

# 複単語表現を考慮した英語の依存構造解析モデリング

加藤 明彦<sup>1,a)</sup> 進藤 裕之<sup>1,b)</sup> 松本 裕治<sup>1,c)</sup>

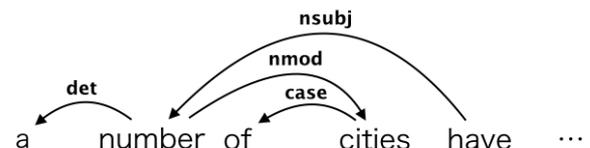
**概要:** 複単語表現 (MWE) を考慮した英語の依存構造解析とは、文に含まれる MWE を一つのトークンとする依存構造解析であり、MWE 認識と依存構造解析の双方について高い精度を達成するモデルが望まれる。しかしながら、どのようなモデルが英語 MWE を考慮した依存構造解析に適しているかは自明ではない。そこで本研究では (1) MWE の範囲同定と依存構造解析のパイプラインモデル、(2) MWE の範囲と依存構造を同時解析する Joint モデル、(3) 予測した MWE の範囲を Joint モデルの追加素性として用いる Joint+supertagging モデル、の 3 種のモデルを構築した。実験の結果、MWE 認識において、Joint+supertagging モデルは Joint モデルよりも 4 ポイント以上、パイプラインモデルよりも約 1 ポイント高い F 値を示した。また、我々は Joint+supertagging モデルとパイプラインモデルのハイブリッドモデルを構築し、依存構造解析において Joint+supertagging モデルよりも 0.14 ポイント高い UAS 値を得た。

## 1. はじめに

複単語表現 (MWE) とは、統語構造あるいは意味構造上の単位として取り扱う必要のある複数の単語のまとまりである。MWE は「単語境界を越える特異的な表現」としても知られる [1]。

MWE はしばしば意味的な非構成性を持つため、MWE の認識は多くの応用タスクで重要な役割を果たしている。例としては機械翻訳 [2]、情報検索 [3]、意見マイニング [4] などが挙げられる。

しかし、英語 MWE のアノテーションを付与したコーパスについてはその多くが小規模であり、注釈対象の MWE が複合名詞や動詞など特定のタイプに限られていた。この状況を受けて近年、包括的な MWE アノテーションの試みがいくつか行われている。まず Schneider ら [5] は、English Web Treebank [6] 上に非連続 MWE を含む様々な種類の MWE を注釈付けした。また Shigeto ら [7] は、英語の複合機能語の辞書を用いて Ontonotes [8] の Wall Street Journal セクションに対して MWE アノテーションを付与した。Shigeto らの研究では、英語の Wiktionary<sup>\*1</sup> からエントリを抽出して網羅的に辞書を作成している。複合機能語の例としては“a number of”, “even though”, “after all” 等が挙げられる。



(a) 単語ベースの依存構造



(b) MWE を考慮した依存構造

図 1: 単語ベースの依存構造と MWE を考慮した依存構造。後者では MWE (図中の “a number of”) は単一トークンとして表現されている。

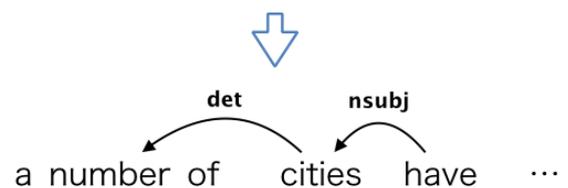


図 2: パイプラインモデルでは系列ラベリングの予測に基づいて各 MWE を単一トークンにまとめ上げた後に依存構造解析を行う。

<sup>1</sup> 奈良先端科学技術大学院大学  
Nara Institute of Science and Technology  
a) kato.akihiro.ju6@is.naist.jp  
b) shindo@is.naist.jp  
c) matsu@is.naist.jp  
<sup>\*1</sup> <https://en.wiktionary.org/>

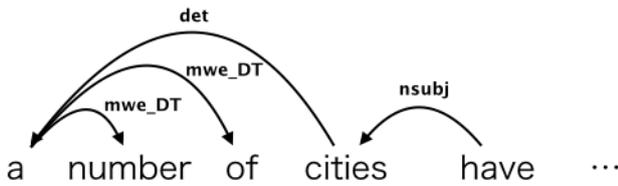


図 3: Joint モデル及び Joint+supertagging モデルでは MWE の範囲を係り受けラベルとして表現し、従来の単語ベースの依存構造解析を行う。

しかし、これらのコーパスでは MWE の範囲が句構造木と整合的である事は保証されていない。つまり、MWE の範囲は単一の非終端ノードが支配する範囲に対応するとは限らない。従って、MWE の範囲と統語構造の双方を組み合わせた深い解析を必要とする自然言語処理タスクに取り組む場合に、個々のタスクで得られた情報を上手く組み合わせる事ができないという問題点があった [9]。

近年 Kato ら [10] は、Ontonotes 上の句構造木を一部修正する事によって MWE の範囲と句構造木との整合性を保証し、MWE-aware dependency corpus (2章) を構築した。図 1 に単語ベースの依存構造と MWE を考慮した依存構造の違いを示す。前者では MWE の範囲は表現されていない (図 1a) のに対し、後者では MWE が単一トークンとして表現されている (図 1b)。MWE を統語構造あるいは意味構造上の単位として取り扱う必要のある応用タスクでは、単語ベースの依存構造よりも MWE を考慮した依存構造を用いる方が望ましいと考えられる [11]。

