論文 / 著書情報
Article / Book Information

題目(和文)	特許翻訳のためのバイリンガル知識の獲得に関する研究
Title(English)	
著者(和文)	田村晃裕
Author(English)	Akihiro Tamura
出典(和文)	学位:博士(工学), 学位授与機関:東京工業大学, 報告番号:甲第9327号, 授与年月日:2013年9月25日, 学位の種別:課程博士, 審査員:奥村 学,小林 隆夫,住田 一男,熊澤 逸夫,篠崎 隆宏,高村 大也
Citation(English)	Degree:Doctor (Engineering), Conferring organization: Tokyo Institute of Technology, Report number:甲第9327号, Conferred date:2013/9/25, Degree Type:Course doctor, Examiner:,,,,,
学位種別(和文)	博士論文
Type(English)	Doctoral Thesis

特許翻訳のための
バイリンガル知識の獲得に関する研究

東京工業大学大学院
総合理工学研究科
物理情報システム専攻
指導教員　奥村 学 教授

田 村　晃 裕

2013年9月

目 次

図 目 次

表 目 次

第1章

序論

1.1 研究の背景

近年，インターネットの発達により，外国語による文書や動画などに容易にアクセス可能となった．また，企業のグローバル化に伴い，仕事の上でも外国語と接する機会が増加している．このような背景の下，言語の壁を超えるため，翻訳を自動的に行う機械翻訳の需要が高まっている．そして，機械翻訳に関する様々な研究が行われ，多くの機械翻訳サービスが提供されている．例えば，機械翻訳サービスとして,「Google 翻訳[1]」や「Yahoo!翻訳[2]」といった，テキストやウェブページのオンライン翻訳サービスや,「VoiceTra+[3]」や「はなして翻訳[4]」といった，スマートフォンやタブレット向けの翻訳アプリケーションなどが提供されている．

これらの機械翻訳は，人手により定められた翻訳規則や対訳辞書により翻訳を行う「ルールベース機械翻訳」と大量の対訳データから自動的に学習した統計量を用いて翻訳を行う「統計的機械翻訳」に大別できる．ルールベース機械翻訳は，翻訳規則を作成するコストが高いという大きなデメリットがある．一方で，統計的機械翻訳は，対訳データさえあれば自動的にシステムを構築できる．したがって，本研究では「統計的機械翻訳」に焦点をあて，以降の議論を進める．

前述のとおり，機械翻訳に関する数多のサービス，研究が行われているが，一般的な自然言語処理同様，機械翻訳においても任意の分野に対応することは難しい．そのため，サービスの大半は，翻訳対象の分野を限定し，使用する語彙やそれらの使われ方のバリエーションを狭めることで，高い翻訳性能を実現している．例えば，前述した「VoiceTra+」や「は

[1] http://translate.google.co.jp/
[2] http://honyaku.yahoo.co.jp/
[3] http://voicetra-plus.jp/
[4] http://www.nttdocomo.co.jp/service/communication/hanashite_honyaku/

なして翻訳」は，翻訳対象を旅行会話・日常会話に限定することで，実用的な翻訳性能を達成している．また，学術的な研究においても，研究課題を明確化するために，翻訳対象の分野を限定して研究が行われている．

翻訳対象として，旅行会話，レシピ，アパレル等，様々な分野が扱われているが，その中で注目を集めている分野の一つに特許がある．前述のとおり，グローバル化に伴い，海外に進出する企業が増加している昨今，日本国内のみならず進出先の国においても知的財産を守る必要性が高まっている．そのため，日本語以外の特許調査や特許出願を効率良く行う手助けとなる，特許の機械翻訳（特許翻訳）が注目を集めている．例えば，日本特許情報機構（Japio）は，アメリカ，中国など五カ国の特許と PCT 出願[5]された特許を機械翻訳により翻訳し，言語を統一したデータベースを作成することにより，それらの特許を日本語又は英語で横断的に検索できる世界特許情報検索サービス[6]を行っている．

また，このようなビジネス的なニーズに加えて，特許データは公開・電子化されており，利用がしやすいこともあり，学術的にも研究が盛んな分野でもある．例えば，NTCIR プロジェクトにおいて，2007 年 NTCIR-7 から，Patent Translation Task[20] という特許翻訳に関する評価型のワークショップが毎年開催されている．そこで，本研究では特許翻訳に着目する．

特許翻訳には，特許の特徴に起因する難しさがある．本節の以降では，二つの特許の特徴に基づいて，特許翻訳の現状を説明する．

まず，特許の特徴の一つに，専門用語，造語が多いという特徴がある．一般的に，統計的機械翻訳は，確率付き対訳辞書の働きをもつ翻訳モデルと，翻訳先の言語としての自然さを評価する言語モデルに基づいて翻訳を行う．図 1.1 に，日本語から英語への翻訳モデルの一例を示す．以降，翻訳元の言語を「原言語」，翻訳先の言語を「目的言語」と呼ぶ．図 1.1 では，日本語が原言語，英語が目的言語である．図 1.1 の句単位の翻訳モデルは，日本語「図 4 は」が確率 0.4 で英語「Figure 4」に，確率 0.6 で英語「Fig. 4」に翻訳されることを示している．この機械翻訳は，翻訳モデルに含まれていない単語（以降,「未知語」と呼ぶ）は翻訳できない．そして，専門用語，造語は，一般的に使われる単語ではないため，未知語になる可能性が高い．以上をまとめると，特許翻訳では，専門用語，造語由来の未知語が多く出現し，性能が低い原因となっている．

二つ目の特許の特徴として，重文や複文が多い上，修飾句も多用されるため，長く複雑な構造を持つ文が多いという特徴がある．複雑で長い文の翻訳には，近年，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 [55, 14, 74, 82, 62] や句構造解析

[5]PCT 出願とは，PCT（特許協力条約）に基づいて自国の特許庁で手続きを行うことにより，PCT の各締約国に出願したのと同じ効果を与える出願制度である．

[6]http://www.japio.or.jp/service/service05.html

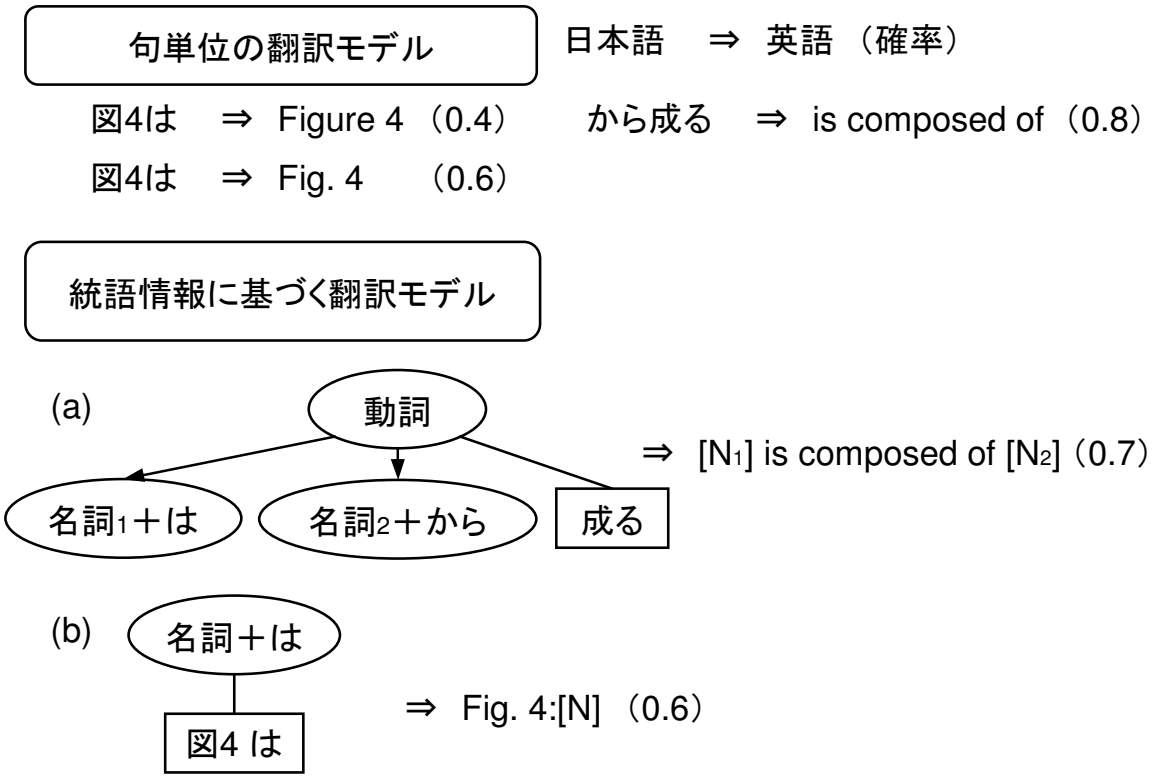

図 1.1: 翻訳モデルの例

[40, 58, 28, 61, 92, 9, 59, 62, 91] の結果を利用する，統語情報に基づく機械翻訳が有望視されている．図 1.1 に，一般的な機械翻訳の翻訳モデル（句単位の翻訳モデル）と，統語情報に基づく機械翻訳の翻訳モデルを示す．図 1.1 の統語情報に基づく機械翻訳は，原言語の構文解析結果を用いている．日本語の構文解析では，係り受け解析が行われるのが一般的であり，図 1.1 においても，矢印で示されるノード間の依存関係は，係り受け関係である．図 1.1 に示されるとおり，統語情報に基づく機械翻訳は，構文解析の結果得られる統語的関係（図 1.1 の場合，係り受け関係）で定義されるモデルを用いる．

例として，日本語文「図 4 は断面が梯形となるスリットを規則的に開口するウェッジワイヤ２７と，これらを束ねて指示するサポートロッド２８から成る.」を翻訳する場合を考える．この例文では,「図 4 は」の翻訳「Fig. 4」の後に続く動詞の候補は,「なる」,「開口する」,「指示する」,「成る」など複数存在する．これら複数の候補の中で,「成る」の翻訳が次に来るべきである．しかし，句単位の機械翻訳の場合，原言語と目的言語で語順を大きく変える翻訳は難しいことが知られている [34]．そのため,「図 4 は」と距離の離れた候補「成る」の翻訳を次に選ぶのは難しい．一方で，統語情報に基づく機械翻訳は,「図 4 は」が

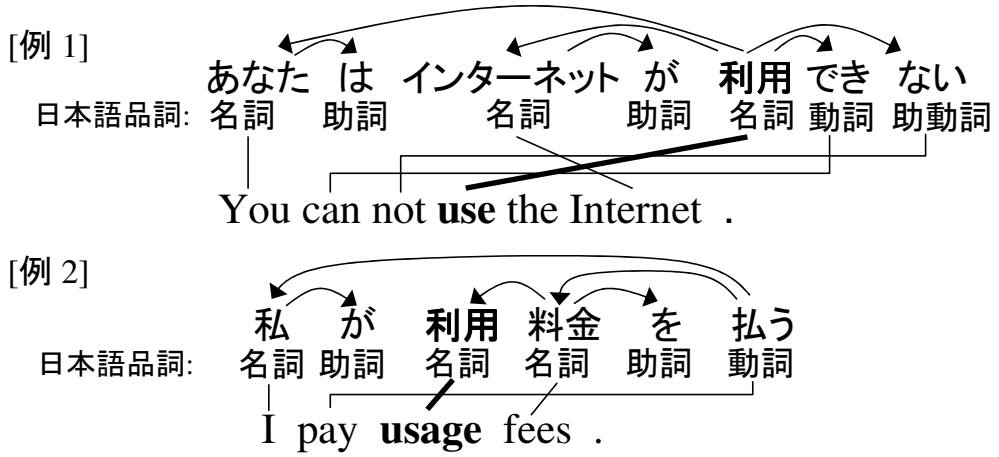

図 1.2: 既存の日本語品詞と係り受け構造の例

「成る」に係るという係り受け関係を利用する（図 1.1 のルール (a) と (b) を適用する）ことで,「Fig. 4」の次に「成る」の翻訳「is composed of」を位置づけることができる．このように，統語情報に基づく機械翻訳は，翻訳先を決める際に文の構造を利用できるため，複雑で長い文が多い特許翻訳に有効である．

統語情報に基づく機械翻訳の翻訳モデルでは，図 1.1 のとおり，統語的カテゴリとして品詞の情報が使われるのが一般的である．しかし，どのような品詞が翻訳に最適かは明らかになっていない．例えば，品詞は階層構造[7]で定義されているが，どの粒度の品詞が翻訳に最適かは明らかになっていない．さらに，既存の品詞体系は，品詞付与対象の言語を分析して構築されたものであり，翻訳時に原言語又は目的言語となる翻訳相手の言語のことは考慮されていない．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 1.2 を例に説明する．図 1.2 は，日本語と英語における単語単位の対応関係，既存の日本語品詞及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，4.2.1 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 1.2 の例 1 では，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場合もある．このような異なる振る舞いは，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体系になっておらず，日英翻訳に最適とは言い切れない．

[7]例えば,「名詞」の下の階層に「一般名詞」や「固有名詞」などの細分化された品詞が定義されている．

1.2 本研究の目的

1.1 節で述べた背景を踏まえて，本研究では，特許翻訳の性能向上を最終目標とする．この目標を実現するためには，1.1 節で説明した，特許翻訳で影響の大きい問題を解決する必要がある．すなわち，(1) 特許には専門用語や造語が多いため，特許翻訳では未知語が多い，(2) 複雑で長い文が多い特許翻訳では統語情報に基づく機械翻訳が有効だが，そこで使われる品詞が翻訳に適しているとは限らないという 2 つの問題である．本研究では，(1) 翻訳対，(2) 翻訳に適した品詞の 2 つの知識をコーパスから獲得することで，これらの問題解決を図る．

まず，一つ目の知識「翻訳対」を獲得するため，コーパスから翻訳対を抽出する手法を提案する．この提案手法で未知語に関する翻訳対を抽出することにより，未知語の削減を試みる．2.1 節で後述するが，これまでもコーパスからの翻訳対抽出手法は数多く提案されている．本研究では，従来手法より精度の高い手法を提案することで，その分多くの未知語を削減することを目的とする．

次に，二つ目の知識「翻訳に適した品詞」を獲得するため，コーパスから翻訳のための品詞を導出する手法を提案する．2.2 節で後述するが，これまでもコーパスからの品詞導出手法は数多く提案されている．しかし，それらは翻訳を想定した品詞を導出するものではない．そこで，提案手法により翻訳のための品詞を導出し，その導出した品詞を統語情報に基づく機械翻訳の中で使うことで，特許翻訳の性能を向上させることを目的とする．

1.3 本論文の構成

本章以降の構成を示す．

2 章では，本研究の関連研究を述べる．まず最初に，2.1 節で，コーパスからの翻訳対抽出手法に関する従来手法を紹介する．その中で，本研究では，コンパラブルコーパスと呼ばれるコーパスからの抽出に着目することを述べる．同時に，従来のコンパラブルコーパスからの翻訳対抽出手法は，翻訳関係を特定する際に使う既存の翻訳対が小規模だと性能が悪いという問題があることを説明する．その後，2.2 節で，従来のコーパスからの品詞導出手法を紹介し，従来手法は，翻訳を想定した品詞導出ではないという問題があることを説明する．

3 章では，2.1 節で説明する従来手法の問題を緩和した，コンパラブルコーパスからの翻訳対抽出手法を提案する．提案手法は，従来手法では捉えられない，単語間の間接的な関係を考慮することで性能改善を行う．そして，特許コーパスを用いた評価を通じて，提案手法の効果を示し，提案手法は従来手法よりも翻訳対の抽出精度が高いことを示す．

4 章では，コーパスから翻訳のための品詞を導出する手法を提案する．そして，導出した品詞を用いた特許翻訳の評価を通じて，提案手法で導出した品詞は，従来手法で導出した品詞や既存の品詞よりも翻訳に有効であり，特許翻訳の性能を改善できることを示す．

最後に，5 章において，本研究のまとめと今後の課題について述べる．

第2章

関連研究

本章では，コーパスからの翻訳対抽出手法と品詞導出手法に関する従来研究について述べ，その問題点を説明する．まず，2.1 節で，翻訳対抽出手法に関する研究について述べ，2.2 節で，品詞導出手法に関する研究について述べる．最後に，2.3 節で本章のまとめを行う．

2.1 翻訳対抽出

翻訳対（例えば，対訳辞書）は，機械翻訳だけではなく，多言語横断検索等の多言語が関わるタスクにおいて重要なリソースである．それゆえ，これまで多くの研究者が，コーパスからの翻訳対獲得を試みてきた．その一つの流れに，パラレルコーパスから翻訳対を抽出する研究がある [89, 23, 69]．パラレルコーパスとは，文単位で対応関係がある対訳文の集合である．しかし，パラレルコーパスの作成には人手を要するため，その作成コストは膨大となる．そのため，現在，大規模なパラレルコーパスを利用できるのは特定の言語対のみであり，加えて特定の分野に限られている．

この問題の解決策として有望視されているのが，コンパラブルコーパスからの翻訳対抽出である．コンパラブルコーパスとは，直接の対訳関係はないが，トピックや分野を共有する，言語が異なる文書集合である[1]．このコンパラブルコーパスは，複数言語で同一分野の文書を集めるだけで作成できるため，多言語間，多分野において大規模なコーパスが比較的容易に構築できる．

前述の 2 種類のアプローチ以外に，ウェブを活用する手法も提案されている．Lu ら [60] は，ウェブ文書中のアンカーテキストとリンク構造を解析することにより，翻訳対を発見

[1] Vulić ら [86] は，ウィキペディアの文書のような文書単位で対応関係のある文書集合をコンパラブルコーパスと考えている．しかし，本論文では，この狭義の定義ではなく，文書単位でも対応関係のない文書集合をコンパラブルコーパスと考える．

している．しかし，この手法は，リンクで結ばれた翻訳対しか獲得できないという欠点がある．また，複数言語が混在する文書から翻訳対を獲得するアプローチも提案されている．この種の手法は，まず，ウェブ検索エンジンや単純なルールを用いて，複数言語が混在する文書を抽出する．その後，抽出した文書を分析することにより翻訳対を獲得する．文書を分析する際は，様々な手がかりが使われている．例えば，Zhang ら [93] は，文書内での単語間の共起情報を基に翻訳関係を特定している．Cheng ら [7] は，共起情報に加えて文脈の類似情報を用いている．また，Huang ら [39] は，音訳情報，出現位置，フレーズ単位での意味の一致度に基づき，フレーズ単位の翻訳対を抽出している．さらに，括弧で出現する翻訳対（例えば,「インターネット（Internet）」）をマイニングする手法も提案されている [56]．しかし，これらの手法は，同一文書内に共起する翻訳対しか抽出できないという欠点がある．

また，コーパスからの翻訳対獲得とは違うアプローチとして，トランスリタレーションがある．トランスリタレーションとは，ある言語の単語を，音声が等価な他の言語に変換する技術である [46, 45]．例えば，英語を日本語のカタカナで表現することである．このトランスリタレーションの技術を用いることで，音声が等価な翻訳対の獲得が可能である．しかし，このアプローチは，固有名詞や外来語に対しては有効であるが，音声的に異なる翻訳対は獲得できないという欠点がある．

以上より，本研究では，手法の適用範囲，有効範囲が最も広い，コンパラブルコーパスからの翻訳対抽出に着目する．2.1.1 節では，従来のコンパラブルコーパスからの翻訳対抽出手法とその問題点を説明し，3 章では，従来手法の問題を緩和する，新たなコンパラブルコーパスからの翻訳対抽出手法を提案する．

2.1.1 コンパラブルコーパスからの翻訳対抽出

コンパラブルコーパスから翻訳対を獲得する試みは，Rapp ら [75] や Fung[22] によって最初に行われ，現在まで数多くの手法が提案されてきている．そのほとんどは，次の仮定 (I) に基づいている．**仮定 (I)**：**翻訳関係にある単語は，言語を超えて，似た文脈で出現する傾向がある [76]**．そして，この仮定 (I) に基づき，文脈間の類似度（文脈類似度）が高い単語のペアを翻訳対として抽出する．このアプローチによる手法を「文脈類似度に基づく手法」と呼ぶ．

本節の以降では，この最も標準的なアプローチである，文脈類似度に基づく手法の詳細を説明し，その問題点を述べる．

文脈類似度に基づく手法

文脈類似度に基づく手法は，前述の仮定 (I) に基づき，文脈類似度が高い単語のペアを翻訳対として抽出する手法である．このアプローチによる手法は，次の 3 つの共通ステップからなる．(1) 文脈のモデル化，(2) 文脈類似度の計算，(3) 翻訳対の抽出である．以降，各ステップについて説明する．

ステップ 1．文脈のモデル化

本ステップでは，コンパラブルコーパス中の各単語の文脈をモデル化する．一般的には，文脈ベクトルと呼ばれるベクトルでモデル化する．具体的には，各単語に対して，文脈内に出現する単語（以降,「文脈単語」と呼ぶ）を特定し，同時に，各文脈単語との文脈における共起度合い（以降,「文脈共起度」と呼ぶ）を計算する．そして，各次元が文脈単語に相当し，その重みが文脈共起度であるベクトルで，各単語の文脈をモデル化する．

文脈単語の定義は様々なものが使われている．Laroche ら [52] は同一文内の単語，Fung ら [24] は同一パラグラフ内の単語，Rapp[76] や Andrade ら [4] は予め定めたサイズの窓内の単語を文脈単語として用いている．Garera ら [30] は，係り受け木において祖先子孫関係となる単語を文脈単語とし，Otero[70] らは，予め定めた統語的な構造に基づくパタンにマッチする単語を文脈単語として用いている．また，文脈における出現位置毎に区別して文脈単語を求める手法 [76] もあれば，出現位置を考慮しない手法 [25] もある．文脈中での出現位置で区別する手法は，同じ単語であっても，出現位置が異なれば別々のものとして扱う．例えば，1 単語前と 2 単語前の同じ単語を別々に扱う．

さらに，文脈共起度の計算には様々な尺度が使われている．例えば，Rapp[76] や Chiao ら [8] は対数尤度比，Fung ら [25] は tf-idf，Andrade ら [4] は自己相互情報量（PMI），Fung[22] は文脈の不均質性を示す尺度を用いている．

