情報処理学会論文誌 Vol.55 No.7 1665–1680 (July 2014)

係り受け木における機械翻訳のための品詞の教師なし学習

田村 晃裕[1,†1,a)] 渡辺 太郎[1,b)] 隅田 英一郎[1,c)] 高村 大也[2,d)] 奥村 学[2,e)]

受付日 2013年7月30日, 採録日 2014年4月4日

概要：本稿では，統語情報に基づく機械翻訳の翻訳性能を向上させるため，ノンパラメトリックベイズ法により，単語間の係り受け構造から各単語の品詞を推定する手法を提案する．提案手法は，単言語における無限ツリーモデル（Infinite Tree Model）を，原言語と目的言語の両言語を考慮するバイリンガルなシナリオに拡張した手法である．提案モデルでは，原言語の係り受け木における各品詞を隠れ状態とし，各隠れ状態は，原言語の単語とともに，対応する目的言語の単語をシンボルとして出力する．本稿では，原言語の単語と目的言語の単語を結合させて出力する「結合モデル」と，別々に出力する「独立モデル」を提案する．NTCIR-9 の日英特許翻訳タスクにおける評価実験を通じて，提案手法により推定した日本語の品詞タグを使うことにより，forest-to-string 翻訳システムの性能を改善できることを示す．また，独立モデルは，結合モデルが抱えるシンボルのスパースネス問題を解決し，既存の品詞を使う従来の翻訳よりも BLEU で 1%以上性能を改善できることを示す．

キーワード：統計的機械翻訳，品詞推定，ノンパラメトリックベイズ

Unsupervised Learning of Part-of-Speech in Dependency Trees for Machine Translation

Akihiro Tamura[1,†1,a)] Taro Watanabe[1,b)] Eiichiro Sumita[1,c)]
Hiroya Takamura[2,d)] Manabu Okumura[2,e)]

Received: July 30, 2013, Accepted: April 4, 2014

Abstract: This paper proposes a nonparametric Bayesian method for inducing Part-of-Speech (POS) tags in dependency trees to improve the performance of machine translation (MT). In particular, we extend the monolingual infinite tree model to a bilingual scenario: each hidden state (POS tag) of a source-side dependency tree emits a source word together with its aligned target word, either jointly (joint model), or independently (independent model). Evaluations of Japanese-to-English translation on the NTCIR-9 data show that our induced Japanese POS tags for dependency trees improve the performance of a forest-to-string MT system. Our independent model gains over 1 point in BLEU by resolving the sparseness problem introduced in the joint model.

Keywords: statistical machine translation, part-of-speech induction, nonparametric Baysian method

1. はじめに

昨今の統計的機械翻訳では，単語以外の翻訳の手がかりとして品詞が利用される場合がある．たとえば，factored translation model [18] において，品詞は単語と並び，1 つの factor として考慮されている．また，原言語や目的言語，あるいはその両方において，係り受け解析 [4], [22], [27], [35], [39] や句構造解析 [3], [9], [14], [24], [25], [26], [27], [46], [47] の結果を利用する，統語情報に基づく機械翻訳においては，統語情報として品詞が利用される．しかし，既存の品詞体

[1] 情報通信研究機構
National Institute of Information and Communications Technology, Soraku, Kyoto 619–0289, Japan
[2] 東京工業大学
Tokyo Institute of Technology, Yokohama, Kanagawa 226–8503, Japan
[†1] 現在，日本電気株式会社
Presently with NEC Corporation
a) a-tamura@ah.jp.nec.com
b) taro.watanabe@nict.go.jp
c) eiichiro.sumita@nict.go.jp
d) takamura@pi.titech.ac.jp
e) oku@pi.titech.ac.jp

系は品詞付与対象の言語を分析して構築されたものであるため，翻訳時に原言語または目的言語となる翻訳相手の言語のことを考慮した品詞ではない．したがって，既存の品詞体系が機械翻訳に必ずしも最適とは限らない．

図 1 を例に説明する．図 1 は，日本語と英語における単語単位の対応関係，既存の日本語品詞および日本語の係り受け構造の例である．係り受け構造は，文節単位の係り受け構造を，4.1 節で後述するヒューリスティクス *Cont* により単語単位に変換した構造である．図 1 の例 1 では，下線箇所のサ変名詞「利用」は英語で「use」という動詞に訳される．これに対して，例 2 では，同じサ変名詞「利用」は「usage」という名詞に訳される．このように，日本語の名詞の中には，英語の動詞として振る舞う場合もあれば，名詞として振る舞うものもある．このような異なる振舞いは，日本語を英語に翻訳する際に有効な手がかりとなりうる．しかし，既存の品詞体系は，少なくともこれら 2 つの働きを区別できる品詞になっておらず，日英翻訳に最適とはいい切れない．また，**図 2** においては，下線箇所の中国語の動詞は，例 3 の場合，英語では「learning」という名詞に訳され，例 4 の場合，英語では「learn」という動詞に訳される．このように，中国語の既存の品詞体系は，英語における異なる振舞いを区別できない場合があることから，中英翻訳において最適とはいい切れない．

この問題に対応するため，図 1 や図 2 のような事例を分析し，既存の品詞体系で区別できない機械翻訳に有効な違いを検出するルールを人手で作成するアプローチが考えられる．しかし，網羅的なルール作成にはコストが膨大になることに加えて，言語ごとにルールを作成する必要があるため，あらゆる言語の翻訳を想定した場合，このアプローチは非現実的である．

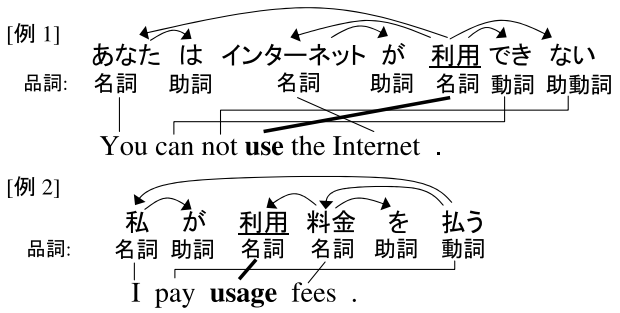

図 1 既存の日本語品詞と係り受け構造の例

Fig. 1 Examples of existing Japanese POS tags and dependency structures.

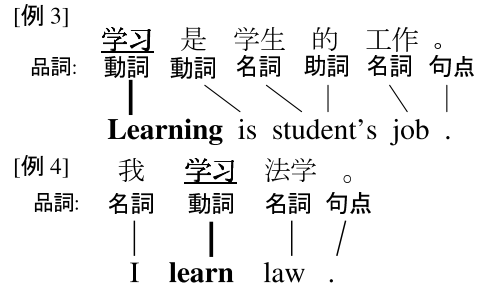

図 2 中国語と英語の単語対応の例

Fig. 2 Examples of word alignment between Chinese and English.

統語情報に基づく機械翻訳は，言語の構造を利用するため，長文や複雑な文を翻訳するための有望な手法として注目されている．これらの状況をふまえて，本稿では，対訳コーパスから機械翻訳に適した品詞体系を学習し，学習した品詞体系を使うことにより，統語情報に基づく機械翻訳の性能改善を試みる．日本語においては，構造解析として係り受け解析が一般的に採用される．係り受け解析を利用する機械翻訳では，句構造解析結果から得られるフレーズ単位の統語情報を活用できないため，品詞の担う役割が大きい．そこで，本稿では，特に，係り受け解析を利用する統語情報に基づく機械翻訳に注目し，機械翻訳に適した品詞を学習することにより，その翻訳性能の改善を試みる．具体的には，コーパス中の係り受け木内の各単語に対して，機械翻訳のための品詞タグを推定する教師なし手法を提案する．そして，その品詞タグが推定されたコーパスを学習データとして機械翻訳システムを構築することにより，翻訳性能の向上を目指す．

提案する品詞推定手法は，Finkel らが提案した単言語における無限ツリーモデル [6] を，原言語と目的言語の両言語を考慮するバイリンガルなシナリオに拡張した手法である．単言語における無限ツリーモデルは，係り受け木を入力とし，単語間の係り受け構造から品詞タグを推定するノンパラメトリックなベイズ手法である．隠れ状態は品詞タグ，隠れ状態が生成する観測可能なシンボルは単語を表し，隠れ状態の遷移は係り受け関係で規定される．たとえば，このモデルにより原言語の各単語の品詞を推定する場合，シンボルは原言語の単語を表す．一方，提案手法は，入力として原言語の係り受け木と，原言語と目的言語の単語間の対応関係を受け取る．そして，原言語の単語に加えて，その単語に対応する目的言語の単語を観測可能なシンボルとして用いる．提案手法は，原言語と目的言語の両方の情報を持つバイリンガルなシンボルに基づいて品詞タグを推定するため，目的言語における違いを考慮して原言語の品詞タグを推定できる．

たとえば，図 1 において，例 1 と例 2 の日本語単語「利用」の品詞タグを推定する場合を考える．単言語における品詞推定（たとえば，無限ツリーモデル）では，例 1 と例 2 のどちらの場合も，「利用」の品詞を示す隠れ状態はシンボル「利用」を出力する．したがって，それらの隠れ状態は同じシンボル出力確率に基づいて推定されるため，例 1 と例 2 の「利用」の品詞は同じになる可能性が高い．一方，提案手法では，対応する目的言語の単語をシンボルに組み込むため，例 1 の「利用」のシンボルは英単語「use」を表

すシンボル，例 2 の「利用」のシンボルは英単語「usage」を表すシンボルとなる．この異なるシンボルに基づいて品詞を推定するため，例 1 と例 2 の「利用」に対して異なる品詞を推定できる可能性が高い．

本稿では，シンボルの生成過程が異なる結合モデルおよび独立モデルの 2 種類のモデルを提案する．結合モデルでは，各隠れ状態は，原言語の単語とその単語に対応する目的言語の単語を結合させて 1 つのシンボルとして出力する．独立モデルでは，原言語の単語とその単語に対応する目的言語の単語を，別々，独立に出力する．

品詞推定モデルにおける各変数の推定は，スライスサンプリングと動的計画法を組み合わせたビームサンプリング [7] により効率良く行う．提案手法の評価は，NTCIR-9 の日英特許翻訳タスクにおける forest-to-string 翻訳システムの翻訳性能で行う．評価実験を通じて，提案手法で推定した品詞体系を用いることで，既存の品詞体系を用いた場合や従来の単言語における無限ツリーモデルで推定した品詞体系を用いた場合よりも性能が良いことを示す．さらに，結合モデルが抱えるシンボルのスパースネス問題を解決した独立モデルを用いることにより，BLEU を 1%以上改善できることを示す．

以降，2 章では本研究の関連研究を説明する．そのなかで，従来の品詞推定手法について説明し，3 章では目的言語を考慮した品詞推定手法を提案する．4 章では，日英特許翻訳での評価実験を行い，提案手法により推定した品詞体系を使うことで特許翻訳の性能が改善できることを示す．また，5 章では提案手法の効果や性質についての考察を行う．最後に，6 章で本稿のまとめを行う．

2. 関連研究

本章では，本研究の関連研究について述べる．従来の統計的機械翻訳の研究には，品詞などの既存の統語情報は機械翻訳の手がかりには粗いという問題に着目し，統語情報に基づく機械翻訳の性能を向上させるため，統語情報の細分化を行う研究がある．たとえば，Wang ら [42] は，言語学的知見に基づく細分化や Petrov ら [34] の統計的な手法による細分化を行うことで，中英の String-to-Tree 翻訳の性能を改善している．また，須藤ら [48] は，統語的役割や句どうしの関係をより詳細に記述する言語素性に基づいた細分化が日英の String-to-Tree 翻訳に有効であることを示している．しかし，これらの従来研究は，いずれも，細分化対象の言語を分析，解析するものであり，本研究のように翻訳相手の言語における違いを考慮した品詞の細分化や品詞体系の学習は行わない．