従来、MWE 認識と依存構造解析は別個の自然言語処理タスクとして扱われてきたため、パイプラインモデルは MWE を考慮した依存構造解析を行う上で自然な選択肢である。パイプラインモデルは以下のステップからなる。まず、条件付き確率場 (CRF) の様な系列ラベリング手法によって MWE の範囲を予測し、各 MWE に品詞タグを割り当てる。次に、各 MWE を単一トークンとする依存構造解析を行う (図 2)。しかし、パイプラインモデルでは系列ラベリングからの誤り伝播によって精度が低下するという問題点がある。

近年、French Treebank をデータセットとして、MWE の範囲同定と依存構造解析の Joint モデルに関する研究が報告されている [11], [12]。Joint モデルでは MWE の範囲は、MWE の全ての後続要素が先頭要素を主辞に持つ Head-initial な依存構造 (図 3) として表され、これは Universal Dependency [13] の複合機能語の表現形式と同様である。

フランス語 MWE を考慮した依存構造解析では、多くの研究が French Treebank [14] を用いている。French Treebank では 18535 文中の 29718 回の MWE の出現が陽に注釈付けされている。一方、英語 MWE を考慮した依存構造解析に利用可能なコーパスはこれまで限られていた。上述した様

に Schneider ら [5] は、English Web Treebank 上に MWE の注釈付けを行ったが、これは依存構造解析器の訓練データとしては比較的小規模である (約 3800 文中の約 3400 回の MWE の出現)。また彼らの MWE アノテーションは句構造との整合性が保証されていない。

これとは対照的に、前述の MWE-aware dependency corpus [10] は Ontonotes の Wall Street Journal セクション全体をカバーし、37015 文中に 6960 回の MWE の出現が注釈付けされている。また、MWE の範囲と句構造木との整合性が保証されている。この大規模コーパスによって、英語 MWE を考慮した依存構造解析を行う事が可能である。

しかしながら、MWE を考慮した依存構造解析において、MWE 認識と依存構造解析の双方について高い精度を達成するモデルの構築方法は自明ではない。そこで我々は本研究で以下の 3 種のモデルを構築し (3章)、MWE を考慮した依存構造解析の性能を比較する。

### パイプラインモデル (3.1)

系列ラベリングで MWE の範囲を予測した後に、MWE を単一トークンとする依存構造解析を行うモデル

### Joint モデル (3.2)

MWE の範囲を係り受けラベルとして表現し、従来の単語ベースの依存構造解析を行うモデル

### Joint+supertagging モデル (3.3)

系列ラベリングで予測した MWE の範囲を Joint モデルの追加素性として用いるモデル

ここで Joint+supertagging モデルの狙いは以下の二点である。

- (1) MWE の範囲同定と依存構造解析のパイプラインモデルで生じる誤り伝播の問題を緩和する事ができる。
- (2) 系列ラベリングによる予測を追加素性として用いる事によって、文全体の情報を考慮できないという Joint モデルの欠点を緩和する事ができる。

上記に加え、MWE を含まない文において、パイプラインモデルの方が Joint+supertagging モデルよりも依存構造解析の精度が高い点を考慮して、我々は Joint+supertagging モデルとパイプラインモデルのハイブリッドモデル (3.4) を提案する。

我々は MWE-aware dependency corpus [10] を用いた実験を行い、以下の結果を得た (5章) :

- (1) パイプラインモデルと Joint モデルを比較した所、MWE 範囲同定において、パイプラインモデルの方が 4 ポイント以上高い F 値を示した。
- (2) Joint+supertagging モデルとパイプラインモデルを比較した所、MWE 範囲同定において、Joint+supertagging モデルの方が約 1 ポイント高い F 値を示した。
- (3) ハイブリッドモデルと Joint+supertagging モデルを比較した所、依存構造解析においてハイブリッドモデルの方が 0.14 ポイント高い UAS 値を示した。

Words	901,673
Sentences	37,015
#MWE Instances	6,960
#MWE Types	571

Length	2	3	≥ 4	total
#MWE Instances	5,667	1,077	216	6,960

表 1: Kato ら [10] のコーパスの統計量

次章以降は以下の様に構成される。2 章では利用したデータセットについて述べ、3 章では上述した各モデルを詳述する。4 章では実験設定を、5 章では実験結果と考察を述べる。6 章では関連研究について述べ、最後に 7 章でまとめを行う。