前述の一般的なモデル化とは別のモデル化も多数存在する．Ismail ら [41] は，文脈内の単語のうち，特定ドメインに属する単語のみを文脈単語として使っている．Andrade ら [2] は，文脈の構成要素として，単語ではなく係り受け関係を用いている．Pekar ら [72] は，データスパースネスに対応するため，コーパスにおける頻度をスムージングした後，文脈ベクトルを作成している．Andrade ら [4] は，出現する文書に正の相関がある単語集合を文脈として用いている．また，Laws ら [53] は，全単語の文脈関係を，単語が頂点，予め定めた 3 種類の統語的関係（形容詞的修飾など）となる頂点間が辺で結ばれたグラフで表現している．Shao ら [81] は，文脈（ある単語列）が与えられたときに，ある単語が出現する確率を与える確率モデル（言語モデル）をコンパラブルコーパスから構築することで，文脈をモデル化している．

ステップ 2. 文脈類似度の計算

本ステップでは，異言語の単語間で，ステップ 1 でモデル化した文脈同士の類似度を計算する．ステップ 1 では，異言語の単語の文脈は，異なる言語でモデル化される．そこで，類似度を計算する際には，既存の翻訳対を用いて，これらの文脈を同一空間に写像する．以降，この写像に用いる既存の翻訳対を「シード翻訳対」と呼ぶ．その後，同一空間で表現された文脈に対して，文脈類似度を計算する．

類似度計算には様々な尺度が用いられている．例えば，Rapp[76] はマンハッタン距離，Fung ら [25] はコサイン類似度，Hazem ら [37] は重み付き Jaccard 指数，Pekar ら [72] はジェンセン・シャノン・ダイバージェンス，Andrade ら [4] は重複する文脈単語の数，Laws ら [53] は SimRank，Fung[22] はユークリッド距離を用いている．

また，Andrade ら [3] は，文脈内の位置による重要度を考慮するため，文脈ベクトルを線形変換した後，類似度を計算している．Gaussier ら [31] は，シード翻訳対の同義語や多義性を捉えるために，シード翻訳対の潜在クラスを考慮して，文脈ベクトルの次元間の対応付けを行っている．Fišer ら [19] や Kaji[44] は，非対称の類似度尺度を用いて，両言語方向からの類似度を計算した後，統合している．

ステップ 3. 翻訳対の抽出

本ステップでは，ステップ 2 で算出した文脈類似度が高い単語のペアを，翻訳対として特定する．

文脈類似度以外の情報も考慮して翻訳対を特定する手法も提案されている．例えば，Déjean ら [13] は，複数のシソーラスから獲得した，単語の概念を表すクラスも手がかりに翻訳対を特定している．また，Prochasson ら [73] は，パラレル文書における共起情報，Shao ら [81] は，音素における類似度を考慮している．

従来手法の問題点

本節の以降では，従来手法の問題点を説明する．

前述のとおり，文脈類似度に基づく手法は，一般的に，文脈類似度を計算する際，シード翻訳対を用いて文脈ベクトルを同一空間に写像する．この写像において，シード翻訳対で表現できない情報は，同一空間に写像できないため失われる．したがって，シード翻訳対が小規模である場合，写像後の文脈ベクトルは，スパースで識別能力が低いベクトルとなる．そして，その結果，誤った翻訳対が抽出されやすくなる．

この問題点について，図 2.1 を用いて具体的に説明する．図 2.1 は，文脈類似度に基づく手法及び提案手法で，日本語単語「ピラニア」と翻訳関係にある英単語を抽出する処理を表す．シード翻訳対として,「アマゾン-Amazon」,「ジャングル-jungle」,「魚-fish」の 3 つの翻訳対を使う場合を考える．

文脈類似度に基づく手法は，まず最初に，各単語（以降,「クエリ」と呼ぶ）に対する文脈ベクトルを生成する．最上部の 2 つのテーブルは，クエリとその文脈単語との文脈共起度を示す．この文脈共起度に基づき，日本語単語「ピラニア」の文脈ベクトルは,「$F_{アマゾン} = 0.8$, $F_{ジャングル} = 0.6$，$F_{淡水} = 0.5$」となる．ここで，文脈単語 x に対応する次元の値が a であるとき,「$F_x = a$」と記す．同様に，英単語「piranha」と「anaconda」の文脈ベクトルは，それぞれ,「$F_{Amazon} = 0.8, F_{jungle} = 0.6, F_{freshwater} = 0.5$」,「$F_{Amazon} = 0.8, F_{jungle} = 0.6$」となる．

次に，これらの文脈ベクトル間の類似度を計算する．その際，文脈ベクトルは，シード翻訳対に基づいて同一空間に写像される.「ピラニア」の文脈ベクトルの写像を考えると，シード翻訳対は「淡水」を含んでいないため，写像後は「淡水」との共起情報が失われる．同様に,「piranha」の場合も，写像を通じて「freshwater」との共起情報が失われる．その結果，写像後は,「ピラニア」,「piranha」,「anaconda」の 3 単語の文脈ベクトルは，全て同じベクトル「$F_{アマゾン-Amazon} = 0.8$，$F_{ジャングル-jungle} = 0.6$」となる．そして，この写像後の文脈ベクトル間の類似度を計算すると,「ピラニア」と「piranha」の文脈類似度と「ピラニア」と「anaconda」の文脈類似度は共に 1 となり,「ピラニア」と「anaconda」の単語対が翻訳対として誤って抽出される．

前述した，シード翻訳対が小規模の場合に性能が悪化するという問題を緩和する試みもされている．Morin ら [64] は，パラレルコーパスから抽出した翻訳対をシード翻訳対に追加することで，問題を緩和している．しかし，この手法はパラレルコーパスを必要とするため，2.1 節の初めで説明したとおり，パラレルコーパスを用意するためのコストがかかるという問題がある．Koehn ら [48] は，スペルが同じ単語対をシード翻訳対に追加している．しかし，この手法は，日本語と英語のように異なる文字が使われる言語対には適用できないという問題がある．Hazem ら [37] は，クエリの k 近傍語により，クエリの文脈情報を補足することで，問題を緩和している．しかし，この手法の効果は，パラメータ k の値に強く依存しており，適切なパラメータを与えないと性能が低いことに加え，パラメータの調整が難しいという問題がある．

シード翻訳対を必要としない手法も提案されている．Rapp[75] は，各言語で，全単語間の文脈共起度を表す行列を考え，言語間で，それらの行列の類似度が最大となるような置換を行うことで，翻訳対を抽出する手法を提案している．しかし，この手法は，計算量が極め

て多いという問題がある．Ismail ら [41] は，異言語の文脈ベクトル間の類似度を，翻訳対を使わずに計算する尺度を提案している．また，Fung[22] は，文脈の不均質性を基に作成した，言語に依存しない文脈ベクトルを用いている．しかし，Ismail ら [41] と Fung[22] ともに，少量のシード翻訳対を用いた従来手法に比べて性能が低いという問題がある．Haghighi ら [35] や Daumé III ら [12] は，確率的正準相関分析に基づいた生成モデルを通じて，翻訳関係を特定している[2]．このモデルは，各単語をスペルと文脈単語とで特徴付ける．しかし，彼らの実験結果によると，スペルの素性が手法の効果に大きく貢献している．言い換えると，異なる文字を使う言語対（例えば，日本語と英語）に対しては性能が悪いという問題がある．

2.2 品詞導出

本節では，コーパスからの品詞導出手法の従来研究について述べる．

これまで，品詞を導出する教師無し手法は数多く提案されてきた．しかし，初期の手法は，品詞タグの種類数を予め与えなければならないという問題があった [43, 29]．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 [18, 27, 6, 83].

Gael ら [27] 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）[5, 85] を品詞導出に適応している．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を基に求める．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 [6] 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 過程を導入している．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 [83] は，iHMM を基に，品詞導出と単語分割を同時に一つの問題として扱うモデルを提案している．また，Finkel ら [18] は，単語列から品詞タグを導出する iHMM を木構造に拡張することで，単語の係り受け構造から品詞タグを導出する手法を提案している．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．

[2]Haghighi ら [35] や Daumé III ら [12] の手法も，シード翻訳対との間接的関係を考慮できるとも捉えられる．しかし，彼らの手法は，間接的関係をトポロジカルに考慮する一方で，提案手法は，間接的な関連度を直接求める点が異なる．

以降では，4 章で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 [18] 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[3]．

2.2.1 有限ツリーモデル

本節では，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 [18] について説明する．

有限ツリーモデルは，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，次の式（2.1）のとおり再帰的に定義できる：

$$p_T(T_t) = p(x_t|z_t) \prod_{t' \in c(t)} p(z_{t'}|z_t) p_T(T_{t'}). \tag{2.1}$$

式（2.1）において，$c(t)$ はノード t の子ノードの集合を表す．

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 2.2 に示す．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．各隠れ状態が取り得る状態（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，次の式（2.2），(2.3）のとおり表記できる：

$$\phi_k \sim H, \tag{2.2}$$

$$x_t|z_t \sim F(\phi_{z_t}). \tag{2.3}$$

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を表す．$\boldsymbol{\pi}_k$ は，

[3] independent children model 以外の 2 つのモデルも，4 章で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．

ρ をパラメータとするディリクレ (Dirichlet) 分布から生成される．よって，$\boldsymbol{\pi}$ の生成は，次の式（2.4）のとおり表記できる：

$$\boldsymbol{\pi}_k|\rho \sim \text{Dirichlet}(\rho, \ldots, \rho). \tag{2.4}$$

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 Multinomial($\boldsymbol{\pi}_{z_t}$) で確率的に決定される．よって，$z_{t'}$ の生成は，次の式（2.5）のとおり表記できる：

$$z_{t'}|z_t \sim \text{Multinomial}(\boldsymbol{\pi}_{z_t}). \tag{2.5}$$

以上より，有限ツリーモデルは，式（2.2）から（2.5）をまとめて次のように定義できる：

$$\begin{aligned}
&\boldsymbol{\pi}_k|\rho \sim \text{Dirichlet}(\rho, \ldots, \rho),\\
&\phi_k \sim H,\\
&z_{t'}|z_t \sim \text{Multinomial}(\boldsymbol{\pi}_{z_t}),\\
&x_t|z_t \sim F(\phi_{z_t}).
\end{aligned} \tag{2.6}$$

2.2.2 無限ツリーモデル

本節では，4 章で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．

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）[85] 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．単純なディリクレ過程 (DP)[4][17] ではなく階層ディレクレ過程を用いた理由は，状態が異なる親ノードから生成される子ノードの状態を結びつけるためである．HMM から iHMM への拡張にも，同じ動機で階層ディレクレ過程が用いられている [5].

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，$G_0 \sim \text{DP}(\gamma, H)$，各 $G_k \sim \text{DP}(\alpha_0, G_0)$ である．この HDP は，Stick-breaking 過程[5] [80] の観点から考えると，$G_0 = \sum_{k'=1}^{\infty} \boldsymbol{\beta}_{k'} \delta_{\phi_{k'}}$，

[4]DP 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 は，DP(α_0, G_0) と表記する．そして，G がこの DP にしたがうとき，$G \sim \text{DP}(\alpha_0, G_0)$ と表記する．

[5]Sethuraman[80] は，DP(α_0, G_0) にしたがう G は，それぞれ互いに独立で同一の分布にしたがう 2 つの無限個の確率変数の列 $(\pi'_k)_{k=1}^{\infty}$ と $(\phi_k)_{k=1}^{\infty}$ を用いて，$\pi_k = \pi'_k \prod_{l=1}^{k-1}(1-\pi'_l)$，$G = \sum_{k=1}^{\infty} \pi_k \delta_{\phi_k}$ のように生成できることを示した．ここで，$(\pi'_k)_{k=1}^{\infty}$ と $(\phi_k)_{k=1}^{\infty}$ は，$\pi'_k|\alpha_0 \sim \text{Beta}(1, \alpha_0)$，$\phi_k \sim G_0$ のとおり生成される．$\boldsymbol{\pi}$ がこの過程で生成されるとき，$\boldsymbol{\pi} \sim \text{GEM}(\alpha_0)$ と表記する．

$G_k = \sum_{k'=1}^{\infty} \pi_{kk'} \delta_{\phi_{k'}}$ と定義できる．ここで，$\boldsymbol{\beta}$，$\boldsymbol{\pi}_k$，$\phi_{k'}$ は，それぞれ，$\boldsymbol{\beta} \sim \mathrm{GEM}(\gamma)$，$\boldsymbol{\pi}_k \sim \mathrm{DP}(\alpha_0, \boldsymbol{\beta})$，$\phi_{k'} \sim H$ のとおり生成される．

上記定義を無限ツリーモデルに対応させると，G_0 は子ノードの状態に共通の基底測度，G_k は親ノードの状態に特有の基底測度となる．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次のように定義できる：

$$
\begin{aligned}
&\boldsymbol{\beta}|\gamma \sim \mathrm{GEM}(\gamma), \\
&\boldsymbol{\pi}_k|\alpha_0, \boldsymbol{\beta} \sim \mathrm{DP}(\alpha_0, \boldsymbol{\beta}), \\
&\phi_k \sim H, \\
&z_{t'}|z_t \sim \mathrm{Multinomial}(\boldsymbol{\pi}_{z_t}), \\
&x_t|z_t \sim F(\phi_{z_t}).
\end{aligned}
\tag{2.7}
$$

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 2.3 に示す．有限ツリーモデル（図 2.2）との違いは，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．

推定

シンボル（$x_{1:T}$）を基に，その背後にある品詞タグを表す隠れ状態（$z_{1:T}$）を推定する方法を説明する．推定では，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．しかし，状態数が無限なモデルでは，取り得る全ての状態に対して，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 [85] 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 [18] は，Teh ら [85] により提案された 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ている．これらのギブスサンプリングによる推定では，まず各変数に初期値を与える．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．

無限ツリーモデルでは，隠れ状態は，次の式（2.8）の事後分布からサンプリングする

[18]：

$$p(z_t = k|z^{-t}, \boldsymbol{\beta}) \propto p(z_t = k, (z_{t'})_{t' \in s(t)}|z_{d(t)}) \cdot p((z_{t'})_{t' \in c(t)}|z_t = k) \cdot f_k^{-x_t}(x_t). \tag{2.8}$$

ここで，z^{-t} は z_t 以外の隠れ状態を表し，$s(t)$ はノード t の兄弟ノードの集合，$d(t)$ はノード t の親ノード，$f_k^{-x_t}(x_t)$ は状態が k であるときの x_t の事後確率である．式（2.8）より，各品詞は，隠れ状態の遷移確率（式（2.8）の最初の 2 項）とシンボル出力確率（式（2.8）の最後の項）を基に決まることが分かる．

2.2.3 従来手法の問題点

本節では，従来の品詞導出手法の問題点について述べる．

2.2.2 節で説明した無限ツリーモデルを含め，従来の品詞導出手法は，品詞導出対象の言語の状況に基づいて品詞を導出する．例えば，無限ツリーモデルは，隠れ状態の遷移確率と，品詞導出対象の単語を表すシンボルの出力確率を基に品詞を導出する．つまり，翻訳相手の言語を考慮した品詞導出を行わない．したがって，1.1 節で述べた，既存の品詞体系の問題と同様，導出した品詞が機械翻訳に必ずしも最適とは限らないという問題がある．

例えば，図 1.2 の例 1 と例 2 で品詞を導出する場合を考える．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，同じシンボル「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導出されるため，例 1 と例 2 の「利用」の品詞は同じになる可能性が高い．しかし，1.1 節でも述べたとおり，例 1 の「利用」は，英語では動詞として振る舞い，例 2 の「利用」は，英語では名詞として振る舞う．このように，従来の品詞導出手法では，翻訳相手の言語における振る舞いの違いを反映した品詞を導出することは難しい．

2.3 本章のまとめ

本章では，コーパスからの翻訳対抽出手法と品詞導出手法に関して，従来研究とその問題点を説明した．

翻訳対抽出手法の説明では，本研究では，手法の適用範囲が広いコンパラブルコーパスからの抽出に焦点をあてることを述べた．また，従来のコンパラブルコーパスからの抽出手法の代表的な手法として，文脈類似度に基づく手法を紹介し，シード翻訳対が小規模の場合には性能が悪いという問題があることを説明した．

品詞導出手法の説明では，4 章で提案する品詞導出手法のベースとなる，無限ツリーモデルを紹介した．また，無限ツリーモデルを含めた従来手法は，翻訳相手の言語を考慮して品詞を導出しないため，導出した品詞が機械翻訳に必ずしも最適とは限らないという問題があることを説明した．

日本語

クエリ – 文脈単語	共起度
ピラニア – アマゾン	0.8
ピラニア – ジャングル	0.6
ピラニア – 淡水	0.5
淡水 – 魚	0.8

英語

クエリ – 文脈単語	共起度
piranha – Amazon	0.8
piranha – jungle	0.6
piranha – freshwater	0.5
anaconda – Amazon	0.8
anaconda – jungle	0.6
freshwater – fish	0.8

シード翻訳対 (日本語単語 – 英単語) :
アマゾン – Amazon, *ジャングル – jungle*, *魚 – fish*

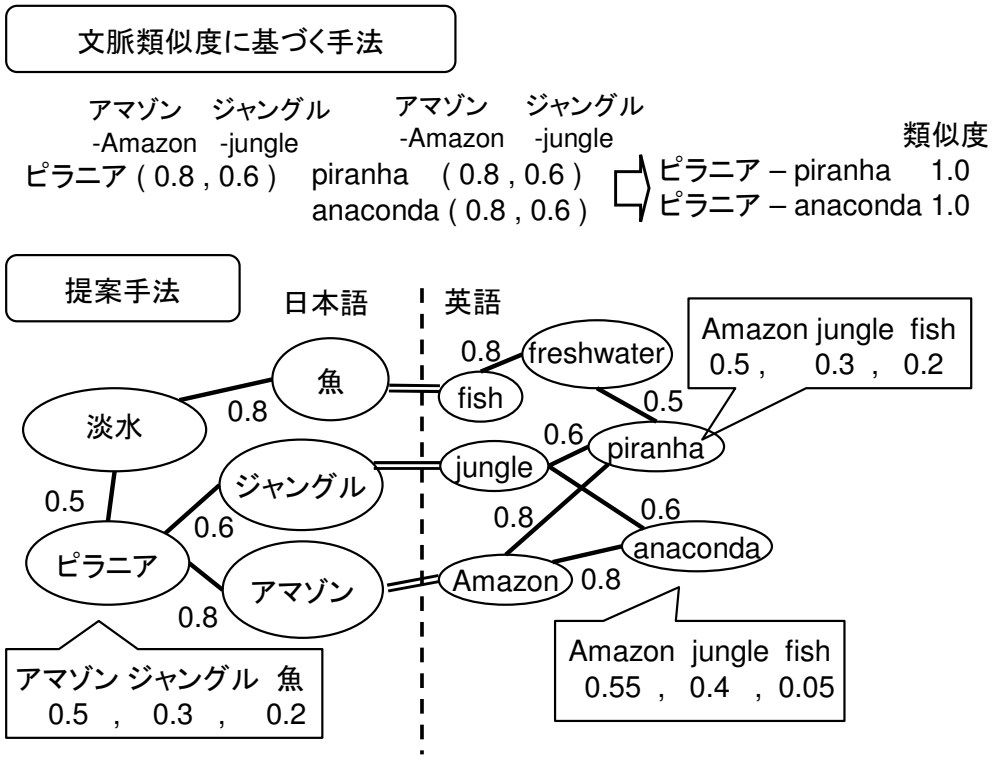

図 2.1: 文脈類似度に基づく手法と提案手法の例

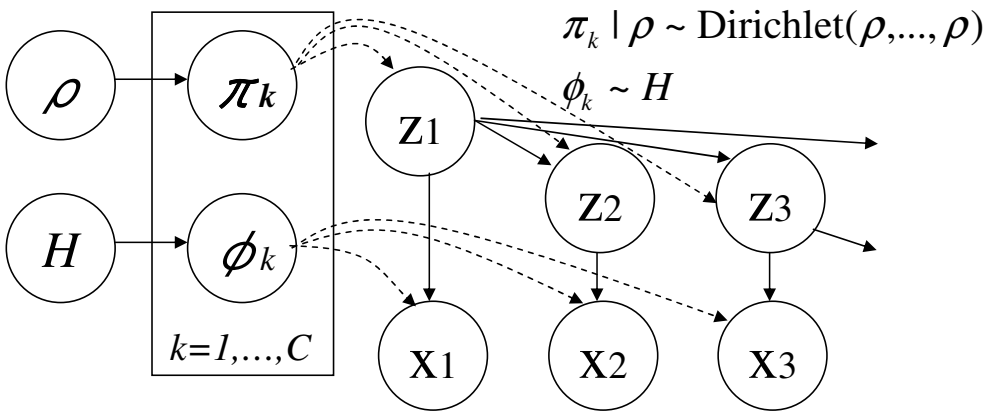

図 2.2: 有限ツリーモデルのグラフィカルモデル

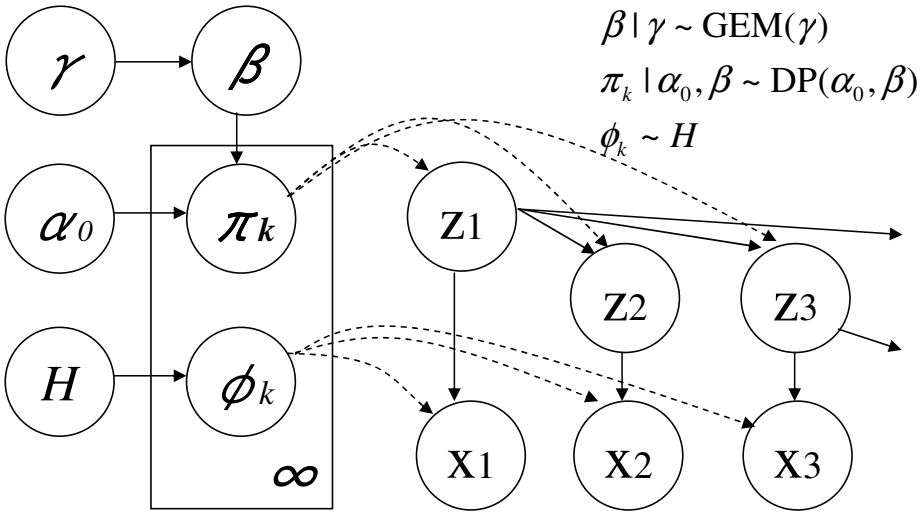

図 2.3: 無限ツリーモデルのグラフィカルモデル

第3章

ラベル伝播によるコンパラブルコーパスからの翻訳対抽出

本章では，コンパラブルコーパスから翻訳対を抽出する提案手法について述べる．まず，3.1 節で，2.1.1 節で説明した従来手法の問題点を踏まえて， グラフベースの手法であるラベル伝播 [94] を用いた，コンパラブルコーパスからの翻訳対抽出手法を提案する．そして，3.2 節では，特許コーパスを用いた評価を行い，提案手法が従来手法よりも性能が良いことを示す．また，3.3 節では，提案手法の効果や性質についての考察を行う．最後に，3.4 節で本章のまとめを行う．