機械翻訳のための基本処理の研究の中に，翻訳相手の言語の情報を利用した単語分割がある．Xu ら [45] や Nguyen ら [30] は，単言語での単語分割は機械翻訳に最適ではないという問題を解決するため，原言語と目的言語のアライメント関係に基づいた，機械翻訳のための単語分割手法を提案した．これらの研究は，翻訳相手の言語を含めたバイリンガルな情報に基づき，機械翻訳のための基本処理を行う点で本研究と関連している．

次に，コーパスから品詞を推定する教師なし手法に関する従来研究について述べる．初期の手法は，品詞タグの種類数をあらかじめ与えなければならないという問題があった [10], [16]．この問題を解決するため，ノンパラメトリックなベイズ的アプローチにより，品詞タグの種類も自動的にデータに適合させる手法が提案されている [2], [6], [8], [40]．

Gael ら [8] は，ノンパラメトリックな隠れマルコフモデル（HMM）である Infinite HMM（iHMM）[1], [41] を品詞推定に適用した．HMM では，隠れ状態を品詞タグ，隠れ状態が出力する観測可能なシンボルを単語として文の生成過程をモデル化する．そして，シンボル列（単語列；文）を観測している下で，その背後にある隠れ状態列（品詞タグ列）を，隠れ状態の遷移確率と隠れ状態がシンボルを出力するシンボル出力確率に基づき決定する．Infinite HMM は，この HMM において無限の状態を扱えるようにしたものである．

Blunsom ら [2] は，iHMM において，隠れ状態の遷移確率とシンボル出力確率の事前分布に階層 Pitman-Yor 過程を導入した．これにより，データが小規模な場合でも，一般化を適切に行うスムージングが可能となる．Sirts ら [40] は，iHMM を基に，品詞推定と単語分割を同時に 1 つの問題として扱うモデルを提案した．また，Finkel ら [6] は，単語列から品詞タグを推定する iHMM を木構造に拡張することで，単語間の係り受け構造から品詞タグを推定する手法を提案した．これは，無限の状態を持つ隠れ状態間に木構造関係を仮定してモデル化する手法である．このモデルは，無限ツリーモデル（Infinite Tree Model）と呼ばれている．

以降では，後述する提案手法のベースとなる，無限ツリーモデルについて説明する．Finkel ら [6] は，ノードの依存関係，生成過程が異なる 3 種類のモデルを提案している．(1) 子ノードはそれぞれ独立に親ノードから生成される，independent children model，(2) 同一の親ノードを持つ子ノードは同時にその親ノードから生成される，simultaneous children model，(3) 子ノードは親ノードと直前の子ノードに依存して生成される，Markov children model である．3 種類のモデルのうち，提案手法のベースとして用いる independent children model を以降で説明する[*1]．

2.1 有限ツリーモデル

無限ツリーモデルを説明する前に，状態が有限な有限ツリーモデル [6] について説明する．有限ツリーモデルは，

[*1] independent children model 以外の 2 つのモデルも，後述するような提案手法への拡張は可能であるが，今後の課題とする．

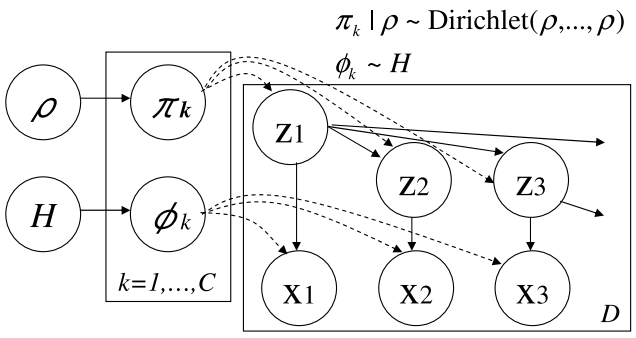

図 3 有限ツリーモデルのグラフィカルモデル
Fig. 3 A graphical representation of the finite tree model.

HMM を木構造に拡張したものであり，隠れ状態間に木構造を仮定する．以降，ルートノードが t である木を T_t と表す．有限ツリーモデルでは，ノード t は，品詞タグを表す隠れ状態 z_t と単語を表すシンボル x_t を持つ．すると，T_t の確率 $(p_T(T_t))$ は，$p_T(T_t) = p(x_t|z_t)\prod_{t' \in c(t)} p(z_{t'}|z_t)p_T(T_{t'})$ と再帰的に定義できる．ここで，$c(t)$ はノード t の子ノードの集合を表す．また，以降，ノード t の親ノードを $d(t)$ で表す．

有限ツリーモデルによるシンボル x_t の生成過程のグラフィカルモデルを**図 3** に示す．図 3 は，$1, \ldots, D$ 個の木に対する生成過程である．グラフィカルモデルとは，変数間の依存関係を有向グラフで表現したものである．図 3 において，隠れ状態（z_t）間の依存関係を示す矢印は係り受け関係である．各隠れ状態が取りうる状態（品詞）は C 個であり，k で指し示される．各状態 k は，パラメータ ϕ_k により規定されるシンボル出力確率分布 $F(\phi_k)$ を持ち，ϕ_k は共通の事前分布 H から生成される．つまり，シンボル x_t は，z_t の状態で具体化する ϕ_{z_t} により規定される分布 $F(\phi_{z_t})$ から生成される．よって，ϕ_k，x_t の生成は，それぞれ，$\phi_k \sim H$，$x_t|z_t \sim F(\phi_{z_t})$ と表記できる．Finkel ら [6] は，シンボル出力確率分布 F として多項分布，F のパラメータ ϕ_k の事前分布 H として対称なディリクレ（Dirichlet）分布を用いている．

状態遷移は，HMM 同様，$\boldsymbol{\pi}$ でパラメータ化されるマルコフ過程である．ここで，π_{ij} は $p(z_{c(t)} = j|z_t = i)$ であり，$\boldsymbol{\pi}_k$ は，親ノードの状態が k のときの状態遷移確率の集合を表す．$\boldsymbol{\pi}_k$ は，ρ をパラメータとするディリクレ分布から生成される．よって，$\boldsymbol{\pi}$ の生成は，$\boldsymbol{\pi}_k|\rho \sim \mathrm{Dirichlet}(\rho, \ldots, \rho)$ と表記できる．また，子ノードの状態 $z_{t'}$ は，親ノードの状態 z_t で具体化する $\boldsymbol{\pi}_{z_t}$ をパラメータとする，多項分布 $\mathrm{Multinomial}(\boldsymbol{\pi}_{z_t})$ で確率的に決定される．よって，$z_{t'}$ の生成は，$z_{t'}|z_t \sim \mathrm{Multinomial}(\boldsymbol{\pi}_{z_t})$ と表記できる．

以上をまとめると，有限ツリーモデルは，

$$\begin{aligned}
&\boldsymbol{\pi}_k|\rho \sim \mathrm{Dirichlet}(\rho, \ldots, \rho),\\
&\phi_k \sim H,\\
&z_{t'}|z_t \sim \mathrm{Multinomial}(\boldsymbol{\pi}_{z_t}),\\
&x_t|z_t \sim F(\phi_{z_t})
\end{aligned}$$

と定義できる．ここで，$\boldsymbol{\pi}_k$ と ϕ_k は状態（品詞）ごとのパラメータ，x_t と z_t は単語ごとのパラメータであることを確認しておく．

2.2 無限ツリーモデル

無限ツリーモデルは，有限ツリーモデルに階層ディリクレ過程（hierarchical Dirichlet process；HDP）[41] を適用して拡張することで，無限の状態が扱えるようにしたものである．

ディリクレ過程（DP）[5] は，確率分布に対する分布で，集中度パラメータ α_0 と基底測度 G_0 で定義される．この DP を $\mathrm{DP}(\alpha_0, G_0)$ と表記する．そして，G がこの DP に従うとき，$G \sim \mathrm{DP}(\alpha_0, G_0)$ と表記する．また，Sethuraman [38] は，$\mathrm{DP}(\alpha_0, G_0)$ に従う G は，それぞれ互いに独立で同一の分布に従う 2 つの無限個の確率変数[*2]の列 $(\psi_k)_{k=1}^{\infty}$ と $(\theta_k)_{k=1}^{\infty}$ を用いて，$\psi_k = \psi'_k \prod_{l=1}^{k-1}(1-\psi'_l)$，$G = \sum_{k=1}^{\infty} \psi_k \delta_{\theta_k}$ のように生成できることを示した．$(\psi'_k)_{k=1}^{\infty}$ と $(\theta_k)_{k=1}^{\infty}$ は，$\psi'_k|\alpha_0 \sim \mathrm{Beta}(1, \alpha_0)$，$\theta_k \sim G_0$ のとおり生成される．また，δ_{θ_k} はディラック測度である．この G の生成過程は Stick-breaking 過程と呼ばれ，$\boldsymbol{\psi}$ がこの過程で生成されるとき，$\boldsymbol{\psi} \sim \mathrm{GEM}(\alpha_0)$ と表記する．

無限ツリーモデルへの拡張の際，この単純なディリクレ過程（DP）ではなく階層ディリクレ過程（HDP）を用いた理由は，違った親ノードから生成される子ノードの状態を共有するためである．HMM から iHMM への拡張にも，同じ動機で階層ディリクレ過程が用いられている [1]．

HDP は，共通の基底測度によって関連付けられた DP の集合であり，その基底測度は，グローバルな基底測度で定義される DP により生成される．つまり，共通の基底測度を G_0，グローバルな基底測度を H とすると，HDP は，$G_0 \sim \mathrm{DP}(\gamma, H)$，各 $G_k \sim \mathrm{DP}(\alpha_0, G_0)$ である．ここで，グローバルな基底測度 H は人手であらかじめ決められるものであり，共通の基底測度 G_0 はモデルにより生成されることを特筆しておく．この HDP は，Stick-breaking 過程の観点から考えると，$G_0 = \sum_{k'=1}^{\infty} \boldsymbol{\beta}_{k'} \delta_{\phi_{k'}}$，$G_k = \sum_{k'=1}^{\infty} \pi_{kk'} \delta_{\phi_{k'}}$ と解釈でき，$\boldsymbol{\beta}$，$\boldsymbol{\pi}_k$，$\phi_{k'}$ を，それぞれ，$\boldsymbol{\beta} \sim \mathrm{GEM}(\gamma)$，$\boldsymbol{\pi}_k \sim \mathrm{DP}(\alpha_0, \boldsymbol{\beta})$，$\phi_{k'} \sim H$ のとおり生成する．

上記の HDP を無限ツリーモデルに適用すると，G_0 は子ノードの状態に共通の確率分布，G_k は親ノードの状態に特有の確率分布に対応する．そして，G_0 は，子ノードの状態に共通の DP についてのパラメータ（$\boldsymbol{\beta}_{k'}$）と，状態が k' のときのシンボル出力（$\phi_{k'}$）で規定される．また，各 G_k は，親ノードの状態が k のときの状態遷移（$\boldsymbol{\pi}_k$）と，状態が k' のときのシンボル出力（$\phi_{k'}$）で規定されると解釈できる．

