修 士 論 文

SVMと３つ組／４つ組法による
日本語係り受け解析に関する研究

指導教官　鳥澤健太郎 助教授

北陸先端科学技術大学院大学
情報科学研究科情報処理学専攻

石村 健二

2003年2月

要 旨

本研究では, 金山ら [2] が提案した３つ組／４つ組モデルにおいて，係り受け確率を推定している最大エントロピー法 (ME) に変わって，Support Vector Machine(SVM) を用いることにより係り受け確率の精度向上を図ることである．

近年, 日本語の構文解析において, 係り受け解析は自然言語処理の基本技術の一つとして認識されており, 従来から多くの研究が行われている．日本語の構文解析において係り受け解析は，重要な基本技術の一つとして認識されている．一般的に係り受け解析は，２文節間の係りやすさを数値化した係り受け行列を作成し， 動的計画法などを用いて文全体が最適な係り受け関係を求めることである．

従来の係り受け解析では， 複数の係り先文節に対してそれぞれ独立な確率で係り先文節を決定していた．それに対して，本研究のベースとなる３つ組／４つ組モデルでは，複数の係り先文節候補の素性情報を同時に考慮することによって，相対的・合理的に係り先文節を決定することができるモデルである．

このモデルにおける, 係り受け確率の推定には ME 法が使われている．しかしながら, ME 法は人手により学習に用いる素性を適切に選択する必要があり，素性が適切に選択されなかった場合, 過学習を起こすなどの問題が指摘されている．一方，SVM 等の学習サンプルの分類境界の間隔を最大化する戦略に基づく手法が提案されている．この SVM は，高い汎化能力を持っていて，ME 法では扱えなかった数の素性を扱うことができ，過学習を起こしにくいため ME 法では出来なかった学習が可能である．さらに, 係り元文節・係り先文節候補の主辞を学習データ追加することで係り受け確率の精度向上を計る．

本論文では, 統計モデルを使った構文解析器の改良について論じた．しかしながら，３つ組／４つ組モデルの係り受け確率推定に SVM を用いて, 係り受け精度の向上を図った結果, 得られた文節正解率は最大 87.7%であり、金山らが ME 法を使って達成した性能には及ばなかった．

目 次

第 1 章

はじめに

1.1 研究の背景と目的

日本語係り受け解析における３つ組／４つ組モデルに Support Vector Machine(SVM) を導入し，構文解析器の解析精度の向上を図る．近年，文章フォーマリズムが発達し，新聞や雑誌などの文章に対して構文木を出力できる文法が研究されているが，文法は曖昧性が高い．３つ組／４つ組法はこの曖昧性を取り除くために開発された統計的手法で，他の類似した手法に比べ高い精度が報告されている．しかしながら，３つ組／４つ組モデルは，他の類似した手法よりもさらに多くの素性を取り扱うため，ここで用いられている最大エントロピー法による係り受け解析では, 扱う素性を慎重に選択する必要があった．そこで，高次元の素性集合を用いても過学習しにくい SVM を係り受け解析に導入する．さらに，金山らが ME 法を用いて３つ組／４つ組モデルを構築したときには、データの過疎性の問題が生じるため素性に一部しか入れられなかった各文節の主辞の単語を素性にすべていれ，さらなる性能の向上を図った．

日本語の構文解析において係り受け解析は，重要な基本技術の一つとして認識されている．一般的に係り受け解析は，２文節間の係りやすさを数値化した係り受け行列を作成し， 動的計画法などを用いて文全体が最適な係り受け関係を求めることである．従来の係り受け解析では， 複数の係り先文節に対してそれぞれ独立な確率で係り先文節を決定するのに対して，３つ組／４つ組モデルでは，複数の係り先文節を確率の条件部に入れることにより，相対的，合理的に係り先文節を決定することができる．

金山ら [2] は，３つ組／４つ組モデルの係り受け確率推定には, 最大エントロピー (ME)

法を使用している．この ME 法は，人手により学習に用いる素性を適切に選択する必要があり，素性が適切に選択されなかった場合, 過学習を起こすなどの問題が指摘されている．一方，SVM 等の学習サンプルの分類境界の間隔を最大化する戦略に基づく手法が提案されている．この SVM は，高い汎化能力を持っていて，多量の素性を扱うことができ，過学習を起こしにくいため ME 法では出来なかった学習が可能である．さらに，各文節の主辞の単語を素性に加えることでさらなる精度向上を図る.

第 2 章

関連研究

本章では, これまでに提案されてきた日本語構文解析のための統計的アプローチと、本研究で構文解析に用いる日本語文法 SLUNG, 及び, 金山らの提案した 3 つ組／4 つ組モデルの確率モデルの推定に用いられた最大エントロピー法を紹介する.

2.1 統計的アプローチによる構文解析

日本語の係り受け解析のための統計的アプローチとして, 様々なモデルが提案されているが, これらは下記の定義により 「生起確率を計算するモデル」, 「文中の係り受け確率の積をとるモデル」の二種類に分別することができる.

生起確率を計算するモデル　文 s が与えられた時に, ある構文木 T が生起する確率を求める. 数式では, 式 *2.1*$\tilde{}$となる.

$$\underset{T}{argmax}\, P(T|s) \tag{2.1}$$

文中の係り受け確率の積をとるモデル　文節 i と文節 j が係り受け関係にある確率を $P(i \rightarrow j)$ と定義し, 式 *2.2*$\tilde{}$に示すような, 文中にある全ての係り受けの積を最大化する係り受け関数 $dep(i)$ を求める方法である.

$$\underset{dep}{argmax} \prod_{i} P(i \rightarrow dep(i)) \tag{2.2}$$

前者に属するものとして, 確率文脈自由文法を用いたもの（*Mori and Nagao 1998*）や, 確率一般化 LR 法を用いたもの（白井, 乾, 徳永, 田中 *1998*）などがある. これらは, 数学的に妥当な確率を用いることができ, 形態素解析など様々なレベルとの統合が容易であると言う利点がある. しかしながら, 現状では係り受け解析精度は最高でも白井らの *85*〜*86* %に留まっている.

一方, 後者の手法は, 比較的学習が容易なため, 高い解析精度が得られる手法が多数提案されている. 実際に本研究でベースとなっている *3* つ組／ *4* つ組モデルは *88.6* %と、生起確率に基づくアプローチよりも高い精度が報告されている. このように比較的学習が容易なため, このアプローチに基づく, さまざまな手法が提案されている. 以下にいくつかの手法を紹介する. 決定木を用いたモデル *(Haruno et al. 1998)*, 最大エントロピー法を用いたモデル *(Uchimoto el al. 1999)*, 距離確率と語彙確率を用いたモデル *(*藤尾, 松本 *1999)* では, 係り元文節 i の品詞, 語彙や読点の有無など, 係り先文節 j の品詞や語彙, そして, 二文節間の距離, 読点や副助詞「は」の数などを属性として, ある属性を持った二文節が存在する時にそれが係り受け関係にある確率を二文節間の係り受けのしやすさであると定義している. 英語における統計的構文解析では二語間の距離が係り受けを決定する重要な要素となる *(Collins 1996)* のと同様に, 日本語係り受け解析においても二文節間の距離が重要であると言う認識があり, 上記のモデルは, 文節間にある文節数を属性として使用している. しかしながら, これらのモデルでは, 文節 i と j 以外の文節の情報は, 文節間の距離などの属性を除いては反映されていない.

