_j), v_{\odot_{i-1}}, v_m - \phi_{[i]} + 1)(1 - \delta(v_j, v_{j-1})) \end{aligned}$$

$v_j$  は  $j$  番目までの空白数,  $A$  は英語の単語クラス  $B$  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

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

## 3.4 GIZA++

GIZA++[8] とは, 統計翻訳で用いることを前提に作られた単語対応のアライメントを行うツールである . IBM モデル1~5を学習し, 単語の対応関係の確率値を計算する .

### 3.5 フレーズテーブル作成法

まず，GIZA++を用いて学習文から日英，英日方向の双方向で最尤な単語アライメントを得る．日英方向の単語対応の例を表 3.2，英日方向の単語対応の例を表 3.3 に示す．尚，表中の●は得られた単語アライメントを示す．

表 3.2 日英方向の単語対応

	I	was	dazzled	by	the	headlights	of	a	car
自動車			●						
の							●		
ヘッドライト						●			
に							●		
目				●					
が								●	
くらん									●
だ									●

表 3.3 英日方向の単語対応

	I	was	dazzled	by	the	headlights	of	a	car
自動車	●								
の							●		
ヘッドライト		●				●			
に									
目					●				
が								●	
くらん				●					●
だ			●						

次に，得られた双方向の単語アライメントを用いて，複数単語のアライメントを得る．このアライメントは双方向の単語対応の和集合と積集合から求める．ヒューリスティクスとして双方向ともに対応する単語対応を用いる“intersection”，双方向のどちらか一方でも対応する単語対応を全て用いる“union”がある．表 3.2 と表 3.3 を用いた，“intersection”での例を表 3.4，“union”での例を表 3.5 に示す．

表 3.4 intersection の例

	I	was	dazzled	by	the	headlights	of	a	car
自動車									
の							•		
ヘッドライト						•			
に									
目									
が								•	
くらん									•
だ									

表 3.5 union の例

	I	was	dazzled	by	the	headlights	of	a	car
自動車	•								
の							•		
ヘッドライト		•				•			
に							•		
目				•	•				
が								•	
くらん				•					•
だ			•						•

また “intersection” と “union” の中間のヒューリスティックスとして “grow” と “grow-diag” がある . これら 2 つのヒューリスティックスでは “intersection” の単語対応と “union” の単語対応を用いる . “grow” は縦横方向 , “grow-diag” は縦横対角方向に , “intersection” の単語対応から “union” の単語対応が存在する場合にその単語対応も用いる . “grow” の例を表 3.6 , “grow-diag” の例を表 3.7 に示す .

表 3.6 grow の例

	I	was	dazzled	by	the	headlights	of	a	car
自動車									
の							•		
ヘッドライト						•			
に									
目									
が								•	
くらん									•
だ									•

表 3.7 grow-diag の例

	I	was	dazzled	by	the	headlights	of	a	car
自動車									
の							•		
ヘッドライト						•			
に							•		
目									
が								•	
くらん									•
だ									•

また、この“grow”と“grow-diag”の最後に行う処理として“final”と“final-and”がある。“final”は“union”の単語対応があれば用いる。“final-and”では、“final”に加えて、双方向ともに単語対応がないアライメントも用いる。“grow-diag-final”の例を表3.8，“grow-diag-final-and”の例を表3.9に示す。

表 3.8 grow-diag-final の例

	I	was	dazzled	by	the	headlights	of	a	car
自動車	•								
の							•		
ヘッドライト		•				•			
に							•		
目					•				
が								•	
くらん				•					•
だ			•						•

表 3.9 grow-diag-final-and の例

	I	was	dazzled	by	the	headlights	of	a	car
自動車	•								
の							•		
ヘッドライト						•			
に							•		
目					•				
が								•	
くらん									•
だ									•

そして、得られた単語アライメントから、全ての矛盾しないフレーズ対を得る。このとき、そのフレーズ対に対して翻訳確率を計算し、フレーズ対に確率値を付与する。“grow-diag-final-and”で作成されたフレーズテーブルの例を表 3.10 に示す。

表 3.10 grow-diag-final-and で作成されたフレーズテーブルの例

自動車	I was dazzled by	1	0.333333	0.25	0.000751315	2.718
自動車	I was dazzled	1	0.333333	0.25	0.00826446	2.718
自動車	I was	1	0.333333	0.25	0.0909091	2.718
自動車	I	0.333333	0.333333	0.25	1	2.718
がくらんだ	a car	1	0.166667	1	0.666667	2.718
がくらんだ。	a car .	1	0.166667	1	0.666667	2.718



## 3.6 言語モデル

### 3.6.1 言語モデルの概要

言語モデルは単語列の生じる確率を与えるモデルである。日英翻訳では、翻訳モデルで生成された翻訳候補から英語として自然な文を選出する。統計翻訳では一般に、 $N$ -gram モデルを用いる。表 3.11 に言語モデル (2-gram) の例を示す。

表 3.11 言語モデルの例

	...	
-1.941598	7 billion	-0.8050393
-0.3828715	Almost all	-0.2935186
-3.080696	be silent	-0.7815382
-3.256787	be similar	-0.5834416
-1.527517	both hands	-0.05090702
-1.355388	characters with	-0.545048
-1.665734	few hours	-0.7934405
-1.392733	few minutes	-0.1146355
-2.38916	her house	-0.06943518
-3.250479	his birthday	-0.1799351
-3.588309	is designed	-0.03691489
-0.5900388	located in	-0.2151646
	...	

表の 1 行目の例では、左の数値は 7 の後に *billion* が来る確率  $P(\textit{billion}|\textit{7})$  を常用対数  $\log_{10}$  でとった値  $\log_{10} P(\textit{billion}|\textit{7})$  である。次に、2-gram で表された単語列 *7 billion*、そしてバックオフスムージングで推定された、7 の後に *billion* が来る確率  $P(\textit{billion}|\textit{7})$  を常用対数  $\log_{10}$  でとった値  $\log_{10} P(\textit{billion}|\textit{7})$  である。バックオフスムージングに関しては、第 3.6.2 節で述べる。

### 3.6.2 $N$ -gram モデル

言語モデルの代表的なモデルとして  $N$ -gram モデルがある．この  $N$ -gram モデルとは 1 次元の単語列  $w_1, w_2, w_3, \dots, w_n = w_1^n$  における  $i$  番目の単語  $w_i$  の生起確率  $P(w_i)$  は  $w_i$  の直前の  $(N - 1)$  単語に依存するモデルであり，以下の式で表現される．

$$P(w_1^n) = P(c_1) \times P(c_2|c_1) \times P(c_3|c_1c_2) \cdots \quad (3.23)$$

$$= \prod_{i=1}^n P(w_i|c_1c_2c_3 \cdots c_{i-1}) \quad (3.24)$$

$$\simeq \prod_{i=1}^n P(w_i|w_{i-N+1}, w_{i-N+2}, \dots, w_{i-1}) \quad (3.25)$$

“She is cute.” という英語文  $e$  に対する  $N = 2$  とした 2-gram モデルの例を以下に示す．

$$P(e = \textit{She is cute} .) \simeq P(\textit{She})P(\textit{is}|\textit{She})P(\textit{cute}|\textit{is})P(.|\textit{cute}) \quad (3.26)$$

