文節区切りの学習データを用いた，日英統計翻訳の検討

猪澤雅史　村上仁一　徳久雅人　池原悟
鳥取大学 工学部 知能情報工学科
{s042009,murakami,tokuhisa,ikehara}@ike.tottori-u.ac.jp

1　はじめに

現在，機械翻訳において，対訳データから自動的に翻訳規則を獲得し，翻訳を行う統計翻訳 [1] が注目されている．日英統計翻訳において，日本語文はいくつかのフレーズ対を用いて変換され，順序を並び替え，英語文に翻訳される. しかし，重文複文といった複雑な日本語文を翻訳する場合，多くのフレーズ対が必要となる．そのため，並び替えの候補が膨大になり，翻訳精度が低くなる傾向がある [2]．

そこで本研究では，長いフレーズを持つフレーズ対を増すことで，翻訳に利用するフレーズ対の数を減らし，並び替えの候補を減らす手法を提案する．具体的には，学習データの日本語文を文節区切りとし，日本語フレーズが長いフレーズテーブルを生成する．そして，従来の形態素区切りのフレーズテーブルと併用し，翻訳精度の向上を目指す．

2　日英統計翻訳システム

2.1　基本的な考え方

日英統計翻訳は，日本語文 j が与えられたとき，全ての組み合わせの中から確率が最大になる英語文 $\hat{e}$ を探索することによって翻訳を行う．以下に基本モデルを示す．

$$\begin{aligned}\hat{e} &= argmax_e P(e \mid j) \\ &\simeq argmax_e P(j \mid e)P(e)\end{aligned}$$

$P(j \mid e)$ は翻訳モデル，$P(e)$ は言語モデルと呼ぶ．また，$\hat{e}$ を探索する翻訳システムをデコーダと呼ぶ．

2.2　翻訳モデル

翻訳モデルは日本語の単語列から英語の単語列へ確率的に翻訳を行うためのモデルである．翻訳モデルには，大きくわけて語に基づく翻訳モデルと句に基づく翻訳モデル [3] がある．現在は句に基づく翻訳モデルが主流となっている．句に基づく翻訳モデルは表 1 に示すフレーズテーブルと呼ばれる表で管理される．

表 1: フレーズテーブルの例

庭 ||| the garden ||| 0.043 0.38 0.05 0.02
庭 から ||| from the garden ||| 0.5 0.17 1 0.03
庭 で ||| at garden ||| 0.333 0.165 0.0833 0.048

左から ,日本語フレーズ ,英語フレーズ ,フレーズの英日翻訳確率 $P(j \mid e)$，英日方向の単語の翻訳確率 (IBM モデル) の積，フレーズの日英翻訳確率 $P(e \mid j)$，日英方向の単語の翻訳確率 (IBM モデル) の積である．本稿では，日本語フレーズ，英語フレーズ，各種確率の 3 つをまとめて，フレーズ対と呼ぶ．

2.3　言語モデル

言語モデルは単語列に対して，それらが生成される確率を与えるモデルである．日英翻訳では，言語モデルを用いて，訳文候補の中から英語として自然な文を選出する．言語モデルとして代表的なものに N-gram モデルがある．