係り元・係り先とそのまわりの文節を考慮するモデル *(*内元など *1998)* では, 係り元文節 i の係り先文節 j への係る確率計算に, 係り元文節 i 以降の全ての文節の情報を用いる. そのため, 二文節間の関係を係り先文節に対して「係る」「係らない」の二値ではなく,「手前の文節に係る」「係る」「この文節よりも遠くに係る」の三値を出力するものとして学習している. そして，i が j に係る確率を i が i,j 間の文節を「越える」確率と i が j より「手前に係る」確率の積で補正する. これにより，ある種の文節情報が取り扱えることになり，解析精度が *(Uchimoto et al. 1999)* より約 *1* %向上したことが報告されている. しかしながら，このモデルでは，文中の個々の係り受け関係が互いに独立であると仮定しなければならない.

本研究で用いる *3* つ組／ *4* つ組モデルでは，*2* つ又は *3* つの係り先文節候補の属性を

同時に考慮できるため，文節情報が扱える．さらに，上述にはない利点も存在する．

2.2 日本語文法SLUNG

本論文で使用する *3* つ組／ *4* つ組モデルでは, 人手で書かれた文法により係り受け文節候補を絞ることが必要である．*SLUNG* とは, *HPSG* の枠組みで記述された日本語文法であり, *8* つのスキーマと, *48* 個の語彙項目テンプレート, *105* 個の語彙項目からなっている．さらに, *EDR* コーパスの文に対して *98.4* %と, 非常に高い被覆率を示している日本語文法である．

この文法自身は曖昧性解消の機構を持っていないため, *SLUNG* を構文解析に適用した場合, 被覆率が高いため文法的に許されいる全ての構文木を出力する．この *SLUNG* により出力された構文木から抽出された各係り受け関係に対して, 本研究では *3* つ組／ *4* つ組モデルを用いることにより, 出力された全ての構文木から最も優先度が高いものを選び出すことができるようになる．

2.3 最大エントロピー法

従来の *3* つ組／ *4* つ組モデルでは統計モデルの推定に, 最大エントロピー法（*ME* 法）を用いている．*ME* 法では, 学習コーパス中の履歴（これまでに得た文脈と出力値）の特徴（素性）を集計し, さまざまな素性に対するパラメーターを出力値の確率分布が最も一様分布に近づくように調整し求める．*ME* 法では, 素性によって文脈を観測するので, 素性の数が多いほど多様な文脈を扱える．（あるモデルで使用されている素性数が k 個のとき, ある文脈中にそれぞれの素性が存在するか否かによって全ての文脈は 2^k 種類に分類できる）さらに, データスパースネスの発生を軽減できる．さらに, *ME* 法は, 日本語係り受け解析でも非常に有用で, 品詞の情報だけでなく, 頻度の高い単語に対しては語彙的情報も加えて学習すると言った柔軟な素性の追加が容易であるという特徴がある．

金山ら *[2]* の実験における精度は, *ME* 法を用いることにより多くの素性を追加できたことにより, 単純な相対頻度で推定した *3* つ組／ *4* つ組モデルよりも約 *1.9* %向上してる．

第 3 章

3つ組／4つ組モデル

本章では, 金山ら *[2]* が開発した, 人手で記述させた文法および統計情報を用いて日本語の係り受け関係を求める手法について述べる. 特に, 文法とヒューリスティクスにより文節の係り先の候補をしぼった時に構成することができるモデルを解説し, それにより高い係り受けの精度（文節正解率 *88.6*％）が得られることを示す.

3.1　3つ組／4つ組モデルについて

金山らのグループでは、*HPSG* の枠組みに基づいた文法を作成している。現在では、新聞や雑誌などの実世界の文章のほとんどに対して構文木を出力できる, 被覆率の高い日本語文法 *SLUNG* を開発した. しかしながら, 文法的に可能な構造を列挙するだけでは, 曖昧性が大きいため, 実用に耐えない。また、今後の研究されるであろう高度な自動学習のためにも、曖昧性の解消が要求去れている.

3 つ組／ *4* つ組モデルでは, 文法を用いた構文解析の結果の曖昧性解消を目的として, 文節単位の係り受け解析によって、最も可能性の高い統語構造を選択できるようにするモデルでる. また, 係り受け解析を行う際に文法を用いることが精度の向上に寄与している. 係り受け解析は、係り受け解析は以下のような手順でなされる.

- 日本語文法 *(SLUNG)* で構文解析し, 各文節の係り先の候補を, 文法が許す文節に絞る.
- 文法により係り受け文節候補を絞った各文節に対して, 係り先文節候補が *4* 文節以

上存在する場合, 係り受け文節候補を 3 文節以下に絞る.

- 係り元文節がそれぞれの候補に係る確立を, 3つ組／ 4 つ組モデルを用いて求める.
- 算出された各係り受け確率を元に, 最も優先度が高い文を選択する.

3つ組／ 4 つ組モデルでは *SLUNG* により構文解析をしたのちに, 各係り元文節に対して係り受け文節候補を 3 文節以下に絞る. 候補を絞り込む方法としては, ただ闇雲に係り元文節から近いため係り受け文節候補を絞り込むのではなく, 観察に基づいたヒューリスティクスにより絞り込む.

係り先文節候補の絞り込む基準として, 3つ組／ 4 つ組モデルでは, 各係り元文節から見て *(1)* 最も近い文節, *(2)2* 番目に近い文節, *(3)* 最も遠い文節, の 3 文節を係り先文節候補としている. これは, 係り元文節と前述の 3 文節のいずれかと係り受け関係が成立する確率が非常に高いためである.

係り先文節候補を 3 文節以下に絞り込んだ後に, 係り元文節と係り先候補である 3 文節間の係り受け確率を算出する. 3つ組／ 4 つ組モデルでは, 係り受け確率を推定する際に, 係り元文節の属性と全ての係り先文節候補の属性を同時に考慮するという特徴により, 各係り先文節候補を相対的に評価することができる.

各文節の係り受け確率の推定が終了した後, 推定した係り受け確率を構文木に反映し, 反映された係り受け確率の積が, 最大になる構文木を選択することにより最も最適な係り受け関係を持った文が選ばれることになる.