## 2. MWE-aware Dependency Corpus

本章ではデータセットとして用いた Kato ら [10] のコーパスについて簡単に述べる。このコーパスは “a number of”, “even though”, “after all” の様な英語の複合機能語を考慮した依存構造のツリーバンクである。句動詞の様な非連続 MWE や複合名詞はこのコーパスの MWE アノテーションには含まれていない。以下ではコーパス構築手順について述べる。まず、句構造木として Ontonotes Release 5.0 (LDC2013T19) の Wall Street Journal 部分を用い、句構造木と整合性が保たれる様に Shigeto ら [7] の MWE アノテーションを統合する。具体的には各 MWE の範囲が句構造木の非終端ノードに対応するかどうかを調査し、対応しないものに関しては、できるだけ元の木構造を残しつつ、MWE が単一の部分木になる様に句構造木を修正する。次に MWE に対応する部分木を単一のトークンにまとめる。このトークンの品詞タグは MWE の全体品詞であり、その表層形は、MWE の全ての構成単語の表層形をアンダースコアで連結したものである。最後に句構造を依存構造に変換する (図 1b) \*2。MWE を考慮した依存構造では MWE が構文上の単位として扱われており、一方、従来の単語ベースの依存構造では MWE の範囲と全体品詞は陽に表現されない (図 1a)。Kato ら [10] のコーパスでは 37015 文の内、6928 文が 1 つ以上の MWE を含んでいる。コーパス中の MWE の異なり数は 571 である。表 1 にコーパス統計量を示す。

## 3. モデル

### 3.1 パイプラインモデル

MWE の範囲同定と依存構造解析のパイプラインモデルは以下のステップからなる (図 2)。

- (1) MWE の範囲と全体品詞を系列ラベリング手法によって推定し、MWE を単一トークンにまとめる。

\*2 Kato ら [10] らは、Stanford Dependency を採用している。

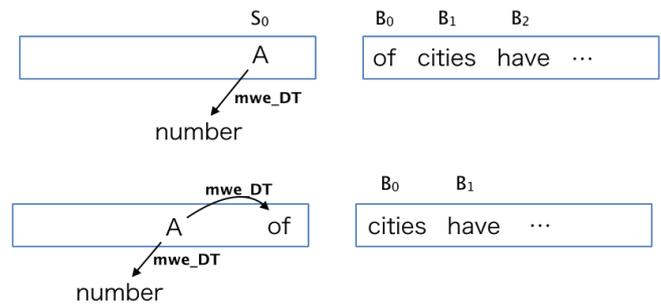


図 4: Joint モデルの依存構造解析で課す制約。上図から下図へと MWE 専用ラベル (“mwe\_DT”) 付きの RIGHT アクションで遷移するためには、MWE 辞書に基づく制約と遷移履歴に基づく制約が両方とも満たされている必要がある (3.2 節)。

- (2) MWE を単一トークンとした依存構造を ArcEager 法に基づく遷移ベースの解析器によって推定する。

ステップ (1) では

$$B_{-}\langle MWE\_POS\rangle, I_{-}\langle MWE\_POS\rangle, O\_POS$$

のタグセットを用いる。

例えば “a number of cities” で始まる文に対する正しいタグ系列は a / B\_DT, number / I\_DT, of / I\_DT, cities / O\_NNS となる。このタグセットでは、MWE の範囲に関する BIO タグと全体品詞のペアと、MWE でない単語の品詞タグとが統合されている。

### 3.2 Joint モデル

MWE の範囲と依存構造を同時に推定するために、Joint モデルでは MWE の範囲と全体品詞を係り受けラベルとして表現し、ArcEager 法に基づく遷移ベースの解析器を用いて従来の単語ベースの依存構造解析を行う。

データ表現

Joint モデルで利用する依存構造は、MWE を考慮した依存構造 (図 1b) 中の MWE を、MWE の範囲と全体品詞をエンコードした Head-initial な部分木に変換したもの (図 3) である。これは Universal Dependency [13] の複合機能語の表現形式に類似している。

制約

Joint モデルの依存構造解析では、明らかに誤った依存構造木を推定しない様にするために以下の制約を用いる (図 4)。

- (1) MWE 辞書に基づく制約 \*3:

MWE 辞書がスタックトップ ( $S_0$ ) からバッファの先頭 ( $B_0$ ) までのトークン列 (図では “a number of”) で始まるエントリを一つも含まないならば、依存構造解析器は MWE 専用ラベル付きの RIGHT アクションは選択しない。

\*3 本研究では Shigeto ら [7] によって作成された MWE 辞書を用いた。

## (2) 遷移履歴に基づく制約:

$S_0$  と  $B_0$  の間にあるいずれかのトークンが  $S_0$  を主辞として持たない、または、 $S_0$  から各トークンに向かうエッジの係り受けラベルが、このアクションの係り受けラベルに一致しないならば、依存構造解析器は MWE 専用ラベル付きの RIGHT アクションは選択しない。

### 3.3 Joint+supertagging モデル

我々の予備実験によると、パイプラインモデルは MWE 認識で高い F 値を示したが、依存構造解析で若干の性能低下が見られた。その一因としては系列ラベリングからの誤り伝播が考えられる。一方、Joint モデルは MWE 認識において、CRF を用いた系列ラベリング手法よりも低い F 値を示した。この結果は、MWE 内部係り受けとして表現した MWE の範囲と全体品詞を依存構造解析器で直接推定する事の難しさを示唆している。そこでこれらの欠点を緩和するために、我々は Joint+supertagging モデルを設計した。このモデルは系列ラベリングによる予測を Joint モデルに統合したものである。本モデルは具体的には以下のステップからなる。

