句に基づく構文トランスファ方式統計翻訳

今村賢治[†,☆] 大熊英男[†,☆☆] 隅田英一郎[†,☆☆]

本稿では，構文トランスファ方式統計翻訳の 1 つの実現形態について提案する．従来，句に基づく統計翻訳では，句（連続した複数単語）を単位として，それらの確率的並び替えによって翻訳文を作成していた．一方，構文情報を利用し，階層的に単語を並べ替える統計翻訳も提案されてきているが，実際に機械翻訳を行ったときの特徴はいまだ不明である．本稿で提案する翻訳方式は，句単位の変換と 2 言語の構文木に基づいた階層的並べ替えを行い，統計翻訳の指標に基づいて最適な翻訳文を出力する．実験では，構文情報と句を併用して翻訳したとき，最も良い翻訳品質を示した．

Statistical Machine Translation Based on Syntactic Transfer with Phrases

Kenji Imamura,[†,☆] Hideo Okuma[†,☆☆] and Eiichiro Sumita[†,☆☆]

This paper presents a practical approach to statistical machine translation (SMT) based on syntactic transfer. Conventionally, phrase-based SMT generates an output sentence by combining phrase (multi-word sequence) translation and phrase reordering without syntax. On the other hand, SMT based on tree-to-tree mapping, which involves syntactic information, is theoretical, so its features remain unclear from the point of a practical system. The SMT proposed in this paper translates phrases with hierarchical reordering based on the bilingual parse tree. In our experiments, the best translation was obtaind when both phrases and syntactic information were used for the translation process.

1. はじめに

Brown ら[2] によって提案された統計翻訳も，当初の単語を単位とした翻訳モデルに変わり，句（連続した複数単語で，翻訳に適するように自動抽出した単語列．本稿では，句構造と明示的に区別する場合，長単位句と呼ぶ）を単位とした翻訳モデルが提案されてきている[12),23),26)]．これを句に基づく統計翻訳（Phrase-based SMT）と呼ぶ．句に基づく統計翻訳の場合，句の内部での語順は局所的に変更されているため，語順調整のコストは低くなる．しかし，句そのものの順序が大幅に異なる言語間では，句の順序を調整しなければ適切な翻訳文とはならない．従来は，句の順序調整を平坦な構造上で確率的に行っていた．

一方，構文情報を利用し，木構造どうしのマッピングをとることにより翻訳を行う統計翻訳も提案されてきている[8),13)]．これらは階層的に単語や句を並べ替えることが可能であるが，理論にとどまっており，実際に翻訳したときの特徴は明らかになっていない．

本稿では，木構造マッピング型統計翻訳の 1 つとして，実際に動作しうる構文トランスファ方式統計翻訳を提案する．構文トランスファ方式は，古くから用いられている翻訳方式であり，構文構造が異なる言語（たとえば，SVO 形式の言語と SOV 形式の言語）間の翻訳に適していると考えられる．本稿で提案する方式は，階層的に語順を調整するだけでなく，句に基づく統計翻訳で用いられているような長単位句を単位とすることができる．

以下，2 章では，構文トランスファ方式の概要を説明し，3 章では，同方式に統計モデルを導入する．4 章，5 章ではそれぞれ，訓練法，デコード法について説明し，6 章では，実験を通じて本方式の議論を行い，7 章では関連研究を紹介する．

† ATR 音声言語コミュニケーション研究所
ATR Spoken Language Communication Research Laboratories
☆ 現在，日本電信電話株式会社 NTT サイバースペース研究所
Presently with NTT Cyber Space Laboratories, NTT Corporation
☆☆ 現在，独立行政法人情報通信研究機構および ATR 音声言語コミュニケーション研究所
Presently with National Institute of Information and Communications Technology and ATR Spoken Language Communication Research Laboratories

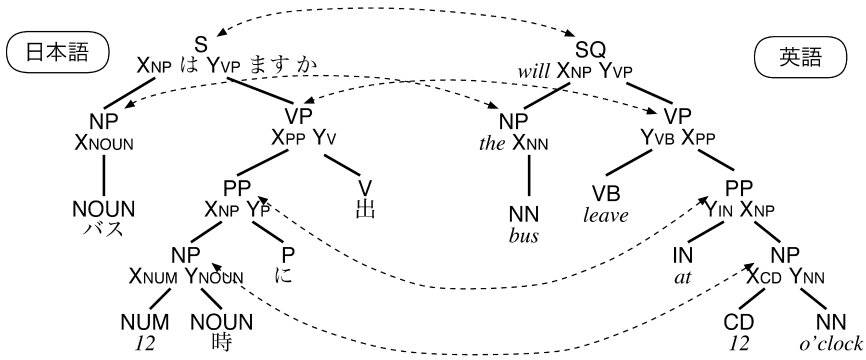

図 1 構文トランスファによる翻訳過程の例（日英翻訳）
Fig. 1 Example of translation process (Japanese-to-English) by Syntactic Transfer.

原言語木構造テーブル

ルール名	親	確率	子供
θ_1	S	$\overset{2.8e-1}{\Rightarrow}$	X_{NP} は Y_{VP} ます か
θ_2	VP	$\overset{3.8e-1}{\Rightarrow}$	X_{PP} Y_V
θ_3	VP	$\overset{4.8e-2}{\Rightarrow}$	X_{NP} に Y_V
θ_4	NP	$\overset{5.3e-1}{\Rightarrow}$	X_{NN}
θ_5	NP	$\overset{8.3e-3}{\Rightarrow}$	X_{CD} Y_{NN}
		:	

木構造マッピングテーブル

原言語	確率 逆方向	確率 順方向	目的言語
θ_1	$\overset{2.0e-1}{\Leftarrow}$	$\overset{1.2e-1}{\Rightarrow}$	π_1
θ_2	$\overset{1.2e-1}{\Leftarrow}$	$\overset{1.9e-1}{\Rightarrow}$	π_2
θ_2	$\overset{4.3e-3}{\Leftarrow}$	$\overset{3.2e-2}{\Rightarrow}$	π_3
:	:		:

目的言語木構造テーブル

ルール名	親	確率	子供
π_1	SQ	$\overset{3.2e-1}{\Rightarrow}$	*will* X_{NP} Y_{VP}
π_2	VP	$\overset{4.3e-2}{\Rightarrow}$	Y_{VP} X_{NP}
π_3	VP	$\overset{1.0e-2}{\Rightarrow}$	Y_{VP} X_{PP}
π_4	NP	$\overset{1.2e-2}{\Rightarrow}$	*the* X_{NN}
π_5	NP	$\overset{8.4e-4}{\Rightarrow}$	X_{CD} Y_{NN}
		:	

図 2 構文トランスファ方式における翻訳モデルの例
Fig. 2 Example of tables for Syntactic-transfer-based Translation Model.

