

Support Vector Machine に基づく f-structure の選択

吉村 宏樹[†] 増市 博[†] 大熊 智子[†] 杉原 大悟[†]

[†]富士ゼロックス株式会社 研究本部

〒259-0157 神奈川県足柄上郡中井町境 430

E-mail: [†]{Hiroki.Yoshimura, hiroshi.masuichi, ohkuma.tomoko, daigo.sugihara}@fujixerox.co.jp

あらまし 現在我々が研究開発を行っている日本語 LFG システムが出力する解析結果の曖昧性を解消する実験をおこなった。今回の実験では、解析結果を一つに絞り込む方法として Support Vector Machine (SVM) を用いた。今回の実験は小規模ではあるが、EDR コーパス 200 文を対象に、再現率 80.3%、適合率 82.5%の結果が得られ、本結果は、従来の日本語 LFG システムの係り受け精度の平均値と比較して、良い結果であった。また代表的な日本語構文解析システムである KNP および Cabocha の解析精度とほぼ同等の値が得られていることがわかった。

キーワード Lexical Functional Grammar, 曖昧性解消, 機械学習, Support Vector Machine

Disambiguation of f-structures based on Support Vector Machine

Hiroki YOSHIMURA[†] Hiroshi MASUICHI[†] Tomoko OHKUMA[†] Daigo SUGIHARA[†]

[†]Corporate Research Group, Fuji Xerox Co., Ltd.

430 Sakai, Nakai-machi, Asagarakami-gun, Kanagawa, 259-0157 Japan

E-mail: [†]{Hiroki.Yoshimura, hiroshi.masuichi, Ohkuma.Tomoko, daigo.sugihara}@fujixerox.co.jp

Abstract We report experimental results of disambiguation on analysis results that a Japanese parsing system based on Lexical Functional Grammar (LFG) outputs. In this experiment, we used Support Vector Machine to evaluate disambiguation performances. The recall/precision of dependency accuracy in this experiment reached 80.3%/82.5% applied to EDR corpus (200 sentences). We were able to obtain the better result, as compared the average accuracy of all the parsing results our Japanese LFG parser outputs. Furthermore, the result is almost the same as the analysis accuracy of KNP and Cabocha.

Keyword Lexical Functional Grammar, Disambiguation, Machine Learning, Support Vector Machine

1. はじめに

言語処理において、宣言的な文法規則に基づく自然言語解析の研究・開発は不可欠のタスクである。我々は、Lexical Functional Grammar (LFG)[1]に基づいた実用的な日本語解析システムを開発している[2]。本システムの第一の特徴は、精緻な日本語文法規則に基づく深い解析を行う点である。第二の特徴は、実文を対象とした評価が可能な高い解析力カバー率を達成している点、すなわち解析対象が口語的・非文法的文であっても解析可能な高いカバー率を持つ点である。本システムの実装により、LFG に基づく日本語解析システムとしては初めて、文法機能の情報を含めた解析精度の評価実験を行うことが可能となった。さらに第三の特徴として、LFG の解析結果が持つ言語普遍性の特徴を活かすため、他言語の LFG 文法と高い整合性・無矛盾性を保っている点を挙げる事ができる。

この日本語 LFG システムは、一つの入力文に対して

文法的に正しいと認められる解析結果を全て出力する方針で構築されている。しかしながら、日本語 LFG システムをアプリケーションに適用することを考えた場合、後工程として、解析結果の曖昧性を完全に解消するためのシステムが必須である。

そこで本稿では、日本語 LFG システムから出力される解析結果に対して統計的手法を適用することにより曖昧性を解消する実験を行う。本実験においては、テキスト分類などの言語処理タスクで最も高い精度が報告されている機械学習手法である Support Vector Machine(SVM)[3]を適用する。

本稿の構成は以下の通りである。2 節では、LFG および、上記第三の特徴に関係する取り組みとして、我々が属している Parallel Grammar Project での活動を概観する。3 節において SVM について説明した後、4 節では SVM を用いた曖昧性解消実験について述べる。4.1 から 4.3 において、本実験の手順を説明し、4.4 では、