この処理より, 本章で述べているモデルと他の研究での統計モデルとは大きく異なる. 従来の研究で用いられている統計モデルは, 係り元文節 i 係り先文節 j に対して, 係り元文節の属性 Φ_i 及び係り先の文節の属性と係り元文節と係り先文節間の属性 $\Psi_{i,j}$ が存在するとき, 係り受けが成立する（構文木 T ができる）条件つき確立を求めていた.

$$P\left(i \rightarrow j\right) = P\left(T|\phi_i, \psi_{i,j}\right) \tag{3.1}$$

これに対し, 本章で用いる 3つ組／ 4 つ組モデルでは, 係り元文節 i の係り先文節候補 t_n について, 係り元文節 i の属性を ϕ_i, 係り先文節候補 t_k の属性と, 係り元文節 i と係り先文節候補 t_k の文節間の属性を ψ_{i,t_k} とするとき, Φ_i と全ての係り先候補 t_k に対する Ψ_{i,t_k} が存在するとき, n 番目の候補が選ばれる条件つき確立を求めるものである.

$$P(i \to t_n) \;=\; P(n|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}) \quad \text{（候補が } 2 \text{ つのとき；} n = 1,2\text{）} \tag{3.2}$$

$$P(i \to t_n) \;=\; P(n|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3}) \quad \text{（候補が } 3 \text{つのとき；} n = 1,2,3\text{）} \tag{3.3}$$

上記の式 *3.3,* **??**をそれぞれ *3* つ組モデル・*4* つ組モデルと呼ぶ. なお、ここでの n 番目の候補とは、表層文中で係り元から数えて n 番目の文節ではなく, 文法的に許される係り先の内 *2* つ又は, *3* つに絞ったものの中で, 係り元文節から n 番目の文節であるかを示す。

3.2　3つ組／4つ組モデルの解析の流れ

本節では, 「*3* つ組／ *4* つ組モデル」も用いて係り受け解析をする手順を説明する. 本モデルの係り受け解析の流れは図 *3.4* のようになっている.

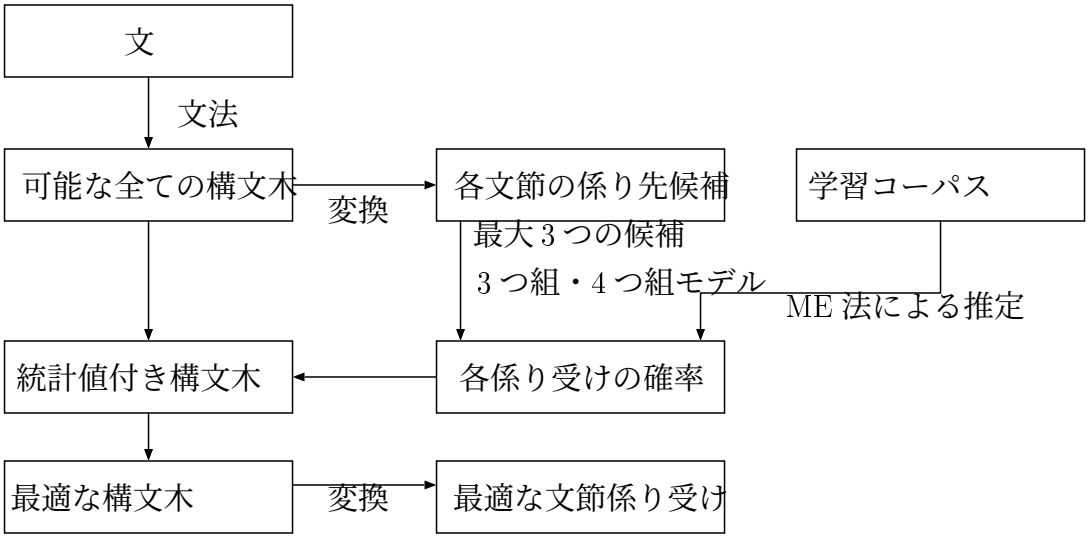

図 3.1 3 つ組／ 4 つ組モデルでの係り受け解析の流れ

このモデルでは，文を入力とし，*JUMAN* で形態素解析をした後，文法 *SLUNG* で構文解析をする．*SLUNG* は，*JUMAN* により区切られた形態素を解析の単位としており，文法的に正しいすべての構文木を出力することができる．この *SLUNG* によって出力された各構文木中の部分木に対して，図 *3.2* のように，部分木と係り受け構造との対応付

表 3.1 係り先文節候補の数に対する, 正しい係り先の分布（単位 %）
「比率」とは候補の数の分布を示し, 括弧付きの値は他項との重複を示す.

候補の数	比率	第一	第二	第三	第四	..	最遠	第一, 第二, 第三
1	32.7	100	-	-	-	-	(100)	100
2	28.1	74.3	26.7	-	-	-	(26.7)	100
3	17.5	70.6	12.6	(16.8)	-	-	16.8	100
4	9.9	70.4	11.1	4.7	(13.8)	-	13.8	95.3
5	5.4	70.1	11.6	4.2	2.5	..	11.5	93.2
6 以上	6.4	70.3	10.8	3.9	2.4	..	9.6	90.7
合計	100	-	-	-	-	..	-	**98.6**

けを行う．図 *3.2* の例では，部分木 M 中の左部分木 L，右部分木 R のもっとも右側にある語を, それぞれ l,r と定義し, それらが属する文節を $b(l)$, $b(r)$ とするとき, $b(l)$ が $b(r)$ に係ることになる.

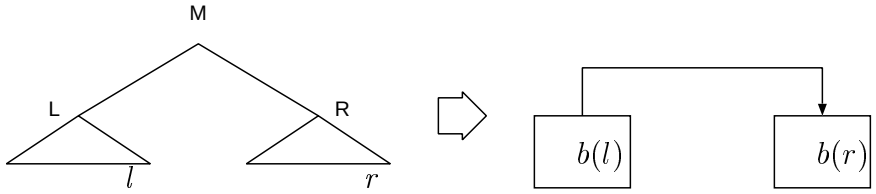

図 3.2 部分木から文節係り受けへの変換

図 *3.4* , 図 *3.2* の様に, *JUMAN* および *SLUNG* にて文法的に正しい構文木から係り受け関係へ変換後, 係り受け候補をヒューリスティクスにより最大 *3* 文節までに絞る.

日本語の文節の係り先の傾向として, 以下のような傾向が（丸山, 荻野 *1992*）でも分析されている.

(1) 係り元文節から遠くなるにつれて, 係り受け確率が減少する.

(2) 最も遠い文節に係る場合だけは比較的多いことが知られている.

SLUNG により係り先文節候補を絞った場合にもこの傾向にある. *EDR* コーパスの文を *SLUNG* で解析した際の, 係り先候補の数, 及び正しい係り先文節の位置の分布を表

3.1 に示す. 表中の「第一」「第二」... は, 文法で制限された係り先文節候補を, 係り元文節から何番目に近い文節であるかを意味している. 更に, 「最遠」とは, 係りもと文節から最も遠い係り先候補を意味してる.

表 *3.1* に示すデータより, 係り元文節から（*1*）最も近い文節, （*2*）二番目に近い文節, （*3*）最も遠い文節のいずれかに係る場合だけで *98.6%*を占めることがわかる. この性質を利用して, 係り先の候補が *4* 文節以上存在する場合にも上記の *3* 文節だけを考慮し, 残りの文節の情報を無視する. このように制限する事によって, 係り受け精度の上限は *98.6%*となるが, 残りの *1.4%*を犠牲にすることにより問題を大幅に単純化することができる.

