統計数理(2016)
第 64 巻 第 2 号 145–160

特集「統計的言語研究の現在」
[研究詳解]

文に隠れた構文構造を発見する統計モデル

能地 宏†

(受付 2016 年 5 月 4 日；改訂 9 月 8 日；採択 10 月 7 日)

要　　旨

本稿は，自然言語の文法を単語列から自動で抽出する教師なし構文解析について，過去 20 年間に渡る研究の進展について紹介を行う．この研究で本質的に重要となるのは，言語の文法に関するバイアス，もしくは知識をどのようにモデルに組み込むか，という点である．本稿ではこの観点から様々な既存のモデルを比較し，どのような知識を仮定することでどの程度の文法が獲得できるようになったのかについてまとめることで，教師なし構文解析が今後向かうべき方向性についての議論の指針としたい．

キーワード：計算言語学，教師なし構文解析．

1. はじめに

自然言語処理において文の統語構造(木構造)を明らかにする構文解析は最も基礎的かつ重要な技術である．例えば図 1(b)のような依存構造木が得られれば，ここから単語間の意味関係，例えば read の目的語が the book であることが読み取れ，これらが機械翻訳や質問応答で利用される．

本稿では，構文解析に関する研究のうち，特に教師なし構文解析，もしくは文法推定(grammar induction)の問題に関する最近の進展についてまとめる．この問題は自然言語処理が統計的アプローチにシフトし始めた 90 年代初期の頃から存在し，また当時から非常に困難な問題として知られていた(Lari and Young, 1990)．木構造に対するモデルとして文脈自由文法などの簡単なモデルを仮定すれば，そのパラメータ(文法の各書き換えルールに対する重み)は Expectation-Maximization(EM)アルゴリズムを用いて文のみの集合から機械的に計算することができる．しかしながらそのようにして得られた文法は言語学的に正しいとされるものとは大きくかけ離れており，長い間文法の教師なし獲得は不可能であると信じられていた(Manning and Schütze, 1999)．このように一旦停滞しかけていた研究であるが，2004 年の Klein らの研究(Klein and Manning, 2004)によるモデル及び学習法によって再び注目を集め，その後約 10 年間で様々な改良が行われ，現在に至っている．本稿では，特にこの過去およそ 10 年間の進展をまとめることにより，今後の教師なし構文解析の方向性に関する議論の指針を与えたい．

近年行われている研究の多くは，モデルに文法に関する事前知識，もしくは常識をどのように取り入れるか，という点に焦点を当てたものが多い．これは言い換えれば，最小限の労力で新しい言語に対するツリーバンクを構築することを目標とした際に必要な事前知識，もしくは外部知識を明らかにする立場であるといえる．例えば Klein らの研究では本質的には EM アルゴリズムの初期値が最も精度の向上に寄与しており，この初期値は言語学的直感に基づいて設

†奈良先端科学技術大学院大学 情報科学研究科：〒 630–0192 奈良県生駒市高山町 8916–5

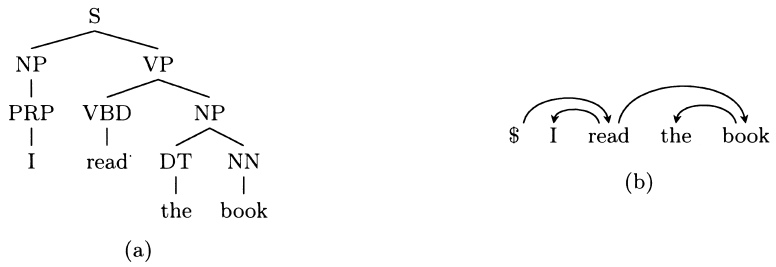


図 1. 構文解析が扱う木構造の例．(a)は句構造木，(b)は依存構造木を表す．$ は常に文頭に置かれる仮想的なノードで文のルート(read)を子に持つ．

計されている(3.1 節)．より最近の研究，例えば Bisk and Hockenmaier (2013)は，名詞と動詞間の言語普遍的な振る舞いを語彙化文法(組合せ範疇文法)の枠組みでモデルに組み込んでいる．この観点から見ると，過去の研究は，どのような知識の入れ方が最も効率的であるかについての試行錯誤であったと捉えることができ，また研究が進展するにつれ，どこまで知識を仮定すればどの程度文法が得られるのかについての知見が蓄積されてきたように思われる．機械学習の立場から別の興味深い問題は，モデルが正しいと仮定した際に最適解をどのように得るか，という問いであるが，この点に関する研究はまだ少ない．本稿では大きく扱わないが，分枝限定法(Gormley and Eisner, 2013)やモーメント法(Hsu et al., 2012)などの適用が検討されてきた．

上記の立場は基本的には工学的な有用性を探求する立場と言えるが，教師なし構文解析は理学的，あるいは哲学的にも興味深い問題であり，特に初期の研究にはそういった立場に立脚したものも見受けられる．例えば Clark (2001)は計算機が文の集合から文法が獲得できることを示すことにより，理論言語学における刺激の貧困(poverty of stimulus) (Chomsky, 1986)に対する経験的な反証が行えると主張している．本稿ではこの点についてはあまり触れないが，言語獲得のモデル化という観点からは両者は切り離せるものではないだろう．例えば上で述べた文法に関する事前知識を利用した学習は，言語の文法がどの程度生得的なのか，という問いに対する示唆を計算言語学の立場から与えるものであると考えられる．

本稿で扱うほぼ全てのモデルは，確率文脈自由文法に対する EM アルゴリズムもしくはその拡張によって説明が行える．その上でほとんどの議論は，どのような文法の枠組みが文の集合のみからの学習に適するか，そしてどのような推論法がより学習を促進させるのに有効か，という点に集約される．本稿ではまず 2 節でこの大きな枠組みを説明した後，その単純な応用である句構造文法の EM アルゴリズムがうまくいかなかったことを述べる．その後 3 節で大きな転換点となった依存構造の学習に焦点を当て，主要な研究をかいつまんで解説する．最後に 4 節で近年注目を集めている言語理論に基づく組合せ範疇文法の教師なし学習について紹介し，5 節でまとめを行う．

2. 確率文脈自由文法と EM アルゴリズム

2.1 確率文脈自由文法

確率文脈自由文法(PCFG)は $G = (N, \Sigma, P, S, \theta)$ の 5 つ組で表現される．ここで (N, Σ, P, S) は一つの文脈自由文法(CFG)であり，N が非終端記号，Σ が終端記号，P が書き換えルールの集合を表す．終端記号は構文木の葉ノードに出現し，非終端記号はそれ以外の内側に出現する記号として区別される．各書き換えルール $r \in P$ は $A \to \beta$ の形をもつ．ここで $A \in N$,

$\beta \in (N \cup \Sigma)^*$(空または記号の列)である．$S \in N$ は特別な文法の開始記号である．図 1(a)はある CFG が入力文 I read the book に対して与える解析例を示しており，NP, VP などが非終端記号，I, read などが終端記号，S $\rightarrow$ NP VP などがルールの例となっている．θ はパラメータであり，各 $r \in P$ に対して確率値を割り当てる．ここで θ_r で r に対する確率を表すと，

$$\forall A \in N, \sum_{A \rightarrow \beta \in P} \theta_{A \rightarrow \beta} = 1$$

