

概要

現在機械翻訳の多くの言語モデルには N -gram モデルが使用されている．しかし， N -gram は分野依存性があり，別分野の言語データを追加しても翻訳精度があまり変わらないことが知られている [18] ．

そこで本研究では， N -gram モデルの代わりに交差エントロピーの拡張である High order Joint Probability を使用する手法を提案した．翻訳モデルには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) を用いた．言語モデルには Tri-gram と High order Joint Probability を用いた．対訳学習文とテスト文には，電子辞書から抜き出した文 (辞書文) で単文を用いた．追加データには wikipedia から抜き出した文 (Wikipedia 文) を用いた．

その結果，自動評価と人手評価の結果より，対訳学習文とは別分野のデータを追加データとして用いた場合，High order Joint Probability の方が Tri-gram よりも翻訳精度が良いことが確認できた．また，追加実験として追加データを利用しない実験と，追加データに対訳学習文と同一分野のデータを追加する実験を行った．追加データには辞書文から抽出した複文を用いた．追加データを利用しない場合と追加データに対訳学習文と同一分野のデータを追加する場合のどちらについても，High order Joint Probability の方が Tri-gram よりも翻訳精度が良いことが確認できた．

目次

第1章	はじめに	1
第2章	統計翻訳システム	2
2.1	概要	2
2.2	翻訳モデル	2
2.2.1	IBM 翻訳モデル	4
2.2.1.1	Model1	4
2.2.1.2	Model2	6
2.2.1.3	Model3	7
2.2.1.4	Model4	8
2.2.1.5	Model5	8
2.2.2	GIZA++	9
2.2.3	フレーズテーブルの作成方法	10
2.3	言語モデル	13
2.4	デコーダー	13
第3章	N -gram(従来手法)	14
3.1	従来手法の概要	14
3.2	従来手法の問題点	14
第4章	High order Joint Probability(提案手法)	15
4.1	提案手法の概要	15
4.2	提案手法の利点	16
第5章	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)	17
5.1	TDPBSMT の概要	17

5.2	TDPBSMT の手順	17
5.2.1	学習の手順	17
5.2.2	翻訳の手順	20
第 6 章	評価方法	22
6.1	自動評価	22
6.1.1	BLEU	22
6.1.2	NIST	23
6.1.3	METEOR	23
6.1.4	RIBES	24
6.1.5	TER	24
6.2	人手評価	25
第 7 章	実験方法	26
7.1	実験の概要	26
7.2	実験データ	26
第 8 章	実験結果	28
8.1	自動評価 (wikipedia 文追加) Joint Probability vs Tri-gram	28
8.2	人手評価 (wikipedia 文追加) Joint Probability vs Tri-gram	28
8.3	誤出力の解析 (Wikipedia 文)	31
8.4	評価結果のまとめ	31
第 9 章	追加実験	32
9.1	追加前 (Joint Probability vs Tri-gram)	32
9.2	誤出力の解析 (追加前)	35
9.3	同分野 (複文) Joint Probability vs Tri-gram	36
9.4	誤出力の解析 (複文)	39
9.5	評価結果のまとめ	39
9.6	Joint Probability:追加前 vs 複文	40
9.7	Tri-gram:追加前 vs 複文	40
9.8	Joint Probability:追加前 vs Wikipedia 文	41
9.9	Tri-gram:追加前 vs Wikipedia 文	41

第 10 章 考察	42
10.1 Joint Probability の解析	42
10.2 Tri-gram の解析	43
10.3 解析の評価	44
第 11 章 おわりに	45

目 次

2.1	統計翻訳の流れ	3
5.1	TDPBSMT の流れ図	21

表 目 次

2.1	日英翻訳におけるフレーズテーブルの例	4
2.2	日英方向の単語対応	10
2.3	英日方向の単語対応	10
2.4	intersection の例	11
2.5	union の例	11
2.6	grow の例	11
2.7	grow-diag の例	11
2.8	grow-diag-final の例	12
2.9	grow-diag-final-and の例	12
4.1	入力文	16
4.2	出力候補文	16
5.1	対訳単語作成に用いる学習文対	18
5.2	作成される対訳単語	18
5.3	単語レベル文パターンの作成例	18
5.4	変換テーブルの作成例	19
5.5	変換テーブルの分割	19
5.6	文パターンの作成例	19
5.7	入力文への変換テーブル _{CD} 適用例	20
5.8	文パターンへの変換テーブル _{CD} 適用例	20
5.9	TDPBSMT の出力例	21
6.1	対比較の出力例	25
6.2	対比較の評価基準	25
7.1	実験データ	26
7.2	平均単語長	26

7.3	データ例	27
8.1	自動評価結果 (wikipedia 文追加) Joint Probability vs Tri-gram	28
8.2	人手評価結果 (wikipedia 文追加) Joint Probability vs Tri-gram	29
8.3	Joint Probability の出力例 1	29
8.4	Joint Probability の出力例 2	29
8.5	Joint Probability の出力例 3	29
8.6	Tri-gram の出力例 1	30
8.7	Tri-gram の出力例 2	30
8.8	解析結果	31
9.1	実験データ	32
9.2	自動評価結果 : 追加前 (Joint Probability vs Tri-gram)	32
9.3	人手評価結果 : 追加前 (Joint Probability vs Tri-gram)	32
9.4	Joint Probability の出力例 1	33
9.5	Joint Probability の出力例 2	33
9.6	Joint Probability の出力例 3	33
9.7	Trigram の出力例 1	34
9.8	Trigram の出力例 2	34
9.9	Trigram の出力例 3	34
9.10	解析結果	35
9.11	実験データ	36
9.12	平均単語長	36
9.13	データ例	36
9.14	自動評価結果 (複文)	37
9.15	人手評価結果 (複文)	37
9.16	Joint Probability の出力例 1	37
9.17	Joint Probability の出力例 2	37
9.18	Joint Probability の出力例 3	37
9.19	Tri-gram の出力例 1	38
9.20	Tri-gram の出力例 2	38
9.21	Tri-gram の出力例 3	38
9.22	解析結果	39

9.23	人手評価結果 (Joint Probability) 追加前 vs 複文	40
9.24	人手評価結果 (Tri-gram) 追加前 vs 複文	40
9.25	人手評価結果 (Joint Probability) 追加前 vs Wikipedia 文	41
9.26	人手評価結果 (Tri-gram) 追加前 vs Wikipedia 文	41
10.1	出力候補文の翻訳確率 (Tri-gram)	42
10.2	出力候補文の翻訳確率 (Joint Probability)	43
10.3	出力候補文の翻訳確率 (Tri-gram)	43
10.4	出力候補文の翻訳確率 (Joint Probability)	44

第1章 はじめに

現在機械翻訳の多くの言語モデルには N -gram モデルが使用されている。しかし、 N -gram は分野依存性があり、別分野の言語データを追加しても翻訳精度があまり変わらないことが知られている [18]。

そこで本研究では、 N -gram モデルの代わりに交差エントロピーの拡張である High order Joint Probability を使用する手法を提案する。 N -gram が近隣の情報を利用しているのに対し、High order Joint Probability は遠隔の情報を利用できる。したがって、文法的な情報を利用できる。そのため分野依存性が少なく、別分野の言語データを追加しても翻訳精度が向上すると考える。機械翻訳機には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) を用いる。言語モデルには Tri-gram と High order Joint Probability を用いる。対訳学習文とテスト文には、電子辞書から抜き出した文 (辞書文) で単文を用いる。追加データには wikipedia から抜き出した文 (Wikipedia 文) を用いる。

その結果、自動評価と人手評価の結果より、対訳学習文とは別分野のデータを追加データとして用いた場合、High order Joint Probability の方が Tri-gram よりも翻訳精度が良いことが確認できた。また、追加実験として追加データを利用しない実験と、追加データに対訳学習文と同一分野のデータを追加する実験を行った。追加データには辞書文から抽出した複文を用いた。追加データを利用しない場合と追加データに対訳学習文と同一分野のデータを追加する場合のどちらについても、High order Joint Probability の方が Tri-gram よりも翻訳精度が良いことが確認できた。

本論文の構成は以下の通りである。まず、2章で統計翻訳システムの概要について述べる。3章では、従来手法について述べる。4章では、提案手法について述べる。6章では、評価方法について述べる。7章では、実験に用いるデータやツールといった実験環境について述べる。8章では、実験の結果を示す。9章では、追加実験について示す。そして、10章で考察を行い、最後に11章で結論を述べ、まとめる。