3.1 提案手法

2.1.1 節で述べたとおり，従来手法は，シード翻訳対が小規模だと性能が悪化するという問題がある．本節では，この問題を解決するため，シード翻訳対との間接的関係を考慮した翻訳対抽出手法を提案する．

例として，図 2.1 を考える．左上の表より，日本語側では,「ピラニア」は「魚」と同一文脈で共起しないが,「淡水」を通じて「魚」（翻訳対「魚 - fish」）とある程度関連があることが分かる．また，右上の表より，英語側でも同様に,「piranha」は「freshwater」を通じて「fish」（翻訳対「魚 - fish」）とある程度関連があることが分かる．一方で,「anaconda」は「fish」（翻訳対「魚 - fish」）との関連はほとんどない．したがって，翻訳対「魚 - fish」に対する間接的関係を考慮することで,「piranha」と「anaconda」を区別できる．このように，シード翻訳対との間接的関係は，翻訳対を正しく特定するための有効な手がかりになると考えた．

Algorithm 翻訳対抽出

Require: コンパラブルコーパス（D^e, D^f），
シード翻訳対 S（言語 e，f のシード単語集合：S^e，S^f）

Ensure: 翻訳対 T を出力

1-1: $G^e = \{E^e, V^e, W^e\} \leftarrow$ construct-graph(D^e)

1-2: $G^f = \{E^f, V^f, W^f\} \leftarrow$ construct-graph(D^f)

2-1: $\tilde{G^e} = \{E^e, V^e, W^e, Q^e\} \leftarrow$ propagate-seed (G^e, S^e)

2-2: $\tilde{G^f} = \{E^f, V^f, W^f, Q^f\} \leftarrow$ propagate-seed (G^f, S^f)

3: $T \leftarrow$ extract-translation (Q^e, Q^f, S)

図 3.1: 提案手法のアルゴリズム

提案手法では，翻訳対抽出において間接的な関係を利用するために，次の仮定 (II) を導入する．**仮定 (II)** : **翻訳対関係にある単語は，シード翻訳対との直接的及び間接的共起関係が，言語を超えて似る傾向がある**．従来の文脈類似度に基づく手法が導入した仮定 (I) は，単語間の文脈共起関係は言語を横断して保存されることを意味している（2.1.1 節参照）．この仮定 (I) を文脈に対して再帰的に適用することで，仮定 (II) を導出できる．提案手法は，この仮定 (II) に基づき，シード翻訳対との直接的及び間接的共起関係が類似している単語対を翻訳対として特定する．

提案手法では，グラフベースの手法であるラベル伝播 [94] を用いて，シード翻訳対との間接的関係も含めた共起関係を獲得する．ラベル伝播とは，データを頂点としたグラフ上で，ラベル付きデータに付与されたラベルを，辺を通じてラベルの付いていないデータへと伝播させる手法である．ラベル伝播では，各データ（頂点）のラベルは，ラベルの分布としてソフトに表現される．このラベル分布は，各要素は各ラベルに対応し，重みは対応するラベルになる確率を示す分布である．

提案手法では，各単語をグラフの頂点，直接的関係をグラフの辺，シード翻訳対内の単語（以降,「シード単語」と呼ぶ）をラベルとする．そして，ラベル伝播により，シード単語をラベルとして伝播させ，各単語に対してシード単語の分布（以降,「シード分布」と呼ぶ）を得る．シード分布の各要素は，各シード単語に対応し，その重みが，対応するシード単語との間接的関係も考慮した関連度を表す．各単語のシード分布が得られた後は，シード分布間の類似度が高い，異言語の単語ペアを翻訳対として特定する．

提案手法のアルゴリズムを図 3.1 に示す．図 3.1 は，言語 e の単一言語コーパス D^e と言

語 f の単一言語コーパス D^f からなるコンパラブルコーパスから，シード翻訳対 S を用いて，翻訳対 T を抽出するアルゴリズムである．ステップ 1-1 は，言語 e におけるグラフ構築を表し，本ステップを通じて，D^e から，頂点集合 V^e，辺集合 E^e，辺の重み集合 W^e であるグラフ G^e を構築する．ステップ 1-2 では，ステップ 1-1 と同様に，言語 f において D^f からグラフ G^f を構築する．次に，ステップ 2-1 では，グラフ G^e 上で言語 e のシード単語 $S^e = \{v^e | \langle v^e, v^f \rangle \in S\}$ を伝播し，V^e 中の全ての単語に対するシード分布 Q^e を生成する．同様に，ステップ 2-2 では，グラフ G^f 上で言語 f のシード単語 $S^f = \{v^f | \langle v^e, v^f \rangle \in S\}$ を伝播し，V^f 中の全ての単語に対するシード分布 Q^f を生成する．最後に，ステップ 3 で，シード翻訳対 S により対応づけられるシード分布 Q^e と Q^f に基づいて，翻訳対 T を抽出する．

以上をまとめると，提案手法は次の 3 つのステップからなる．(1) 各言語に対するグラフ構築，(2) 各グラフにおけるシード伝播，(3) 翻訳対抽出である．以降の節では，各ステップを詳細に説明する．

3.1.1 グラフ構築

本節では，各言語において，単語間の関係を表すグラフを構築する方法を説明する．

提案手法では，単語間の関係として，同一文脈における共起関係など，方向性のない関係を用いるため，構築するグラフは無向グラフである．グラフは次の 3 つのステップで構築する．

ステップ 1. 頂点の割り当て： 単一言語コーパスから単語を抽出し，抽出した単語を各頂点に割り当てる．頂点集合を $V = \{v_1, \cdots, v_n\}$ で表す．

ステップ 2. 辺の重みの計算： 2 単語間の直接的な関連度（例えば，文脈共起度）を計算し，対応する頂点間の辺の重みとする．辺の集合と辺の重みの集合を，それぞれ，E と W で表す．そして，v_i と v_j を結ぶ辺を $e_{ij} \in E$ で表し，e_{ij} の重みを $w_{ij} \in W$ で表す．ここで，$|E| = |W|$ である．

ステップ 3. 辺の枝刈り： シード伝播時の計算コストを削減するため，重みが小さい辺をグラフから除く．

本論文では，ステップ 2 で用いる単語間の関連度の尺度が異なる，2 種類のグラフを提案する．関連度の尺度として文脈共起度を用いる「共起グラフ」と，文脈類似度を用いる「類似グラフ」である．以降，それぞれのグラフについて説明する．

共起グラフ

共起グラフは，前述の仮定 (II) を直接利用する方法であり，文脈共起度に基づき構築される．図 2.1 の下のグラフでは，同一文脈内で共起関係がある 2 単語間が辺で結ばれ，それらの辺は文脈共起度を重みとしてもつ．このグラフが共起グラフの例である．

共起グラフ構築におけるステップ 2「辺の重みの計算」，ステップ 3「辺の枝刈り」を具体的に説明する．ステップ 2 では，まず，各単語ペアに対して，同一文脈での共起回数を数え上げる．その後，その共起回数に基づいて，同一文脈における共起度を求める．文脈の定義や共起度の尺度は，文脈類似度に基づく手法が文脈をモデル化する際に使う，様々なものが利用できる（2.1.1 節ステップ 1 参照）．本論文では，文脈はサイズが 10 の窓，共起度の尺度は次の式（3.1）で定義される PMI を用いる：

$$w_{ij} = PMI(x_i, x_j) = log\frac{p(x_i, x_j)}{p(x_i) \cdot p(x_j)} = log\frac{p(x_i|x_j)}{p(x_i)}. \tag{3.1}$$

式（3.1）において，x_i と x_j は，それぞれ，頂点 v_i と v_j に割り当てられた単語を表す．$p(x_i)$ は，x_i が文脈内に出現する確率である．また，$p(x_j)$ は，$p(x_i)$ と同様の定義である．$p(x_i, x_j)$ は，x_i と x_j が同一文脈において出現する確率である．$PMI(x_i, x_j)$ は，次の式（3.2）のように，単純に x_i と x_j の出現回数に基づき計算することが可能である：

$$PMI(x_i, x_j) = log\frac{\frac{f(x_i, x_j)}{N_{all}}}{\frac{f(x_i)}{N_{all}} \cdot \frac{f(x_j)}{N_{all}}}. \tag{3.2}$$

式（3.2）において，$f(x_i)$ は x_i が出現する文脈数，$f(x_j)$ は $f(x_i)$ と同様の定義，$f(x_i, x_j)$ は x_i と x_j が共起する文脈数，N_{all} は全文脈数である．しかし，式（3.2）のように単純に計算した $PMI(x_i, x_j)$ は，低頻度語に対して信頼性が低くなることが知られている [4]．そこで，本論文では，Andrade ら [4] により提案されたベイズ的手法により，$PMI(x_i, x_j)$ を推定する．具体的には，$p(x_i|x_j)$ と $p(x_i)$ を，次の式（3.3）のとおり，ベータ分布からサンプリングする：

$$p(x_i|x_j) \sim \text{BETA}(\alpha'_{x_i|x_j}, \beta'_{x_i|x_j}),\ p(x_i) \sim \text{BETA}(\alpha'_{x_i}, \beta'_{x_i}). \tag{3.3}$$

式（3.3）における各パラメータ（$\alpha'_{x_i|x_j}$，$\beta'_{x_i|x_j}$，α'_{x_i}，β'_{x_i}）は，次の式（3.4）のとおりで

ある：

$$
\begin{aligned}
\alpha'_{x_i|x_j} &= f(x_i, x_j) + \alpha_{x_i|x_j}, \\
\beta'_{x_i|x_j} &= f(x_j) - f(x_i, x_j) + \beta_{x_i|x_j}, \\
\alpha'_{x_i} &= f(x_i) + \alpha_{x_i}, \\
\beta'_{x_i} &= N_{all} - f(x_i) + \beta_{x_i}.
\end{aligned} \tag{3.4}
$$

また，式（3.4）における各ハイパーパラメータ（$\alpha_{x_i|x_j}$，$\beta_{x_i|x_j}$，α_{x_i}，β_{x_i}）は，次の式（3.5）のように，単語出現確率の平均で定義する：

$$
\begin{aligned}
\gamma_{x_i|x_j} &= \frac{1}{x_j\text{と共起する単語数}} \sum_{x_i \in x_j\text{と共起する単語}} \frac{f(x_i, x_j)}{f(x_j)}, \\
\alpha_{x_i|x_j} &= \gamma_{x_i|x_j},\ \beta_{x_i|x_j} = 1 - \gamma_{x_i|x_j}, \\
\gamma_{x_i} &= \frac{1}{\text{全単語数}} \sum_{x_i \in \text{ 全単語}} \frac{f(x_i)}{N_{all}}, \\
\alpha_{x_i} &= \gamma_{x_i},\ \beta_{x_i} = 1 - \gamma_{x_i}.
\end{aligned} \tag{3.5}
$$

ステップ 3 では，出現に負の相関がある単語間の辺，つまり $PMI(x_i, x_j) \leq 0$ となる辺をグラフから除く．

類似グラフ

共起グラフは，同一文脈における共起関係を直接表現するため，偶然生じる共起関係に基づいた誤った辺が生成されやすい．そこで，文脈共起度の代わりに文脈類似度に基づき辺を生成する，類似グラフを提案する．この文脈類似度は，全ての文脈単語との文脈共起関係を大局的に捉えて計算される．そのため，類似グラフは，文脈類似度を計算する過程で，偶然の共起関係をその他の正しい共起関係で補正することが可能である．

類似グラフを用いた翻訳対抽出は，次の仮定 (III) に基づいている．**仮定 (III)**：**翻訳対関係にある単語は，各シード単語との文脈類似度が，言語を超えて似る傾向がある**．この仮定 (III) は，文脈類似度は単語間の共起関係を基に計算されること，そして，単語間の共起関係は言語を横断して保存されること（仮定 (I)）より導出可能である．

類似グラフ構築におけるステップ 2「辺の重みの計算」，ステップ 3「辺の枝刈り」を具体的に説明する．ステップ 2 では，まず，各単語の文脈ベクトルを生成する．この文脈のベクトル化は，文脈類似度に基づく手法と同様である（2.1.1 節ステップ 1 参照）．本論文

では，文脈単語は，サイズが 4 の窓内の単語，共起度の尺度は，PMI（式（3.1），(3.3），(3.4)，(3.5)）を用いる．また，文脈単語は文脈内での出現位置毎に区別して扱う．具体的には，文脈内での位置（2 単語前，1 単語前，1 単語後，2 単語後）毎にベクトルを作成し，その後，作成した 4 つのベクトルを結合したベクトルを文脈ベクトルとする．各単語の文脈ベクトルを生成した後は，文脈ベクトル間の類似度を計算し，その類似度を辺の重みとする．類似度尺度は，文脈類似度に基づく手法が文脈ベクトル間の類似度を計算する際に使う，様々な尺度が利用できる（2.1.1 節ステップ 2 参照）．本論文では，次の式（3.6）で定義されるコサイン類似度を用いる：

$$w_{ij} = Cos(\vec{f_i}, \vec{f_j}) = \frac{\vec{f_i} \cdot \vec{f_j}}{|\vec{f_i}||\vec{f_j}|}. \tag{3.6}$$

式（3.6）において，$\vec{f_i}$ は，頂点 v_i に対応する単語 x_i の文脈ベクトルである．$\vec{f_j}$ は，$\vec{f_i}$ と同様の定義である．

ステップ 3 では，各頂点に対して，辺の重みが上位 100 位以内の辺のみを残し，それ以外の辺は削除する．

3.1.2 シード伝播

本節では，シード単語との間接的関係も含めた関連度を得るため，3.1.1 節で構築したグラフ上でシード単語を伝播させる，シード伝播を説明する．

シード伝播は，ラベル伝播により実現する．ラベル伝播は，ラベル無しデータのラベルを推定するため，ラベル付きデータのラベルをグラフ上で伝播させるグラフベースの手法である．このラベル伝播は，主に，ラベル付きデータの量は乏しいがラベル無しデータが大量に存在する場合に使われ，効果を発揮する．自然言語処理の分野においても，語義の曖昧性解消 [67, 1]，単語への品詞付与 [11]，品詞体系の獲得 [1] など，様々なタスクに適用され，効果が確認されている．

ラベル伝播は，次の式（3.7）の目的関数を最適化することにより，各頂点に対して，ソフトなラベル，つまり，ラベルの相応しさを示す重みの総和が 1 となる，ラベルの分布を

生成する [94]：

$$
\begin{aligned}
C(q) &= \sum_{v_i \in V \setminus V_S,\, v_j \in N(v_i)} w_{ij} \cdot \parallel q_i - q_j \parallel \\
&= \sum_{v_i \in V \setminus V_S,\, v_j \in N(v_i)} w_{ij} \cdot \sum_z (q_i(z) - q_j(z))^2 \qquad (3.7) \\
&\text{s.t.} \ \sum_z q_i(z) = 1 \, \forall v_i \ \text{ and } \ q_i(z) \geq 0 \, \forall v_i, z \ \text{ and } \ q_i = r_i \, \forall v_i \in V_S.
\end{aligned}
$$

式（3.7）において，$V_S \subseteq V$ は，ラベル付きの頂点の集合を表し，全頂点 V の部分集合である．また，$N(v_i)$ は，v_i と辺で結ばれている頂点の集合，$q_i(i = 1 \cdots |V|)$ は，頂点 v_i のラベル分布，$q_i(z)$ は，ラベル分布 q_i 中のラベル z に対応する次元の重みである．また，r_i は，ラベル付き頂点に対するラベル分布で，$r_i(z = v_i) = 1, r_i(z \neq v_i) = 0$ である．この目的関数は，グラフにおいて近い頂点ほど，それらのラベル分布が似るように，全頂点のラベル分布を平滑化するものである．

提案手法は，各シード単語をラベルと捉えたラベル伝播により，全シード単語との間接的関係も含めた関連度を得る．つまり，式（3.7）において，$V_S \subseteq V$ は，シード単語に対応する頂点の集合，ラベル分布は，シード分布となる．そして，ラベル伝播を通じて，シード単語に対応する頂点に近いほど，そのシード単語に対応する次元の重みが大きく，離れるほど，その重みが小さいシード分布を生成できる．図 2.1 の下のグラフの各頂点に付与されている吹き出しが，シード伝播により生成されるシード分布の一例である．例えば，英単語「piranha」のシード分布は，シード単語「Amazon」,「jungle」,「fish」に対応する次元の重みが，それぞれ，0.5，0.3，0.2 の分布である．

式（3.7）の最適化問題は，行列操作による閉形式解が存在する [95]．しかし，その解法は $|V|$ オーダーの逆行列が必要であり，本研究のように頂点の数が多い場合，計算量は膨大となる．そこで，提案手法では，Zhu ら [94] のように，各頂点の分布の重みを反復的に更新することで，式（3.7）の最適化問題を解く．

以上をまとめると，提案手法は，各シード単語をラベルに割り当てる．そして，共起グラフ又は類似グラフ上で，重みの付いた辺を通じて，シード単語を繰り返し伝播させる．シード単語の伝播過程では，各頂点のシード分布の重みが更新される．そして，伝播完了後のシード分布の各要素が，各シード単語との間接的関係も含めた関連度を表す．以降では，シード分布の重みの更新について具体的に説明する．

まず，各頂点 v_i のシード分布を，次の式（3.8）のとおり初期化する：

$$q_i^0(z) = \begin{cases} r_i & \text{if } v_i \in V_S \\ u(z) & \text{otherwise} \end{cases} . \tag{3.8}$$

式（3.8）において，$q_i^k(i = 1 \cdots |V|)$ は，シード伝播を k 回行った後の v_i のシード分布を表し，$q_i^k(z)$ は，シード分布 q_i^k 中のラベル z に対応する次元の重みを表す．また，u は一様分布である．

全頂点のシード分布を初期化した後は，シード分布を，重みの付いた辺を通じて隣接する頂点へと繰り返し伝播させる．各伝播では，重みの大きい辺で結ばれた頂点のシード分布の情報がより伝わるように，確率的に伝播を行う．この伝播により，距離の近い頂点ほど類似するシード分布を生成できる．具体的には，各頂点のシード分布を，次の式（3.9）のように更新する：

$$q_i^k(z) = \begin{cases} q_i^0(z) & \text{if } v_i \in V_S \\ \dfrac{\sum_{v_j \in N(v_i)} w_{ij} \cdot q_j^{k-1}(z)}{\sum_{v_j \in N(v_i)} w_{ij}} & \text{otherwise} \end{cases} . \tag{3.9}$$

3.2 節で後述する評価では，全頂点に対して，式（3.9）の処理を 10 回繰り返した．

具体例として，図 2.1 の日本語側のグラフにおけるシード伝播を説明する．以降，シード分布の第 1，2，3 番目の次元に対応するシード単語を，それぞれ,「アマゾン」,「ジャングル」,「魚」とし，シード分布を (「アマゾン」に対する重み,「ジャングル」に対する重み,「魚」に対する重み) で簡単に表現する．

まず最初に，式（3.8）のとおり，各頂点のシード分布を初期化する．シード単語である「アマゾン」,「ジャングル」,「魚」のシード分布は，それぞれ，$(1, 0, 0)$，$(0, 1, 0)$，$(0, 0, 1)$ に初期化される．そして，シード単語以外の単語「淡水」,「ピラニア」のシード分布は，一様分布 $(0.\dot{3}, 0.\dot{3}, 0.\dot{3})$ に初期化される．

次に，式（3.9）のとおり，シード分布を重みの付いた辺を通して伝播させる．シード単語である「アマゾン」,「ジャングル」,「魚」のシード分布は，以降の伝播過程全般で，初期化されたシード分布のままである.「淡水」には，その隣接頂点である「魚」と「ピラニア」のシード分布の情報が伝播される．第 1 回目のシード伝播では,「淡水」のシード分布

は次のように更新される：

$$\begin{aligned} q^1_{\text{淡水}} &= \frac{w_{\text{淡水·魚}} \times q^0_{\text{魚}} + w_{\text{淡水·ピラニア}} \times q^0_{\text{ピラニア}}}{w_{\text{淡水·魚}} + w_{\text{淡水·ピラニア}}} \\ &= (0.128,\ 0.128,\ 0.744). \end{aligned}$$

同様に,「ピラニア」には，隣接頂点「淡水」,「ジャングル」,「アマゾン」からシード分布の情報が伝わり，第 1 回目のシード伝播では,「ピラニア」のシード分布は次のように更新される：

$$\begin{aligned} q^1_{\text{ピラニア}} &= \frac{w_{\text{ピラニア·淡水}} \times q^0_{\text{淡水}} + w_{\text{ピラニア·ジャングル}} \times q^0_{\text{ジャングル}} + w_{\text{ピラニア·アマゾン}} \times q^0_{\text{アマゾン}}}{w_{\text{ピラニア·淡水}} + w_{\text{ピラニア·ジャングル}} + w_{\text{ピラニア·アマゾン}}} \\ &= (0.509,\ 0.403,\ 0.088). \end{aligned}$$