が成り立つ．すなわち各非終端記号 A は子の書き換えに関する多項分布をもつ．

PCFG は構文木に対する分布を与える．ある PCFG G のもとで，一つの構文木 z は開始記号 S からの再帰的な書き換えルールの集合とみなすことができるので，その確率は，

$$p(z|\theta) = \prod_{r \in z} \theta_r$$

である．また，ある PCFG G が与えられたとき，入力文 $x = x_1, x_2, \ldots, x_n$(各 x_i は単語)に対する最適な構文木は，動的計画法である CKY アルゴリズム(Kasami, 1965; Younger, 1967)を用いて効率的に計算できる．

$$\hat{z} = \arg \max_{z \in \mathcal{Z}(x)} p(z|\theta)$$

ここで $\mathcal{Z}(x)$ は x に対してあり得る構文木の集合である．

本稿で扱う CFG は全て，ルールの右辺 β の大きさが 1 または 2 のものに限られる．上で述べた CKY アルゴリズム，もしくは以下の内側外側アルゴリズムは，この仮定により大きく簡略化される．

2.2 EM アルゴリズムによる学習

PCFG は隠れマルコフモデル(HMM)の木構造への一般化とみなすことができる．そしてこの観察から，HMM に対する EM アルゴリズムと同じように PCFG に対する EM アルゴリズムを導出することができる．文の集合 $\boldsymbol{x} = x^{(1)}, x^{(2)}, \ldots, x^{(m)}$ が与えられたとき，EM アルゴリズムは次の対数尤度を上昇させるように θ を更新する．

$$L(\theta) = \sum_{x \in \boldsymbol{x}} \log p(x|\theta) = \sum_{x \in \boldsymbol{x}} \log \sum_{z \in \mathcal{Z}(x)} p(x, z|\theta). \tag{2.1}$$

各更新は E ステップと M ステップの二段階からなる．E ステップでは，現在の θ のもとでのルール r の期待値 $e(r)$ を計算する．

$$e(r) \leftarrow \sum_{x \in \boldsymbol{x}} e_x(r),$$

$$e_x(r) \leftarrow \sum_{z \in \mathcal{Z}(x)} p(z|x) f(r, z). \tag{2.2}$$

ここで $e_x(r)$ は文 x における r の期待値，$f(r, z)$ は構文木 z 中でルール r が使われた回数である．M ステップでは，この期待値を正規化することで θ を更新する．

$$\theta_{A \rightarrow \beta} \leftarrow \frac{e(A \rightarrow \beta)}{\sum_{\alpha : A \rightarrow \alpha \in R} e(A \rightarrow \alpha)}$$

ここで最も重要なのは，ルールの期待値 $e_x(r)$ を効率よく計算することである．これには HMM での前向き後ろ向きアルゴリズムとよく似た内側外側アルゴリズム(inside-outside algorithm)が利用できる．以下 $r = A \rightarrow B\ C$，つまり右辺の大きさが 2 のルールを仮定する．またスパン

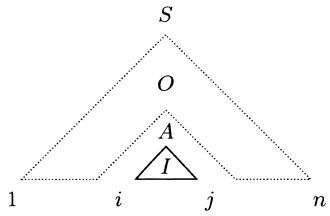


図 2. 内側確率(I)と外側確率(O)の概念図．n を文長，i,j を文中の単語のインデックスとして，各スパン (i,j) 及び非終端記号 A 毎にこれらが計算される．$x_{i:j} = x_i, x_{i+1}, \ldots, x_j$ を表す．

(i,j) で，文中の i 番目の単語から j 番目の単語までの範囲を表す．

まず，$e_x(r)$ を次のように分解しよう．

$$e_x(r) = \sum_{1 \le i \le j \le k \le n} e_x(r,i,j,k). \tag{2.3}$$

$e_x(r,i,j,k)$ は，ルール $r = A \to B\ C$ の B がスパン (i,j) を，C が $(j+1,k)$ を張ることに対する期待値である．これは二値の確率に対する期待値であるから，それが発生する確率自体に等しく

$$e_x(r,i,j,k) = p(r,i,j,k|x,\theta) = \frac{p(r,i,j,k,x|\theta)}{p(x|\theta)}$$

となる．従って，入力 x の周辺確率 $p(x|\theta)$ 及び 各 (r,i,j,k) 毎に $p(r,i,j,k,x|\theta)$ を計算できれば，式(2.3)によって r の期待値が得られる．

内側外側アルゴリズムは，これらの量を動的計画法によって効率的に計算するアルゴリズムである．これは，各スパン (i,j) 及びその根の記号 A 毎に，二つの量 $I[i,j,A]$（内側確率）と $O[i,j,A]$(外側確率)を計算していく．図 2 に概念図を示す．$I[i,j,A]$ は A を開始記号としたときの $w_i, \ldots, w_j$ に対する周辺確率であるのに対し，$O[i,j,A]$ は (i,j,A) の外側の構造に対する周辺確率となっている．

内側確率が全て求まれば，文に対する周辺確率は $p(x|\theta) = I[1,n,S]$ として得られる．

$p(r,i,j,k,x|\theta)$ は若干複雑だが，まずこれは次のように，x が (r,i,j,k) を含む構文木から生成される確率を表すことに注意する．

$$p(r,i,j,k,x|\theta) = \sum_{z \in \mathcal{Z}(x):(r,i,j,k) \in z} p(z,x|\theta).$$

ここで $(r,i,j,k) \in z$ は r が (i,j,k) の位置で構文木 z 中に出現することを表す．そしてこの確率は，次のように (r,i,j,k) の前後で内側確率と外側確率を用いて分解することができる(θ への依存性は省略した)．

$$\begin{aligned} p(r,i,j,k,x) &= p(x_{1:i-1}, A, x_{k+1:n}) \times p(A \to B\ C) \times p(x_{i:j}|B) \times p(x_{j+1:k}|C) \\ &= O(i,k,A) \times \theta_{A \to B\ C} \times I(i,j,B) \times I(j+1,k,C). \end{aligned}$$

r の右辺の大きさが 1 の場合は省略するが，同様の式を導くことができる．以上が学習の概略であるが，内側確率，外側確率の再帰式など，アルゴリズムのより詳細については Manning and Schütze (1999)などの教科書を参照されたい．

2.3 句構造文法の推定

内側外側アルゴリズムによる期待値計算により，PCFG のパラメータ θ は，CFG (N, Σ, P, S) と特定の初期値 $\theta^{(0)}$ を定めれば推定を行うことができる．

90 年代，この考えに基づき図 1(a)のような句構造文法を教師なしで学習する研究がいくつか行われたものの，得られた文法は言語学者の考える正解とは大きくかけ離れていた(Carroll and Charniak, 1992)．この失敗の原因として，次のような点が考えられる．

