

係り受け構造との同時予測による A* CCG 解析

吉川 将司 能地 宏 松本 裕治
 奈良先端科学技術大学院大学 情報科学研究科

{yoshikawa.masashi.yh8, noji, matsu}@is.naist.jp

1 はじめに

組み合わせ範疇文法 (Combinatory Categorical Grammar; CCG) [12, 15] は、語彙化文法の一つであり、文の各語に付与された、周囲の語との結合に関する制約を表すカテゴリ (以下 CCG カテゴリ) と、少数の組み合わせ規則によって、文構造を記述する (図 1).

Lewis ら [8, 9] はこの CCG の性質を活かした CCG 解析の手法を提案した. 彼らの手法では、文 $\mathbf{x} = x_1, \dots, x_N$ に対する CCG 木 \mathbf{y} の確率を各 x_i の CCG カテゴリ c_i の確率の積で表す (locally-factored モデル).

$$P(\mathbf{y}|\mathbf{x}) = \prod_{i \in [1, N]} P_{tag}(c_i|x_i). \quad (1)$$

このモデルのもとで最も確率の高い CCG 木を求めることは、A*探索を用いることで効率的に解くことができるが (2 節参照)、モデルの単純さにより、同じ CCG カテゴリ列から複数の CCG 木が導出可能な場合、それらについて、A*でどれを優先的に探索すべきかという曖昧性が生じる. そのときは決定的なルールに従ってこの曖昧性を解消する. 研究 [9] では、CCG 木から変換して得られる係り受け木を利用し、係り受けエッジの長さの合計がより大きいものを優先する (長距離依存を伴う木を選好することに当たる.). CCG カテゴリの結合可能性が、可能な木構造についての強力な制約となり、英語においてはこの単純なモデルとヒューリスティックにより、高速、並列に解析ができるうえに、高い精度を示すことができている.

Uematsu らによって日本語 CCGBank [13] が公開されているが、日本語においては、同じようなルールによる方法で曖昧性を解消することは有効ではなく、この手法を適用することは困難である. 例えば、連体修飾節や副詞節がもつ CCG カテゴリ S/S はこの曖昧性の原因の一つであり、その右に複数のカテゴリ S が存在した場合に、長距離依存を伴うものを優先するというルールを用いても、誤った解析結果になってしまうことが多い (図 1)¹. この図 1 のような場合、「昨

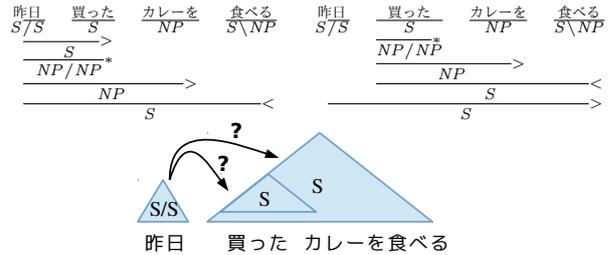


図 1: CCG における文「昨日買ったカレーを食べる」の解析例. S/S と結合する S についてルールで決められない.

日」がどちらの S と結合しやすいかという知識が必要である.

Lee ら [7] は、Lewis らの手法の拡張として、再帰型ニューラルネットを用いて A*で探索する経路 (= 構築途中の部分木) を評価する項を式 (1) に加えたモデルを提案した (global モデル). 彼らの手法は精度は高いが、問題点として、新たなノードを探索するたびに再帰型ニューラルネット で計算を行うため、もはや locally-factored なモデルではなく Lewis らの手法の単純さや高速性を失ってしまう.

本研究では、A* CCG 解析において、高速性のために再帰型ニューラルネットを用いない locally-factored なモデルを保ちながらも、上述の曖昧性の問題に対処するため、モデル (式 (1)) を、明示的に係り受け構造を考慮するモデルに拡張することを提案する. 具体的には、Lewis らの研究 [8] で提案された Bi-directional LSTM [11] (以下、Bi-LSTM) を用いて P_{tag} を予測するモデルを、近年係り受け解析で強力であることが示されている Dozat ら [4] の手法で、同時に文の係り受け構造を予測するように拡張する. CCG 木のモデルを、CCG カテゴリ列の確率と係り受け構造の確率の積とすることで、図 1 の「昨日」がどこに係るかということを明示的にモデルに組み込む.