第2章 統計翻訳システム

本章は，善行ら [18] の論文を参照して記述した．

2.1 概要

統計翻訳とは，翻訳モデルより導き出された言語間の句の出現確率と，言語モデルにより導き出された単語の遷移確率の統計より，最適の翻訳結果を自動的に出力する翻訳方法である．現在の統計翻訳では“句に基づく統計翻訳”が主流である．句に基づく統計翻訳は，文脈における訳語の選択や語順の並び替えの精度において，“単語に基づく統計翻訳”より優れているためである．よって本研究でも，句に基づく統計翻訳を用いる．

図 2.1 に日英統計翻訳の例を示す．

日英統計翻訳では，日本語入力文を J とすると，翻訳モデル $P(J|E)$ と言語モデル $P(E)$ の全ての組み合わせから，確率が最大になる英語文 \hat{E} を出力文とする．また， \hat{E} を探索する翻訳システムをデコーダと呼ぶ．

以下にその基本モデルを示す．

$$\hat{E} = \arg \max_E P(E|J) \quad (2.1)$$

$$\simeq \arg \max_E P(J|E)P(E) \quad (2.2)$$

2.2 翻訳モデル

翻訳モデルは，原言語から目的言語の単語列へ，確率的に翻訳を行うためのモデルである．翻訳モデルは一般的に、フレーズテーブルと呼ばれる表で管理されている．以下の表 2.1 にフレーズテーブルの例を示す．

左から順に，日本語フレーズ，英語フレーズ，日英方向の翻訳確率 $P(J|E)$ ，日英方向の単語の翻訳確率の積，英日方向の翻訳 $P(E|J)$ ，英日方向の単語の翻訳確率の積である．

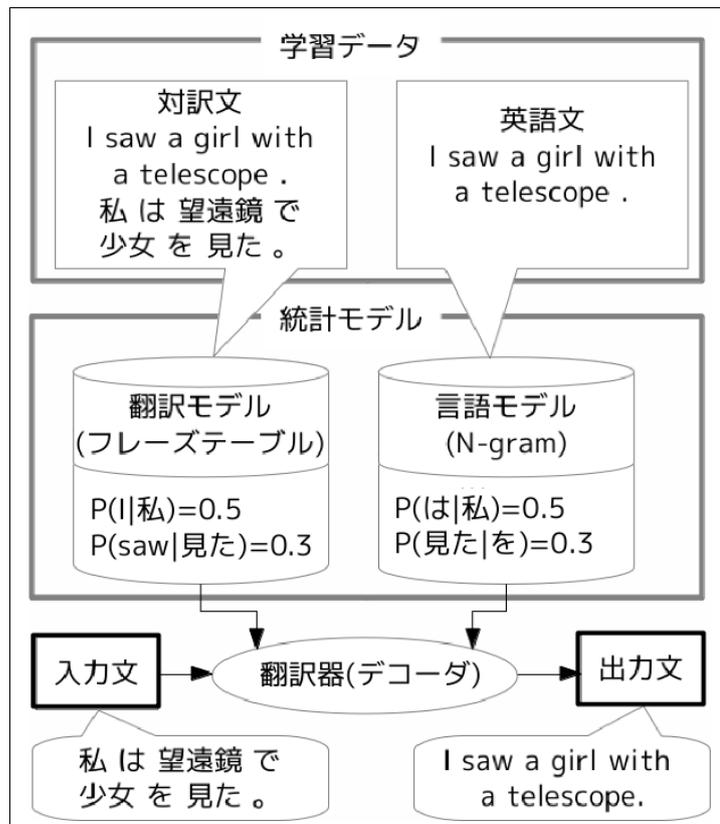


図 2.1: 統計翻訳の流れ

表 2.1: 日英翻訳におけるフレーズテーブルの例

日フレーズ	英フレーズ	$P(J E)$	$\prod P(J E)$	$P(E J)$	$\prod P(E J)$
あの歌	that song	1.2	0.2	1.1	0.4
おもちゃ	toy	0.5	0.6	0.3	0.3
雨の後	after the rain	0.3	1.2	0.1	1.8

2.2.1 IBM 翻訳モデル

翻訳モデルの代表例として，Brown らが提案した IBM 翻訳モデル [4] がある．IBM 翻訳モデルは，Model1 から Model5 まで 5 つのモデルからなり，順に複雑な計算となる．IBM 翻訳モデルは本来，仏英翻訳を想定としているが，本研究では日英翻訳と英日翻訳を行うため，日英翻訳を想定して説明をする．

英語文 E ，日本語文 J の翻訳モデル $P(J|E)$ を計算するためにアラインメント a を用いる．アラインメントとは，日本語単語 J と英単語 E の対応を意味する．以下に IBM モデルの式を示す．

$$P(J|E) = \sum_a P(J, a|E) \quad (2.3)$$

IBM モデルのアラインメントでは，日英統計翻訳の場合，日本語単語 j に対する英単語 e は 1 つあり，各英単語 e に対する日本語単語 j は 0 から n 個あると仮定する．また，日本語単語の対応関係として適切な英単語がなかった場合，英語文の文頭に空単語 e_0 があると仮定し，その日本語単語 j と空単語 e_0 を対応させる．

2.2.1.1 Model1

(2.3) 式は次のように置き換えられる．

$$P(J, a|E) = P(m|E) \prod_{k=1}^m P(a_k|a_1^{k-1}, j_1^{k-1}, m, E) P(j_k|a_1^k, j_1^{k-1}, m, E) \quad (2.4)$$

m は日本語の文長であり， a_1^{k-1} は日本語文の 1 単語目から $k-1$ 単語目までのアラインメントである．また， j_1^{k-1} は日本語文の 1 番目から $k-1$ 番目までの単語を示す．(2.4) 式の右辺は，パラメータが多く複雑なため，計算が困難である．そこで，Model1 では (2.4) 式のパラメータを簡略化する．

- 文の長さの確率 ϵ は, m と E に依存しない

$$\epsilon \equiv P(m|e)$$

- アライメントの確率は英語文の長さ l に依存する

$$P(a_k|a_1^{k-1}, J_1^{k-1}, m, E) \equiv (l+1)^{-1}$$

- 日本語の翻訳確率 $t(J_k|e_{a_k})$ は, 日本語単語 J_k に対応づけられる英単語 e_{a_k} に依存する

$$P(J_k|a_1^k, J_1^{k-1}, m, E) \equiv t(J_k|e_{a_k})$$

これらの定理を用いて, パラメータを簡略化した場合の $P(J, a|E), P(J|E)$ は以下になる.

$$P(J, a|E) = \frac{\epsilon}{(l+1)^m} \prod_{k=1}^m t(j_k|e_{a_k}) \quad (2.5)$$

$$P(J|E) = \frac{\epsilon}{(l+1)^m} \sum_{a_1=0}^l \cdots \sum_{a_m=0}^l \prod_{k=1}^m t(j_k|e_{a_k}) \quad (2.6)$$

$$= \frac{\epsilon}{(l+1)^m} \prod_{k=1}^m \sum_{i=0}^l t(j_k|e_i) \quad (2.7)$$

Model1 は, 翻訳確率 $t(j|e)$ の初期値が 0 以外の時, EM アルゴリズムを繰り返して得られる唯一の極大値より最適解を推定する. EM アルゴリズムは以下の手順で行われる.

手順 1 翻訳確率 $t(j|e)$ の初期値を設定する

手順 2 日本語と英語の対訳文 $(J^{(s)}, E^{(s)})$, $1 \leq s \leq S$ において, 日本語単語 j と英単語 e が対応する回数の期待値を計算する. ここで $\delta(j|j_k)$ は日本語文 J において日本語単語 j が出現する回数を表し, $\delta(e|e_i)$ は英語文 E において英単語 e が出現する回数を表す.

$$c(j|e; J, E) = \frac{t(j|e)}{t(j|e_0) + \dots + t(j|e_l)} \sum_{k=1}^m \delta(j|j_k) \sum_{i=1}^l \delta(e|e_i) \quad (2.8)$$

手順 3 英語文 $E^{(s)}$ のうち 1 回以上出現する英単語 e に対し, 翻訳確率 $t(j|e)$ を計算する. ここで S は日英対訳文の文数を表す.