(1) 3.1 で述べた様に、CRF を用いた系列ラベリング手法によって MWE の範囲と全体品詞を予測する。

(2) 予測した MWE の範囲と全体品詞を Joint モデルの追加素性として用いる。

パイプラインモデルでは系列ラベリングによる予測を決定的に用いて依存構造解析を行うのに対し、Joint+supertagging モデルでは系列ラベリングによる予測を素性として用いているために、系列ラベリングからの誤り伝播を緩和できると期待される。さらに追加素性が CRF に基づいている事から、遷移ベースのパーサーによって MWE を考慮した依存構造を直接推定する事の難しさを緩和できると期待される。依存構造解析で用いるデータ表現と制約は Joint モデルと同一である。

### 3.4 ハイブリッドモデル

予備実験で Joint+supertagging モデルをパイプラインモデルと比較した所、MWE を含む文ではより高い UAS / LAS 値となったが、MWE を含まない文では逆に精度の低下が見られた。そこで我々は、この問題を解決するために Joint+supertagging モデルとパイプラインモデルのハイブリッドモデルを提案する。具体的なアルゴリズムを以下に示す。まず、テスト文が MWE 辞書にマッチするトークン列を全く含まない場合には、パイプラインモデルの依存構造解析器で解析する。一方、辞書にマッチするトークン列が文中に存在する場合には、Joint+supertagging モデルを用いて解析を行う。

もし MWE 辞書のカバレッジが十分高ければ、辞書マッ

チによって、MWE が含まれる文かどうかを高い精度で判別できるため、ハイブリッドモデルは有効に機能する。本研究で実験に用いた Kato ら [10] のコーパスでは辞書に含まれる MWE のみをアノテーションしているため、この仮定が成り立つと考えられる。

### 3.5 素性

#### デフォルト素性

パイプラインモデルと Joint モデルの依存構造解析では、Redshift [16] のデフォルト素性を用いる。デフォルト素性は局所素性と大域素性からなっている（これらはそれぞれ Zhang ら [15] の表 1, 表 2 に対応している）。

#### MWE 素性

Joint+supertagging モデルでは Zhang ら [15] の局所素性を拡張した MWE 素性を Redshift のデフォルト素性に加えて用いる。MWE 素性を構成する原子素性としては単語の表層形と単語レベルの品詞タグだけでなく MWE 全体品詞と、MWE の範囲を表現した BIO タグも用いている。MWE 素性の一覧を表 2 に示す。

## 4. 実験設定

まずはじめに、MWE を考慮した依存構造解析タスクについて述べる。全てのモデルにおいて入力単語分割と品詞タグ付けが行われた文であり、推定対象は MWE を考慮した依存構造木 (図 1b) と各 MWE の全体品詞である。我々はこのタスクにおける各モデルの性能を比較する。

### 4.1 前処理

品詞タグ付けについては Stanford POS tagger [17] を用いた。訓練セット (セクション 02-21) への品詞タグ付けは 20-way jackknifing を用いて行った。テストセット (セクション 23) への品詞タグ付けは訓練セットで訓練したモデルを用いて行った\*4。

### 4.2 CRF による MWE 認識の実験設定

パイプラインモデル、Joint+supertagging モデル、ハイブリッドモデルについては CRF の実装として CRFSuite [18] を採用した。CRF の素性としては、単語の表層形を原子素性とする組み合わせ素性や、文字、文字種などに加えて MWE 辞書素性を用いた\*5。辞書素性の原子素性としては、マッチした辞書エントリに基づく BIO タグを利用した\*6。辞書素性としては、BIO タグの予測対象トークンを中心とする幅 5 のウインドウ内の原子素性に基づき、1-3 gram の組み合わせ素性を採用した。訓練セット (セクショ

\*4 テストセットにおける品詞タグ付けの精度として 97.1%を得た。

\*5 単語の品詞タグは予測対象であるため、素性に含めていない。

\*6 例えば “a number of” という MWE 辞書エントリが文中に出現しているならば、“a”, “number”, “of” の各トークンにそれぞれ “B.DT”, “I.DT”, “I.DT” という素性値を与えている。

from single words
foreach LOC in $[S_0, B_0, B_1, B_2]$ : LOCm; LOCb; LOCwm; LOCwb; LOCmb; LOCpm; LOCpb; LOCpmb; LOCwmb; LOCwpm; LOCwpb; LOCwpmb;
from word pairs
$\{S_0p, S_0wp\} \otimes \{B_0m, B_0wm\}$ $\{S_0b, S_0wb\} \otimes \{B_0b, B_0wb\}$ $\{S_0p, S_0w, S_0wp\} \otimes \{B_0mb, B_0pmb, B_0wmb, B_0wpmb\}$ $\{S_0mb, S_0pmb, S_0wmb, S_0wpmb\} \otimes \{B_0w, B_0p, B_0wp, B_0mb, B_0pmb, B_0wmb, B_0wpmb\}$ Atomic ( $B_0$ ) $\otimes$ Atomic ( $B_1$ ) ( $B_0p B_1p$ を除く)
from three words
$B_0mB_1pB_2p$ ; $B_0pB_1mB_2p$ ; $B_0pB_1pB_2m$ ; $S_0mB_0pB_1p$ ; $S_0pB_0mB_1p$ ; $S_0pB_0pB_1m$ ; $S_1mS_0pB_0p$ ; $S_1pS_0mB_0p$ ; $S_1pS_0pB_0m$ ; Atomic ( $B_0$ ) $\otimes$ Atomic ( $B_1$ ) $\otimes$ Atomic ( $B_2$ ) ( $B_0p B_1p B_2p$ を除く) Atomic ( $S_0$ ) $\otimes$ Atomic ( $B_0$ ) $\otimes$ Atomic ( $B_1$ ) ( $S_0p B_0p B_1p$ を除く) Atomic ( $S_1$ ) $\otimes$ Atomic ( $S_0$ ) $\otimes$ Atomic ( $B_0$ ) ( $S_1p S_0p B_0p$ を除く) Atomic ( $S_0$ ) $\otimes$ Atomic ( $S_0l$ ) $\otimes$ Atomic ( $B_0$ ) ( $S_0p S_0lp B_0p$ を除く) Atomic ( $S_0$ ) $\otimes$ Atomic ( $S_0r$ ) $\otimes$ Atomic ( $B_0$ ) ( $S_0p S_0rp B_0p$ を除く) Atomic ( $S_0$ ) $\otimes$ Atomic ( $B_0$ ) $\otimes$ Atomic ( $B_0l$ ) ( $S_0p B_0lp B_0lp$ を除く)

