Supertag を利用した依存構造解析

大内 啓樹　　Kevin Duh　　松本 裕治
奈良先端科学技術大学院大学 情報科学研究科
{ouchi.hiroki.nt6, kevinduh, matsu}@is.naist.jp

1　はじめに

遷移型依存構造解析 ([9], [7], [4], [5]) の利点として，多様な素性表現が使用可能であることが挙げられる．しかし，一般的に現在の高精度な解析器でも語形と品詞の組み合わせを基本とした素性表現のみの使用に留まっていることが多い．本研究では，詳細な統語現象を表現可能な supertag を導入し，それを素性として解析モデルに組み込み，遷移型依存構造解析の精度向上をねらう．

本稿において，英語依存構造解析における supertag は依存構造が付与されたコーパスから抽出された語彙テンプレートであり，文脈における複雑な言語的制約を表現する ([2]) ものと定義する．supertag はこれまでも語彙化文法 (語彙化木接合文法，主辞駆動句構造文法，組合せ範疇文法など) を基盤とする構文解析において使用されてきたが、依存構造解析にはほとんど使用されていない。

Foth 他 [3] は supertag が重み付き制限依存文法を用いたドイツ語依存構造解析の精度向上に寄与することを実証しており，Ambati 他 [1] は 組合せ範疇文法の supertag をヒンズー語依存構造解析に利用することによって解析精度向上が可能となり，特に長距離依存構造 (並列構造や関係詞節) の解析精度が向上したと報告している．Zhang 他 [11] は主辞駆動句構造解析のために長距離依存構造を supertagging の素性として組み込んでいる．これらの先行研究は依存構造解析において supertag が良い影響を与える可能性を示唆している．本稿では、(1) 英語依存構造解析において効果的に働く supertag の設計と調査，(2)supertag 素性を組み込んだ遷移型依存構造解析器の提案をする．

本稿の構成は以下の通りである．2 章で supertag の設計手法を述べ，3 章で依存構造解析に使用する supertag 素性の説明をする．さらに，4 章で Penn Treebank([6]) を用いた supertag 自動付与と依存構造解析の評価実験を示す．依存構造解析実験では自動付与された supertag を使用し，ラベルなし精度で 1.22% の向上，ルート正解率で 1.86% の向上，文正解率で 3.54% の向上を実証した．

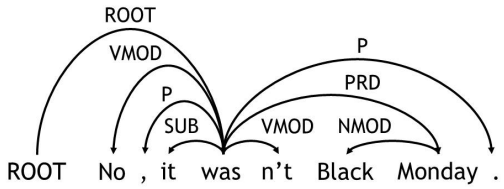

図 1　依存構造解析例

表 1　Model 1，2 の supertag 例

Word	Model 1	Model 2
No	VMOD/R	VMOD/R
,	P/R	P/R
it	SUB/R	SUB/R
was	ROOT+L_R	ROOT+SUB/L_PRD/R
n't	VMOD/L	VMOD/L
Black	NMOD/R	NMOD/R
Monday	PRD/L+L	PRD/L+L
.	VMOD/L	VMOD/L

2　Supertag の設計

Supertag の設計において，いかにタグの粒度と表現力間の適切なバランスを保つかが問題となる．より詳細な統語情報を表現するために supertag の粒度を細かく設計することが理想だが，タグセットのタグ数増大に従ってそれらの自動付与がより困難になるという傾向があり，トレードオフの関係にある．本稿では，依存構造解析に重要と考えられる統語情報に焦点を絞り，粒度の異なる 2 種類の supertag モデルを設計した．

簡単のため，表 1 の文を例にそれぞれの supertag モデルについて説明する．1 つ目のモデルである Model 1 は右方向 (R) や左方向 (L) といった単語の主辞の相対的位置を統語情報として表現する．単語の主辞が

ROOT の場合は位置関係がないものとして考える．さらに，単語と主辞間の依存関係ラベルを追加する．例えば，表 1 の'No' は右方向に依存関係ラベル'VMOD' の主辞を持つ．この supertag を'VMOD/R' と表す．これらの情報は文内においてその単語が担う統語的役割を推定する手がかりとなる．このような情報に加え，単語が右または左に従属辞を持つという情報も追加する．例えば，表 1 の単語'Monday' は左従属辞として'Black' を持つので，'Monday' の supertag は'PRD/L+L' と表す．この supertag において，'+' の前の部分 ('PRD/L') は主辞の情報を表し，後の部分 ('L') は従属辞の相対的位置を表している (L は左，R は右を表す)．表 1 の'was' のように，単語が左右どちらの方向にも従属辞を持つ場合は，'ROOT+L_R' のように'_' で 2 つの従属辞の情報をつなぐ．従属辞は左従属辞から並べ，右従属辞はその後に来るように表記する．Penn Treebank のデータにおいて Model 1 の supertag の総数は 79 となった．