- 定数 λ_e を以下の式で計算する .

$$\lambda_e = \sum_j \sum_{s=1}^S c(j|e; J^{(s)}, E^{(s)}) \quad (2.9)$$

- 翻訳確率 $t(j|e)$ を以下の式で再計算する .

$$\begin{aligned} t(j|e) &= \lambda_e^{-1} \sum_{s=1}^S c(j|e; J^{(s)}, E^{(s)}) \\ &= \frac{\sum_{s=1}^S c(j|e; J^{(s)}, E^{(s)})}{\sum_k \sum_{s=1}^S c(j|e; J^{(s)}, E^{(s)})} \end{aligned} \quad (2.10)$$

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

2.2.1.2 Model2

Model1 において , アライメントの確率は英語文の長さ l のみに依存する . そこで , Model2 では , k 単語目のアライメント a_k , 日本語文の長さ m にも依存するとし , 以下のように示す .

$$a(a_k|k, m, l) \equiv P(a_k|a_1^{k-1}, j_1^{k-1}, m, l) \quad (2.11)$$

これより , (2.6) 式は以下の式のようになる .

$$P(J|E) = \epsilon \sum_{a_1=0}^l \dots \sum_{a_m=0}^l \prod_{k=1}^m t(j_k|e_{a_k}) a(a_k|k, m, l) \quad (2.12)$$

$$= \epsilon \prod_{k=1}^m \sum_{i=0}^l t(j_k|e_{a_k}) \alpha(l|k, m, l) \quad (2.13)$$

Model2 において , 対訳文中の英単語 e と日本語単語 j が対応付けされる回数の期待値 $c(j|e; J^{(s)}, E^{(s)})$ と , 日本語単語の位置 j と英単語の位置 i が対応付けられる回数の期待値 $c(i|k, m, l; J^{(s)}, E^{(s)})$ が存在する . これらは以下の式で求められる .

$$\begin{aligned} c(j|e; J^{(s)}, E^{(s)}) &= \frac{t(j|e)}{t(j|e_0) + \dots + t(j|e_l)} \sum_{k=1}^m \delta(j|j_k) \sum_{i=1}^l \delta(e|e_i) \\ &= \frac{t(j|e) \alpha(i|k, m, l) \delta(f|f_k) \delta(e|e_i)}{t(j_k|e_0) \alpha(0|k, m, l) + \dots + t(j_k|e_l) \alpha(l|k, m, l)} \end{aligned} \quad (2.14)$$

$$\begin{aligned}
c(i|k, m, l; J^{(s)}, E^{(s)}) &= \sum_a P(a|e, j) \delta(i, a_k) \\
&= \frac{t(j|e) \alpha(i|k, m, l)}{t(j_k|e_0) \alpha(0|k, m, l) + \dots + t(j_k|e_l) \alpha(l|k, m, l)} \quad (2.15)
\end{aligned}$$

なお, Model2 は, EM アルゴリズムで計算した場合, 複数の極大値を持ち, 最適解を獲得できない場合がある. しかし, Model1 は, Model2 において, $a(i|k, m, l) = (l+1)^{-1}$ となる特殊な状態であり, 最適解を求めることができる. このため, Model2 で最適解を求めるとき, Model1 を用いる.

2.2.1.3 Model3

Model3 は, Model1 や Model2 とは異なり, 1 つの単語が複数の単語に対応する場合や, 単語の翻訳位置の距離を考慮する. なお, Model3 は以下 4 つのパラメータを用いる.

- $P(j|e)$
英単語 e が日本語単語 j に翻訳される確率
- $n(\phi|e)$
英単語 e が日本語単語に ϕ 個対応する確率
- $d(k|i, m, l)$
英語文の長さ l , 日本語文の長さ m のときに, i 番目の英単語 e_i が k 番目の日本語単語 j_k に翻訳される確率
- p_0, p_1
英単語に対応しない日本語単語が ϕ_0 個あるときの確率

なお p_0 は以下の式で表される.

$$P(\phi_0|\phi_1^l, e) = \binom{\phi_1 + \dots + \phi_l}{\phi_0} p_0^{\phi_1 + \dots + \phi_l} p_1^{\phi_0} \quad (2.16)$$

よって, Model3 は以下の式で表される.

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

$$\begin{aligned}
&= \sum_{a_1=0}^l \cdots \sum_{a_m=0}^l \binom{m - \phi_0}{\phi_0} p_0^{m-2\phi_0} p_1^{\phi_0} \prod_{i=1}^l \phi_i! n(\phi_i | e_i) \\
&\quad \times \prod_{k=1}^m t(j_k | e_{a_k}) d(k | a_k, m, l)
\end{aligned} \tag{2.17}$$

Model3 は、すべてのアライメントを計算する量が莫大であるため、EM アルゴリズムによる近似で求める。

2.2.1.4 Model4

Model4 は、Model2 や Model3 と違い、単語の位置を絶対位置ではなく相対位置で考慮する。また、各単語ごとの位置を考慮する。なお、Model4 は、歪み確率 $d(k|i, m, l)$ を以下の2通りで考慮する。

- 英単語に対応する日本語単語が1つ以上あるとき、その中で最も文頭に近い場合

$$P\left(\prod_{[i]1} = k | \pi_1^{[i]-1}, \gamma_0^l, \phi_0^l, E\right) = d_1(k - \odot_{i-1} | A(e_{[i-1]}), B(j_k)) \tag{2.18}$$

- それ以外の場合

$$P\left(\prod_{[i]x} = k | \pi_{[i]1}^{x-1}, \pi_1^{[i]-1}, \gamma_0^l, \phi_0^l, E\right) = d_{>1}(k - \pi_{[i]x-1} | B(j_k)) \tag{2.19}$$

ここで A は英単語のクラス、 B は日本語単語のクラスを表している。

2.2.1.5 Model5

Model4 は、単語の位置に関して直前の単語のみを考慮している。そのため、複数単語が同じ位置に生じたり、単語が存在しない位置に生成される問題がある。その問題を改善するために Model5 では、単語を空白部分に配置するように制約が施されている。

2.2.2 GIZA++

GIZA++[5]とは、統計翻訳に用いるための単語の確率値を計算するツールである。IBMモデルのModel1~5を用いて、翻訳する言語と目的言語における単語の対応関係の確率値を計算する。GIZA++を用いた場合、以下のファイルが出力される。

1. T TABLE (Translation Table)

T TABLEは、Model1からModel3により作成された翻訳確率 $P(f|e)$ のデータである。 f は翻訳する言語で、 e は目的言語である。T TABLEは各行が、目的言語の単語ID(e_id)、翻訳する言語の単語ID(f_id)、翻訳する言語の単語から目的言語の単語へ翻訳する確率($P(f_id|e_id)$)で構成される。

2. N TABLE (Fertility Table)

N TABLEは、目的言語の単語における繁殖数を表したデータである。N TABLEは各行が、目的言語の単語ID(e_id)、繁殖数が0である確率(p_0)、繁殖数が1である確率(p_1)、...、繁殖数が n である確率(p_n)で構成される。

2.2.3 フレーズテーブルの作成方法

GIZA++よりIBMモデルを推定することにより，最尤な単語アライメントを得る．これを日英・英日方向に対してそれぞれ行う．日本語文“風で松明が消えた”とその対訳英語文“The wind blew out the torch”を学習データとしたときの例として，日英方向の単語対応を表2.2に，英日方向の単語対応を表2.3に示す．また，表中の“●”は対応点である．

表 2.2: 日英方向の単語対応

	The	wind	blew	out	the	torch
風		●				
で			●			
松明						●
が			●			
消え			●			
た			●			

表 2.3: 英日方向の単語対応

	The	wind	blew	out	the	torch
風	●	●				
で					●	
松明			●	●		●
が						
消え						
た	●					

次に，両方向のアライメントから，両方向に1対多の対応を認めた単語アライメントの計算をヒューリスティックなルールにより行う．ヒューリスティックとは，ヒトの日々の意思決定に類似した直感的かつ発見的な思考方法である．基本のヒューリスティックとして，“intersection(積集合)”と，“union(和集合)”，“grow(成長)”，そして“grow-diag”がある．grow, grow-diag は intersection と union の中間である．intersection からスタートし，既に採用した対応点の周りに union の対応点を加えていく．grow では縦と横の方向に，grow-diag では縦と横と対角に union の対応点がある場合に，その対応点を用いる．表2.4に intersection の例を，表2.5に union の例を，表2.6に grow の例を，表2.7に grow-diag の例を示す．なお，表中の ● は，intersection から追加された対応点を示す．

表 2.4: intersection の例

	The	wind	blew	out	the	torch
風		•				
で						
松明						•
が						
消え						
た						

表 2.5: union の例

	The	wind	blew	out	the	torch
風	•	•				
で			•		•	
松明			•	•		•
が			•			
消え			•			
た	•		•			