表 2: Zhang ら [15] の局所素性を元に拡張した MWE 素性.

w は単語の表層形, p は品詞タグ, m は MWE の全体品詞, b は MWE の BIO タグを示す.

LOC はコンフィグレーション中の位置を示す (例: スタックトップ ( $S_0$ ), バッファの先頭要素 ( $B_0$ )).

LOC l/r は LOC の最左/最右の子ノードを, Atomic (LOC) は {LOCp, LOCmb, LOCpmb} を示す.

例えば “a number of” で始まる文に対して CRF に基づく系列ラベリング手法で以下のタグ系列が推定された

とする: a / B.DT, number / L.DT, of / L.DT. この場合, “a” の MWE 全体品詞と BIO タグは各々 “DT”, “B” となる.

Model	依存構造解析				MWE 認識	
	テストセット全体		MWE を含む文		FUM	FTM
Pipeline	91.63	79.23	89.81	77.62	96.60	95.90
Joint	91.57	79.28	89.47	77.50	92.27	91.33
Joint+supertagging	91.51	79.24	90.06	78.21 <sup>†</sup>	97.70 <sup>*‡</sup>	97.20 <sup>**‡</sup>
Hybrid	91.65	79.34	90.06	78.21 <sup>†</sup>	97.70 <sup>*‡</sup>	97.20 <sup>**‡</sup>

表 3: テストセットに対する実験結果. シャッフルした事例を用いて訓練を行っているため, 本表の各セルに示した数値は 3 回の独立した訓練/推定のイテレーションで得られた精度の平均である. 統計的有意性の有無については各イテレーションで McNemar 検定を行った. † と ‡ は Joint モデルに対しての有意性が全イテレーションで確認された事を示す (各々, p-value ≤ 0.02 及び p-value ≤ 0.001). \* と \*\* はパイプラインモデルに対しての有意性が全イテレーションで確認された事を示す (各々, p-value ≤ 0.02 及び p-value ≤ 0.003).

ン 02-21) への BIO タグ付けは 20-way jackknifing を用いて行った. テストセット (セクション 23) への BIO タグ付けは訓練セットで訓練したモデルを用いて行った<sup>\*7</sup>.

#### 4.3 依存構造解析の実験設定

依存構造解析器については全てのモデルで Redshift [16]

を用いた. Redshift では, 学習に平均化パーセプトロン [19], 推論にビームサーチを採用している. 全ての実験でビームサイズは 16 に設定した. 依存構造解析器として ArcEager 型の遷移システムを採用している事を考慮し, 交差を持つ依存構造木を訓練データから除去した上で学習を行った. 一方, テスト時はテストセット中の全事例を解析した.

#### 4.4 評価

依存構造解析の評価指標としてはラベルなし正解率

<sup>\*7</sup> テストセットについては FUM (MWE の範囲の予測に関する F 値) として 96.60, FTM (MWE の範囲と全体品詞の同時予測に関する F 値) として 95.90 を得た.

(UAS ; unlabeled attachment scores) 及びラベルあり正解率 (LAS ; labeled attachment scores) を用いた。パイプラインモデルについては MWE の構成トークンを連結して作成したトークンを Head-initial な構造に変換した後に、正解の依存構造と比較した。Joint モデル, Joint+supertagging モデル, ハイブリッドモデルについては、推定された依存構造木を正解の依存構造と直接比較した。

MWE 認識の評価指標としては FUM / FTM (The F-measure for untagged / tagged MWEs) を用いた。FUM は MWE の範囲の予測に関する F 値である。一方, FTM は MWE の範囲と全体品詞の同時予測に関する F 値である。パイプラインモデルについては CRF が推定した MWE の範囲と全体品詞を正解データと比較した。Joint モデル, Joint+supertagging モデル, ハイブリッドモデルについては、推定された依存構造中の係り受けラベルとして表現された MWE の範囲と全体品詞を正解データと比較した<sup>\*8</sup>。