3-gram モデルであれば  $P(\textit{cute}|\textit{She is})$  , 4-gram モデルであれば  $P(.|\textit{She is cute})$  となる．また， $N = 1$  のモデルを uni-gram ,  $N = 2$  のモデルを bi-gram ,  $N = 3$  のモデルを tri-gram と特有の呼びかたをする． $N = 4$  以上は 4-gram など数値を用いて呼ぶ．一般的に tri-gram の精度が高いとされている．(3.9) 式を最尤推定により求めると以下の式となる．

$$P(w_n|w_{n-N+1}^{n-1}) = \frac{C(w_{n-N+1}^n)}{C(w_{n-N+1}^{n-1})} \quad (3.27)$$

この (3.27) 式から信頼性の高い値を推定するためには単語列  $w_1^n$  が多く出現している必要がある．しかし実際には多くの単語列は出現数が 0 となることが多いため信頼できる値を推定できない場合が多い．したがって，スムージングを行う．スムージングとは生起確率が 0 となるのを防ぐために，大きい確率値を小さく，小さい確率値を大きくし，確率値の平滑化する手法であり，代表的な手法にバックオフスムージングがある．バックオフスムージングでは学習データに出現しない  $N$ -gram を  $(N - 1)$ -gram の値から推定を行う．tri-gram の場合には確率が以下の式で推定される．

$$P(w_n|w_{n-2}^{n-1}) = \begin{cases} \lambda(w_{n-2}^n) \frac{C(w_{n-2}^n)}{C(w_{n-2}^{n-1})} & \text{if } C(w_{n-2}^n) > 0 \\ (1 - \lambda_0(w_{n-2}^{n-1}))\alpha P(w_n|w_{n-1}) & \text{else if } C(w_{n-2}^{n-1}) > 0 \\ P(w_n|w_{n-1}) & \text{otherwise} \end{cases} \quad (3.28)$$

$\lambda$  はディスカウントと呼ばれる係数であり，出現しない  $N$ -gram に対して確率を付与するために， $(N - 1)$ -gram から確率を推定する役割を持つ． $\alpha$  は確率の和を 1 にするための正規化係数である．ディスカウント係数を求める方法は多数存在する．例として，“-ndiscount” や“-kndiscount” や“-ukndiscount” などがある．

$N$ -gram モデルは局所的な情報であり，少ない文法情報しか保持できない．したがって，文法構造が似ている言語間では，部分的な翻訳で十分な翻訳文が生成でき，翻訳精度が高い傾向がある．また，日本語と英語のような文法構造の異なる言語間では特異な文が生成される場合があり翻訳精度が低下する傾向がある．

## 3.7 デコーダ

デコーダは翻訳モデルと言語モデルの全ての組合せから確率が最大となる出力文を探索して翻訳を行う。入力文「彼は昼食を食べた」に対する探索例を図 3.3 に示す。

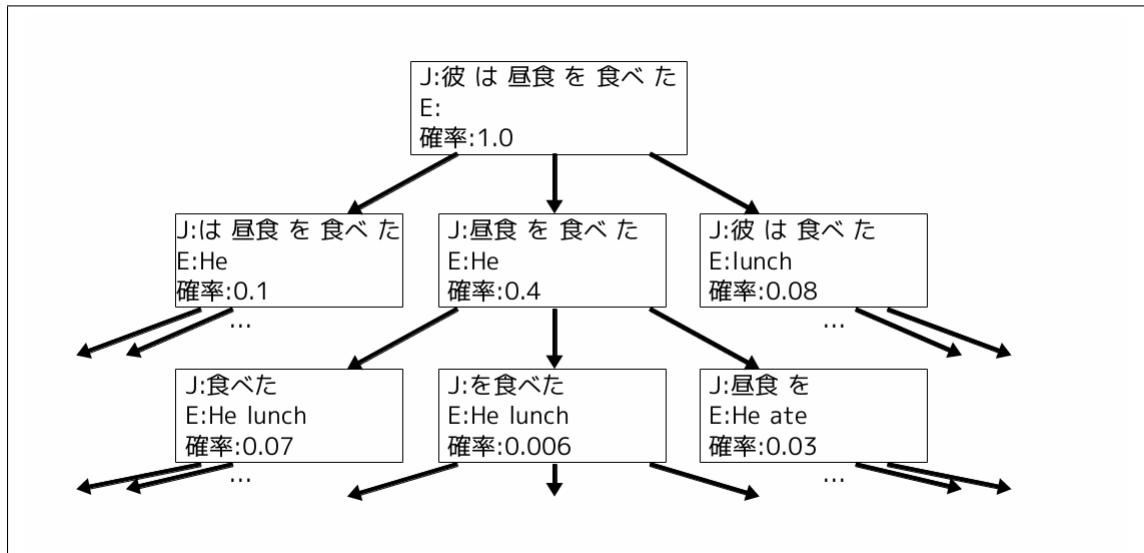


図 3.3 デコーダの探索例

デコーダは全く翻訳されていない仮説から探索を始める。そして入力文 J の翻訳されていないフレーズを選択肢、翻訳モデルのフレーズ対を用いて翻訳候補を生成する。このとき、翻訳候補の確率を計算し、その翻訳候補のスコアとする。この動作を繰り返し翻訳を行う。しかし入力文が長くなる場合、用いるフレーズ対の組み合わせは莫大となる。したがって、翻訳候補を全探索し、最適な翻訳文を決定することは困難である。

そこで、現在、計算量を削減する方法として、ビームサーチ法とマルチスタック法を組み合わせた手法が一般に用いられている。

### 3.7.1 ビームサーチ法

ビームサーチ法は探索にかかる計算量を削減するために用いられる。翻訳候補の探索機において、翻訳確率の低い翻訳候補を枝刈りし、探索範囲を限定する。枝刈りは“histogram pruning”と“threshold pruning”によって行う。“histogram pruning”は確率の高い翻訳候補のみを一定数残す枝刈り法である。“threshold pruning”は一定の確率以上の翻訳候補のみを残す枝刈り法である。この2つの枝刈り法を用いて、探索範囲を限

定する．しかし，図 3.3 を見てもわかるように，翻訳が進むほどに，翻訳候補の確率は低くなる．そのため，翻訳が進んだ翻訳候補と翻訳が進んでいない翻訳候補を比較したときに，翻訳が進んだ翻訳候補ほど枝刈りの対象となる可能性が高い．

### 3.7.2 マルチスタック法

ビームサーチ法の問題を解決するために，ビームサーチ法とマルチスタック法を組み合わせる．マルチスタック法は翻訳候補を翻訳した単語の数ごとに分ける．そして，分けた翻訳候補のなかで，ビームサーチ法を適用する．図 3.4 にマルチスタック・ビームサーチ法の適用例を示す．