提案手法を用いて、日本語 CCGBank [13] で実験を行った結果、文節係り受け精度の指標において、既存手法を上回った. また、英語 CCGBank [5] における実験では、Lewis ら [8] の精度を上回り、特にラベルなし F1 で大きな改善が見られ、木構造を捉える点で強力であることがわかった.

¹本論文では CCG カテゴリの素性値 (S_{dcl} の dcl 等) は、紙面の都合から省略する. しかし実際の解析において、組み合わせ規則が適用可能かどうかを考えると素性値を考慮して行う.

2 A* CCG 解析

Lewis ら [8, 9] による A*探索を用いた CCG 解析の手法を紹介する。彼らの手法では、文 x に対して可能な CCG 木の集合 $Y(x)$ から、モデル (式 (1)) のもとで最も確率の高い y を求める。以下、 $I = [1, N]$ とする。

$$y = \operatorname{argmax}_{y' \in Y(x)} \sum_{i \in I} \log P_{tag}(c_i | x_i). \quad (2)$$

この解を求める問題は、閉区間 $[i, j]$ と CCG カテゴリ C の組 $C_{i,j}$ をノードとするような有向ハイパーグラフ²上における最良経路問題として定式化でき (図 2)、以下で定義する始点 \emptyset から $C_{i,j}$ までの経路のスコア b と $C_{i,j}$ から終点 $C'_{1,N}$ までの経路のスコアの推定値 a を用いた A*探索を用いて解くことができる。

$$b(C_{i,j}) = \sum_{k \in [i,j]} \log P_{tag}(c_k | x_k), \quad (3)$$

$$a(C_{i,j}) = \sum_{k \in I \setminus [i,j]} \max_{c_k} \log P_{tag}(c_k | x_k). \quad (4)$$

有向ハイパーグラフにおける経路 $\emptyset \rightarrow \dots \rightarrow C_{i,j}$ は、その経路を辿ることで区間 $[i, j]$ の語からなる CCG の部分木 $y_{i,j}$ に対一に対応する (以下、 $C_{i,j}$ と $y_{i,j}$ は、交換可能に用いる)。 b はモデル (式 (1)) における $C_{i,j} = y_{i,j}$ の内側確率であり、 a はその外側確率の上界である。このような定義における経路探索問題が構文解析と対応することが示されている [6]。 a は定義から常に真の外側確率の上界を与えるため許容的 (admissible)、また単調なヒューリスティック関数であるためノード $C_{i,j}$ が初めて探索されたとき、その経路は必ず最適なものになる。

A*探索では $b+a$ が最も高い経路から探索を行うが、定義から同じ CCG カテゴリ列から導出される木は同じ値を持つ。このときどれを優先的に探索するかは決定的なルールに従う。研究 [9] では、CCG 部分木から決定的に変換された係り受け木を利用し (変換ルールは 3.2 節参照)、木に含まれるすべての係り受けエッジの長さの合計が大きいものを優先する。

3 提案手法

3.1 モデル

2 節の Lewis らの手法を係り受け構造で拡張したモデルを提案する。これによって図 1 のような曖昧性を柔軟に解消することができる。

²有向ハイパーグラフ $H = (V, E)$ とはノードの集合 V とエッジの集合 E の組であり、 $E = [(T, H) | T, H \subset V]$ であるものである。本研究の CCG 解析においては、常に $1 \leq |T| \leq 2, |H| = 1$ である。