## 5. 実験結果と考察

実験結果を表 3 に示す。まずテストセット全体における依存構造解析の結果を見ると、パイプラインモデルは Joint+supertagging モデルを UAS 値で 0.12 ポイント上回っている。テストセットの 1640 文中、1321 文が MWE 辞書にマッチしない事を考慮すると、MWE を含まない文に関しては Joint+supertagging モデルよりもパイプラインモデルの方がより適している事をこの実験結果は示唆している。一方、ハイブリッドモデルの精度は他の全てのモデルを上回り、Joint+supertagging モデルよりも 0.14 / 0.10 ポイント高い UAS / LAS 値となった。ハイブリッドモデルのパイプラインモデルに対するわずかな改善は、MWE を含む文についてはパイプラインモデルよりも Joint+supertagging モデルの方が適している事を示唆している。ただ総じて各モデルの精度には余り大きな差は見られなかった。

次に MWE を含む文 (266 事例) における依存構造解析の結果を見ると、Joint+supertagging モデルの UAS / LAS 値はパイプラインモデルを 0.25 / 0.59 ポイント、Joint モデルを 0.59 / 0.71 ポイント上回っている<sup>\*9</sup>。従って Joint+supertagging モデルで用いている MWE 素性は、MWE の周辺や MWE 内部の依存構造の推定に有効である事が分かった。ここで Joint+supertagging モデルの、依存構造解析に対する Supertagging モデル [20] との類似点について言及しておく。Ouchi ら [20] は、主辞の位置と係り受けラベルについての情報を Supertag にエンコードし、系列ラベリングによって予測した Supertag の系列を追加素性として依存構造解析を行っている。同様に我々の Joint+supertagging モデルは

MWE の内部係り受けについての情報を BIO タグにエンコードし、系列ラベリングによって予測した BIO タグの系列を追加素性として依存構造解析を行っている。

最後に MWE 認識について見ると、Joint+supertagging モデルとハイブリッドモデルはパイプラインモデルを FUM / FTM 値で 1.1 / 1.3 ポイント上回った<sup>\*10</sup>。一方、Joint モデルは他のモデルに対して FUM / FTM 値で少なくとも 4.0 ポイント下回る結果となった。これらの実験結果は、依存構造解析器による MWE 認識において MWE 素性が有効である事と、MWE 素性無しでは CRF による MWE 認識に及ばない事を示唆している。また、MWE 認識で Joint モデルが相対的に劣る結果になった一因として、Joint モデルでは Head-initial な依存構造 (図 3) を表現するために MWE 専用の係り受けラベルを追加した結果、パイプラインモデルよりも出力空間が広がっており、このため、より多くの探索誤りが発生しうる事が挙げられる。

定性的分析として、まず Joint+supertagging モデルがパイプラインモデルよりも正確に MWE 認識を行った事例を紹介する。図 5 でパイプラインモデルは “in cash” を誤って MWE と推定している (図 5b)。一方、Joint+supertagging モデルはこの文中の “in cash” は MWE ではない事と、“cash equivalents” が複合名詞である事を正しく推定している (図 5a)。次にパイプラインモデルと Joint+supertagging モデルが Joint モデルよりも正確に MWE 認識を行った事例を紹介する。図 6 でパイプラインモデルと Joint+supertagging モデルは “thanks to” が MWE である事を正しく推定している (図 6a)。一方、Joint モデルはこの MWE の検出に失敗している (図 6b)。

## 6. 関連研究

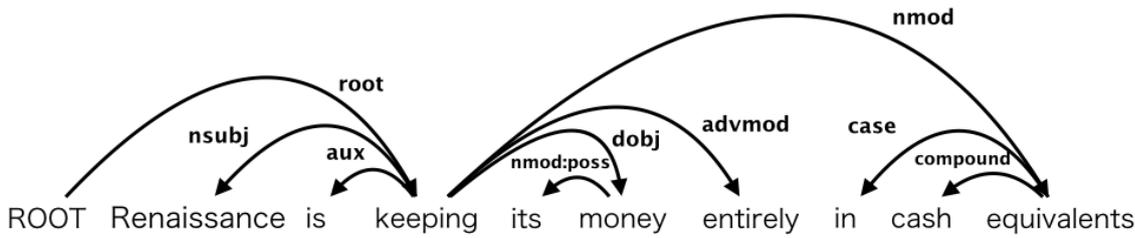
MWE を考慮した依存構造解析については、1 章で言及した Candito ら [11] の研究の他に以下の 2 つの関連研究が挙げられる。まず、Constant ら [21] は、文中の MWE の範囲に関する情報を依存構造木とは別の木として表現する事を提案している。彼らは MWE 認識と依存構造解析に対して、パイプラインまたは同時解析によって easy-first の枠組みで取り組み、MWE 認識が依存構造解析を助けるという実験結果を得ている。また、Constant ら [12] は、依存構造木と、lexical unit (Fixed MWE もしくは MWE 以外のトークン) の森の双方を推定するための遷移ベースの解析器を提案している。