全頂点に対して第 1 回目のシード伝播が終了した後，第 2 回目のシード伝播が行われる．第 2 回目のシード伝播により,「淡水」,「ピラニア」のシード分布は次のように更新される：

$$\begin{aligned} q^2_{\text{淡水}} &= \frac{w_{\text{淡水·魚}} \times q^1_{\text{魚}} + w_{\text{淡水·ピラニア}} \times q^1_{\text{ピラニア}}}{w_{\text{淡水·魚}} + w_{\text{淡水·ピラニア}}}, \\ q^2_{\text{ピラニア}} &= \frac{w_{\text{ピラニア·淡水}} \times q^1_{\text{淡水}} + w_{\text{ピラニア·ジャングル}} \times q^1_{\text{ジャングル}} + w_{\text{ピラニア·アマゾン}} \times q^1_{\text{アマゾン}}}{w_{\text{ピラニア·淡水}} + w_{\text{ピラニア·ジャングル}} + w_{\text{ピラニア·アマゾン}}}. \end{aligned}$$

以降，同様の伝播が繰り返し行われる．

3.1.3 翻訳対抽出

本節では，3.1.2 節のシード伝播により得られたシード分布に基づき，翻訳対を抽出する方法を説明する．

提案手法は，前述の仮定 (II) 又は仮定 (III) に基づき，シード分布が類似している，異言語の 2 単語を翻訳対として特定する．シード分布は，各次元がシード単語に対応し，シード単語との関連度を重みとするベクトルと捉えられる．したがって，シード分布間の類似度は，文脈類似度に基づく手法が文脈ベクトル間の類似度を計算する際に用いた尺度（2.1.1 節ステップ 2 参照）により，計算できる．本論文では，次の式（3.10）で定義されるコサイ

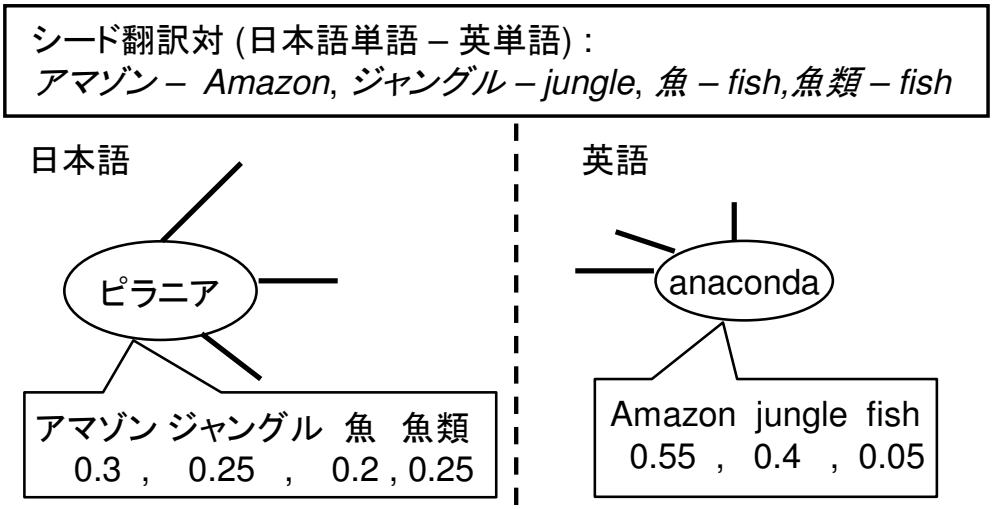

図 3.2: シード翻訳対とシード分布の例

ン類似度を用いる：

$$Cos(q_x^f, q_y^e) = \frac{\sum_{\langle v_k^f, v_l^e \rangle \in S} q_x^f(v_k^f) \cdot q_y^e(v_l^e)}{\sqrt{\sum_{\langle v_k^f, v_l^e \rangle \in S} (q_x^f(v_k^f))^2} \sqrt{\sum_{\langle v_k^f, v_l^e \rangle \in S} (q_y^e(v_l^e))^2}}. \tag{3.10}$$

式（3.10）において，q_x^f は，言語 f の単語 x が持つシード分布であり，q_y^e は，言語 e の単語 y が持つシード分布である．また，S は，シード翻訳対全体を表し，各翻訳対は，言語 f のシード単語 v_k^f と言語 e のシード単語 v_l^e のペア $\langle v_k^f, v_l^e \rangle$ で表す．そして，$q_x^f(v_k^f)$ は，シード分布 q_x^f 中のシード単語 v_k^f に対応する次元の重みを示す．$q_y^e(v_l^e)$ は，$q_x^f(v_k^f)$ と同様である．式（3.10）は，異言語のシード単語で表されたシード分布を，同一空間（シード翻訳対の空間）に写像しながら，類似度の計算をする．

ここで，シード翻訳対は，1 対多，多対 1 あるいは多対多の翻訳対を含んでいることを特筆しておく．⟨bank，銀行⟩と⟨bank，土手⟩の 2 つの翻訳対は，1 対多の翻訳対の一例である．複数単語に対応するシード単語（前述の例では「bank」）の次元は，シード翻訳対空間の複数箇所に写像される．式（3.10）では，次元の重みが複数回利用されることを意味する．したがって，シード伝播中や伝播後のシード分布は確率分布であるが，シード分布間の類似度を計算する中では，シード分布は確率分布として考えることはできない．そのため，本論文では，翻訳対抽出において，カルバック・ライブラー・ダイバージェンスなどの確率分布間の違いを測定する尺度を使用しない．

具体例として，図 3.2 において，日本語単語「ピラニア」と英単語「anaconda」のシード

分布間の類似度を計算する場合を考える．式（3.10）の分子は次のように計算される：

$$\sum_{(v_k^f, v_l^e) \in S} q_{\text{ピラニア}}(v_k^f) \cdot q_{anaconda}(v_l^e)$$

$$= q_{\text{ピラニア}}(\text{アマゾン}) \times q_{anaconda}(Amazon) + q_{\text{ピラニア}}(\text{ジャングル}) \times q_{anaconda}(jungle)$$
$$+ q_{\text{ピラニア}}(\text{魚}) \times q_{anaconda}(fish) + q_{\text{ピラニア}}(\text{魚類}) \times q_{anaconda}(fish).$$

ここで,「anaconda」のシード分布中のシード単語「fish」に対応する次元（$q_{anaconda}(fish)$）が 2 回使われていることを特筆しておく．これは，シード翻訳対に，シード単語「fish」に関する翻訳対が 2 つ（「魚-fish」と「魚類-fish」）含まれているからである．式（3.10）の分母も同様に計算できる．まとめると,「ピラニア」と「anaconda」のシード分布間の類似度は次のように求められる：

$$\begin{aligned} Cos(q_{\text{ピラニア}}, q_{anaconda}) &= \frac{0.3 \times 0.55 + 0.25 \times 0.4 + 0.2 \times 0.05 + 0.25 \times 0.05}{\sqrt{0.3^2 + 0.25^2 + 0.2^2 + 0.25^2}\sqrt{0.55^2 + 0.4^2 + 0.05^2 + 0.05^2}} \\ &= 0.83. \end{aligned}$$

3.2 評価

本節では，3.1 節で述べた提案手法の性能及び有効性を評価する．3.2.1 節では，評価で用いるコンパラブルコーパスの詳細を説明し，3.2.2 節では，評価で使用するシード翻訳対について述べる．また，3.2.3 節では，評価対象の単語について述べ，3.2.4 節では，評価で比較する手法について説明する．そして，3.2.5 節で評価結果を示す.

3.2.1 コンパラブルコーパス

評価では，1993 年から 2005 年に United States Patent and Trademark Office（USPTO）から発行された英語特許データと，同期間に日本特許情報機構（Japio）から発行された日本語特許データを用いる．これらの日本語及び英語特許データには，特許内容の分野を示す, International Patent Classification (IPC) コードが付与されている．この IPC コードに基づき，日本語及び英語特許データから「物理学」[1]の分野に属する文書を抽出し，それらの文書集合をコンパラブルコーパスとして評価で使用する．全分野の中で「物理学」に属する文書が最も多いため,「物理学」の分野を採用した．コンパラブルコーパス中の日本語文書は約 150 万文書，英語文書は約 44 万文書である．表 3.1 にコンパラブルコーパスの詳細を示す.

[1]セクション「G」の IPC コードが「物理学」の分野を示す.

		日本語	英語
文書	NTCIR 対訳データ	29,554	25,148
	NTCIR 対訳データ以外	1,450,277	413,079
	合計	1,479,831	438,227
内容語	「EDR 対訳辞書・NTCIR 対訳データ」	81,006	588,097
	「EDR 対訳辞書・NTCIR 対訳データ」以外	1,030,296	3,511,728
	合計	1,111,302	4,099,825
名詞・未知語	「EDR 対訳辞書・NTCIR 対訳データ」	65,207	571,145
	「EDR 対訳辞書・NTCIR 対訳データ」以外	1,012,267	3,499,444
	合計	1,077,474	4,070,589

表 3.1: コンパラブルコーパスの詳細

コンパラブルコーパス中の日本語文書と英語文書は，対応付いていないことを特筆しておく．ただし，このコンパラブルコーパスには，NTCIR-8 の特許翻訳タスク [21] で，トレーニング，ディベロップメント及びテストのために使われた対訳文 (以降,「NTCIR 対訳データ」と呼ぶ) が含まれている．この対訳文は，日本語特許とそれを起源に米国出願された特許をペアとした，文書単位の対応付けデータから生成されたものである．しかし，この対応付けデータは，全データと比較すると極めて少量である．具体的には，日本語では全データの 2.0% (=29,554/1,479,831)，英語では 5.7% (=25,148/438,227) にしかすぎない．さらに，これらの文書は，特許の起源情報を基に自動的に対応付けられているため，内容が必ずしも全く同一とは限らない．したがって，本節で行う評価は，まさにコンパラブルコーパスからの翻訳対抽出の評価であることを強調しておく．

コンパラブルコーパス中の日本語文書は，ChaSen[2]で形態素解析を行い，単語の分割と品詞を付与する．英語文書は，TreeTagger[79] により単語分割と品詞付与を行う．その後，各文書の機能語を削除する．機能語は，語義の情報をほとんど持たずに多くの単語と共起するため，共起する単語の意味を特定する手がかりにはならず，むしろノイズになると考えたからである．結果として，評価では，約 110 万種類の日本語の内容語が含まれる日本語コーパスと約 410 万種類の英語の内容語が含まれる英語コーパスで構成されるコンパラ

[2]形態素解析の辞書は IPAdic Version 2.7.0 (http://chasen.naist.jp/snapshot/ipadic/ipadic/doc/ipadic-ja.pdf) を用いた．ChaSen は http://chasen-legacy.sourceforge.jp/ からダウンロードして使用できる．

	翻訳対	シード単語（日本語）	シード単語（英語）
Lex_S	2,742	2,566	2,326
Lex_L	28,053	18,587	12,893

表 3.2: シード翻訳対の詳細

ブルコーパスを使用する（表 3.1 参照）．ここで留意すべきは，日本語と英語ではデータの書式が異なるため，英単語には表や数式内の単語も含まれるが，日本語単語にはそれらが含まれないことである．それゆえ，英語の文書数は日本語の文書数より少ないにもかかわらず，英語の内容語の種類は，日本語の約 4 倍となっている．

表 3.1 より，EDR 対訳辞書 [16] と NTCIR 対訳データの両方を用いても，コンパラブルコーパス中の日本語単語の約 7.3% (=81,006/1,111,302)，英単語の約 1.4% (=588,097/4,099,825) しかカバーできないことが分かる．これは，特許文書には専門用語や造語が多く含まれているからである．このことより，特に特許翻訳においては，パラレルではないコンパラブルコーパスからの翻訳対抽出が重要であることが分かる．

3.2.2 シード翻訳対

評価で使うシード翻訳対は，(1)EDR 対訳辞書と (2)NTCIR 対訳データの 2 つのソースに含まれる翻訳対から作成する．ソース (2) に関しては，GIZA++[69] により単語単位で自動的に対応付けた単語ペアを用いる．ただし，自動対応付け結果であるので，翻訳対として不適切な単語対も多く含む．そこで，両方向（日英，英日）の単語翻訳確率の平均が 0.5 以下の単語対は除く．このソース (1) と (2) の翻訳対のうち，コンパラブルコーパスに出現する名詞のペアのみを評価で用いる．その結果，ソース (1)，(2) から抽出した翻訳対は，それぞれ，27,353 対，2,853 対となった．これらの翻訳対は排他的ではない．

Morin ら [64] は，コンパラブルコーパスと同じ分野のパラレルコーパスから自動的に抽出した翻訳対を，シード翻訳対に加えることにより，コンパラブルコーパスからの翻訳対抽出性能を改善している．そして，本節の評価の前に行った予備評価でも，同様の結果を確認した．つまり，誤った翻訳対を含むソース (2) の翻訳対も使うことで，性能が改善することを確認した[3]．したがって，以降の評価では，ソース (1) の翻訳対にソース (2) の翻訳対

[3] ソース (1) の翻訳対のみをシード翻訳対とした場合，提案手法（$Cooc$）のトップ 1 正解率は 7.9%，トップ 20 正解率は 23.4%であった．一方，ソース (1) と (2) の両方の翻訳対をシード翻訳対とした場合，トップ 1 正解率は 9.2%，トップ 20 正解率は 28.3%となった．

を加えたものをシード翻訳対として用いる．

シード翻訳対の規模が翻訳対抽出性能に与える影響を評価するため，規模の異なる 2 種類のシード翻訳対（Lex_L と Lex_S）を作成し，評価で用いる．Lex_L は，ソース (1) と (2) の翻訳対の和集合であり，大規模なシード翻訳対を想定したものである．一方，Lex_S は，ソース (1) と (2) の翻訳対それぞれから，ランダムに 10 分の 1 ずつ抽出した翻訳対の和集合であり，小規模なシード翻訳対を想定したものである．Lex_L 及び Lex_S の詳細を表 3.2 に示す．Lex_L と Lex_S は，1 対多，多対 1 あるいは多対多の翻訳対を含んでいることを特筆しておく．

3.2.3 評価データ

評価では，ChaSen 又は TreeTagger により，名詞又は未知語と特定された単語に対する翻訳対抽出性能に着目する．その理由は，特許翻訳において翻訳できずに問題となる専門用語や造語の多くは，名詞や未知語だからである．評価で用いるコンパラブルコーパスには，名詞と未知語が，日本語文書に 1,077,474 種類，英語文書に 4,070,589 種類存在する．

また，本研究の目的は，既存の対訳辞書やパラレルデータではカバーできない単語の翻訳対を抽出することである．したがって，EDR 対訳辞書又は NTCIR 対訳データに含まれる単語を評価データから除く．その結果，評価対象は，日本語単語 1,012,267 種類，英単語 3,499,444 種類となった（表 3.1 参照）．

最終的に，この 1,012,267 種類の日本語単語の中からランダムで抽出した 1,000 種類の日本語単語を，評価データとした．つまり，今回の評価では，EDR 対訳辞書や NTCIR 対訳データではカバーできない，日本語の名詞又は未知語に対する翻訳対抽出性能を評価する．ここで，評価データの日本語単語に対する翻訳が，評価で用いるコンパラブルコーパスに必ずしも出現するとは限らないことを特筆しておく．

3.2.4 比較手法

評価では，ラベル伝播を利用する 2 種類の提案手法（$Cooc$ と Sim）と，2 種類のベースライン手法（$Rapp$ と $Andrade$）の性能の評価，比較を行う．$Cooc$ は，ラベル伝播時に共起グラフを用いる提案手法であり，Sim は，類似グラフを用いる提案手法である．

$Rapp$ は，2.1.1 節で説明した文脈類似度に基づく手法の代表的な手法である [76]．文脈ベクトルを生成する際には，サイズが 10 の窓内の単語を文脈単語として使い，文脈単語は文脈中の出現位置毎に区別して扱う．また，文脈共起度は，対数尤度比 [15] により算出す

る．そして，文脈ベクトル間の類似度は，マンハッタン距離で計算する．2 つのベクトル ($\vec{x} = (x_1, \ldots, x_n), \vec{y} = (y_1, \ldots, y_n)$) のマンハッタン距離は，次の式（3.11）で計算される：

$$Manhattan(\vec{x}, \vec{y}) = \sum_{i=1}^{n} |x_i - y_i|. \tag{3.11}$$

Andrade は，文脈類似度に基づく手法の中の高度な手法である [4]．文脈は，出現する窓（サイズは 10）に正の相関があるシード単語の集合を用いる．相関を測る尺度は，ベイズ的手法で推定する PMI（式（3.1），(3.3），(3.4），(3.5)）を使う．これは，共起グラフ構築のステップ 2「辺の重みの計算」で使った尺度と同じである（3.1.1 節参照）．また，文脈間の類似度は，文脈間で重複した単語数を基に，次の式（3.12）のとおり計算する：

$$Sim(x, y) = -log(P(matches = m)). \tag{3.12}$$

式（3.12）において，x は評価データの単語（今回の評価では日本語単語），y は翻訳候補の単語（今回の評価では英単語），m は単語 x と y の文脈間で重複した単語数である．また，$P(matches = m)$ は，単語 x と y の文脈において，偶然 m 個の単語が重複する確率であり，次の式（3.13）のように算出される：

$$P(matches = m) = \frac{{}_qC_m \cdot {}_{w-q}C_{c-m}}{{}_wC_c}. \tag{3.13}$$

式（3.13）において，q は単語 x の文脈中のシード単語の数，c は単語 y の文脈中のシード単語の数，w はシード翻訳対の総数である．

3.2.5 評価結果

3.2.4 節で述べた各手法の翻訳対抽出性能を図 3.3 に示す．以降，Lex_L を使ったときの 手法 を「手法名 (L)」，Lex_S を使ったときの 手法 を「手法名 (S)」と表記する．

翻訳対抽出性能は，トップ N 正解率（$1 \leq N \leq 20$）で評価する．トップ N 正解率は，「Acc_N」と表記する．トップ N 正解率とは，「手法が特定した上位 N 個の翻訳候補の中に，正しい翻訳が含まれている評価データの数」を「評価データの総数（=1,000）」で割った値である．翻訳候補の中に正しい翻訳が含まれているか否かは，人手で判定した．今回の評価では，評価データのほとんどが既存の対訳辞書に含まれていないため，既存の対訳辞書に基づく自動評価は行わなかった．

表 3.3 に，トップ 1 正解率とトップ 20 正解率をまとめる．表 3.3 より，Lex_S と Lex_L のど

	Lex_S		Lex_L	
	Acc_1	Acc_{20}	Acc_1	Acc_{20}
$Rapp$	1.5%	3.8%	4.8%	17.6%
$Andrade$	1.9%	4.2%	5.6%	17.6%
$Cooc$	3.2%	8.6%	9.2%	28.3%
Sim	4.1%	11.5%	10.8%	30.6%

表 3.3: 翻訳対抽出性能

ちらを使った場合も，提案手法（$Cooc$，Sim）の方がベースライン手法 ($Rapp$，$Andrade$) よりも性能が良いことが分かる．この結果が有意な差であるかどうかを符号検定を用いて検定すると，有意差水準 1%で有意差が認められた．この結果より，シード翻訳対との間接的関係を考慮することは，翻訳対抽出において有効であることが実験的に確認できる．この提案手法の有効性は，図 3.3 において，提案手法のトップ N 正解率曲線が，全ての N でベースライン手法よりも上回っていることからも確認できる．

また，表 3.3 と図 3.3 より，Lex_S と Lex_L のどちらを使った場合も，類似グラフを用いる Sim の方が，共起グラフを用いる $Cooc$ よりも性能が良いことが分かる．特に，Acc_{20} の性能差は，符号検定により有意差水準 5%で統計的に有意と認められた．この結果より，類似グラフは，共起グラフより翻訳対抽出に適していることが実験的に確認できる．

表 3.3 と図 3.3 におけるベースライン手法の性能は，従来研究で報告されている性能よりも低い．Andrade ら [4] では，$Andrade$ に対応する手法の性能は，Acc_1 が 14%，Acc_{10} が 46%と報告されている．また，Rapp[76] では，$Rapp$ に対応する手法の性能は，Acc_1 が 72%と報告されている．この理由の一つは，従来研究では，コンパラブルコーパス中に正しい翻訳が存在する単語のみを評価しているのに対し，今回の評価では，正しい翻訳が存在しない単語も評価対象になっていることである．正しい翻訳が存在しない単語の数だけ，達成可能な抽出性能の上限が低くなる．さらに，従来研究では，高頻度語のみが評価対象であることも理由の一つである．Rapp[76] は，普通名詞を評価対象とし，Andrade ら [4] は，コンパラブルコーパス中に 50 回以上出現する単語に限って評価を行っている．一方で，今回の評価対象には，多くの低頻度語が含まれている．一般的に，高頻度語に対する解析の方が，信頼性の高い情報を多く使えるため，低頻度語に対する解析よりも性能が良い．解析対象の出現頻度が翻訳抽出性能に与える影響については，3.3.3 節で考察を行う．

	$Sim(L)$ (2)	$Cooc(L)$ (5)	$Andrade(L)$ (181)
1	psychosis	polynephropathy	disease
2	manic-depression	neuroleptic	bowel
3	epilepsy	iridocyclitis	disorder
4	insomnia	Tic	symptom
5	dementia	manic-depression	sclerosis
	$Sim(S)$ (974)	$Cooc(S)$ (1652)	$Andrade(S)$ (1747)
1	ulceration	dyslinesia	bulimia
2	ulcer	encephalomyelopathy	spasticity
3	naphthol	ganglionic	Parkinson
4	dementia	corticobasal	Asymmetric
5	gastritis	praecox	anorexia

表 3.4: 「躁鬱病」に対する翻訳候補

3.3 考察

本節では，提案手法の効果や性質についての考察を行う．3.3.1 節では，シード単語との間接的関係の効果を，実例を用いて考察する．そして，シード翻訳対の規模，解析対象の単語の出現頻度が，翻訳対抽出性能に与える影響を，それぞれ，3.3.2 節，3.3.3 節で考察する．3.3.4 節では，類似グラフの効果を考察し，3.3.5 節では，シード伝播回数が翻訳対抽出性能に与える影響を考察する．そして，3.3.6 節では，提案手法の抽出誤りの原因を考察する．

3.3.1 シード単語との間接的関係の効果

本節では，日本語単語「躁鬱病」に対する翻訳対獲得を例にとり，シード単語との間接的関係の効果を確認する．

各手法で抽出した,「躁鬱病」に対する翻訳候補の上位 5 単語を表 3.4 に示す．表 3.4 において，手法名の隣にある括弧内の数字は，正しい翻訳「manic-depression」の順位を表す．また，日本語単語「躁鬱病」，英単語「manic-depression」に対して関連度の高い上位 5 個のシード単語を表 3.5 に示す．表 3.5 において，シード単語の下段にある数値は関連度を示