以上をまとめると，無限ツリーモデルは，

[*2] 多くの場合，π と ϕ を用いて説明されるが，状態遷移とシンボル出力に関する π や ϕ との混同を避けるため，ψ と θ を用いる．

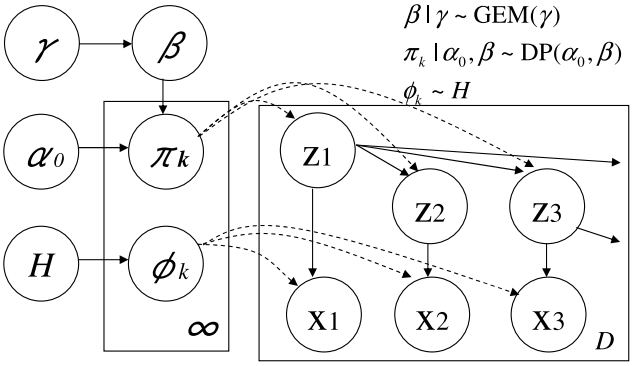

図 4 無限ツリーモデルのグラフィカルモデル
Fig. 4 A graphical representation of the infinite tree model.

$$\boldsymbol{\beta}|\gamma \sim \mathrm{GEM}(\gamma),$$
$$\boldsymbol{\pi}_k|\alpha_0,\boldsymbol{\beta} \sim \mathrm{DP}(\alpha_0,\boldsymbol{\beta}),$$
$$\phi_k \sim H,$$
$$z_{t'}|z_t \sim \mathrm{Multinomial}(\boldsymbol{\pi}_{z_t}),$$
$$x_t|z_t \sim F(\phi_{z_t})$$

と定義できる．

無限ツリーモデルによるシンボル x_t の生成過程のグラフィカルモデルを**図 4** に示す．有限ツリーモデル（図 3）との違いは，子ノードの状態に共通の DP についてのパラメータ $\boldsymbol{\beta}$ が導入され，状態数が無限になっている点である．

3. バイリンガルな無限ツリーモデル

本章では，係り受け木内の各単語に対して，翻訳のための品詞を推定する提案手法について述べる．提案手法は，2.2 節で説明した単言語における無限ツリーモデルを，対応する目的言語の情報を利用するように拡張した手法である．具体的には，入力として，原言語の係り受け木のほかに，原言語と目的言語の単語間の対応関係を受け取る．この係り受け木や単語対応は，人手で特定したものに限らず，既存ツールの解析結果も想定している．後述する実験では，それぞれ，CaboCha [20] および GIZA++ [32] の解析結果を用いた．そして，この単語の対応関係に基づいて，原言語の係り受け木に，対応する目的言語の単語を埋め込むことで，原言語と目的言語の両方の情報を持つバイリンガルなシンボルを用いる．本章では，シンボルの生成過程が異なる 2 種類のモデル，「結合モデル」と「独立モデル」を説明する．

3.1 結合モデル

本節では，提案手法の 1 つである結合モデルを説明する．結合モデルは，単言語における無限ツリーモデルを，原言語と目的言語の 2 言語が対象となるように単純に拡張したモデルである．結合モデルの形式的な定義およびグラフィカルモデルは，単言語における無限ツリーモデルと同じである（2.2 節参照）．ただし，Finkel ら [6] は，α_0 や γ としてあらかじめ決めた定数を使用するが，提案手法では，α_0 と γ を，それぞれ，ハイパーパラメータ α_a と α_b，γ_a と γ_b を持つガンマ分布によりパラメータ化する．このパラメータ化は提案手法の本質ではないため，詳細は 3.5 節で後述し，本節では割愛する．

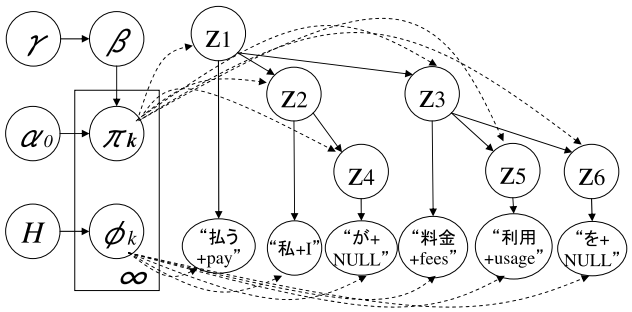

図 5 結合モデルによる生成過程の例
Fig. 5 An example of the joint model.

単言語における無限ツリーモデルとの違いは，シンボル（x_t）のインスタンスである．単言語における無限ツリーモデルは，原言語の単語を表すシンボルを使うのに対し，結合モデルは，原言語の単語とその単語に対応する目的言語の単語との結合文字列をシンボルとして用いる．対応する目的言語が複数ある場合は，原言語の単語にそれらの目的言語の単語をアルファベット順ですべて結合させた文字列をシンボルとして用いる．対応する目的言語が存在しない場合は「NULL」を結合させる．

提案モデルは原言語を中心にモデル化するため，原言語の複数の単語に対応付く目的言語の単語は，シンボルとして複数回出力される．また，原言語のどの単語にも対応しない目的言語の単語は一度も出力されない．これらは，次の 3.2 節で説明する独立モデルにおいても同じである．

図 5 に，図 1 の例 2 を結合モデルにより生成する過程を示す．図 5 で示されているとおり，各隠れ状態は，日本語単語とその日本語単語に対応する英単語が結合したシンボルを出力する．たとえば，「払う」の品詞タグ（z_1）は，日本語単語「払う」とその対応英単語「pay」の結合文字列「払う+pay」をシンボル（x_1）として出力する．また，対応する英単語が存在しない「が」の品詞タグ（z_4）は，日本語単語「が」と「NULL」の結合文字列「が+NULL」をシンボル（x_4）として出力する．そして，「利用」の品詞タグ（z_5）は，日本語単語「利用」とその対応英単語「usage」との結合文字列「利用+usage」をシンボル（x_5）として出力する．同様に，図 1 の例 1 の「利用」の品詞タグは，文字列「利用+use」をシンボルとして出力する．

このように，結合モデルでは，例 1 と例 2 の「利用」のシンボルとして異なるインスタンスが使われる．そして，例 1 と例 2 の「利用」の品詞タグは，異なるシンボル出力確率に基づいて推定される（3.5 節参照）ため，異なる品

詞が割り当てられ，区別できる可能性が高い．

3.2 独立モデル

3.1 節の結合モデルは，原言語の単語とその単語に対応する目的言語の単語の組合せをシンボルとするため，シンボルのスパースネスの問題に陥りやすい．そこで，本節では，各隠れ状態が原言語の単語とその単語に対応する目的言語の単語を別々に独立に出力する，独立モデルを提案する．

独立モデルでは，各隠れ状態 z_t に対して，原言語の単語を表すシンボル x_t に加え，目的言語の単語用のシンボル x'_t を設ける．また，各状態 k は，パラメータ ϕ_k で規定される，原言語用のシンボル出力確率分布に加えて，パラメータ ϕ'_k で規定される，目的言語用のシンボル出力確率分布を持つ．ここで，ϕ'_k は共通の事前分布 H' から生成される．つまり，シンボル x'_t は，シンボル x_t とは独立に，z_t の状態で具体化する ϕ'_{z_t} により規定される分布 $F'(\phi'_{z_t})$ から生成される．対応する目的言語の単語が複数ある場合は，それらの目的言語の単語は，分布 $F'(\phi'_{z_t})$ から別々に生成される．以上をまとめると，独立モデルは，

$$
\begin{aligned}
&\boldsymbol{\beta}|\gamma \sim \mathrm{GEM}(\gamma),\\
&\boldsymbol{\pi}_k|\alpha_0, \boldsymbol{\beta} \sim \mathrm{DP}(\alpha_0, \boldsymbol{\beta}),\\
&\phi_k \sim H,\ \ \phi'_k \sim H',\\
&z_{t'}|z_t \sim \mathrm{Multinomial}(\boldsymbol{\pi}_{z_t}),\\
&x_t|z_t \sim F(\phi_{z_t}),\ \ x'_t|z_t \sim F'(\phi'_{z_t})
\end{aligned}
$$

と定義される．

図 6 に，図 1 の例 2 を独立モデルにより生成する過程を示す．図 6 で示されているとおり，対応する英単語用の x'_t と ϕ'_k が導入されている．そして，たとえば，「利用」の品詞タグ（z_5）は，日本語単語「利用」を x_5 として出力し，その対応英単語「usage」を x'_5 として別に出力する．このように，独立モデルは，原言語と目的言語のシンボルを分けて扱うことでシンボルのスパースネスの問題を緩和する．

3.3 目的言語の素性

前節までは，提案モデルのシンボルに加える目的言語の情報として，対応する目的言語の単語の表層を考えていた．本節では，目的言語の単語の表層以外の情報として，対応する目的言語の単語の品詞を提案モデルに導入する．ここで，提案モデルは原言語の単語への品詞付与を仮定しているため，目的言語の単語の品詞は，既存の目的言語の品詞タガーが付与する品詞を使い，提案モデルで推定しないことを特筆しておく．

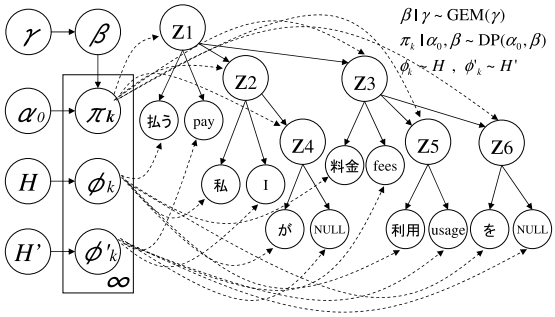

図 6 独立モデルによる生成過程の例
Fig. 6 An example of the joint model.

単純には，対応する目的言語の単語の表層を用いる代わりに，その品詞を代用することで，提案モデルに導入できる．品詞の代用により，シンボルのスパースネス問題がより緩和されると思われる．例として，図 1 の例 2 を生成する過程を考える．結合モデルでは，たとえば，「利用」の品詞タグ（z_5）は，日本語単語「利用」とその対応英単語「usage」の品詞「noun」の結合文字列「利用+noun」をシンボル（x_5）として出力する．一方，独立モデルでは，日本語単語「利用」を x_5 として出力し，その対応英単語の品詞「noun」を x'_5 として出力する．

また，対応する目的言語の単語の表層とその品詞の両者をシンボルに反映することもできる．結合モデルでは，その両者の情報を結合させて 1 つのシンボルとする．たとえば，図 1 の例 2 を生成する過程では，「利用」の品詞タグ（z_5）は，日本語単語「利用」とその対応英単語の表層「usage」とその英単語の品詞「noun」の結合文字列「利用+usage+noun」をシンボル（x_5）として出力する．

一方，独立モデルでは，それぞれの情報に，シンボルとシンボル出力確率分布を導入する．具体的には，原言語の単語用のシンボル x_t とパラメータ ϕ_k，対応する目的言語の単語の表層用のシンボル x'_t とパラメータ ϕ'_k，対応する目的言語の単語の品詞用のシンボル x''_t とパラメータ ϕ''_k を設ける．そして，各隠れ状態 z_t は，x_t，x'_t，x''_t を，それぞれ，z_t の状態で具体化するパラメータ ϕ_{z_t}，ϕ'_{z_t}，ϕ''_{z_t} により規定される分布から互いに独立に生成する．たとえば，図 1 の例 2 を生成する過程では，「利用」の品詞タグ（z_5）は，日本語単語「利用」，対応英単語の表層「usage」，対応英単語の品詞「noun」を，シンボル x_5，x'_5，x''_5 としてそれぞれ独立に出力する．

3.4 品詞細分化

前節までの提案モデルは，品詞推定に何の制約も設けず，z_t で表された新しい品詞体系を推定することを想定していた．本節では，提案モデルのそのほかの適用方法として，既存の品詞体系の細分化 [6], [21] を説明する．品詞細分化では，既存の品詞体系で表されている，人間が見い出した原言語における違いを区別しつつ，目的言語の情報（対応する目的言語の単語の表層や品詞）を反映した品詞体系を推定できる．