曖昧性解消実験の評価結果を述べる。5 節に今後の課題を記す。

2. Lexical Functional Grammar

LFG は自然言語文の解析を行うための文法理論であり、[4]によってその概念が提唱された後、[1]によって現在の形の定式化が完成した。LFG に基づく解析では、解析結果として、c(onstituent)-structure と f(unctional)-structure と呼ばれる 2 種の構造を出力する。c-structure は自然言語文の構造を、文の形態素を上位のフレーズへと纏め上げることによって木構造として表現するものである。一方、f-structure は、Subject や Object といった文法機能(grammatical function)の概念に基づき、文の格構造、時制、様相、語法等の意味情報を属性-属性値のマトリックス構造で表現するものである。言語が異なれば同じ意味内容を表現する文であってもその句構造すなわち c-structure は大きく異なる一方で、文法機能に基づく構造である f-structure の違いは多くの場合極めて小さいことが知られている[5][6]。

(1) ジョンは本をそのテーブルに置いた。

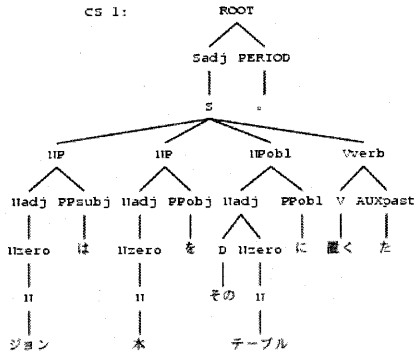


図 1: 文(1)に対応する c-structure

「ジョンは本をそのテーブルに置いた。」

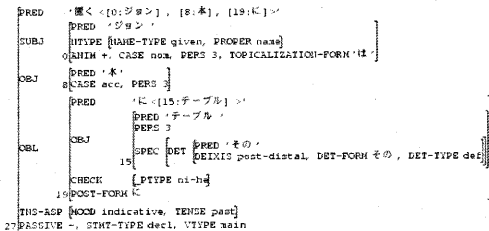


図 2: 文(1)に対応する f-structure

図 1 に文(1)に対応する c-structure を示し、図 2 に、文(1)に対応する f-structure を示す。

2.1. Parallel Grammar Project

我々は、Parallel Grammar (ParGram)と呼ばれるプロジェクト活動[7]の中で日本語 LFG 文法の記述を行っている。ParGram は、LFG を共通の理論基盤として英語、ドイツ語、フランス語、ノルウェー語、ウルドゥー語、デンマーク語、韓国語そして日本語の各言語を解析するシステムの実現を目標としている。ParGram では、半年に一度全言語の文法記述担当者が集まってミーティングを開催し、複数言語間で f-structure の整合性を可能な限り高める、あるいは、矛盾を回避する機会を継続して持つことにしている。すなわち、f-structure の構成から属性・属性値の用法やネーミングコンベンションに至るまでの詳細を議論し、ParGram の標準仕様を決定する作業を行っている。日本語 LFG 文法における、肯定文、否定文、疑問文、受動文、並置表現等の基本的な構文に対して出力される f-structure は、この ParGram 標準仕様[5]に準じている。ただし、この標準仕様は、f-structure の基本的な構成である属性および属性値については全言語でほぼ合意がなされているが、詳細な文法事項に関しては現在も継続的に議論が続けられている。

2.2. Xerox Linguistic Environment

ParGram では共通のプラットフォームとして Xerox Linguistic Environment(XLE) と呼ばれる LFG の処理系を開発している。XLE は LFG 理論の仕様をほぼ完全に実装した parser であると同時に、LFG の文法記述を行う際のデバック環境であり、かつ、f-structure から自然言語文を生成する generator でもある[8]。XLE が持つ最も重要な特徴はその解析速度である。LFG では、c-structure を構成するための句構造規則(文脈自由文法規則)と、f-structure を構成するために句構造規則(の右辺の各構成素)に付与する機能的注釈(functional annotation; functional schemata)を同時に記述する。XLE は、句構造規則に基づく解析処理を実行した後、処理が必要な機能的注釈を動的に決定する。すなわち、計算量の大きい機能的注釈の処理を選択的に行うことによって解析速度の向上を実現している[9]。また、解析結果の排他性・独立性を考慮した単一化手法[10]も高い解析効率に寄与している。XLE を採用することによって、大規模な LFG 文法に基づく解析を実時間で行うことが可能となった。

