漸進的構文解析における接合操作の導入

加藤 芳秀† 松原 茂樹††

† 名古屋大学大学院国際開発研究科
†† 名古屋大学情報連携基盤センター
E-Mail: yosihide@gsid.nagoya-u.ac.jp

1 はじめに

漸進的構文解析とは，自然言語文を単語の出現順序に従って解析し，文の入力途中の段階で，その構文構造を捉える枠組みである．同時通訳システムやリアルタイム字幕生成などの実時間音声言語処理システムの実現に必要不可欠な要素技術の一つである [1, 2, 6].

これまでにいくつかの漸進的構文解析手法が提案されている [5, 10, 11]．これらの手法では，単語が入力されるたびに，それまでに入力された文の断片に対してそれを覆う部分構文木を生成する．

これらの解析手法は，いずれも下降型構文解析と位置づけられるが，下降型の解析処理に起因する共通の問題を抱えている．左再帰構造による局所的曖昧性の問題である．下降型構文解析では，左再帰構造の深さを解析途中の段階で決定できず，あらゆる深さの左再帰構造を想定して，部分構文木（の候補）を生成しなければならない．

この問題を回避するために，本稿では漸進的構文解析に接合操作を導入する．接合操作は，木接合文法 [7] において用いられる構文木に関する操作の一つであるが，これにより，左再帰構造に起因する局所的曖昧性の問題を回避することができる．この方法は Lombardo らによりすでに提案されているが [8]，それは，すべての左再帰構造を接合操作により処理する単純な方法である．そのために，中間解析結果である部分構文木の一貫性が保たれないという別の問題が生じる．そこで提案手法では，単語間の依存関係に注目し，依存関係の一貫性が保たれるような操作のみを許容する．接合操作の導入により，従来の下降型解析に基づく漸進的構文解析と比べてより高精度な解析が実現できることを実験により確認した．

本稿の構成は以下のとおりである．2 節では，従来の漸進的構文解析手法として Collins らの手法について説明し，その問題点を議論する．3 節では，Collins らの手法に接合操作を導入する．4 節は，実験による提案手法の評価について報告する．5 節は，本稿のまとめである．

2 下降型漸進的構文解析

本節では，Collins らの提案した漸進的構文解析手法 [5] とその問題点について議論する．

2.1 Collins らの手法

Collins らの手法では，文法は，6 項組 $(V, T, S, \#, C, B)$ により定義される．V は非終端記号（範疇）の集合，T は終端記号（単語）の集合である．S は開始記号と呼ばれ，$S \in V$ である．$\#$ は構成素の終わりを表す特別な記号であり，$\# \in V$ である．C は allowable chain の集合である．allowable chain は，非終端記号と終端記号からなる系列で，最後の要素が終端記号であり，それ以外の要素は，非終端記号である．構文木中のある節点から左端の子を順にたどるときに得られるラベルの系列に相当する．B は allowable triple の集合である．allowable triple は 3 項組 $\langle X, Y, Z \rangle$ である．ここで，$X, Y, Z \in V$ である．allowable triple$\langle X, Y, Z \rangle$ は，親のラベルが X，左隣の兄弟のラベルが Y であるようなノードのラベルとして Z が可能であることを意味する．

Collins らの手法では，上述の文法を用いて文を漸進的に構文解析する．文を先頭から順に読みながら，読み込んだ部分を覆う部分構文木を生成することにより解析は進行する．最初の単語に対しては，開始記号で始まる allowable chain により部分構文木を生成する．以降は，allowable triple を用いて，部分構文木に付加可能な allowable chain を求め，allowable chain を付加することにより解析が進行する．

例として，以下の文の断片について考える．

We describe ... (1)

1 番目の単語 We に対して，例えば allowable chain ⟨ S → NP → PRP → We ⟩ が文法に存在すれば，図 1(a) に示す部分構文木を生成する．S で始まり，We で終わるその他の allowable chain が文法に存在すれば，それを用いて同様に中間解析結果の候補として部分構文木を生成する．続いて，2 番目の単語に対して，⟨ VP → VBP → describe ⟩ のような allowable chain を部分構文木 (a) に付加することにより解析