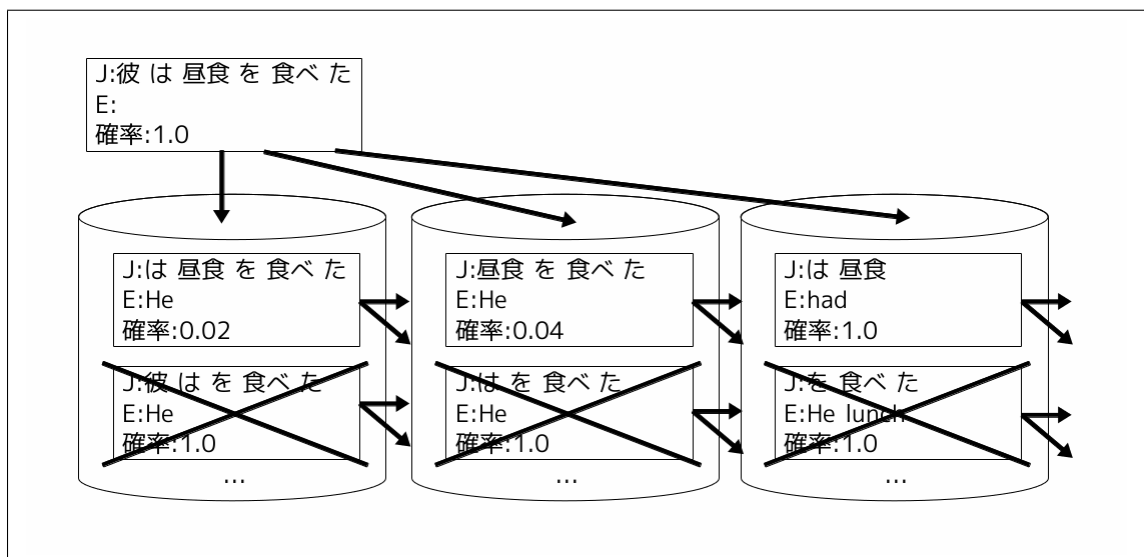


図 3.4 マルチスタック・ビームサーチ法の適用例

図 3.4 において，“histogram pruning” の残す翻訳候補を 1 とした場合，× のついた翻訳候補からは探索が行われない．

## 3.8 パラメータチューニング

デコーダには翻訳確率の重みなどのパラメータが複数存在する．このパラメータを最適化する手法として，一般に Minimum Error Rate Training(MERT)[9] が用いられる．MERT は目的の評価関数 (一般に BLEU) を最大にするための翻訳結果を選択するためにパラメータのチューニングを行う．このときディベロップメントデータと呼ばれる，試

し翻訳を行うデータを使用し、各文に対し上位 100 文程度の翻訳候補を出力する。その候補の中で重みを変えることでよりよい翻訳候補が上位にくるようにパラメータを調整する。

## 3.9 評価方法

機械翻訳システムの翻訳精度を自動的に評価する手法として、あらかじめ用意した正解文と、翻訳システムが出力した文とを比較する手法が一般的である。自動評価法には多くの方法がある。本研究では、BLEU[10] と NIST[11] と METEOR[12] を用いる。

### 3.9.1 BLEU

自動評価法のなかで最も代表的な手法が BLEU である。BLEU は以下の式を用いて評価値の計算を行う。

$$BLEU = BP_{BLEU} * exp\left(\frac{1}{N} \sum_{n=1}^N \log p_n\right) \quad (3.29)$$

値域は 0~1 であり、高いほど翻訳文が正解文に近く、翻訳精度が高いことを示す。(3.29) 式の  $N$  には、一般的に  $N = 4$  が用いられる。また、 $BP_{BLEU}$  は翻訳文が正解文と比較して短い場合に用いるペナルティ係数である。これは、翻訳文が正解文よりも短い場合に適合率が上がってしまうことを回避するために用いられる。 $BP_{BLEU}$  は翻訳文の単語数を  $c$ 、正解文の単語数を  $r$  とし、以下の式で計算される。

$$BP_{BLEU} = \begin{cases} 1 & (c > r) \\ e^{1-\frac{r}{c}} & (c \leq r) \end{cases} \quad (3.30)$$

翻訳文の文字数が正解文よりも長い場合には、 $BP_{BLEU} = 1$  であり、BLEU 値に対して影響を及ぼさない。長い場合のペナルティは後述する。翻訳文の文字数が正解文よりも短い場合には、その長さに応じて、 $BP_{BLEU}$  が小さくなり、結果として BLEU 値が小さくなる。尚、単語数  $c$  と  $r$  を算出するときの対象文が 1 文ではなく、コーパス全体であるので、各文の長さの自由度が残されている。また、(3.29) 式の指数関数の  $p_n$  は以下の式で計算される。

$$p_n = \frac{\sum_{\text{全翻訳文}} \sum_{\text{正解文}} \text{正解文と一致する } N\text{-gram 数}}{\sum_{\text{全翻訳文}} \sum_{\text{正解文}} \text{全 } N\text{-gram 数}} \quad (3.31)$$

$p_n$  は正解文と一致する  $N$ -gram 数を数えるときに正解文の要素を重複して数えることを回避するための処理を行う。例えば、翻訳文が “a a a a” であり、正解文が “This is a book” である場合にこの処理を行わなければ、翻訳文の 4 つの “a” と一致するものとし

て、正解文の“a”が重複して数えられ  $p_1 = \frac{4}{4}$  となる。この値に対して処理を行うことで、 $p_1 = \frac{1}{4}$  となる。また、正解文に対して、翻訳文の方が長い場合には、 $p_n$  の値は小さくなる。つまり、BLEU 値が小さくなるので、ペナルティが (3.29) 式に含まれていることがわかる。

### 3.9.2 NIST

NIST は BLEU と類似する自動評価法である。NIST は以下の式を用いて評価値の計算を行う。

$$NIST = BP_{NIST} * \sum_{n=1}^N \frac{\sum_{\text{全翻訳文}} \sum_{\text{正解文と一致する}} \text{Inf}o(w_1 \cdots w_n)}{\sum_{\text{全翻訳文}} \sum_{\text{正解文}} \text{全 } N\text{-gram 数}} \quad (3.32)$$

NIST の値域は 0 ~ 1 であり、高いほど翻訳精度が高い。BP<sub>NIST</sub> は翻訳文が正解文と比較して短い場合に用いるペナルティ係数である。翻訳文の単語数を  $c$ 、全ての正解文を対象とした正解文の平均単語数を  $r$  とし、以下の式で計算される。

$$BP_{NIST} = \exp\{\beta \log^2[\min(\frac{c}{r}, 1)]\} \quad (3.33)$$

(3.33) 式の  $\beta$  は  $\frac{c}{r} = \frac{2}{3}$  のとき、BP<sub>NIST</sub> の値が 0.5 となるように選択される。つまり、 $\beta$  の値は以下の計算で行われる。

$$0.5 = \exp\{\beta \log^2(\frac{2}{3})\} \quad (3.34)$$

$$\log 0.5 = \beta \log^2(\frac{2}{3}) \quad (3.35)$$

$$\beta = \frac{\log 0.5}{\log^2(\frac{2}{3})} \quad (3.36)$$

この選択によって、正解文と比較して非常に短い翻訳文に対して協力的なペナルティが与えられ、評価値がより小さくなる。反対に、正解文と長さが近い翻訳文に対するペナルティは軽減され、評価への影響が小さくなる。また、(3.32) 式の  $\text{Inf}o(w_1 \cdots w_n)$  は以下の式で定義される。

$$\text{Inf}o(w_1 \cdots w_n) = \log_2 \frac{\text{評価コーパス中の } w_1 \cdots w_{n-1} \text{ 数}}{\text{評価コーパス中の } w_1 \cdots w_n \text{ 数}} \quad (3.37)$$



より頻度の低い  $N$ -gram, つまり, より情報量の多い  $N$ -gram がより重み付けされる. 例えば, bi-gram の場合を考えると, 評価コーパスにおいて “a” が 5 回出現し, そのうち 4 回が “a pen” であり, 1 回が “a pencil” であったとする. この場合には, 1 回しか出現しない “a pencil” の情報量が多いと言える. 反対に, 4 回出現する “a pen” のそれぞれ 1 つずつの持つ情報量は少ない. したがって, この場合では, (3.37) 式によって, “a pencil” がより重み付けされる.