表 2.6: grow の例

	The	wind	blew	out	the	torch
風		•				
で						
松明						•
が						
消え						
た						

表 2.7: grow-diag の例

	The	wind	blew	out	the	torch
風		•				
で						
松明						•
が						
消え						
た						

最後に、最終処理のヒューリスティックスとして、“final”と、“final-and”を用いる。finalは、少なくとも片方の言語の単語の単語対応がない場合に、unionの単語対応を追加する。また、final-andは、両側言語の単語の単語対応がない場合に、unionの候補対応点を追加する。表 2.8 に grow-diag-final の例を、表 2.9 に grow-diag-final-and の例を示す。ここでの表中の \bullet は、grow-diag から追加された対応点を示す。

表 2.8: grow-diag-final の例

	The	wind	blew	out	the	torch
風	•	•				
で			•		•	
松明						•
が						
消え						
た						

表 2.9: grow-diag-final-and の例

	The	wind	blew	out	the	torch
風	•	•				
で			•		•	
松明						•
が						
消え						
た						

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、その確率値をフレーズ対に付与することで、フレーズテーブルを作成する。

2.3 言語モデル

言語モデルは、モノリンガルデータから単語列の生成確率を付与するモデルである。日英翻訳では、翻訳モデルを用いて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ため、英語文の生成確率を用いることで、翻訳結果を出力することができる。

2.4 デコーダー

デコーダは翻訳モデルと言語モデルの全ての組み合わせから候補を探索して、確率が最大となるものを出力する。代表的なデコーダに `moses`[6] がある

また、デコーダには言語モデルや翻訳モデルに対して重みを与えることができる。各モデルに与える重みをパラメータと呼び、このパラメータを最適化するために、パラメータチューニングを行う。

第3章 N -gram(従来手法)

3.1 従来手法の概要

一般的な手法では，言語モデルに N -gram モデルを使用している．以下の式 (3.1) に本研究で使用する Tri-gram の計算式を示す．

$$\sum_{i=0}^{N-1} \log_2 \left(\frac{\text{count}(E_{i-2}, E_{i-1}, E_i)}{\text{count}(E_{i-2}, E_{i-1})} \right) \quad (3.1)$$

E_i : 出力英語単語 N : 英文の単語数 count : 対訳学習文の頻度

3.2 従来手法の問題点

Trigram は近隣の情報を利用している．そして遠隔の情報を利用せず，名詞と動詞の関係などを利用しない．そのため分野依存性が高くなる．この問題を解決するために，本実験では異なる言語モデルとして High order joint Probability を提案する．

第4章 High order Joint Probability(提案手法)

4.1 提案手法の概要

本研究では, 言語モデルに N -gram モデルの代わりに High order Joint Probability (以下 Joint Probability) を用いる. 以下の式 (2) にその計算式を示す [19].

$$\begin{aligned}
 & \sum_{j=0}^{M-1} \sum_{i=0}^{N-1} P(J_{j-2}, J_{j-1}, J_j, E_{i-2}, E_{i-1}, E_i) \\
 & \quad \times \log_2 \frac{P(J_{j-2}, J_{j-1}, J_j, E_{i-2}, E_{i-1}, E_i)}{P(J_{j-2}, J_{j-1}, J_j)P(E_{i-2}, E_{i-1}, E_i)} \\
 & = \sum_{j=0}^{M-1} \sum_{i=0}^{N-1} \frac{\text{count}(J_{j-2}, J_{j-1}, J_j, E_{i-2}, E_{i-1}, E_i)}{N_{je}} \\
 & \quad \times \log_2 \frac{\frac{\text{count}(J_{j-2}, J_{j-1}, J_j, E_{i-2}, E_{i-1}, E_i)}{N_{je}}}{\frac{\text{count}(J_{j-2}, J_{j-1}, J_j)}{N_j} \frac{\text{count}(E_{i-2}, E_{i-1}, E_i)}{N_e}}
 \end{aligned} \tag{4.1}$$

J_j : 日本語単語 M : 日本文の単語数

E_i : 英語単語 N : 英文の単語数

P : 出現確率 count : 対訳学習文中の頻度

N_{je} : 対訳学習文中の日本語単語の三つ組と英語単語の三つ組の総数

N_j : 対訳学習文中の日本語単語の三つ組の総数

N_e : 対訳学習文中の英語単語の三つ組の総数

4.2 提案手法の利点

N -gram は近隣の英単語の情報を使用している．しかし遠隔の情報が翻訳に必要な場合がある．例題として入力文を表 4.1 に，出力候補文を表 4.2 に示す．

N -gram(Tri-gram) は近隣の英単語の情報を使用する．候補文 1 では「tree」が「singing」の関係性を，候補文 2 では「tree」が「howling」の関係性をを用いる．しかし，この近接の単語情報だけでは，表 4.2 中の出力候補文の選択が困難である．

一方，Joint Probability は遠隔の日英単語の情報を使用できる．したがって，候補文 1 では「ウグイス」が「singing」の関係性を，候補文 2 では「ウグイス」が「howling」の関係性をを用いる．通常ウグイスは「singing」するが、「howling」しない．この遠隔の情報を使用することで，分野の違う言語データを追加しても翻訳精度が向上すると考える．

表 4.1: 入力文

入力文	木の上でウグイスが鳴いている。
-----	-----------------

表 4.2: 出力候補文

候補文 1	The nightingale on the tree is singing .
候補文 2	The nightingale on the tree is howling .

第5章 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)

本章は，中村ら [21] の論文を参照して記述した．

5.1 TDPBSMT の概要

“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)” は中村ら [21] が提案した翻訳手法である．

5.2 TDPBSMT の手順

本節で TDPBSMT の手順を示す．

5.2.1 学習の手順

TDPBSMT における学習は「変換テーブルの作成」と「文パターンの作成」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[4]) 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 5.1 に示す学習文対を使用して，表 5.2 に示す対訳単語を作成する．表 5.2 の値は例であり，実際の数値とは異なる．

手順 2 単語レベル文パターンの作成

学習文対内で対訳単語に当たる部分を変数化し，単語レベル文パターンを作成する．例を表 5.3 に示す．

表 5.1: 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 5.2: 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 3 変換テーブルの作成

学習文対と単語レベル文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 5.4 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．

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．

手順 5 変換テーブルを分割

変換テーブルを変換テーブル AB と変換テーブル CD に分割する．表 5.5 では 3 つの変換テーブルを分割している．

手順 6 文パターンを作成

表 5.3: 単語レベル文パターンの作成例

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.
単語レベル文パターン (日本語側)	$X0$ の $X1$ は $X2$ だ
単語レベル文パターン (英語側)	$X0$ $X1$ is a $X2$

表 5.4: 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.
単語レベル文パターン (日本語側)	$X0$ の $X1$ は $X2$ だ。
単語レベル文パターン (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

表 5.5: 変換テーブルの分割

変換テーブル $_{AB}1$	A : 振幅	B : amplitude
変換テーブル $_{CD}1$	C : 数値	D : value
変換テーブル $_{AB}2$	A : 大きくな	B : increased
変換テーブル $_{CD}2$	C : 走る	D : run
変換テーブル $_{AB}3$	A : 大きくな	B : grow
変換テーブル $_{CD}3$	C : 小さくな	D : was small

学習文対の，変換テーブル $_{AB}$ に当たる単語を変数化し，文パターンを作成する．
表 5.6 では 2 つの対訳単語を変数に置き換えている．

表 5.6: 文パターンの作成例

	日本語側	英語側
学習文対	その振幅は大きくなった。	The amplitude increased.
変換テーブル $_{AB}1$	A : 振幅	B : amplitude
変換テーブル $_{AB}2$	A : 大きくな	B : increased
作成される文パターン	その $X0$ は $X1$ った。	The $X0$ $X1$.

手順 7 文パターンに確率を付与

対訳単語確率 P_w を利用して，文パターンに確率を付与する．この確率を文パターン確率 P_p とする．以下にその手順を示す．

1. 文パターン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について，対数の総和を求める．

5.2.2 翻訳の手順

本節で翻訳の手順を示す。

手順1 入力文に日本語側の変換テーブルを適用

入力文中の変換テーブル $_{CD}$ に存在する語句を変数に変換し、文パターンに一致させる。表5.7に例を示す。

表 5.7: 入力文への変換テーブル $_{CD}$ 適用例

入力文	その数値は小さくなった。
変換テーブル $_{CD1}$	C : 数値
変換テーブル $_{CD3}$	C : 小さくな
一致する文パターン (日本語側)	その $X0$ は $X1$ った。

手順2 文パターンに変換テーブルの英語側を適用

文パターンの変数部に変換テーブル $_{CD}$ の英語側を代入し、出力候補文を作成する。表5.8に例を示す。

表 5.8: 文パターンへの変換テーブル $_{CD}$ 適用例

一致した文パターン (日本語側)	その $X0$ は $X1$ った。
一致した文パターン (英語側)	The $X0$ $X1$.
変換テーブル $_{CD1}$	D : value
変換テーブル $_{CD3}$	D : was small
出力候補文	The value was small.