(1) 第一に EM アルゴリズムは局所探索法であるため，得られる文法は対数尤度(式(2.1))の大域的最適解ではなく局所解だという点である．木構造の探索は範囲が非常に大きく，この局所解の問題が特に問題となる．Carroll and Charniak (1992)はこの影響を調べており，人工データに対して 300 回の試行でランダムに初期化したモデルは全て違う局所解に陥ったと報告している．

(2) もう一つの問題は，句構造文法の恣意性と表現力の弱さである．PCFG の学習において，固定されている情報は開始記号 S 及び観察された終端記号の列(単語もしくは品詞)のみである．ここでの問題は，ある木構造が与えられたとき，終端記号は多くの情報量を持っているものの，これらと木の中間ノードとの結びつきが，木の上方になるほど指数的に失われていくという点である．これは本質的には，句構造文法で扱う非終端記号が単なる抽象的な記号としてしか振る舞わないことに起因する．例えばモデルに $y_1 \to y_2\ y_3$ というルールが存在したとする．ここで y_1 と終端記号との結びつきを考えると，y_1 は $y_2\ y_3$ を通してでしかこれらと関連を持たず，結果結びつきは急速に失われる．

(3) 最後にこれと関連するが，EM アルゴリズムが見つけやすい構造と言語学者が正しいと考える構造の間には隔たりがあることが指摘されている．例えば正解データから教師あり学習で得たモデルを EM アルゴリズムの初期値として使用した場合，学習を進める毎に尤度(式(2.1))は上昇するものの精度は逆に悪化してしまう(Liang and Klein, 2008)．また英語の文集合に対しモデルが見つけやすい典型的な間違いとして，頻度の高い語の並びを句にまとめてしまうという挙動があげられる．例えば，英語では代名詞，動詞という並びが典型的なため，図 1(a)のような構造ではなく，主語である代名詞と動詞が直接句を形成するような文法が学習されやすい(Pereira and Schabes, 1992)．

次節で述べる依存文法の学習は，学習する PCFG の構造に制限を加えることで上記の 2 と 3 の問題を緩和しようとするものだといえる．1 は非凸関数の最適化に起因する問題であるが，これについても初期値の工夫など様々な改善がなされてきた．

初期の EM アルゴリズムを用いた学習で唯一成功したのは，人手で構築した正解データを用いてあり得る句構造のスパンに制約を課す方法(Pereira and Schabes, 1992)であり，これがその後のコーパス主導の教師あり構文解析へと発展していく(Charniak, 1996; Collins, 1997)．また EM 以外の学習方法として Johnson et al. (2007)はモデルのベイズ化及びサンプリング(マルコフ連鎖モンテカルロ法)に基づく推論を試みているが，状況は変わらなかったと報告している．

3. 依存構造の学習へ

PCFG に基づく文法の推定で初めてある程度の成功を収めたのは，依存文法の推定である(図 1(b))．句構造文法が文の構造を名詞句，動詞句など句同士の階層構造によって表現するのに対し，依存文法は単語間の依存関係によって表現する．各依存関係は head(主辞)から dependent (従属辞)に引かれ，多くの場合 dependent が head を修飾する関係となっている．例えば図 1 (b)において，the は book の dependent である．また通常，文全体の head(文のルート)を表

現するために，文頭に仮想的なノード(\$)を用意し，この右の子を文のルートとする．これにより，後に述べる PCFG への変換(図 5)などが簡略化される．

依存構造木と句構造木は一見大きく異なるものの，両者には透過的な関係がある[1)]．例えば図 1(b)において，the は book の左の子であり，the book が一つの小さな部分木，もしくは句を形成しているとみなせる．本節では依存構造の学習の例を示すが，これは，依存構造を表現する PCFG を定義しそのパラメータを推定するということである．具体的には，まず依存構造木に対する生成モデルを定義し，そのモデルを等価な PCFG に変換する．解析の際には，入力文を PCFG で解析し，得られた木から依存構造を復元すれば良い．

3.1 子の数を考慮に入れたモデル

ここでは研究の転換点となった Klein and Manning (2004)のモデルについて述べる．これは dependency model with valence(DMV)と呼ばれている．このモデルの一つの特徴そして限界として，単語ではなく品詞の上に定義されたモデルであるという点が挙げられる．彼らの研究は主に英語で行っており，扱う品詞の数はおよそ 40 程度である．自然言語の単語は数万から数十万以上であるため，これにより学習しなければならないパラメータの次元数が大幅に削減され，学習が行いやすくなる．ただしこの問題設定にすることで，実用的には，解析の前段階でまず品詞を予測する必要が生じる．文法の純粋な教師なし学習を考えた場合，正解の品詞が全ての文に付与されていることを前提とするのは現実的でない．一つの解決策は，まず単語に対する何らかのクラスタリングを行い，品詞の代わりに単語をクラスタで置き換えてモデルを構築することである．この方向性の研究としては，Headden III et al. (2008)や Bisk et al. (2015)などが存在する．

3.1.1 生成モデル

依存構造木に対して考えうるもっとも単純な生成モデルは，単語間の依存関係のみをモデル化したものであろう．この場合，各依存関係は $p_{\mathrm{A}}(d|h, dir)$ で，依存構造木の確率はこの要素の積でモデル化される．ここで $dir \in \{\leftarrow, \rightarrow\}$ は h(head)から d (dependent)への依存関係の方向である．

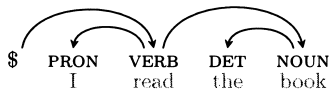
DMV はこのモデルを基本に，木の形を制御する別の要素 $p_{\mathrm{S}}(stop|h, dir, adj)$ を考慮したモデルとなっている．これはベルヌイ分布となっており，$stop \in \{\mathrm{STOP}, \neg\mathrm{STOP}\}$，そして $adj \in \{\mathrm{TRUE}, \mathrm{FALSE}\}$ が条件付け変数で，h が dir 方向に既に子を生成しているかどうかを判断し，まだしていなければ TRUE となる．図 3 に具体的なパラメータの例を示す．NOUN，VERB はそれぞれ名詞，動詞を表す．

具体的なモデルの生成過程を見るため，図 4 に DMV がある依存構造木に与える生成確率を示す．DMV では左側と右側の子はそれぞれ独立に生成される．h が各 d を生成する確率は $p_{\mathrm{S}}(\neg\mathrm{STOP}|h, \cdot, \cdot)$ と $p_{\mathrm{A}}(d|h, \cdot)$ の積で与えられる．最後に，各方向に子の生成を停止する $p_{\mathrm{S}}(\mathrm{STOP}|h, \cdot, \cdot)$ をかけ合わせる．