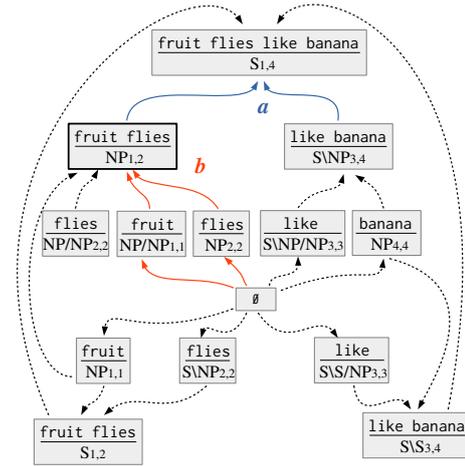


図 2: ハイパーグラフ上の A*探索による CCG 解析。 \emptyset は始点を表す。 b は内側確率で計算する経路のスコア、 a はヒューリスティック関数で外側確率の上界を用いる。ハイパーグラフ上の経路と CCG 木は一対一に対応する。

入力文 x に対する CCG 木 y の確率モデルを係り受け木に関する項で拡張し、A*で解く問題を以下のように定義する。ここで、 $root(y)$ を、部分木 y の係り受け構造におけるルートの語のインデックスとする。

$$y = \operatorname{argmax}_{y' \in Y(x)} \left(\sum_{i \in I} \log P_{tag}(c_i | x_i) + \sum_{i \in I \setminus \{root(y')\}} \log P_{dep}(h_i | x_i) \right). \quad (5)$$

2 節同様、 P_{tag} は単語の CCG カテゴリの確率であり、新たに追加された項 $P_{dep}(x_j | x_i)$ は語 x_i がその係り受け関係の親として語 x_j を取る確率である³。A*探索の \emptyset から $C_{i,j} = y_{i,j}$ の経路のスコア b も、モデル (式 (5)) のもとで内側確率となるように係り受け構造の項で拡張する。

$$b(y_{i,j}) = \sum_{k \in [i,j]} \log P_{tag}(c_k | x_k) + \sum_{k \in [i,j] \setminus \{root(y_{i,j})\}} \log P_{dep}(h_k | x_k). \quad (6)$$

具体的に、ハイパーグラフ上で $\{A_{i,k}, B_{k,j}\} \rightarrow \{C_{i,j}\}$ と探索をすすめるとき、ある組み合わせ規則が存在し、それによって A, B から C が作られる。すべての組み合わせ規則について、CCG カテゴリの組 (A, B) の関数として、 $root(y_{i,k}), root(y_{k,j})$ のどちらが係り受けの親となるかが定義されているから (3.2 節参照)、それにより例えば、 $h = root(y_{i,k}), d = root(y_{k,j})$ として係り受け関係 $h \rightarrow d$ が決まれば、 $b(y_{i,j}) = b(y_{i,k}) + b(y_{k,j}) + \log P_{dep}(h|d)$ と求める。

³CCG 木には、係り受け木におけるダミーのルートシンボルに相当するものは含まれないため、それに関する確率 $P_{dep}(ROOT|root(y))$ は除く (式 (5) における $I \setminus \{root(y)\}$)。

同様に、提案モデルにおける a を外側確率の上界となるよう以下のように定義する。

$$a(\mathbf{y}_{i,j}) = \sum_{k \in I \setminus [i,j]} \max_{c_k} \log P_{tag}(c_k | x_k) + \sum_{k \in L} \max_{h_k} \log P_{dep}(h_k | x_k),$$

ここで、 $L = I \setminus [k' | k' \in [i, j], \text{root}(\mathbf{y}_{i,j}) \neq k']$.