MWE を考慮した依存構造のデータ表現の関連研究と

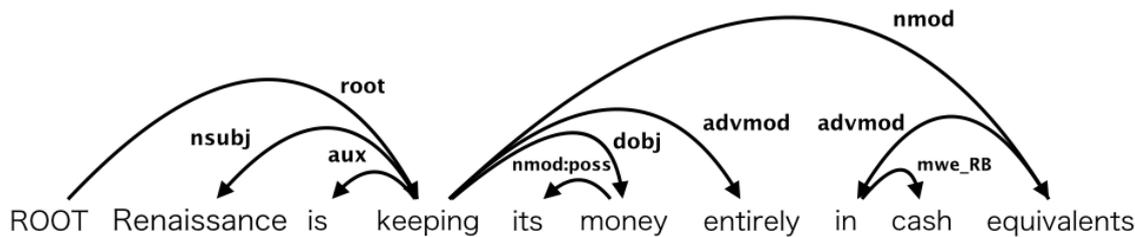
<sup>\*8</sup> MWE の構成トークン間のエッジの係り受けラベルは MWE 専用の接頭辞 “mwe\_” を持つ。

<sup>\*9</sup> ハイブリッドモデルは MWE を含む文については Joint+supertagging モデルの推定結果を採用するため、その UAS / LAS 値は Joint+supertagging モデルと同一である。

<sup>\*10</sup> ハイブリッドモデルの FUM / FTM 値は Joint+supertagging モデルと同一である。その理由を以下に示す。(1) ハイブリッドモデルはテスト文が MWE 辞書にマッチしない場合はパイプラインモデルの依存構造解析器で解析を行う (3.4)。(2) Joint+supertagging モデルは MWE 辞書にマッチしない文に対して MWE を検出する事は無い。これは MWE 辞書に基づく制約 (3.2) を課しているためである。

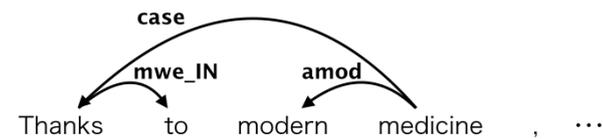


(a) Joint+supertagging モデルによって推定された依存構造

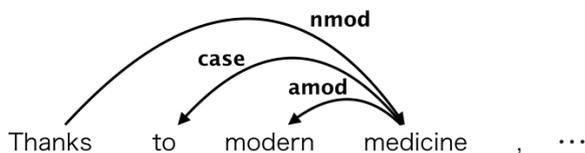


(b) パイプラインモデルによって推定された依存構造

図 5: Joint+supertagging モデルが正しい推定を行ったのに対し、パイプラインモデルが誤った推定を行った事例



(a) パイプラインモデルと Joint+supertagging モデルによって推定された依存構造



(b) Joint モデルによって推定された依存構造

図 6: パイプラインモデルと Joint+supertagging モデルが正しい推定を行ったのに対し、Joint モデルが誤った推定を行った事例

しては Universal Dependency [13] が挙げられる。Universal Dependency は言語横断的に一貫した基準で複数の言語についてツリーバンクを構築する事を目標としている。英語の Universal Dependency における複合機能語は、MWE の全ての後続要素が先頭要素を主辞に持つ Head-initial な構造でアノテーションされており、その係り受けラベルは MWE 専用 (“mwe”) となっている。また、英語 MWE を考慮した句構造解析については Cafferkey ら [22] が、MWE を単一トークンにまとめてから句構造解析を行い、その後、MWE 由来のトークンに対応する部分木に置換する手法を提案している。彼らは外部資源をテスト文にマッピングできるかチェックした後に固有表現認識器を適用する事によって MWE 認識を行っている。

## 7. 結論

本研究では英語 MWE を考慮した依存構造解析タスクにおいて、パイプラインモデル、Joint モデル、Joint+supertagging モデルの 3 種のモデルを構築し、実験を行った。その結果、MWE の範囲同定において、Joint+supertagging モデルはパイプラインモデル、Joint モデルの双方を F 値で上回った。また、我々は Joint+supertagging モデルとパイプラインモデルのハイブリッドモデルを構築し、依存構造解析において Joint+supertagging モデルよりも高い精度を得た。

## 参考文献

- [1] Sag, I. A., Baldwin, T., Bond, F., Copestake, A. A. and Flickinger, D.: Multiword Expressions: A Pain in the Neck for NLP, *Proceedings of the Third International Conference on Computational Linguistics and Intelligent Text Processing, CICLing '02*, London, UK, UK, Springer-Verlag, pp. 1–15 (2002).
- [2] Carpuat, M. and Diab, M.: Task-based Evaluation of Multiword Expressions: a Pilot Study in Statistical Machine Translation, *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, Association for Computational Linguistics, pp. 242–245 (2010).
- [3] Newman, D., Koilada, N., Lau, H. J. and Baldwin, T.: Bayesian Text Segmentation for Index Term Identification and Keyphrase Extraction, *Proceedings of COLING 2012*, The COLING 2012 Organizing Committee, pp. 2077–2092 (2012).
- [4] Berend, G.: Opinion Expression Mining by Exploiting Keyphrase Extraction, *Proceedings of 5th International Joint Conference on Natural Language Processing*, Asian Federation of Natural Language Processing, pp. 1162–1170 (2011).
- [5] Schneider, N., Onuffer, S., Kazour, N., Danchik, E., T. Mordowanec, M., Conrad, H. and A. Smith, N.: Comprehensive Annotation of Multiword Expressions in a Social Web Corpus, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*, Euro-