図 3 は、日本語 LFG システムの構成図である。XLE による入力形態素列の解析には、入力された文の形態素解析結果[11]を受けて、予め用意されたテンプレートとルールから動的に各形態素の語彙規則を生成する入力文語彙規則に加えて、予め用意した基本語彙規則、動詞語彙規則、形容詞語彙規則、形容動詞語彙規則、

および、文法規則を使用する。動詞語彙規則、形容詞語彙規則、形容動詞語彙規則は、IPAL 辞書[12]の結合価値情報をベースに拡張を行い、LFGの規則として必要な記述を追加したものである。現在のところ、動詞語彙規則は2,366の異なり語彙に対して10,387の語彙規則(41,115の機能的注釈)を含んでいる。形容詞語彙規則および形容動詞語彙規則は、それぞれ302および67の異なり語彙に対して810,137の語彙規則(1,891,306の機能的注釈)を含んでいる。基本語彙規則は、助詞、助動詞、代名詞、接続詞等の基本語彙に加え、接続助詞的な働きをする「間」「時」といった名詞等、統語上重要と考えられる語の語彙規則を独自に記述したものである。現状では、675の異なり語彙に対して1,252の語彙規則(1,913の機能的注釈)を含んでいる。

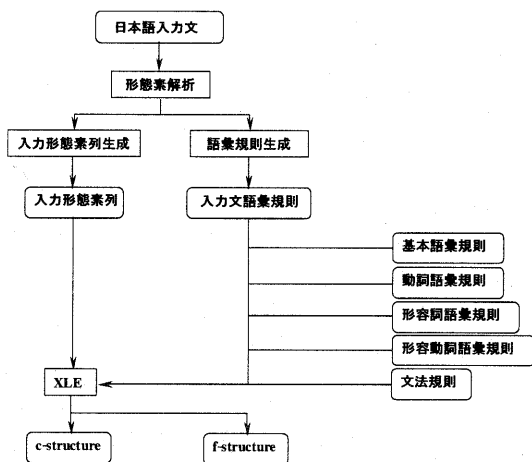


図 3: 日本語 LFG システムの構成図

2.3. 日本語 LFG システムの解析カバー率

これまででも、LFG や Head-driven Phrase Structure Grammar (HPSG)等の単一化文法に基づいた日本語の解析に関する研究は数多く行われている。しかし、比較的最近まで、f-structure のような素性構造の単一化は大きな計算量を要する作業であるため、大規模な文

法を用いた解析の実行は困難であるとの認識があった[13]。さらに、単一化文法の枠組みにおいて、日本語の文法事項を網羅的に含み高いカバー率を実現する文法規則を記述することは容易ではないと指摘されている[14]。しかしながら、前者の解析効率の問題に関しては、[9]、[15]等で提案されている解析アルゴリズムの進展によって一定の解決策が得られつつある。また、この前者の問題の解決は、後者の問題とも大きく関係している。単一化に基づく解析を実行するシステムは制約充足問題を解決するシステムである。制約数すなわち文法規則の数が多くなるにしたがって計算量の爆発が生じる。前者の問題が解決されていないこれまでの状況においては、システムの制約上、大規模な文法を記述し試すことが困難であったため、高い解析カバー率を持つ文法の構築に焦点を当てた研究には至らなかった。これに対し、我々は XLE が持つ高い水準の解析速度の特徴を活かし、大規模な日本語 LFG 文法の開発を行っている。文法規則の規模は、積和標準形に変換した場合の項の総計が2,353であり、1,223の機能的注釈を含んでいる。

