日本語意味解析システムSAGEの高速化・高精度化とコーパスによる精度評価

原田 実[†] 田淵和幸[††,☆] 大野博之[††,☆☆]

原田研究室ではこれまで，EDR 電子化辞書に記載された情報を元に，日本語文を意味解析し格フレーム群に自動変換するシステム SAGE（Semantic frame Automatic GEnerator）を開発してきた．既存の SAGE は機能的には正しく動作するが，解析時間が文節数の指数オーダのため実利用するには問題があった．また解析精度に対する客観的な検証がされていなかった．そこで本研究では，Jiri らによる英文の構文木への語意割当て用の高速アルゴリズムの考え方を SAGE における係り受け木への語意と格決定用に適用して，SAGE の解析速度を向上させた．この結果，解析速度は文節数の線形オーダになった．また，EDR の解析済みコーパスを用いて SAGE の解析精度を自動的に評価するシステムを開発した．その結果，語意正解率は 81.1%，格正解率は 60.7%，格の宛先正解率は 73.3%であった．これによって SAGE は速度面でも精度面でも意味解析システムとして実利用を開始できるレベルに至ったといえる．

Improvement of Speed and Accuracy of Japanese Semantic Analysis System SAGE and Its Accuracy Evaluation by Comparison with EDR Corpus

Minoru Harada,[†] Kazuyuki Tabuchi[††,☆] and Hiroyuki Oono[††,☆☆]

In the Harada laboratory, a semantic analysis system SAGE (Semantic frame Automatic GEnerator) has been developed, which converts a Japanese sentence into case frames based on the statistical information in the EDR electronic dictionary. Though SAGE operated correctly, there was such a problem in actual use that it requires the time of the exponential order of the number of clauses. In this research, based on Jiri's deterministic algorithm for assigning the word meaning to nodes of the parse tree of English sentence, the deterministic algorithm for deciding the meaning of words represented by nodes and the deep case of the relations among such nodes in the dependency tree of Japanese sentence is developed. As a result, the analytical speed became the linear order of the number of clauses. Moreover, the system to evaluate the analytical accuracy of SAGE is developed by using EDR analyzed Corpus. This evaluation revealed that the word meaning accuracy is 81.1%, the destination accuracy of case relation is 73.3% and the case relation accuracy is 60.7%. As a result, it can be said that SAGE has reached to the level that we can begin its acctual use for Japanese semantic analysis.

1. はじめに

原田研究室ではこれまで，EDR 電子化辞書に記載された情報を元に，日本語文を意味解析し格フレーム群に自動変換するシステム SAGE98（Semantic frame Automatic GEnerator）[9] とその改良版の SAGE99[2),8)] を開発し，オブジェクト指向分析システム CAMEO[12),13)] の自然語要求仕様の意味解析などに応用してきた．この SAGE は機能的には正しく動作するが，解析時間において，実利用するには十分なレベルに達していない．また解析精度に対する客観的な検証がまだされていない．本研究の目的は，解析精度と解析速度の両面で実用可能レベルの意味解析システム SAGE2000 の開発を行うことであり，具体的には以下の 3 つを行う．

† 青山学院大学理工学部情報テクノロジー学科
Department of Integrated Information Technology, Aoyama Gakuin University
†† 青山学院大学理工学研究科経営工学専攻
Graduate School of Industrial and System Engineering, Aoyama Gakuin University
☆ 現在，株式会社 NTT データ
Presently with NTT Data Corporation
☆☆ 現在，日本電気株式会社
Presently with NEC Corporation

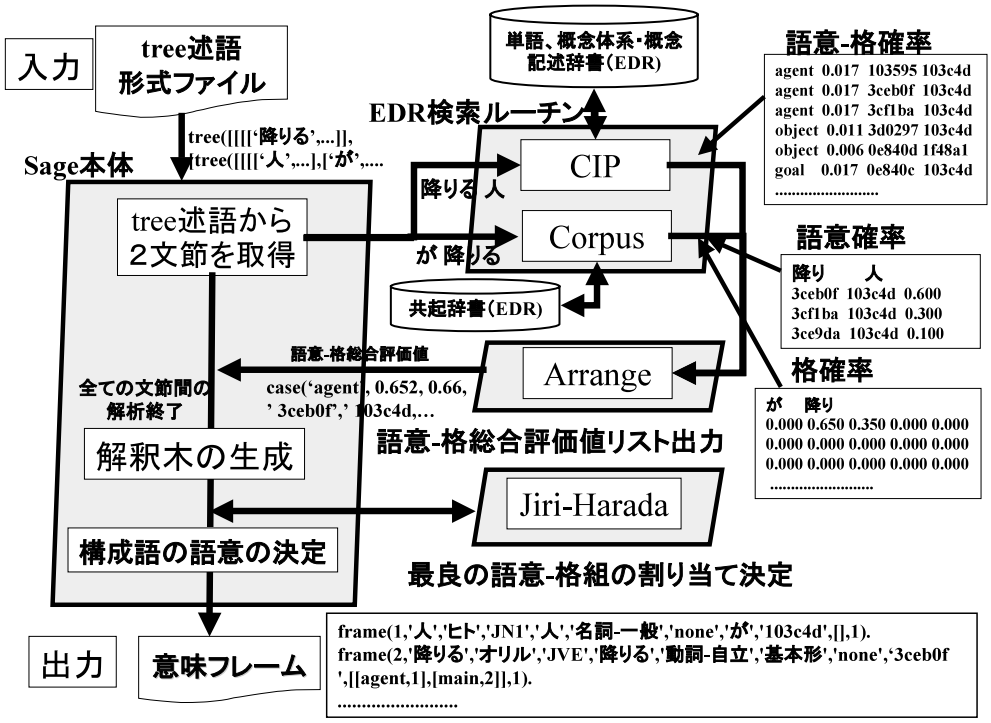

図 1 SAGE の基本処理の流れ
Fig. 1 Basic system flow of SAGE.