Model 2 では，単語の品詞が動詞でありその従属辞が動詞に必須の場合，その単語と従属辞間の依存関係ラベルを supertag に追加した．本稿では動詞と従属辞間の依存関係ラベルが，'SUB'，'OBJ'，'PRD'，'VC' のものを動詞に必須の従属辞として定義した．依存関係ラベルが必須従属辞でない場合，従属辞の相対的位置のみが supertag として表現される．例えば，表 1 の'was' は必須従属辞として左方向に'SUB'，右方向に'PRD' を持つので，その supertag は'ROOT+SUB/L_PRD/R' と表される．動詞が同じ方向に複数の必須従属辞を持つ場合は，それらを連続して並べて表す．例えば，1 つの主語 (SUB) と 2 つの目的語 (OBJ) をとる動詞があった場合，'X/X+SUB/L_OBJ/R_OBJ/R' と表す．Model 2 の supertag の総数は 312 となった．Model 2 は Foth 他 [3] の Model F に類似しているが，前置詞や従属接続詞の必須従属辞は考慮していない点で異なる．これは，係り受け木を構築する上で動詞が最重要な役割を担うと考え，かつ supertag の総数の増加を抑えるため，本稿では動詞の必須従属辞のみに限定した．

3 遷移型依存構造解析における supertag 素性

本稿では，高精度な遷移型依存構造解析器である Goldberg と Elhadad[4] の Easy-First モデルを採用する．Easy-First モデルに supertag を使用した素性を組み込む方法に関する説明を行う．ただし，Left-to-Right の *arc-eager* や *arc-standard* モデル ([8], [9]) のような他の遷移型依存構造解析の枠組みにも同様に適応可能である．

表 2 Supertag 素性テンプレート．w = 表層形; t = 品詞; s = supertag; sh = supertag 主辞情報; sld = supertag 左従属辞情報; srd = supertag 右従属辞情報

unigrams of supertags
$p \in \{p_{i-2},\ p_{i-1},\ p_i,\ p_{i+1},\ p_{i+2},\ p_{i+3}\}$ に対して $w_p s_p,\ t_p s_p$
bigrams of supertags
$(p,\ q) \in \{(p_i,\ p_{i+1}),\ (p_i,\ p_{i+2}),\ (p_{i-1},\ p_i),\ (p_{i-1},\ p_{i+2}),\ (p_{i+1},\ p_{i+2})\}$ に対して $s_p s_q,\ t_p s_q,\ s_p t_q,\ w_p s_q,\ s_p w_q$
head-dependent of supertags
$(p,\ q) \in \{(p_i,\ p_{i+1}),\ (p_i,\ p_{i+2}),\ (p_{i-1},\ p_i),\ (p_{i-1},\ p_{i+2}),\ (p_{i+1},\ p_{i+2})\}$ に対して $w_p sh_p w_q sld_q,\ t_p sh_p t_q sld_q,\ w_p srd_p w_q sh_q,\ t_p srd_p t_q sh_q$

Easy-First アルゴリズムでは，n 単語から成る文 w_1,...,w_n を初期状態とし，解析過程で得られる部分木構造 p_1,...,p_k のリストに対して AttachLeft(i) と AttachRight(i) の 2 種類のアクションを適用することによって係り受け木を構築する．AttachLeft(i) は $(p_i,\ p_{i+1})$ をまとめ，p_{i+1} を部分木構造リストから取り除く．AttachRight(i) は $(p_{i+1},\ p_i)$ をまとめ，p_i を部分木構造リストから取り除く．まとめ上げを行う位置の部分木構造と隣接する 2 つの部分木構造 p_{i-2}, p_{i-1}, p_i, p_{i+1}, p_{i+2}, p_{i+3} から素性を抽出する．表 2 にこれらの隣接する部分木構造から抽出する supertag 素性をまとめた．supertag 素性は [4] で使用されているベースラインとなる素性に追加する形で組み込む．

部分木構造 p において，その主辞の単語が持つ情報に基づき素性を定義する．部分木構造の主辞の単語の表層形を w_p，品詞を t_p，supertag を s_p と表す．さらに，supertag をそのまま使用するだけでなく，それぞれの supertag を部分に分割しても使用する．例えば supertag'ROOT+SUB/L_PRD/R' を分割し，'ROOT'，'DEP:SUB/L'，'DEP:PRD/R' に分割する．主辞と従属辞の表記が混同しないよう，分割後の従属辞には'DEP:' を付与する．これらはそれぞれ，supertag 主辞情報 sh_p，supertag 左従属辞情報 sld_p，supertag 右従属辞情報 srd_p として表す．