3.3 3つ組／4つ組モデル

*3*つ組／ *4* つ組モデルは, 文節 i が文節 t_n に係る確率 $P(i \to t_n)$ を式 *3.5, 3.5*で算出する. 但し, t_n は文節 i の *3*文節以下に制限された係り先文節候補であり, Φ_i は文節 i の属性, Ψ_{i,t_n} は t_n と文節 i, t_n 間の属性を表す.

$$P(i \to t_n) = P(n|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}) \quad \text{（候補が } 2 \text{ つのとき：} n = 1,2) \tag{3.4}$$

$$P(i \to t_n) = P(n|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3}) \quad \text{（候補が } 3 \text{つのとき：} n = 1,2,3) \tag{3.5}$$

このモデルの特徴は, 上記の式から推測される通り, **「係り元文節と, 係り先文節の候補となる全ての文節の属性を同時に考慮できること」, そして「それぞれの係り先の候補の係りやすさを求めるのではなく, 各候補が選ばれる確率を求める」ことである. これらの意義は次の 3 点にある. これらの意義は次の 3 点になる.**

意義 1 文節間の距離ではなく, 係り先の文節候補中の相対的な位置 を用いて係り先を選択できる.

意義 2 着目している候補だけでなく, 文脈, すなわち他の候補の属性を考慮できる.

意義 3 ある係り元に対する全ての候補への係りやすさを, 同じ条件の下で計算できる.

以下に, 上記の意義について順に詳しく述べる.

意義 1：候補の中での相対的位置

文節間の距離は, 係り受け解析における重要な要素として考えられているが, 係り先の候補中の位置の方が重要な場合がある. 例として, 1 の各文における「彼女が」の係り先を推定する時を考える. 両者とも「走るのを」が正しい係り先と考えられる.

1(a) 彼女が 走るのを 見た ことが ありますか.
(b) 彼女が ゆっくり 走るのを 見た ことが ありますか.

文法を用いずに文節数を距離とするモデルでは,「彼女が」と「走るのを」の文節間距離は a では 1，b では 2 と異なっている反面，a での「彼女が → 見た」と b での「彼女が → 走るのを」が，係り元からの距離が 2 つである動詞であるという点で，似た事象であると見なすことができる.

一方，文法で係り先を絞った場合，a,b とも「彼女が」の係り先文節候補は「走るのを」と「見た」の二つになる. このように，係り先候補のみに着目するれば，両者を同じ事象としてあつかえるので，より効率のよい学習が行えるようになる.

意義 2：文脈の考慮 1 において,「私の」についての係り先を考える. 正解文節は，それぞれ「娘に」「友人の」である.

1(a) 私の かわいい 娘に 道で ばったり 会った.
(b) 私の 友人の 娘に 道で ばったり 会った.

係り元文節と係り先文節，及び文節間の距離を考えるモデルでは，a,b における「私の → 娘に」は区別されることなく，まったく同じ係り受け確立が付与される. しかしながら，この確率は非常に低くなる. なぜなら，実際に EDR コーパスの一部を観察したデータによると，a の「N_1 の A N_2」という構文に対して，b のような「N_1 の N_2 N_3」の構文の頻度が 4 倍程度あり，後者の構文では，N_1 は近くの N_2 を修飾する場合が約 75%と，圧倒的に多いからである. したがって，a において,「私の → 娘に」に比べて「私の → かわいい」の確率の方が高くなり，解析誤りを引き起こす.

係り元と係り先の 3 つの候補全てを同時に考慮すると，この誤りを防ぐことができる. a において「私の」と，その係り先候補である「かわいい」「娘に」「会った.」を同時に考えて，三文節がそれぞれが選ばれる確率を計算した場合，第二候補であっても，第一

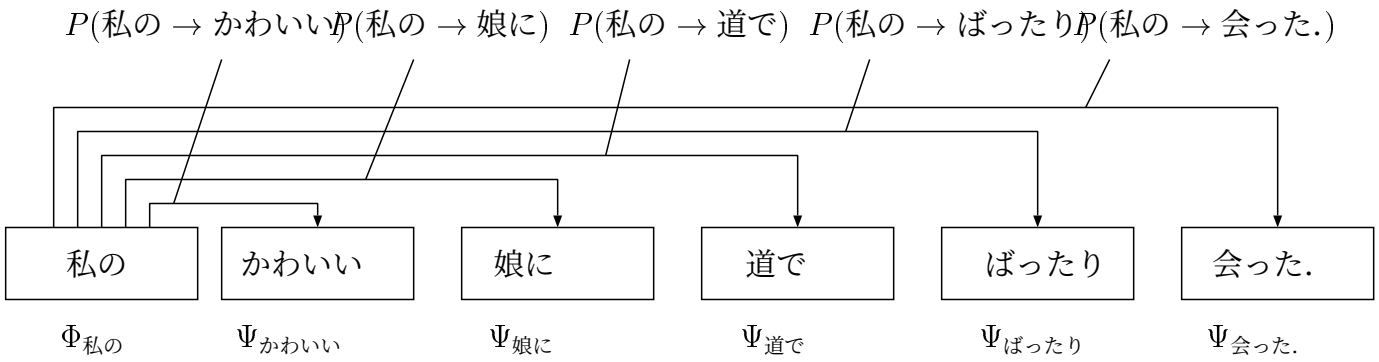

図 3.3 3 つ組／ 4 つ組モデルで考慮する条件 係り元と 3 つに絞られた係り先文節候補の属性を用いて，それぞれの候補に係る確率を求める

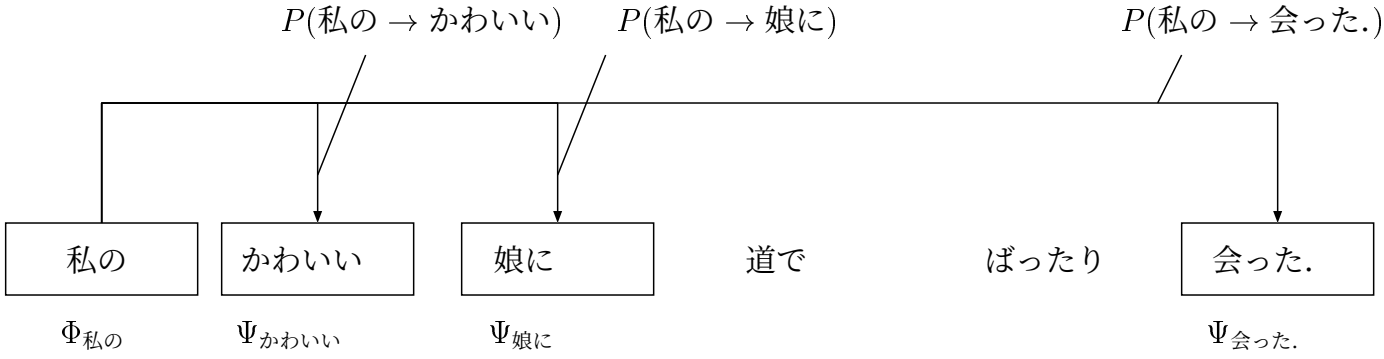

図 3.4 3 つ組／ 4 つ組モデルで考慮する条件 係り元と 3 つに絞られた係り先文節候補の属性を用いて，それぞれの候補に係る確率を求める

候補の形容詞連体形よりも高い確率が当てられ，正しくかかり先を求めることができる．このような現象は，第一候補である形容詞や副詞を飛び越えて第二候補にかかるケースなどで一般に数多く見受けられる．