また高いカバー率を達成するためには、例外的な表現を含む文や非文法的な文を解析できなければならない。しかし、例外的な記述を含む文法規則を取り入れた場合、正しい解析結果と同時に不適切な解析結果も多数出力される可能性がある。そこで、この問題を回避し日本語 LFG システムの頑健性を高めるために、Optional Theory(OT)マーク[16]と部分解析機能を追加している。OT マークは、各機能的注釈に対して付与される優先順位規則である。OT マークを付与する目的は、言語学的な根拠を基に解析結果の曖昧性を減少させ解析結果数の爆発的な増大を防ぐと同時に、例外的な文法事項を見通しよく記述することである。また部分解析機能は、非文法的表現を含む文のカバー率を向上させるために導入されている。通常文法(正規文法)では解析結果が得られない場合に、正規文法とは異なる別の文法体系(Fragment 文法)を用意して、部分的な解析結果のリストを得ることにしている。これらの処理を施すことによって、例外的な表現を含む文や

表 1: 日本語 LFG システムのカバー率

	Num. of sentences	Ave. num. of words in a sentence	Coverage (%)			Ave. time. to analyze a sentence without Skimming* (sec)	Ave. num. of STANDARD results
			Total	STANDARD	FRAGMENT		
(A)	10,000	22.6	97.3	91.6	5.7	1.9	14.6
(B)	460	21.3	98.7	94.4	4.3	1.1	10.1
(C)	9,637	16.3	97.9	87.0	10.9	4.8	46.2

*2.8GHz CPU
2GB memory

非文法的な文に対しても解析結果を出力することが可能となり、カバー率を上げることができる。

表 1 に日本語 LFG システムのカバー率を示す[20]。以下の 3 種類のテキストを対象に解析実験を行い、カバー率を測定している。

(A)EDR コーパス[17] 10,000 文

(B)コピー機マニュアル文[18] 460 文

(C)お客様相談センター(Voice of Customer)文 9,637 文

(A)の 10,000 文と(B)の 460 文は文法に則った文が多い。一方(C)は、富士ゼロックスのお客様相談センターへの電話による問い合わせ内容を比較的忠実に人手でテキスト化したものである。したがって、口語的・非文法的な文が多く含まれている。解析実験の結果得られたカバー率を表 1 に示す。カバー率は、c-structure だけの出力ではなく、f-structure を伴う解析結果が得られた場合を対象としたものである。なお、表 1 中の STANDARD は、正規の文法で解析できたカバー率を示す。同様に FRAGMENT は、Fragment 文法のみ解析可能であった場合のカバー率を示す。また日本語 LFG システムでは、正規文法および Fragment 文法で解析できなかった場合は、Skimming モードに移る。これは、計算機資源の問題で解析できない文に対して、近似解析結果を出力するモードである。Fragment 文法および Skimming モードに関する詳細は、[19]を参照のこと。

3. Support Vector Machine

Support Vector Machine (SVM)は、Vapnik によって提案された機械学習手法である[3]。SVM は、学習データを正例と負例にわけ、正負例間の距離が最大になるような超平面を求める学習器であり、未学習データに対して識別性能(汎化性能)が高い点が特徴である。これまででも SVM は、日本語単語の多義性解消実験で他の手法と比較して良好な結果を得ており[20]、その他にも構文解析[21]などに幅広く応用されている。

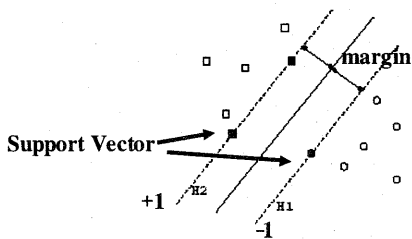


図 4: Support Vector Machine の概念図

図 4 に SVM の概念図を示す。SVM は、正例と負例を分類する分離超平面を求める。分離超平面に対して、正例と負例の 2 種類の学習データとの間の距離をマージンと呼ぶ。また分離超平面にもっとも近い学習デー

タをサポートベクトルと呼ぶ。SVM では、マージンが最大になるような分離超平面を求め、これによって過学習を防ぎ、最も汎化性能の高い学習器を構成する。つまりマージンを最大化する方法を定式化すると、2 次計画問題に帰着することができる。そのため、最適解が唯一に定まり、局所的最適解に陥る心配がない。