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(5.1)に従って最終的な出力文を決定する。ここで P_m は言語モデルで得られる確率である。

$$\log P = \log P_v + \log P_p + \log P_m \quad (5.1)$$

図 5.1 に TDPBSMT の流れ図を示す .

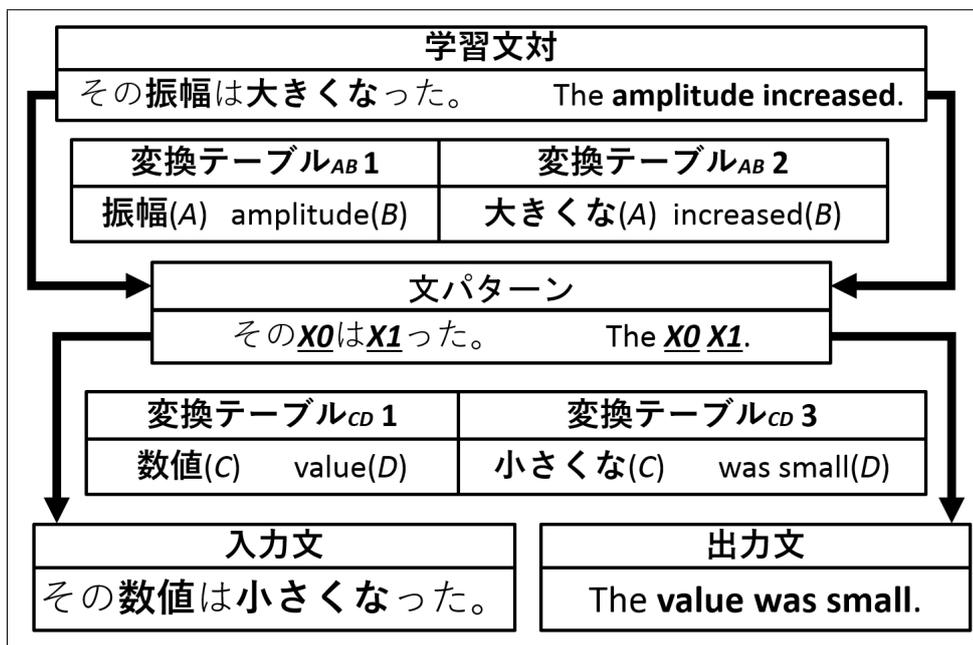


図 5.1: TDPBSMT の流れ図

また , 実際の実出力例を表 5.9 に示す .

表 5.9: TDPBSMT の出力例

入力文	視界は極めて良好であった。
文パターン (日本語側)	X_2 は $X_0 X_1$ であった。
文パターン (英語側)	The X_2 was $X_0 X_1$.
変換テーブル $_{CD}(X_0)$	C : 極めて D : extremely
変換テーブル $_{CD}(X_1)$	C : 良好 D : good
変換テーブル $_{CD}(X_2)$	C : 視界 D : sight
出力文	The sight was extremely good.

第6章 評価方法

本章は、善行ら [18] の論文を参照して記述した。本研究では、出力結果の評価に自動評価と人手評価を用いる。

6.1 自動評価

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、翻訳システムで出力した出力文を比較する手法が一般的である。本研究では、BLEU[7]、NIST[7]、METEOR[8]、RIBES[9]、TER[10] を用いる。

6.1.1 BLEU

BLEU は機械翻訳分野において、一般的に使用される自動評価の手法である。BLEU は語順 (4-gram) が正しい場合に高いスコアを出す。BLEU は以下の式で計算される。

$$BLEU_{score} = BP \times \exp \left(\sum_{n=1}^N \frac{1}{N} \log p_n \right) \quad (6.1)$$

$$p_n = \frac{\sum_i \text{出力文 } i \text{ と正解文 } i \text{ で一致した } N\text{-gram の数}}{\sum_i \text{出力文 } i \text{ の } N\text{-gram の数}} \quad (6.2)$$

ここで、 P_n は出力文と正解文の N -gram の一致率を表している。BLEU はこの一致率を 1-gram から 4-gram まで計算し、その幾何平均をとる。また、出力文が正解文より短い場合、“ $\sum_i \text{出力文 } i \text{ の } N\text{-gram の数}$ ” が小さくなり、不当にスコアが高くなる可能性がある。そこで、正解文より短い文に対するペナルティとして、 BP を用いる。 BP の条件式を以下に示す。

$$BP = \begin{cases} 1 & (c > r) \\ e^{1-\frac{r}{c}} & (c \leq r) \end{cases} \quad (6.3)$$

ここで、 c は出力文の単語数、 r は正解文の単語数を表している。出力文が正解文より長い場合は $BP = 1$ となり、出力文が正解文より短い場合は、その長さに応じて BP が小さくなり、BLEU 値も小さくなる。BLEU は 0 から 1 の値をスコアとして出力し、スコアの値が高いほど翻訳精度が高いことを示す。

6.1.2 NIST

NIST では BLEU と同様に語順 (5-gram) の正しさで比較を行うが、BLEU とは情報量によって重み付けをしている点が異なる。また、ペナルティ関数が異なっており、NIST の方が文長を考慮するペナルティが課せられる。NIST は以下の式で計算される。

$$NIST_{score} = BP \times \sum_{N=1}^N \frac{\sum_i \left(\sum_{\substack{\text{出力文 } i \text{ と正解文 } i \text{ に} \\ \text{共通する } w_1 \dots w_n}} Info_i(w_1 \dots w_n) \right)}{\sum_i \text{出力文 } i \text{ と正解文 } i \text{ で一致した } N\text{-gram の数}} \quad (6.4)$$

$$Info_i(w_1 \dots w_n) = \log_2 \frac{\text{評価コーパス中 } (w_1 \dots w_{n-1}) \text{ の数}}{\text{評価コーパス中の } (w_1 \dots w_n) \text{ の数}} \quad (6.5)$$

NIST は 0 から ∞ の値をスコアとして出力し、スコアの値が高いほど翻訳精度が高いことを示す。

6.1.3 METEOR

METEOR は再現率 R と適合率 P に基づく F 値に対して、単語の非連続性に対するペナルティ関数 Pen を利用した評価基準である。METEOR は以下の式で計算される。

$$METEOR_{score} = F_{mean} \times (1 - Pen) \quad (6.6)$$

$$F_{mean} = \frac{P \times R}{\alpha \times P + (1 - \alpha) \times R} \quad (6.7)$$

$$Pen = \gamma \times \left(\frac{c}{m}\right)^\beta \quad (6.8)$$

ペナルティ関数 Pen において、 m は出力文と正解文の単語の一致率を表す。そして、 c は一致した単語を対象に、正解文と語順が同じものを 1 つのまとまりとして統合した場合の、まとまりの数を表す。そのため、出力文と正解文が同じ文であるとき $c = 1$ となる。また、一致率の計算において、WordNet による類義語を用いて、似た意味を持つ単語は同一であると判断される。METEOR は単語属性 (3 人称単数など) が正しい場合に、高い

スコアを出す特性を持つ． α, β, γ の値はパラメータである．METEORは0から1の値をスコアとして出力し，スコアの値が高いほど翻訳精度が高いことを示す．METEORは日本語辞書が存在しないため，本研究では出力文が日本語文の英日翻訳では使用しない．

6.1.4 RIBES

RIBESは，正解文と出力文の間で，共通単語の出現順序を順位相関係数を用いて，評価を行う自動評価法である．RIBESは以下の式で計算される．

$$RIBES(S) = NSR \times P^\alpha \quad (6.9)$$

$$RIBES(K) = NKT \times P^\alpha \quad (6.10)$$

$$P = \frac{n}{h} \quad (6.11)$$

ここで， NSR はスピアマンの順位相関係数であり， NKT はケンドールの順位相関係数である．また P はペナルティであり， n は出力文と正解文との間で共通な単語の数， h は出力文の単語数で計算する． α はペナルティに対する主として使用され， $0 \leq \alpha \leq 1$ の値である．単語の出現順を順位相関係数を用いて評価することで，文全体の語順に着目することができる．RIBESは0から1の値をスコアとして出力し，スコアの値が高いほど翻訳精度が高いことを示す．