このモデルは英語の語順を多分に考慮に入れつつ設計されている．Balack Obama talked ... など，英語では NOUN NOUN VERB という並びはよく現れるが，最初に述べた p_{A} のみのモデルでは，NOUN ← VERB に対する重みが強くなった場合，二つの NOUN はどちらも VERB の子になってしまう．ここで正解の解析は NOUN ← NOUN ← VERB である．これに対し，もし $p_{\mathrm{S}}(stop|h, dir, adj)$ が正しく学習されれば，$p_{\mathrm{S}}(\mathrm{STOP}|\mathrm{VERB}, \leftarrow, \mathrm{FALSE})$ の値が大きくなり，英語の動詞(VERB)が左側に通常子を一つ，すなわち動詞に対する主語しか持たないことをモデル化できる．モデルはまた，英語の冠詞(DET)や代名詞(PRON)が通常左右に子を持たないことも捉えらえる．このためには，図 4(b)の 3 行目と 5 行目のパラメータがどれも高くなるように学習がされていれば良い．

パラメータ	具体例	説明
$p_S(stop \mid h, dir, adj)$	$p_S(\neg\text{STOP} \mid \text{VERB}, \rightarrow, \text{TRUE})$	VERB が右側に子を持たない状態から，一つ子を生成する
$p_A(d \mid h, dir)$	$p_A(\text{NOUN} \mid \text{VERB}, \rightarrow)$	右側の具体的な子として NOUN を選ぶ

図 3. DMV の二種類のパラメータとその具体例.



(a)

$$
\begin{aligned}
p(t) = {} & \underbrace{p_A(\text{VERB} \mid \$, \rightarrow)}_{\text{文のルートを VERB に}} \cdot \underbrace{p_S(\neg\text{STOP} \mid \text{VERB}, \rightarrow, \text{TRUE})}_{\text{VERB の最初の右の子を NOUN に}} \cdot \underbrace{p_A(\text{NOUN} \mid \text{VERB}, \rightarrow)}_{} \cdot \underbrace{p_S(\text{STOP} \mid \text{VERB}, \rightarrow, \text{FALSE})}_{\text{VERB の右の子の生成を停止}} \cdot \\
& \underbrace{p_S(\neg\text{STOP} \mid \text{VERB}, \leftarrow, \text{TRUE})}_{\text{VERB の最初の左の子を PRON に}} \cdot \underbrace{p_A(\text{PRON} \mid \text{VERB}, \leftarrow)}_{} \cdot \underbrace{p_S(\text{STOP} \mid \text{VERB}, \leftarrow, \text{FALSE})}_{\text{VERB の左の子の生成を停止}} \cdot \\
& \underbrace{p_S(\text{STOP} \mid \text{PRON}, \rightarrow, \text{TRUE}) \cdot p_S(\text{STOP} \mid \text{PRON}, \leftarrow, \text{TRUE})}_{\text{PRON は左右に子を持たない}} \cdot \\
& \underbrace{p_S(\text{STOP} \mid \text{NOUN}, \rightarrow, \text{TRUE})}_{\text{NOUN は右に子を持たない}} \cdot \underbrace{p_S(\neg\text{STOP} \mid \text{NOUN}, \leftarrow, \text{TRUE})}_{\text{NOUN の最初の左の子を DET に}} \cdot \underbrace{p_A(\text{DET} \mid \text{NOUN}, \leftarrow)}_{} \cdot \underbrace{p_S(\text{STOP} \mid \text{NOUN}, \rightarrow, \text{FALSE})}_{\text{NOUN の左の子の生成を停止}} \cdot \\
& \underbrace{p_S(\text{STOP} \mid \text{DET}, \rightarrow, \text{TRUE}) \cdot p_S(\text{STOP} \mid \text{DET}, \leftarrow, \text{TRUE})}_{\text{DET は左右に子を持たない}}
\end{aligned}
$$

(b)

図 4.（a）品詞に対する依存構造の例（灰色の単語は観測されない）．（b）この依存構造木の DMV のもとでの生成確率.

3.1.2 パラメータの学習

このモデルで特に重要なのは，パラメータの学習が 2.2 節で述べた内側外側アルゴリズムに基づく EM アルゴリズムで行えるという点である．これは DMV による生成過程が等価な PCFG によって表現可能であるという観察に基づく．図 5 に，PCFG の各ルールと DMV のパラメータの対応，そして CFG での解析例を示す．全ての書き換えルールは DMV のパラメータと一対一対応があり，これは PCFG となっている．本 PCFG は常に右の子を全て生成し，その後左の子を生成するが，この方向の固定により，CFG の解析と依存構造木とが一対一に対応する．

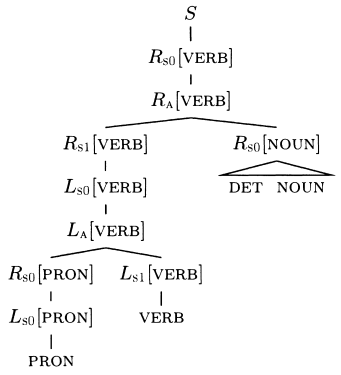
3.1.3 議論

英語での品詞列からの実験で，DMV は英語の基本的な語順を学習できることが示された．評価には，学習したモデルが人手で構築した正解の依存構造木の依存関係をどれだけ復元できたか，という精度を用い，これを文単位でなく単語単位で計算する．DMV は長さ 10 単語以下の文のみで評価した場合におよそ 44%の精度を達成した．英語では，非常に単純なベースラインとして，常に右隣の単語を子とすることで精度 34%が達成できることが知られていた．DMV はこの数値を初めて上回り，PCFG に対する EM アルゴリズムで意味のある構造が学習できることが示された.

評価の問題について少し触れておく．依存構造の教師なし解析の評価は様々な問題点が孕む

書き換えルール	DMV のパラメータ
$S \to R_{\mathrm{s}0}[h]$	$p_{\mathrm{A}}(h\|\$, \to)$
$R_{\mathrm{s}0}[h] \to L_{\mathrm{s}0}[h]$	$p_{\mathrm{s}}(\mathrm{STOP}\|h, \to, \mathrm{TRUE})$
$R_{\mathrm{s}0}[h] \to R_{\mathrm{A}}[h]$	$p_{\mathrm{s}}(\neg\mathrm{STOP}\|h, \to, \mathrm{TRUE})$
$R_{\mathrm{A}}[h] \to R_{\mathrm{s}1}[h]\ R_{\mathrm{s}0}[d]$	$p_{\mathrm{A}}(d\|h, \to)$
$R_{\mathrm{s}1}[h] \to L_{\mathrm{s}0}[h]$	$p_{\mathrm{s}}(\mathrm{STOP}\|h, \to, \mathrm{FALSE})$
$R_{\mathrm{s}1}[h] \to R_{\mathrm{A}}[h]$	$p_{\mathrm{s}}(\neg\mathrm{STOP}\|h, \to, \mathrm{FALSE})$
$L_{\mathrm{s}0}[h] \to h$	$p_{\mathrm{s}}(\mathrm{STOP}\|h, \leftarrow, \mathrm{TRUE})$
$L_{\mathrm{s}0}[h] \to L_{\mathrm{A}}[h]$	$p_{\mathrm{s}}(\neg\mathrm{STOP}\|h, \leftarrow, \mathrm{TRUE})$
$L_{\mathrm{A}}[h] \to R_{\mathrm{s}0}[d]\ L_{\mathrm{s}1}[h]$	$p_{\mathrm{A}}(d\|h, \leftarrow)$
$L_{\mathrm{s}1}[h] \to h$	$p_{\mathrm{s}}(\mathrm{STOP}\|h, \leftarrow, \mathrm{FALSE})$
$L_{\mathrm{s}1}[h] \to L_{\mathrm{A}}[h]$	$p_{\mathrm{s}}(\neg\mathrm{STOP}\|h, \leftarrow, \mathrm{FALSE})$