提案手法による既存の品詞（s）の細分化では，まず，既存の原言語の品詞タガーなどにより，各ノード t に対して，

既存の原言語の品詞タグ s_t を割り当てる．その後，各ノードに対し，s_t を制約とした状態の推定を行うことで既存の品詞の細分化を実現する．具体的には，各ノードの状態（品詞）を，s_t と z_t のペア (s_t, z_t) で規定する．たとえば，ノード t の初期状態は，$(s_t, 0)$ に設定する．また，既存の品詞タグ（s）ごとに，取りうる状態を指し示すパラメータ（k_s），状態遷移確率（$\boldsymbol{\pi}^s_{k_s}$）とシンボル出力確率分布（$\phi^s_{k_s}$，$\phi^{'s}_{k_s}$，$\phi^{''s}_{k_s}$）を考える．そして，各ノード t の状態遷移やシンボルの出力を s_t に対応する分布に従って行う．つまり，状態遷移は $\boldsymbol{\pi}^{s_t}_{k_{s_t}}$ で規定される分布に従い，シンボル出力は $\phi^{s_t}_{k_{s_t}}$，$\phi^{'s_t}_{k_{s_t}}$，$\phi^{''s_t}_{k_{s_t}}$ で規定される分布に従う．このモデルで各ノード t の状態を推定（3.5 節参照）することにより，ノード t の状態として，s_t を細分化した状態（たとえば，$(s_t, 1)$，$(s_t, 2)$，(s_t, k_{s_t})）が推定される．

3.5 推定

本節では，提案モデル（結合モデル，独立モデル）において，シンボル（$x_{1:T}$）を基に，その背後にある品詞を示す隠れ状態（$z_{1:T}$）を推定する方法を説明する．以降，変数 $_1$, 変数 $_2$, $\cdots$, 変数 $_T$ を変数 $_{1:T}$ と表記する．推定とは，シンボルが与えられたときの事後確率（$P(z_{1:T}|x_{1:T})$）が最大となる状態を特定することである．

状態数が無限なモデルでは，取りうるすべての状態に対して，この事後確率を計算することは不可能である．そこで，Teh ら [41] は，iHMM において，ギブスサンプリングにより推定する方法を提案している．Finkel ら [6] は，Teh ら [41] が提案した iHMM のためのギブスサンプリングによる推定手法を無限ツリーモデルに適用した．提案モデルでも，Finkel ら [6] と同様のギブスサンプリングによる推定を行うことができる．ギブスサンプリングによる推定は，まず各変数に初期値を与え，その後，それぞれの変数に対して，他の変数を固定したリサンプリングを繰り返すことで値を更新していく．したがって，各隠れ状態は，その他の隠れ状態を固定してリサンプリングされるため，隠れ状態間で強い依存関係を持つ HMM のような系列モデルに対しては，ギブスサンプリングは収束が遅いことが示されている [7]．

そこで，本節では，iHMM に対するビームサンプリング [7] を拡張し，結合モデルと独立モデルのためのビームサンプリングによる推定方法を説明する．ビームサンプリングは，スライスサンプリング [29] により，各ノードが取りうる状態遷移を有限に絞り込む．そして，動的計画法を用いて，有限となった状態遷移の全候補を考慮したサンプリングを行う．ビームサンプリングでは，全状態が一度にリサンプリングされるため，ギブスサンプリングで生じる収束が遅いという問題は緩和される．また，ビームサンプリングは，ギブスサンプリングよりも変数の初期値やハイパーパラメータの値に頑健であることが示されている [7]．

具体的には，スライスサンプリングにより各ノードが取りうる状態遷移を絞り込むために，各ノードに補助変数 u_t（$t = 1, \ldots, T$）を設ける．そして，次の 6 種類の変数のサンプリングを交互に繰返し行う．(1) 補助変数 $\boldsymbol{u}$，(2) 状態変数 $\boldsymbol{z}$，(3) 遷移確率 $\boldsymbol{\pi}$，(4) 共通の DP についてのパラメータ $\boldsymbol{\beta}$，(5) ハイパーパラメータ α_0，(6) ハイパーパラメータ γ である．各サンプリングにおいては，その他の変数の値を固定してサンプリングする．

以降，各変数のサンプリングについて説明する．$\boldsymbol{\pi}$，$\boldsymbol{\beta}$，α_0，γ のサンプリングに関しては，Teh ら [41] と同じ方法である．結合モデルにおける推定と独立モデルにおける推定の違いは，$\boldsymbol{z}$ のサンプリング中に行う，シンボルを条件とした状態の事後確率（$p(z_{1:T}|x_{1:T})$，$p(z_{1:T}|x_{1:T}, x'_{1:T})$）の計算だけである．

$\boldsymbol{u}$ のサンプリング

各 u_t は，$u_t \sim \mathrm{Uniform}(0, \pi_{z_{d(t)}z_t})$ のとおり，区間 $[0, \pi_{z_{d(t)}z_t}]$ の一様分布からサンプリングする．遷移確率 $\pi_{z_{d(t)}z_t}$ は 0 より大きい値なので，u_t は正の値になることを特筆しておく．この u_t は，$\boldsymbol{z}$ のサンプリング中で，状態遷移の絞り込みに使われる．

$\boldsymbol{z}$ のサンプリング

$\boldsymbol{z}$ は，forward filtering-backward sampling [7] を木構造に拡張し，サンプリングを行う．forward filtering では前順走査，backward sampling では後順走査で処理を行う．その際，提案モデルは independent children model を仮定しているため，兄弟ノードを考慮しないことを特筆しておく．具体的には，まず，u_t に基づいて状態遷移をフィルタリングしながら，前向きアルゴリズムにより，シンボルを条件とした z_t の事後分布を計算する（forward filtering）．その後，計算した前向きの事後分布を使って，後ろ向きに z_t の事後分布を求め，求めた事後分布から z_t をサンプリングする（backward sampling）．Gael ら [8] は，iHMM に対するビームサンプリングにおいて，計算量を削減するため，これらの動的計画法およびサンプリングを文ごとに並列に処理している．本稿の実験でも，Gael ら [8] にならい，並列に処理を行う[*3]．以降，forward filtering と backward sampling をそれぞれ説明する．

forward filtering：

各 z_t の取りうる状態 k は，u_t を用いて次の 2 つの集合に分割できる．$\pi_{z_{d(t)}k} > u_t$ を満たす有限集合と $\pi_{z_{d(t)}k} \leq u_t$ を満たす無限集合である．ビームサンプリングでは，後者の無限集合を切り捨て，前者の有限集合のみを考える．

この状態遷移の有限化により，動的計画法で前向きに，全ノードに対してシンボルを条件とした z_t の事後分布

[*3] 1 文ごとに処理し，順次モデルを更新する場合とは結果が異なるが，計算量を削減するため，Gael ら [8] の近似的手法を採用した．このヒューリスティクスが本タスクに与える影響の調査は，今後の課題とする．

$(p(z_t|x_{\sigma(t)}, u_{\sigma(t)})$，$p(z_t|x_{\sigma(t)}, x'_{\sigma(t)}, u_{\sigma(t)}))$ を計算できる．$x_{\sigma(t)}$ は，ルートノードからノード t までの経路上にある x_t の集合を表す．同様に，$u_{\sigma(t)}$ は，ルートノードからノード t までの経路上にある u_t の集合を表す．

ルートノードを z_1 とすると，その事後分布は $p(x_1|z_1)$ とする．そして，ルートノード以外のノード z_t の事後分布は，結合モデルの場合，

$$
\begin{aligned}
&p(z_t|x_{\sigma(t)}, u_{\sigma(t)}) \\
&\propto p(z_t, u_t, x_t|x_{\sigma(d(t))}, u_{\sigma(d(t))}) \\
&= \sum_{z_{d(t)}} p(x_t|z_t) \cdot p(u_t|z_t, z_{d(t)}) \cdot p(z_t|z_{d(t)}) \\
&\quad \cdot p(z_{d(t)}|x_{\sigma(d(t))}, u_{\sigma(d(t))}) \\
&= p(x_t|z_t) \cdot \sum_{z_{d(t)}} [\pi_{z_{d(t)}z_t} > u_t] \cdot p(z_{d(t)}|x_{\sigma(d(t))}, u_{\sigma(d(t))}) \\
&= p(x_t|z_t) \cdot \sum_{z_{d(t)}:\pi_{z_{d(t)}z_t} > u_t} p(z_{d(t)}|x_{\sigma(d(t))}, u_{\sigma(d(t))})
\end{aligned}
$$

で計算できる[*4]．ここで，角括弧 [] はアイバーソンの記法で，条件 C が真の場合，$[C] = 1$，それ以外の場合，$[C] = 0$ である．上記導出過程の 2 段階目の変形では，u_t の確率密度が $p(u_t|z_{d(t)}, z_t, \boldsymbol{\pi}) = \frac{[0<u_t<\pi_{z_{d(t)}z_t}]}{\pi_{z_{d(t)}z_t}}$，$p(z_t|z_{d(t)}) = \pi_{z_{d(t)}z_t}$ であることを用いた．同様に，独立モデルの場合，z_t の事後分布は，

$$
\begin{aligned}
&p(z_t|x_{\sigma(t)}, x'_{\sigma(t)}, u_{\sigma(t)}) \\
&\propto p(z_t, u_t, x_t, x'_t|x_{\sigma(d(t))}, x'_{\sigma(d(t))}, u_{\sigma(d(t))}) \\
&\quad = \sum_{z_{d(t)}} p(x_t|z_t) \cdot p(x'_t|z_t) \cdot p(u_t|z_t, z_{d(t)}) \cdot p(z_t|z_{d(t)}) \\
&\quad \cdot p(z_{d(t)}|x_{\sigma(d(t))}, x'_{\sigma(d(t))}, u_{\sigma(d(t))}) \\
&= p(x_t|z_t) \cdot p(x'_t|z_t) \\
&\quad \cdot \sum_{z_{d(t)}} [\pi_{z_{d(t)}z_t} > u_t] \cdot p(z_{d(t)}|x_{\sigma(d(t))}, x'_{\sigma(d(t))}, u_{\sigma(d(t))}) \\
&= p(x_t|z_t) \cdot p(x'_t|z_t) \\
&\quad \cdot \sum_{z_{d(t)}:\pi_{z_{d(t)}z_t} > u_t} p(z_{d(t)}|x_{\sigma(d(t))}, x'_{\sigma(d(t))}, u_{\sigma(d(t))})
\end{aligned}
$$

で計算できる．

後述する 4 章の評価実験では，Finkel ら [6] の実験設定にならい，$F(\phi_k)$ は多項分布 Multinomial(ϕ_k)，H は対称なディリクレ分布 Dirichlet$(\rho, \ldots, \rho)$ を用いた．この仮定の下では，シンボルの事後確率は，$p(x_t|z_t) = \frac{\nu_{x_t z_t} + \rho}{\nu_{\cdot z_t} + N\rho}$ により計算できる[*5]．ここで，$\nu_{x_t z_t}$ は，状態が z_t であるシンボル x_t の数，$\nu_{\cdot z_t}$ は，シンボル $\boldsymbol{x}$ の中で状態が z_t であるシンボル数，N は，シンボル $\boldsymbol{x}$ の総数である．また，独立モデルにも同様の仮定を設けると，シンボル x'_t の事後確率は，$p(x'_t|z_t) = \frac{\nu'_{x'_t z_t} + \rho'}{\nu'_{\cdot z_t} + N'\rho'}$ により計算できる．$\nu'_{x'_t z_t}$ は，状態が z_t であるシンボル x'_t の数，$\nu'_{\cdot z_t}$ は，シンボル $\boldsymbol{x}'$ の中で状態が z_t であるシンボル数，N' は，シンボル $\boldsymbol{x}'$ の総数である．

backward sampling：

全ノードに対して z_t の前向きの事後分布を計算した後は，後ろ向きに z_t をサンプリングしていく．まず，各葉ノードの状態を，その前向きの事後分布からサンプリングする．その後，葉ノードからルートノードに遡りながら，$z_{c(t)}$ のサンプリング結果を使って計算される事後分布から，バックトラックで z_t をサンプリングしていく．具体的には，結合モデルの場合，$p(z_t|z_{c(t)}, x_{1:T}, u_{1:T}) \propto p(z_t|x_{\sigma(t)}, u_{\sigma(t)}) \cdot \prod_{t' \in c(t)} p(z_{t'}|z_t, u_{t'})$ からサンプリングする．また，独立モデルの場合，$p(z_t|z_{c(t)}, x_{1:T}, x'_{1:T}, u_{1:T}) \propto p(z_t|x_{\sigma(t)}, x'_{\sigma(t)}, u_{\sigma(t)}) \cdot \prod_{t' \in c(t)} p(z_{t'}|z_t, u_{t'})$ からサンプリングする．本サンプリング中は $\boldsymbol{z}$ 以外のパラメータは変化させないことを特筆しておく．すなわち，新しい状態（品詞）が複数回サンプリングされた場合，それらは同一視し，新しく状態を増やすことはしない．