す．具体的には，$Cooc$ の場合，シード伝播後に得られるシード分布の要素，つまり，間接的な共起関係も考慮した関連度である．一方，$Andrade$ の場合，同一文脈における出現の相関度，つまり，直接的な共起関係に基づく関連度である．

表 3.4 より，$Cooc(L)$ は,「躁鬱病」の正しい翻訳を抽出できたのに対し，$Andrade(L)$ は，抽出できなかったことが分かる．これは，表 3.5 に示されているとおり，$Cooc(L)$ は，コンパラブルコーパス中の同一文脈で共起しない関係の深いシード単語（「躁鬱病」に対しては「神経症」,「不眠症」など,「manic-depression」に対しては「neurosis」,「insomnia」など）を多く活用できたからである．一方で，$Andrade(L)$ は，同一文脈で共起しない，これらの単語の情報は利用できない．以上より，間接的に関係のあるシード単語は，翻訳対を特定する上で重要な手がかりであり，提案手法はその手がかりを有効活用できることが事例より確認できる．

3.3.2 シード翻訳対の規模の影響

本節では，シード翻訳対の規模が翻訳対抽出性能に与える影響を考察する．

表 3.3 における Lex_S と Lex_L を用いた場合の性能比較から，全ての手法で，シード翻訳対が小規模だと性能が低いことが分かる．ベースライン手法（$Rapp$ と $Andrade$）の場合，Lex_S を用いると，文脈類似度を計算する過程で，文脈単語の多くがシード翻訳対空間に写像されず，それらの情報が失われるからである（2.1.1 節参照）．3.3.1 節の $Andrade(L)$ と $Andrade(S)$ を例にとり説明する．表 3.5 より，$Andrade(S)$ は，関連度が小さく関係の薄い文脈単語により,「躁鬱病」や「manic-depression」を表現していることが分かる．これは，$Andrade(L)$ が活用している関連度の高い文脈単語は，Lex_S に含まれていないからである．例えば,「精神病」は Lex_S に含まれないため，$Andrade(S)$ は,「精神病」を使って「躁鬱病」を特徴付けることができない．それゆえ，$Andrade(S)$ は，$Andrade(L)$ に比べて正しい翻訳を抽出することが難しい．

提案手法において，小規模なシード翻訳対を使った場合に性能が悪い理由を考察する．3.3.1 節の $Cooc(L)$ と $Cooc(S)$ を例にとり説明する．表 3.5 より，$Cooc(S)$ は,「躁鬱病」や「manic-depression」を滑らかなシード分布で表現していることが分かる．これは，Lex_S には，間接的な関係を考慮しても,「躁鬱病」や「manic-depression」と関係の強いシード単語が存在しないためである．この滑らかな分布は特徴が少なく，他の分布と識別しにくい．そのため，$Cooc(S)$ は正しい翻訳を抽出できなかったと考えられる．一方，Lex_L には,「躁鬱病」や「manic-depression」と関係の強い単語（「躁鬱病」に対しては「神経症」,「不眠症」など,「manic-depression」に対しては「neurosis」,「insomnia」など）が含まれる．そのた

め，$Cooc(L)$ では，それらの単語に対応する次元が尖った特徴的なシード分布により，「躁鬱病」や「manic-depression」を表現でき，正しい翻訳を特定できたと考えられる．

各手法のシード翻訳対の規模に対する頑健性を評価する．頑健性は，次で定義される Acc_{20} の悪化率で評価する．「悪化率 =(Lex_L 使用時の Acc_{20} - Lex_S 使用時の Acc_{20}) / Lex_L 使用時の Acc_{20}」である．この悪化率は，値が低いほどシード翻訳対に頑健であることを示す．評価の結果，$Rapp$，$Andrade$，$Cooc$ 及び Sim の悪化率は，それぞれ，78.4%，76.1%，69.6%，62.4%であった．$Cooc$ と $Andrade$ の悪化率の差が有意か否かを符号検定を用いて検定すると，有意差水準 1%で有意差が認められた．この結果より，提案手法の方がベースライン手法より，シード翻訳対の規模が小さくなっても性能劣化が少ないことが実験的に確認できる．これは，ベースライン手法は，文脈中のシード単語しか利用できないのに対して，提案手法は，間接的関係があるシード単語も利用して，各単語を特徴付けることができるからである．

提案手法の方がシード翻訳対の規模に対して頑健であることを更に考察する．文脈単語の中にシード単語が一つも存在しない，評価データの数を調べた．その結果，$Rapp(S)$ では 570 個，$Rapp(L)$ では 387 個，$Andrade(S)$ では 572 個，$Andrade(L)$ では 388 個あった．ベースライン手法では，文脈類似度を計算する際に行われる，シード翻訳対空間への写像により，これらの単語の文脈ベクトルは消失する．そのため，これらの単語の翻訳対を抽出することは原理的に不可能である．この種の単語は，Lex_L を用いた場合でも存在する．そして，翻訳対が小規模になるほど，その数は増大する．一方，提案手法では，各単語は全てのシード単語により特徴付けられるため，文脈単語にシード単語が存在しない単語に対しても翻訳対を抽出できる可能性がある．このことからも，シード翻訳対の規模が小規模な場合，提案手法はベースライン手法よりも有効であるといえる．

3.3.3 単語出現頻度の影響

本節では，解析対象の単語の出現頻度が翻訳対抽出性能に与える影響を考察する．

評価データは，EDR 対訳辞書や NTCIR 対訳データではカバーできない単語であるため，多くの低頻度語が存在する．評価データ 1,000 単語のうち，624 単語は，コンパラブルコーパス中に 50 回以下しか出現しない単語であった．以降，これらの単語を低頻度語と呼び，その他 376 単語を高頻度語と呼ぶ．表 3.6 に，Lex_L を使ったときの各手法の Acc_1 と Acc_{20} を，低頻度語と高頻度語に分けて示す．

表 3.6 より，低頻度語に対する性能は，高頻度語に対する性能と比べて極端に悪いことが分かる．これは，解析対象が高頻度語の場合，信頼性の高い文脈情報が豊富に利用できるのに対し，低頻度語の場合，統計的に信頼性の低い文脈情報しか使えないからである．

低頻度語が提案手法に与える影響を考える．低頻度語の場合，文脈情報が少ないため，偶然の共起関係と本来の共起関係の区別が難しい．そのため，提案手法でグラフを構築する過程で，しばしば，低頻度語の頂点と繋がる，偶然の共起関係に基づいた辺が生成される．そのようなグラフ上でシード伝播を行うと，本来繋がるべきではない辺を通じて，無関係なシード単語の情報が伝わり，シード分布に誤った間接的関係を混入させる可能性がある．それゆえ，提案手法においても低頻度語に対する性能が低くなる．

表 3.6 において $Sim(L)$ と $Cooc(L)$ を比較すると，低頻度語に対する性能の差の方が，高頻度語に対する性能の差よりも大きい．これは，$Sim(L)$ の方が，$Cooc(L)$ よりも低頻度語の悪影響を受けていないことを示している．この結果より，$Cooc$ は，共起関係を直接表現する共起グラフを用いるため，偶然の共起関係による悪影響を受けやすいが，Sim は，全文脈単語との共起関係を大局的に捉えて辺を生成する，類似グラフを用いることで，この問題を緩和できるといえる．

また，表 3.3 と表 3.6 を比較すると，提案手法の Acc_{20} は，高頻度語に評価を限ることで 10%以上良くなっている．これは，提案手法の現実的な適用方針の一つを示している．つまり，提案手法を高頻度語にのみ適用し，トップ 20 の翻訳候補を人手で整備することで，コストを抑えつつ大量の翻訳対を獲得できる見込みがあることを示している．

3.3.4 類似グラフの効果

3.3.3 節では，類似グラフは，偶然の共起関係の悪影響を緩和する効果があることを説明した．本節では，類似グラフのその他の効果について考察する．

評価データにはシード単語の同義語が含まれている．そこで，同義語に対する Acc_N を調査した．その結果，$Sim(L)$ の Acc_1 は 15.6%，Acc_{20} は 56.3%，$Cooc(L)$ の Acc_1 は 9.4%，Acc_{20} は 37.5%であり，$Sim(L)$ の方が，シード単語の同義語に対する性能が良かった．この結果より，Sim は，同義語を同一視し，それらの翻訳先に同じ単語を割り当てることに長けていると考えられる．

例として,「イオディン」に対する翻訳対抽出結果を紹介する．Lex_L には翻訳対「ヨウ素-iodine」が存在する．そして，解析対象の「イオディン」は，シード単語「ヨウ素」の同義語である．この時，$Sim(L)$ は，英単語「iodine」を翻訳候補第 1 位として正しく特定できたのに対し，$Cooc(L)$ では,「iodine」は翻訳候補第 36 位であった．

同義語は，文脈が似る傾向がある．それゆえ，類似グラフでは，同義語間は直接辺で結ばれやすく，シード伝播後に似たシード分布を持ちやすい．一方，共起グラフでは，同義語は共通の文脈単語を介して間接的に結ばれる傾向がある．したがって，類似グラフと比

較すると，異なるシード分布になりやすい．そのため，類似グラフの方が，翻訳対抽出において同義語を巧みに同一視できると考えられる．

本論分の研究対象である特許文書には，多くの外来語が存在する．外来語は文書毎に異なる表記で記述されやすい．例えば，英単語「user」を日本語で記述する場合,「ユーザ」と書かれる場合もあれば，最後に長音符を付けて「ユーザー」と書かれる場合もある．したがって，特許文書は他の分野の文書と比較して，同義語が多く含まれる文書であり，類似グラフの効果が特に発揮される分野と考えられる．

3.3.5 シード伝播回数の影響

本節では，シード伝播回数が翻訳対抽出性能に与える影響を考察する．

提案手法において，シード伝播回数 k を 0，5，10 回としたときの性能を表 3.7 に示す．$Cooc(k=0)$ や $Sim(k=0)$ は，シード伝播を行わない場合の性能であり，特に，$Cooc(k=0)$ は，従来の文脈類似度に基づく手法に相当する．$Cooc(k=0)$ と $Andrade$ は，文脈類似度を計算する際に使う類似度尺度が違うだけである．具体的には，$Cooc(k=0)$ はコサイン類似度（式（3.10））, $Andrade$ は文脈単語の重複を基にした尺度（式（3.12））を用いる．

表 3.7 より，Lex_S と Lex_L のどちらを用いた場合も，伝播回数を増やすことで性能が向上することが分かる．特に，伝播回数 0 回から 5 回への性能改善は，符号検定により有意差水準 1%で統計的に有意と認められた．これは，伝播を多数繰り返すことで，共起グラフや類似グラフ上でシード単語から離れた頂点に対しても，それらのシード単語の情報が伝播され，各シード単語との関連度が正確に把握できるようになるためである．

3.3.6 誤り分析

本節では，提案手法の抽出誤りの原因を，シード翻訳対の規模（3.3.2 節参照），出現頻度（3.3.3 節参照），シード伝播回数（3.3.5 節参照）以外の観点で考察する．

評価データの中には，根本的に翻訳先を見つけられない単語が 2 種類存在する．一つ目は，正解の翻訳である英単語が，コンパラブルコーパスの英語文書に含まれない場合である．1,000 個の評価データ中，この種の日本語単語は 133 単語（高頻度語が 12 単語，低頻度語が 121 単語）存在した．この問題は，コンパラブルコーパスから翻訳対抽出を行う際には避けられない問題である．

二つ目は，正解となる翻訳が，複数の英単語から構成される場合である．日本語の形態素解析では，複数の単語をまとめて一つの単語（複合語）にする処理がある程度行われる．一方，評価で使った英語の解析器は，スペースを手がかりに単語を分割するのみで，複合

語の処理を行わない．そのため，この問題が頻出した．例えば，日本語単語「掌紋」の翻訳は,「palm pattern」あるいは「palm print」であり，2 つの英単語から構成される．提案手法は,「掌紋」の翻訳先として，正しい翻訳の一部である英単語「palm」を特定できたが，翻訳対としては誤りである．この種の日本語単語は，1000 個の評価データ中，142 単語（高頻度語が 20 単語，低頻度語が 122 単語）存在した．この問題は，日英共に複合語の検出を行い，複合語も含めて共起グラフや類似グラフを構築することで解決できる．しかし，頂点の数が増加し，計算量が膨大となるため，今後の課題とする．

評価設定を変更しない限り，前述の 2 種類の単語に対する正しい翻訳対を抽出することはできない．そこで，評価データからこれら 2 種類の単語を除き，各手法の純粋な性能を再評価した．結果を表 3.8 に示す．表 3.8 中の「高頻度語」は，正しい翻訳がコンパラブルコーパス中に存在する高頻度語に対する性能を示し，従来の標準的な評価設定に相当する．

その他の誤りの主な原因は，日本語と英語で異なる，単語の語義曖昧性である．例えば，日本語単語「右」は，英語の「right」や「conservatism」などの意味を持つ．また，英単語「hill」は，日本語の「丘」や「坂」などの意味を持つ．提案手法では，シード伝播の際，片方の言語でのみ，この多義語を通じて異なる意味が混ざる．例えば，英語側でのみ，英単語「hill」を通じて,「丘」と「坂」の意味が混ざり合う．この非対称の多義性により，翻訳対となるべき単語のシード分布が，英語と日本語で異なる分布となり，正しい翻訳対を獲得できなくなる可能性がある．この問題は，各単語の語義の曖昧性を解消し，英単語と日本語単語の意味の粒度を揃えてから，共起グラフや類似グラフを生成することで，理論的には解決できる．例えば，英単語「hill」を「丘の意味の hill」,「坂の意味の hill」のように意味毎に分割してからグラフを生成する．しかし，語義曖昧性の解消自体，自然言語処理で長年取り組まれている難しい問題であるため，今後の課題とする．

3.4 本章のまとめ

本章では，従来の文脈類似度に基づく手法の問題点である，シード翻訳対が小規模な場合，性能が悪いという問題を解決するため，ラベル伝播を利用して，コンパラブルコーパスから翻訳対を抽出する手法を提案した．従来手法は，シード翻訳対との文脈共起関係（直接的関係）が類似した単語対を翻訳対として抽出するのに対し，提案手法は，シード翻訳対との直接的関係だけではなく，間接的関係も含めた関連性が似た単語対を，翻訳対として抽出する．提案手法では，各単語をシード単語の分布で表現し，そのシード分布を，単語間の直接的関係を繋いだグラフ上で伝播させることで，間接的関係も考慮した関連性を獲得する．日本語と英語の特許文書から作成したコンパラブルコーパスを用いた評価によ

り，提案手法は，従来の文脈類似度に基づく手法 [76, 4] よりも性能が良いことを確認した．そして，シード翻訳対との間接的関係は，翻訳対抽出において有効な手がかりになることを示した．

また，ラベル伝播で用いるグラフとして，同一文脈での共起関係に基づき構築する「共起グラフ」と，文脈の類似関係に基づき構築する「類似グラフ」の 2 つのグラフを提案した．そして，評価により，類似グラフの方が，共起グラフよりも有効であることを確認した．また，考察により，類似グラフは共起グラフに比べて，偶然の共起関係がもたらす悪影響を緩和でき，また，同義語を巧みに同一視できることを示した．

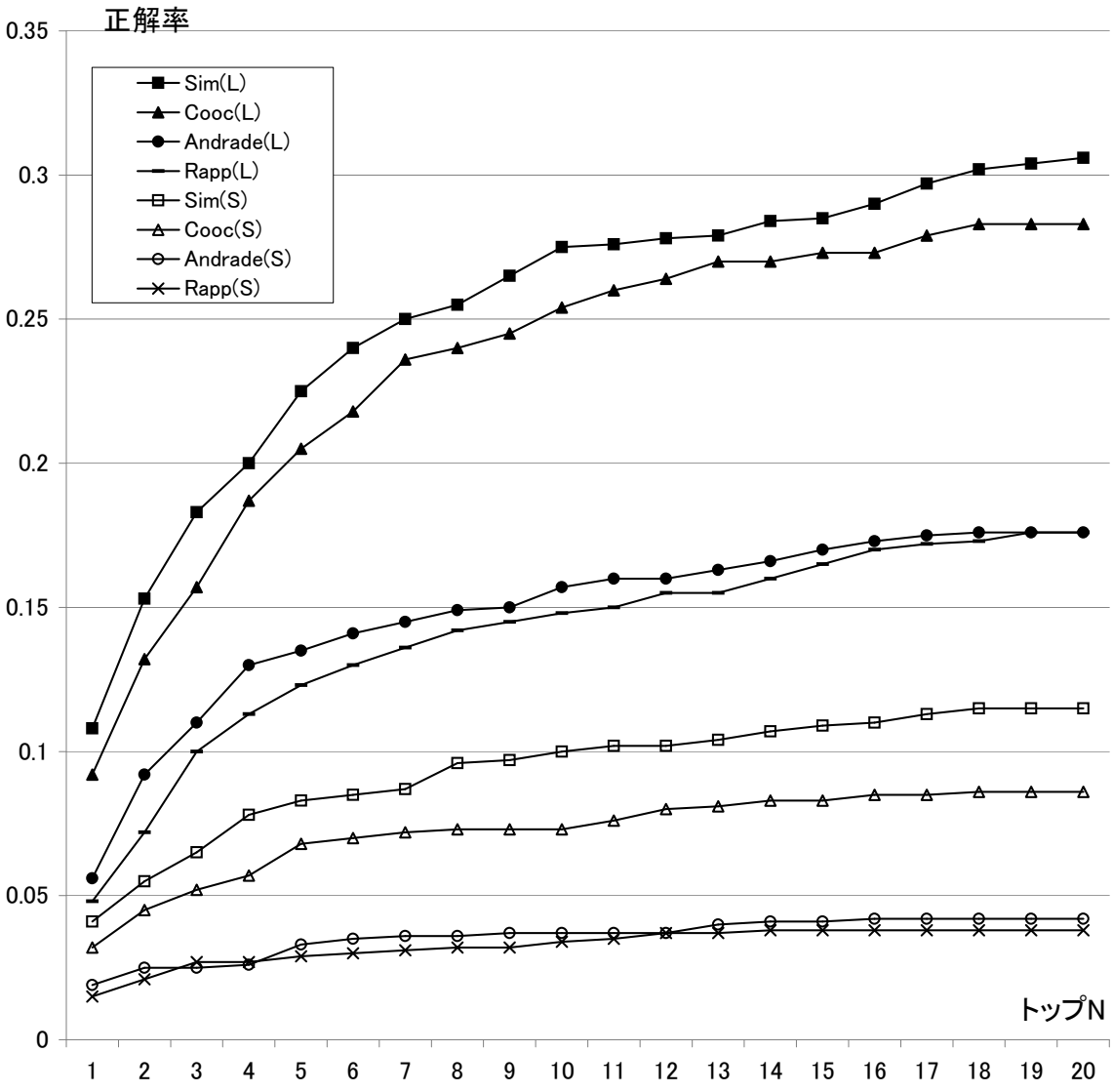

図 3.3: トップ N 正解率グラフ

躁鬱病				
	$Cooc(L)$	$Andrade(L)$	$Cooc(S)$	$Andrade(S)$
1	睡眠薬	睡眠薬	痴呆	後天
	0.12	7.6	0.016	5.0
2	精神病	老年	継子	痴呆
	0.11	6.3	0.014	3.7
3	神経症	精神病	後天	潰瘍
	0.08	6.3	0.012	3.2
4	ホルモン	気管支炎	陽性	ピリオド
	0.05	5.6	0.012	2.9
5	不眠症	後天	潰瘍	重度
	0.04	5.0	0.011	2.5

manic-depression				
	$Cooc(L)$	$Andrade(L)$	$Cooc(S)$	$Andrade(S)$
1	illness	illness	ganja	galop
	0.15	8.6	0.012	7.0
2	neurosis	psychotherapeutics	carbanilide	madness
	0.11	7.0	0.011	5.4
3	seizure	galop	paludism	libido
	0.07	7.0	0.011	5.2
4	psychosis	psychosis	resignation	vitiligo
	0.06	6.8	0.010	4.6
5	insomnia	somnambulism	galop	dementia
	0.04	6.7	0.009	4.3

表 3.5: シード単語（関連度上位 5 個）

	低頻度語		高頻度語	
	Acc_1	Acc_{20}	Acc_1	Acc_{20}
$Rapp(L)$	3.3%	12.8%	7.2%	25.6%
$Andrade(L)$	3.8%	12.3%	8.6%	26.3%
$Cooc(L)$	6.4%	20.9%	13.9%	40.7%
$Sim(L)$	8.2%	23.7%	15.0%	42.0%

表 3.6: 低頻度語，高頻度語に対する翻訳対抽出性能

	Lex_S		Lex_L	
	Acc_1	Acc_{20}	Acc_1	Acc_{20}
$Cooc(k=0)$	1.5%	3.9%	5.2%	17.4%
$Cooc(k=5)$	2.7%	7.3%	8.5%	26.9%
$Cooc(k=10)$	3.2%	8.6%	9.2%	28.3%
$Sim(k=0)$	2.3%	5.2%	6.6%	19.5%
$Sim(k=5)$	3.6%	9.1%	8.9%	27.7%
$Sim(k=10)$	4.1%	11.5%	10.8%	30.6%
$Andrade$	1.9%	4.2%	5.6%	17.6%

表 3.7: シード伝播回数の影響

	Acc_1 (低頻度語，高頻度語)	Acc_{20} (低頻度語，高頻度語)
$Rapp$ (L)	6.6% (5.5%, 7.8%)	24.3% (21.0%, 27.9%)
$Andrade$ (L)	7.7% (6.3%, 9.3%)	24.3% (20.2%, 28.8%)
$Cooc$ (L)	12.7% (10.5%, 15.1%)	39.0% (34.1%, 44.5%)
Sim (L)	14.9% (13.4%, 16.6%)	42.2% (38.8%, 45.9%)