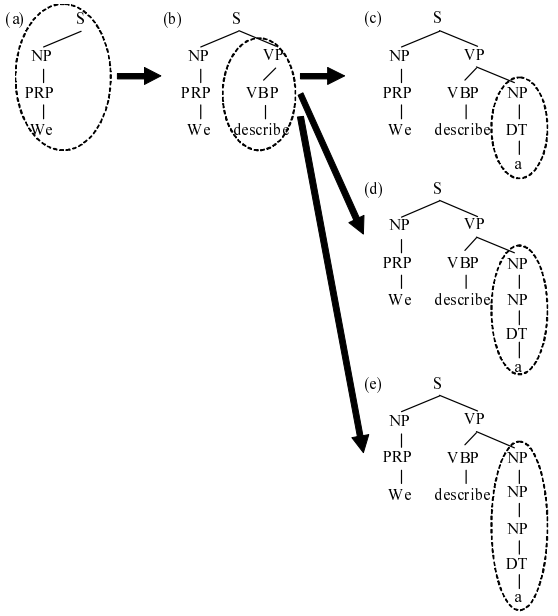

図 1: 漸進的構文解析の解析過程

が進行するが（図 1(b) 参照），付加可能な allowable chain は allowable triple により決められる．すなわち，allowable triple ⟨S, NP, VP⟩ が文法に存在する場合に限り，部分構文木 (a) の S の下に付加できる．このように，部分構文木に対して allowable chain を付加することにより，解析が進行し，文の断片に対して，それを覆う部分構文木を生成することができる．

2.2 下降型漸進的構文解析の問題点

下降型漸進的構文解析の問題として左再帰構造の局所的曖昧性の問題が挙げられる．左再帰構造は，付加詞や等位構造など構文木中に頻出するが，下降型漸進的構文解析において局所的曖昧性の原因となる [8]．左再帰構造の深さは解析途中の段階で決定できず，あらゆる深さの左再帰構造を想定して，部分構文木を生成しなければならないからである．

例として，以下の文の断片について考える．

We describe a ... (2)

この断片に対する部分構文木の候補として，例えば，図 1(c)〜(e) のような部分構文木が考えられる．(c) の部分構文木は，a で始まる名詞句が前置詞句などを取らない場合の部分構文木である．(d) は，名詞句が前置詞句を一つとるような場合の部分構文木，(e) は，名詞句が前置詞句を取り，さらにそれが等位構造の一部を構成するような場合の部分構文木である．これらは左再帰構造を含む部分構文木の一例であるが，この時点において左再帰構造の深さを決定するための情報は存在しないため，可能性としては任意の深さの左再帰構造が考えられる．すなわち，文の断片 (2) に対して，無限個の部分構文木の候補が存在することを意味する．

3 接合操作の導入

前節では，左再帰構造に起因する局所的曖昧性の問題について述べた．本節では，この問題に対する一つの解決法を提案する．左再帰構造の局所的曖昧性の問題に対して，従来の手法 [5, 10, 11] では，(1) 処理対象とする左再帰構造の深さを制限する，あるいは，(2) 構文木を変換して左再帰構造を除去するなどしている．(1) の方法に従えば，左再帰構造を含む部分構文木の候補を無限に生成し続ける無限ループは回避できるものの，依然として，様々な深さの左再帰構造を想定して部分構文木の候補を生成しなければならず，局所的曖昧性の問題は残る．(2) の方法では，左再帰構造が文法上存在しなくなるため問題自体が生じないが，構文木の構造がそもそも変わってしまい，部分構文木が構文的関係を正しく表現できるのかは明らかでない．

本稿では，別のアプローチを提案する．接合操作の導入である．接合操作により，左再帰構造が必要になった段階で，それを後から部分構文木に追加することにより，局所的曖昧性の問題を回避できる．この方法は，Lombardo らによりすでに提案されているが [8]，すべての左再帰構造を接合操作で処理する単純な方法である．そのために，別の問題が生じる．解析過程で生成される部分構文木の一貫性が保てないという問題である．本節では，この問題について単語間の依存関係に注目して議論する．この問題を解決するために，提案手法では，接合操作により処理する左再帰構造の種類を制限する．

3.1 接合操作

まず接合操作について説明する．接合操作は，木接合文法において用いられる操作である．構文木中に別の木を挿入する操作であり，挿入する木を補助木と呼ぶ．補助木には，足と呼ばれる特別な節点があり，そのラベルは，補助木の根のラベルと同一である．接合操作は以下のように定義できる．

接合 構文木 σ のある節点 η で σ を上下に分割し，間に補助木 β を挿入する（上側の木と β の根，下側の木と β の足を結合する）．ただし，η のラベルと，補助木の根のラベルは同一であるものとする．