### 3.9.3 METEOR

METEOR は適合率  $P$  と再現率  $R$  から  $F$  値を求めた後に, 単語の非連続生に基づいて  $F$  値を修正して評価を行う自動評価法である.  $F$  値と修正に用いるペナルティ関数  $Pen$  は以下の式で計算される.

$$F = \frac{P \times R}{\alpha \times P + (1 - \alpha) \times R} \quad (3.38)$$

$$METEOR = F \times (1 - Pen) \quad (3.39)$$

$$Pen = \gamma \times \left(\frac{c}{m}\right)^\beta \quad (3.40)$$

METEOR の値域は 0~1 であり, 高いほど翻訳精度が高い. (3.40) 式において,  $m$  は翻訳文と正解文の一致率であり,  $c$  は単語を対象として, 翻訳文と一致するものを 1 つに統合したときの数である.  $\alpha, \beta, \gamma$  はパラメータである. 一致率  $c$  を計算するときの類義語などのマッチングには一般に WordNet を用いる. このマッチングでは, 意味や概念が近い類義語のマッチング, 単語の語形変化を同一のものとして処理することが可能であり, 人手評価に近い評価結果を得ることができる.

# 第4章 翻訳システムの自動作成

## 4.1 本研究の翻訳システムの概要

パターン翻訳では、文パターン辞書と単語辞書を用いて翻訳を行う。文パターンが有する大局的な文法情報を用いることで翻訳文全体の構造を保持した翻訳精度の高い翻訳文を生成できる利点がある。しかし、従来、文パターン辞書の作成は人手で行うため、開発にコストがかかるという欠点がある。

そこで本研究では、文パターン辞書に対訳文対から自動的に作成する手法を検討する。文パターン辞書の自動作成により、開発にかかるコスト削減が可能となる。また、パターン翻訳を統計翻訳の前処理に用いることで、文法構造が類似し翻訳精度が向上すると考えられる。図 4.1 に提案手法の枠組みを示す。

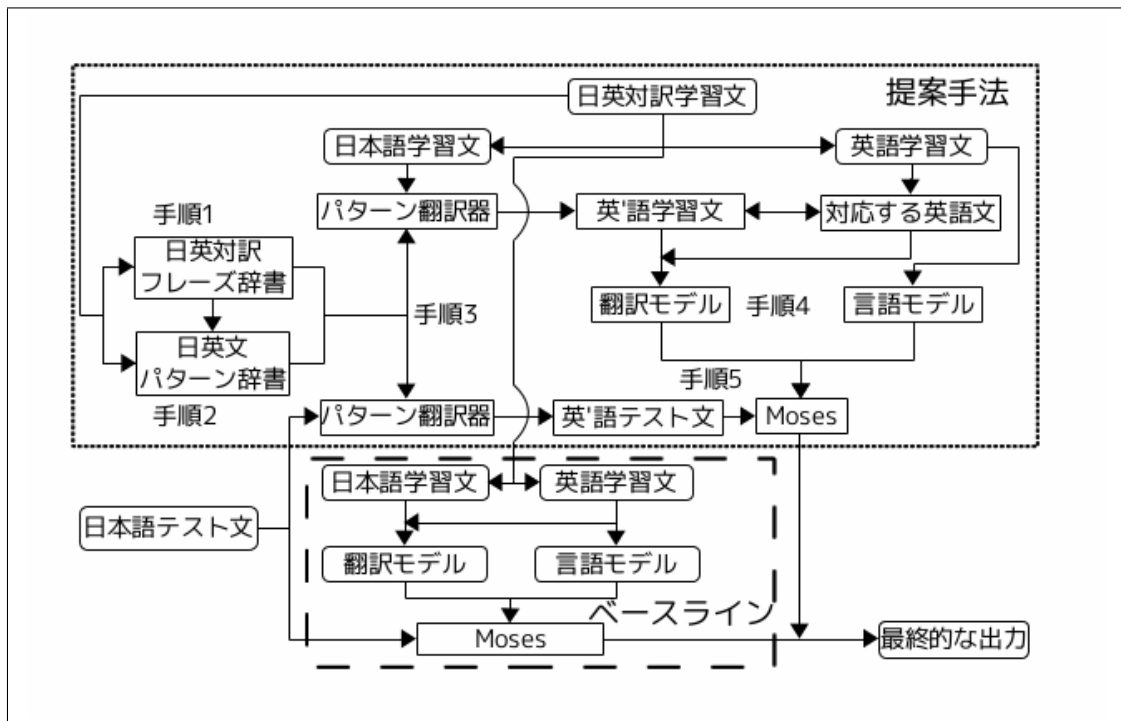


図 4.1 提案手法の枠組み

## 4.2 本研究の翻訳システムの手順

本研究の翻訳システムの手順を以下に示す。

### 手順 1 日英対訳フレーズ辞書の作成

まず日英対訳学習文から Moses[13] に付属している “train-model.perl” を用いてフレーズテーブルを生成する。次に，各フレーズ対の両方向へのフレーズ翻訳確率と単語翻訳確率について，それぞれ積を計算する。そして，閾値  $\alpha$  以上の確率を持つフレーズ対を抽出し，日英対訳フレーズ辞書を作成する。図 4.2 に日英対訳フレーズ辞書の作成手順を示す。

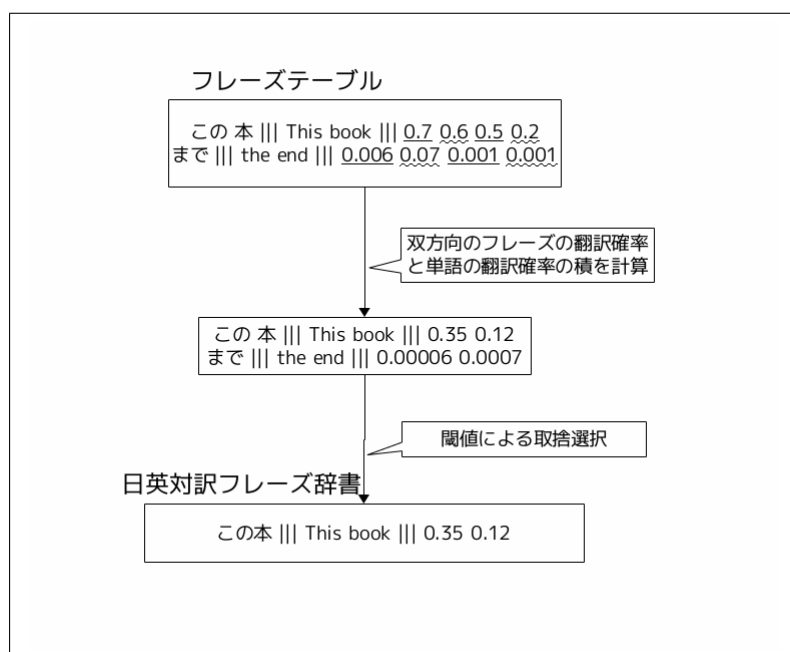


図 4.2 日英対訳フレーズ辞書の作成手順

最初に，各フレーズ対に対して，両方向へのフレーズの翻訳確率（下線部が直線の確率値）と，両方向への単語の翻訳確率（下線部が波線の確率値）の積を計算する。そして計算した確率値に対して，各確率に対して閾値  $\alpha$  以上の確率を持つフレーズ対のみを抽出し，日英対訳フレーズ辞書の作成を行う。図 4.2 で作成できた日英対訳フレーズ辞書において，左から，日本語フレーズ，英語フレーズ，双方向へのフレーズ翻訳確率の積，双方向への単語翻訳確率の積を表している。