1) SAGE の高精度化：従来の SAGE では，係り受け関係にある用言と体言については EDR 辞書より統計情報を基に語意と格を決定するが，用言と用言間の格については，辞書にその間の格に関するデータが記録されていないので，この間の格を決定できなかった．これについては接続助詞などの表層的な情報から求める経験的なルールを作成して決定する．
2) SAGE の高速化：Jiri らによる英文の構文木への語意割当て用の高速アルゴリズムを SAGE における係り受け木への語意と格決定用に拡張して解析速度を向上させる．
3) 解析精度の自動評価：EDR の解析済みコーパスを用いて SAGE の解析精度を自動的に評価する．

なお，従来の SAGE99 と本研究で開発した SAGE 2000 における意味解析の処理の流れに大きな差はない．その差は，後で述べるように，係り受け木に対する語意と格の割当てアルゴリズムを高速化したこととその解析精度向上の工夫をしたことである．

日本語意味解析の研究としては，初期の段階のものとして電子化辞書を使用しない平川ら[3] の研究がある．また Jiri らは，構文木中の各節の語意を決定する高速アルゴリズムを提案している[4]．一方，このような意味解析研究以外に，電子化辞書を用いた研究には，解析済みの例文の集合であるコーパスから格フレームを獲得する東ら[10] の研究や共起パターン辞書や格フレーム辞書から入力文の格フレーム候補を得てこれらのうちのどれが最もよく入力文を表しているかを求める格フレームの選択に関する内山ら[11] や黒橋ら[5] の研究がある．これらについては SAGE99[2] でも論じたように，いずれも我々の研究とは目的が異なる．

一方，我々の研究は EDR という本格的な電子辞書を意味解析に用いるには，どのようなシステム構成をとればよいか，各辞書からの情報をどのように総合すればよいか，辞書の不完全な箇所をどう補完すべきか，どう高速化すればよいかなどに重点をおいた工学的な研究である．

2. SAGE での意味解析の概要

SAGE で入力した日本語文章を意味解析する前段階として形態素解析と係り受け解析を行う．本研究では，形態素解析システムには『茶筌』を，係り受け解析システムには『茶掛』を利用した．これらのシステムは奈良先端科学技術大学院大学の松本研究室で開発されたツールである[7]．また preSAGE では，茶掛の出力ファイルを，prolog で扱いやすい形のリスト形式（tree 述語形式）に変換する．この tree 述語を

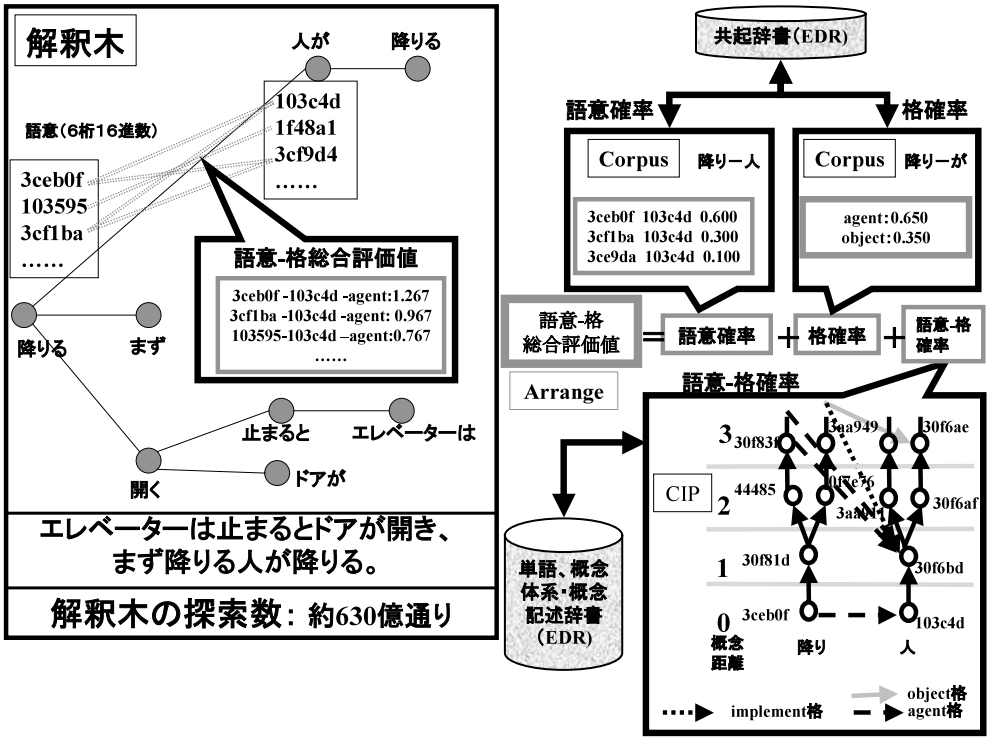

図 2 解釈木と EDR 辞書からの確率
Fig. 2 Interpretation tree and probability gained from EDR dictionary.