3　提案手法

本章では，提案手法である文節区切りの学習データを用いたフレーズテーブルの学習について説明する．本手法の枠組みを図 1 に示す．

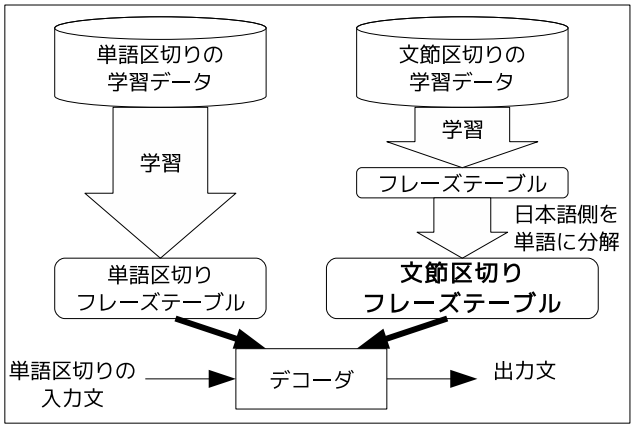

図 1: 本実験の枠組み

日英統計翻訳において，一般に学習データの日本語文は形態素解析を用いて，単語に区切られる．そして，単語区切りの学習データを用いて，フレーズテーブルを学習する．本稿では，単語区切りの学習データから学習されるフレーズテーブルを単語区切りフレーズテーブルと呼ぶ．しかし，単語区切りフレーズテーブルは単語対応のフレーズ対や短いフレーズを持つフレーズ対が多いため，翻訳に多くのフレーズ対を必要とする．そのため，並び替えの候補が膨大になり，翻訳精度が低下する．

この問題を解決するために，長い日本語フレーズを持つフレーズ対を増やすことで，翻訳に利用するフレーズ対の数を減らす手法を提案する．具体的には，学習データの日本語文を文節に区切り，長い日本語フレーズを持つフレーズテーブルを学習する．本稿では，このフレーズテーブルを文節区切りフレーズテーブルと呼ぶ．そし

て，文節区切りフレーズテーブルを従来の単語区切りフレーズテーブルと併用し，翻訳を行う．文節区切りフレーズテーブルの生成手順を以下に示す．

1. 日本語文の文節区切り
学習データの日本語文を文節に区切り，文節区切りの学習データを生成する．

文節区切り日本語文

彼-の お母さん-が ああ 若い-と-は 思わ-なかっ-た 。
あした 返-すから 3-，-0-0-0-円 貸し-て-ください 。

2. フレーズテーブルの生成
文節区切りの学習データから，フレーズテーブルを生成する．

1 から生成されたフレーズテーブル

道路-の ||| of the road ||| 1 0.018 0.167 0.002
読ん-だ ||| have read ||| 1 0.013 1 0.030
2-0-人-の ||| 20 people ||| 1 0.002 0.5 0.003

3. 日本語フレーズの処理
従来手法のフレーズテーブルと区切りを統一するために，生成されたフレーズテーブルの日本語フレーズを単語に区切る．

文節区切りフレーズテーブル

道路 の ||| of the road ||| 1 0.018 0.167 0.002
読ん だ ||| have read ||| 1 0.013 1 0.030
2 0 人 の ||| 20 people ||| 1 0.002 0.5 0.003

4 実験環境

4.1 実験データ

実験には，辞書の例文から抽出した，単文コーパス 181,988 文 [4] と重文複文コーパス 121,719 文 [5] を用いる．単文コーパスから，Open テストデータ 9,000 文と development データ 1,000 文をランダムに抽出し，残りの 171,988 文を学習データに用いる．また，重文複文コーパスからも同様に，Open テストデータ 9,000 文と development データ 1,000 文をランダムに抽出し，残りの 111,719 文を学習データに用いる．単文コーパスと重文複文コーパス中の対訳文の例を表 2 に示す．

表 2: 単文コーパスと重文複文コーパスの例

単文コーパス	
日本語文	彼は有能な商人です。
英語文	He is an able merchant.
日本語文	花子は、悲しそうに俯いていた。
英語文	Hanako appeared sad and downcast.
重文複文コーパス	
日本語文	彼は偏見がありそのため信頼できなかった。
英語文	He was biased, and so unreliable.
日本語文	その鳥は山を越えて飛んでいった。
英語文	The bird winged its flight over the hills.