$\boldsymbol{\pi}$ のサンプリング

親ノードの状態が i で状態が j のノード数を変数 $n_{ij} \in \boldsymbol{n}$ で表す．そして，$\boldsymbol{\pi}$ は，

$$
\begin{aligned}
&\left(\pi_{k1}, \ldots, \pi_{kK}, \sum_{k'=K+1}^{\infty} \pi_{kk'}\right) \\
&\sim \text{Dirichlet}\left(n_{k1} + \alpha_0\beta_1, \ldots, n_{kK} + \alpha_0\beta_K, \alpha_0 \sum_{k'=K+1}^{\infty} \beta_{k'}\right)
\end{aligned}
$$

のとおり，ディリクレ分布からサンプリングする．ここで，K は $\boldsymbol{z}$ 中の状態の異なり数である．

$\boldsymbol{\beta}$ のサンプリング

Teh ら [41] の式 (36)，(40) と同様のサンプリングを行う．$\boldsymbol{\pi}_i$ の要素の中で β_j に対応する，すなわち状態 j への遷移を表す要素数を変数 $m_{ij} \in \boldsymbol{m}$ で表す．m_{ij} の事後分布は，$p(m_{ij}|\boldsymbol{z}, \boldsymbol{\beta}, \alpha_0) \propto S(n_{ij}, m_{ij})(\alpha_0\beta_j)^{m_{ij}}$ である．ここで，$S(a, b)$ は符号なし第 1 種スターリング数[*6]である．

この媒介変数 m_{ij} を用いて，$\boldsymbol{\beta}$ は，$\left(\beta_1, \ldots, \beta_K, \sum_{k'=K+1}^{\infty} \beta_{k'}\right) \sim \text{Dirichlet}(m_{\cdot 1}, \ldots, m_{\cdot K}, \gamma)$ のとおり，ディリクレ分布からサンプリングする．ここで，$m_{\cdot k} = \sum_{k'=1}^{K} m_{k'k}$ である．

$\boldsymbol{\alpha_0}$ のサンプリング

α_0 は，ハイパーパラメータ α_a と α_b を持つガンマ分布によりパラメータ化する．具体的には，各状態 $(k = 1, \ldots, K)$ に対して，$w_k \in [0, 1]$ と $v_k \in \{0, 1\}$ の 2 つの媒介変数を導入し，分布 $q(\alpha_0, \boldsymbol{w}, \boldsymbol{v}) \propto$

[*4] 導出を見やすくするため，条件となる $\boldsymbol{\pi}$ と $\boldsymbol{\phi}$ の表記は省略してある．

[*5] F と H の仮定の下，シンボル出力確率は観測値 $\boldsymbol{\nu}$ から算出し，ϕ の直接の推定は行わない．

[*6] $S(0, 0) = S(1, 1) = 1$．$a > 0$ の場合，$S(a, 0) = 0$．$b > a$ の場合，$S(a, b) = 0$．その他の場合，$S(a + 1, b) = S(a, b - 1) + aS(a, b)$．

$\alpha_0^{\alpha_a-1+m_{\cdot\cdot}} e^{-\alpha_0\alpha_b} \prod_{k=1}^{K} w_k^{\alpha_0}(1-w_k)^{n_{\cdot k}-1} \left(\frac{n_{\cdot k}}{\alpha_0}\right)^{v_k}$ を定義する [41]．ここで，$m_{\cdot\cdot} = \sum_{k'=1}^{K}\sum_{k''=1}^{K} m_{k'k''}$ である．そして，α_0 は，この分布を α_0 以外の変数について周辺化して得た事後分布 $q(\alpha_0|\boldsymbol{w},\boldsymbol{v}) \propto \alpha_0^{\alpha_a-1+m_{\cdot\cdot}-\sum_{k=1}^{K} v_k} e^{-\alpha_0\left(\alpha_b-\sum_{k=1}^{K}\log w_k\right)}$ からサンプリングする．この事後分布はガンマ分布 $\mathrm{Gamma}(\alpha_a + m_{\cdot\cdot} - \sum_{k=1}^{K} v_k, \alpha_b - \sum_{k=1}^{K}\log w_k)$ である．

また，α_0 を条件とすると，w_k と v_k の事後分布が独立に得られる．これらの事後分布は，それぞれ，ベータ分布 $\mathrm{Beta}(\alpha_0+1, n_{\cdot k})$，二項分布 $\mathrm{Bin}\left(1, \frac{n_{\cdot k}}{\alpha_0}\right)$ である．したがって，w_k および v_k は，それぞれ，$q(w_k|\alpha_0) \propto w_k^{\alpha_0}(1-w_k)^{n_{\cdot k}-1}$，$q(v_k|\alpha_0) \propto \left(\frac{n_{\cdot k}}{\alpha_0}\right)^{v_k}$ からサンプリングする．ここで，$n_{\cdot k} = \sum_{k'=1}^{K} n_{k'k}$ である．

γ のサンプリング

γ も α_0 と同様に，ハイパーパラメータ γ_a と γ_b を持つガンマ分布によりパラメータ化する．具体的には，媒介変数 $\eta \in [0,1]$ と分布 $q(\gamma,\eta) \propto \gamma^{\gamma_a-1+K} e^{-\gamma\gamma_b} \eta^{\gamma}(1-\eta)^{m_{\cdot\cdot}-1}$ を定義する．そして，γ は，この分布を η について周辺化して得た事後分布 $q(\gamma|\eta) \propto \gamma^{\gamma_a-1+K} e^{-\gamma(\gamma_b-\log\eta)}$ からサンプリングする．この事後分布はガンマ分布 $\mathrm{Gamma}(\gamma_a+K, \gamma_b-\log\eta)$ である．また，η は，γ を条件とした事後分布 $q(\eta|\gamma) \propto \eta^{\gamma}(1-\eta)^{m_{\cdot\cdot}-1}$ からサンプリングする．この事後分布はベータ分布 $\mathrm{Beta}(\gamma+1, m_{\cdot\cdot})$ である．

4. 評価実験

本章では，3 章で述べた提案手法の性能および有効性を評価する．提案手法の目的は，翻訳性能を向上させるため，翻訳に適した品詞を推定することである．したがって，提案手法により推定した品詞を使った翻訳システムの性能評価を行う．翻訳システムは，原言語の統語情報（係り受け木）を用いる forest-to-string 翻訳システムを使う．また，評価は，NTCIR-9 の日英特許翻訳タスク [11] において行う．このタスクでは，訓練データとして約 320 万の対訳文，ディベロップメントデータおよびテストデータとして，それぞれ，2,000 の対訳文が提供されている．本章の実験では，これらのデータに加え，NTCIR-7 で提供されたディベロップメントデータ（2,741 の対訳文）を，ディベロップメント時におけるテスト目的で用いる．これをディベロップメントテストデータと呼ぶ．

4.1 評価手順

本節では，評価対象の翻訳システムを構築する手順を説明する．翻訳システムは，(1) データの前処理，(2) 原言語の品詞タグ推定，(3) 原言語の品詞付与および係り受け解析器の学習，(4) forest-to-string 翻訳モデルの学習の 4 ステップで構築する．以降，各ステップの説明を行う．

ステップ 1. 前処理

本ステップでは，3 章で提案した品詞推定手法を適用するために必要な前処理を行う．NTCIR-9 の訓練データのうち，最初の 10,000 の日英対訳文を，機械翻訳のための日本語の品詞を推定するために使用する[*7]．まず，各文に対して単語分割および品詞付与を行う．日本語文は MeCab[*8]，英文は TreeTagger [37] により行う．日本語の品詞は，IPA 品詞体系の 2 階層目の品詞を用いる．英語の品詞は，Penn Treebank で定義されている品詞を用いる．10,000 文対への品詞付与の結果，日本語では 43 種類，英語では 58 種類の品詞タグが使われていた．ここで付与する日本語の品詞タグは，隠れ状態の初期値として使われ，英語の品詞タグは，シンボルに組み込む目的言語の情報として使われることを確認しておく．

次に，各対訳文に対して，単語単位の対応付けを行う．GIZA++ [32] により，日英，英日の両方向で単語単位の対応付けを行い，その結果を「grow-diag-final-and」ヒューリスティクス [19][*9]により統合する．ここでは，対応付けを精度良く行うため，NTCIR-9 の訓練データすべてを使って GIZA++を実行する．

また，単語単位の品詞推定を行うため，各日本語文に対して，単語単位の係り受け木を構築する．係り受け解析は CaboCha [20] を用いて行う．しかし，CaboCha は，英語や中国語の係り受け解析器とは異なり，単語単位ではなく文節[*10]単位の係り受け関係を解析する．そこで，内容語を主辞とする「*Cont*」と機能語を主辞とする「*Func*」の 2 つのヒューリスティクスを導入し，文節単位の係り受け木を単語単位の係り受け木に変換する．本稿では，内容語以外の単語を機能語と考えた．すなわち，句読点や記号も機能語として扱った．

Cont および *Func* は，まず，各文節の主辞になる単語を特定する．*Cont* は文節内の最後の内容語を主辞とし，*Func* は最後の機能語を主辞とする．文節内に機能語が存在しない場合は，*Func* も最後の内容語を主辞とする．そして，文節間の係り受け関係を文節の主辞間の係り受け関係とし，文節内の主辞以外の単語は主辞に依存させることで，文節単位の係り受け木を単語単位の係り受け木に変換する．

図 7 に，*Cont* および *Func* で構築した係り受け木を示す．(b) と (c) は単語間の係り受け関係を表し，(d) と (e)

[*7] すべての訓練データを使うと計算量が膨大となるため，今回は，品詞推定時には一部のみを使う．大規模なデータへの適用は今後の課題とする．なお，今回の品詞推定は，8 コアで並列処理を行い，1 週間程度かかった．

[*8] http://mecab.googlecode.com/svn/trunk/mecab/doc/index.html

[*9] Koehn ら [19] は，「diag-and」ヒューリスティクスとして説明している．

[*10] 文節とは，文を区切ったときに意味をなす最小単位で，1 つの内容語と機能語（たとえば，名詞と助詞など）で構成される．

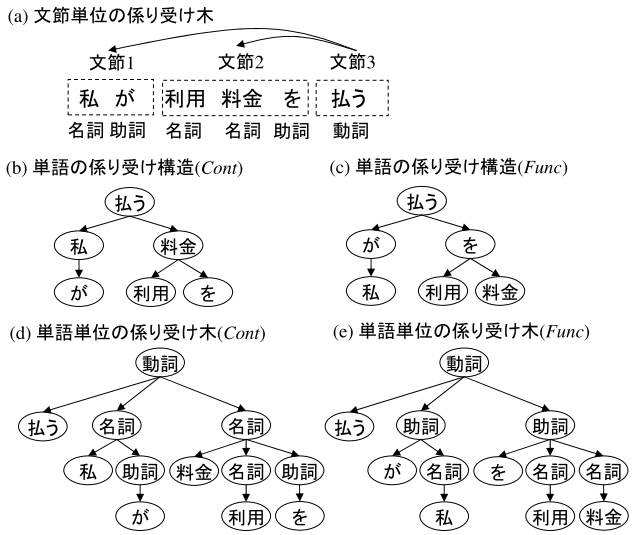

図 7 係り受け木の例
Fig. 7 Examples of dependency trees.