受けて SAGE は意味解析を行う．SAGE99 は，**図 1** に示すように EDR を元に意味解析を行うシステムであり，SAGE 本体，CIP，Corpus，Arrange という 4 つのコンポーネントからなる．処理の流れを「人が降りる」という例にそって説明する．なお，図中の語意確率の「人」の下の「103c4d」などの 6 桁の 16 進数は，人に対する EDR 辞書における語意を表す概念 Id で，また語意–格確率の agent や object などは EDR 辞書における深層格である．SAGE 本体が tree 述語形式ファイルを読み込み，そこから係り受け関係にある 2 文節（例では「人が」と「降りる」）を取り出す．このとき,「降りる」にあたる語を係り先語,「人が」にあたる語を係り元語と呼ぶ（依存文法では前者を支配語，後者を従属語と呼ぶ）．これら 2 文節を CIP と Corpus に引き渡す．CIP では，**図 2** にも示すように，渡された 2 文節の中心語（「人」と「降りる」）の語意とそれらの間にどのような格関係が考えられるのかを EDR 辞書で検索し，それぞれの語意と格の組合せ（これを語意–格組と呼ぶ）の尤もらしさを語意–格確率として求める．具体的には，EDR 単語辞書を検索して 2 語の語意 m1，m2 を求め，次にこれらの上位概念 um1，um2 を EDR 概念体系辞書を検索して求め，最終的にある格 c を介して語意–格組 (um1,c,um2) が EDR 概念記述辞書に存在すれば，p = 1/(m1 と um1 の概念距離 + m2 と um2 の概念距離) を語意–格組 (um1,c,um2) の実質的な出現率と考え，このような率 p をすべての上位概念の組合せに対して求めその合計値 S を用いて，p/S を (um1,c,um2) の語意–格確率とする．Corpus では，助詞（「〜が」）と係り先語（「降りる」）から，その助詞と単語がともに出現した場合の係り元語と係り先語間における格 c の出現確率を EDR 共起辞書から求めて格確率とする．さらに，係り先中心語（「降りる」）と係り元中心語（「人」）から，この 2 つの語がともに出現した場合の，2 語の語意の組 (m1,m2) の出現確率を EDR 共起辞書から求めて語意確率とする．これらの 3 つの確率の詳しい求め方は文献 2) にある．Arrange では個々の語意–格組ごとに語意–格確率と格確率と語意確率の和を語意–格総合評価値として算出し，図 1 に示すように Case 述語として SAGE 本体に引き渡す．

これらの作業を係り受け関係にあるすべての 2 文節に対して行い，図 2 に示すように文の係り受け木の各枝が表す語意–格組にその語意–格総合評価値を割り当てる．これを解釈木と呼び，これらの解釈木ごとにすべての枝に対する語意–格総合評価値の和を求める．これを確信度と呼ぶ．なお，ここで「文全体の意味とし

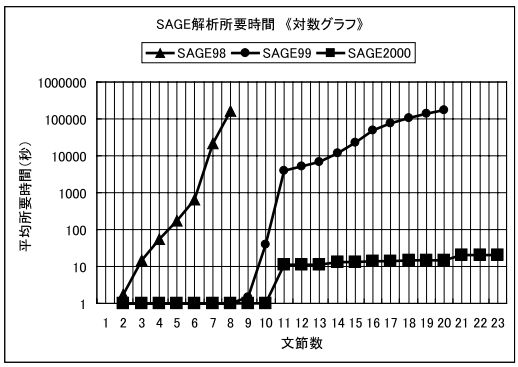

図 3 解析所要時間
Fig. 3 Analysis time.

ては文末の述語が重要である」などの応用的見地を考えれば，解釈木の根に近い語ほど重い重み付けをして和をとることも考えられるが，目的は各語の語意を決定することであり，その点においては各語に重みの差はないと考え単純和とした．このような重み付けした和を用いた実験を繰り返したが，精度に有意な差が現れなかったので最終的に単純和とした．あらゆる解釈木の中から，確信度が最も大きくなるような解釈木を統計的に尤もらしい木として採択する．この最良の解釈木が決定されると，各枝に割り当てられた語意–格総合評価値の 2 つの語意を両端節の語の語意とし，格をその間の格として frame 述語形式で出力する．

係り受け木における 1 つの枝には多くの語意–格組候補があり，解釈木の数も膨大なものとなる．図 2 に示す「エレベータは止まるとドアが開き，まず降りる人が降りる」という例に対しては，約 630 億通りもの解釈木が存在する．SAGE99 における最大確信度を求めるアルゴリズムは基本的にはこれらすべての組合せを生成しその最大を探索していた．したがって，最良の解を得ることはできるが，**図 3** の SAGE98 に示すように指数オーダの時間がかかる．この問題点に対し水野ら[8] は精度を落とさないことを基本方針に,「最大評価優先法」と「分枝限定法」という 2 つの手法による高速化を提案・実装した．その結果として図 3 の SAGE99 に示すように約 10 文節までの解析を数秒で行うことに成功した．しかし 10 文節を超えた文に対しては実用的な時間内では解析できず，線形オーダのアルゴリズムが求められていた．