2. 構文トランスファ方式の概要

2.1 構文トランスファ方式機械翻訳

構文トランスファ方式機械翻訳は，入力文を構文解析し，その構文木を出力の構文木にマッピングすることにより翻訳文を生成する．

実際には，構文木全体を直接マッピングすることはできないため，構文木を文脈自由文法（CFG）規則の集合に分解し，ノード間でマッピングを行う．**図 1** は，構文トランスファで日本語「バスは 12 時に出ますか」を英語 “*will the bus leave 12 o'clock*” に翻訳する例である．同図において，点線矢印は，対応する構文ノードを表す☆．各ノードの上段（S，NP 等）は CFG 規則の左辺に相当し，部分木の構文ラベルまたは単語の品詞である．下段は右辺に相当し，X，Y は原言語・目的言語間で対応する子ノードを示すためのラベルである．

このような処理を行うためには，3 種類のルールセットが必要である（**図 2**）．

- 原言語の構文木を構成するための CFG 規則（原言語木構造テーブル）
- 目的言語の構文木を構成するための CFG 規則（目的言語木構造テーブル）
- 原言語・目的言語の CFG 規則どうしの対応（木構造マッピングテーブル）

入力文は原言語木構造テーブルを参照し，構文解析される．次に，木構造マッピングテーブルを参照し，原言語の構文木に使われた CFG 規則を目的言語の CFG 規則にマッピングする．そして，目的言語木構造テーブルを参照して目的言語の構文木を構築し，翻訳文を生成する．

両言語間の構文ノードは 1 対 1 に対応しており，すべての部分木が部分翻訳として成り立っている．このため，構文トランスファ方式で使用される構文木は，単言語の構文解析の出力とは若干異なる．本稿では，あらかじめ訓練時に構文ノードを対応付け，対応しない単言語のノードを削除することにより，1 対 1 対応を生成する．

語順変更は，変換時の子ノードの順序として表される．子の部分木は複数単語から成り立っているので，構文トランスファ方式は，句を単位とした語順変更を階層的に行うことができる．

2.2 長単位句

構文トランスファ方式においては，長単位句も構文木の 1 つと見なされ，翻訳時には曖昧性の 1 つとして扱われる．すなわち，原言語および目的言語の構文

☆ 終端記号の対応は省略してある．

木として，長単位句を使ったものと，文脈自由文法規則により，細かい単位（たとえば単語）をつなげたものの両方が作成され，個々の構文木に対応する訳文が別個に作られる．最終的に統計翻訳のスコアリングに従い，最適な訳文と構造が選択される．ただし，このような処理を行うためには，長単位句に対しても構文ラベルをあらかじめ付与しておく必要がある点が，一般的な句に基づく統計翻訳における長単位句と異なる（4，5 章参照）．

3. 構文トランスファ方式統計翻訳

本章では，2 章で述べた方式を，統計翻訳の枠組みで説明する．まず，ソースチャネルモデルに基づくモデルを説明し，それを Log-linear モデルに拡張する．

3.1 翻訳モデル

統計翻訳は，入力の単語列 $\mathbf{f}$ が与えられたとき，確率を最大化する出力の単語列 $\mathbf{e}$ を，すべての可能な組合せ中から探索することにより，翻訳を行う方式である．ソースチャネルモデルを適用する場合，通常，探索には以下の式が用いられる．

$$\begin{aligned}\hat{\mathbf{e}} &= \operatorname*{argmax}_{\mathbf{e}} P(\mathbf{e}|\mathbf{f}) \\ &= \operatorname*{argmax}_{\mathbf{e}} P(\mathbf{e})P(\mathbf{f}|\mathbf{e}). \end{aligned} \tag{1}$$

$P(\mathbf{e})$ は言語モデル確率，$P(\mathbf{f}|\mathbf{e})$ は翻訳モデル確率と呼ばれる．構文トランスファ方式の統計翻訳は，翻訳モデル中に隠れ変数として原言語，目的言語の構文木（それぞれ $\mathcal{F}$，$\mathcal{E}$ と示し，単語列 $\mathbf{f}$，$\mathbf{e}$ を生成する）を仮定し，木構造どうしのマッピングを行うことにより，翻訳文を生成する．

$$\begin{aligned}P(\mathbf{f}|\mathbf{e}) &= \sum_{\mathcal{F},\mathcal{E}} P(\mathbf{f},\mathcal{F},\mathcal{E}|\mathbf{e}) \\ &\approx \sum_{\mathcal{F},\mathcal{E}} P(\mathbf{f}|\mathcal{F})P(\mathcal{F}|\mathcal{E})P(\mathcal{E}|\mathbf{e}). \end{aligned} \tag{2}$$

本稿では，$P(\mathcal{E}|\mathbf{e})$ を目的言語の木構造確率，$P(\mathcal{F}|\mathcal{E})$ を目的言語→原言語（逆方向）木構造マッピング確率と呼ぶ．各モデルは独立と仮定している．確率 $P(\mathbf{f}|\mathcal{F})$ は，$\mathcal{F}$ の定義から 1.0 となるため，式 (2) は以下のとおり書き直される．

$$P(\mathbf{f}|\mathbf{e}) \approx \sum_{\mathcal{F},\mathcal{E}} P(\mathcal{F}|\mathcal{E})P(\mathcal{E}|\mathbf{e}). \tag{3}$$

3.2 素性関数

Log-linear モデル[16] は，ソースチャネルモデルを含み，より拡張性の高いモデルである．これを用いる場合，条件付き確率 $P(\mathbf{e}|\mathbf{f})$ は以下の式で表される．

$$P(\mathbf{e}|\mathbf{f}) = \frac{1}{Z(\mathbf{f})} \exp\left(\sum_{m=1}^{M} \lambda_m h_m(\mathbf{e},\mathbf{f})\right). \tag{4}$$