表 3.8: 純粋な翻訳対抽出性能

第4章

機械翻訳のための品詞導出

本章では，コーパスから翻訳のための品詞を導出する提案手法について述べる．まず，4.1節で，2.2.3節で説明した従来手法の問題点を踏まえて，翻訳相手の言語を考慮した品詞導出手法を提案する．そして，4.2節では，特許翻訳の評価を行い，提案手法により導出した品詞を使うことで，特許翻訳の性能が改善できることを示す．また，4.3節では，提案手法の効果や性質についての考察を行う．最後に，4.4節で本章のまとめを行う．

4.1 提案手法

2.2.3節で述べたとおり，従来手法は，翻訳相手の言語における振る舞いの違いを反映した品詞を導出することは難しい．しかし，本研究では，そのような品詞は翻訳する際の手がかりになると考えた．例えば,「日本語の名詞」を「英語で名詞になる日本語の名詞」と「英語で動詞になる日本語の名詞」に区別できれば，日英翻訳に有効であると考えられる．そこで，本節では，翻訳相手の言語を考慮して，機械翻訳に適した品詞を導出する教師無し手法を提案する．そして，統語情報に基づく機械翻訳の中で，提案手法により導出した品詞を使うことにより，翻訳性能の改善を目指す．

提案手法は，2.2.2節で説明した単言語における無限ツリーモデルを，多言語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では，シンボルは品詞付与対象言語の単語を表す．一方，提案手法のシンボルは，品詞付与対象言語の単語に加えて，その単語に対応する翻訳相手の言語の単語を表す．言い換えると，原言語の品詞を導出する場合，原言語の品詞タグを表す隠れ状態は，原言語の単語と共に，対応する目的言語の単語をシンボルとして出力する．同様に，目的言語の品詞を導出する場合，目的言語の品詞タグを表す隠れ状態は，目的言語の単語と共に，対応する原言語の単語をシンボルとして出力する．以

降では，説明を簡単にするため，原言語の品詞付与を想定して説明する．

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導出するため，目的言語における違いを考慮した，原言語の品詞タグを導出できる．例えば，図 1.2 において，例 1 と例 2 の日本語単語「利用」の品詞タグを導出する場合を考える．提案手法は，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 のシンボルは英単語「use」も表し，例 2 のシンボルは英単語「usage」も表す．この異なるシンボルに基づいて品詞を導出するため，例 1 と例 2 の「利用」に対して異なる品詞を導出できる可能性が高い．

本節では，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，一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．

以降，4.1.1 節で結合モデルを説明し，4.1.2 節で独立モデルを説明する．また，4.1.3 節では，目的言語の品詞の情報をシンボルに反映させる．そして，4.1.4 節では，既存の品詞を細分化する品詞導出モデルを説明し，4.1.5 節では，提案モデルにおいて，シンボルを基に品詞タグを推定する方法を説明する．

4.1.1 結合モデル

本節では，提案手法の一つである結合モデルを説明する．

結合モデルは，2.2.2 節で説明した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及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（形式的な定義は式（2.7），グラフィカルモデルは図 2.3）である．

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 4.1.2 節で説明する独立モデルにおいても同じである．

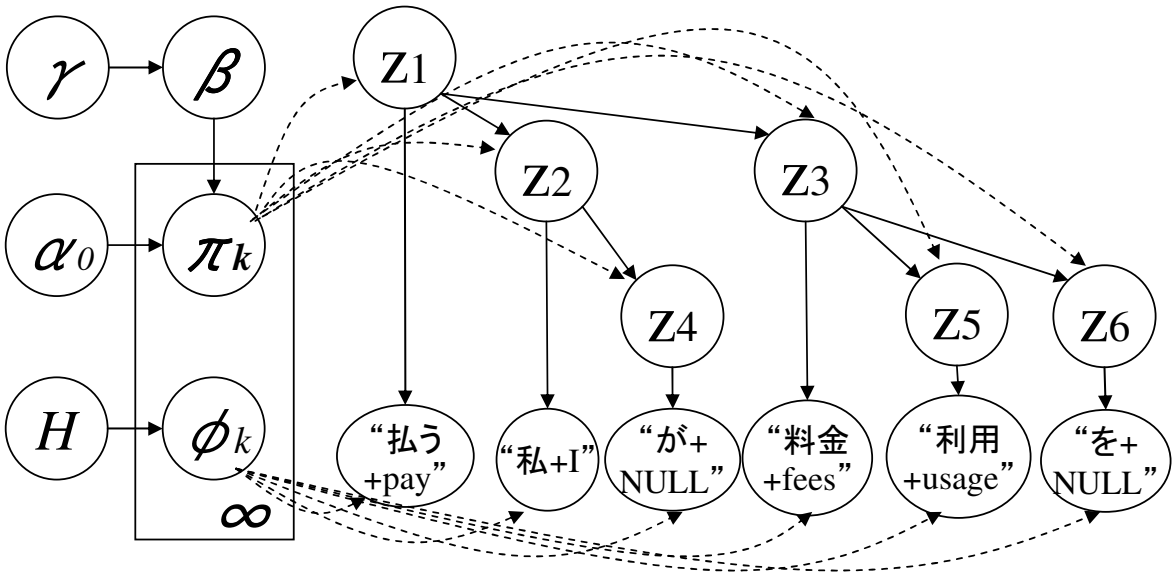

図 4.1: 結合モデルによる生成過程の例

図 4.1 に，図 1.2 の例 2 を結合モデルにより生成する過程を示す．図 4.1 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．例えば,「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 1.2 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導出される（4.1.5 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．

4.1.2 独立モデル

4.1.1 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み合わ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

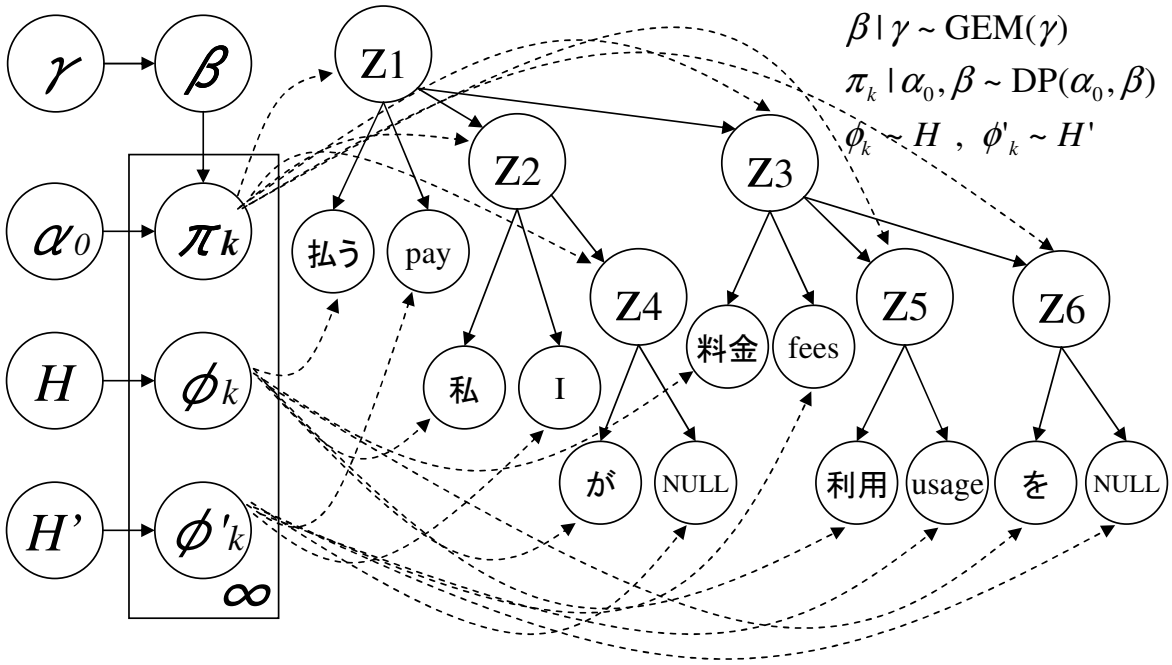

図 4.2: 独立モデルによる生成過程の例

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，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，独立モデルは次のように定義される：

$$
\begin{aligned}
&\boldsymbol{\beta}|\gamma \sim \mathrm{GEM}(\gamma), \\
&\boldsymbol{\pi}_k|\alpha_0, \boldsymbol{\beta} \sim \mathrm{DP}(\alpha_0, \boldsymbol{\beta}), \\
&\phi_k \sim H,\ \ \phi'_k \sim H', \\
&z_{t'}|z_t \sim \mathrm{Multinomial}(\boldsymbol{\pi}_{z_t}), \\
&x_t|z_t \sim F(\phi_{z_t}),\ \ x'_t|z_t \sim F'(\phi'_{z_t}).
\end{aligned}
\tag{4.1}
$$

図 4.2 に，図 1.2 の例 2 を独立モデルにより生成する過程を示す．図 4.2 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，例えば,「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．

4.1.3 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，提案モデルで導出しないことを特筆しておく．

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 1.2 の例 2 を生成する過程を考える．結合モデルでは,例えば,「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，一つのシンボルとする．例えば，図 1.2 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，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，それぞれ生成する．例えば，図 1.2 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．

4.1.4 品詞細分化

前節までの提案モデルは，品詞導出に何の制約も設けず，新しい品詞体系を導出することを想定していた．本節では，提案モデルのその他の適用方法として，既存の品詞体系の細分化 [18, 54] 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導出できる．

提案手法による品詞細分化では，既存の原言語の品詞タグ (s) 毎に，状態遷移確率（$\boldsymbol{\pi}_k^s$）とシンボル出力確率分布（$\boldsymbol{\phi}_k^s$，$\boldsymbol{\phi}_k'^s$，$\boldsymbol{\phi}_k''^s$）を考える．具体的には，まず，既存の原言語の品詞タガー等により，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 t の状態遷移やシンボル出力は，s_t に対応する分布にしたがって行われる．つまり，状態遷移は $\boldsymbol{\pi}_k^{s_t}$，シンボル出力は $\boldsymbol{\phi}_k^{s_t}$，$\boldsymbol{\phi}_k'^{s_t}$，$\boldsymbol{\phi}_k''^{s_t}$ で規定される分布にしたがう．このように，品詞細分化においては，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．その他は，新しい品詞体系の導出と同じである．

4.1.5 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，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．

2.2.2 節でも述べたとおり，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．しかし，提案モデルも，無限ツリーモデルと同様，状態数は無限であるため，取り得る全ての状態に対して事後確率を計算することは不可能である．

提案モデルでも，Finkel ら [18] 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．しかし，変数毎に他の変数を固定したリサンプリングを繰り返すギブスサンプリングでは，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．それゆえ，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 [26].

そこで，本節では，iHMM に対するビームサンプリング [26] 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 [66] により，各ノードが取り得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 [26].

具体的には，スライスサンプリングにより，各ノードが取り得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り返し行う．(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 [85] 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

次の式（4.2）のとおり，各 u_t は，区間 $[0, \pi_{z_{d(t)}z_t}]$ の一様分布からサンプリングする：

$$u_t \sim \text{Uniform}(0, \pi_{z_{d(t)}z_t}). \tag{4.2}$$

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[26] を木構造に拡張し，サンプリングを行う．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）．これらの動的計画法及びサンプリングは，文毎に独立であるため，文毎に並列処理が可能である．以降，forward filtering と backward sampling をそれぞれ説明する．

forward filtering :

各 z_t の取り得る状態 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（$p(z_t|x_{\sigma(t)}, u_{\sigma(t)})$，$p(z_t|x_{\sigma(t)}, x'_{\sigma(t)}, u_{\sigma(t)})$）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．

結合モデルの場合，z_t の事後分布は，次の式（4.3）[1]で計算できる：

$$
\begin{aligned}
& p(z_t|x_{\sigma(t)}, u_{\sigma(t)}) \\
& \quad \propto \; p(z_t, u_t, x_t|x_{\sigma(d(t))}, u_{\sigma(d(t))}) \\
& \quad = \; \sum_{z_{d(t)}} p(x_t|z_t) p(u_t|z_t, z_{d(t)}) p(z_t|z_{d(t)}) p(z_{d(t)}|x_{\sigma(d(t))}, u_{\sigma(d(t))}) \\
& \quad = \; p(x_t|z_t) \sum_{z_{d(t)}} [\pi_{z_{d(t)} z_t} > u_t] p(z_{d(t)}|x_{\sigma(d(t))}, u_{\sigma(d(t))}) \\
& \quad = \; p(x_t|z_t) \sum_{z_{d(t)}:\pi_{z_{d(t)} z_t} > u_t} p(z_{d(t)}|x_{\sigma(d(t))}, u_{\sigma(d(t))}).
\end{aligned} \tag{4.3}
$$

式（4.3）の導出過程では，u_t の確率密度は，$p(u_t|z_{d(t)}, z_t, \boldsymbol{\pi}) = \dfrac{[0 < u_t < \pi_{z_{d(t)} z_t}]}{\pi_{z_{d(t)} z_t}}$[2]であることを用いている．

同様に，独立モデルの場合，z_t の事後分布は，次の式（4.4）で計算できる：

$$
\begin{aligned}
& p(z_t|x_{\sigma(t)}, x'_{\sigma(t)}, u_{\sigma(t)}) \\
& \quad \propto \; p(z_t, u_t, x_t, x'_t|x_{\sigma(d(t))}, x'_{\sigma(d(t))}, u_{\sigma(d(t))}) \\
& \quad = \; \sum_{z_{d(t)}} p(x_t|z_t) p(x'_t|z_t) p(u_t|z_t, z_{d(t)}) p(z_t|z_{d(t)}) p(z_{d(t)}|x_{\sigma(d(t))}, x'_{\sigma(d(t))}, u_{\sigma(d(t))}) \\
& \quad = \; p(x_t|z_t) p(x'_t|z_t) \sum_{z_{d(t)}} [\pi_{z_{d(t)} z_t} > u_t] p(z_{d(t)}|x_{\sigma(d(t))}, x'_{\sigma(d(t))}, u_{\sigma(d(t))}) \\
& \quad = \; p(x_t|z_t) p(x'_t|z_t) \sum_{z_{d(t)}:\pi_{z_{d(t)} z_t} > u_t} p(z_{d(t)}|x_{\sigma(d(t))}, x'_{\sigma(d(t))}, u_{\sigma(d(t))}).
\end{aligned} \tag{4.4}
$$

後述する 4.2 節の評価では，Finkel ら [18] の実験設定に倣い，$F(\phi_k)$ は多項分布 Multinomial(ϕ_k) を仮定し，H はディリクレ分布 Dirichlet$(\rho, \ldots, \rho)$ を用いた．この仮定の下では，シンボル

[1]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．

[2]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．

の事後確率は，次の式（4.5）のとおり計算できる：

$$p(x_t|z_t) = \frac{\dot{n}_{x_t z_t} + \rho}{\dot{n}_{\cdot z_t} + N\rho}. \tag{4.5}$$

式（4.5）中の $\dot{n}_{x_t z_t}$ は，状態が z_t であるシンボル x_t の数，$\dot{n}_{\cdot z_t}$ 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，次の式（4.6）のとおり計算できる：

$$p(x'_t|z_t) = \frac{\dot{n'}_{x'_t z_t} + \rho'}{\dot{n'}_{\cdot z_t} + N'\rho'}. \tag{4.6}$$

式（4.6）中の $\dot{n'}_{x'_t z_t}$ は，状態が z_t であるシンボル x'_t の数，$\dot{n'}_{\cdot z_t}$ 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，次の式（4.7）又は（4.8）で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：

$$p(z_t|z_{c(t)}, x_{1:T}, u_{1:T}) \propto p(z_t|x_{\sigma(t)}, u_{\sigma(t)}) \prod_{t' \in c(t)} p(z_{t'}|z_t, u_{t'}). \tag{4.7}$$

$$p(z_t|z_{c(t)}, x_{1:T}, x'_{1:T}, u_{1:T}) \propto p(z_t|x_{\sigma(t)}, x'_{\sigma(t)}, u_{\sigma(t)}) \prod_{t' \in c(t)} p(z_{t'}|z_t, u_{t'}). \tag{4.8}$$

$\boldsymbol{\pi}$ 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，次の式（4.9）のディリクレ分布からサンプリングする：

$$(\pi_{k1}, \ldots, \pi_{kK}, \sum_{k'=K+1}^{\infty} \pi_{kk'}) \sim \text{Dirichlet}(n_{k1} + \alpha_0\beta_1, \ldots, n_{kK} + \alpha_0\beta_K, \alpha_0 \sum_{k'=K+1}^{\infty} \beta_{k'}). \tag{4.9}$$

式（4.9）において，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

$\boldsymbol{\pi}_i$ の要素の中で β_j に対応する数を，変数 $m_{ij} \in \boldsymbol{m}$ で表す．m_{ij} の事後分布は，次の式（4.10）である：

$$p(m_{ij}|\boldsymbol{z}, \boldsymbol{\beta}, \alpha_0) \propto S(n_{ij}, m_{ij})(\alpha_0 \beta_j)^{m_{ij}}. \tag{4.10}$$

式（4.10）における $S(a, b)$ は，符号なし第1種スターリング数である．具体的には，$S(0, 0) = S(1, 1) = 1$，$a > 0$ の場合，$S(a, 0) = 0$，$b > a$ の場合，$S(a, b) = 0$，その他の場合，$S(a + 1, b) = S(a, b - 1) + aS(a, b)$ で定義される．

この媒介変数 m_{ij} を用いて，$\boldsymbol{\beta}$ は，次の式（4.11）のディリクレ分布からサンプリングする：

$$(\beta_1, \ldots, \beta_K, \sum_{k'=K+1}^{\infty} \beta_{k'}) \sim \text{Dirichlet}(m_{\cdot 1}, \ldots, m_{\cdot K}, \gamma). \tag{4.11}$$

式（4.11）において，$m_{\cdot k} = \sum_{k'=1}^{K} m_{k'k}$ である．

α_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，次の式（4.12）の分布を定義する [85]：

$$q(\alpha_0, \boldsymbol{w}, \boldsymbol{v}) \propto \alpha_0^{\alpha_a - 1 + m_{\cdot\cdot}} e^{-\alpha_0 \alpha_b} \prod_{k=1}^{K} w_k^{\alpha_0} (1 - w_k)^{n_{\cdot k} - 1} (\frac{n_{\cdot k}}{\alpha_0})^{v_k}. \tag{4.12}$$

式（4.12）において，$m_{\cdot\cdot} = \sum_{k'=1}^{K} \sum_{k''=1}^{K} m_{k'k''}$ である．

式（4.12）を α_0 以外の変数について周辺化すると，式（4.13）のとおり，α_0 の事後分布が得られる：

$$q(\alpha_0|\boldsymbol{w}, \boldsymbol{v}) \propto \alpha_0^{\alpha_a - 1 + m_{\cdot\cdot} - \sum_{k=1}^{K} v_k} e^{-\alpha_0 (\alpha_b - \sum_{k=1}^{K} \log w_k)}. \tag{4.13}$$

α_0 は，この事後分布からサンプリングする．

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．それぞれ，次の式

(4.14), (4.15) の通りである：

$$q(w_k|\alpha_0) \propto w_k^{\alpha_0}(1-w_k)^{n_{\cdot k}-1}, \tag{4.14}$$

$$q(v_k|\alpha_0) \propto (\frac{n_{\cdot k}}{\alpha_0})^{v_k}. \tag{4.15}$$

式 (4.14), (4.15) において，$n_{\cdot k} = \sum_{k'=1}^{K} n_{k'k}$ である．これらの分布より，w_k と v_k をサンプリングする．

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 $\eta \in [0,1]$ と次の式 (4.16) の分布を定義する：

$$q(\gamma, \eta) \propto \gamma^{\gamma_a - 1 + K} e^{-\gamma\gamma_b} \eta^{\gamma} (1-\eta)^{m_{\cdot\cdot}-1}. \tag{4.16}$$

式 (4.16) を η について周辺化すると，式 (4.17) のとおり，γ の事後分布が得られる：

$$q(\gamma|\eta) \propto \gamma^{\gamma_a - 1 + K} e^{-\gamma(\gamma_b - \log\eta)}. \tag{4.17}$$

この事後分布から γ をサンプリングする．また，η は，式 (4.16) で γ を条件とした事後分布 (4.18) からサンプリングする：

$$q(\eta|\gamma) \propto \eta^{\gamma}(1-\eta)^{m_{\cdot\cdot}-1}. \tag{4.18}$$

4.2 評価

本節では，4.1 節で述べた提案手法の性能及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導出することである．したがって，提案手法により導出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[3]を使う．また，評価は，NTCIR-9 の日英特許翻訳タスク [33] において行う．このタスク

[3]翻訳システムは，しばしば，処理単位別に「X-to-Y 翻訳システム」で大別される.「X」は原言語の処理単位,「Y」は目的言語の処理単位を表す.「X」や「Y」は，String（「文字列」単位），Tree（「木」単位），Forest（木の集合である「森」単位）のいずれかである.