一般に，日英統計翻訳では，前処理として各コーパスの日本語文を形態素解析を用いて単語に区切る．本研究では，形態素解析器として “MeCab[6]” を用いる．また，単語区切りの学習データとは別に，文節区切りフレーズテーブルの学習のために，構文解析器 “CaboCha[7]” を用いて，文節区切りの学習データも生成する．また，英語文に対しては句読点の前後にスペースを入れる．一般に，英語文に対しては，大文字の小文字化を行うが，本研究では行わない．

4.2 フレーズテーブルの学習

フレーズテーブルの学習には，多くの方法がある．本研究では，“train-phrase-model.perl[9]” を用いる．このプログラムは IBM model1 ~ 5[1] に基づく，“GIZA++[8]” を利用している．

4.3 $N-$gram モデルの学習

言語モデルは，$N-$gram モデルを用いる．$N-$gram モデルの学習には “SRILM[10]” の ngram-count を用いる．なお，本研究ではスムージングに “-ndiscount” を用いる．

4.4 デコーダのパラメータ

デコーダは “Moses[11]” を使用する．本研究では，“Moses” が用いる言語モデルやフレーズテーブルの重みを最適化する．最適化は “Moses” 付属の “mert-moses.pl” を用いて行う．提案手法は，2 つのフレーズテーブルを併用して用いる．最適化により，2 つのフレーズテーブルには異なる重みが与えられる．また，単文の翻訳には単文の development データを，重文複文の翻訳には重文複文の development データを用いる．

4.5 評価方法

出力文の評価には自動評価法である BLEU[12] と METEOR[13] を使用する．

5 実験

本章では，各フレーズテーブルを用いたときの，単文と重文複文の翻訳実験の結果について述べる．翻訳実験は従来手法である単語区切りフレーズテーブルのみを用いた実験，文節区切りフレーズテーブルのみを用いた実験，そして提案手法である両フレーズテーブルを併用した実験の 3 つを行う．また，人手による評価として，対比較実験も行う．

5.1 フレーズテーブル

フレーズテーブルの学習には，4.1 節で示した，単文 171,988 文と重文複文 111,719 文，計 283,707 文を用いる．単語区切りフレーズテーブルと，文節区切りフレーズテーブルのフレーズ対の数を表 3 に示す．

表 3: 各フレーズテーブルのフレーズ対の数
(学習データ:283,707 文)

	フレーズ対の数
単語区切り (従来手法)	1,742,020
文節区切り	1,041,805

表 3 から，単語区切りフレーズテーブルと比較して，文節区切りフレーズテーブルのフレーズ対の数が，約 6 割であることがわかる．これは，文節区切りの学習データの文節数が，単語区切りの学習データの単語数と比較して，半分程度であることが原因である．

5.2 翻訳精度の評価

テストデータに単文と重文複文を用いて翻訳実験を行う．単文と重文複文の翻訳実験の評価結果を表 4 に示す．

表 4: 翻訳精度 (テストデータ:各 9,000 文)

単文		
	BLEU	METEOR
従来手法	0.2015	0.4437
文節区切り	0.1522	0.3118
提案手法	**0.2058**	**0.4455**
重文複文		
	BLEU	METEOR
従来手法	0.1746	0.4034
文節区切り	0.1425	0.3066
提案手法	**0.1784**	**0.4148**

表 4 から，提案手法の翻訳精度が従来手法の翻訳精度と比較して向上していることがわかる．また，文節区切りフレーズテーブルのみを用いた結果が従来手法と比較して大きく低下している．

5.3 対比較実験

表 4 の単文と重文複文の翻訳結果に対して人手による対比較実験を行う．

5.3.1 評価基準

対比較実験は従来手法の結果と提案手法の結果から，それぞれ 100 文を抽出し，どちらの文が入力文の翻訳結果として適切であるかを判断する．評価基準を以下に示す．

提案手法○　提案手法の結果が従来手法の結果より入力文の翻訳として優れている
提案手法×　提案手法の結果が従来手法の結果より入力文の翻訳として劣っている
差がない　どちらの翻訳結果も同程度に意味が理解できる．または，同程度に意味が理解できない
同一　翻訳結果が同一の文である