ここで，$Z(\mathbf{f})$ は正規化用の定数である．また，$h_m(\mathbf{e},\mathbf{f})$ は，原言語，目的言語単語列に依存した特徴量を出力する素性関数（feature function）で，λ_m は，素性関数の重みである．M は素性関数の数である．したがって，式 (4) を最大化する目的言語の単語列 $\hat{\mathbf{e}}$ は，式 (5) で表される．

$$\begin{aligned}\hat{\mathbf{e}} &= \operatorname*{argmax}_{\mathbf{e}} P(\mathbf{e}|\mathbf{f}) \\ &= \operatorname*{argmax}_{\mathbf{e}} \left\{\sum_{m=1}^{M} \lambda_m h_m(\mathbf{e},\mathbf{f})\right\}. \end{aligned} \tag{5}$$

素性関数の重み付き和（$\sum_{m=1}^{M} \lambda_m h_m(\mathbf{e},\mathbf{f})$）を，便宜上統計翻訳スコアと呼ぶ．

素性関数は，通常，モデルの確率値の対数が用いられる．本稿では，式 (3) を原言語から目的言語，目的言語から原言語へ双方向に適用し，以下の 5 関数を用いる．

- 原言語木構造確率☆
  $h_1(\mathbf{e},\mathbf{f}) = \log P(\mathcal{F}|\mathbf{f}) \approx \log P(\mathcal{F})$.
- 目的言語木構造確率
  $h_2(\mathbf{e},\mathbf{f}) = \log P(\mathcal{E}|\mathbf{e}) \approx \log P(\mathcal{E})$.
- 木構造マッピング確率
  $h_3(\mathbf{e},\mathbf{f}) = \log P(\mathcal{F}|\mathcal{E})$,
  $h_4(\mathbf{e},\mathbf{f}) = \log P(\mathcal{E}|\mathcal{F})$.
- 言語モデル確率．本稿では，単語 n-gram を用いる．
  $h_5(\mathbf{e},\mathbf{f}) = \log P(\mathbf{e})$.

翻訳モデルを双方向に適用することは，逆翻訳（一度原言語から目的言語に翻訳した文を，再度原言語に翻訳すること．たとえば文献 25) 参照）を行うことと同等である．逆翻訳は，従来の機械翻訳において，翻訳結果が誤っている場合には原文に戻らない性質を利用し，翻訳結果の検証に用いられてきた．本稿の素性関数はこの要素を採り入れたものである．

3.3 内側確率

原言語・目的言語木構造モデルは，確率文脈自由文法（PCFG）と見なすことができる．すなわち，木の各ノードが独立に生成されると仮定し，親ノードが子ノード列を生成する確率の総積（内側確率．対数確率の場合，総和）で木構造確率を表す．

$$\log P(\mathcal{F}) = \sum_{\theta:\theta\in\mathcal{F}} \log P(\theta), \tag{6}$$

☆ きわめて荒い近似であるが，0 次近似を行い，木構造モデルでは単語列生成確率を無視した．

$$\log P(\mathcal{E}) = \sum_{\pi:\pi\in\mathcal{E}} \log P(\pi). \quad (7)$$

ここで，θ，π は，それぞれ構文木 $\mathcal{F}$，$\mathcal{E}$ を構成する文脈自由文法規則で，$P(\theta)$，$P(\pi)$ は，親ノード N_i が直下の子ノード列 $N_{i+1}\ldots N_{i+K}$ を生成する確率である．

木構造マッピング確率も同様に，内側確率として以下の式で定義する．2.1 節で述べた図 2 は，この木構造マッピングモデルと，原言語・目的言語木構造モデルの例である．

$$\log P(\mathcal{E}|\mathcal{F}) = \sum_{(\theta,\pi)\in\left\{(\theta_i,\pi_i)\,\middle|\,\substack{i=1,\ldots,n,\\ \theta_i\in\mathcal{F},\pi_i\in\mathcal{E}}\right\}} \log P(\pi|\theta), \quad (8)$$

$$\log P(\mathcal{F}|\mathcal{E}) = \sum_{(\theta,\pi)\in\left\{(\theta_i,\pi_i)\,\middle|\,\substack{i=1,\ldots,n,\\ \theta_i\in\mathcal{F},\pi_i\in\mathcal{E}}\right\}} \log P(\theta|\pi). \quad (9)$$

ただし，(θ_i, π_i) は，原言語，目的言語の構文木に含まれる CFG 規則のうち，対応する組を表す．n は構文木の非終端ノード数である．

内側確率として定義した場合，翻訳モデルの素性関数値は，子供の部分木の値から再帰的に計算される．すなわち，子供の部分木の値の総和に，最上位の CFG 規則の原言語・目的言語木構造対数確率と，木構造マッピングの対数確率（双方向）を重み付き加算したもので計算可能である．このように，本稿で用いる翻訳モデルは，ボトムアップパーザを 2 言語に拡張することにより自然に計算可能なモデルとなっている．

4. 訓　　練

問題を簡単にするため，訓練時には，構文木は既存の構文解析器を用いてあらかじめ生成済みとする．したがって，訓練フェーズでは，(1) 構文木のノード間の対応を抽出すること（句アライメント）と，(2) 原言語・目的言語の木構造モデル，木構造マッピングモデルのパラメータを推定することが課題となる．

4.1 句アライメント

本稿で用いる句アライメントは，Alignment Template[18] の獲得とほぼ同様なアプローチをとる．ただし，最も異なる点は，Alignment Template が単語アライメントの連続性だけを基に句を抽出するのに対して，本稿では，文献 9) と同様に構文木を制約として用い，原言語，目的言語ともに部分木として成り立ちうる句のみを対応づける点にある．

句アライメントの例を**図 3** に示す．本処理は，以下の手順で行われる．

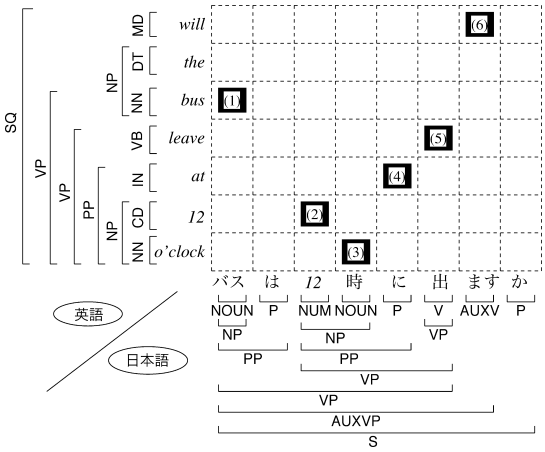

図 3 句アライメントの例
Fig. 3 Example of phrase alignment.