4. SVM を用いた曖昧性解消実験

本章では、これまで述べてきた日本語 LFG システムと SVM を組み合わせて、日本語 LFG システムから出力される解析結果の曖昧性を解消する実験について詳述する。

本実験の手順は、日本語 LFG システムの解析結果から素性を抽出するための実験の前処理、素性抽出、機械学習と評価の順に分けられる。なお、本実験では EDR コーパス[16]の中で正規の文法によって解析結果が得られた文の中から 200 文をランダムに抽出して実験を行った。以下、実験手順を詳述する。

4.1. 実験の前処理

本実験で用いた素性は、日本語 LFG システムの解析結果である f-structure が持つ属性情報から抽出している。解析精度を測定するための準備として、日本語 LFG システムが出力する解析結果である f-structure を木構造へ変換した[19]。これは、基本的に f-structure 中の PRED(icate)の値をノードとし、リンクラベルには文法機能の parameter を付与することで f-structure から文節間の依存構造を表す木構造を生成する処理を行った。表 2 に、日本語 LFG システムで使用する文法機能の抜粋を示す。文法機能は、Grammatical Functions と Parameters から構成されており、現在、Grammatical Functions は、8 種類用意し、リンクラベルとして利用する parameter は、53 種類用意している。

表 2: 文法機能(抜粋)

Grammatical Functions	Parameters
Subject	SUBJ
Object	OBJ
Oblique	OBL_kara
	OBL_kara-de
	OBL_kara-to
	OBL_ga

図 5 に、前述の文(1)に対応する f-structure から木構造へ変換する過程と出力結果を示す。f-structure ごとに PRED を取り出して、その依存関係を木構造に変換することを行った。f-structure の PRED に格助詞が出力された場合、木構造のノードには自立語を対応させ

る変換処理も行っている。例えば、図5の左側で示した f-structure では、OBL(ique)の PRED に格助詞「に」が出力されているが、図5右側の木構造では、ノードを「テーブル」とし、リンクラベルを OBL_ni と変換している。

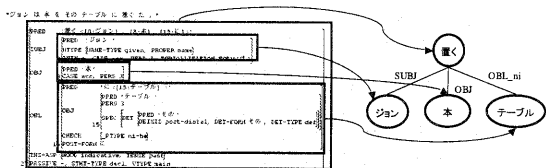


図5: 文(1)に対応する f-structure を木構造へ変換する過程と出力結果

4.2. 素性抽出

まず EDR コーパス 200 文に対して、人手で正解の木構造を作成した。次に、一文ごとに日本語 LFG システムが出力するすべての f-structure(木構造)の精度(F 値)を算出した。

SVM では、正例と負例の素性データが必要であるため、木構造の F 値を参照して、学習データ用の正例と負例を選別した。抽出方法は、正例と負例の数のバランスを考慮して、F 値 87.5% 以上を正例とし、F 値 60% 未満を負例とした。そして抽出された正例と負例の木構造から、以下の 2 種類の素性を取り出した。素性は、両者の合計で約 1500 種類存在する。

- A) 木構造に含まれるのリンクラベルの uni-gram とその個数
- B) 木構造中で、係り先が同じであるリンクラベルの bi-gram とその個数

例えば、文(1)の場合、木構造のノードとして、「ジョン」、「本」、「テーブル」が抽出され、リンクラベルとして、「SUBJ(ect)」、「OBJ(ect)」、「OBL_ni」が付与されていることがわかる。また、「ジョン」、「本」、「テーブル」の係り先は、すべて「置く」(“PRED”)であることが木構造からわかる。A)の素性として、リンクラベルである“SUBJ”、“OBJ”、“OBL_ni”とそれらの個数はすべて 1 である。また B)の素性として、“SUBJ”、“OBJ”、“OBL_ni”の bi-gram は、“SUBJ-OBJ”、“SUBJ-OBL_ni”、“OBJ-OBL_ni”となり、個数はすべて 1 である。