## 手順 2 日英文パターン辞書の作成

手順 1 で作成した日英対訳フレーズ辞書を用いて日英対訳学習文から日英文パターン辞書を自動的に作成する。日英対訳フレーズ辞書を参照し、日英対訳学習文中で適合したフレーズ対を変数化して、日英文パターン辞書を作成する。尚、パターン翻訳を行うとき、変数部分に対して候補が莫大になることを防ぐために、変数部分を変換可能なフレーズに含まれる単語数について制限を設ける。図 4.3 に日英文パターン辞書の作成手順を示す。

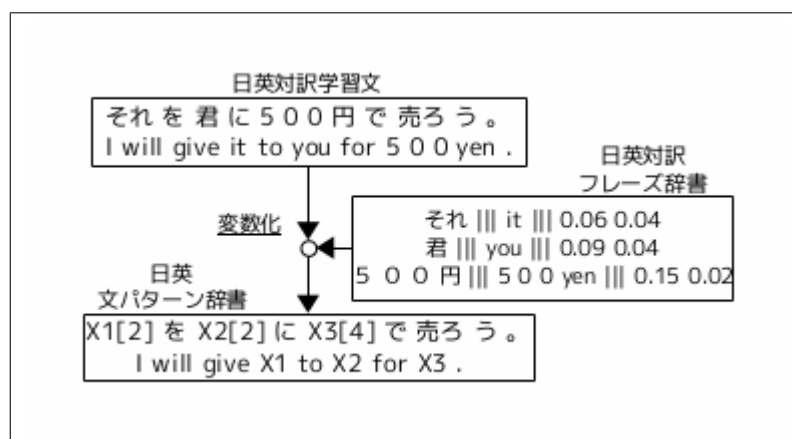


図 4.3 日英文パターン辞書の作成手順

図 4.3 の例では、日本語文“それを君に500円で売ろう。”と英語文“I will give it to you for 500 yen.”に対して、日英対訳フレーズ辞書を参照し、適合するフレーズに対して変数化を行う。3つのフレーズ“それ”、“君”、“500円”が変数化され、日文パターン“X1[2]をX2[2]にX3[4]で売ろう。”が生成される。そして同時に英文パターン“I will give X1 to X2 for X3.”も生成される。日文パターンの変数部分 X に付与されている大括弧内の数値は変数化するとき用いたフレーズが含む単語数であり、単語数が1である場合には“2”とする。“2”とする理由としては、入力文に対してカバー率を上げるためである。

### 手順3 パターン翻訳

手順1で作成した日英対訳フレーズ辞書と手順2で作成した日英文パターン辞書を用いて、日英パターン翻訳を行う。図4.4に日英パターン翻訳の流れを示す。

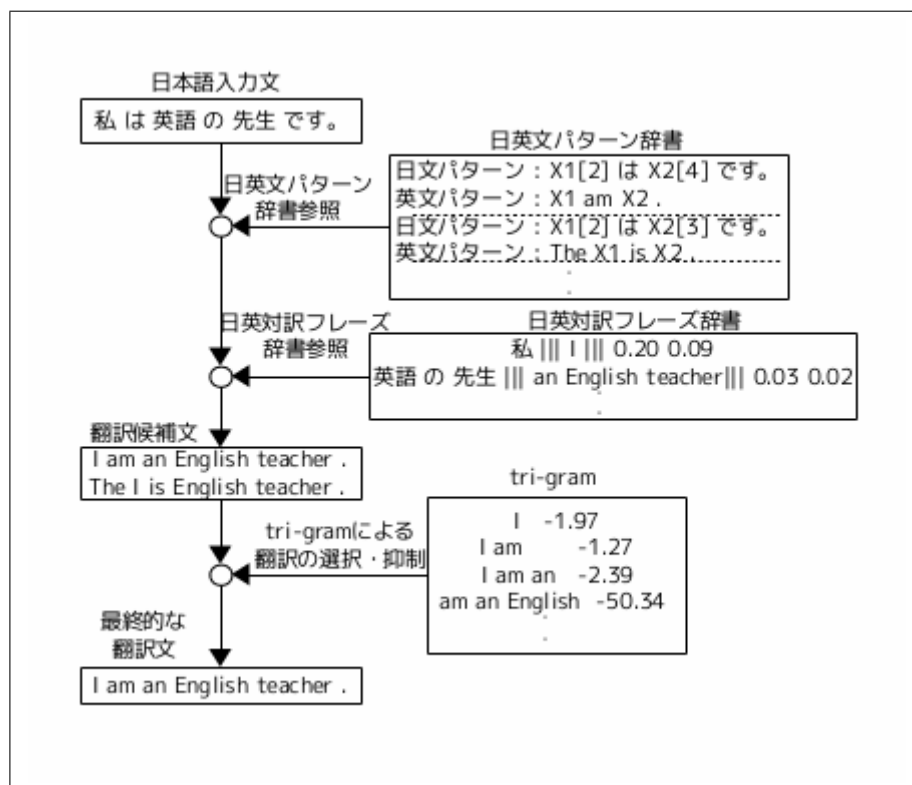


図 4.4 日英パターン翻訳の流れ

図4.4において、最初に日本語入力文“私は先生です。”に対して、日英文パターン辞書を参照し、適合するパターンを選択する。次に、日英対訳フレーズ辞書を参照し、適合した日文パターンの変数部分にあたる日本語フレーズ“私”、“英語の先生”のそれぞれについて、英語フレーズを“I”、“an English teacher”を得る。そして、英文パターンと変数部分の英語フレーズを用いて翻訳候補を生成する。複数の候補が出現する場合には、tri-gramを用いて候補を1文に絞り込む。図4.4では、2つの翻訳候補から最終的な翻訳文“I am an English teacher.”を得る。尚、本研究では、パターン翻訳を用いて翻訳した日本語テスト文と日本語学習文をそれぞれ、英語テスト文、英語学習文と呼ぶ。また、翻訳結果のtri-gramのスコアが低い場合には、適合すべきでないパターンにマッチしているなど、出力結果に誤りを含む可能性が高いと考えられる。そこで本研究では、翻訳結果のtri-gramでのスコア