(1) まず，訓練コーパスを，単語アライメントツールとして広く利用されている GIZA++[17] を用い，原言語→目的言語，目的言語→原言語の単語アライメントを行う．そして，ビタビアライメントを抽出する．

(2) 次に，双方向のビタビアライメントが一致した単語対応を，信頼アライメント（sure alignment）とする．図 3 の■は信頼アライメントを表す．

(3) 信頼アライメントの組合せに対し，そのアライメントだけを含み，他の信頼アライメントを含まない構文ノードを句アライメントとして抽出する．ただし，構文ノードの対応に曖昧性がある場合は，構文ラベルの共起頻度が最も高いノードから，最良優先に対応づける．

たとえば，図 3 の例において，信頼アライメント (2)(3) に着目した場合，これだけを含み，他の信頼アライメント（つまり (1)(4)(5)(6)）を含まない構文ノードとしては，NP → *12 o'clock*，NP → *12* 時があるので，これは句アライメントとして抽出される．しかし，信頼アライメント (4)(5) に着目した場合，これだけを含み，他の信頼アライメントを含まない構文ノードはないため，“*leave at*” と「に出」は句アライメント結果には含まれない．図 3 から抽出される句の一覧を**表 1** に示す．

このように抽出された句アライメントは，句に基づく統計翻訳における長単位句と見なされる．構文ラベルが付いているため，そのまま構文解析器に適用することが可能である．

また，句アライメント結果は階層性を持っているた

表 1 図 3 から抽出された句アライメント一覧
Table 1 Extracted phrases from Fig. 3.

番号	日本語			英語		
1	NOUN	→	バス	NN	→	*bus*
2	NP	→	バス	NP	→	*the bus*
3	NUM	→	*12*	CD	→	*12*
4	NOUN	→	時	NN	→	*o'clock*
5	NP	→	*12* 時	NP	→	*12 o'clock*
6	P	→	に	IN	→	*at*
7	PP	→	*12* 時 に	PP	→	*at 12 o'clock*
8	V	→	出	VB	→	*leave*
9	VP	→	*12* 時 に 出	VP	→	*leave at 12 o'clock*
10	S	→	バス は *12* 時 に 出 ます か	SQ	→	*will the bus leave at 12 o'clock*

め，これを利用して原言語・目的言語の文脈自由規則が作成される．たとえば，表 1 のアライメント 9 は，8 と 7 のアライメントを直下に持っている．そこで，直下のアライメントを非終端記号とすると，日本語の規則 VP → PP V と英語の規則 VP → VB PP が得られる．

4.2 パラメータ推定

本稿では，モデルのパラメータをすべて相対頻度から算出する．たとえば，原言語の木構造モデルのパラメータは，以下の式で与える．

$$P(\theta_i) = P((N_{i+1} \dots N_{i+K})|N_i) = \frac{count((N_{i+1} \dots N_{i+K}), N_i)}{count(N_i)}. \quad (10)$$

ただし，$count(N_i)$ は，構文ノード N_i がコーパス中に出現する頻度で，$count((N_{i+1} \dots N_{i+K}), N_i)$ は，親ノード N_i が直下に子ノード列 $(N_{i+1} \dots N_{i+K})$ を持つ事象の出現頻度をコーパス上でカウントしたものである．目的言語の木構造モデルのパラメータも同様に算出する．

木構造マッピングモデルの順方向パラメータは以下の式で与える．逆方向も同様である．

$$P(\pi_i|\theta_i) = \frac{count(\pi_i, \theta_i)}{count(\theta_i)}. \quad (11)$$

本稿の方式は，単語と句を区別せずに扱っているため，単語の翻訳確率も相対頻度を基に算出している．なお，スムージングは行っていない．

5. デコーディング/翻訳処理

構文トランスファ方式機械翻訳は，基本的には文脈自由文法を用いたパーザに，変換部，生成部を付加して実現できる．本稿では，ボトムアップチャートパーザを基にする．デコードの手順は以下のとおりである．

(1) まず，原言語の木構造モデルを用いて，入力文をボトムアップに構文解析する．
(2) 入力の部分木ができた時点で，木構造マッピングモデルおよび目的言語の木構造モデルを参照し，出力の部分木構造を作成する．
(3) 出力の部分木を展開し，部分単語列を作成する．単語列の統計翻訳スコアは，3.3 節で述べた内側確率と，言語モデル確率によって与えられる．
(4) 入力の部分木のカバー範囲と入力の部分木の構文ラベルが同じものについて，出力単語列リスト（出力側の構文ラベルを含む）をマージする．そして枝刈りを行い，統計翻訳スコアの上位 N 個のみを，その部分木の翻訳結果とする．
(5) 以上ステップ (1)〜(4) を，入力文全体が構文解析完了するまで繰り返す．

図 4 に，日本語の部分文字列「*12* 時に出」の翻訳例を示す．入力の構文解析結果に曖昧性があるとき，または変換候補が複数存在するときは，個別に変換を行い，ステップ (4) で出力の単語列リストをマージする．このような処理を行うことにより，出力の単語列だけでなく，入出力の構文ラベルも得ることができ，さらに上位の構造を解析・変換することが可能となる．

長単位句の翻訳は，構文解析の曖昧性の 1 つとして扱われる．つまり，終端記号のみから成る規則を各モデルに追加しておくと，上記処理により，単語・句を区別せずに翻訳される．

なお，入力文または出力文の構文解析に失敗した場合は，翻訳器の内部に保持されている部分木の列を取り出し，最も少ない部分木で構成されて，かつ統計翻訳スコアの総和が最大となる単語列を出力する．