3. SAGE の高速化

3.1 Jiri アルゴリズム

Jiri らは，英文の構文木中の各節の語意を決定する高速アルゴリズムを提案している．彼らは，係り受け関係における係り先語（head）と係り元語（modifier）の間の構文的な関係を分類し，それごとに両語の語意の確率をコーパスから統計的に求め，これを関係行列 R（relational matrix）として算出している．各語の語意の決定は，まず語ごとにその様々な語意の確率ベクトル M（sense score vector）の初期値を統計的に求め，次に構文木の葉から始めて，それらが修飾している head との関係行列から head の sense score vector を更新する．同時に各 modifier の sense score vector を head の語意ごとに並べて意味得点行列 Q（sense score matrix）とする．この過程を head が構文木の根になるまで行い，根の sense score vector が確定すると，その中の最大確率を持つ語意を根の語意とする．ここまでを Bottom-up 集約という．これが決定すると今度は Top-down 決定を行う．ここでは根から始めて，modifier の語意を順に決定していく．この際，すでに決まっている head の語意を固定して，その中で sense score matrix 要素が最大値になる modifier の語意を決定する．このプロセスは探索を含まず決定的に行われるので高速に実行できる．

3.2 Jiri-Harada アルゴリズム

我々の目的は各語の語意を決定するだけでなく，語間の深層格も決定するということにある．したがって，語意の決定も単純に語ごとの語意確率というよりは，他の語との深層格の関係においての語意の確率を重要視している．Jiri らの方法も確かに語意を表す sense vector の更新を他の語の関係を表す relational matrix を用いて行っているが，我々は係り受け関係にある 2 語の語意とその間の深層格の 3 つ組ごとにその出現確率を用いる方がこれらの最適値を決定するにはより適切であると考えている．そこで我々は以下のように Jiri らのアルゴリズムを拡張した．ここでは主に語意–格組の出現確率の算出方法を変更し，Bottom-up 集約と Top-down 決定という全体的なアルゴリズムの流れは同様とした．**図 4** にそって以下にそのアルゴリズムを示す．

【Jiri-Harada アルゴリズム】

Step1（Bottom-Up 集約）：まず各ノード m_i に対して，sense score vector の各要素 $M_i(u)$ の初期値を，m_i（語意 $u_1, .., u, \ldots$ をとる）とそれが係っている係り先語 h（語意 $j_1, .., j, \ldots$ をとる）の語意の組合せ (u, j) に対する語意確率のうち，語意 u を持つものの和とする．ただし，根については係り先がないので，逆にすべての係り元語との語意確率から，根の各語意についてその値を持つ語意確率の和を求めることにする．これは，語 m_i の語意次元に沿った確率ベクトル

を表している．さらに，語 m_i の sense score matrix の各要素 $Q_i(k,j,u)$ に，m_i の語意 u と h の語意 j とその間の格 k に対する語意–格総合評価値を割り当てる．次に最下層より 1 つ上以上の各ノード（例，h）において，その sense score vector $M_h(j)$ を，その直下のノード群 $\{m_i\}$ の sense score matrix を用いて式 (1) のように更新する．

$$M_h(j) = \frac{L_j}{L} M_h(j) \quad (1)$$

$$L_j = \sum_k max_u(Q_i(k,j,u)) \quad (2)$$

$$L = \sum_j L_j \quad (3)$$

ここで (2) の $max_u(Q_i(k,j,u))$ は，h の語意 j を一定にして m_i の語意 u を変化させたときの最大値，$\sum_k$ は上記の最大値を格 k を変化させたときの和である．式 (3) の $\sum_j$ は h の語意 j を変化させたときの和である．

Step2（Top-Down 決定）：まず，最上位のノード r の語意をその sense score vector M_r の要素の最大値を与えるインデックス l とする．次に，この最上位のノード h から始めて，その係り元語 m_i の語意と h との間の格を，m_i の sense score matrix $Q_i(k,j,u)$ を用いて，h の語意 j を固定して m_i の語意 u と h との間の格 k を変化させたとき，$Q_i(k,j,u)$ の最大値を与える語意 u と格 k とする．

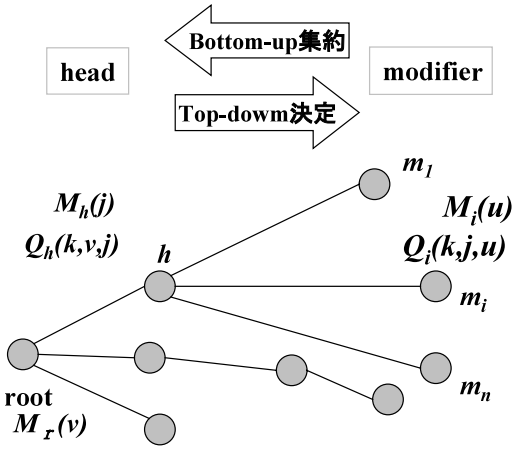

図 4 Jiri-Harada アルゴリズム
Fig. 4 Jiri-Harada Algorism.