- pean Language Resources Association (ELRA) (2014).
- [6] Bies, A., Mott, J., Warner, C. and Kulick, S.: English Web Treebank, *Technical Report LDC2012T13, Linguistic Data Consortium, Philadelphia, Pennsylvania, USA.* (2012).
- [7] Shigeto, Y., Azuma, A., Hisamoto, S., Kondo, S., Kouse, T., Sakaguchi, K., Yoshimoto, A., Yung, F. and Matsumoto, Y.: *Proceedings of the 9th Workshop on Multiword Expressions*, chapter Construction of English MWE Dictionary and its Application to POS Tagging, pp. 139–144, Association for Computational Linguistics (2013).
- [8] Pradhan, S. S., Hovy, E. H., Marcus, M. P., Palmer, M., Ramshaw, L. A. and Weischedel, R. M.: OntoNotes: A Unified Relational Semantic Representation, *Proceedings of the First IEEE International Conference on Semantic Computing (ICSC 2007), September 17-19, 2007, Irvine, California, USA*, pp. 517–526 (2007).
- [9] Finkel, R. J. and Manning, D. C.: Joint Parsing and Named Entity Recognition, *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, Association for Computational Linguistics, pp. 326–334 (2009).
- [10] Kato, A., Shindo, H. and Matsumoto, Y.: Construction of an English Dependency Corpus incorporating Compound Function Words, *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016.* (2016).
- [11] Candito, M. and Constant, M.: Strategies for Contiguous Multiword Expression Analysis and Dependency Parsing, *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, pp. 743–753 (2014).
- [12] Constant, M. and Nivre, J.: A Transition-Based System for Joint Lexical and Syntactic Analysis, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, pp. 161–171 (2016).
- [13] McDonald, R., Nivre, J., Quirnbach-Brundage, Y., Goldberg, Y., Das, D., Ganchev, K., Hall, K., Petrov, S., Zhang, H., Täckström, O., Bedini, C., Bertomeu Castelló, N. and Lee, J.: Universal Dependency Annotation for Multilingual Parsing, *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Association for Computational Linguistics, pp. 92–97 (2013).
- [14] Abeillé, A., Clément, L. and Toussanel, F.: Building a Treebank for French, *Treebanks: Building and Using Parsed Corpora*, Springer, pp. 165–188 (2003).
- [15] Zhang, Y. and Nivre, J.: Transition-based Dependency Parsing with Rich Non-local Features, *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, Association for Computational Linguistics, pp. 188–193 (2011).
- [16] Honnibal, M., Goldberg, Y. and Johnson, M.: *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, chapter A Non-Monotonic Arc-Eager Transition System for Dependency Parsing, pp. 163–172, Association for Computational Linguistics (2013).
- [17] Toutanova, K., Klein, D., Manning, C. and Singer, Y.: Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network, *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics* (2003).
- [18] Okazaki, N.: CRFsuite: a fast implementation of Conditional Random Fields (CRFs) (2007).
- [19] Collins, M.: Discriminative Training Methods for Hidden Markov Models: Theory and Experiments with Perceptron Algorithms, *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, pp. 1–8 (2002).
- [20] Ouchi, H., Duh, K. and Matsumoto, Y.: Improving Dependency Parsers with Supertags, *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, volume 2: Short Papers*, Association for Computational Linguistics, pp. 154–158 (2014).
- [21] Constant, M., Le Roux, J. and Tomeh, N.: Deep Lexical Segmentation and Syntactic Parsing in the Easy-First Dependency Framework, *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Association for Computational Linguistics, pp. 1095–1101 (2016).
- [22] Cafferkey, C., Hogan, D. and van Genabith, J.: Multi-word units in treebank-based probabilistic parsing and generation., *Proceedings of the 10th International Conference on Recent Advances in Natural Language Processing (RANLP-07)*, Borovets, Bulgaria (2007).

ページ番号	章番号	行番号	誤	正
5	5章	表3	-	別表を参照
6	5章	9-10行目	0.14 / 0.10 ポイント高い UAS / LAS 値となった.	0.14 / <b>0.16</b> ポイント高い UAS / LAS 値となった.
6	5章	16-17行目	パイプラインモデルを 0.25 / 0.59 ポイント,	パイプラインモデルを 0.25 / <b>0.62</b> ポイント
6	5章	17-18行目	Joint モデルを 0.59 / 0.71 ポイント	Joint モデルを 0.59 / <b>0.67</b> ポイント

正誤表

Model	依存構造解析 テストセット全体				MWE 認識	
	UAS	LAS	UAS	LAS	FUM	FTM
Pipeline	91.63	90.47	89.81	88.13	96.60	95.90
Joint	91.57	90.48	89.47	88.08	92.27	91.33
Joint+supertagging	91.51	90.41	90.06	88.75	97.70*‡	97.20**‡
Hybrid	91.65	90.57	90.06	88.75	97.70*‡	97.20**‡

表 3: テストセットに対する実験結果. シャッフルした事例を用いて訓練を行っているため, 本表の各セルに示した数値は 3 回の独立した訓練/推定のイテレーションで得られた精度の平均である. 統計的有意性の有無については各イテレーションで McNemar 検定を行った. ‡ は Joint モデルに対しての有意性が全イテレーションで確認された事を示す ( $p\text{-value} \leq 0.001$ ). \* と \*\* はパイプラインモデルに対しての有意性が全イテレーションで確認された事を示す (各々,  $p\text{-value} \leq 0.02$  及び  $p\text{-value} \leq 0.003$ ).