に対して、閾値  $\beta$  を設け、 $\beta$  以下のスコアを持つ最終的な翻訳文は出力しないものとする。

#### 手順 4 統計翻訳の翻訳モデルと言語モデルの学習

英' 語学習文と対応する英語学習文を用いて翻訳モデルを学習し、英語学習文を用いて言語モデルを学習する。図 4.5 にモデル学習の例を示す。

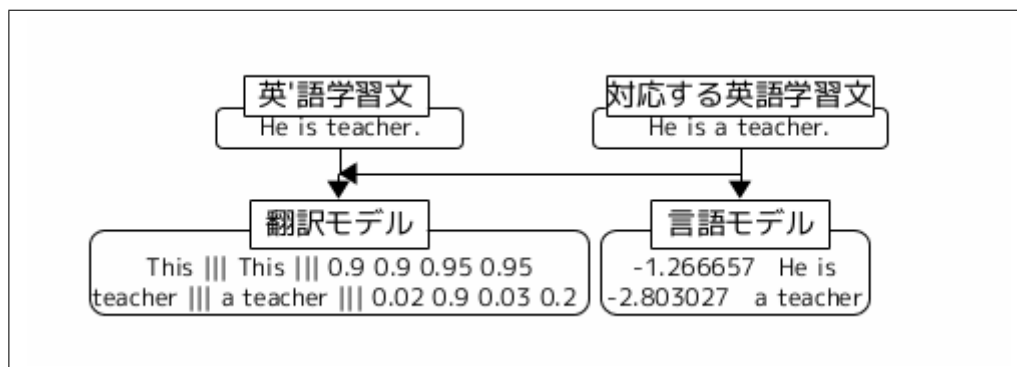


図 4.5 モデル学習の例

#### 手順 5 統計翻訳における英語文生成

英' 語テスト文に対して、手順 4 で学習した翻訳モデルと言語モデルを用いて英' 英統計翻訳を行う。図 4.6 に英' 英統計翻訳の例を示す。

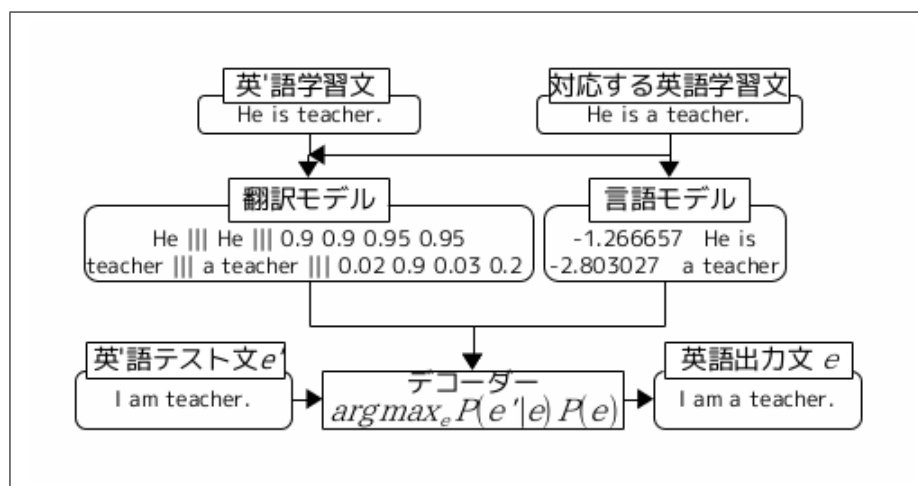


図 4.6 英' 英統計翻訳の例

パターン翻訳の出力に対して、さらに統計翻訳を行うことで、パターン翻訳におけ

る局所的な誤りを修正できると考えられる．本研究では，デコーダに Moses を用いる．

### 4.3 ベースラインシステム

Moses を用いた日本語テスト文から英語出力文への翻訳を行うシステムをベースラインシステムと呼ぶ．翻訳モデルの学習には日英対訳学習文を，言語モデルの学習には英語学習文を用いる．

## 第5章 実験環境

### 5.1 実験データ

実験データには、辞書の例文から抽出した日英対訳文の単文、重文複文を用いる [6]。単文の翻訳実験には学習文に単文 100,000 文、テスト文に単文 10,000 文を用いる。重文複文の実験には学習文に重文複文 100,000 文、テスト文に重文複文 10,000 文を用いる。また前処理として、日本語文に対しては MeCab[14] を用いて形態素解析を行い、形態素と句読点の間にスペースを入れる。英語文に対してはコンマ、ピリオド、数字の前後にスペースを入れる。表 5.1 に単文の例を、表 5.2 に重文複文の例を示す。

表 5.1 単文の例

日本語文	彼は科学上の偉大な発見をした。
英語文	He made a great scientific discovery .
日本語文	彼は父の財産を継いだ。
英語文	He succeeded to his father 's fortune .
日本語文	この中から 1 つ取りなさい。
英語文	Take one from among these .

表 5.2 重文複文の例

日本語文	彼は父の後を継いで医者になった。
英語文	He followed in his father 's footsteps and became a doctor .
日本語文	彼は勉強したから成功した。
英語文	He succeeded owing to his diligence .
日本語文	これは法律学のバイブルと言われる書物だ。
英語文	This is called the Bible of law .



## 5.2 翻訳モデルの学習

翻訳モデルの学習には，Moses に付属している “train-model.perl” を用いて行う．このプログラムは GIZA++ を利用し，IBM モデルの計算を行う．本実験では，フレーズテーブルを作成するヒューリスティックとして “grow-diag-final-and” を用い，max-phrase-length はデフォルト値の 20 とする．

## 5.3 言語モデルの学習

統計翻訳に用いる言語モデルには  $N$ -gram モデルを用いる．本研究では，SRILM[15] の “ngram-count” を用いて 5-gram の言語モデルを学習する．尚，スムージングに “-ndiscount” を用いる．

## 5.4 デコーダのパラメータ

本研究では，“distortion-limit” を “-1” とし，その他のパラメータについては Moses のデフォルトの値を用いる．尚，本研究ではパラメータチューニングは行わない．

## 第6章 翻訳実験

### 6.1 提案手法におけるパターン翻訳

日本語テスト文と日本語学習文を用いてパターン翻訳器の作成とパターン翻訳を行う。まず、日本語学習文を用いて、日英対訳フレーズ辞書を作成する。今回の実験では、日英対訳フレーズ辞書作成時に用いる閾値  $\alpha$  を“0.001”とする。次に作成した日英対訳フレーズ辞書と日本語学習文から日英文パターン辞書を作成する。そして日英対訳フレーズ辞書と日英文パターン辞書を用いて、日本語テスト文と日本語学習文に対して、パターン翻訳を行う。翻訳文の出力抑制に用いる閾値  $\beta$  は“-1000”とする。作成した日英対訳フレーズ辞書のフレーズ対数、日英文パターン辞書の文パターン数を表 6.1 に示す。

表 6.1 各辞書のエントリ数

	フレーズ数	パターン数
単文	40,787	95,853
重文複文	38,581	96,803

### 6.2 パターン翻訳部の実験結果

各実験条件における、日本語入力文でのパターン翻訳の出力文数を表 6.2 に示す。

表 6.2 パターン翻訳の出力文数

	テスト文	学習文
単文	1,143	64,584
重文複文	349	50,402

尚、閾値によって出力を抑制された文は、単文のテスト文で 87 文、重文複文のテスト文で 6 文であった。

### 6.3 提案手法における統計翻訳

翻訳モデルの学習には表 6.2 の学習文の出力結果 (英' 語学習文) と, 英' 語学習文に対応する英語学習文を用いる. 言語モデルの学習には英語学習文を用いる. 学習した翻訳モデルと言語モデルを用いて英' 語テスト文に対して英' 英統計翻訳を行う.

### 6.4 ベースラインシステムにおける実験

日本語テスト文 10,000 文に対して, ベースラインの翻訳モデルの学習には日英対訳学習文 100,000 文を用いる. 言語モデルの学習には英語学習文 100,000 文を用いる. 学習した翻訳モデルと言語モデルを用いて日英統計翻訳を行う.