一般には，上で述べた条件を満たす任意の木を補助木とすることができるが，提案手法では，補助木としてもっとも単純な木を用いる．根と足の二つの節点からなる補助木である．このような補助木を適宜，接合することにより，漸進的構文解析における左再帰構造の問題を回避することができる．

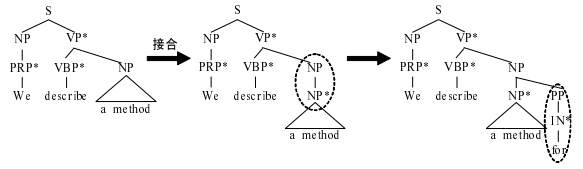

図 2: 接合操作の例

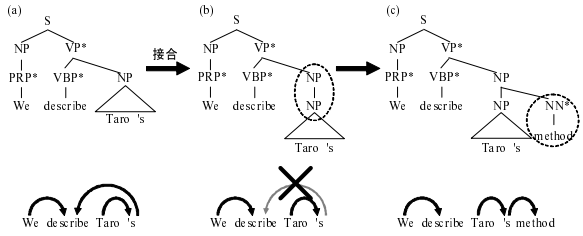

図 3: 非単調な接合操作の例

例として，以下の文について考える．

We describe a method for incremental parsing. (3)

2 節で示したように，従来の下降型漸進的構文解析では，この文を単語 a まで読むと，図 1 の (c)〜(e) のような部分構文木を生成する．これに対して，提案手法では，(c) のような部分構文木だけ生成すればよい．左再帰構造が必要になった段階で，接合操作により挿入できるからである．解析が進行し，単語 for を呼んだ時点で，図 2 に示すように，まず接合操作により補助木を挿入し，その後に，for に対する allowable chain ⟨ PP → IN → for ⟩ を付加する．この例が示すように，従来の手法と異なり，あらゆる深さの左再帰構造を想定して部分構文木を生成する必要がない，すなわち，左再帰構造による局所的曖昧性の問題を回避できる．

3.2 接合操作と単調性

前節で述べたように，接合操作により左再帰構造の局所的曖昧性の問題を回避できるが，すべての左再帰構造を接合操作により処理すると別の問題が生じる．部分構文木の一貫性が保てなくなるという問題である．この問題について，以下では例を用いて説明する．

例として，以下の文を考える．

We describe Taro 's method. (4)

この文において 's まで読んだ段階では，図 3(a) の部分構文木が生成される．この部分構文木は，Taro 's が describe の目的語であることを表している．次の語 method を読んだ時点では，まず接合操作を適用し，図 3(b) の部分構文木を生成し，これに対して allowable chain ⟨ NN → method ⟩ を付加して，図 3(c) を生成する．この部分構文木においては，Taro 's が method を修飾している．このように，接合操作により左再帰構造を処理すると，解析の進行過程において，部分構文木の表現する構文的関係が一貫しない場合が存在する．

部分構文木の一貫性の問題を形式的に定義するために，本稿では，単語間の依存関係に注目する．構文木において head child となる節点を決めると，構文木から単語間の依存関係の集合へと変換できる [4]. この依存関係に注目すると，部分構文木の一貫性の問題は，依存関係の単調性の問題とみなすことができる．すなわち，部分構文木に付加操作，あるいは接合操作を適用するとき，依存関係は単調に増えるか否かという問題である．

図中の * は head child であることを表すが，図 3 の例では，図 3(a) の部分構文木を依存関係に変換すると，We → describe, Taro → 's, および 's → describe の 3 つの依存関係が得られる．ところが，図 3(a) に接合操作を適用すると，'s → describe は失われる，すなわち，依存関係は単調に増大していない．一方，図 2 の例では，接合操作を適用した後も，依存関係は失われず，依存関係は単調に増加している．

提案手法では，部分構文木の一貫性，すなわち依存関係の単調性の制約を満足する接合操作のみを許容する．具体的には，以下の条件を満たす場合にのみ接合操作を適用する．

接合操作適用の前提条件 補助木の足が head child となる場合に限り，接合操作を適用する．

図 2 における接合操作はこの条件を満たし，図 3 におけるそれは，この条件を満たさない．

3.3 文法の獲得

カバー率の高い文法を実現するために，提案手法では，構文木を漸進的構文解析の逆の過程で分解することにより文法を獲得する．構文木コーパスを利用することにより大規模な文法を獲得できる．

head child が決められた構文木を以下の手順で分解し，文法を獲得する．

1. η を構文木中の最右に出現し，子を一つだけ持つ節点とし，その子を η_c とする．η と η_c のラベルが同一であり，η_c が head child であるならば，η と η_c からなる部分を補助木として取り出す．補助木が取り出せなくなるまでこの操作を繰り返す．
2. 構文木を次の条件を満たす節点 η で分割し，allowable chain を取り出す．
   - η は最右の単語を子孫に持つ．
   - η の親は二つ以上の子を持つ．
   - η およびその子孫は，子を一つだけ持つ．
3. 残された部分構文木を 1.〜3. の手順で分解する．