上記の手順を EDR コーパス 200 文に対して行うことで、正例と負例の学習データセットと評価データセットを構成できる。

4.3. SVM による機械学習と評価

EDR コーパス 200 文のうち、150 文を学習データセットとし、残りの 50 文を評価データセットとし、これ

を一回の実験として合計 4 回、SVM による機械学習ならびに評価を以下の手順で行った。

まず 150 文の学習データセットを SVM によって学習させ、この学習結果に基づいて、評価データセット 50 文に対して、分類を行った。次に、本実験では、SVM のサポートベクタ(正例である+1)に近い値を持つ木構造を取り出した。これによって各文の解析結果を一つに絞り込んだ。最後に、一つに絞り込まれた解析結果をあらかじめ人手によって作成された正解の解析結果と比較し、再現率と適合率を求めた。

この一連の実験プロセスを 4 セット行い、EDR コーパス 200 文の再現率と適合率の平均値を算出した。

4.4. 結果と考察

これまでの手順によって得られた実験結果を表3に示す。EDR コーパス 200 文を対象に、SVM を用いた曖昧性解消のための実験を行った結果、再現率 80.3%、適合率 82.5%となった(表3中の Stochastic の値)。

表3中の Average とは、「1つの文に対して得られる複数の f-structure に対応する各木構造を正解木と比較することにより再現率・適合率を算出して平均をとり、さらに 200 文の平均をとった値」である。Upper Bound とは、「1つの文に対して得られる複数の f-structure に対応する各木構造の中から再現率あるいは適合率の最も高いものを選択し、200 文の平均をとった値」である。上限値は、解析結果の曖昧性が理想的に解消された場合に得ることができる値である。

比較対象として、同じ 200 文を代表的な日本語構文解析システムである KNP[22]と Cabocha[21]を用いて構文解析した場合の評価結果を示す。BASELINE は、正解木中の各ノードが文の並びの中で全て右隣に係るとした場合の値である。

これらの結果から、SVM を用いた曖昧性解消の実験では、日本語 LFG システムの係り受け精度の平均値と比較して、良い結果を得ることができた。また KNP および Cabocha の解析精度とほぼ同等の値が得られていることがわかった。

表3: 実験結果

		Prec. (%)	Rec. (%)
LFG	Upper Bound	89.1	87.7
	Stochastic	82.5	80.3
	Average	81.3	79.5
KNP		77.2	78.9
Cabocha		82.6	82.0
BASELINE		58.6	56.0

5. 今後の課題

今回の実験はEDRコーパス200文を対象に行った小規模なものである。今後は、学習データ数や素性数を増加させることで、実験の質を向上させることが課題である。特に今回の実験で使用した日本語LFGシステムが出力する属性・属性値を活かした素性をさらに増加させることで、精度の向上が期待できる。

また、2章で述べたParallel Grammar Projectにおいても、英語LFG解析システムから出力する解析結果を対象として、Maximum Entropy (ME)を用いた曖昧性解消実験[23]を行っており、今後、日本語においてSVMとMEの比較実験を行っていきたい。

6. おわりに

本稿では、SVMに基づく機械学習によって、日本語LFGシステムが出力する解析結果の曖昧性を解消する実験を行った。小規模な実験ではあるが、EDRコーパス200文を対象に、再現率80.3%、適合率82.5%を得ることができた。この結果は、日本語LFGシステムから出力される係り受け精度の平均値と比較して、良い結果であった。またKNPおよびCabochaの解析精度とほぼ同等の値が得られていることがわかった。今後、学習データを大規模にすることによって、さらなる精度の向上をすすめていきたい。