P_{tag} と P_{dep} を計算するため、Bi-LSTM [11] を用いる。全体のネットワークの構造は図 3 を参照。以下、MLP で多層パーセプトロン、 W のような大文字で行列、 U のような太字の大文字で 3 階のテンソルとする。入力文の各単語 x_i に対し、Bi-LSTM で隠れベクトル \mathbf{r}_i を計算し、Dozat ら [4] の手法をもとに、まず以下の式で P_{dep} を計算する。

$$\mathbf{h}_i^{dep} = MLP_{child}^{dep}(\mathbf{r}_i), \quad \mathbf{h}_j^{dep} = MLP_{head}^{dep}(\mathbf{r}_j), \quad (7)$$

$$P_{dep}(x_j | x_i) \propto \exp((\mathbf{h}_i^{dep})^T W_{dep} \mathbf{h}_j^{dep} + \mathbf{w}_{dep} \mathbf{h}_j^{dep}).$$

また、最も確率の高い親 $x_j = \text{argmax}_{x_j} P_{dep}(x_j | x_i)$ を用いて、

$$\mathbf{h}_i^{tag} = MLP_{child}^{tag}(\mathbf{r}_i), \quad \mathbf{h}_j^{tag} = MLP_{head}^{tag}(\mathbf{r}_j),$$

$$\ell = (\mathbf{h}_i^{tag})^T \mathbf{U}_{tag} \mathbf{h}_j^{tag} + W_{tag} \begin{bmatrix} \mathbf{h}_i^{tag} \\ \mathbf{h}_j^{tag} \end{bmatrix} + \mathbf{b}_{tag}, \quad (8)$$

$$P_{tag}(c_i | x_i) \propto \exp(\ell_c),$$

として、各単語について CCG カテゴリの確率を求める。式 (7) は Dozat らによって提案された biaffine 変換であり、双線形変換に、親のみに依存する項 $\mathbf{w}_{dep} \mathbf{h}_j^{dep}$ を加えることで、係り受けの親と子、のような非対称性のある組の関係を捉えることができる。

式 (8) では、単語 x_i の CCG カテゴリの予測に、係り受け関係における親 x_j を用いることで予測精度が改善されたためこのような手法を採用する。

本手法は、上で定義した P_{dep} 、 P_{tag} のみに依存し、またこれらは A*探索の前に行列の形で求めることができるため、Lee ら [7] の手法と同様に文の構造を考慮する一方で、locally-factored なモデルを保つので高速に処理できる。

3.2 係り受け構造の定義

変換ルールは、組み合わせ規則とそれが適用可能な CCG カテゴリの組 (A, B) に対して定義され、また、順方向の規則 (順関数適用、順関数合成など)、逆方向の規則 (逆関数適用、逆関数合成など)、それぞれに共通である⁴。

⁴その他の規則として日本語の SSEQ [13] では常に右、Lewis ら [9] の英語の並列句の規則 (Conjunction) では常に右、句読点除去の規則 (RemovePunctuation) では句読点でないほうを親と定

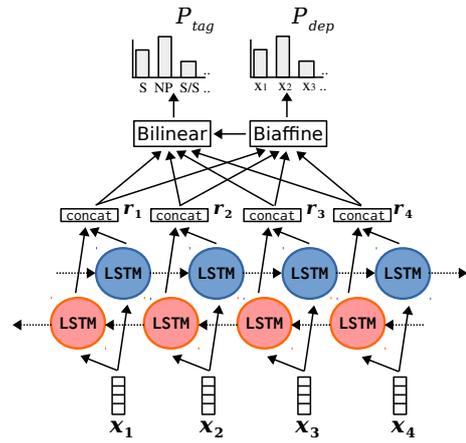


図 3: 入力の単語列 x_i の埋め込みベクトルを Bi-directional で隠れベクトル \mathbf{r}_i に変換し、係り受け、CCG カテゴリをそれぞれ式 (7)、(8) に基づいて同時に予測する。

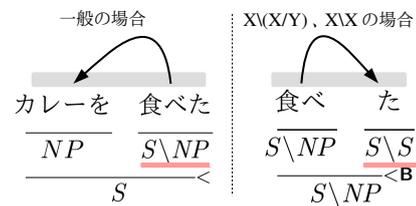


図 4: CCG 木から係り受け木への変換の例。逆方向の規則の場合、一般に右の部分木が親になるが、右のカテゴリが $X \setminus (X/Y)$ もしくは $X \setminus X$ にマッチするときは逆にする。