ユニグラム素性として，1 つの部分木構造内の情報を使用する．例えば，表層形と supertag の結合 $(w_p s_p)$，品詞と supertag の結合 $(t_p s_p)$ がある．また，より文脈を考慮するため，部分木構造のペアのバイグラムを素性とする．部分木構造 p_{i-2}, p_{i-1}, p_i, p_{i+1}, p_{i+2}, p_{i+3} におけるそれぞれのペア (p,q) に対して，supertag の

表 3　Supertagging の正解率

Model	# tags	Dev	Test
Model1	79	87.81	88.12
Model2	312	87.22	87.13

バイグラム ($s_p s_q$) などを素性として使用する．

最後に，主辞と従属辞の一貫性をチェックするため，1 つの部分木構造が持つ supertag の主辞情報部分と隣接する部分木構造が持つ supertag の従属辞情報部分を結合し，”head-dependent 素性”として使用する．例えば，表 1 で単語’Black’ の supertag は主辞情報部分として’NMOD/R’ を持ち，これは自身の右方向の単語を主辞として持とうとする．単語’Monday’ の supertag は従属辞情報として’L’ を持ち，左方向にある単語を従属辞として持とうとする．つまり，主辞従属辞関係の一貫性が守られているため，これら 2 つの単語がまとめ上げられる確率が高くなる．head-dependent の情報は単語の表層形や品詞と結合して使用する．

4　実験

Supertag 素性の効果を評価するため，Penn2Malt[*1]で依存構造形式に変換した Penn Treebank(PTB) で実験を行った．標準的な実験手法を採用し，PTB を分割しセクション 2-21 を訓練データに，セクション 22 を開発データに，23 を評価データとした．訓練データの品詞タグは [5] に従い，10 分割 jackknifing で自動付与した．開発，評価データには訓練データで訓練した品詞タガーで自動付与した．

4.1　Supertagging 実験

上記の訓練データを用い，条件付確率場 (CRFs) でそれぞれの supertag モデルの supertagger を訓練し，開発，評価データで正解率を算出した．CRFs の実装として CRFsuite[*2]を使用した．素性としてターゲットとなる単語自身とその前後 3 単語の表層形，品詞タグのユニグラム，バイグラム，トライグラムを使用した．表 3 に supertagging の正解率を示す．supertagging の正解率はどちらのモデルでも約 87-88% となり，ほとんどの supertag は標準的な CRFs で効果的な学習が可能だということが示唆された．

エラー分析において，Model 2 の必須従属辞を含む supertag を正しく付与することが困難であることがわかった．評価データにおいて，必須従属辞を含む supertag の総数は 5432 であり，その正解率は 74.61% となった (Model 1 において対応する supertag の正解率は 82.18% であった)．特に，主辞部分が従属接続詞を表す依存関係ラベル’SBAR’ であり従属辞部分が必須従属辞を表す supertag(e.g.，SBAR/L+SUB/L_PRD/R) を予測することは困難であった．そのような supertag の正解率は約 60%(e.g., supertag’SBAR/L+SUB/L_PRD/R’ の正解率は 57.78%) であった．一方で，依存関係ラベル’VC’ を含む従属辞を表した supertag はほぼ正確に付与されていた (e.g.，supertag’VC/L+VC/R’ の正解率は 97.41%)．従属節内の動詞は一般的に従属接続詞を主辞として持ち，その依存関係は予測が難しいとされる単語間の距離が長い依存となる傾向にある．’VC’ は動詞補語を表し，動詞と依存関係にある現在分詞や過去分詞などが該当する．それらは直前の動詞の従属辞として頻繁に現れ，単語間の距離が近いので予測が比較的容易である．

表 4　各素性テンプレートのラベルなし精度

feature	Model1	Model2
baseline	90.25	90.25
+unigram of supertag	90.59	90.76
+bigram of supertag	91.37	91.08
+head-dependent	91.22	91.28

表 5　依存構造解析ラベルなし精度

Model	UAS	Root	Complete
baseline	90.05	91.10	37.41
Model 1	91.27	92.96	40.89
Model 2	91.23	92.72	41.35

4.2　依存構造解析実験

はじめに，3 節で提案した supertag 素性テンプレートの効果を評価する．まず，品詞タグ付与工程と同じく，10 分割 jackknifing で訓練データに supertag を自動付与する．次に，自動付与した supertag の素性を組み込んだ Easy-First 依存構造解析器を訓練する．開発データ，評価データに対しては，訓練データで訓練した supertagger で supertag を自動付与する．

表 4 は開発データにおける supertag 素性の効果を示している．ベースライン素性からはじめ，supertag のユニグラム，バイグラム，head-dependent 素性テンプレートを追加していった．Model 1 では，supertag のユニグラム素性を追加するとラベルなし精度 (UAS) がベー

[*1] http://stp.lingfil.uu.se/ nivre/research/Penn2Malt.html
[*2] http://www.chokkan.org/software/crfsuite/