Jiri アルゴリズムとの差は，本アルゴリズムでは relation matrix を必要としないこと，また sense score matrix を 2 次元ではなく格 k の次元を加えた 3 次元行列として，その値を語意–格総合評価値で直接的に与えていることである．これは，SAGE では図 4 に示すように 2 語の語意と語間の格のすべての組合せに対する出現確率が，先に述べたように，EDR から求まるからである．

3.3 精度評価

図 3 の SAGE2000 に示すように Jiri-Harada アルゴリズムを採用することにより文節数の線形オーダの時間での解析が可能となった．ただしこのアルゴリズムでは，語意ベクトル（sense score vector）の値を直下の係り元語との関係のみによって決定しているので，遠くのノードにおける語意や格まですべて変化させた中での最適解を求める SAGE99 より精度が落ちる．EDR コーパス辞書に記述されている 100 個の文章を SAGE99 と SAGE2000 で解析して比較した．この際，両者の解析結果が不一致の場合，その内容を詳細に検討した結果 SAGE99 の解析結果が不正解で SAGE2000 の解析結果が正解の場合と，どちらも正解といえる場合は評価の対象外とした．この評価によると一致度は，**表 1** に示すように語意一致度が 97.8%で，格一致度が 100%であり，精度上ほとんど問題ないと判断できる．この結果，その高速性から SAGE2000 では Jiri-Harada アルゴリズムを採用した．なおこの実験結果は，各語の語意はその直接の係り元の語意との相関でローカルにほぼ決定されることを示している．

4. 精度向上

従来の SAGE の解析結果を分析したところ，用言間の格，複合語の構成語の語意などに誤りが多いことが分かったので，以下のような改良を行った．

4.1 複文の格と語意の決定

複文には中心となる用言が複数存在し，さらに用言どうしが係り受け関係を構成している．すなわち，基

表 1 SAGE99 と SAGE2000 の精度差
Table 1 Accuracy comparison between SAGE99 and SAGE2000.

	U：評価データ	A：相違した個所	P(相違) = A/U	1-P(一致度)
フレームの違い	1399	0	0%	100%
語意の違い	1399	10	2.2%	97.8%
格関係の数違い	2330	0	0%	100%
格関係の違い	2330	0	0%	100%
格関係の宛先の違い	2330	0	0%	100%

本的には複文の解析とは用言間の関係の解析であると考えてよい（ただし，連体節の場合のみ用言と名詞の関係になる）．また，中心語が用言である文節どうしであっても，複文になる場合とならない場合がある．ところが EDR の概念記述辞書や共起辞書には用言間の語意–格組は登録されていない．したがって，2 文節の中心語が用言である場合には，辞書に依存しない方法で語意–格組候補を決定しなければならない．そこで，複文を文法的に『基礎日本語文法—改訂版』[6] に従って，**表 2** のように分類した．当該文が複文であるかどうか，また複文であればどの複文にあてはまるかは，係り元の助詞の種類によって判断できる．しかし，(A) 補足節を構成する格助詞・提題助詞・取り立て助詞と，(B) 副詞節・並列節を構成する接続助詞（節と節を接続）に，表層的に見て同一の助詞（が，となど）が存在するという問題がある．そこで，(A) の助詞が複文を形成する場合には，必ず助詞を持つ文節に形式名詞が出現することから，助詞を持つ文節内に形式名詞（の，ことなど）が現れた場合は (A)，現れなかった場合は (B) であると判断することにした．この判断の後，**表 3** に述べるような方法で，これら用言間の語意–格組に対する語意–格総合評価値を決定する．

なお，ここで，表 3 の評価値 0 補完法とは，すべての語意の対の間に，可能な格のもとで語意–格総合評価値が 0 の選択肢があるものと見なして解釈木の構築を行う方法である．したがって語意については当該 2 文節がそれぞれ係り受け関係にある他の文節の中心語との語意–格総合評価値に依存して決まる．また副詞節の格の決定については，具体的には**図 5** に示す接続詞/接続助詞に基づいて用言間の EDR 格を決定する．またオブジェクト指向における動的分析などの応用研究から[1),13)]，EDR 格中の condition 格と cooccurrence 格を細分化する必要が生じた．このため図 5 に網かけで示した 7 つの格を独自に定義した．これらの格の決定では，接続詞/接続助詞のみではあいまい性が残るので，図 5 下段の*1〜*3 で示すように係り受け関係にある他の用言の語意や助動詞の活用形などを考慮して決定するようにした．具体的には，たとえば，「理由」と「原因」については，前の用言が「行為」であるか「現象」であるかによって区別した．

表 2 複文の文法的な分類（『基礎日本語文法—改訂版』より）
Table 2 Classification of complex sentences.

複文の種類	例
補足節	飛行機が飛んでいくのが見えた．
副詞節	人はエレベータに乗ると行き先階のボタンを押す．
連体節	エレベータは搭乗者の指定した階に止まる．
並列節	音楽を聴いたり，映画を見たりする．

表 3 複文の格と語意
Table 3 Case and meaning of complex sentences.