意義 3：同じ条件下での係りやすさの計算

これは，意義 2 との関連するが，あるひとつの係り元に対する係り受けの確率を，共通の前件を持った条件付き確率で計算できるという利点である．1a の「私の」の係り先を考える際には，従来の手法は式 3.7，3 つ組／ 4 つ組モデルでは式 3.8 を求めることになる．3.7 ではそれぞれの条件付き確率の前件が異なるため，5 つの値の和が 1 にならないのに対して，3.8 では，3 つの和が 1 になる．したがって，3 つ組／ 4 つ組モデルにおいて推定する条件付き確率は，係り元とその係り先文節候補がある文節において，それぞれの係り先文節候補が選ばれる確率に一致することになる．なお，考慮する条件を図示するとそれぞれ図?? 図??のような関係となる．

$$
\begin{aligned}
P(\text{私の} \rightarrow \text{かわいい}) &= P(T|\Phi_{\text{私の}}, \Psi_{\text{かわいい}}) \\
P(\text{私の} \rightarrow \text{娘に}) &= P(T|\Phi_{\text{私の}}, \Psi_{\text{娘に}}) \\
P(\text{私の} \rightarrow \text{道で}) &= P(T|\Phi_{\text{私の}}, \Psi_{\text{道で}}) \\
P(\text{私の} \rightarrow \text{ばったり}) &= P(T|\Phi_{\text{私の}}, \Psi_{\text{ばったり}}) \\
P(\text{私の} \rightarrow \text{会った.}) &= P(T|\Phi_{\text{私の}}, \Psi_{\text{会った.}})
\end{aligned}
\tag{3.6}
$$

$$
\begin{aligned}
P(\text{私の} \rightarrow \text{かわいい}) &= P(1|\Phi_{\text{私の}}, \Psi_{\text{かわいい}}, \Psi_{\text{娘に}}, \Psi\text{会った.}) \\
P(\text{私の} \rightarrow \text{娘に}) &= P(2|\Phi_{\text{私の}}, \Psi_{\text{娘に}}, \Psi_{\text{娘に}}, \Psi\text{会った.}) \\
P(\text{私の} \rightarrow \text{会った.}) &= P(3|\Phi_{\text{私の}}, \Psi_{\text{会った.}}, \Psi_{\text{娘に}}, \Psi\text{会った.})
\end{aligned}
\tag{3.7}
$$

3.4 最適な係り受けの選択方法

各文節間のかかりやすさ $P(i \rightarrow j)$ を求めるにあたって，係り元文節に対する係り先文節の候補数によって，次のようなモデルを用いる．

- **係り先文節候補が 1 つの場合：その係り先文節を確定するため，$p(i \to j) = 1.0$ とする．**
- **係り先文節候補が 2 つの場合：係り元文節と 2 つの係り先文節の情報を考慮する「3 つ組モデル」を用いて選択する．**
- **係り先文節候補が 3 つの場合：係り先文節候補の内，係り元文節に最も近い文節，二番目に近い文節，最も遠い文節の 3 つだけを考え，係り元文節と 3 つの係り先文節候補の情報を考慮した「4 つ組モデル」を用いて選択する．**

上記の方針に従って求められた値を用いて，SLUNG の出力した全ての部分木 M に対して，統計値 $Q(M)$ を以下のようなアルゴリズムで割り当てる．なお，SLUNG の出力する構文木の終端記号は，文節単位ではなく，単語（JUMAN の出力する形態素）を単位としてる．

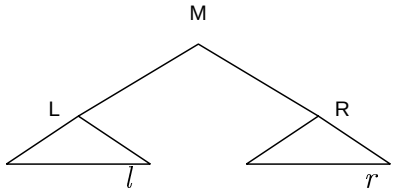

図 3.5 図 5 SLUNG の出力する部分木 M

- **部分木 M が単一の単語からなる場合，$Q(M) = 1.0$**
- **上記以外の場合，図 3.5 の部分木において，左部分木 L の最も右側の単語 l，右部分木 R の最も右側の単語 r，** l,r の属する単語をそれぞれ $b(l), b(r)$ とし，以下の計算式で計算する．

$$Q(M) = Q(L) \times Q(R) \times P(b(l) \to b(r)) \tag{3.8}$$

文全体に対応する構文気で，この統計値が最大になるようなものを探索し，その構文木を再び文節の係り受け関係に変換して出力する．こうして得られた文の係り受けは，必ず文法的に正しい構文木に対応しており，係り受け同士が交差することはない．

第 4 章

Support Vector Machine

Support Vector Machine（SVM）は, Vapnik[1] によって提案された手法であり, 学習データと分離平面の間隔（マージン）を最大化するような戦略に基づいた統計的機械学習手法の一種である.

4.1 SVM の概要

SVM の概念図を, 図 4.1 に示す. SVM は, 正例・負例を表すデータ y_i と n 次元の素性ベクトル x_i との対で表されている l 個の学習データから, 正例と負例を正しく分類できる分離平面を求める, 二値線形分類器である.

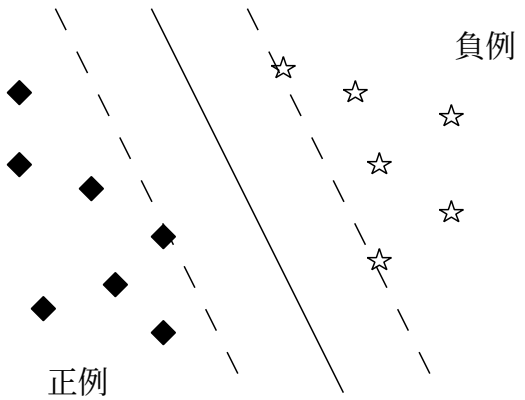

図 4.1 SVM の概念図

SVM における分離平面の決定は, 正例の学習データと負例の学習データとのマージン

(図 *4.1* では, 分離平面に並行で等距離にある一点鎖線の間) が最大になるように, 分離平面を決める. マージンを最大化は $\|w\|$ の最小化を意味しており, これは式 *4.1* を式 *4.2* の条件で最大化する双対問題と等価であることが知られている.

$$\sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j}^{l} y_i y_j \alpha_i \alpha_j (x_j \cdot x_i) \tag{4.1}$$

$$\sum_{i=1}^{l} y_i \alpha_i = 0, \quad 0 \leq \alpha_i \tag{4.2}$$

$$\begin{aligned} f(x) &= sgn(w \cdot x + b) \\ &= sgn\left(\sum_{i=1}^{l} \alpha_i y_i \left(x_i \cdot x\right) + b\right) \end{aligned} \tag{4.3}$$

4.2 Kernel関数

一般的な分類問題においては, 学習データを線形分離することが困難な場合が多々ある. このような場合, 各素性の組み合わせを考慮しつつ, 現在対象にしている次元よりも, より高次元な空間に学習データを写像することにより, 線形分離が容易になることが知られている. しかしながら, ただ単純に学習データの全ての組み合わせを展開し, 高次元空間への写像をおこなうと莫大な計算量が必要になる.