5.3.2 評価結果

対比較実験の評価結果を表 5 に示す．

表 5: 対比較実験の結果

	単文	重文複文
提案手法○	19/100	27/100
提案手法×	11/100	15/100
差がない	33/100	36/100
同一	37/100	22/100

表 5 から，提案手法の翻訳結果が従来手法の翻訳結果よりも優れていることがわかる．

6 学習データの英語文に対する提案手法の適用

5 章では，学習データの日本語文を文節区切りとし，フレーズテーブルの生成を行った．本章では，学習データの英語文に対して，同等の処理を行う．そして，日本語文の文節区切りと同様に，効果があるかを調べる．

6.1 英語文の処理

日本語文の文節区切りに相当する処理として，英語構文解析器 “Apple Pie Parser[14]” を用いて学習データの英語文の単語をフレーズ単位に結合する．結合の例を以下に示す．

表 6: フレーズ単位の統合の例

統合前	He is an able merchant .
統合後	He is an-able-merchant .
統合前	The bird winged its flight over the hills .
統合後	The-bird winged its-flight over the-hills .

英語文をフレーズ単位に統合した学習データからは，英語フレーズがフレーズ単位のフレーズテーブルが生成される．従来手法のフレーズテーブルは単語区切りであるため，同じ区切りにするために，英語フレーズを単語に区切る. そして，従来手法のフレーズテーブルと併用し，翻訳を行う．

また，実験には以下の 2 種類の学習データから生成されたフレーズテーブルを用いる．

1. 日本語文:単語区切り，英語文:フレーズ単位
   単語区切りの日本語文とフレーズ単位の英語文からフレーズテーブルを生成する．これは，一方の言語にのみ処理を行った場合の効果を調べるために用いる．
2. 日本語文:文節区切り，英語文:フレーズ単位
   文節区切りの日本語文とフレーズ単位の英語文からフレーズテーブルを生成する．これは，両言語に処理を行った場合の効果を調べるために用いる．

6.2 フレーズテーブル

フレーズテーブルの学習には，5 章と同様に，単文 171,988 文と重文複文 111,719 文を用いる．生成されたフレーズテーブルのフレーズ対の数を表 7 に示す．

表 7: 各フレーズテーブルのフレーズ対の 数
(学習データ:283,707 文)

	フレーズ対の数
日:単語，英:単語 (従来手法)	1,742,020
日:単語，英:フレーズ	1,147,845
日:文節，英:フレーズ	798,124

6.3 翻訳実験

テストデータに単文と重文複文を用いて翻訳実験を行う．単文と重文複文の翻訳実験の評価結果を表 8 に示す．

表 8: 翻訳精度 (テストデータ:各 9,000 文)

単文		
	BLEU	METEOR
従来手法	0.2015	0.4437
日:単語，英:フレーズ	**0.2063**	**0.4493**
日:文節，英:フレーズ	0.2045	0.4444
重文複文		
	BLEU	METEOR
従来手法	0.1746	0.4034
日:単語，英:フレーズ	0.1760	**0.4072**
日:文節，英:フレーズ	**0.1783**	0.4068

表 8 から，提案手法が英語文に対しても，有効であることがわかる．また，両言語に対して処理を行った場合，重文複文の翻訳精度は高いが，単文の翻訳精度はやや低い．

7 考察

7.1 全てのフレーズテーブルを用いた実験

5 章と 6 章では，従来手法である単語区切りフレーズテーブルと，提案手法により生成された 3 つのフレーズテーブルをそれぞれ併用し翻訳実験を行った．提案手法により生成されたフレーズテーブルは，それぞれ別のフレーズ対を含んでいるため，3 つ全てを用いることでさらに翻訳精度が向上する可能性がある. そこで，本章では，4 つのフレーズテーブルを併用した実験を行う．テストデータは単文と重文複文を用いる．単文と重文複文の翻訳実験の評価結果を表 9 に示す．

表 9: 翻訳精度 (テストデータ:各 9,000 文)