部分木の組、 $(\mathbf{y}_{i,k}, \mathbf{y}_{k,j}) = (A_{i,k}, B_{k,j})$ について、

順方向の組み合わせ規則

一般に $\text{root}(\mathbf{y}_{i,k})$ を $\text{root}(\mathbf{y}_{k,j})$ の親とする。例外として、カテゴリ A が修飾的 (X/X)、もしくは型繰り上げ ($X/(X \setminus Y)$) のときは逆にする。

逆方向の組み合わせ規則

一般に $\text{root}(\mathbf{y}_{k,j})$ を $\text{root}(\mathbf{y}_{i,k})$ の親とする。例外として、カテゴリ B が修飾的 ($X \setminus X$)、もしくは型繰り上げ ($X \setminus (X/Y)$) のときは逆にする。

図 4 に、変換ルールの適用例を示す。

4 実験

4.1 実験設定

日本語、英語の二言語で CCG 解析の実験を行った。日本語の実験では、実験データに日本語 CCG-Bank [13] を用いた。ベースラインとして Noji ら [10] による自然言語処理ツール *jigg* に実装されている shift-reduce 法による CCG パーザを用いる。提案法にお

ける。また、日本語の場合、CCG カテゴリから係り受け関係を決定するとき、カテゴリの素性値は除いて行う。「食べた」の助動詞「た」の CCG カテゴリは一般に $S_{f_1} \setminus S_{f_2}$ で $f_1 \neq f_2, S_{f_1} \neq S_{f_2}$ であるため、修飾的 $X \setminus X$ ではないが、素性値を取り除いて $S \setminus S$ で考えるため、修飾的とみなされる。

る語のベクトル表現として、日本語 Wikipedia エンティティベクトル⁵ で初期化した 200 次元のベクトルに、乱数で初期化した 50 次元の文字ベクトルから畳み込み演算 [3] により 100 次元に変換したベクトルを連結したものをを用いる。Bi-LSTM は順方向、逆方向それぞれ 4 層 300 次元、 MLP_{child}^{dep} 、 MLP_{head}^{dep} 、 MLP_{child}^{tag} 、 MLP_{head}^{tag} はそれぞれ、100 次元 1 層、非線形関数に elu [1] を用いる。最適化は、1 バッチ 32 文、Adam($\beta_1 = 0.9, \beta_2 = 0.9$)、L2 正則化 ($1e^{-6}$) を用い、入力の埋め込みベクトルを 50%、LSTM とすべての MLP の 32% を学習の間 dropout で無効にした。Dozat ら [4] 同様、学習率を $2e^{-3}$ から 2,500 イテレーションごとに 0.75 の比率で減衰させる。

英語の実験では、実験データに英語 CCGBank [5] を用い、既存研究に従い、C&C パーザ [2] に実装された generate プログラムで変換された依存構造をもとに評価する。ベースラインは Lewis ら [8] による Bi-LSTM を用いた手法で、Lewis らによって公開されている tri-training [14] で学習したモデルと、筆者らによって再実装された CCGBank のみで学習したモデルを用いる。語のベクトル表現として、GloVe⁶ で初期化した 100 次元単語ベクトルに、乱数で初期化した 30 次元の接頭辞、接尾辞 (長さ=1 ~ 4) のベクトル表現を連結したものをを用いた。Bi-LSTM、 MLP_{child}^{dep} 、 MLP_{head}^{dep} 、 MLP_{child}^{tag} 、 MLP_{head}^{tag} 、最適化は日本語と同じ設定で行った。

4.2 結果

日本語の実験の結果を表 1 に示す。日本語では、ベースラインの shift-reduce 法を用いた手法に対してカテゴリの予測精度において 0.8 ポイント、文節係り受けの精度で 3.3 ポイントの差で上回った。注目すべき点として、Noji らのスコアは正解の品詞タグを使っている一方で、提案手法は品詞情報は一切使っていない。