では，トレーニングデータとして約 320 万の対訳文，ディベロップメントデータ及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．評価では，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．

以降，4.2.1 節で評価手順を説明し，4.2.2 節で評価結果を示す．

4.2.1 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) 原言語の品詞タグ導出，(3) 原言語の品詞付与及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，4.1 節で提案した品詞導出手法を適用するために必要な前処理を行う．

NTCIR-9 のトレーニングデータのうち，最初の 10,000 の日英対訳文を，日本語の品詞を導出するために使用する[4]．まず，各文に対して単語分割及び品詞付与を行う．日本語文は MeCab[5]，英文は TreeTagger[79] 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++[69] 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 [49]][6]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 のトレーニングデータ全てを使って GIZA++を実行する．

また，単語単位の品詞導出を行うため，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[51] を用いて行う．しかし，CaboCha は，英語や中国

[4]全てのトレーニングデータを使うと計算量が膨大となるため，今回は，品詞導出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．

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

[6]Koehn ら [49] は，「diag-and」ヒューリスティクスとして説明している．

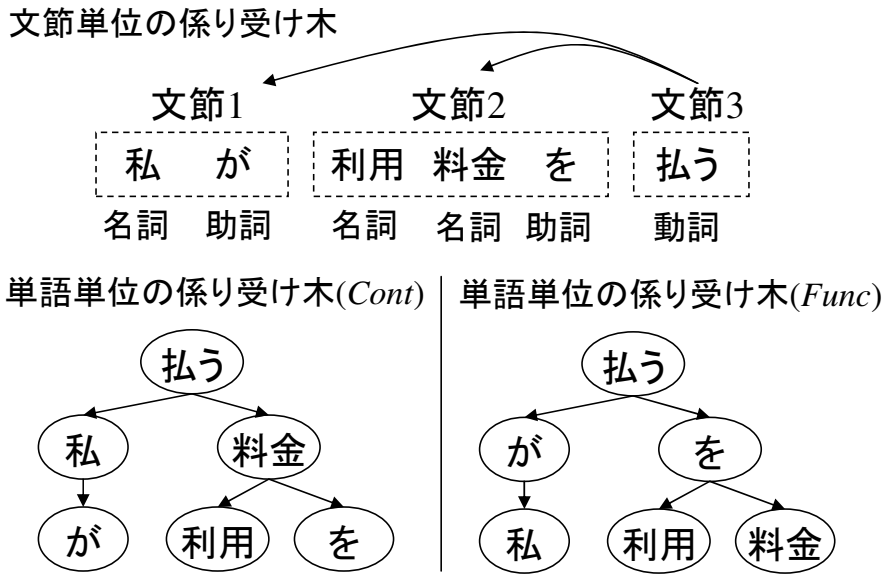

図 4.3: 係り受け木の例

語の係り受け解析器とは異なり，単語単位ではなく文節[7]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．

$Cont$ 及び $Func$ 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（主辞の子ノードとする）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 4.3 に，$Cont$ 及び $Func$ で構築した係り受け木を示す．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（例えば，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．

[7]文節とは，文を区切ったときに意味をなす最小単位で，一つの内容語と機能語（例えば，名詞と助詞など）で構成される．

ステップ 2. 品詞導出

本ステップでは，4.1 節で提案した品詞導出手法（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導出する．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 [18] による導出も行う．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，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）を 10,000 回繰り返す．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 は，Gael ら [26] に倣い，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（4.1.3 節参照）．以降，それぞれ,「s」,「P」,「s+P」で簡単に表記する．

また，品詞を導出する枠組みとして，新しい品詞体系の導出と既存の品詞タグの細分化（4.1.4 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，4.1.5 節で説明した各パラメータの推定を通じて，状態が更新される．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及びシンボル出力確率の分布を用いることを確認しておく．

ステップ 3. 品詞付与及び係り受け解析器の学習

本ステップでは，ステップ 2 で導出した品詞タグの付与及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，導出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 [36] の手法[8]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 で導出した品詞タグを特定することを確認しておく．

[8] http://triplet.cc/software/corbit/

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 翻訳モデルの学習及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*[9]により行う．

具体的には，まず，NTCIR-9 のトレーニングデータ中の全ての日本語文，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 [91] 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 [61] 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[78] として，L-BFGS[57][10]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[68] や PRO[38] よりも安定しており，ランダムな値に陥りにくいことが示されている [78]．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り返し回数）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．

4.2.2 評価結果

4.2.1 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 4.1 に示す．評価尺度は，大文字と小文字を区別した BLEU[71] を用いる．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．

ベースラインとして，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，表 4.1 に BS として示す．また，木構造構築手法 ($Cont$ 又は $Func$) が同じ場合，ベースライン（BS 及び $Mono$）よりも性能が良いシステムを太字で示す．具体的には，$Cont$ 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太字で示す．参考のために，機械翻訳の分野で最もよ

[9]機械翻訳の従来研究（例えば，システムコンビネーション [88] やオンライン学習 [87]）で使われている実績のあるツールで，`http://www2.nict.go.jp/univ-com/multi_trans/cicada/` からダウンロードして使用できる．

[10]L-BFGS は，準ニュートン法においてヘッセ行列を BFGS 公式で更新する BFGS 法を，省メモリで実現した方法である．

	$Cont$		$Func$	
	新品詞	細分化	新品詞	細分化
BS	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(%)

表 4.1: 日英翻訳性能

く使われている，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%であった．

表 4.1 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 $Mono$ の差が有意かどうかを，Koen[47] が提案したブートストラップによる検定手法を用いて検定すると，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，有意差水準 1%で有意差が認められた．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，表 4.1 より，独立モデルは，ベースライン BS よりも性能が良いことが分かる．特に，新品詞，細分化の両枠組みにおいて，$Func$ を用いた場合の性能差は，ブートストラップによる検定手法 [47] により，有意差水準 1%で有意と認められた．この結果より，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導出できることが実験的に確認できる．

また，表 4.1 より，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，ブートストラップによる検定手法 [47] により，有意差水準 1%で有意と認められた．この結果より，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，品詞導出に悪影響を及ぼす場合があることが実験的に確認で

	新品詞	細分化
$Joint$[s+P]	164	620
Ind[s+P]	102	517
IPA 品詞体系（2 階層目）	42	

表 4.2: 品詞タグの種類数

きる．

表 4.1 より，$Mono$ 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とが分かる．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導出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導出した品詞を使うと，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[11] を使った BS 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．

4.3 考察

本節では，提案手法の効果や性質についての考察を行う．4.3.1 節では，提案手法が導出した品詞と IPA 品詞体系の品詞を具体例で比較することにより，提案手法の効果を考察する．4.3.2 節では，翻訳システムの性能に関わる，係り受け解析器の性能について考察する．4.3.3 節では，木構造構築手法 $Cont$ と $Func$ について考察する．

4.3.1 IPA 品詞体系との比較

本節では，提案手法が導出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

[11]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．

いた場合に焦点をあてて議論を進める．

4.2.1 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 4.2 に示す．表 4.2 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，提案手法により導出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，この事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 $\underline{use}$ a card.」，(b)「$\underline{Using}$ the index is faster.」，(c)「I explain $\underline{using}$ an example」．これらの下線箇所の単語は全て，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．

4.3.2 品詞付与及び係り受け解析精度の影響

提案システムの性能は，4.2.1 節のステップ 2 で導出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及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及び係り受け解析精度について考察する．本節においても，4.3.1 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，導出した品詞が付与されたデータ 10,000 文は，全て，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

	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（2 階層目）	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

表 4.3: 品詞付与及び係り受け解析精度

て得られた 10,000 文のうち，最初の 9,000 文をトレーニングデータ，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 で導出した品詞の解析しやすさを示している．

表 4.3 に評価結果を示す．品詞付与及び係り受け解析精度の単位は，パーセント（%）である．表 4.3 中の IPA 品詞体系（2 階層目）とは，MeCab により付与された IPA 品詞タグと，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 4.3 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 4.3 より，提案手法で導出した品詞に対する性能（$Joint$，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 4.3 のように品詞付与精度が相対的に低い状況でも，表 4.1 が示すとおり，$Joint$[P] は除くが，提案手法により導出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よい．これは，表 4.3 に示される，低い解析精度というデメリットよりも，4.3.1 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．

	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（2 階層）	29 種類 (69%)	13 種類 (31%)

表 4.4: 細分化品詞の分布

新品詞及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，トレーニングデータのシンボルに過学習した品詞が導出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一つである．

4.3.3 $Cont$ 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 4.1 より，BS，$Mono$ も含めた全てのシステムで，$Func$ 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．例えば，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．

4.2.1 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捉えやすく，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，$Func$ を用いた場合に導出される品詞タグと $Cont$ を用いた場合に導出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 4.4 に示す．細分化元の品詞にしたがって，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 4.4 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 4.1 が示すとおり，$Cont$ と $Func$ のどちらを用いた場合も，Ind[s+P] は BS 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，共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．

4.4 本章のまとめ

本章では，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導出する手法を提案した．提案手法は，Finkel ら [18] の無限ツリーモデルにおいて，原言語と目的言語の単語間の対応関係に基づき，翻訳相手の言語の情報をシンボルに取り入れる．そして，両言語の情報を持つバイリンガルなシンボルから品詞を導出する．これにより，従来の品詞導出手法では導出できない，翻訳相手の言語における違いを反映したバイリンガルな品詞を導出する．NTCIR-9 データを用いた日英特許翻訳における評価を通じて，Forest-to-String 翻訳システムで提案手法が導出した品詞を使うと，既存の品詞（IPA 品詞体系）や従来手法が導出した品詞を使うより，翻訳性能が良いことを確認した．

また，バイリンガルなシンボルの生成過程が異なる 2 つのモデルを提案した．原言語の情報と目的言語の情報を一つのシンボルとして生成する結合モデルと，それらの情報を独立に別々のシンボルとして生成する独立モデルである．そして，評価を通じて，独立モデルは，結合モデルで生じるシンボルのスパースネス問題を解決し，翻訳により適した品詞を導出できることを確認した．

第5章

結論

5.1 まとめ

本論文では，特許翻訳の性能向上のため，特許翻訳で影響の大きい2つの問題に取り組んだ．

一つ目は，特許では，専門用語，造語が多く使われるため，特許翻訳では，未知語が多いという問題である．この問題を解決するため，コンパラブルコーパスから翻訳対を獲得する手法を提案した．従来のコンパラブルコーパスからの翻訳対抽出手法は，シード翻訳対との文脈共起関係（直接的関係）に基づいて翻訳関係を特定するため，シード翻訳対が小規模な場合，性能が悪いという問題があることを説明した．そこで，グラフベースの手法であるラベル伝播を用いて，シード翻訳対との間接的関係を獲得し，間接的関係も含めた関連性が似た単語対を，翻訳対として抽出する手法を提案した．そして，日本語と英語の特許文書から作成したコンパラブルコーパスを用いた評価を通じて，提案手法は，従来手法よりも精度高く翻訳対を抽出できることを確認した．抽出精度が従来手法より高くなった分，正しい翻訳対が多く抽出でき，未知語を削減できるため，特許翻訳の性能向上につながる．

二つ目は，複雑で長い文が多い特許翻訳に有効な統語情報に基づく機械翻訳では，使われている品詞が翻訳に適しているとは限らないという問題である．この問題を解決するため，コーパスから翻訳に適した品詞を獲得する手法を提案した．従来のコーパスからの品詞導出手法は，翻訳を想定した品詞を導出するものではないことを説明した．そこで，翻訳相手の言語における違いを考慮したバイリンガルな品詞を導出すべく，無限ツリーモデルをベースに，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を導出する手法を提案した．そして，日英特許翻訳による評価を通じて，統語情報に基づく機械翻訳である Forest-to-String 翻訳システムで，提案手法が導出した品詞を使うこ

とにより，既存の品詞や従来手法が導出した品詞を使うよりも，性能が良くなることを確認した．

前述したとおり，本論文で提案した手法は，特許翻訳の性能向上のために開発されたものであるが，各手法は特許翻訳に依存した方法ではない．そして，本論文で取り組んだ前記 2 つの問題は，程度の差こそあれ，特許以外の分野での翻訳においても問題となる．したがって，本論文で提案した手法は，特許翻訳以外の翻訳にも適用可能であり，貢献することを特筆しておく．

5.2 今後の課題

最後に，今後の課題について述べる．

5.2.1 ラベル伝播によるコンパラブルコーパスからの翻訳対抽出

ラベル伝播によるコンパラブルコーパスからの翻訳対抽出に関する，今後検討すべき代表的な課題をまとめる．

- 語義曖昧性の解消
  3.3.6 節で述べたとおり，提案手法の誤りの原因の一つとして，言語間で異なる，単語の語義曖昧性がある．そこで，単語の語義曖昧性を解消し，言語間で意味の粒度を揃えた後でグラフを構築することにより，提案手法の性能を改善できる可能性がある．

- 複合語への対応
  3.3.6 節で述べたとおり，本論文では，複合語の翻訳対を抽出できない．そこで，Daille ら [10] や Morin ら [63] で記載されているように，複合語に対応した手法への改良を検討したい．そのためには，複合語の検出と共に，提案手法における計算量の削減が必要と考えている．

- 文脈共起度と文脈類似度の組み合わせの検討
  本論文では，文脈共起度に基づく共起グラフと文脈類似度に基づく類似グラフの 2 種類を提案した．3.2 節の評価では，類似グラフの方が有効であることは確認したが，両者を組み合わせた手法の評価は行っていない．そこで，文脈共起度と文脈類似度を組み合わせることで，提案手法の性能を改善できる可能性があると考えている．例えば，文脈共起度と文脈類似度を組み合わせた関連度に基づくグラフ構築や，$Cooc$ と Sim が出力する翻訳候補のランキングの統合などを検討したい．

- 間接的関係の獲得方法の検討
  間接的関係は，パラレルコーパスからの言い換え獲得 [50] など，翻訳対抽出以外のタスクでも考慮されている．そこで，それらの研究を参考にして，ラベル伝播以外の手法で間接的関係を獲得することにより，手法の性能改善を試みたい．例えば，Kok ら [50] のようなランダムウォークや，modified adsorption[84] による間接的関係の利用を検討したい．

- 評価対象の拡大
  3.2.3 節で述べたとおり，本論文では，日本語の名詞と未知語に対する評価を行った．そこで，名詞や未知語以外の日本語単語に対する翻訳抽出性能や，英単語に対する翻訳抽出性能の評価を検討したい．そのためには，自動評価の利用など，評価コストの削減も同時に検討する必要がある．

- コンパラブルコーパスからの抽出以外のアプローチとの統合
  本論文では，コンパラブルコーパスからの翻訳対抽出に着目したが，2.1 節で述べたとおり，パラレルコーパスからの抽出やウェブを活用する手法など，その他のアプローチも多数存在する．その他のアプローチは，適用範囲が限られるという大きな問題はあるが，適用条件が満たされる場合においては，コンパラブルコーパスからの抽出よりも精度が良いことが知られている．そこで，提案手法とその他のアプローチを組み合わせることで，性能を相補的に改善できる可能性がある．例えば，まず，その他のアプローチを用いて翻訳対を抽出し，その後，抽出した翻訳対により提案手法のシード翻訳対を拡充することで，提案手法の性能を改善できる可能性がある．

- 機械翻訳システムとの統合
  本論文では，翻訳対抽出手法と機械翻訳システムの統合は行っていない．単純な統合方法としては，パイプラインで統合する方法が考えられる．具体的には，機械翻訳システムで翻訳できなかった未知語を，提案手法により抽出した翻訳単語に置き換えることで統合できる．3.2 節の評価は，この単純な統合方法における提案手法の効果を示している．一方で，提案手法は，翻訳対と同時に，それらのシード分布間の類似度も出力できる．そこで，その類似度を利用した，より効果的な機械翻訳システムへの統合方法を検討したい．

5.2.2 機械翻訳のための品詞導出

機械翻訳のための品詞導出に関する，今後検討すべき代表的な課題をまとめる．

- 計算量の削減

  4.2 節の評価では，品詞を導出する際，計算量の問題で NTCIR-9 のトレーニングデータの一部のみを使った．今後，導出する品詞の質や，その品詞を付与する係り受け解析器の性能を更に改善するためには，より大規模なデータから品詞を導出する必要があると考えている．したがって，提案手法のアルゴリズムを改良して計算量を削減したり，変数推定時の並列化を工夫して計算時間を短縮したりするなど，大規模データに適用するための改良を検討したい．

- 対応付け誤りへの対応

  シンボルとして考慮する翻訳相手の言語の情報は，4.2.1 節で述べたとおり，GIZA++ による単語単位の自動対応付け結果から抽出した．この自動対応付け結果には，誤った対応関係も含まれる．4.2.2 節の評価結果より，誤りを含む対応付け結果を用いても，提案手法の効果があることを確認しているが，今後は，対応付け精度の影響を調査すると共に，対応付け誤りに頑健な手法への改良を検討したい．

- 日英以外の言語対での評価

  4.2 節では，提案手法で導出した日本語の品詞が，既存の日本語の品詞よりも日英翻訳に有効であることを確認した．既存の品詞がどの程度翻訳に適しているかは，言語によっても，あるいは翻訳対象の言語対によっても異なると考えられる．そこで，今後は，提案手法を日英以外の言語対の翻訳に適用し，効果を確認したい．

- Forest-to-String 翻訳システム以外での評価

  4.2 節では，評価システムとして Forest-to-String 翻訳システムを用いた．今後は，String-to-Tree 翻訳システムや Tree-to-Tree 翻訳システムといった，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も，提案手法の有効性を確認したい．

- 単語単位の係り受け木構築手法の検討

  4.2.1 節で述べたとおり，本論文では，$Cont$ 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．しかし，$Cont$ と $Func$ 以外にも，単語単位の係り受け木を得る方法は存在する．例えば，Infinite PCFG モデル [54] により単語単位の係り受け木を構築できる．また，Nakazawa ら [65] の syntactic-head dependency trees や semantic-head dependency trees も利用できる．今後は，それらの構築手法を検討し，提案手法の性能改善を試みたい．

5.2.3 特許翻訳のその他の課題

本論文では，特許翻訳の性能を向上させるため，翻訳対獲得と翻訳のための品詞導出に取り組んだ．本節では，特許翻訳における，その他の代表的な課題を述べる．

- 長文への対応
  複雑で長い文が多い特許翻訳に有効な統語情報に基づく機械翻訳は，原言語や目的言語，あるいはその両方において構造解析結果を用いる．構造解析は，文が長くなると曖昧性が増えるため，精度が低くなるという問題がある．このように，統語情報に基づく機械翻訳を用いる場合も，長文への対応は必須であり，機械翻訳の各コンポーネント（構造解析，単語単位の対応付け，翻訳ルールの抽出など）を長文に頑健な手法に改良する必要がある．同時に，長文をセグメントに分割して短くした後で翻訳するアプローチ [77, 32] も考える必要がある．例えば，Goh[32] らの手法を用いて，重文を単文に分けてから解析を行うことで，性能改善が期待できる．

- 文書の構造把握
  特許は，一般的な文書と異なり，記述されるべき内容とパタンが決まっており，特有の形式で書かれる箇所がある．そのため，特許全体から学習した翻訳モデルにより，記載形式が特有の箇所を翻訳すると，翻訳精度が低下する場合がある．例えば，公開特許公報の場合,「書誌情報」,「要約」,「特許請求の範囲」,「明細書」などと順に記述される．そして,「特許請求の範囲」では，他の箇所と異なり，1 つの請求項が 1 つの名詞句で書かれる．そのため,「特許請求の範囲」以外が多く含まれるデータから学習した翻訳モデルを用いて請求項を翻訳すると，低い翻訳精度となる．そこで，例えば，熊野 [97] のように特許文書の構造を解析し，各パート毎に翻訳モデルの作成や翻訳手法の開発を行うことで翻訳性能を改善できる可能性がある．特に，請求項は，文は長いが構造に特徴があるため，請求項の構造解析 [96] を行い，その結果を翻訳で利用することは有効であると考えられる．

- 語順が異なる言語対の翻訳
  機械翻訳は，一般的に，語順が異なる言語対（日本語と英語など）に対する性能は低いことが知られている [42]．特に，特許は一文が長いため，語順の違いが顕著になり，誤りの原因となる．そこで，原言語の語順を目的言語の語順に並び替えてから翻訳するアプローチが提案されている．このアプローチは，例えば，英日翻訳の際，英文「I bought flowers.」を「I flowers bought.」のように日本語の語順に並べ替える．Isozaki ら [42] は，目的言語が SOV 型言語（日本語）の場合の並べ替え手法を提案し，英日

特許翻訳による評価を通じて有効性を示している．このような並び替え手法の開発や改良は，特許翻訳の性能改善のために重要であると考えられる．

- 言語資源の偏りへの対処
  特許翻訳の学習データ（対訳文）は，通常，複数の国で出願された同一内容の発明に関する特許文書対から抽出して作成される．しかし，同時出願される国には偏りがある．それゆえ，言語対によって対訳文の規模に偏りが生じる．例えば，日英，英韓の対訳文は大量に存在するが，日韓の対訳文は乏しい，というようなケースに遭遇する可能性が高い．翻訳対象の言語間で使える資源が乏しい場合に有効なアプローチとして，ピボット翻訳 [90] がある．ピボット翻訳とは，ある中間言語を介して目標の言語間を翻訳する手法である．例えば，前述のケースでは，英語を中間言語とし，日英翻訳，英韓翻訳を考えることにより，日韓翻訳を実現する．言語資源の偏りが生じやすい特許翻訳では，このピボット翻訳の活用方法は検討すべき課題の一つと考えられる．

謝辞

本研究を進めるにあたり，様々なご指導を頂きました奥村学教授に深く感謝致します．また，数多くの貴重なご意見を頂きました高村大也准教授に感謝致します．奥村学教授，高村大也准教授には，私が学部 4 年生の時から修士課程，博士課程と長きにわたりご指導を賜り，研究者としての物事の考え方，研究の進め方など基礎から丁寧にご教授を頂きました．心からお礼申し上げます．

本研究は，独立行政法人情報通信研究機構ユニバーサルコミュニケーション研究所多言語翻訳研究室において，多くの方々に支えられながら行うことができたものです．多言語翻訳研究室の皆様に感謝の意を表します．特に，隅田英一郎氏には，進学の機会を与えて頂き，様々な面でご教授，ご配慮頂いたことを心から感謝致します．渡辺太郎氏には，本研究の初期段階から終始暖かい激励とご指導，ご鞭撻を頂きました．本研究を進めることができたのは，渡辺氏の熱心なご指導を受けることができたためです．ここに深く感謝致します．

審査教官である，東京工業大学大学院総合理工学研究科物理情報システム専攻の小林隆夫教授，住田一男連携教授，熊澤逸夫教授，篠崎隆宏准教授には，本研究に対し大変貴重なご意見を頂きました．心から感謝致します．

奥村研究室の皆様には，ゼミを通じて有益なご意見を頂きました．ありがとうございます．

日本電気株式会社情報・メディアプロセッシング研究所の中尾敏康氏，石川開氏には，本研究をまとめる上でのご配慮を賜りました．深く感謝致します．

また，ここには記しきれない多くの方々から，本研究に対するご意見，ご議論を頂きました．感謝致します．

最後に，いつも応援してくれた両親と，博士課程入学を快諾し，いつも笑顔で支えてくれた妻のレリエに心から感謝します．

研究業績

論文（査読あり）

- Akihiro Tamura, Taro Watanabe, Eiichiro Sumita, Hiroya Takamura, Manabu Okumura. “Extracting Translation Pairs from Comparable Corpora through Graph-based Label Propagation”, 自然言語処理, Vol.20, No.2, pp.133-160, 2013 年 6 月.
- 田村 晃裕, 高村 大也, 奥村 学. “複数文質問のタイプ同定”, 情報処理学会論文誌, Vol.47, No.6, pp.1954-1962, 2006 年 6 月.