が以降のステップで用いる単語単位の係り受け木である．例として，*Cont* による係り受け木の構築過程を説明する．まず，文節 1，2，3 の主辞として，それぞれ，最後の内容語「私」,「料金」,「払う」を特定する．その後，文節 1 と 2 は文節 3 を修飾しているので，文節 1 と 2 の主辞「私」,「料金」の親ノードを文節 3 の主辞「払う」とする．その他の単語の親ノードは同一文節の主辞とする（たとえば，文節 1 中の単語「が」の親ノードは，文節 1 の主辞「私」とする）ことで係り受け木を構築する．*Func* による係り受け木の構築も同様である．

ステップ 2. 品詞推定

本ステップでは，3 章で提案した品詞推定手法（結合モデル，独立モデル）を用いて，日本語の各単語の品詞タグを推定する．また，比較対象として，目的言語の情報を考慮しない，従来の単言語における無限ツリーモデル [6] による推定も行う．以降，結合モデル，独立モデル，単言語における無限ツリーモデルを，それぞれ，*Joint*，*Ind*，*Mono* と簡単に表記する．

各モデルでは，Gael ら [7] にならい，一連のサンプリング（$\boldsymbol{u}$，$\boldsymbol{z}$，$\boldsymbol{\pi}$，$\boldsymbol{\beta}$，α_0，γ のサンプリング）は 10,000 回繰り返す．また，α_0 のサンプリングと γ のサンプリングで使うハイパーパラメータ α_a，α_b，γ_a，γ_b も，Gael ら [7] にならい，それぞれ，2，1，1，1 とする．また，$\boldsymbol{z}$ のサンプリングで使うパラメータ（ρ，ρ'，ρ''）は 0.01 とする．

提案モデルでは，シンボルに加える目的言語の情報として，対応する英単語の表層，対応する英単語の品詞，その両方の 3 種類を評価する（3.3 節参照）．以降，それぞれ，「s」,「P」,「s+P」で簡単に表記する．

また，品詞を推定する枠組みとして，新しい品詞体系の推定と既存の品詞タグの細分化（3.4 節参照）の 2 つを評価する．以降，それぞれ，「新品詞」,「細分化」と表記する．両枠組みとも，各隠れ状態 z_t は，まず，MeCab により特定された IPA 品詞体系の品詞タグに初期化される．その後，3.5 節で説明した各パラメータの推定を通じて状態を更新する．新品詞の枠組みでは，品詞タグを初期値としてのみ用い，学習した品詞体系は既存の品詞体系によらない（たとえば，「$1, 2, \ldots, k$」のように表される）．一方，細分化の枠組みで各 z_t をサンプリングする際は，初期化された品詞タグに由来する状態遷移確率およびシンボル出力確率の分布を用い，学習した品詞体系は既存の品詞体系に依存する（たとえば，「名詞 $_1$, 名詞 $_2$, ..., 動詞 $_1$...」のように表される）ことを確認しておく．

ステップ 3. 品詞付与および係り受け解析器の学習

本ステップでは，ステップ 2 で推定した品詞タグの付与および係り受け解析を日本語文に対して行う解析器を生成する．以降，この 2 つの解析を行う解析器を単に係り受け解析器と呼ぶ．

ステップ 2 を通じて得られる，推定した品詞タグが付与された 10,000 の日本語文の係り受け木を学習データとする．そして，Hatori ら [12] の手法[*11]により，品詞タグと係り受け関係を同時に学習し，解析時には，文の前から順に逐次，品詞タグと係り先を同時に決定する，transition-based な解析器を生成する．この係り受け解析器は，単語単位で，係り受け関係とステップ 2 で推定した品詞タグを特定することを確認しておく．

ステップ 4. forest-to-string 翻訳モデルの学習

本ステップでは，forest-to-string 翻訳モデルを学習する．forest-to-string 翻訳モデルの学習およびデコーディングは，ハイパーグラフに基づくツールキット *cicada* [*12]により行う．このツールは，機械翻訳の従来研究（たとえば，システムコンビネーション [44] やオンライン学習 [43]）で使われている実績のあるツールである．

具体的には，まず，NTCIR-9 の訓練データ中のすべての日本語文，英文に対して，ステップ 1 で記載した方法により単語分割を行う．その後，ステップ 3 で学習した係り受け解析器により，日本語文に対して，単語単位で品詞タグと係り受け関係を特定し，係り受け木を構築する．そして，Zhang ら [46] の手法により forest-to-string 翻訳モデルを学習する．

翻訳モデルの学習では，まず，各係り受け木を，全フレーズをカバーするような部分木の列（forest；森）に変換する．その後，森単位の GHKM アルゴリズム [26] により翻訳ルールを抽出する．パラメータは，ディベロップメントデータ上で，評価値を xBLEU [36] として L-BFGS [23] により調整する．この xBLEU の勾配降下法に基づく調整は，パラメータ調整で一般的に使われる MERT [31] や PRO [13] よりも安定しており，ランダムな値に陥りにくい

[*11] http://triplet.cc/software/corbit/
[*12] http://www2.nict.go.jp/univ-com/multi_trans/cicada/

ことが示されている [36]．パラメータ調整時のハイパーパラメータ（具体的には，パラメータ調整の繰返し回数）は，ディベロップメントテストデータにより決定する．

テストデータ（日本語文）を翻訳する際は，まず，MeCab により単語分割を行い，ステップ 3 で学習した係り受け解析器により単語単位の係り受け木を構築する．その後，本ステップで学習した forest-to-string 翻訳モデルでデコーディングすることで英語に翻訳する．

4.2 実験結果

4.1 節のとおり構築した翻訳システムのテストデータに対する翻訳性能を**表 1** に示す．評価尺度は，大文字と小文字を区別した BLEU [33] を用いる．また，参考のために RIBES [15] も示す．ただし，BLEU と傾向が同じであったので，以降の議論は BLEU で行う．表 1 の第 1 行目は木構造構築手法（4.1 節ステップ 1 参照），第 2 行目は品詞推定の枠組み（4.1 節ステップ 2 参照）を示す．また，第 1 列目に品詞推定手法を記す．提案手法で考慮する目的言語の情報（「s」，「P」，「s+P」）は，手法名の隣の鉤括弧中に示す．

表 1 にはベースラインとして，単言語における無限ツリーモデルで推定した品詞を用いた翻訳システム *Mono* に加えて，MeCab が付与した品詞タグ（IPA 品詞体系の 2 階層目の品詞）を用いた forest-to-string 翻訳システムの性能を *Baseline* として示す．この *Baseline* は，4.1 節のステップ 1 から 3 を行わず，ステップ 4 において，品詞タグを MeCab で，単語単位の係り受け関係を CaboCha と *Cont*（あるいは *Func*）で解析したシステムである．そして，木構造構築手法（*Cont* または *Func*）が同じ場合，ベースライン（*Baseline* および *Mono*）よりも性能が良いシステムを斜体で示し，最高性能を太字で示す．たとえば，評価指標が BLEU の場合，*Cont* を用いた場合は 25.49%，*Func* を用いた場合は 27.66%よりも性能が良いシステムを斜体で示す．また，参考のために，フレーズ単位の機械翻訳システム Moses（デフォルトの設定）の性能を調べた結果，BLEU は 26.80%，RIBES は 61.38%であった．

表 1 NTCIR-9 テストデータに対する日英翻訳性能
Table 1 Japanese-to-English translation performance on the NTCIR-9 test data.

	Cont		*Func*	
	新品詞	細分化	新品詞	細分化
Baseline	25.49		27.54	
Mono	24.96	24.67	27.66	26.83
Joint[s]	25.46	25.14	*28.00*	*28.00*
Joint[P]	24.40	24.90	26.36	26.72
Joint[s+P]	*25.73*	*25.84*	*27.99*	*27.82*
Ind[s]	*25.83*	*26.51*	*28.00*	*27.93*
Ind[P]	*26.20*	**26.79**	*28.11*	**28.63**
Ind[s+P]	*25.64*	*26.65*	*28.13*	*28.62*

評価指標：BLEU (%)

	Cont		*Func*	
	新品詞	細分化	新品詞	細分化
Baseline	66.54		68.27	
Mono	64.90	66.00	66.49	67.00
Joint[s]	*67.46*	*67.05*	*70.01*	*70.14*
Joint[P]	65.96	66.16	67.79	*68.70*
Joint[s+P]	*67.41*	*68.01*	*69.99*	*69.82*
Ind[s]	*68.52*	*69.09*	*70.54*	*70.55*
Ind[P]	*68.86*	*69.25*	*70.96*	**71.17**
Ind[s+P]	*68.21*	**69.58**	*70.13*	*70.96*

評価指標：RIBES (%)

表 1 より，提案システムは，ベースライン *Mono* よりも性能が良いことが分かる．特に，*Ind*[s+P] と *Mono* の差は，新品詞，細分化の両枠組みにおいて，*Cont*，*Func* のどちらを用いた場合も，Koehn [17] が提案したブートストラップによる検定手法により有意差水準 1%で有意差が認められた．以降，翻訳性能の有意差検定は，ブートストラップによる検定手法 [17] を用いる．この結果より，目的言語の情報を考慮することで，機械翻訳に有益な情報を品詞に反映できることが実験的に確認できる．さらに，独立モデルはベースライン *Baseline* よりも翻訳性能が良く，特に，*Func* を用いた場合，新品詞，細分化の両枠組みにおいて，それらの性能差は有意差水準 1%で有意であった．これより，目的言語の情報を考慮することにより，既存の品詞体系よりも機械翻訳に適した品詞を学習できることが分かる．

また，独立モデルの方が，結合モデルよりも機械翻訳に有効であることが分かる．特に，細分化における *Ind*[s+P] と *Joint*[s+P] の差は，*Cont*，*Func* のどちらを用いた場合も有意差水準 1%で有意であった．これは，目的言語の情報を原言語の単語に結合させたシンボルを使うと，スパースネスの問題が起こり，品詞推定に悪影響を及ぼす場合があることを示唆している．

品詞推定の枠組み（「新品詞」，「細分化」）で比較すると，*Mono* を除いて，細分化のシステムは，新品詞のシステムと同等の性能か，あるいは性能が良い．これは，既存の品詞体系で定義される原言語における違いを保持することで，翻訳により適した品詞を学習できることを示している．一方で，目的言語の情報を使わない *Mono* では，細分化の枠組みにより推定した品詞を使うと，翻訳モデルの学習において過学習の問題が生じ，低い性能になったと考えられる．さらに，IPA 品詞体系の最下層の品詞[*13]を使った *Baseline* の性能（BLEU）を評価した結果，*Cont* を用いた場合は 25.37%，*Func* を用いた場合は 27.49%となり，IPA 品詞体系の 2 階層目の品詞を使った場合よりも性能が悪かった．これは，人手で細分化した品詞体系も，翻訳モ

[*13] 評価で使用したデータには，IPA 品詞体系の最下層の品詞が 377 種類含まれていた．

デルの学習時に過学習の問題を生じさせる可能性があることを示している．以上より，品詞を細分化させ過ぎると過学習の問題を招く可能性があるが，提案手法のような目的言語の情報による細分化は翻訳モデルの構築に有効であることが実験的に確認できる．