今仮に, 学習データ x を写像関数 $\mathbf{\Phi}$ によって, 高次元空間に写像した場合を考える. このとき, 写像関数を識別関数 (式 *4.3*) に代入すると式 *4.4* となる. 式 *4.3* 式 *4.4* では, 関数の重要な部分がベクトルの内積計算となっている. この内積計算を, 写像関数 $\mathbf{\Phi}$ を介する事無く計算する事ができれば大幅に計算量を押えることができる.

$$\begin{aligned} f(x) &= sgn\left(\sum_{i=1}^{l} \alpha_i y_i \left(\Phi(x_i) \cdot \Phi(x)\right) + b\right) \\ &= sgn\left(\sum_{i=1}^{l} \alpha_i y_i K\left(x_i, x\right) + b\right) \end{aligned} \tag{4.4}$$

$$K(x_i, x) = \Phi(x_i) \cdot \Phi(x) \tag{4.5}$$

そこで, 式 *4.5* となる関数 K が存在すればよい. この関数 K を適当に選んだとき, 式 *4.5* を満たす $\mathbf{\Phi}$ の存在が証明されていればよい. 関数 K が式 *4.5* を満たすには, 一部例外が存在するが, *Mercer* 条件を満足すればよいことが知られている.

*Mercer*条件を簡単にいうと下記の様になる.

- $a \cdot b = b \cdot a$ が成り立つ
- 写像した空間で距離が定義できること

このようにして, 条件を満たした関数 K は *kernel* 関数と呼ばれ, 最適化問題や識別問題における計算量を大幅に減少できる. *kernel* 関数の代表的な物として *sigmoid* 関数 *(*式 *4.6)*, *polynomial* 関数 *(*式 *4.7)*, *Radial Basis Function(RBF)(*式 *4.8)* がある.

$$K(x, y) = \tanh(ax \cdot y - b) \tag{4.6}$$

$$K(x, y) = (x \cdot y + 1)^d \tag{4.7}$$

$$K(x, y) = \exp\left(\frac{-\|x - y\|^2}{2\sigma^2}\right) \tag{4.8}$$

本研究では, 他の係り受け解析研究でも使用されている *polynomial* 関数 *(*式 *4.7)* を *kernel* 関数として選択した. *polynomial* 関数は, d 個の素性の組み合わせを考慮した次元への写像を意味し, 直観的解釈が容易なためである.

4.3 SVM による多値分類への応用

SVM は, 本来学習サンプルと分類境界の間隔を最大化するような戦略に基づく手法の二値分類器である. しかしながら, 本研究で用いる日本語係り受け解析モデルである「*3* つ組／ *4* つ組モデル」では, 多値分類を扱う必要がある. そこで, 既存の *SVM* を多値分類器に拡張する手法について説明する.

4.3.1 pairwise 法

pairwise 法は, 分類したい k 個のクラスから任意の *2* つに関する二値分類器を ${}_kC_2$ 個構築する方法である. 例えば, クラス a , クラス b を分類する二値分類器 $f_{ab}(x)$ があるとき. $f_{ab}(X)$ は, $f_{ab}(X) \geq 0$ のとき事象 **x** **をクラス** a と判定し, $f_{ab}(X) < 0$ のときはクラス b と判定する. さらに, クラス c の投票数 V_c を, ${}_kC_2$ 個ある二値分類器のうちクラス

c と判定した分類器の個数とする．これにより，*pairwise* 法では最終的な分類クラスは，投票数 V が最も多いクラスに決定される手法である．図 *4.2* に *pairwise* 法の例を示す．図 *4.2* の様に *A,B,* 及び *C* の *3* クラスに分類を行なうために、任意の *2* つのクラスを分類する二値分類器，$f_{AB}(X), f_{AC}(X), f_{BC}(X)$ を構築する．

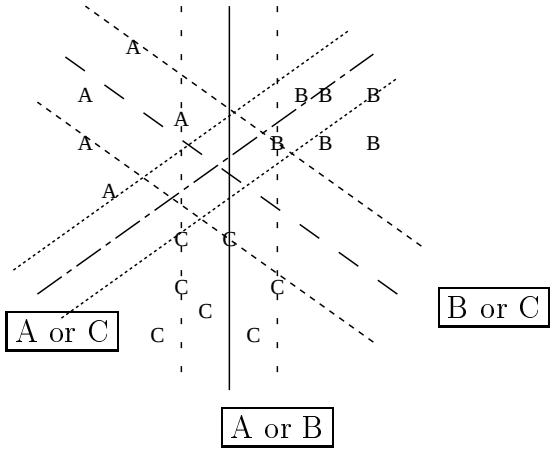

図 4.2 pairwise 法

4.3.2 one vs. rest 法

one vs. rest 法は，k 個の各クラスに対して，あるクラスか，それ以外のクラスであるかという二値分類器 $f_c(X)$ を，k 個構築する手法である．*SVM* でこの手法を適用する場合，未知事例 x' に対して，$f_c(X')$ の値が最大となる分類器に対応するクラスに決定する．図 *4.3* に *one vs. rest* 法の例を示す．図の様に *A,B* 及び *C* の *3* クラスを *one vs. rest* 法により分類するためには，$f_A(X), f_B(X)$，及び $f_C(X)$ の *3* つの分類器を構築する．

本研究では，多値分類にこの *one vs. rest* 法を用いる．上記のような *3* つのクラスに分類する場合 *one vs. rest* 法では，全分類器を作成するのに，全学習データを *3* 回使う必要がある．一方，*pairwise* 法で同じ学習を行なう場合には，一つの分類器に対して使用する学習データは，分類器で判定する *2* つのクラスのデータのみである．したがって，*3* つのクラスを *pairwise* 法で分類するには，全学習データを *2* 回分使用するだけで済むことになりその分だけ学習時間が少ないという利点がある．しかしながら，山田ら *[6]* の報告でもあるように，*one vs. rest* 法の学習速度は，*pairwise* 法に劣るが学習精度では優ってい

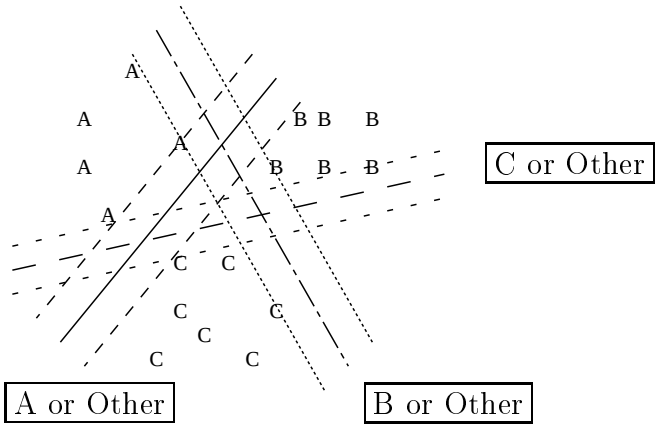

図 4.3 one vs. rest 法

るためである.

4.4 SVM に基づく係り受け解析

最初に日本語における一般的な統計的係り受けモデル，及び解析手法について説明する．まず，あらかじめ文節にまとめられ属性付けされた文節例 $\{b_1, b_2, \cdot, b_n\}$ を B，係り受けパターン列 $\{Dep(1), Dep(2), \cdots, Dep(n-1)\}$ を D と定義する．ただし，$Dep(i)$ は，文節 b_i の係り先文節番号を示す．これ以降，D は以下の制約を満たすものと仮定する.