(a)



(b)

図 5.（a）DMV の PCFG での表現．（b）この文法による図 4(a)の依存構造木の変換．

未解決問題の一つとして知られている(Schwartz et al., 2011; Bisk and Hockenmaier, 2013)．例えば，上で挙げた NOUN NOUN VERB という列を考えると，複合名詞 NOUN NOUN の head がどちらか，というのは恣意的にしか決められないだろう．依存構造の分析からこのような人手による恣意性を取り除くことはできず，一つの正解データの基準をどれだけ復元できるかという評価がどれほど意味があるものなのかは疑わしい．本来は後段の処理で，つまり導出した依存構造木が，機械翻訳や情報抽出などの応用の精度向上にどれほど寄与するかで評価を行うことが客観的な価値判断に有効と考えられるが，そのような研究はほとんど見られない．

最後に DMV のもう一つの重要な貢献である EM の初期値について述べる．言語の重要な特性として，各依存関係の単語間の距離は近いものが好まれる，ということが知られている(Gildea and Temperley, 2010)．DMV ではこの直感を初期値を通じてモデルに埋め込んでいる．具体的には，最初に p_{A} を単語間の距離に反比例する量で定義した後，この正規化されていない分布の上で E ステップを行い，続く M ステップで初期値を得る．実際にはこの初期化が非常に重要であり，後の研究で，この初期化を行わないと精度が 20%以上低下(44%から 21%)することが指摘された(Gimpel and Smith, 2012)．

3.2 学習の工夫

DMV の一定の成功以降，これに対する様々な改良が提案された．1 節で述べたように，多くの研究は，文法に関する様々な仮定を置き，その影響を調べたものが多い．ここではいくつかの代表的な研究をかいつまんで説明する．他の方向性として，正解の品詞が付与されている前提で，品詞間のルールをモデルに埋め込む研究が存在する．一般にこれらのほうがより高い精度を示すが，そちらについては別に 3.3 節でまとめる．

Smith and Eisner (2006)は DMV が初期値を通じて取り込んでいる，短い距離の依存関係を好むという文法の傾向をより明示的にモデルに組み込んでいる．彼らのモデルでは DMV の $p_{\mathrm{A}}(d|h, dir)$ を次で置き換える．

$$p'_{\mathrm{A}}(d|h, dir) \propto p_{\mathrm{A}}(d|h, dir) \cdot e^{\beta|d-h|}$$

ここで $|d-h|$ は，h と d の間に存在する単語の数(距離)である．β (≤ 0) がハイパーパラメータとなっており，これがバイアスの強さを決定する．彼らは更にこの強さを学習中に減衰させるアニーリング法を提案しており，この学習法が様々な言語で有効であることを検証している．

Cohen and Smith (2009)と Berg-Kirkpatrick et al. (2010)は，どちらもモデルは DMV であるが，各パラメータを更に別の対数線形モデルで表現することで，パラメータ同士に相関を持たせている．これらのモデルでは，入力の品詞が与えられたとき，その品詞に対するより粗いカテゴリが利用できることを前提とする．例えば Berg-Kirkpatrick et al. (2010)では，$p_{\text{A}}(d|h, dir)$ を次のようにモデル化する．

$$p'_{\text{A}}(d|h, dir) \propto \exp(\boldsymbol{w} \cdot \boldsymbol{f}(d, h, dir))$$

$\boldsymbol{w}$ は対数線形モデルの重みベクトル，$\boldsymbol{f}$ は DMV のパラメータ毎に特徴ベクトルを抽出する関数である．英語の品詞体系では，代名詞と固有名詞は異なる品詞として区別されるが，両者の振る舞いは似ていることが想定できる．この直感は例えば，$\boldsymbol{f}$ に品詞が粗い名詞に属するか判定する要素を持たせることで，名詞に属する品詞間でパラメータのゆるい相関を持たせることができ，モデル化ができる．Cohen and Smith (2009)はほぼ似た枠組みを，ベイズモデルの事前分布(shared logistic normal prior)によって実現している．実験ではどちらも英語で 63%程度を示すことが分かっており，DMV と比べて大きな改善が行われた．なお，これまで述べたモデルは全て Klein and Manning (2004)の EM の初期値を利用していることに注意する．

Mareček and Straka (2013)は現時点で，品詞間のルールを直接用いない手法の中では最高精度を持つモデルである．これは，依存文法の dependent に対する直感をうまく大規模データから取り出しモデルに組み込んだ研究といえる．彼らが着目したのは句の削減可能性(reducibility)である(Mareček and Žabokrtský, 2012)．彼らは，別の語の dependent になる句は，その句を削除してもしばしば文法的に正しいという性質に着目し，Wikipedia の記事を用いて各品詞 n グラム毎に reducibility を計算，それを DMV の停止確率 p_{s} の計算に組み込んだ．彼らはこのように大規模データからの統計量をうまく利用することで，初期値を工夫せずとも学習がうまくいくことを報告している．

ここまでは，依存構造もしくは品詞に対する言語学的直感をモデルに組み込んだものといえるが，その他様々な学習上の工夫が提案されている．例えば Spitkovsky et al. (2010)は，EM の学習中に短い文から始め徐々に長い文を取り入れる方法，Headden et al. (2009)は大量のランダムな初期化で数回の EM の試行を行い，尤度の高いものを選択する方法，Blunsom and Cohn (2010)は DMV の文法(図 5)を CFG でなく木置換文法(tree substitution grammar)でモデル化することで，CFG の局所性を緩和することに成功している．

3.3 品詞間のルールの組み込み

本節では Naseem et al. (2010)を中心とした，品詞間のルールに対して弱い教師情報を組み込んだモデルについて紹介する．彼女らの手法は，事後分布正則化(posterior reguralization) (Ganchev et al., 2010)に基づき，名詞は動詞の子となりやすい，形容詞は名詞の子となりやすい，といった品詞間の直接的な関係をモデルに組み込む．

EM アルゴリズムは，隠れ変数 $\boldsymbol{z}$ の事後分布の更新(E ステップ)，パラメータ θ の更新(M ステップ)を交互に行う最適化と見なすことができる．まず次のように式(2.1)の下界が導出できる．

$$L(\theta) = \sum_{x \in \boldsymbol{x}} \log \sum_{z \in \mathcal{Z}(x)} p(x, z|\theta) = \sum_{x \in \boldsymbol{x}} \log \sum_{z \in \mathcal{Z}(x)} q(z) \frac{p(x, z|\theta)}{q(z)}$$
$$\geq \sum_{x \in \boldsymbol{x}} \sum_{z \in \mathcal{Z}(x)} q(z) \log \frac{p(x, z|\theta)}{q(z)} = F(q, \theta)$$