5. 考察

5.1 IPA 品詞体系との比較

本節では，提案手法が推定した品詞タグと既存の IPA 品詞体系の品詞を比較することで，提案手法の有効性を考察する．本節では，木構造構築手法として有効であった *Func* を用いた場合に焦点をあてて議論を進める．

4.1 節のステップ 3 で学習した係り受け解析器によりテストデータに付与された品詞タグの種類数を**表 2** に示す．表 2 より，提案手法の品詞の方が，IPA 品詞体系よりも種類が多いことが分かる．これは，提案手法により推定した品詞を使うことで，より曖昧性の少ない翻訳モデルを構築できる可能性があることを示している．以降，このことを実例により確認する．

日本語の動詞は，(a) 英語でも動詞の働きをする場合もあれば，(b) 英語では名詞化する場合もある．また，(c) 英語では過去分詞や現在分詞となって他の単語を修飾する場合もある．次の 3 つの例文中の下線箇所の単語が，上記 (a) から (c) のそれぞれの例である．(a)「I *use* a card.」，(b)「*Using* the index is faster.」，(c)「I explain *using* an example」．これらの下線箇所の単語はすべて，日本語の動詞「使う」に対応する．IPA 品詞体系では，この 3 種類を「動詞」としてまとめて扱うが，細分化の枠組みによる *Ind*[s+P] は，この 3 種類を別々の品詞タグに割り当てることができた．

また，日本語の助詞「に」は，名詞と結合して，(d) 結合した名詞を副詞化する場合もあれば，(e) 結合した名詞に動詞の目的語の役割を与える場合もある．前者の例は，(d)「相互_に」であり，英語の副詞「mutually」に対応する．後者の例は，(e)「彼_に_与える」であり，英語では「give him」となる．IPA 品詞体系では，この 2 種類を「助詞」としてまとめて扱うが，細分化の枠組みによる *Ind*[s+P] は，この 2 種類を区別した品詞タグを生成できた．

これらの実例から，提案手法は，IPA 品詞体系ではまとめて扱われるような，英語で異なる働きをする品詞を区別できることが分かる．そして，提案手法の品詞を使うことで forest-to-string 翻訳システムの性能が向上したことから，それらを区別することは機械翻訳の手がかりとして有効であるといえる．

表 2 品詞タグの種類数
Table 2 The number of POS tags.

	新品詞	細分化
Joint[s+P]	164	620
Ind[s+P]	102	517
IPA 品詞体系	42	

5.2 品詞付与および係り受け解析精度の影響

提案システムの性能は，4.1 節のステップ 2 で推定する品詞の品質に加え，ステップ 3 で学習する係り受け解析器の品詞付与および係り受け解析精度にも依存している．本節では，係り受け解析器の品詞付与および係り受け解析精度について考察する．本節においても，5.1 節同様，木構造構築手法として有効であった *Func* を用いた場合に焦点をあてて議論を進める．

考察するにあたり，ステップ 3 で学習した係り受け解析器の性能を直接評価するべきであるが，推定した品詞が付与されたデータ 10,000 文は，すべて，係り受け解析器の学習データとして使用したため，評価用のテストデータを用意できない．そこで，ステップ 2 を通じて得られた 10,000 文のうち，最初の 9,000 文を訓練データ，1,000 文をテストデータとして係り受け解析器の性能評価を行った．係り受け解析器の学習方法はステップ 3 と同様である．この評価は，ステップ 3 で学習した係り受け解析器の性能を直接評価するものではないが，ステップ 2 で推定した品詞の解析しやすさを示している．

表 3 に評価結果を示す．品詞付与および係り受け解析精度の単位はパーセント（%）である．表 3 中の IPA 品詞体系とは，MeCab により付与された IPA 品詞タグ（2 階層目）と，CaboCha と *Func* により解析された係り受け関係を学習した係り受け解析器の性能であり，MeCab や CaboCha の性能ではないことを確認しておく．また，表 3 の係り受け解析精度は，CaboCha と *Func* により自動的に生成した係り受け関係を正解とした精度であり，統語的に誤りを含まない係り受け関係を正解とした評価ではないことを確認しておく．

表 3 より，提案手法で推定した品詞に対する性能（*Joint*,

表 3 NTCIR-9 訓練データ 1,000 文に対する品詞付与および係り受け解析精度
Table 3 Tagging and dependency accuracy on the 1,000 sentences in the NTCIR-9 training data.

	品詞付与		係り受け解析	
	新品詞	細分化	新品詞	細分化
IPA 品詞体系	90.37		93.62	
Mono	90.75	88.04	91.77	91.51
Joint[s]	89.08	86.73	91.55	91.14
Joint[P]	80.54	79.98	91.06	91.29
Joint[s+P]	87.56	84.92	91.31	91.10
Ind[s]	87.62	84.33	92.06	92.58
Ind[P]	90.21	88.50	92.85	93.03
Ind[s+P]	89.57	86.12	92.96	92.78

Ind）は，IPA 品詞体系や *Mono* に比べて低いことが分かる．これは，目的言語の情報も含むバイリンガルな品詞タグを，原言語の情報のみに基づいて解析することは難しいことを示す．しかし，表 3 のように品詞付与精度が相対的に低い状況でも，表 1 が示すとおり，*Joint*[P] は除くが，提案手法により推定されたバイリンガルな品詞を使う方が，原言語の情報のみの品詞（*Mono* や既存の品詞体系）を使うよりも翻訳性能が良い．これは，表 3 に示される低い解析精度というデメリットよりも，5.1 節で説明した品詞の質の改善というメリットの方が大きいことを示している．

新品詞および細分化の両枠組みにおいて，*Joint*[P] は，他の提案手法と比較して，係り受け解析精度は大差ないが品詞付与精度が極端に低い．これは，*Joint*[P] では，訓練データのシンボルに過学習した品詞が推定されたからと考えられる．この低い品詞付与精度が，*Joint*[P] の翻訳精度が他の提案手法に比べて低くなった原因の 1 つである．

5.3 *Cont* と *Func* の比較

本節では，評価で使用した 2 つの木構造構築手法（*Cont* と *Func*）の比較を行う．表 1 より，*Baseline*，*Mono* も含めたすべてのシステムで，*Func* を使う方が *Cont* を使うよりも性能が良いことが分かる．これは，日本語の場合，統語的な関係は主に機能語により示されるためである．たとえば，単語「彼」に機能語「は」が付属すると主語の機能を持つ．一方，「を」が付属すると動詞の目的語の機能を持つ．

4.1 節のステップ 1 のとおり，*Func* は，文節間の係り受け関係を機能語間の係り受け関係で置き換えるため，*Func* で生成した木構造は，機能語間の関係を直接表現する．その結果，*Func* で構築した係り受け木からは統語的な手がかりをとらえやすく，統語情報に基づく機械翻訳システムの性能が良いと考えられる．

続いて，*Func* を用いた場合に推定される品詞タグと *Cont* を用いた場合に推定される品詞タグを比較する．提案手法の代表として *Ind*[s+P] を考える．*Ind*[s+P] により細分化の枠組みで得られた品詞タグの分布を**表 4** に示す．細分化元の品詞に従って，各品詞タグが内容語の品詞か機能語の品詞かを判定した．

表 4 中の細分化された品詞の割合より，*Cont* を用いた場合，内容語の品詞の細分化に焦点があたり，*Func* を用いた場合，機能語の品詞の細分化に焦点があたることが分かる．加えて，表 1 が示すとおり，*Ind*[s+P] は *Cont* と *Func* のどちらを用いた場合も *Baseline* より性能が良いことを考えると，IPA 品詞体系の内容語と機能語に関する品詞は，ともに機械翻訳に最適とはいえず，提案モデルによって細分化することで，より適した品詞に改良できるといえる．

表 4 細分化品詞の分布
Table 4 Distribution of the refined Sub-POS tags.

	内容語の品詞	機能語の品詞
Ind[s+P]（*Cont*）	611 種類（78%）	170 種類（22%）
Ind[s+P]（*Func*）	346 種類（67%）	171 種類（33%）
IPA 品詞体系	29 種類（69%）	13 種類（31%）

6. まとめ

本稿では，統語情報に基づく機械翻訳システムの性能を改善するため，係り受け木から機械翻訳のための品詞を推定する手法を提案した．提案手法は，原言語と目的言語の両言語の情報を持つバイリンガルなシンボルに基づいて品詞を推定するノンパラメトリックなベイズ手法である．これにより，従来の品詞推定手法では推定できない，目的言語における違いを反映したバイリンガルな品詞を，原言語の各単語に付与することができる．NTCIR-9 データを用いた日英特許翻訳による評価実験を通じて，シンボルをバイリンガル化することにより，翻訳に適した品詞を推定できることを確認した．さらに，提案手法が推定した品詞を使うことにより，既存の品詞（IPA 品詞体系）よりも翻訳性能が良くなることを確認した．

本稿では，GIZA++による単語単位の自動対応付け結果に基づいて，目的言語の情報をシンボルに加えた．今後は，対応付け精度の影響を調査するとともに，対応付け誤りに頑健な手法への改良を検討したい．また，本稿では，*Cont* や *Func* のヒューリスティクスにより，文節単位の係り受け木を単語単位の係り受け木に変換した．今後は，Infinite PCFG モデル [21] による単語単位の係り受け木の構築や，Nakazawa ら [28] の syntactic-head dependency trees や semantic-head dependency trees の利用など，係り受け木構築手法の改良を検討したい．また，本稿の実験では，係り受け関係と品詞タグを同時に学習した係り受け解析器を用いた．一方で，係り受け関係とは独立に，品詞タガーのみを学習し利用する方法も考えられる．たとえば，係り受け構造を利用せず，linear-chain モデルなどで単語と品詞の並びに基づき，品詞タグを学習，解析する方法もある．このような手法の適用は今後の課題である．さらに，日英以外の言語対の翻訳への適用や，forest-to-string 翻訳システム以外の統語情報に基づく機械翻訳での有効性の確認も今後の課題である．

参考文献

[1] Beal, M.J., Ghahramani, Z. and Rasmussen, C.E.: The Infinite Hidden Markov Model, *Advances in Neural Information Processing Systems*, pp.577–584 (2001).
[2] Blunsom, P. and Cohn, T.: A Hierarchical Pitman-Yor Process HMM for Unsupervised Part of Speech Induction, *Proc. 49th Annual Meeting of the Association for Computational Linguistics* (*ACL 2011*), pp.865–874 (2011).
[3] Cohn, T. and Blunsom, P.: A Bayesian Model of Syntax-

Directed Tree to String Grammar Induction, *Proc. 2009 Conference on Empirical Methods in Natural Language Processing* (*EMNLP 2009*), pp.352–361 (2009).

[4] Ding, Y. and Palmer, M.: Machine Translation Using Probabilistic Synchronous Dependency Insertion Grammars, *Proc. 43rd Annual Meeting of the Association for Computational Linguistics* (*ACL 2005*), pp.541–548 (2005).

[5] Ferguson, T.S.: A Bayesian Analysis of Some Nonparametric Problems, *The Annals of Statistics*, Vol.1, No.2, pp.209–230 (1973).

[6] Finkel, J.R., Grenager, T. and Manning, C.D.: The Infinite Tree, *Proc. 45th Annual Meeting of the Association for Computational Linguistics* (*ACL 2007*), pp.272–279 (2007).

[7] Gael, J.V., Saatci, Y., Teh, Y.W. and Ghahramani, Z.: Beam Sampling for the Infinite Hidden Markov Model, *Proc. 25th International Conference on Machine Learning* (*ICML 2008*), pp.1088–1095 (2008).