6. 評価実験

本稿では，SOV 形式と SVO 形式の言語対である，日英翻訳を対象に評価を行う．

6.1 実験条件

6.1.1 コーパス

本稿では対訳コーパスとして，BTEC (Basic Travel Expression Corpus)[10),22)] を用いる．これは，旅行会

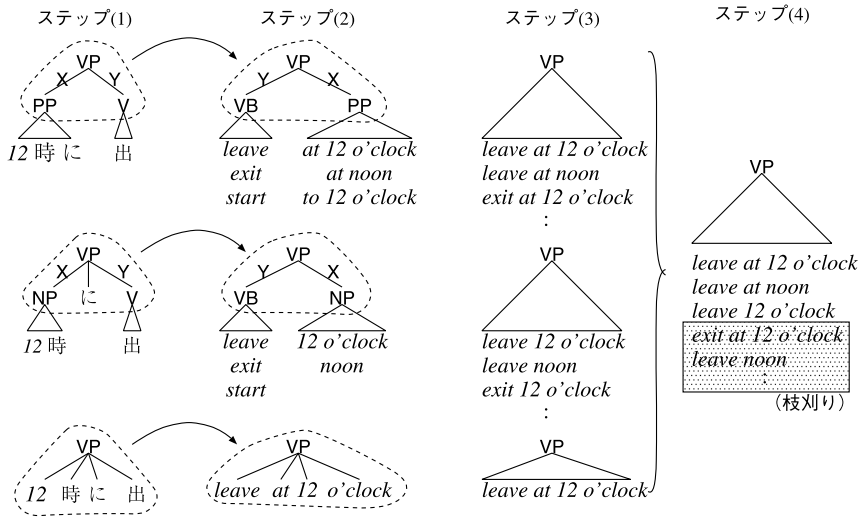

図 4 デコーディング処理例
Fig. 4 Example of decoding.

表 2 コーパスサイズ
Table 2 Corpus size.

セット名	項目	日本語	英語
BTEC/FULL	文数	152,170	
（訓練）	延べ単語数	1,178,682	1,103,655
	異なり単語数	16,626	10,667
BTEC/IWSLT	文数	20,000	
（訓練）	延べ単語数	189,729	182,224
	異なり単語数	8,739	6,086
開発	文数	500	500 ∗ 16
	延べ単語数	4,019	—
	異なり単語数	888	—
テスト	文数	506	—
	延べ単語数	4,066	—
	異なり単語数	914	—

話に頻出する表現を集めた基本表現集である．コーパスサイズを**表 2** に示す．表中の BTEC/FULL は BTEC 全体，BTEC/IWSLT とは，評価型ワークショップ IWSLT[1),7)] で用いられたコーパスで，BTEC のサブセットである．開発セットは同ワークショップの 2004 年，テストセットは 2005 年のテストセットを使用した．なお，表中の異なり単語数は，単語の品詞と原形の種類数である☆．

6.1.2 訓　　練

翻訳モデルを構築するための単語アライメントとして，GIZA++[17)] の IBM モデル 4 のビタビアライメントを使用した．また，英語は Charniak の解析器[3)] を用い，日本語は，独自に開発した句構造解析器を用いて構文解析を行い，訓練した．

言語モデルは SRILM（SRI Language Modeling Toolkit）[20)] を用いて，訓練セットの英語側から BTEC/FULL，BTEC/IWSLT それぞれの単語 trigram を作成して使用した．

各素性関数の重みは，誤り率最小訓練法[15)] を用いて，開発セットの自動評価結果が最良になるよう，設定した．

6.1.3 評 価 基 準

自動評価法として，BLEU[19)]，NIST[6)]，mWER（multiple Word Error Rate）[14)] を用いた．使用した参照訳は 1 原文あたり 16 である．各評価基準において，文献 11) による bootstrap resampling 法を用い，スコアの平均と 95%の信頼区間を算出した．

また，主観評価として，A（完全訳），B（部分訳），C（理解可能訳），D（不可訳）の 4 段階評価を用いた[21)]．なお，mWER のみ，低いスコアが良質な翻訳を意味する．

6.2 実 験 結 果

6.2.1 翻 訳 品 質

まず，提案方法による翻訳品質を測定した．結果を**表 3** および**表 4** に示す．構文情報，長単位句の効果を独立に測定するため，以下の 2 方式でも実験を行った．

- 翻訳モデルから，長単位句を削除した場合（長単位句なし）．最も短い終端記号列から成り立つ規則だけでデコードした．
- 構文情報なしでデコードを行った場合．ここでは，南カリフォルニア大学が開発した句に基づくビー

☆ 本実験では，内部開発の形態素解析による分かち書きや，数詞列を 1 単語として扱う（たとえば，“fifty/CD one/CD” を ‘51/CD’ に置き換える）等の前処理を行っているため，IWSLT で公開されたコーパスの単語数とは若干異なる．

表 3 自動評価による翻訳品質（日英翻訳）
Table 3 Translation quality (Japanese-to-English) by automatic evaluation.

訓練セット	方式	mWER［信頼区間］	BLEU［信頼区間］	NIST［信頼区間］
BTEC/FULL	提案方式（長単位句あり）	**0.311** [0.287, 0.338]	**0.615** [0.572, 0.659]	8.42 [7.89, 8.92]
	提案方式（長単位句なし）	0.384 [0.359, 0.412]	0.530 [0.485, 0.569]	7.33 [6.79, 7.85]
	Pharaoh（構文なし）	0.391 [0.362, 0.422]	0.475 [0.437, 0.513]	**8.93** [8.55, 9.30]
BTEC/IWSLT	提案方式（長単位句あり）	0.464 [0.439, 0.490]	**0.422** [0.387, 0.460]	6.71 [6.25, 7.21]
	提案方式（長単位句なし）	**0.467** [0.439, 0.493]	0.416 [0.381, 0.451]	**7.09** [6.62, 7.56]
	Pharaoh（構文なし）	0.554 [0.526, 0.583]	0.278 [0.249, 0.306]	6.91 [6.63, 7.20]

表 4 主観評価による翻訳品質（日英翻訳）
Table 4 Translation quality (Japanese-to-English) by subjective evaluation.