複文の種類	格	語意
補足節	助詞と係り先中心語をキーにして 共起辞書を検索し格確率を求める．	評価値 0 補完法を用いる．
副詞節・並列節	接続詞/接続助詞や用言の 語意などに基づいて格を決定する．	評価値 0 補完法を用いる．
連体節	節が名詞を修飾しているので， 修飾関係を表す modifier 格に統一する．	評価値 0 補完法を用いる．

edr格	細分格	詳細な意味	接続詞／接続助詞				
condition	reason(*1)	理由を表す関係(*1)	ので	ため(に)	だけに	せいで	結果
	cause(*1)	原因を表す関係(*1)	ので	ため(に)	だけに	せいで	結果
	logical(*2)	条件・譲歩を表す関係(*2)	なら	たら	れば	と	
	timing (*2)						
	reverse	逆接を表す関係	けれども	のに	ながら(*3)	つつも	にもかかわらず
purpose		目的	ために(は)	ように	のに(は)	べく	に
cooccurrence	when	時を表す関係	とき(に)	おり(に)	さい(に)	たび(に)	
	durring	事象・事実の同時関係	ながら(*3)	つつ			
sequece		事象・事実の時間的前後関係	て				

(*1) 理由： 前の用言の上位概念が “行為(30f83e)” である場合
原因： 前の用言の上位概念が “現象(30f7e5)” である場合
(*2) logical-condition： 前の用言の上位概念が “状態(3aa963)” である場合
timing-condition： 前の用言の上位概念が “変化(3f9856)” である場合
(*3) 逆接関係の“ながら”： 前の用言の上位概念が “状態(3aa963)” である場合
同時関係の“ながら”： 前の用言の上位概念が “変化(3f9856)” である場合

図 5 EDR 格の細分化と副詞節・並列節の深層格の決定
Fig. 5 Determination of case in adverb and parallel clause.

4.2 合成語の中心語以外の語意の決定

「12 階建てのビル」という文は「12 階建て」と「ビル」という 2 つの文節として解析される．このように複数の単語で構成されている文節を複合語と呼び，それぞれの単語を構成語と呼ぶ．特に，中心となる構成語を中心語と呼ぶ．さらに,「12 階建て」は「12」と「階」と「建て」という構成語からなる複合語であり，中心語は「建て」となる．これらに対しては，共起辞書を用い，中心語とそれ以外の構成語の 2 語をキーワードとして出現確率の最も高いものを採用することにした．上の例文では「階」と「建て」をキーワードとして検索し「階」の語意を “1f5a3d” と決定する．ただし数詞については辞書からその語意が唯一に決まる．たとえば「12」の語意は “00010c” と決定する．

5. 解析精度の自動評価

SAGE の解析精度を自動的に評価するシステムを構築し，実際に 100 文に対して評価を行った．我々は評価対象文として EDR 電子化辞書のコーパス辞書に記述されている例文をランダムに選ぶことにした．ここで，選んだ例文の 1 文あたりの平均文字数は 47.2，平均文節数は 9.7，平均係り受け数は 7.8 であった．なお，SAGE が解析に使う辞書にはこのコーパス辞書は含まれないので，これらの辞書のデータが各種の確率計算に影響を与えることはない．コーパス辞書は新聞や雑誌などから抽出した文と，それを専門家が意味解析した結果データを保持している．この意味解析済みデータには，①構成要素情報，②形態素情報，③構文情報，④意味情報がある．このうち，**図 6** に示すように，④意味情報は，形式は異なるが，SAGE が出力する意味フレームと同等の情報を保持している．

本評価システムは，**図 7** のように 2 つのコンポーネントから構成されている．形式変換 corpusYxx-Japanese は解析済みコーパスデータを SAGE の出力データの表現形式に変換する．EvalSAGE は両者の照合を行う．SAGE から出力された意味フレームと corpusYxxJapanese から出力されたコーパスフレームを読み込み，**図 8** の Excel 表の各セルに出力する．この際，フレームごとにその語意と，他のフレームと関係があるならば，その相手先のフレームが同じかどうか，その間の格が同じかどうかを調べ，SAGE の意味フレームとコーパスフレームの結果が一致しているものを “1”，一致していないものを “0” として，語意の正誤，格の正誤，あて先の正誤の各列に出力する．なお 100 文に対する出力終了後に，目視チェックを行い，評価の対象外にするものを，“*” に変更した．具

解析済みコーパス

```
1| [main 16;結婚;0e51a0]
   [agent 14;ペルセウス;"ギリシア神話の登場人物"]
   [agent  9;姫;104f79]
   [sequence 11;救;3ceb79]
   [and  | [main 6;な;3ceae3]
         [object 1;島;0edf59]
         [manner 3;たちまち;109a8b]
         [goal 4;石;0ee74f]|]]
```

SAGE-frame

```
frame(1,'ペルセウス','未知語','JSA','ペルセウス','名詞-サ変接続','none','ffffff',[],1).
frame(2,'姫','ヒメ','JN1','姫','名詞-一般','none','104f79',[],1).
frame(3,'救わ','スクワ','JVE','救う','動詞-自立','未然形','3ceb79',[[object,2]],1).
frame(4,'れ','レ','none','れる','動詞-接尾','連用形','000000',[],1).
frame(5,'救われ','none','JPR','救われ','none','none','000000',[[consist,3],[consist,4]],1).
                              ・・・・・・・・・・・
frame(11,'する','スル','JVE','する','動詞-自立','基本形','000000',[],1).
frame(12,'結婚する','none','JPR','結婚する','none','none','000000',[[consist,10],[consist,11]],1).
```

図 6 解析済みコーパスデータと SAGE の出力データの表現形式
Fig. 6 Comparison between the reformed corpus data and SAGE output.