国際会議（査読あり）

- Akihiro Tamura, Taro Watanabe, Eiichiro Sumita, Hiroya Takamura, Manabu Okumura. “Part-of-Speech Induction in Dependency Trees for Statistical Machine Translation”, In *Proceedings of the 51th Annual Meeting of the Association for Computational Linguistics (ACL 2013)*, pp.841-851, Sofia, Bulgaria, August 2013.
- Isao Goto, Masao Utiyama, Eiichiro Sumita, Akihiro Tamura, Sadao Kurohashi. “Distortion Model Considering Rich Context for Statistical Machine Translation”, In *Proceedings of the 51th Annual Meeting of the Association for Computational Linguistics (ACL 2013)*, pp.155-165, Sofia, Bulgaria, August 2013.
- Akihiro Tamura, Taro Watanabe, Eiichiro Sumita. “Bilingual Lexicon Extraction from Comparable Corpora Using Label Propagation”, In *Proceedings of the 2012 Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL 2012)*, pp.24-36, Jeju Island, Korea, July 2012.
- Akihiro Tamura, Hiroya Takamura, Manabu Okumura. “Japanese Dependency Analysis Using the Ancestor-Descendant Relation”, In *Proceedings of the 2007 Conference on Em-*

pirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL 2007), pp.600-609, Prague, Czech Republic, June 2007.

- Akihiro Tamura, Hiroya Takamura, Manabu Okumura. "Classification of Multiple-Sentence Questions", In *Proceedings of the 2nd International Joint Conference on Natural Language Processing (IJCNLP-05)*, pp.426-437, Jeju Island, Korea, October 2005.

全国大会・研究会

- 田村 晃裕, 高村 大也, 奥村 学. "符号化問題として解く日本語係り受け解析", 情報処理学会研究報告, 自然言語処理研究会, 2006-NL-176, pp.17-24, 2006 年 11 月.
- 田村 晃裕, 高村 大也, 奥村 学. "質問事項の抽出とその依存関係の特定", 言語処理学会第 12 回年次大会, pp.328-331, 2006 年 3 月.
- 田村 晃裕, 高村 大也, 奥村 学. "複数文質問のタイプ同定", 言語処理学会 第 11 回年次大会, pp.1084-1087, 2005 年 3 月.

受賞

- 田村 晃裕, 高村 大也, 奥村 学. "符号化問題として解く日本語係り受け解析", 情報処理学会 山下記念研究賞, 2008.
- 田村 晃裕, 高村 大也, 奥村 学. "複数文質問のタイプ同定", 第 22 回電気通信普及財団賞, テレコムシステム技術学生賞 入賞, 2007.

参考文献

[1] Andrei Alexandrescu and Katrin Kirchhoff. Data-Driven Graph Construction for Semi-Supervised Graph-Based Learning in NLP. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 204–211, 2007.

[2] Daniel Andrade, Takuya Matsuzaki, and Junichi Tsujii. Effective Use of Dependency Structure for Bilingual Lexicon Creation. In *Proceedings of the 12th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing 2011) - Volume Part II*, pages 80–92, 2011.

[3] Daniel Andrade, Takuya Matsuzaki, and Junichi Tsujii. Learning the Optimal Use of Dependency-parsing Information for Finding Translations with Comparable Corpora. In *Proceedings of the 4th Workshop on Building and Using Comparable Corpora*, pages 10–18, 2011.

[4] Daniel Andrade, Tetsuya Nasukawa, and Junichi Tsujii. Robust Measurement and Comparison of Context Similarity for Finding Translation Pairs. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, pages 19–27, 2010.

[5] Matthew J. Beal, Zoubin Ghahramani, and Carl E. Rasmussen. The Infinite Hidden Markov Model. In *Advances in Neural Information Processing Systems*, pages 577–584, 2001.

[6] Phil Blunsom and Trevor Cohn. A Hierarchical Pitman-Yor Process HMM for Unsupervised Part of Speech Induction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL 2011)*, pages 865–874, 2011.

[7] Pu-Jen Cheng, Jei-Wen Teng, Ruei-Cheng Chen, Jenq-Haur Wang, Wen-Hsiang Lu, and Lee-Feng Chien. Translating Unknown Queries with Web Corpora for Cross-Language Information Retrieval. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 146–153, 2004.

[8] Yun-Chuang Chiao and Pierre Zweigenbaum. Looking for candidate translational equivalents in specialized, comparable corpora. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING 2002)*, pages 1208–1212, 2002.

[9] Trevor Cohn and Phil Blunsom. A Bayesian Model of Syntax-Directed Tree to String Grammar Induction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP 2009)*, pages 352–361, 2009.

[10] Béatrice Daille and Emmanuel Morin. French-English Terminology Extraction from Comparable Corpora. In *Proceedings of 2nd International Joint Conference on Natural Language Processing (IJCNLP 2005)*, pages 707–718, 2005.

[11] Dipanjan Das and Slav Petrov. Unsupervised Part-of-Speech Tagging with Bilingual Graph-Based Projections. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT 2011)*, pages 600–609, 2011.

[12] Hal Daumé III and Jagadeesh Jagarlamudi. Domain Adaptation for Machine Translation by Mining Unseen Words. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT2011)*, pages 407–412, 2011.

[13] Hervé Déjean, Éric Gaussier, and Fatia Sadat. An approach based on multilingual thesauri and model combination for bilingual lexicon extraction. In *Proceedings of the 19th International Conference on Computational linguistics (COLING 2002)*, pages 1–7, 2002.

[14] Yuan Ding and Martha Palmer. Machine Translation Using Probabilistic Synchronous Dependency Insertion Grammars. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*, pages 541–548, 2005.

[15] Ted Dunning. Accurate Methods for the Statistics of Surprise and Coincidence. *COMPUTATIONAL LINGUISTICS*, 19(1):61–74, 1993.

[16] EDR. Bilingual Dictionary. In *Technical Report TR-029*. Japan Electronic Dictionary Research Institute, Tokyo, 1990.

[17] Thomas S. Ferguson. A Bayesian Analysis of Some Nonparametric Problems. *The Annals of Statistics*, 1(2):209–230, 1973.

[18] Jenny Rose Finkel, Trond Grenager, and Christopher D. Manning. The Infinite Tree. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL 2007)*, pages 272–279, 2007.

[19] Darja Fišer, Nikola Ljubešić, Špela Vintar, and Senja Pollak. Building and using comparable corpora for domain-specific bilingual lexicon extraction. In *Proceedings of the 4th Workshop on Building and Using Comparable Corpora*, pages 19–26, 2011.

[20] Atsushi Fujii, Masao Utiyama, Mikio Yamamoto, and Takehito Utsuro. Overview of the Patent Translation Task at the NTCIR-7 Workshop. In *Proceedings of the 7th NTCIR Workshop*, pages 389–400, 2008.

[21] Atsushi Fujii, Masao Utiyama, Mikio Yamamoto, Takehito Utsuro, Terumasa Ehara, Hiroshi Echizen-ya, and Sayori Shimohata. Overview of the Patent Translation Task at the NTCIR-8 Workshop. In *Proceedings of the 8th NTCIR Workshop*, pages 371–376, 2010.

[22] Pascale Fung. Compiling Bilingual Lexicon Entries from a Non-Parallel English-Chinese Corpus. In *Proceedings of the 3rd Annual Workshop on Very Large Corpora*, pages 173–183, 1995.

[23] Pascale Fung and Kenneth Ward Church. K-vec: A New Approach for Aligning Parallel Texts. In *Proceedings of the 15th International Conference on Computational Linguistics (COLING 1994)*, pages 1096–1102, 1994.

[24] Pascale Fung and Kathleen McKeown. Finding Terminology Translations from Non-parallel Corpora. In *Proceedings of the 5th Annual Workshop on Very Large Corpora*, pages 192–202, 1997.

[25] Pascale Fung and Lo Yuen Yee. An IR Approach for Translating New Words from Nonparallel, Comparable Texts. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, pages 414–420, 1998.

[26] Jurgen Van Gael, Yunus Saatci, Yee Whye Teh, and Zoubin Ghahramani. Beam Sampling for the Infinite Hidden Markov Model. In *Proceedings of the 25th International Conference on Machine Learning (ICML 2008)*, pages 1088–1095, 2008.

[27] Jurgen Van Gael, Andreas Vlachos, and Zoubin Ghahramani. The infinite HMM for unsupervised PoS tagging. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2 - Volume 2 (EMNLP 2009)*, pages 678–687, 2009.

[28] Michel Galley, Jonathan Graehl, Kevin Knight, Daniel Marcu, Steve DeNeefe, Wei Wang, and Ignacio Thayer. Scalable Inference and Training of Context-Rich Syntactic Translation Models. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (COLING-ACL 2006)*, pages 961–968, 2006.

[29] Jianfeng Gao and Mark Johnson. A Comparison of Bayesian Estimators for Unsupervised Hidden Markov Model POS Taggers. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing (EMNLP 2008)*, pages 344–352, 2008.

[30] Nikesh Garera, Chris Callison-Burch, and David Yarowsky. Improving Translation Lexicon Induction from Monolingual Corpora via Dependency Contexts and Part-of-Speech Equivalences. In *Proceedings of the 13th Conference on Computational Natural Language Learning (CoNLL 2009)*, pages 129–137, 2009.

[31] Eric Gaussier, Jean-Michel Renders, Irina Matveeva, Cyril Goutte, and Herve Déjean. A Geometric View on Bilingual Lexicon Extraction from Comparable Corpora. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics (ACL 2004)*, pages 526–533, 2004.

[32] Chooi-Ling Goh and Eiichiro Sumita. Splitting Long Input Sentences for Phrase-based Statistical Machine Translation. 言語処理学会第 *17* 回年次大会講演論文集, pages 802–805, 2011.

[33] Isao Goto, Bin Lu, Ka Po Chow, Eiichiro Sumita, and Benjamin K. Tsou. Overview of the Patent Machine Translation Task at the NTCIR-9 Workshop. In *Proceedings of the 9th NTCIR Workshop*, pages 559–578, 2011.

[34] Spence Green, Michel Galley, and Christopher D. Manning. Improved Models of Distortion Cost for Statistical Machine Translation. In *Proceedings of Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2010)*, pages 867–875, 2010.

[35] Aria Haghighi, Percy Liang, Taylor Berg-Kirkpatrick, and Dan Klein. Learning Bilingual Lexicons from Monolingual Corpora. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: the Human Language Technology Conference (ACL-HLT 2008)*, pages 771–779, 2008.

[36] Jun Hatori, Takuya Matsuzaki, Yusuke Miyao, and Jun'ichi Tsujii. Incremental Joint POS Tagging and Dependency Parsing in Chinese. In *Proceedings of 5th International Joint Conference on Natural Language Processing (IJCNLP 2011)*, pages 1216–1224, 2011.

[37] Amir Hazem, Emmanuel Morin, and Sebastian Peña Saldarriaga. Bilingual Lexicon Extraction from Comparable Corpora as Metasearch. In *Proceedings of the 4th Workshop on Building and Using Comparable Corpora*, pages 35–43, 2011.

[38] Mark Hopkins and Jonathan May. Tuning as Ranking. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP 2011)*, pages 1352–1362, 2011.

[39] Fei Huang, Ying Zhang, and Stephan Vogel. Mining Key Phrase Translations from Web Corpora. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT-EMNLP 2005)*, pages 483–490, 2005.

[40] Liang Huang, Kevin Knight, and Aravind Joshi. A Syntax-Directed Translator with Extended Domain of Locality. In *Proceedings of the Workshop on Computationally Hard Problemsand Joint Inference in Speech and Language Processing*, pages 1–8, 2006.

[41] Azniah Ismail and Suresh Manandhar. Bilingual lexicon extraction from comparable corpora using in-domain terms. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, pages 481–489, 2010.

[42] Hideki Isozaki, Katsuhito Sudoh, Hajime Tsukada, and Kevin Duh. Head Finalization: A Simple Reordering Rule for SOV Languages. In *Proceedings of ACL 2010 Joint Fifth Workshop on Statistical Machine Translation and Metrics MATR (WMT-2010)*, pages 244–251, 2010.

[43] Mark Johnson. Why doesn't EM find good HMM POS-taggers? In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL 2007)*, pages 296–305, 2007.

[44] Hiroyuki Kaji. Extracting Translation Equivalents from Bilingual Comparable Corpora. *IEICE - Trans. Inf. Syst.*, E88-D:313–323, 2005.

[45] Sarvnaz Karimi, Falk Scholer, and Andrew Turpin. Machine Transliteration Survey. *ACM Computing Surveys*, 43(3):1–46, 2011.

[46] Kevin Knight and Jonathan Graehl. Machine Transliteration. *Computational Linguistics*, 24:599–612, 1998.

[47] Philipp Koehn. Statistical Significance Tests for Machine Translation Evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*, pages 388–395, 2004.

[48] Philipp Koehn and Kevin Knight. Learning a Translation Lexicon from Monolingual Corpora. In *Proceedings of ACL Workshop on Unsupervised Lexical Acquisition*, pages 9–16, 2002.

[49] Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical Phrase-Based Translation. In *Proceedings of the 2003 Human Language Technology Conference: North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2003)*, pages 48–54, 2003.

[50] Stanley Kok and Chris Brockett. Hitting the Right Paraphrases in Good Time. In *Proceedings of Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2010)*, pages 145–153, 2010.

[51] Taku Kudo and Yuji Matsumoto. Japanese Dependency Analysis using Cascaded Chunking. In *Proceedings of the 6th Conference on Natural Language Learning (CoNLL 2002)*, pages 63–69, 2002.

[52] Audrey Laroche and Philippe Langlais. Revisiting Context-based Projection Methods for Term-Translation Spotting in Comparable Corpora. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, pages 617–625, 2010.

[53] Florian Laws, Lukas Michelbacher, Beate Dorow, Christian Scheible, Ulrich Heid, and Hinrich Schütze. A Linguistically Grounded Graph Model for Bilingual Lexicon Extraction. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, pages 614–622, 2010.

[54] Percy Liang, Slav Petrov, Michael I. Jordan, and Dan Klein. The Infinite PCFG using Hierarchical Dirichlet Processes. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL 2007)*, pages 688–697, 2007.

[55] Dekang Lin. A Path-based Transfer Model for Machine Translation. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING 2004)*, pages 625–630, 2004.

[56] Dekang Lin, Shaojun Zhao, Benjamin Van Durme, and Marius Pasca. Mining Parenthetical Translations from the Web by Word Alignment. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: the Human Language Technology Conference (ACL-HLT 2008)*, pages 994–1002, 2008.

[57] Dong C. Liu and Jorge Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming B*, 45(3):503–528, 1989.

[58] Yang Liu, Qun Liu, and Shouxun Lin. Tree-to-String Alignment Template for Statistical Machine Translation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (COLING-ACL 2006)*, pages 609–616, 2006.

[59] Yang Liu, Yajuan Lü, and Qun Liu. Improving Tree-to-Tree Translation with Packed Forests. In *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing (ACL-IJCNLP 2009)*, pages 558–566, 2009.

[60] Wen-Hsiang Lu, Lee-Feng Chien, and Hsi-Jian Lee. Anchor Text Mining for Translation of Web Queries: A Transitive Translation Approach. *ACM Transactions on Information Systems*, 22(2):242–269, 2004.

[61] Haitao Mi and Liang Huang. Forest-based Translation Rule Extraction. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing (EMNLP 2008)*, pages 206–214, 2008.

[62] Haitao Mi and Qun Liu. Constituency to Dependency Translation with Forests. In *Proceedings of the 48th Annual Conference of the Association for Computational Linguistics (ACL 2010)*, pages 1433–1442, 2010.

[63] Emmanuel Morin, Béatrice Daille, Koichi Takeuchi, and Kyo Kageura. Bilingual Terminology Mining - Using Brain, not brawn comparable corpora. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL 2007)*, pages 664–671, 2007.

[64] Emmanuel Morin and Emmanuel Prochasson. Bilingual Lexicon Extraction from Comparable Corpora Enhanced with Parallel Corpora. In *Proceedings of the 4th Workshop on Building and Using Comparable Corpora*, pages 27–34, 2011.

[65] Toshiaki Nakazawa and Sadao Kurohashi. Alignment by Bilingual Generation and Monolingual Derivation. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012)*, pages 1963–1978, 2012.

[66] Radford M. Neal. Slice Sampling. *Annals of Statistics*, 31:705–767, 2003.

[67] Zheng-Yu Niu, Dong-Hong Ji, and Chew Lim Tan. Word Sense Disambiguation Using Label Propagation Based Semi-Supervised Learning. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*, pages 395–402, 2005.

[68] Franz Josef Och. Minimum Error Rate Training in Statistical Machine Translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL 2003)*, pages 160–167, 2003.

[69] Franz Josef Och and Hermann Ney. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29:19–51, 2003.

[70] Pablo Gamallo Otero and José Ramom Pichel Campos. Learning Spanish-Galician Translation Equivalents Using a Comparable Corpus and a Bilingual Dictionary. In *Proceedings of the 9th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing 2008)*, pages 423–433, 2008.

[71] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics (ACL 2002)*, pages 311–318, 2002.

[72] Viktor Pekar, Ruslan Mitkov, Dimitar Blagoev, and Andrea Mulloni. Finding Translations for Low-Frequency Words in Comparable Corpora. *Machine Translation*, 20:247–266, 2006.

[73] Emmanuel Prochasson and Pascale Fung. Rare Word Translation Extraction from Aligned Comparable Documents. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT2011)*, pages 1327–1335, 2011.

[74] Chris Quirk, Arul Menezes, and Colin Cherry. Dependency Treelet Translation: Syntactically Informed Phrasal SMT. In *Proceedings of the 43rd Annual Conference of the Association for Computational Linguistics (ACL 2005)*, pages 271–279, 2005.

[75] Reinhard Rapp. Identifying Word Translations in Non-Parallel Texts. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL 1995)*, pages 320–322, 1995.

[76] Reinhard Rapp. Automatic Identification of Word Translations from Unrelated English and German Corpora. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL 1999)*, pages 519–526, 1999.

[77] Yoon-Hyung Roh, Young-Ae Seo, Ki-Young Lee, and Sung-Kwon Choi. Long Sentence Partitioning using Structure Analysis for Machine Translation. In *Proceedings of the 6th Natural Language Processing Pacific Rim Symposium (NLPRS 2001)*, pages 646–652, 2001.

[78] Antti-Veikko Rosti, Bing Zhang, Spyros Matsoukas, and Richard Schwartz. Expected BLEU Training for Graphs: BBN System Description for WMT11 System Combination Task. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 159–165, 2011.

[79] Helmut Schmid. Probabilistic Part-of-Speech Tagging Using Decision Trees. In *Proceedings of the International Conference on New Methods in Natural Language Processing (NeMLaP 1994)*, pages 44–49, 1994.

[80] Jayaram Sethuraman. A Constructive Definition of Dirichlet Priors. *Statistica Sinica*, 4(2):639–650, 1994.

[81] Li Shao and Hwee Tou Ng. Mining New Word Translations from Comparable Corpora. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING 2004)*, pages 618–624, 2004.

[82] Libin Shen, Jinxi Xu, and Ralph Weischedel. A New String-to-Dependency Machine Translation Algorithm with a Target Dependency Language Model. In *Proceedings of the 46th Annual Conference of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT 2008)*, pages 577–585, 2008.

[83] Kairit Sirts and Tanel Alumäe. A Hierarchical Dirichlet Process Model for Joint Part-of-Speech and Morphology Induction. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2012)*, pages 407–416, 2012.

[84] Partha Pratim Talukdar and Koby Crammer. New Regularized Algorithms for Transductive Learning. In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD 2009)*, pages 442–457, 2009.

[85] Yee Whye Teh, Michael I. Jordan, Matthew J. Beal, and David M. Blei. Hierarchical Dirichlet Processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2006.

[86] Ivan Vulić, Wim De Smet, and Marie-Francine Moens. Identifying Word Translations from Comparable Corpora Using Latent Topic Models. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT 2011)*, pages 479–484, 2011.

[87] Taro Watanabe. Optimized Online Rank Learning for Machine Translation. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2012)*, pages 253–262, 2012.

[88] Taro Watanabe and Eiichiro Sumita. Machine Translation System Combination by Confusion Forest. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT2011)*, pages 1249–1257, 2011.

[89] Dekai Wu and Xuanyin Xia. Learning an English-Chinese Lexicon from a Parallel Corpus. In *Proceedings of the First Conference of the Association for Machine Translation in the Americas (AMTA 1994)*, pages 206–213, 1994.

[90] Hua Wu and Haifeng Wang. Pivot Language Approach for Phrase-Based Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL 2007)*, pages 856–863, 2007.

[91] Hao Zhang, Licheng Fang, Peng Xu, and Xiaoyun Wu. Binarized Forest to String Translation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL 2011)*, pages 19–24, 2011.

[92] Min Zhang, Hongfei Jiang, Aiti Aw, Haizhou Li, Chew Lim Tan, and Sheng Li. A Tree Sequence Alignment-based Tree-to-Tree Translation Model. In *Proceedings of the 46th Annual Conference of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT 2008)*, pages 559–567, 2008.

[93] Ying Zhang and Phil Vines. Using the Web for Automated Translation Extraction in Cross-Language Information Retrieval. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 162–169, 2004.

[94] Xiaojin Zhu and Zoubin Ghahramani. Learning from Labeled and Unlabeled Data with Label Propagation. Technical report, CMU-CALD-02-107, 2002.

[95] Xiaojin Zhu, Zoubin Ghahramani, and John Lafferty. Semi-supervised Learning using Gaussian Fields and Harmonic Functions. In *Proceedings of the 20th International Conference on Machine Learning (ICML 2003)*, pages 912–919, 2003.

[96] 新森 昭宏, 奥村 学, 丸川 雄三, 岩山 真. 手がかり句を用いた特許請求項の構造解析. **情報処理学会論文誌**, 45(3):891–905, 2004.

[97] 熊野 明. 中国語特許翻訳を支援する機械翻訳技術. *Japio YEAR BOOK 2011*, pages 254–257, 2011.