英語の実験の結果を表 2 に示す。提案手法は、筆者らによる再実装をラベルあり、ラベルなし F1 のどちらにおいても 0.8、1.0 ポイント差で上回った値を示した。Lewis らによるモデルは半教師ありの tri-training [14] を行っているため、学習データが多い (文にして 2,600,000 文) が、提案手法はラベルなし F1 において 0.4 ポイント差でこれを超え、木の構造を捉える点で強力であることを示した。

5 おわりに

本研究では、A* CCG 解析で、従来の CCG カテゴリ列のみを考慮したモデルを拡張し、係り受け構造を

⁵http://www.cl.ecei.tohoku.ac.jp/~m-suzuki/jawiki_vector/

⁶<http://nlp.stanford.edu/projects/glove/>

手法	カテゴリ	分節係り受け
Noji ら (2016)	93.0	87.5
提案	93.8	90.8

表 1: 日本語 CCG 解析の実験結果

手法	ラベルあり F1	ラベルなし F1
Lewis ら (2016, tritrain)	88.0	92.9
Lewis ら (2016, 再実装)	86.8	92.3
提案	87.6	93.3

表 2: 英語 CCG 解析の実験結果

含める手法を提案した。提案手法は、日本語、英語の両方においてベースラインを上回った精度を示した。今後の課題としては、提案手法が構造を捉える点において有利であることが、含意関係認識等の外部のタスクで有利であるかを評価し、tri-training を用いることで性能のさらなる向上を図ることを行う。

参考文献

- [1] Djork-Arné C., Thomas U., and Sepp H. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). *CoRR*, 2015.
- [2] S. Clark and J. R. Curran. Wide-coverage Efficient Statistical Parsing with CCG and Log-linear Models. *Computational Linguistics*, 2007.
- [3] C. N. dos Santos and B. Zadrozny. Learning Character-level Representations for Part-of-Speech Tagging. *ICML*, 2014.
- [4] T. Dozat and C. D. Manning. Deep Biaffine Attention for Neural Dependency Parsing. *CoRR*, 2016.
- [5] J. Hockenmaier and M. Steedman. CCGbank: a Corpus of CCG derivations and Dependency Structures Extracted from the Penn Treebank. *Computational Linguistics*, 2007.
- [6] D. Klein and C. D. Manning. A* Parsing: Fast Exact Viterbi Parse Selection. In *Proceedings of NAACL-2003*, 2003.
- [7] K. Lee, M. Lewis, and L. Zettlemoyer. Global Neural CCG Parsing with Optimality Guarantees. In *Proceedings of EMNLP-2016*, 2016.
- [8] M. Lewis, K. Lee, and L. Zettlemoyer. LSTM CCG Parsing. In *Proceedings of NAACL-2016*, 2016.
- [9] M. Lewis and M. Steedman. A* CCG Parsing with a Supertag-factored Model. In *Proceedings of EMNLP-2014*, 2014.
- [10] H. Noji and Y. Miyao. Jigg: A Framework for an Easy Natural Language Processing Pipeline. In *Proceedings of ACL-2016 System Demonstrations*, 2016.
- [11] M. Schuster and K. K. Paliwal. Bidirectional Recurrent Neural Networks. *IEEE Transactions on Signal Processing*, 1997.
- [12] M. Steedman. *The Syntactic Process*. The MIT Press, 2000.
- [13] S. Uematsu, T. Matsuzaki, H. Hanaoka, Y. Miyao, and H. Mima. Integrating Multiple Dependency Corpora for Inducing Wide-Coverage Japanese CCG Resources. *ACM Trans*, 2015.
- [14] D. Weiss, C. Alberti, M. Collins, and S. Petrov. Structured Training for Neural Network Transition-Based Parsing. In *Proceedings of ACL-2015*, 2015.
- [15] 戸次大介. 「日本語文法の形式理論 - 活用体系・統語構造・意味合成 -」. くろしお出版 日本語研究叢書 2 4, 2010.