6.1.5 TER

TERはエラーレートであり，WER[10]に句単位での並び替えを加えて語順の制約を緩めた自動評価法である．TERは以下の式で計算される．

$$TER = \frac{\text{挿入語数} + \text{削除語数} + \text{置換語数} + \text{並び替え句数}}{\text{正解文の語数}} \quad (6.12)$$

TERは0から100の値をスコアとして出力し，スコアの値が低いほど翻訳精度が高いことを示す．また，WERについて以下に示す．

WER

WERはエラーレートであり，出力文を正解文に変換するために必要な単語単位の操作(挿入、削除、置換)の回数の割合を表す自動評価法である．WERは以下の式で計算

される。

$$TER = \frac{\text{挿入語数} + \text{削除語数} + \text{置換語数}}{\text{正解文の語数}} \quad (6.13)$$

WER は語順に敏感であり，日本語などの語順の制約の緩い言語では，意味が通る文に対しても，大きなペナルティを課してしまう欠点がある。

6.2 人手評価

本研究では人手評価として，対比較評価を用いる。対比較評価は，二つの文を相対的に比較して，どちらの品質がより優れているかを人手で選択する評価方法である。対比較を行う出力例を表 6.1 に，評価基準を表 6.2 示す。

表 6.1: 対比較の出力例

入力文	I'm always ready to help .
正解文	いつでも喜んでお手伝いします。
ベースライン	私にはいつもました。
提案手法	いつでも助けます。

表 6.2: 対比較の評価基準

ベースライン	ベースラインの出力文が提案手法の出力文より翻訳品質が優れている
提案手法	提案手法の出力文がベースラインの出力文より翻訳品質が優れている
差なし	2つの出力文の翻訳品質に明確な差がない
同出力	2つの出力文が完全に同じ

表 6.1 では，ベースラインの出力文と提案手法の出力文を比較すると，提案手法の出力結果の方が優れている。よって，この例では表 6.2 において“提案手法”を選択する。また，統計翻訳における対比較評価では，翻訳品質を比較するため，必ずしも意味が通る文である必要がない。よって“差なし”や“同出力”では，どちらも意味が通る文である場合と，どちらも意味が通らない文である場合が存在する。

第7章 実験方法

7.1 実験の概要

本実験ではデコーダーに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（以下 TDPBSMT）[?] を用いる．言語モデルは N -gram と Joint Probability を利用する．対訳学習文には辞書文で単文を用いる．テスト文も辞書文で単文を用いる．追加データには wikipedia 文を用いる．翻訳の評価として自動評価と人手評価を用いる．

7.2 実験データ

表 7.1 に実験データを示す．対訳学習文には辞書文で単文 [20] を使用する．テスト文も辞書文で単文を使用する．追加データには、学習文とは別分野のデータである wikipedia 文 [21] を使用する．表 7.2 に対訳学習文と wikipedia 文の平均単語長を示す．表 7.3 にテスト文と wikipedia 文の例を示す．

表 7.1: 実験データ

対訳学習文（辞書文）	159,998 文
テスト文（辞書文）	100 文
追加データ（wikipedia 文）	343,308 文

表 7.2: 平均単語長

対訳学習文	8.7
wikipedia 文	29.4

表 7.3: データ例

テスト文 (辞書文)	
日文 1	ピアノの勉強にヨーロッパに行く。
英文 1	Go to Europe to study the piano .
日文 2	この点ではご同意ができかねます。
英文 2	I cannot agree with you on this point .
日文 3	赤字が埋まった。
英文 3	The loss was covered .
wikipedia 文	
日文 1	彼が大統領に選ばれる公算が大きい。
英文 1	There is every probability that he will be elected President .
日文 2	彼らにとっては幹部に認めてもらう機会であり、腕の見せ所となっている。
英文 2	Playing a " mitate " is an opportunity for them to show their skills and get recognition from high-ranking people .
日文 3	日本からは銅や工芸品が、朝鮮からは木綿や朝鮮人参が送られていた。
英文 3	In those days , copper products and handicrafts from Japan and cotton materials and Asian ginseng from Korea were traded .

第8章 実験結果

8.1 自動評価 (wikipedia文追加) Joint Probability vs Tri-gram

表 8.1 に自動評価結果を示す .

表 8.1: 自動評価結果 (wikipedia 文追加) Joint Probability vs Tri-gram

wikipedia 文追加	BLEU	NIST	METEOR	RIBES	TER
Joint Probability	0.09	3.46	0.38	0.72	0.72
Tri-gram	0.03	2.74	0.29	0.68	0.74

8.2 人手評価 (wikipedia 文追加) Joint Probability vs Tri-gram

得られた出力結果を人手にて評価した . 評価基準を以下に示す . また , 人手評価結果を以下の表 8.2 に示す .

- Joint Probability : Joint Probability の結果が Tri-gram よりも良い
- Tri-gram : Tri-gram の結果が Joint Probability よりも良い
- 差なし : Tri-gram と Joint Probability の翻訳精度に差がない
- 同一 : Tri-gram と Joint Probability の出力結果が同じ
- 未出力 : Tri-gram と Joint Probability の出力結果が共に未出力

表 8.2: 人手評価結果 (wikipedia 文追加) Joint Probability vs Tri-gram

Joint Probability	Tri-gram	差なし	同一	未出力
23	2	58	12	5

表 8.2 における Joint Probability の出力例を表 8.3 , 8.4 , 8.5 に示す .

表 8.3: Joint Probability の出力例 1

入力文	中国人 が 印刷 技術 を 発明 した 。
参照文	The Chinese invented printing .
Joint Probability	The Chinese invented the printing .
Tri-gram	The printing invented the Chinese .

表 8.4: Joint Probability の出力例 2

入力文	公園 は 川 まで 広がっ ている 。
参照文	The park reaches to the river .
Joint Probability	The park is spread to the river .
Tri-gram	park spreading river .

表 8.5: Joint Probability の出力例 3

入力文	この 言葉 は 一般 に 通用 して いる 。
参照文	The words are in common currency .
Joint Probability	The word is understood in the public .
Tri-gram	The word spoken the public .

表 8.2 における Tri-gram の出力例を表 8.6 , 表 8.7 に示す .

表 8.6: Tri-gram の出力例 1

入力文	鞆が肩に掛かっている。
参照文	The bag is hung over his shoulders .
Joint Probability	He is the 掛かつ bag shoulders .
Tri-gram	The bag hanging shoulder .

表 8.7: Tri-gram の出力例 2

入力文	時が彼女の悲しみをいやした。
参照文	Time healed her sorrow .
Joint Probability	She Time healed his grief .
Tri-gram	Time healed She grief .

8.3 誤出力の解析 (Wikipedia 文)

表 8.7 における Joint Probability の出力の解析結果を以下の表 8.8 に示す .

表 8.8: 解析結果

入力文	時 が 彼女の 悲しみを いやした 。
参照文	Time healed her sorrow .
文パターン (日本語側)	N01 N00 N03 を N02 。
文パターン (英語側)	N 00 N 01 N 02 his N 03 .
変換テーブル _{CD} (N00)	C : 彼女 D : She
変換テーブル _{CD} (N01)	C : 時 が D : Time
変換テーブル _{CD} (N02)	C : いやした D : healed
変換テーブル _{CD} (N03)	C : の 悲しみ D : grief
出力文	She Time healed his grief .

表 8.8 より , N00 の変換テーブルの C に「彼女の」ではなく「彼女」が該当しているため , 本来「Her」が適切であるが「She」が出力されたことが分かった .

8.4 評価結果のまとめ

表 8.1 と表 8.2 の結果より , 対訳学習文とは別分野のデータを追加データとして用いた場合 , Joint Probability の方が Tri-gram よりも良いことが分かる . 特に人手評価については大きな差が見られた .

第9章 追加実験

9.1 追加前 (Joint Probability vs Tri-gram)

Tri-gram と Joint Probability の性能を比較するため，表 7.1 において追加データを利用せずに実験を行った．表 9.1 に実験データを，表 9.2 に自動評価結果を，表 9.3 に人手評価結果を示す．

表 9.1: 実験データ

対訳学習文	159,998 文
テスト文	100 文

表 9.2: 自動評価結果：追加前 (Joint Probability vs Tri-gram)

	BLEU	NIST	METEOR	RIBES	TER
Joint Probability (追加前)	0.12	3.60	0.41	0.72	0.69
Tri-gram (追加前)	0.10	3.35	0.38	0.71	0.71

表 9.3: 人手評価結果：追加前 (Joint Probability vs Tri-gram)

Joint Probability	Tri-gram	差なし	同一	未出力
14	6	48	27	5

表 9.3 における Joint Probability の出力例を表 9.4, 表 9.5, 表 9.6 に示す.