[8] Gael, J.V., Vlachos, A. and Ghahramani, Z.: The infinite HMM for unsupervised PoS tagging, *Proc. 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2 - Volume 2* (*EMNLP 2009*), pp.678–687 (2009).

[9] Galley, M., Graehl, J., Knight, K., Marcu, D., DeNeefe, S., Wang, W. and Thayer, I.: Scalable Inference and Training of Context-Rich Syntactic Translation Models, *Proc. 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics* (*COLING-ACL 2006*), pp.961–968 (2006).

[10] Gao, J. and Johnson, M.: A Comparison of Bayesian Estimators for Unsupervised Hidden Markov Model POS Taggers, *Proc. 2008 Conference on Empirical Methods in Natural Language Processing* (*EMNLP 2008*), pp.344–352 (2008).

[11] Goto, I., Lu, B., Chow, K.P., Sumita, E. and Tsou, B.K.: Overview of the Patent Machine Translation Task at the NTCIR-9 Workshop, *Proc. 9th NTCIR Workshop*, pp.559–578 (2011).

[12] Hatori, J., Matsuzaki, T., Miyao, Y. and Tsujii, J.: Incremental Joint POS Tagging and Dependency Parsing in Chinese, *Proc. 5th International Joint Conference on Natural Language Processing* (*IJCNLP 2011*), pp.1216–1224 (2011).

[13] Hopkins, M. and May, J.: Tuning as Ranking, *Proc. 2011 Conference on Empirical Methods in Natural Language Processing* (*EMNLP 2011*), pp.1352–1362 (2011).

[14] Huang, L., Knight, K. and Joshi, A.: A Syntax-Directed Translator with Extended Domain of Locality, *Proc. Workshop on Computationally Hard Problems and Joint Inference in Speech and Language Processing*, pp.1–8 (2006).

[15] Isozaki, H., Hirao, T., Duh, K., Sudoh, K. and Tsukada, H.: Automatic Evaluation of Translation Quality for Distant Language Pairs, *Proc. 2010 Conference on Empirical Methods in Natural Language Processing* (*EMNLP 2010*), pp.944–952 (2010).

[16] Johnson, M.: Why doesn't EM find good HMM POS-taggers?, *Proc. 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (*EMNLP-CoNLL 2007*), pp.296–305 (2007).

[17] Koehn, P.: Statistical Significance Tests for Machine Translation Evaluation, *Proc. 2004 Conference on Empirical Methods in Natural Language Processing* (*EMNLP 2004*), pp.388–395 (2004).

[18] Koehn, P. and Hoang, H.: Factored Translation Models, *Proc. 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (*EMNLP-CoNLL*), pp.868–876 (2007).

[19] Koehn, P., Och, F.J. and Marcu, D.: Statistical Phrase-Based Translation, *Proc. 2003 Human Language Technology Conference: North American Chapter of the Association for Computational Linguistics* (*HLT-NAACL 2003*), pp.48–54 (2003).

[20] Kudo, T. and Matsumoto, Y.: Japanese Dependency Analysis using Cascaded Chunking, *Proc. 6th Conference on Natural Language Learning* (*CoNLL 2002*), pp.63–69 (2002).

[21] Liang, P., Petrov, S., Jordan, M.I. and Klein, D.: The Infinite PCFG using Hierarchical Dirichlet Processes, *Proc. 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (*EMNLP-CoNLL 2007*), pp.688–697 (2007).

[22] Lin, D.: A Path-based Transfer Model for Machine Translation, *Proc. 20th International Conference on Computational Linguistics* (*COLING 2004*), pp.625–630 (2004).

[23] Liu, D.C. and Nocedal, J.: On the limited memory BFGS method for large scale optimization, *Mathematical Programming B*, Vol.45, No.3, pp.503–528 (1989).

[24] Liu, Y., Liu, Q. and Lin, S.: Tree-to-String Alignment Template for Statistical Machine Translation, *Proc. 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics* (*COLING-ACL 2006*), pp.609–616 (2006).

[25] Liu, Y., Lü, Y. and Liu, Q.: Improving Tree-to-Tree Translation with Packed Forests, *Proc. 47th Annual Meeting of the Association for Computational Linguistics and 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing* (*ACL-IJCNLP 2009*), pp.558–566 (2009).

[26] Mi, H. and Huang, L.: Forest-based Translation Rule Extraction, *Proc. 2008 Conference on Empirical Methods in Natural Language Processing* (*EMNLP 2008*), pp.206–214 (2008).

[27] Mi, H. and Liu, Q.: Constituency to Dependency Translation with Forests, *Proc. 48th Annual Conference of the Association for Computational Linguistics* (*ACL 2010*), pp.1433–1442 (2010).

[28] Nakazawa, T. and Kurohashi, S.: Alignment by Bilingual Generation and Monolingual Derivation, *Proc. 24th International Conference on Computational Linguistics* (*COLING 2012*), pp.1963–1978 (2012).

[29] Neal, R.M.: Slice Sampling, *Annals of Statistics*, Vol.31, pp.705–767 (2003).

[30] Nguyen, T., Vogel, S. and Smith, N.A.: Nonparametric Word Segmentation for Machine Translation, *Proc. 23rd International Conference on Computational Linguistics* (*COLING 2010*), pp.815–823 (2010).

[31] Och, F.J.: Minimum Error Rate Training in Statistical Machine Translation, *Proc. 41st Annual Meeting of the Association for Computational Linguistics* (*ACL 2003*), pp.160–167 (2003).

[32] Och, F.J. and Ney, H.: A Systematic Comparison of Var-

ious Statistical Alignment Models, *Computational Linguistics*, Vol.29, pp.19–51 (2003).
[33] 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 2002*), pp.311–318 (2002).
[34] Petrov, S., Barrett, L., Thibaux, R. and Klein, D.: Learning Accurate, Compact, and Interpretable Tree Annotation, *Proc. 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics* (*COLING-ACL 2006*), pp.433–440 (2006).
[35] Quirk, C., Menezes, A. and Cherry, C.: Dependency Treelet Translation: Syntactically Informed Phrasal SMT, *Proc. 43rd Annual Conference of the Association for Computational Linguistics* (*ACL 2005*), pp.271–279 (2005).
[36] Rosti, A.-V., Zhang, B., Matsoukas, S. and Schwartz, R.: Expected BLEU Training for Graphs: BBN System Description for WMT11 System Combination Task, *Proc. 6th Workshop on Statistical Machine Translation*, pp.159–165 (2011).
[37] Schmid, H.: Probabilistic Part-of-Speech Tagging Using Decision Trees, *Proc. International Conference on New Methods in Natural Language Processing* (*NeMLaP 1994*), pp.44–49 (1994).
[38] Sethuraman, J.: A Constructive Definition of Dirichlet Priors, *Statistica Sinica*, Vol.4, No.2, pp.639–650 (1994).
[39] Shen, L., Xu, J. and Weischedel, R.: A New String-to-Dependency Machine Translation Algorithm with a Target Dependency Language Model, *Proc. 46th Annual Conference of the Association for Computational Linguistics: Human Language Technologies* (*ACL-HLT 2008*), pp.577–585 (2008).
[40] Sirts, K. and Alumäe, T.: A Hierarchical Dirichlet Process Model for Joint Part-of-Speech and Morphology Induction, *Proc. 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (*NAACL-HLT 2012*), pp.407–416 (2012).
[41] Teh, Y.W., Jordan, M.I., Beal, M.J. and Blei, D.M.: Hierarchical Dirichlet Processes, *Journal of the American Statistical Association*, Vol.101, No.476, pp.1566–1581 (2006).
[42] Wang, W., May, J., Knight, K. and Marcu, D.: Restructuring, Re-labeling, and Re-aligning for Syntax-based Machine Translation, *Computational Linguistics*, Vol.36, No.2, pp.247–277 (2010).
[43] Watanabe, T.: Optimized Online Rank Learning for Machine Translation, *Proc. 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (*NAACL-HLT 2012*), pp.253–262 (2012).
[44] Watanabe, T. and Sumita, E.: Machine Translation System Combination by Confusion Forest, *Proc. 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies* (*ACL-HLT 2011*), pp.1249–1257 (2011).
[45] Xu, J., Gao, J., Toutanova, K. and Ney, H.: Bayesian Semi-Supervised Chinese Word Segmentation for Statistical Machine Translation, *Proc. 22nd International Conference on Computational Linguistics* (*COLING 2008*), pp.1017–1024 (2008).
[46] Zhang, H., Fang, L., Xu, P. and Wu, X.: Binarized Forest to String Translation, *Proc. 49th Annual Meeting of the Association for Computational Linguistics* (*ACL 2011*), pp.19–24 (2011).
[47] Zhang, M., Jiang, H., Aw, A., Li, H., Tan, C.L. and Li, S.: A Tree Sequence Alignment-based Tree-to-Tree Translation Model, *Proc. 46th Annual Conference of the Association for Computational Linguistics: Human Language Technologies* (*ACL-HLT 2008*), pp.559–567 (2008).
[48] 須藤克仁，進藤裕之，塚田 元，永田昌明：統計翻訳における統語的ラベル細分化の検討，言語処理学会第 19 回年次大会発表論文集，pp.390–393 (2013).

田村 晃裕 （正会員）

1981 年生．2005 年東京工業大学工学部情報工学科卒業．2007 年同大学大学院総合理工学研究科修士課程修了．2007 年から 2011 年まで日本電気株式会社で自然言語処理，特にテキストマイニングに関する研究に従事．2011 年から 2014 年まで独立行政法人情報通信研究機構で統計的機械翻訳に関する研究に従事．2013 年東京工業大学大学院総合理工学研究科博士課程修了．2014 年から日本電気株式会社の研究員，現在に至る．工学博士．ACL 会員．

渡辺 太郎 （正会員）

1994 年京都大学工学部情報工学科卒業．1997 年同大学大学院工学研究科情報工学専攻修士課程修了．2000 年 Language and Information Technologies, School of Computer Science, Carnegie Mellon University, Master of Science 取得．2004 年京都大学博士（情報学）．ATR および NTT で研究員として勤めた後，現在，情報通信研究機構主任研究員．言語処理や機械学習，特に統計的機械翻訳の研究に従事．

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 年京都大学博士（工学）．現在，情報通信研究機構 MASTAR プロジェクトリーダー，多言語翻訳研究室室長，神戸大学大学院工学研究科客員教授．機械翻訳，e ラーニングを研究．NLP，ASJ，ACL，IEEE 各会員．

高村 大也 （正会員）

1997 年東京大学工学部計数工学科卒業．2000 年同大学大学院工学系研究科計数工学専攻修了（1999 年はオーストリアウィーン工科大学で研究）．2003 年奈良先端科学技術大学院大学情報科学研究科博士課程修了．博士（工学）．2003 年から 2010 年まで東京工業大学精密工学研究所助教．2006 年にはイリノイ大学で客員研究員．2010 年より同准教授．計算言語学，自然言語処理を専門とし，特に機械学習の応用に興味を持つ．言語処理学会，人工知能学会，ACL 各会員．

奥村 学 （正会員）

1962 年生．1984 年東京工業大学工学部情報工学科卒業．1989 年同大学大学院博士課程修了．同年，東京工業大学工学部情報工学科助手．1992 年北陸先端科学技術大学院大学情報科学研究科助教授，2000 年東京工業大学精密工学研究所助教授，2009 年同教授，現在に至る．工学博士．自然言語処理，知的情報提示技術，語学学習支援，テキスト評価分析，テキストマイニングに関する研究に従事．電子情報通信学会，人工知能学会，AAAI，言語処理学会，ACL，認知科学会，計量国語学会各会員．http://oku-gw.pi.titech.ac.jp/~oku/.