表 1: ラベル再現率とラベル精度

	再現率 (%)	精度 (%)	F 値
接合操作なし	86.3	86.8	86.6
接合操作あり-非単調	86.1	87.1	86.6
接合操作あり-単調	**87.2**	**87.7**	**87.4**

4 実験による評価

漸進的構文解析における接合操作の有効性を確認するために，構文解析実験を行った．本実験では，漸進的構文解析において接合操作を用いた場合と用いなかった場合を比較する．また，依存関係の単調性を考慮した場合と考慮しなかった場合も比較する．ラベル再現率とラベル精度を用いて，それぞれの場合の構文解析の性能を評価する．文法は，Penn Treebank[9] の WSJ コーパスのセクション 2–21 から抽出した．head child は，文献 [4] の方法に従って決定した．構文木のランク付け，及び解析の枝刈りのために，最大エントロピー法 [3] を用いて確率モデルを構築した．素性として，付加や接合を行う節点の祖先や兄弟のラベル，head child のラベル，head word とその品詞，head word との距離などを使用した．

セクション 23 の 2416 文に対するラベル再現率，ラベル精度を表 1 に示す．この結果は，接合操作を用いない場合とすべての左再帰構造を接合操作で処理する場合における，再現率・精度はそれほど変わらないことを示している．これに対して，依存関係の単調性の制約を満たす場合にのみ接合操作を許す提案手法は，再現率・精度ともに高い値を示している．このことは，接合操作において，依存関係の単調性を考慮することの重要性を示しており，提案手法は有効であると考えられる．

5 おわりに

本稿では，接合操作を用いた漸進的構文解析手法を提案した．提案手法では，構文木の一貫性を保つために，単語間の依存関係に着目し，依存関係の単調性を満たす接合操作のみを許容する．この制約を用いることにより解析精度が向上することを実験により確認した．

本稿で提案した手法では，直接的な左再帰構造しか扱っていないが，間接的な左再帰構造について検討することが今後の課題である．

参考文献

[1] G. Aist, J. Allen, E. Campana, C. G. Gallo, S. Stoness, M. Swift, and M. K. Tanenhaus. Incremental understanding in human-computer dialogue and experimental evidence for advantages over nonincremental methods. In R. Artstein and L. View eds., *Proceedings of the 11th Workshop on the Semantics and Pragmatics of Dialogue*, pp. 149–154, 2007.

[2] J. Allen, G. Ferguson, and A. Stent. An architecture for more realistic conversational systems. In *Proceedings of International Conference of Intelligent User Interfaces*, pp. 1–8, 2001.

[3] A. L. Berger, S. A. D. Pietra, and V. J. D. Pietra. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71, 1996.

[4] M. Collins. *Head-Driven Statistical Models for Natural Language Parsing*. PhD thesis, University of Pennsylvania, 1999.

[5] M. Collins and B. Roark. Incremental parsing with the perceptron algorithm. In *Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL'04), Main Volume*, pp. 111–118, 2004.

[6] Y. Inagaki and S. Matsubara. Models for incremental interpretation of natural language. In *Proceedings of the 2nd Symposium on Natural Language Processing*, pp. 51–60, 1995.

[7] A. K. Joshi. Tree adjoining grammars: How much context sensitivity is required to provide a reasonable structural description? In D. R. Dowty, L. Karttunen, and A. M. Zwicky eds., *Natural Language Parsing*, pp. 206–250. Cambridge University Press, 1985.

[8] V. Lombardo and P. Sturt. Incremental processing and infinite local ambiguity. In *Proceedings of the 19th Annual Conference of the Cognitive Science Society*, pp. 448–453, 1997.

[9] M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):310–330, 1993.

[10] B. Roark. Probabilistic top-down parsing and language modeling. *Computational Linguistics*, 27(2):249–276, 2001.

[11] B. Roark. Robust garden path parsing. *Natural language engineering*, 10(1):1–24, 2004.