体的には，語意において対象外としたものには，次の 5 つがある．①当該フレームが固有名詞を表す場合，②コーパス辞書に語意が存在せず日本語で直接記述されている場合，③記号の場合，④接頭語・接尾語の場合，⑤形態素要素が異なってい場合．格において対象外としたものは次の 5 つがある．① SAGE によって複合語と判定された語の各構成語間の格で “modifier” 格となっている場合，②指示代名詞の場合，③ゼロ代名詞を指している場合，④修飾側と被修飾側の間の格がコーパスで “which” 格と判定されている場合，⑤宛先の形態素要素が異なっている場合．また特殊な評価としては，以下の 2 つがある．①格の宛先が複合語の構成語をばらばらに指しているものは，そのまとまりで 1 つの誤りとしてみる．②連体修飾で，SAGE では係り先から係り元へ “modifier” 格で係っているが，コーパスでは係り先の意味的役割を考え，係り元から係り先へ “object” 格や “agent” 格で係っている場合，それら相互で 1 つの誤りとしてみる．

このように作成した Excel 表の結果が図 8 である．たとえば,「石」と「なり」の語意が異なっていることや，コーパスでは「救わ」と「姫」の間に object 格がないことが分かる．この例では前者はコーパス辞書が正しいが，後者ではむしろ SAGE の解析結果のほうが正しいことが分かる．

コーパス辞書の例文 100 文において，SAGE2000 が生成した意味フレームを，この精度評価システムを用いて評価した結果，語意正解率は 81.1%，格正解率は 60.7%，格の宛先正解率は 73.3%であった．なお，これらの値はコーパス辞書が正しいとした場合の正解率であるが，先にも指摘したようにコーパス辞書が誤っていることもあり，実際の正解率はもう少し高くなると思われる．また，この評価実験においては，茶筅と茶掛（精度 90%といわれている）の出力である係り受

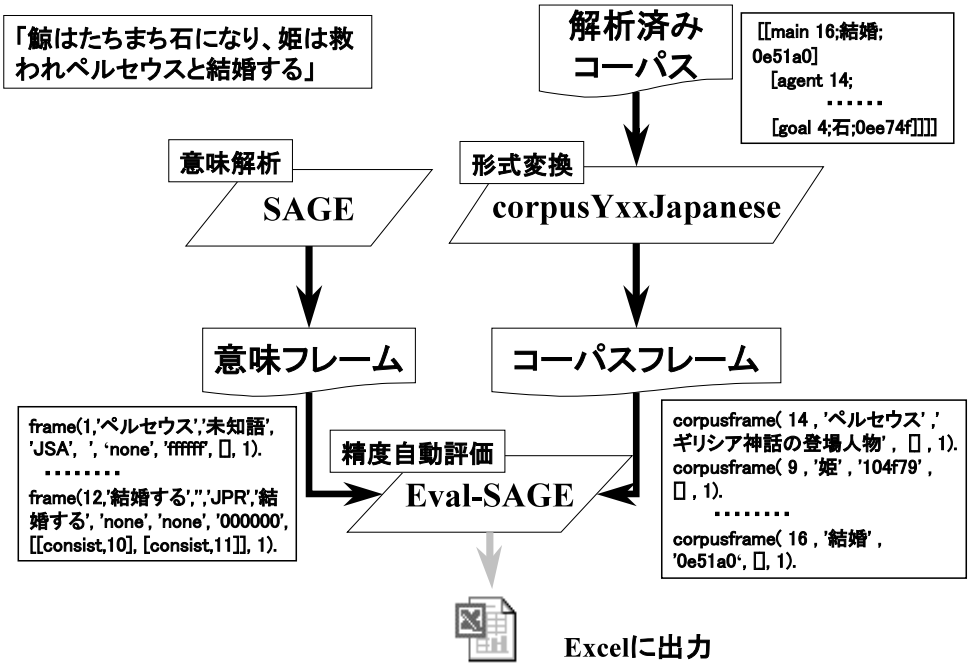

図 7 精度評価システムの流れ
Fig. 7 Accuracy estimation flow.

Corpusframe番号	Sagefame番号	語意	CorpusID	SageID	IDの正誤(%)	Corpus格	Corpus格あて先	Sage格	Sage格あて先	格の正誤(%)	あて先の正誤(%)	文番号
					71.4%					75.0%	83.3%	
鯨はたちまち石になり、姫は救われペルセウスと結婚する。												
14	1	ペルセウス	ギリシア神話の登場人物	ffffff	*	[	]	[	]	1	1	1
9	2	姫	104f79	104f79	1	[	]	[	]	1	1	1
11	3	救わ	3ceb79	3ceb79	1	[	]	[object	姫]	0	0	1
1	9	鯨	0edf59	0edf59	1	[	]	[	]	1	1	1
3	7	たちまち	109a8b	109a8b	1	[	]	[	]	1	1	1
4	6	石	0ee74f	0e434c	0	[	]	[	]	1	1	1
6	8	なり	3ceae3	101d96	0	[manner	たちまち]	[manner	たちまち]	1	1	1
6	8	なり	3ceae3	101d96	*	[goal	石]	[goal	石]	1	1	1
6	8	なり	3ceae3	101d96	*	[object	鯨]	[	]	0	0	1
16	10	結婚	0e51a0	0e51a0	1	[and	なり]	[sequence	なり]	0	1	1
16	10	結婚	0e51a0	0e51a0	*	[sequence	救わ]	[sequence	救わ]	1	1	1
16	10	結婚	0e51a0	0e51a0	*	[agent	ペルセウス]	[agent	ペルセウス]	1	1	1

図 8 誤りを分類した結果
Fig. 8 Classification of analysis errors.