文 献

- [1] Ronald M. Kaplan and Joan Bresnan. Lexical-Functional Grammar: A formal system for grammatical representation. In *The Mental Representation of Grammatical Relations* (Joan Bresnan, ed.), pp. 173-281. The MIT Press, Cambridge, Massachusetts, 1982.
- [2] 増市博, 大熊智子. Lexical Functional Grammarに基づく実用的な日本語解析システムの構築, 自然言語処理, 10(2): 79-109, 2003.
- [3] Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*, Springer, 1995.
- [4] Joan Bresnan. A realistic transformational grammar. In *Linguistic Theory and Psychological Reality* (Halle, M., Bresnan, J. and Miller, G. A., eds.). The MIT press, Cambridge, MA, pp. 1-59, 1978.
- [5] Miriam Butt, Tracy H. King, María-Eugenia Niño, and Frédérique Segond. A Grammar Writer's Cookbook. Number 95 in CSLI Lecture Notes. CSLI Publications, Stanford, California, 1999.
- [6] Mary Dalrymple. SYNTAX and SEMANTICS Lexical Functional Grammar. Academic Press, CA, 2001.
- [7] Miriam Butt, Helge Dyvik, Tracy H. King, Hiroshi Masuichi, and Christian Rohrer. The Parallel Grammar Project. In Proceedings of the 19th International Conference on Computational Linguistics (COLING'02) Workshop on Grammar Engineering and Evaluation, pp. 1-7, 2002.
- [8] Ronald M. Kaplan and Jürgen Wedekind. LFG generation produces context-free languages. In Proceedings of the 18th International Conference on Computational Linguistics (COLING-2000), Saarbrücken, pp. 425-431, 2000.
- [9] John T. Maxwell III and Ronald M. Kaplan. The interface between phrasal and functional constraints. *Computational Linguistics*, 19(4): 571-589, 1993.
- [10] John T. Maxwell III and Ronald M. Kaplan. A Method for Disjunctive Constraint Satisfaction. In *Current Issues in Parsing Technology* (Tomita, M. ed.). Kluwer, Boston, pp. 173-190, 1991.
- [11] 松本裕治, 北内啓, 山下達雄, 平野善隆, 今一修, 今村友明. (1999). "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.
- [12] 情報処理振興事業協会技術センター. 計算機用日本語基本動詞/形容詞辞書 IPAL(Basic Verbs/Adjectives) -辞書編. 日本情報処理振興協会, 1987.
- [13] 橋田浩一. "言語への情報科学的アプローチ" 言語の科学入門 (松本裕治, 今井邦彦, 田窪行則, 橋田浩一, 郡司隆男), 岩波書店, pp. 79-125, 1997.
- [14] 鳥澤健太郎. 高機能な構文解析器に向けて-HPSGのための実用的な構文解析器-. 情報処理, Vol. 40, No. 4, pp. 380-386, 1999.
- [15] Daniel Flickinger, Stephan Oepen, Hans Uszkoreit, and Tsujii Jun'ichi (eds.). Special Issues on Efficient Processing with HPSG. *Journal of Natural Language Engineering*, 6(1): 1-112, 2000.
- [16] Joan Bresnan. Optimal Syntax. In *Optimal Theory: Phonology, Syntax and Acquisition*, pp. 334-385. Oxford University Press, 2001.
- [17] EDR (Japanese Electronic Dictionary Research Institute, Ltd.). EDR electronic dictionary version 1.5 technical guide, 1996.
- [18] 富士ゼロックス. Document Gate 操作ガイド. 富士ゼロックス(株), 2000.
- [19] Hiroshi Masuichi, Tomoko Ohkuma, Hiroki Yoshimura, and Yasunari Harada. Japanese parser on the basis of the Lexical-Functional Grammar and its Evaluation, In Proceedings of The 17th Pacific Asia Conference on Language, Information and Computation (PACLIC17), Singapore, pp. 298-309, 2003.
- [20] 村田真樹, 内山将夫, 内元清貴, 馬青, 井佐原均. 種々の機械学習手法を用いた多義解消実験, 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, pp. 7-14, 2001
- [21] Taku Kudo and Yuji Matsumoto. Japanese Dependency Analysis using Cascaded Chunking. In Proceedings of Sixth Conference on Natural Language Learning (CoNLL'02), Taipei, Taiwan, pp. 63-69, 2002.
- [22] Sadao Kurohashi and Makoto Nagao. A Syntactic Analysis Method of Long Japanese Sentences Based on the Detection of Conjunctive Structures. *Computational Linguistics*, 20(4): 507-534, 1994.
- [23] Stefan Riezler, Tracy H. King, Ronald M. Kaplan, Richard Crouch, John T. Maxwell, and Mark Johnson. Parsing the Wall Street Journal using a Lexical-Functional Grammar and Discriminative Estimation Techniques. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, pp. 271-278, 2002.