表 9.4: Joint Probability の出力例 1

入力文	この言葉は一般に通用している。
参照文	The words are in common currency .
Joint Probability	This word is not accepted for the public .
Tri-gram	This word is not accepted for the public .

表 9.5: Joint Probability の出力例 2

入力文	彼女の結婚は一大行事となるだろう。
参照文	Her marriage will be a great event .
Joint Probability	Her marriage will be a great event .
Tri-gram	She will be a great event in marriage .

表 9.6: Joint Probability の出力例 3

入力文	私はその仕事に慣れていない。
参照文	I am not used to the task .
Joint Probability	I am not used to the work .
Tri-gram	I was not used to the job .

表 9.3 における Tri-gram の出力例を表 9.7, 表 9.8, 表 9.9 に示す.

表 9.7: Trigram の出力例 1

入力文	鞆 が 肩 に 掛かっ て いる 。
参照文	The bag is hung over his shoulders .
Joint Probability	He is 掛かつ shoulders of the bag .
Tri-gram	the bag is hanging on the shoulders .

表 9.8: Trigram の出力例 2

入力文	幕 が だらり と 垂れ下 がっ て いる 。
参照文	The curtain hangs down loosely .
Joint Probability	The curtain is drooping tongue .
Tri-gram	lollod curtain is hanging down .

表 9.9: Trigram の出力例 3

入力文	彼 は その 白馬 を 追いか け た 。
参照文	He went after the white horse .
Joint Probability	白馬 He was after the right .
Tri-gram	He follows the 白馬 off .

表 9.2 と表 9.3 の結果より, 追加前においても, JointProbability の方が Tri-gram よりも良いことが分かる.

9.2 誤出力の解析 (追加前)

表 9.8 における Joint Probability の出力の解析結果を以下の表 9.10 に示す .

表 9.10: 解析結果

入力文	幕 が だらり と 垂れ下がっ ている 。
参照文	The curtain hangs down loosely .
文パターン (日本語側)	N02 が N00 N01 ている 。
文パターン (英語側)	The N 02 is N 01 N 00 .
変換テーブル _{CD} (N00)	C : だらり と D : tongue
変換テーブル _{CD} (N01)	C : 垂れ下がっ D : drooping
変換テーブル _{CD} (N02)	C : 幕 D : curtain
出力文	The curtain is drooping tongue .

表 9.10 より , N00 の変換テーブルの C の「だらりと」に対して不適切な「tongue」が該当しているため誤った出力となった .

9.3 同分野 (複文) Joint Probability vs Tri-gram

対訳学習文と同分野のデータを追加した場合の比較を行った．表 9.11 に実験データを示す．言語モデルには Tri-gram と Joint Probability を使用した．追加データには辞書文から抽出した複文を利用した．表 9.12 に学習文と複文の平均単語長を示す．表 9.13 に複文の例を示す．表 9.14 に自動評価結果を示す．表 9.15 に人手評価結果を示す．

表 9.11: 実験データ

対訳学習文 (辞書文の単文)	159,998 文
テスト文 (辞書文の単文)	100 文
追加データ (辞書文の複文)	100,000 文

表 9.12: 平均単語長

対訳学習文	8.7
複文	11.3

表 9.13: データ例

複文	
日文 1	そんな事があったとは少しも知りませんでした。
英文 1	I did not know at all that there was such a thing .
日文 2	貴社からのこれまでのご援助にお礼を申し上げますとともに、両社の素晴らしい関係が今後とも続きますことを期待しております。
英文 2	I also want to thank you for your past support and I am looking forward to continued excellent relations between our companies .
日文 3	彼はそれを聞くと、驚いた様子をした。
英文 3	He looked surprised when he heard it .

表 9.14: 自動評価結果 (複文)

複文追加	BLEU	NIST	METEOR	RIBES	TER
Joint Probability	0.15	3.67	0.42	0.75	0.68
Tri-gram	0.11	3.39	0.38	0.69	0.71

表 9.15: 人手評価結果 (複文)

Joint Probability	Tri-gram	差なし	同一	未出力
17	6	45	27	5

表 9.15 における Joint Probability の出力例を表 9.16, 表 9.17, 表 9.18 に示す.

表 9.16: Joint Probability の出力例 1

入力文	溶岩が少し後退した。
参照文	The lava has retreated a little .
Joint Probability	The lava retreated little .
Tri-gram	little lava retreated .

表 9.17: Joint Probability の出力例 2

入力文	私はその仕事に慣れていない。
参照文	I am not used to the task .
Joint Probability	I am not used to the work .
Tri-gram	I was not used to the job .

表 9.18: Joint Probability の出力例 3

入力文	中国人が印刷技術を発明した。
参照文	The Chinese invented printing .
Joint Probability	The Chinese invented printing .
Tri-gram	He invented a printing of the Chinese .

表 9.15 における Tri-gram の出力例を表 9.19, 表 9.20, 表 9.21 に示す.

表 9.19: Tri-gram の出力例 1

入力文	この言葉は一般に通用している。
参照文	The words are in common currency .
Joint Probability	This word has accepted the public .
Tri-gram	This word is spoken in public .

表 9.20: Tri-gram の出力例 2

入力文	彼の性質はひねくれている。
参照文	He has an uneven disposition .
Joint Probability	He has distorted in the nature of his .
Tri-gram	His character is twisted .

表 9.21: Tri-gram の出力例 3

入力文	幕がだらりと垂れ下がっている。
参照文	The curtain hangs down loosely .
Joint Probability	The curtain is drooping tongue .
Tri-gram	lolloped curtain is hanging down .

表 9.14 と表 9.15 の結果より, 対訳学習文と同分野のデータを追加データとした場合, Joint Probability の方が Tri-gram よりも良いことが分かる.

9.4 誤出力の解析(複文)

表 9.20 における Joint Probability の出力の解析結果を以下の表 9.22 に示す。

表 9.22: 解析結果

入力文	彼の性質はひねくれている。
参照文	He has an uneven disposition .
文パターン(日本語側)	N03 の N04 N01 N02 N00 。
文パターン(英語側)	N 00 N 02 in N 01 N 04 of N 03 .
変換テーブル _{CD} (N00)	C : いる D : He
変換テーブル _{CD} (N01)	C : は D : the
変換テーブル _{CD} (N02)	C : ひねくれて D : has distorted
変換テーブル _{CD} (N03)	C : 彼 D : his
変換テーブル _{CD} (N04)	C : 性質 D : nature
出力文	He has distorted in the nature of his .

表 9.22 より, N00 の変換テーブルの C の「いる」に対して不適切な「He」が該当しているため誤った出力となった。

9.5 評価結果のまとめ

どの場合においても Joint Probability の方が Tri-gram よりも良かった。同一分野の言語データを追加した場合でも, Joint Probability の方が Tri-gram よりも良かった。他分野の言語データを追加した場合は, 特に Joint Probability が有効であることが分かった。

9.6 Joint Probability:追加前 vs 複文

Joint Probability について追加前と複文追加後の比較を行った．人手評価結果を以下の表 9.23 に示す．

表 9.23: 人手評価結果 (Joint Probability) 追加前 vs 複文

追加前	複文	差なし	同一	未出力
2	6	27	60	5

9.7 Tri-gram:追加前 vs 複文

Tri-gram について追加前と複文追加後の比較を行った．人手評価結果を以下の表 9.24 に示す．

表 9.24: 人手評価結果 (Tri-gram) 追加前 vs 複文

追加前	複文	差なし	同一	未出力
3	2	17	73	5

表 9.23 と表 9.24 より同一分野のデータを追加した際に Tri-gram よりも Joint Probability の方が翻訳精度が良いことが分かる．