訓練セット	方式	主観評価 A	主観評価 A+B	主観評価 A+B+C
BTEC/FULL	提案方式（長単位句あり）	**57.1%**	**67.6%**	**74.3%**
	提案方式（長単位句なし）	42.3%	54.2%	66.0%
	Pharaoh（構文なし）	47.4%	58.1%	70.8%
BTEC/IWSLT	提案方式（長単位句あり）	**29.8%**	**43.3%**	**55.7%**
	提案方式（長単位句なし）	28.5%	**43.3%**	**55.7%**
	Pharaoh（構文なし）	22.3%	27.5%	45.7%

ム探索デコーダ，Pharaoh [12)]☆を使用した．なお，長単位句については，提案方法と同じものを使用した．

まず，BTEC/FULL に着目すると，BLEU および mWER による自動評価では，提案方式（長単位句あり）は，長単位句なし，Pharaoh に比べて信頼区間の重なりがなく，良質な翻訳結果を示している．主観評価においても，提案方式（長単位句あり）が最も翻訳品質が良く，同様な結果となっている．NIST スコアでは Pharaoh の品質が良いが，提案方式（長単位句あり）と信頼区間が重なっており，有意な差とはなっていない．

一方，BTEC/IWSLT に着目すると，BLEU および mWER による自動評価では，提案方式の長単位句あり，なしのスコアはほぼ同じで，Pharaoh のみ，翻訳品質が劣っているという結果となった．主観評価においてもほぼ同様の傾向を示している．NIST スコアに関しては，信頼区間がどの方式も重なっており，有意な差は認められなかった．NIST スコアは，単語の情報量によって重みづけがされた評価方法であるため，resampling 法のような，テストセットが変化する（いい換えると，単語の情報量が変化する）ような検定方法では信頼区間が広くとられたためと考えられる．

NIST スコアを除いて考えると，コーパスサイズが小さい場合（BTEC/IWSLT），長単位句の量も少なくなるため，長単位句あり/なしの品質差は少なくなるが，コーパスサイズが大きくなると（BTEC/FULL），長単位句で翻訳可能な句が増加するため，長単位句を使用する提案方式と Pharaoh の翻訳品質が向上する．つまり，長単位句も構文情報も翻訳品質向上のためにはどちらも有効であり，本提案方式のように両者を併用すべきである．

6.2.2 構文構造生成失敗

本提案方式は，構文解析を用いるため，下位の構造で入力文の構文解析に失敗した場合や，出力の構文構造生成に失敗した場合，それより上位の構造を作成することができない．つまり，獲得されたモデルがデータスパースネス等により十分なカバレッジを持っていない場合，構文構造生成が失敗する．本実験では，BTEC/FULL コーパスで 46 文（9.1%），BTEC/IWSLT で 93 文（18.4%）が構文構造生成に失敗し，部分翻訳結果の組合せで翻訳された．

提案方式（BTEC/FULL）による，構文構造生成成功/失敗別の主観品質を**図 5** に示す☆☆．明らかに，構文構造生成成功時の翻訳品質が良く，逆にいうと，失敗した場合，ほとんど良好な翻訳結果が得られていない．

本方式は単言語の構文解析に比べ，（1）原言語・目的言語の双方が構文木として成り立たなければならない．（2）翻訳として等価な規則しか構文解析/構造生成に利用できないため，どうしても失敗が多くなる．さらに翻訳品質を向上させるためには，この構文構造生成失敗を抑えることが必要である．

☆ http://www.isi.edu/licensed-sw/pharaoh/

☆☆ 自動評価による品質測定は，テストセットが固定でなければならないため，ここでは主観評価結果のみを示す．

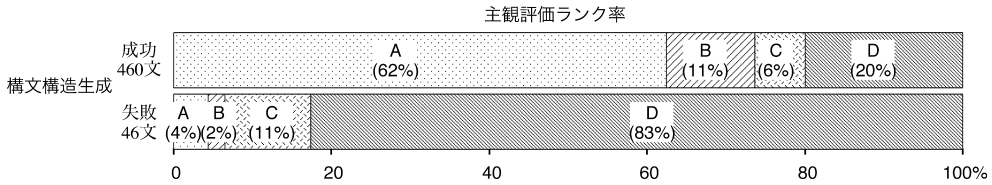

図 5 BTEC/FULL コーパスにおける構文構造生成成功/失敗別主観品質（構文トランスファ長単位句あり）
Fig. 5 Subjective evaluation according to parsing success/failure with BTEC/FULL corpus (in proposed method with phrases).

番号	原文　翻訳結果（方式，ランク，翻訳文）		
1	旅行者小切手で支払いできますか.		
	長単位句あり	A	Do you accept traveler's checks?
	長単位句なし	A	Do you accept traveler's checks?
	Pharaoh	D	Shall we'll pay by traveler's checks?
2	電気がつかないんです.		
	長単位句あり	A	The light doesn't work.
	長単位句なし	A	The light doesn't work.
	Pharaoh	C	Has the light doesn't work.
3	このキャビンで夕食を食べたいのですが.		
	長単位句あり	D	The dinners in the cabin.
	長単位句なし	B	I'd like to have dinner with me in the cabin.
	Pharaoh	B	But I want to eat dinner on this cabin.
4	釣りをしたいのですが.		
	長単位句あり	A	I'd like to go fishing.
	長単位句なし	D	Small change please.
	Pharaoh	A	I'd like to go fishing.
5	明日忘れずに私に電話してください.		
	長単位句あり	A	Please remember to call me tomorrow.
	長単位句なし	D	Please forget to call me tomorrow.
	Pharaoh	A	Please remember to call me tomorrow.

図 6 BTEC/FULL における各方式別翻訳例
Fig. 6 Translation examples according to the systems trained by BTEC/FULL set.