スラインから 0.34% のわずかな向上が見られた．さらに，supertag のバイグラム素性を加えると，0.78% という比較的大きな向上が見られた．一方，Model 2 における supertag のユニグラム素性は 0.51%，バイグラム素性は 0.32% の精度向上に寄与し，ユニグラムの方が大きく精度を向上させた．考えられる理由として，Model 2 の supertag は Model 1 のそれより詳細な統語情報を表しているため，個々の独立した supertag がユニグラム素性として精度向上に寄与できる可能性が挙げられる．しかし，Model 2 の supertag はエラーも多く含み，バイグラム素性のような複数の supertag の組合せはエラーを伝搬させる可能性がある．

すべての素性を使用すると，Model 2 の精度はさらに 0.20% 向上し，逆に Model 1 では 0.15% の低下が見られた．Model 1 の精度低下の理由は明確ではないが，1 つの仮説として粗い supertag が head-dependent において悪影響を及ぼすことがあるということが挙げられる．開発データにおけるすべての素性を組み込んだ解析器での最終的な精度は 91.22%(Model 1) と 91.28%(Model 2) という結果となり，それぞれベースラインから 0.97% と 1.03% の向上が見られた．

次に，評価データの解析結果を示す．開発データでの実験結果を受け，Model 1 では supertag のユニグラムとバイグラム素性を，Model 2 では head-dependent 素性を含めたすべての素性を組み込んだ解析器を用いた．表 5 はラベルなし精度，ROOT 正解率，文正解率を示している．Model 1 と Model 2 の両方ともすべての評価指標においてベースラインからの改善が見られる．特に，Model 1 でラベルなし精度で 1.22%，ROOT 正解率で 1.86%，文正解率で 3.54% の向上を達成した．表には記述していないが，Model 1 に関しても全ての supertag 素性を組み込んだ追加実験を行い，ラベルなし精度で 91.35%，ROOT 正解率で 93.17%，文正解率で 41.35% という結果が得られた．この結果から Model 1 の head-dependent 素性の効果は不安定である可能性が示唆された．

5 おわりに

本稿では，英語依存構造解析において素性としての supertag の効果を実証した．先行研究では，単語の主辞や従属辞などの統語情報は，解析が進んで部分木が構築された後にしか素性として使用できていない ([10], [4])．supertag を付与し素性として使用することにより，部分木の構築を待たずして粒度の細かい統語情報の使用が可能となり，英語依存構造解析における精度向上への寄与が確認された．これから，解析アルゴリズムにおいて supertag を直接的に利用する解析モデルの開発と，supertag 設計において有用なパターン発見手法の調査を進めていきたい．

参考文献

[1] Bharat R Ambati, Tejaswini Deoskar, and Mark Steedman. Using CCG categories to improve Hindi dependency parsing. In *In Proceedings of ACL*, pp. 604–609, 2006.

[2] Srinivas Bangalore and Aravind K Joshi. Supertagging: An approach to almost parsing. *Computational Linguistics*, Vol. 25(2), pp. 237–265, 1999.

[3] Kilian Foth, Tomas By, and Wolfgang Menzel. Guiding a Constraint Dependency Parser with Supertags. In *In Proceedings of COLING/ACL 2006*, pp. 289–296, 2006.

[4] Yoav Goldberg and Michael Elhadad. An Efficient Algorithm for Easy-First Non-Directional Dependency Parsing. In *In Proceedings of HLT/NAACL*, pp. 742–750, 2010.

[5] Liang Huang and Kenji Sagae. Dynamic programming for linear-time incremental parsing. In *In Proceeding of ACL*, pp. 1077–1086, 2010.

[6] M P Marcus, B Santorini, and M Marcinkiewicz. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, Vol. 19(2), pp. 313–330, 1993.

[7] Joakim Nivre, Johan Hall, Jens Nilsson, Atanas Chanev, Gülsen Eryiğit, Sandra Kübler, Svetoslav Marinov, and Erwin Marsi. Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, Vol. 13(2), pp. 95–135, 2007.

[8] Joakim Nivre, Johan Hall, Jens Nilsson, Gülsen Eryiğit, and Svetoslav Marinov. Labeled pseudo-projective dependency parsing with support vector machines. In *In Proceedings of CoNLL*, pp. 221–225, 2006.

[9] H Yamada and Y Matsumoto. Statistical dependency analysis using support vector machines. In *In Proceedings of IWPT*, pp. 195–206, 2003.

[10] Yue Zhang and Joakim Nivre. Transition-based Dependency Parsing with Rich Non-local Feaures. In *In Proceedings of ACL*, pp. 188–193, 2011.

[11] Yao zhong Zhang, Takuya Matsuzaki, and Jun'ichi Tsujii. A Simple Approach for HPSG Supertagging Using Dependency Information. In *In Proceedings of HLT/NAACL*, pp. 645–648, 2010.