- 文末を除き, 各文節はその文節の後方側に必ず一つの係り先を持つ.
- 係り受け関係は交差しない

統計的係り受け解析とは, 上記の二つの制約のもとで, 入力文節列 B に対する条件付き確率 $P(D|B)$ が最大にする係り受けパターン列 D を求めることと定義できる.

$$D_{best} = \underset{D}{argmax}\, P(D|B) \tag{4.9}$$

さらにここで, それぞれの係り受け関係は独立であると仮定すると, $P(D|B)$ は,

$$P(D|B) = \prod_{i=1}^{n-1} P(Dep(i) = j|f_{ij}) \tag{4.10}$$

$$f_{ij} = \{f_1, f_2, \cdots, f_n\} \in R^n \tag{4.11}$$

のように変形できる．ここで，確率 $P(Dep(i) = j|f_{ij})$ は文節 b_i と文節 b_j が言語的素性集合 f_{ij} を持つときに，文節 b_i が文節 b_j に係る確率を示す．f_{ij} は文節 b_i と文節 b_j に関する様々な言語的特徴を表す n 次元の特徴ベクトルである．

一般的な統計的な係り受け解析では上記の式により算出された確率値をもとに D_{best} を決定している．SVM は，この一般的な統計的係り受けモデルに準拠した形で表現される．SVM は正例，負例の二値分類を行なう学習モデルである．したがって，何を正例，負例として学習するかを決める必要がある．さらに，本研究では，3 つ組／4 つ組モデルに SVM を組み込むため，多値分類に適応したデータを与える必要がある．

第 5 章

3つ組／4つ組法とSVMを用いた構文解析手法

本章では, 本研究の要である *3* つ組／ *4* つ組モデルの係り受け確率の推定を, *ME* 法から *SVM* に変更した際の相違点について詳しく説明する.

5.1 3つ組／4つ組法へのSVMの導入

本研究では, 従来の *3* つ組／ *4* つ組モデルで文節係り受け確率の推定に使われている *ME* 法に変わって, 図 *5.1* の様に *SVM* を, 文節の係り受け確率推定に用いる.

SVM を *3* つ組／ *4* つ組モデルに組み込む利点として, 以下の理由をあげることができる.

- 素性を人手で選択することなく, 学習することができる.
- *ME* 法のように, 明示的に素性どうしの共起, 依存関係を与えなくても *SVM* では, *Kernel* 関数により自動的に学習が可能になる.
- *SVM* は, 数ある統計的確率推定モデルの中で最も高い汎化能力を持ち過学習をおこしにくい.

ME 法や隠れマルコフモデルなどの, 従来の統計的な確率推定モデルは, 素性どうしの組み合わせを効率良く学習できず, 有効な組合せの多くは人間の発見的な手続きで決定

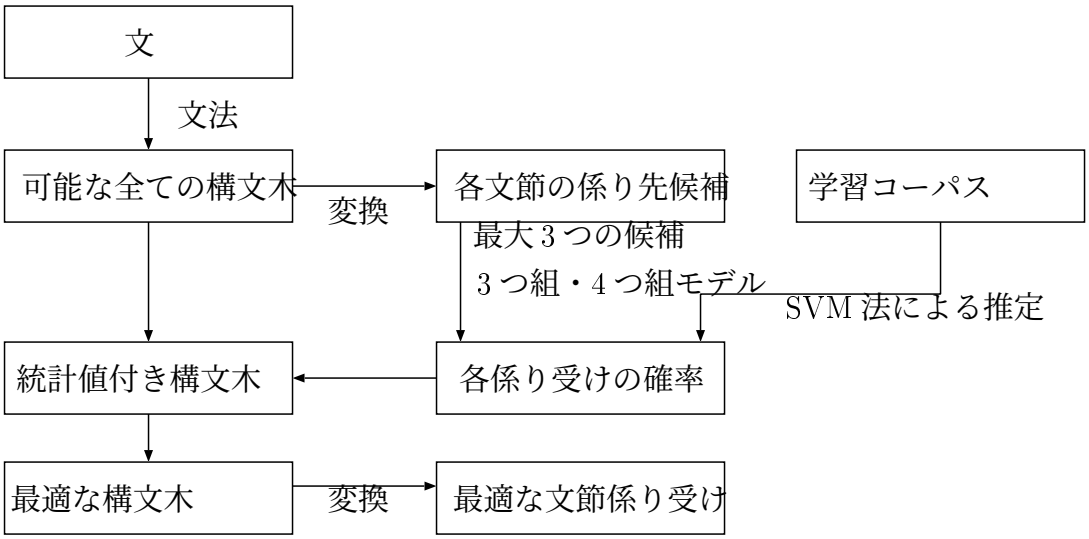

図 5.1 SVM を組み込んだ 3 つ組／ 4 つ組モデルでの係り受け解析の流れ

されている. さらに, 多くの機械学習アルゴリズムは, 高い精度を求めるために慎重に素性の選択をする必要があり, 人手による発見的な手続きにたよっている場合が多い.

SVM は, 学習データと分類境界面との間隔（マージン）が最大になるように線形分離する線形分離器であるが, *Kernel* 関数を用いることにより非線形分離が可能になり, さらに, *kernel* 関数による写像時に素性の組み合わせが動的に計算されている. 以下に, 次数が $d = 2$ のときの *polynomial* 関数による例を上げる.

$$
\begin{aligned}
& d = 2 \quad x = (a_1, a_2) \in R^2, \quad y = (b_1, b_2) \in R^2 \\
K(x, y) \quad &= \quad (x \cdot y + 1)^2 = (a_1 b_1 + a_2 b_2 + 1)^2 \\
&= \quad a_1^2 b_1^2 + a_2^2 b_2^2 + 2a_1 b_1 + 2a_2 b_2 + 2a_1 a_2 b_1 b_2 + 1 \\
&= \quad (a_1^2, a_2^2, \sqrt{2}a_1, \sqrt{2}a_2, \sqrt{2}a_1 a_2, 1) \cdot (b_1^2, b_2^2, \sqrt{2}b_1, \sqrt{2}b_2, \sqrt{2}b_1 b_2, 1)^T
\end{aligned}
$$

このように, 素性ベクトルが *2*次元で, 次数が *2*である場合 *polynomial* 関数では, 素性ベクトルが *2*次元から *6*次元へ写像することができる. ここで注目すべきは, $\sqrt{2}a_1 a_2, \sqrt{2}b_1 b_2$ の部分である. この部分は, *kernel* 関数を計算することによって動的に, 素性の組み合わせ計算が行なわれている. 一般に, d 次の *polynomial* 関数は, d 個までの組み合わせを含めた学習モデルとなる.