6.2.3 翻 訳 例

誤訳の原因は，構文構造生成失敗のほかにも，アライメントエラー，デコード時の句の選択ミス，句や単語の組み立てエラー，不完全なモデル，コーパス不足等，多岐にわたり，それらが複合して発生するため，定量的分析が非常に困難である．

そのため本項では，提案方式（長単位句あり）とその他の方式を比較し，主観評価の差が 2 段階以上ある文について，その特徴的なエラーを，翻訳例とともに例示する（**図 6**）．なお，本例はすべて BTEC/FULL セットを用いたものである．

6.2.3.1 Pharaoh との比較

提案方式（長単位句あり）と Pharaoh との差異は，構文情報を使用しているか否かである．両者で，主観評価が 2 段階以上の差がついた文は，提案方式が優ったもの 58 文，Pharaoh が優ったもの 20 文であった．

図 6 の番号 1 および 2 は，提案方式が優った例である．Pharaoh による翻訳は，どちらも 1 文中に助動詞が 2 語存在し，文法的には明らかに誤りである．このように，Pharaoh の誤訳には，文法的なエラーが多く，より強力な言語モデルによるチェックが必要であることを示している．

逆に，番号 3 は，Pharaoh が優った例である．提案方式（長単位句あり）は，名詞句に訳されている．本稿の方式は，慣用表現等も翻訳できる柔軟さを持つが，訓練に使用した対訳文中に日本語文が英語名詞句に翻訳されているものが含まれるため，このようなエラーも発生する．Pharaoh の場合，長単位句として一致しない限り，文が名詞句に翻訳されることは少ない．

6.2.3.2 提案方式（長単位句なし）との比較

両者の差異は，長単位句を使用しているか否かである．両者で主観評価が 2 段階以上の差がついた文は，長単位句ありが優ったもの 78 文，なしが優ったもの 13 文であった．

図 6 の番号 4 および 5 は，長単位句ありが優った例である．長単位句なしは，単語自体を誤訳しているため，評価が低いが，どちらも文法的には誤っていない．長単位句を導入することにより，より広い文脈を

参照した翻訳ができるため，単語の誤訳は減少する．

7. 関連研究

構文情報を用いる統計翻訳には，以下のものが提案されている．

文献 24) は，原言語（入力）の単語列を目的言語（出力）の木構造に変換する翻訳モデルを提案している．これは目的言語側のみ構文木を用いている．文献 4) は，上記翻訳モデルに加えて，目的言語の構文木のもっともらしさを語彙化確率文脈自由文法（LPCFG）で検証する方式を提案している．つまり，言語モデルとして n-gram ではなく，LPCFG を用いた方式である．目的言語の構文情報を用いることにより，翻訳文の流暢さが向上することが確認されている．

上記方式と比較すると，本稿の方式は原言語・目的言語ともに構文木を使用している点が大きく異なる．本稿では，素性関数として原言語側の構文木の生成確率も含んでいるため，原言語・目的言語双方の構文木が正しく解析・生成されると期待できる．一方目的言語に限ると，本稿の方式は語彙化されていない PCFG を使用しているため，文献 4) に比べ検証力は弱い．そのため，標準的な n-gram も使用して PCFG を補完している．

両言語の構文情報を用いるものとしては，文献 8) および文献 13) が提案を行っているが，実際に翻訳したときの特徴については述べられていない．本稿の提案方式は，これらの実現形態の 1 つと位置づけられる．

木構造を変換するタイプの統計翻訳としては，文献 5) がある．これは，同期文脈自由文法形式の規則を集めたモデルを対訳コーパスより学習し，それに基づいて翻訳を行うという点で，提案方式と類似している．しかし，訓練時に文法を用いず，対訳文と単語アライメント情報のみから同期文脈自由文法形式の規則を獲得しているため，すべての規則が同じ構文ラベルしか持たない☆．そのため，文献 5) で生成される木構造は，文法的意味づけが困難である．本稿の提案方式は，文法的に意味がある木構造を生成しようと試みている．

一方，構文情報をまったく用いない句に基づく統計翻訳としては，文献 12)，23)，26) 等が提案している．文献 12) は，長単位句の獲得の際，構文情報を用いて制約すると，翻訳性能が劣化すると報告している．しかし，本方式のように，デコード時の制約として構文情報を利用すると，十分に品質向上に寄与する．

☆ 論文ではラベルとして X を使用している．ただし，例外的に部分木を連続的に結合する規則を人手で与え，これには S（木構造の最上位を意味する）を付与している．

8. まとめ

本稿では，句に基づく構文トランスファ方式統計翻訳を提案した．本稿の提案方式は，従来の句に基づく統計翻訳で用いられている長単位句と構文情報を組み合わせることができ，構文トランスファ単体，長単位句単体の翻訳に比べ，品質が向上することを示した．

本稿では，ソースチャネルモデル，Log-linear モデル双方で適用可能な素性関数のみ使用したが，Log-linear モデルを用いる場合，比較的柔軟に素性関数を追加することができる．今後は，構文構造生成失敗の対策を検討するとともに，本方式に有効な素性関数の検討も行う必要がある．

謝辞 本研究は独立行政法人情報通信研究機構の研究委託「大規模コーパスベース音声対話翻訳技術の研究開発」により実施したものである．

参考文献

1) Akiba, Y., Federico, M., Kando, N., Nakaiwa, H., Paul, M. and Tsujii, J.: Overview of the IWSLT04 Evaluation Campaign, *Proc. IWSLT 2004*, pp.1–12 (2004).
2) Brown, P.F., Pietra, S.A.D., Pietra, V.J.D. and Mercer, R.L.: The Mathematics of Machine Translation: Parameter Estimation, *Computational Linguistics*, Vol.19, No.2, pp.263–311 (1993).
3) Charniak, E.: A Maximum-Entropy-Inspired Parser, *Proc.1st Conference of the North American Chapter of the Association for Computational Linguistics* (*NAACL-2000*), pp.132–139 (2000).
4) Charniak, E., Knight, K. and Yamada, K.: Syntax-based Language Models for Statistical Machine Translation, *Proc. Machine Translation Summit IX*, pp.40–46 (2003).
5) Chiang, D.: A Hierarchical Phrase-Based Model for Statistical Machine Translation, *Proc. 43rd Annual Meeting of the Association for Computational Linguistics* (*ACL'05*), Ann Arbor, Michigan, Association for Computational Linguistics, pp.263–270 (2005).
6) Doddington, G.: Automatic Evaluation of Machine Translation Quality Using N-gram Co-Occurrence Statistics, *Proc. HLT Conference*, San Diego, California (2002).
7) Eck, M. and Hori, C.: Overview of the IWSLT 2005 Evaluation Campaign, *Proc. IWSLT 2005* (2005).
8) Graehl, J. and Knight, K.: Training Tree