け木をそのまま SAGE への入力とした．したがって，係り受け解析の誤り（10%程度）は特に格とその宛先の誤りを誘発し，その分それらの精度を下げていると考えられる．このような結果から，SAGE2000 は実利用を開始できる精度に至ったといえる．

6. おわりに

本研究により，SAGE は解析速度と解析精度ともに実利用可能なレベルに近づいたといえる．今後は，速度面では辞書検索の速度の向上，精度面ではさらなる誤り分析による改良を行う必要がある．また前章で評価の対象外とした各ケースにおいて，改善を行うべきと考えている．特に複合語については，いろいろな区切りでの複合語を辞書引きすべきと思われる．さらに，接頭語・接尾語については，今後次のように改善していこうと考えている．接尾語は単独で単語辞書を引き語意を決定していく．ただし，単位のように語意が唯一に決定できるものはルールを作成して決定することが可能である．連体修飾については，現在は “modifier” 格に統一しているが，今後は連体修飾内での各語の意味的役割を表す深層格を解析していこうと考えている．

謝辞 本研究を進めるにあたり,『茶筌』と『茶掛』を提供してくださった奈良先端科学技術大学院大学の松本裕治教授に深く感謝いたします．なお，本研究の

一部は，文部科学省科学研究費基盤研究 C『日本語文章の常識を用いた意味理解・文脈理解システムの開発研究』の補助金を用いて行われました．

参 考 文 献

1) 原田　実，野村佳秀，山本幸二，大野雅志，田村浩樹，高橋史郎：自然語要求仕様からオブジェクト指向設計図を自動生成するシステム CAMEO，情報処理学会論文誌，Vol.38, No.10, pp.2031–2039 (1997).
2) 原田　実，水野高宏：EDR を用いた日本語意味解析システム SAGE，人工知能学会論文誌，Vol.16, No.1, pp.85–93 (2001).
3) 平川秀樹，天野真家：日本語解析における最適解探索，情報処理学会研究報告「自然言語処理」，No.74, pp.9–16 (1989).
4) Jiri S. and Nagao, M.: General Word Sense Disambiguation Method Based on a Full Sentential Context, *Journal of Natural Language Processing*, Vol.5, No.2, pp.47–74 (1998).
5) 黒橋禎夫，長尾　眞：格フレーム選択における意味マーカと例文の有効性について，情報処理学会研究報告「自然言語処理」，Vol.91, pp.79–86 (1992).
6) 益岡隆志，田窪行則：基礎日本語文法—改訂版，くろしお出版 (1992).
7) 松本裕治，北内　啓，山下達雄，平野善隆，今一修，今村友明：日本語形態素解析システム『茶筌』version 2.0 使用説明書，奈良先端科学技術大学院大学松本研究室 (1999).
8) 水野高宏，原田　実：日本語意味解析システム SAGE の高速化と精度向上，人工知能学会第 14 回全国大会論文集，pp.149–152 (2000).
9) 尾見孝一郎，原田　実，岩田隆志，水野高宏：日本語文章からの意味フレーム自動生成システム SAGE（Semantic frame Automatic GEnerator）の開発研究，人工知能学会第 13 回全国大会論文集，pp.213–216 (1999).
10) 東　優，峰恒　憲，雨宮真人：既存の概念辞書を用いた動詞語義による文の分類，電子情報通信学会（言語理解とコミュニケーション研究会），Vol.96, No.294, pp.39–44 (1996).
11) 内山将夫，板橋秀一：格フレームを選択する三手法の比較，言語処理学会第 2 回年次大会発表論文集，pp.377–380 (1996).
12) 矢後友和，原田　実：日本語要求仕様文章からオブジェクト指向による動的モデルを生成するシステム CAMEO/D の開発，情報処理学会第 62 回全国大会論文集，pp.95–98 (2001).
13) 矢後友和，原田　実：日本語要求仕様文章からシーケンス図を自動生成するシステム CAMEO/D の開発と販売管理システム問題への適用，情報処理学会オブジェクト指向 2001 シンポジウム論文集，pp.9–16 (2001).

(平成 13 年 11 月 5 日受付)
(平成 14 年 7 月 2 日採録)

原田　　実（正会員）

1951 年生．1975 年東京大学理学部物理学科卒業．1980 年同大学理学系大学院博士課程修了．理学博士．（財）電力中央研究所研究員を経て，1989 年より青山学院大学理工学部経営工学科助教授，2000 年より同情報テクノロジー学科教授，2002 年 University of California at San Diego Visiting Scholor，現在に至る．1986 年電力中央研究所経済研究所所長賞．1992 年人工知能学会全国大会優秀論文賞．1996〜1998 年 EAGL 推進事業機構「ソフトウェア開発の統合的自動化」プロジェクトリーダー．主たる研究は，ソフトウェア分析・設計の自動化，自然語意味理解，ルールベースの自動更新等．編著書「自動プログラミングハンドブック」等．電子情報通信学会，人工知能学会，ソフトウェア科学会，IEEE，ACM，AAAI 各会員．

田淵　和幸

1999 年青山学院大学理工学部経営工学科卒業．2001 年同大学大学院修士課程修了．現在，株式会社 NTT データ．

大野　博之

2000 年青山学院大学理工学部経営工学科卒業．2002 年同大学大学院修士課程修了．現在，日本電気株式会社．