ここで二行目への変換は Jensen の不等式を用いた．通常の EM アルゴリズムは，下界 $F(q, \theta)$ を q, θ について交互に最適化する．E ステップでは，

$$q(z) \leftarrow \arg\max_{q(z)} F(q,\theta) = \arg\min_{q(z)} \mathrm{KL}(q(z)||p(z|x,\theta)) = p(z|x,\theta), \tag{3.1}$$

つまり，現在の θ を元に z の事後分布を決める．M ステップでは，

$$\theta \leftarrow \arg\max_{\theta} F(q,\theta)$$

を求めるが，PCFG のような多項分布の場合，これは式(2.2)のような期待値の正規化に帰着する．

事後分布正則化は，上記の E ステップを次のように変化させる．

$$q(z) \leftarrow \arg\max_{q(z)\in\mathcal{Q}(x)} F(q,\theta) = \arg\min_{q(z)\in\mathcal{Q}(x)} \mathrm{KL}(q(z)||p(z|x,\theta)).$$

つまり，事後分布 $q(z)$ の範囲を，特定の空間 $\mathcal{Q}(x)$ に制限する．これは式(3.1)のように閉じた形では解けないが，$\mathcal{Q}$ に凸空間を仮定することで最適解を求めることができる．Naseem et al. (2010) は $\mathcal{Q}$ として，$E_q[f(z)] \leq b$ という制約を用いている．ここで f は依存構造木 z が与えられたとき，事前に定めた品詞間のルールに**属さない**依存関係の割合，b はそのようなルールに属さない依存関係の割合の許される最小値を決めるパラメータである．

このルールを定める $f(z)$ がモデルの振る舞いを決定する．彼女らの実験では，名詞，動詞，形容詞など基本的な品詞に対して，NOUN→VERB などのルール(方向は定めない)を計 13 種類記述し，上記の $\mathcal{Q}$ に対して，b の値を 0.2，つまり 8 割の依存関係が(期待値の上で)定めたルールを満たす必要があると定め，学習を行った．

3.2 節で述べてきた様々な拡張は基本的にヒューリスティックスであり，特定の言語では逆に精度が悪化するなど，効果も言語によっては限定的なものが多かった．それに対しこの手法は，様々な言語を通じて，10 単語以下の文に対して 60%–70%の精度という安定した結果を示した．本研究で示されたのは，入力文の品詞が完全に同定されているという状況であれば，品詞間のルールをモデルに組み込むことで安定した精度が実現できる，という点である．ただしこの仮定が現実的なものかについては疑問が残る．品詞への依存性を高めるほど，高精度の品詞解析器を用意しなければ精度が達成されない，ということを意味するからである．4 節で紹介する Bisk and Hockenmaier (2013)のモデルはこの点を緩和したものといえ，彼らはより少ない言語学上の仮定から Naseem et al. (2010)と同程度の精度を達成することに成功している．

Naseem et al. (2010)に関連する研究として，Grave and Elhadad (2015)は似たアイデアを識別クラスタリング(Xu et al., 2005)の枠組みに適用し，教師なし構文解析をその上で定式化することで生成モデルによるアプローチよりも高い精度を実現できることを示している．

4. 組合せ範疇文法の学習

2.3 節で，句構造文法の EM アルゴリズムによる学習は，句構造の各記号の持つ意味が少ないためうまくいかなったことを述べた．組み合わせ範疇文法(CCG) (Steedman, 2000)などの語彙化文法はこれに対し，各非終端記号は統語的な振る舞いを決めるという点で意味を持っており，任意の記号ではない．この点に着目し，近年，少量の手がかりを人手で与えることでこれらの文法を学習する手法が研究され始めている．

図 6 に CCG による構文解析の例を示す．CCG の解析の基本単位はカテゴリであり，N や S\S などが属する．図では S → N S\N などのルールが存在するが，これは S\N の機能により決まる振る舞いで，このカテゴリは左側の別の N と結合することで S になるという意味を持つ．N/N は右側の N と結合し N になるという意味を持つ．CCG は多数のカテゴリと少数のこのような結合ルール(高々 10 個程度)からなる文法となっている．

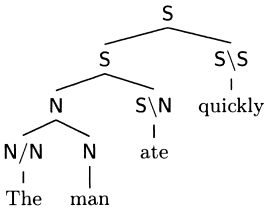


図 6. CCG による構文木.

The	man	ate	quickly
DT	NNS	VBD	RB
N/N	**N**, S/S	**S**, N\N	S\S
(S/S)/(S/S)	(N\N)/(N\N)	S\N	(N\N)\(N\N)
	(N/N)\(N/N)	(S/S)\(S/S)	
		(S\S)/(S\S)	

図 7. Bisk and Hockenmaier (2013)での品詞毎のカテゴリ候補の収集例. DT, NNS, VBD, RB は品詞である. 太字のカテゴリが最初に与えられる知識である.

CFG と異なり，CCG では各単語に割り当てられたカテゴリが統語上の大きな意味を持つ．例えば英語で read などの他動詞には (S\N)/N というカテゴリが割り当てられるが，これは右側の N(目的語)と結合し S\N となり，更に左側に N(主語)をとる，という情報が埋め込まれている．

Bisk and Hockenmaier (2013)はこのような CCG 木に対する生成モデルを提案し，これを教師なし学習することで多数の言語でこれまでの最高精度を達成することを報告している．評価に際しては，CCG の構文木はカテゴリの情報を読み取ることで依存構造木に変換できることを利用する．例えば N/N や S\S など，同じカテゴリを並べたカテゴリは修飾語として振舞うので結合した別の語の dependent とできる．

この手法で鍵となっているのは，訓練中の各単語に対するカテゴリの候補の与え方である．CCG などの語彙化文法は，入力文の各単語のカテゴリが定まると，その上に構築されうる構文木はほぼ決定されるという特徴を持つ(Matsuzaki et al., 2007; Lewis and Steedman, 2014)．言い換えると，統語上のほとんど全ての曖昧性は単語に対するカテゴリ割り当てに集約され，他の部分の曖昧性はほとんど存在しない．Bisk らは CCG のこの特性を，入力の品詞毎にあり得るカテゴリに制約をかけることで効率的に抽出している．彼らが前提として与える知識は，1)文のルートは動詞または名詞であること，そして 2)名詞は動詞の目的語となること，の二点である．この知識をモデルに与えるため，次のような方法で学習を始める前段階として，各品詞毎にあり得るカテゴリに制限をかける．まず，動詞に属する品詞のみに S のカテゴリを許し，名詞に属する品詞のみに N のカテゴリを許す．その後，この情報を元に，他の品詞のカテゴリ候補を拡大していく．例えば，訓練文中で S が割り当てられた単語(動詞)の左に隣接する品詞には S/S(S を左から修飾する)，右に隣接する品詞には S\S(S を右から修飾する)というカテゴリを候補として加える．このような処理を順次繰り返し，品詞毎にカテゴリの候補を拡大していく．図 7 に，さきほどの例文でのカテゴリ候補の拡大例を示す．図では例えば，VBD(動詞の過去形)にまず S が割り当てられ，また左に名詞(N)が存在することから S\N が追加され，更に NNS(名詞)に対して (N\N)/(N\N) という複雑なカテゴリが右側の VBD に対する N\N を基に生成されることなどが見てとれる．