Transducers, *HLT-NAACL 2004: Main Proceedings*, Dumais, S., Marcu, D. and Roukos, S. (Eds.), Boston, Massachusetts, USA, Association for Computational Linguistics, pp.105–112 (2004).
9) 今村賢治：構文解析と融合した階層的句アライメント，自然言語処理，Vol.9, No.5, pp.23–42 (2002).
10) Kikui, G., Sumita, E., Takezawa, T. and Yamamoto, S.: Creating Corpora for Speech-to-Speech Translation, *Proc. EuroSpeech 2003*, pp.381–384 (2003).
11) Koehn, P.: Statistical Significance Tests for Machine Translation Evaluation, *Proc. EMNLP 2004*, Lin, D. and Wu, D. (Eds.), Barcelona, Spain, Association for Computational Linguistics, pp.388–395 (2004).
12) Koehn, P., Och, F.J. and Marcu, D.: Statistical Phrase-Based Translation, *HLT-NAACL 2003: Main Proceedings*, Hearst, M. and Ostendorf, M. (Eds.), Edmonton, Alberta, Canada, Association for Computational Linguistics, pp.127–133 (2003).
13) Melamed, I.D.: Statistical Machine Translation by Parsing, *Proc. 42nd Meeting of the Association for Computational Linguistics* (*ACL'04*), *Main Volume*, Barcelona, Spain, pp.653–660 (2004).
14) Nießen, S., Och, F.J., Leusch, G. and Ney, H.: An Evaluation Tool for Machine Translation: Fast Evaluation for MT Research, *Proc. 2nd International Conference on Language Resources and Evaluation* (*LREC 2000*), pp.39–46 (2000).
15) Och, F.J.: Minimum Error Rate Training in Statistical Machine Translation, *Proc. 41st Annual Meeting of the Association for Computational Linguistics*, Hinrichs, E. and Roth, D. (Eds.), pp.160–167 (2003).
16) Och, F.J. and Ney, H.: Discriminative Training and Maximum Entropy Models for Statistical Machine Translation, *Proc. 40th Annual Meeting of the Association for Computational Linguistics* (*ACL*), pp.295–302 (2002).
17) Och, F.J. and Ney, H.: A Systematic Comparison of Various Statistical Alignment Models, *Computational Linguistics*, Vol.29, No.1, pp.19–51 (2003).
18) Och, F.J. and Ney, H.: The Alignment Template Approach to Statistical Machine Translation, *Computational Linguistics*, Vol.30, No.4, pp.417–449 (2004).
19) Papineni, K., Roukos, S., Ward, T. and Zhu, W.-J.: BLEU: a Method for Automatic Evaluation of Machine Translation, *Proc. 40th Annual Meeting of the Association for Computational Linguistics* (*ACL*), pp.311–318 (2002).
20) Stolcke, A.: SRILM — An Extensible Language Modeling Toolkit, *Proc. 7th International Conference on Spoken Language Processing* (*ICSLP 2002*), pp.901–904 (2002).
21) Sumita, E., Yamada, S., Yamamoto, K., Paul, M., Kashioka, H., Ishikawa, K. and Shirai, S.: Solutions to Problems Inherent in Spoken-language Translation: The ATR-MATRIX Approach, *Machine Translation Summit VII*, pp.229–235 (1999).
22) Takezawa, T., Sumita, E., Sugaya, F., Yamamoto, H. and Yamamoto, S.: Toward a Broad-coverage Bilingual Corpus for Speech Translation of Travel Conversations in the Real World, *Proc. 3rd International Conference on Language Resources and Evaluation* (*LREC 2002*), pp.147–152 (2002).
23) Vogel, S., Zhang, Y., Huang, F., Tribble, A., Venugopal, A., Zhao, B. and Waibel, A.: The CMU Statistical Machine Translation System, *Proc. 9th Machine Translation Summit* (*MT Summit IX*), pp.402–409 (2003).
24) Yamada, K. and Knight, K.: A Syntax-based Statistical Translation Model, *Proc. 39th Annual Meeting of the Association for Computational Linguistics*, pp.523–530 (2001).
25) Yasuda, K., Sumita, E., Kikui, G., Yamamoto, S. and Yanagida, M.: Real-Time Evaluation Architecture for MT Using Multiple Backward Translations, *Proc. Recent Advances in Natural Language Processing* (*RANLP-2003*), Borovets, Bulgaria, pp.518–522 (2003).
26) Zens, R. and Ney, H.: Improvements in Phrase-Based Statistical Machine Translation, *HLT-NAACL 2004: Main Proceedings*, Dumais, S., Marcu, D. and Roukos, S. (Eds.), Boston, Massachusetts, USA, Association for Computational Linguistics, pp.257–264 (2004).

(平成 18 年 5 月 22 日受付)
(平成 19 年 1 月 9 日採録)

今村 賢治（正会員）

1985 年千葉大学工学部電気工学科卒業．同年日本電信電話株式会社入社．2000〜2006 年 ATR 音声言語コミュニケーション研究所．2006 年より NTT サイバースペース研究所主任研究員．現在に至る．主として自然言語処理の研究・開発に従事．博士（工学）．電子情報通信学会，言語処理学会各会員．

大熊 英男

1987 年東京理科大学理工学部卒業．同年日本シンボリックス（株）入社．1991 年 ATR 自動翻訳電話研究所にて機械翻訳システムの開発に従事．2003 年 ATR 音声言語コミュニケーション研究所．現在に至る．

隅田英一郎（正会員）

1982 年電気通信大学大学院修士課程修了．1999 年京都大学工学博士．現在，ATR 音声言語コミュニケーション研究所室長．NiCT 知識創成コミュニケーション研究センター研究マネージャ，神戸大学大学院自然科学研究科連携教授，ATR-Lang 取締役副社長兼務．機械翻訳，e ラーニングの研究に従事．IPSJ，IEICE，NLP，ASJ，ACL，IEEE の会員，ACM/TSLP の Associate Editor.