## 第7章 実験結果

日本語テスト文 10,000 文での結果と、表 6.2 の日本語テスト文のパターン翻訳で得られた出力文のみでの実験結果について示す。

### 7.1 自動評価結果

自動評価には BLEU, NIST, METEOR を用いる。表 7.1 に日本語テスト文 10,000 文での結果、表 7.2 にパターンに適合した日本語テスト文での結果を示す。尚、表 7.2 中の“パターン”はパターン翻訳のみでの翻訳結果を示している。

表 7.1 10,000 文での自動評価結果

単文			
	BLEU	NIST	METEOR
ベースライン	0.1130	4.5211	0.3160
提案手法	0.1101	4.5131	0.3175
重文複文			
	BLEU	NIST	METEOR
ベースライン	0.0947	4.0980	0.3021
提案手法	0.0977	4.1406	0.3049

単文の場合には提案手法の評価値がベースラインよりも低くなっている。一方、重文複文では提案手法の評価値がベースラインよりも高く、提案手法の有効性が見られる。

表 7.2 パターン翻訳を通じた出力文での自動評価結果

単文 (1,143 文)			
	BLEU	NIST	METEOR
ベースライン	0.2218	5.2390	0.4363
提案手法	0.1821	4.8417	0.4426
パターン	0.1630	4.5314	0.4230
重文複文 (349 文)			
	BLEU	NIST	METEOR
ベースライン	0.2814	5.1235	0.4562
提案手法	0.3618	5.8849	0.5438
パターン	0.3384	5.7004	0.5326

## 7.2 人手による評価

表 7.2 のパターンに適合した文からランダムに抽出した 100 文に対して対比較評価を行う。

### 7.2.1 判断基準

人手による 4 つの判断基準に基づいて評価を行う。評価基準と評価例を以下に示す。

評価 1(>)	提案手法の結果がベースラインの結果よりも優れている
入力文	彼女は 5 人の子供を育てた。
正解文	She has brought up five children .
ベースライン	She is five children .
提案手法	She brought up five children .
評価 2(<)	提案手法の結果がベースラインの結果よりも劣っている
入力文	農園は道路に接している。
正解文	The farm abuts on the road .
ベースライン	farm adjoins the road .
提案手法	The farm is roads are .
評価 3(≈)	どちらも同程度に意味を理解できる、または、どちらも同程度に意味を理解できない
入力文	彼によろしくお伝えください。
正解文	Give him my good wishes .
ベースライン	Please give my best regards to him .
提案手法	Please send him my best wishes .

評価 4(=) 提案手法とベースラインの翻訳結果が同一である  
 入力文 このビールは気が抜けている。  
 正解文 This beer is flat .  
 ベースライン This beer tastes flat .  
 提案手法 This beer tastes flat .

## 7.2.2 評価結果

表 7.3 に人手による対比較評価の結果を示す。

表 7.3 対比較評価の結果

	単文	重文複文
評価 1(>)	30 / 100	33 / 100
評価 2(<)	9 / 100	4 / 100
評価 3( $\approx$ )	50 / 100	30 / 100
評価 4(=)	11 / 100	33 / 100

単文での実験において、自動評価結果とは異なり、対比較評価では提案手法がベースラインよりも優れている翻訳文が多くなっている。また重文複文の実験では、対比較評価でも提案手法がベースラインよりも性能が高い。

## 第8章 考察

### 8.1 翻訳精度が高い出力の解析

重文複文の自動評価結果において、提案手法がベースラインよりも評価値が高くなっており、パターン翻訳のみでもベースラインよりも高くなっている。提案手法がベースラインよりも優れている翻訳文について、表 8.1 に単文での翻訳例を、表 8.2 に重文複文での翻訳例を示す。

表 8.1 短文での提案手法が優れている翻訳例

入力文	お前に限る。
ベースライン	I accepted .
提案手法	There is nothing like you .
パターン	There is nothing like You .
日文パターン	X1[2]に限る。
英文パターン	There is nothing like X1
参照文	No one but you can do it .

表 8.2 重文複文での提案手法が優れている翻訳例

入力文	子供には好きなようにさせている。
ベースライン	A child as you please .
提案手法	I allow the children do as they like .
パターン	I allow the children do as they like .
日文パターン	X1[2]には X2[2]なようにさせている。
英文パターン	I allow the X1 do as they X2 .
参照文	I let the children do as they like .

単文、重文複文のどちらの例においても、入力文に対して適切なパターンが適合し、翻訳精度の高い文が得られている。単文の実験において、提案手法はベースラインよりも

自動評価による評価値が低い。しかし，人手による対比較評価では，提案手法がベースラインよりも優れていた文が多い。この要因として，提案手法がベースラインよりも優れていた 30 文のうち，ベースラインの出力に動詞が存在しない文が 16 文あり，人手による対比較評価において提案手法がベースラインを上回ったと考えられる。



## 8.2 誤り解析

単文の実験において，自動評価結果ではベースラインが提案手法よりも優れている結果となったが，反対に，人手による対比較評価では提案手法がベースラインよりも優れている結果となった．そこで，対比較評価の結果に対して，提案手法がベースラインよりも劣っていた9文に対して誤り解析を行った．解析の結果，適合した文パターンが不適切であった場合，日英対訳フレーズ辞書に登録されているフレーズ対に問題がある場合，提案手法の統計翻訳が悪影響を及ぼしている場合，日本語単語の意味の違いによって誤った翻訳をしている場合の4種類が確認できた．表8.3に誤り解析の結果を示す．

表 8.3 対比較評価の結果

場合	文数
適合した文パターンが不適切	4
フレーズ対に問題あり	2
統計翻訳が悪影響	2
単語の意味の違いによる誤翻訳	1

8.2.1 節に適合した文パターンが不適切であった場合，8.2.2 節に日英対訳フレーズ辞書に登録されているフレーズ対に問題がある場合，8.2.3 節に提案手法の統計翻訳が悪影響を及ぼしている場合，8.2.4 節に日本語単語の意味の違いによって誤った翻訳をしている場合の，各場合における例と考察をそれぞれ示す．

## 8.2.1 適合したパターンが不適切であった場合

表 8.4 に適合した文パターンが不適切であった場合の翻訳例を示す。

表 8.4 適合した文パターンが不適切であった場合の翻訳例

入力文	土壌が肥えている。
ベースライン	The soil is fertile .
提案手法	An early summer rain is fertile soil .
パターン	An early summer rain is fertile soil .
日文パターン	X1[2] が X2[2] て X3[2] 。
英文パターン	An early summer rain X3 X2 X1 .
日本語原文	五月雨がしとしと降っている。
英語原文	An early summer rain is falling gently .
参照文	The soil is rich .

表 8.4 から、適合した文パターンに問題があることがわかる。そこで、適合した文パターンについて解析を行った結果、パターン作成時に用いる日英対訳フレーズ辞書のフレーズ対に問題があった。日本語原文と英語原文を見ると、日本語フレーズ“五月雨”は英語フレーズ“An early summer rain”に対応すると考えられる。しかし、日文パターンでは“五月雨”が変数化されているが、英文パターンには“An early summer rain”は変数化されていない。ここで、日英対訳フレーズ辞書のフレーズ対を確認すると、“五月雨”が英語フレーズ“gently”に対応していた。また日本語フレーズ“しとしと降っ”は英語フレーズ“falling”に対応していた。

このことから、日英対訳フレーズ辞書が不適切なフレーズ対を含むことで、作成される日英文パターン辞書の品質を低下させ、翻訳精度の低下を引き起こすと考えられる。したがって、日英対訳フレーズ辞書を作成するときに用いる閾値やの調整や、品詞タグの利用により、この問題に対処できると考えられる。

## 8.2.2 フレーズ対に問題があった場合

表 8.5 に日英対訳フレーズ辞書に登録されているフレーズ対に問題があった場合の翻訳例を示す。

表 8.5 フレーズ対に問題があった場合の翻訳例

入力文	彼の家はどこかこのあたりにある。
ベースライン	His house is somewhere in this area .
提案手法	His house is somewhere vicinity here .
パターン	His house is somewhere vicinity here .
日文パターン	X1[3] はどこかこの X2[2] に X3[2] 。
英文パターン	X1 X3 somewhere X2 here .
日本語原文	彼の家はどこかこの辺りにある。
英語原文	His house is somewhere around here .
参照文	His house is somewhere about here .