9.8 Joint Probability:追加前 vs Wikipedia文

Joint Probability について追加前と Wikipedia 文追加後の比較を行った．人手評価結果を以下の表 9.25 に示す．

表 9.25: 人手評価結果 (Joint Probability) 追加前 vs Wikipedia 文

追加前	Wikipedia 文	差なし	同一	未出力
12	6	53	24	5

9.9 Tri-gram:追加前 vs Wikipedia文

Tri-gram について追加前と Wikipedia 文追加後の比較を行った．人手評価結果を以下の表 9.26 に示す．

表 9.26: 人手評価結果 (Tri-gram) 追加前 vs Wikipedia 文

追加前	複文	差なし	同一	未出力
22	1	57	15	5

表 9.25 と表 9.26 より別分野のデータを追加した場合，Tri-gram は翻訳精度が著しく低下するが，Joint Probability は Tri-gram と比較すると低下率が少ないことが分かる．

第10章 考察

TDPBSMT は確率を用いて最終的な出力文を決定する．その計算式を以下に示す．

$$\log_2 P = \log_2 P_v + \log_2 P_p + \log_2 P_m \quad (10.1)$$

P : 翻訳確率

P_v : 対訳学習文中の単語の出現回数に基づく確率

P_p : 文パターンに付与された確率

P_m : 言語モデルで生成された確率

10.1 Joint Probability の解析

表 8.3 における Tri-gram の出力候補文の翻訳確率を表 10.1 に，Joint Probability の出力候補文の翻訳確率を表 10.2 に示す．

表 10.1: 出力候補文の翻訳確率 (Tri-gram)

出力候補文	The printing invented the Chinese .
$\log_2 P_v$	-16.4
$\log_2 P_p$	-7.64
$\log_2 P_m$ (Tri-gram)	-71.8
$\log_2 P_m$ (Joint Probability)	-2539.3
$\log_2 P$	-95.8

表 10.2: 出力候補文の翻訳確率 (Joint Probability)

出力候補文	The Chinese invented the printing .
$\log_2 P_v$	-13.2
$\log_2 P_p$	-5.94
$\log_2 P_m$ (Tri-gram)	-90.9
$\log_2 P_m$ (Joint Probability)	-2474.7
$\log_2 P$	-2493.8

表 10.1 と表 10.2 より , $\log_2 P_m$ (Tri-gram) の値は表 10.1 の方が表 10.2 よりも大きいことが分かる . また , $\log_2 P_m$ (Joint Probability) の値は , 表 10.2 の方が表 10.1 よりも大きいことが分かる .

10.2 Tri-gram の解析

表 8.6 における Tri-gram の出力候補文の翻訳確率を表 10.3 に , Joint Probability の出力候補文の翻訳確率を表 10.4 に示す .

表 10.3: 出力候補文の翻訳確率 (Tri-gram)

出力候補文	The bag is hung over his shoulders .
$\log_2 P_v$	-18.6
$\log_2 P_p$	-4.80
$\log_2 P_m$ (Tri-gram)	-82.8
$\log_2 P_m$ (Joint Probability)	-2082.7
$\log_2 P$	-106.2

表 10.4: 出力候補文の翻訳確率 (Joint Probability)

出力候補文	He is the 掛かつ bag shoulders .
$\log_2 P_v$	-147.6
$\log_2 P_p$	-6.86
$\log_2 P_m$ (Tri-gram)	-91.7
$\log_2 P_m$ (Joint Probability)	-1813.6
$\log_2 P$	-1968.1

表 10.3 と表 10.4 より, $\log_2 P_m$ (Tri-gram) の値は表 10.3 の方が表 10.1 よりも大きいことが分かる。また, $\log_2 P_m$ (Joint Probability) の値は, 表 10.4 の方が表 10.3 よりも大きいことが分かる。

10.3 解析の評価

10.1 項と 10.2 項の結果の比較より, Joint probability の翻訳確率と Tri-gram の翻訳確率に同様の傾向が見られた。Tri-gram と判断したもう 1 文についても同様の傾向が見られた。この事は方式の限界を示している。また, P_v を見ると言語モデル無しで正しい出力候補文を選択できる可能性がある。今後は他にも様々な言語モデルがあるので, それらを用いて実験を行いたいと考えている。

第11章 おわりに

本研究では，言語モデルに一般的に使用される Tri-gram の代わりに Joint Probability を使用する手法を提案した．実験結果より，学習データと分野の異なるデータを追加した場合，Joint Probability では翻訳精度が大きく向上することが確認できた．また，追加実験より，追加データを利用しない場合と追加データに対訳学習文と同一分野のデータを追加する場合のどちらについても，High order Joint Probability の方が Tri-gram よりも翻訳精度が良いことが確認できた．今後は他にも様々な言語モデルがあるので，それらを用いて実験を行いたいと考えている．

謝辞

最後に，1年間に渡りご指導いただきました鳥取大学工学部電気情報系学科自然言語処理研究室の村上仁一准教授，村田真樹教授そして自然言語処理研究室の方々に厚く御礼申し上げます．

また，参考にさせていただいた論文の著者の方々に対して深く感謝申し上げます．

参考文献

- [1] Li et al.: “Parallel Aligned Treebanks at LDC: New Challenges Interfacing Existing Infrastructures”, LREC 2012, pp.1848-1855, 2012.
- [2] Ruiqiang Zhang and Eiichiro Sumita: “Boosting Statistical Machine Translation by Lemmatization and Linear Interpolation”, Proceedings of the ACL 2007, pp.181-184, 2007.
- [3] 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.
- [4] Peter F.Brown, Stephen A.Della Pietra, Vincent J.Della Pietra, Robert L.Mercer, “ The mathematics of statistical machine translation:Parameter Estimation ”, Computational Linguistics, 1993.
- [5] GIZA++ : <http://www.fjoch.com/GIZA++>
- [6] Moses: Philipp Koehn, Marcello Federico, Brooke Cowan, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, Evan Herbst, “Moses: Open Source Toolkit for Statistical Machine Translation”, Proceedings of the ACL 2007 Demo and Poster Sessions, pp.177-180, 2007.
- [7] BLEU: Papineni Kishore, Salim Roukos, Todd Ward, Wei-Jing Zhu, “BLEU: a method for automatic evaluation of machine translation”, 40th Annual meeting of the Association for Computational Linguistics, pp.311-318, 2002.
- [8] METEOR: Banerjee Satanjeev, Lavie Alon, “METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments”, Proceedings of Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization at the 43th Annual Meeting of the Association of Computational Linguistics (ACL-2005), pp.65-72, 2005.

- [9] RIBES: 平尾努, 磯崎秀樹, Kevin Duh, 須藤克仁, 塚田元, 永田昌明, “RIBES: 順位相関に基づく翻訳の自動評価法”, 言語処理学会第17年次大会発表論文集, pp.1111-1114, 2011.
- [10] Richard Schwartz, Linnea Micciulla, John Makhoul. “A Study of Translation Edit Rate with Targeted Human Annotation”, AMTA, pp.223-231, 2006.
- [11] SRILM: Andreas Stolcke, “SRILM - an Extensible Language Modeling Toolkit”, 7th International Conference on Spoken Language Processing, pp.901-904, 2002.
- [12] Mert: Franz Josef Och: “Minimum Error Rate Training in Statistical Machine Translation”, In Proceeding of the 41st Annual Meeting of the Association for Computational Linguistics, pp.160-167, 2003.
- [13] MeCab: Taku Kudo, Kaoru Yamamoto, Yuji Matsumoto, “Applying Conditional Random Fields to Japanese Morphological Analysis”, Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP-2004), pp.230-237, 2004.
- [14] tokenizer.sed
<http://www.cis.upenn.edu/treebank/tokenizer.sed>
- [15] 松本拓也, 村上仁一, 徳久雅人 “機械翻訳における人手評価と自動評価の考察”, NLP-2012 , pp.505-508, 2012
- [16] Thorsten Brants, Ashok C. Popat, Peng Xu, Franz J. Och, and Jeffrey Dean, “Large Language Models in Machine Translation”, EMNLP-2007, pp.858-867, 2007.
- [17] Holger Schwenk, “Investigations on Large-Scale Lightly-Supervised Training for Statistical Machine Translation”, Proceedings of IWSLT 2008, pp.182-189, 2008.
- [18] 善行 佑介, 村上 仁一, 徳久 雅人, “モノリンガルデータを増加させた場合の統計的機械翻訳の精度調査”, 言語処理学会第21回年次大会, 2014.
- [19] X.D.HAUNG, “HIDDEN MARKOV MODELS FOR SPEECH RECOGNITION”, p.49.
- [20] Wikipedia 日英京都関連文書対訳コーパス, <https://alaginrc.nict.go.jp/WikiCorpus/>

- [21] 中村 勇太, 村上 仁一, “ パターンに基づく変換主導型統計機械翻訳 (TDPBSMT) の提案 ”, 言語処理学会第 25 回年次大会, 2019.