この前処理の後，品詞を入力として，生成モデルのパラメータを変分ベイズ法で推定する．この学習の際に，文全体を張るカテゴリは S(動詞が存在しない場合 N)に制限される．この制限が本質的に重要であり，これによって，実質的に文のルートが動詞であるという制限をモデルに与え，効率的な学習を可能としている．

以上が大まかな枠組みであるが，その後の研究で Bisk and Hockenmaier (2015)はこのモデルを様々な方法で拡張し，詳細なエラー分析を行っている．また Bisk et al. (2015)では，この学習の枠組みが教師なし品詞推定の出力に対しても適用可能であることを示している．通常教師なし品詞推定は単語のクラスタリングを行うものであるため，各カテゴリが名詞に属するのか，形容詞に属するのか，などは分からない．彼らはこれに対し，手法が名詞と動詞の二つの品詞の同定にしか依存していないことから，これらを人手で選定することで最小の労力で手法が適用できることを示している．

Bisk and Hockenmaier (2015)のモデルの拡張，および分析は非常に丁寧に行われており，現時点での教師なし構文解析の限界を示しているものともいえるだろう．彼らの分析によると，現時点で解けていない問題の多くは，単語の**意味**に起因する問題であるという．例えばモデルは “I gave her a gift” という文に対し，“her a gift” を一つの名詞句とする判断をしたと報告している．英語の文法を知らない学習者からすると，“her” が “a gift” を修飾する形容詞と働く可能性も排除できないだろう．これに対し正しい構造，つまり gave が右に目的語を二つとるという構造を得るためには，モデルがこちらを好むような何らかの仕組みもしくはバイアスを外部から組み込む必要があるのではないかと考えられる．

最後に，CCG に基づく他の関連研究についても紹介しておきたい．Garrette et al. (2015)は，品詞を用いずに単語を直接入力とする CCG の学習を提案している．彼らは品詞の情報に頼る代わりに，いくつかの単語に対し，正解のカテゴリが付与された辞書の存在を仮定する．ここから EM 的な学習により，他の単語についてのカテゴリも順次学習することで，高精度を達成できることを示している．二つのアプローチの本質的な違いは，文法に対する事前知識の与え方である．彼らは辞書を利用するが，辞書の構築は人手がかかる作業であることを考えると，Bisk らの手法のほうが汎用性は高いといえるだろう．ただし，先に述べた gave の意味などの問題は，本手法のような直接的な知識の与え方によって解決できる可能性が高い．

5. おわりに

過去 20 年間の間の教師なし構文解析の進展について概説した．90 年代の単純な句構造の学習は失敗に終わったが，その後 Klein and Manning (2004)の品詞に基づく依存構造の学習で研究の方向性を示し，それが様々な方向で拡張された．依存構造の学習において特に重要な研究といえるのは，人手で与えた品詞間のルールを利用する(Naseem et al., 2010)であろう．これは精度の上では CCG に基づく Bisk and Hockenmaier (2013)とほぼ同等であり，現在の品詞または単語列からのみの学習のアプローチの限界を示しているともいえる．どちらも多言語を通して，精度は 6 割もしくは 7 割で頭打ちである．これは，教師なし構文解析は少量の言語学的仮定をモデルに課すことで言語毎の基本的語順を発見できるようにはなったが，単語の意味に起因するようなより深い分析が必要な構文については解くことが難しい状況であることを示唆している．

我々の目標は，そのような深い分析が必要な解析も扱える解析器を，最小の人手の労力によって構築する手段を確立することである．つまり，あらゆるタスク，言語について教師データである構文木などを付与するのは現実的でなく，より効率的な教師情報の与え方を確立したいのである．

過去 20 年間の教師なし構文解析の研究によって，表層的な入力のみからでは学習に限界があることが明らかになった．つまり，より実用的なシステムのためには，何らかの方法で外部知識をモデルに与えてやる必要がある．

このための一つの方向性は，構文解析を別のタスクのための隠れ変数とみなして学習を行う方法であろう．例えば Liang et al. (2011) は，質問応答という非常に限られたドメインに対してではあるが，質問文とその答えのみを入力として，質問文に対するデータベースのクエリを教師なしで学習するモデルを得ることに成功している．このようなアプローチとここで述べた教師なし構文解析との最大の違いは，教師なし構文解析では学習中の構造に対しフィードバックが与えられないという点である．つまり，モデルは構造を探索するが，その構造の言語的良し悪しは全く判断することができない．何らかのフィードバックが与えられれば，それが学習中に必要なバイアスとなりうる．もちろん，このような方法の制約は，モデル化や学習法がタスクに依存し汎用性が低下するという点である．汎用性を残しつつも，外部知識を効率的に取り入れ，深い分析も含めて教師なしに近い状況で学習を行える枠組みを探求するというのが，今後の研究の最大の課題であると考えている．

注．

1) 本稿では以後，依存構造木として図 1(b) のような依存関係の矢印の間に交差が存在しない構造を仮定する．依存構造木の PCFG への変換はこのような制限のもとにおいてのみ可能となる．実際の言語には交差を含む構造も出現するが，多言語にわたってそのような構造の頻度は小さいことが知られている (Kuhlmann, 2013)．

参 考 文 献

Berg-Kirkpatrick, T., Bouchard-Côté, A., DeNero, J. and Klein, D. (2010). Painless unsupervised learning with features, *Proceedings of Human Language Technologies*: *The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 582–590, Los Angeles, California.

Bisk, Y. and Hockenmaier, J. (2013). An HDP model for inducing combinatory categorial grammars, *Transactions of the Association for Computational Linguistics*, **1**, 75–88.

Bisk, Y. and Hockenmaier, J. (2015). Probing the linguistic strengths and limitations of unsupervised grammar induction, *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing* (*Volume 1*: *Long Papers*), 1395–1404, Beijing, China.

Bisk, Y., Christodoulopoulos, C. and Hockenmaier, J. (2015). Labeled grammar induction with minimal supervision, *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing* (*Volume 2*: *Short Papers*), 870–876, Beijing, China.

Blunsom, P. and Cohn, T. (2010). Unsupervised induction of tree substitution grammars for dependency parsing, *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, 1204–1213, Cambridge, Massachusetts.

Carroll, G. and Charniak, E. (1992). Two experiments on learning probabilistic dependency grammars from corpora, *Working Notes of the Workshop Statistically-based NLP Techniques*, 1–13, AAAI Press, Palo Alto, California.

Charniak, E. (1996). Tree-bank grammars, *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, 1031–1036.

Chomsky, N. (1986). *Knowledge of Language. Its Nature, Origin, and Use*, Praeger Publications, New York.