表 8.5 から，入力文に対して適合した文パターンは適切である。しかし，日本語フレーズ“あたり”の翻訳について，英語フレーズ“vicinity”と翻訳されている。ここで，日英対訳フレーズ辞書のフレーズ対を確認すると，“あたり”に対して，英語フレーズ“vicinity”と“FSX”と“hovers”の3種類のフレーズ対が存在していた。しかし，どの英語フレーズについても，“あたり”に対して適切なフレーズではないと言える。

このことから日英対訳フレーズ辞書に登録されているフレーズ対が不適切である場合に誤翻訳される可能性が高くなると考えられる。したがって，日英対訳フレーズ辞書に不適切なフレーズ対を登録しないために，品詞タグの利用などによって，この問題に対処できると考えられる。

### 8.2.3 統計翻訳が悪影響を及ぼしている場合

表 8.6 に提案手法の統計翻訳が悪影響を及ぼしている場合の翻訳例を示す。

表 8.6 統計翻訳が悪影響を及ぼした場合の翻訳例

入力文	この仕事はやりがいがある。
ベースライン	This job is very challenging .
提案手法	This job worthwhile .
パターン	This work is worthwhile .
日文パターン	X1[2] は X2[2] が X3[2] 。
英文パターン	X1 X3 X2 .
日本語原文	彼女はいたずらが過ぎる。
英語原文	She is too mischievous .
参照文	This work is worth doing .

表 8.6 から、パターン翻訳では英語の動詞“is”が存在しているが、提案手法では“is”が無くなっている。この原因として、提案手法におけるパターン翻訳後の統計翻訳が悪影響を及ぼしている。この問題については、パターン翻訳での出力結果が統計翻訳の出力よりも優れた翻訳文が出力された場合に、どちらの翻訳文が優れているかの取捨選択を行うことで対処できると考えられる。

## 8.2.4 単語の意味の違いによって誤った翻訳をしている場合

表 8.7 に日本語単語の意味の違いによって誤った翻訳をしている場合の翻訳例を示す。

表 8.7 単語の意味の違いによって誤った翻訳をしていた場合の翻訳例

入力文	刀の柄に手をかけた。
ベースライン	He put his hand on the sword .
提案手法	She put her hand on the sword pattern .
パターン	She put her hand on the sword pattern .
日文パターン	X1[2] の X2[2] に X3[2] を X4[2] た。
英文パターン	She X4 her X3 on the X1 X2 .
日本語原文	ドアのノブに手をかけた。
英語原文	She put her hand on the door knob .
参照文	He grasped the hilt of his sword .

表 8.7 から，パターン翻訳時に日本語フレーズ“柄”が英語フレーズ“pattern”に翻訳されている．この原因として，“柄”の読み方が“がら”や“つか”など異なる場合に，入力文の持つフレーズの意味とは異なる翻訳が行われると考えられる．この問題については，適合した文パターンの変数部分に対して意味的制約を用いることで，異なる解釈を制限し，正しく翻訳できると考えられる．

## 第9章 おわりに

本研究では，日英対訳文対から日英文パターン翻訳器を自動的に作成する手法を提案した．実験の結果，単文の実験では，自動評価において提案手法はベースラインよりも評価値は低いが，人手による対比較評価では，提案手法はベースラインよりも良い結果となった．また重文複文の実験では，自動評価と対比較評価のどちらにおいても提案手法はベースラインよりも良い結果となった．しかし，日英対訳フレーズ辞書の不適切なフレーズ対が日英文パターン辞書の品質の低下，パターン翻訳の翻訳精度の低下を引き起こしている．この問題に対して，今後は，日英対訳フレーズ辞書の作成時に用いる閾値の調整，品詞タグの利用などにより，さらに翻訳精度が向上すると考えられる．

## 謝辞

最後に、本研究を進めるにあたり、様々のご助言を頂きました村田真樹教授にお礼を申し上げます。本論文をまとめるにあたり、様々のご助言を頂きました松村幸輝教授にお礼を申し上げます。3年間に渡り、種々のご助言を頂きました徳久雅人講師にお礼を申し上げます。本研究を進める上で、終始に渡り多大なご協力、ご助言を頂きました村上仁一准教授にお礼を申し上げます。

また、参考にさせて頂いた論文の著者の方々、本研究にご協力頂いた計算機工学講座Cの皆様にも深く感謝いたします。

## 参考文献

- [1] 池原 悟, 佐良木 昌, 宮崎 正弘, 池田 尚志, 新田 義彦, 白井 諭, 柴田 勝征, “等価的類推思考の原理による機械翻訳方式”, 電子情報通信学会技術研究報告, pp7-12, 2002.
- [2] 池原 悟, 宮崎 正弘, 白井 諭, 横尾 昭男, 中岩 浩巳, 小倉 健太郎, 大山 芳史, 林 良彦, “日本語語彙大系”, 岩波書店, 1997.
- [3] Richard Zens, Franz Josef Och, Hermann Ney, “Phrase-based Statistical Machine Translation”, KI 2002, pp35-56, 2002.
- [4] Holger Schwenk, Marta R.Costa-jussà, and José A.R.Fonollosa, “Continuous space language models for the IWSLT 2006 Task ”, International Workshop on Spoken Language Translation 2006, pp166-173, 2006.
- [5] Hideki Isozaki, Katsuhito Sudoh, Hajime Tsukada, Kevin Duh, “Head Finalization : A simple Reordering Rule for SOV Languages”, Association for Computational Linguistics 2010, pp.244-251, 2010.
- [6] 西山 七絵, 村上 仁一, 徳久 雅人, 池原 悟, “単文文型パターン辞書の構築”, 言語処理学会第 11 回年次大会, pp.372-375, 2005.
- [7] Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, Robert L. Mercer, “The Mathematics of Statistical Machine Translation: Parameter Estimation”, Association for Computational Linguistics 1993, pp263-311, 1993.
- [8] GIZA++, Training of statistical translation models  
<http://www.fjoch.com/GIZA++.html>
- [9] Franz Josef och, “Minimum Error Rate Training in Statistical Machine Translation”, Association for Computational Linguistics 2003, pp160-167, 2003.



- [10] Kishore Papineni, Salim Roukos, Todd Ward, Wei- Jing Zhu, “BLEU : a Method for Automatic Evaluation of Machine Translation”, Association for Computational Linguistics 2002, pp311-318, 2002.
- [11] George Doddington, “Automatic evaluation of machine translation quality using n-gram co-occurrence statistics”, Human Language Technology '02, pp128-132, 2002.
- [12] Satanjeev Banerjee, Alon Lavie, “METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments”, Association for Computational Linguistics 2005, pp65-72, 2005.
- [13] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, Evan Herbst, “Moses: Open Source Toolkit for Statistical Machine Translation”, Association for Computational Linguistics 2007, pp177-180, 2007.
- [14] MeCab : Yet Another Part-of-Speech and Morphological Analyzer  
<http://mecab.sourceforge.net/>
- [15] The SRI Language Modeling Toolkit  
<http://www.speech.sri.com/projects/srilm/>