単文		
	BLEU	METEOR
従来手法	0.2015	0.4437
4 つを併用	0.2086	0.4473
重文複文		
	BLEU	METEOR
従来手法	0.1746	0.4034
4 つを併用	0.1797	0.4099

表 9 から，単文と重文複文共に，翻訳精度が向上していることがわかる．また，BLEU スコアは 5 章と 6 章の結果と比較して，最も高い値となった．

7.2 単文と重文複文の翻訳精度の傾向の違い

表 4 と表 8 において，単文の翻訳は併用したフレーズテーブルのフレーズ対の数が多いほど，翻訳精度が高くなる傾向があった．しかし．重文複文の翻訳は，フレーズ対の数とは無関係に，併用したフレーズテーブルの学習データの日本語文が文節区切りであった場合，翻訳精度が高い傾向があった．この原因について，次のように考えている．

日英統計翻訳は日本語をフレーズ対を用いて英語に変換し，並び替えにより英語文に翻訳する．単文は文法構造が単純であり，短文が多いため，翻訳に用いるフレーズ対が少なく，並び替えの候補は少ない．そのため，日本語フレーズの長さとは無関係に，フレーズ対の数に比例して翻訳精度が向上する傾向があった．

しかし，重文複文は文法構造が複雑であり，長文が多いため，翻訳に用いるフレーズ対が多く，並び替えの候補が膨大になる．日本語文を単語区切り，英語文をフレーズ区切りとした学習データから生成されたフレーズテーブルは日本語フレーズは短いため，翻訳に用いるフレーズ対を減らす効果が小さく，翻訳精度の向上は小さかった．しかし，日本語文を文節区切りとした学習データから生成されたフレーズテーブルは日本語フレーズが長いため，翻訳に用いるフレーズ対を減らす効果が大きい．そのため，翻訳精度の向上が大きかった．

このことから，単文の翻訳には単純にフレーズ対を増やすことが効果的であり，重文複文の翻訳には，長いフレーズ対を増やすことが効果的であると考えている．

8 おわりに

本研究では，学習データの日本語文を文節区切りとし，精度の高いフレーズテーブルを従来のフレーズテーブルと併用し，翻訳実験を行った．

実験の結果，提案手法の効果を確認できた．また，英語文に対しても，同等の処理を行い，他言語においても有効性を確認できた．

参考文献

[1] Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, Robert L. Mercer, “The Mathematics of Statistical Machine Translation, Parameter Estimation”, Computational Linguistics, 19(2), 1993.

[2] 猪澤雅史, 村上仁一, 徳久雅人, 池原悟, “統計翻訳における，単文と重文複文の翻訳精度の評価”, 情報処理学会研究報告, pp.79-84, 2008.

[3] Philipp Koehn, Franz J. Och, and Daniel Marcu, “Statistical phrase-based translation”, In Proceedings of HLT-NAACL 2003, pp. 127-133, 2003.

[4] 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.

[5] 村上仁一, 池原悟, 徳久雅人, “日本語英語の文対応の対訳データベースの作成”, 「言語，認識，表現」第 7 回年次研究会, 2002.

[6] MeCab, Yet Another Part-of-Speech and Morphological Analyzer, http://mecab.sourceforge.net/

[7] CaboCha, Yet Another Japanese Dependency Structure Analyzer, http://chasen.org/ taku/software/cabocha

[8] GIZA++, http://www.fjoch.com/GIZA++

[9] training-release-1.3.tgz, http://www.statmt.org/wmt06 /shared-task/baseline.html

[10] SRILM, The SRI Language Modeling Toolkit, http://www.speech.sri.com/projects/srilm

[11] Moses: moses.2007-05-29.tgz, http://www.statmt.org/moses/

[12] NIST Open Machine Translation, http://www.nist.gov/speech/tests/mt

[13] The METEOR Automatic Machine Translation Evaluation System, http://www.cs.cmu.edu/ alavie/METEOR/

[14] Apple Pie Parser, http://nlp.cs.nyu.edu/app/