Clark, A. (2001). *Unsupervised Language Acquisition*: *Theory and Practice*, Ph.D. Thesis, School of Cognitive and Computing Sciences, University of Sussex.

Cohen, S. and Smith, N. A. (2009). Shared logistic normal distributions for soft parameter tying in unsupervised grammar induction, *Proceedings of Human Language Technologies*: *The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 74–82, Boulder, Colorado.

Collins, M. (1997). Three generative, lexicalised models for statistical parsing, *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, 16–23, Madrid, Spain.

Ganchev, K., Graa, J., Gillenwater, J. and Taskar, B. (2010). Posterior regularization for structured latent variable models, *Journal of Machine Learning Research*, **11**, 2001–2049.

Garrette, D., Dyer, C., Baldridge, J. and Smith, N. (2015). Weakly-supervised grammar-informed bayesian ccg parser learning, *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence* (*AAAI-15*), Austin, Texas.

Gildea, D. and Temperley, D. (2010). Do grammars minimize dependency length?, *Cognitive Science*, **34**(2), 286–310.

Gimpel, K. and Smith, N. A. (2012). Concavity and initialization for unsupervised dependency parsing, *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics*: *Human Language Technologies*, 577–581, Montréal, Canada.

Gormley, M. R. and Eisner, J. (2013). Nonconvex global optimization for latent-variable models, *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics* (*Volume 1*: *Long Papers*), 444–454, Sofia, Bulgaria.

Grave, E. and Elhadad, N. (2015). A convex and feature-rich discriminative approach to dependency grammar induction, *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing* (*Volume 1*: *Long Papers*), 1375–1384, Beijing, China.

Headden III, W. P., McClosky, D. and Charniak, E. (2008). Evaluating unsupervised part-of-speech tagging for grammar induction, *Proceedings of the 22nd International Conference on Computational Linguistics* (*Coling 2008*), 329–336, Manchester, U.K.

Headden III, W. P., Johnson, M. and McClosky, D. (2009). Improving unsupervised dependency parsing with richer contexts and smoothing, *Proceedings of Human Language Technologies*: *The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 101–109, Boulder, Colorado.

Hsu, D. J., Kakade, S. M. and Liang, P. S. (2012). Identifiability and unmixing of latent parse trees, *Advances in Neural Information Processing Systems*, **25** (eds. F. Pereira, C. J. C. Burges, L. Bottou and K. Q. Weinberger), 1511–1519, Curran Associates, Inc., Redhook, New York.

Johnson, M., Griffiths, T. and Goldwater, S. (2007). Bayesian inference for PCFGs via Markov chain Monte Carlo, *Proceedings of Human Language Technologies 2007*: *The Conference of the North American Chapter of the Association for Computational Linguistics*, 139–146, Rochester, New York.

Kasami, T. (1965). An efficient recognition and syntax-analysis algorithm for context-free languages, Technical Report, AFCRL-65-758, Air Force Cambridge Research Lab., Cambridge, Massachusetts.

Klein, D. and Manning, C. (2004). Corpus-based induction of syntactic structure: Models of dependency and constituency, *Proceedings of the 42nd Meeting of the Association for Computational Linguistics* (*ACL'04*), *Main Volume*, 478–485, Barcelona, Spain.

Kuhlmann, M. (2013). Mildly non-projective dependency grammar, *Computational Linguistics*, **39**(2),

355–387.

Lari, K. and Young, S. (1990). The estimation of stochastic context-free grammars using the inside-outside algorithm, *Computer Speech & Language*, **4**(1), 35–56.

Lewis, M. and Steedman, M. (2014). A* CCG parsing with a supertag-factored model, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing* (*EMNLP*), Doha, Qatar.

Liang, P. and Klein, D. (2008). Analyzing the errors of unsupervised learning, *Proceedings of ACL-08*: *HLT*, 879–887, Columbus, Ohio.

Liang, P., Jordan, M. and Klein, D. (2011). Learning dependency-based compositional semantics, *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*: *Human Language Technologies*, 590–599.

Manning, C. D. and Schütze, H. (1999). *Foundations of Statistical Natural Language Processing*, The MIT Press, Cambridge, Massachusetts.

Mareček, D. and Straka, M. (2013). Stop-probability estimates computed on a large corpus improve unsupervised dependency parsing, *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics* (*Volume 1*: *Long Papers*), 281–290, Sofia, Bulgaria.

Mareček, D. and Žabokrtský, Z. (2012). Exploiting reducibility in unsupervised dependency parsing, *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 297–307, Jeju Island, Korea.

Matsuzaki, T., Miyao, Y. and Tsujii, J. (2007). Efficient HPSG parsing with supertagging and CFG-filtering, *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence* (*IJCAI-07*), 1671–1676.

Naseem, T., Chen, H., Barzilay, R. and Johnson, M. (2010). Using universal linguistic knowledge to guide grammar induction, *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, 1234–1244, Cambridge, Massachusetts.

Pereira, F. and Schabes, Y. (1992). Inside-outside reestimation from partially bracketed corpora, *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics*, 128–135, Newark, Delaware, U.S.A.

Schwartz, R., Abend, O., Reichart, R. and Rappoport, A. (2011). Neutralizing linguistically problematic annotations in unsupervised dependency parsing evaluation, *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*: *Human Language Technologies*, 663–672, Portland, Oregon, U.S.A.

Smith, N. A. and Eisner, J. (2006). Annealing structural bias in multilingual weighted grammar induction, *Proceedings of the International Conference on Computational Linguistics and the Association for Computational Linguistics* (*COLING-ACL*), 569–576, Sydney, Australia.

Spitkovsky, V. I., Alshawi, H. and Jurafsky, D. (2010). From baby steps to leapfrog: How "less is more" in unsupervised dependency parsing, *Human Language Technologies*: *The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 751–759, Los Angeles, California.

Steedman, M. (2000). The Syntactic process, *Language, Speech, and Communication*, MIT Press, Cambridge, Massachusetts.

Xu, L., Neufeld, J., Larson, B. and Schuurmans, D. (2005). Maximum margin clustering, *Advances in Neural Information Processing Systems*, **17**, 1537–1544, MIT Press, Cambridge, Massachusetts.

Younger, D. H. (1967). Recognition and parsing of context-free languages in time n^3, *Information and Control*, **10**(2), 189–208.

160 Proceedings of the Institute of Statistical Mathematics Vol. 64, No. 2, 145–160 (2016)

Statistical Models to Induce Latent Syntactic Structures

Hiroshi Noji

Graduate School of Information Science, Nara Institute of Science and Technology

This article describes the advancement of unsupervised syntactic parsing in the past 20 years. Unsupervised parsing aims to obtain the grammar of the language automatically from the input sentences without manually created syntactic trees. The essential point in this task is how to exploit the bias or knowledge of the grammar of the language. In this article, we compare several existing approaches from this perspective and discuss what kind of information we should provide to the model and what can be learned from such knowledge, to guide the future research direction on this area.

Key words: Computational linguistics, unsupervised syntactic parsing.