一般的な統計的確率推定モデルでは, 素性数が多くなると過学習を起こしやすい. *SVM* では, *VC*次元と呼ばれる学習モデルの複雑さを示す指標がある. *SVM* では, この *VC*次

元を最小化することで汎化誤差を最小化することができる．SVM の VC 次元 h は, 学習データ全体を覆う最小超球の直径 D, 分離平面の間隔（マージン）ρ と素性データの次元数 n により式 *5.1* に示す上限値の存在が証明されている．

$$h \leq \min\left(\frac{D^2}{\rho}, n\right) + 1$$

ここで n が十分に大きい場合, ρ（マージン）の最大化が VC 次元を最小化する．その結果, 高次元空間でも汎化誤差の上限を最小にすることが可能である．したがって, 素性数が多くとも過学習をおこしにくい．

本研究では, *3* つ組／ *4* つ組モデルの確率推定に用いる SVM に, 工藤 *[8]* が開発した, バイナリの素性表現に特化して高速化を試みられた SVM である $TinySVM$ を用いる．

5.2 3つ組／4つ組モデルへのSVMの適用

本研究で使用する *3* つ組／ *4* つ組モデルは各文節に対して, *3.3* でも説明したように, 最も近い文節, 二番目に近い文節, 最も遠い文節の *3* 値に分類しなければならない．そこで, SVM には *4.3* 節で述べた通り, 多値分類器である ”*one vs. rest* 法 ”を用い, *3* つ組モデル, *4* つ組モデルのそれぞれ係り受け関係に対して作成する．作成するモデルは, *3* つ組／ *4* つ組法において文法に基づいて制限された係り受け関係基づき, *(1)* 係り元文節と係り元文節に最も近い文節間に係り受け関係が成立するか否か, *(2)* 係り元文節と係り元文節に *2* 番目に近い文節間に係り受け関係が成立するか否か, *(3)* 係り元文節と係り元文節から最も遠い文節間に係り受け関係が成立するか否か, の *3* つのモデルを SVM によって学習する．

式 *5.2* に *4* つ組モデルにおける SVM での各係り受け確率の算出式を示す．式 *5.2* では, 評価データ x と分離平面間の距離をシグモイド関数に代入した形となっている．これは, SVM での距離関数を疑似的に確率値の値域に正規化することで, 従来からある確率モデルの枠組で解析することができる．

$$\begin{aligned} P(1|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3})' &= \tanh\left(\sum_{i=1}^{l} \alpha_{i1} y_{i1} \left(x_{i1} \cdot x\right) + b\right) \\ P(2|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3})' &= \tanh\left(\sum_{i=1}^{l} \alpha_{i2} y_{i2} \left(x_{i2} \cdot x\right) + b\right) \end{aligned} \tag{5.1}$$

$$P(3|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3})' = \tanh\left(\sum_{i=1}^{l} \alpha_{i3} y_{i3} (x_{i3} \cdot x) + b\right)$$

しかしながら, 式 *5.2* だけでは, *3* つ組／ *4* つ組モデルで算出される条件付き確率の和が *1* になるという *3* つ組／ *4* つ組モデルの原則に反する. 従って, 式 *5.3* の計算を行ない *3* つ組／ *4* つ組モデルとの整合性をとる.

$$\begin{aligned}
&P(1|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3}) = \\
&\qquad \frac{P(1|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3})'}{P(1|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3})' + P(2|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3})' + P(3|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3})'} \\
&P(2|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3}) = \\
&\qquad \frac{P(2|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3})'}{P(1|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3})' + P(2|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3})' + P(3|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3})'} \\
&P(3|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3}) = \\
&\qquad \frac{P(3|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3})'}{P(1|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3})' + P(2|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3})' + P(3|\Phi_i, \Psi_{i,t_1}, \Psi_{i,t_2}, \Psi_{i,t_3})'}
\end{aligned} \tag{5.2}$$

5.3 SVM で使用する素性について

本研究上で *SVM* での学習及び, 識別に使用する素性は, 従来の金山ら ??が *3* つ組／ *4* つ組モデルの係り受け確率推定に用いていた *ME* 法で使用していた素性の内, *ME* 法の為に人手により作成された素性の組合せを考慮した素性以外の素性と, 学習コーパス中に現われた係り元文節及び, 各係り先文節候補における *Head* 素性として現われる単語を, それぞれの文節毎にナンバリングし, それを, *SVM* における素性として利用した.

学習に利用した素性の品詞等の分類は, *JUMAN* の分類体系を使用しており, 以下に各素性について解説する.

品詞　語形・主辞ともに, JUMAN の品詞細分類を用いている.

助詞・副詞　頻度の高い 26 種類の助詞と 69 種類の副詞.

主辞語彙　品詞とは別に, 主辞として現われる語の内, 頻度が高い 294 種類の語彙.

語形語彙　品詞が「助動詞」「接尾辞」となるもの語の内, 頻度が高い 70 種類の語彙.

活用形　JUMAN の活用形を, 「基本形」「連用形」「連体形」「テ形」「タ形」「その他」の 6 種類に分類したもの.

文節間読点の数・「は」の数　係り元と係り先の文節間にある読点の数を, 「0」「1」「2」「3 以上」の 4 値で表す. 同様に, 副助詞「は」の数を「0」「1」「2 以上」の 3 値で表す.

主辞として現われる単語　学習コーパス中に表れた, 全ての係り受け関係に対して, 係り元文節の主辞と, 各係り先文節候補の主辞を, それぞれ, 係り元文節, 係り元文節に最も近い文節, 二番目に近い文節, 最も遠い文節別に, 集計しそれぞれの文節で出現した単語に対して, 素性番号を与え学習データに加える.

この表 5.1 中の ”取りうる数 ”とは，文字通り各素性が取りうる値の総数である．この表中で，主辞の単語に関する素性の ”取りうる数 ”の上限が示されていない．これは，この主辞の単語を表す素性は，与えられた学習コーパス上から係り受け関係を抽出した時に動的に発見される物であるため，事前に取りうる数を把握することができないためである．表に記されている数値は，EDR コーパスを 19 分割した物の一つを学習コーパスとして与えた時に，得られた単語数である．

実際の 3 つ組／ 4 つ組モデルの係り元文節と係り先文節候補の集合は，係り先文節候補数分（3 つ組モデルの場合，2 文節．4 つ組モデルの場合，3 文節）の ”係り先文節に関する素性 ”が存在する．

上記のレベルの素性は，従来通りの ME 法による 3 つ組／ 4 つ組モデルでの話である．本研究で使用する SVM には与える素性データを，表 5.1 の素性番号をそのまま素性として利用し，”取りうる数 ”をその素性の値とするの手法ではなく，他の手法を使用する．他の手法とは，現在注目している（表 5.1 の）素性番号が示している値に，前回までの素性が取りうる値の合算を加算した値を素性番号とし，その素性が存在していることを示す値として 1 を素性の値とする．これは，一つの評価軸（素性）で複数の品詞などを評価するより，一つの評価軸に対して一つの品詞などが評価する方が直感的に分かりやすく，評価しやすいからである．

表 5.1 SVM による学習に使われた素性

係り元文節に関する素性		
素性番号	素性の種類	取りうる数
1	係り元主辞品詞	24
2	係り元語形品詞	34
3	係り元助詞	27
4	係り元副詞	70
5	係り元語形語彙	71
6	係り元活用形	6
7	係り元読点の有無	2
係り先文節に関する素性		
8	係り先主辞品詞	24
9	係り先語形品詞	34
10	係り先主辞語彙	295
11	係り先助詞	27
12	係り先語形語彙	71
13	係り先活用形	6
14	係り先読点の有無	2
15	係り先「は」の有無	2
16	係り先引用「と」の有無	2
17	文節間の読点数	4
18	文節間「は」	3
主辞の単語に関する素性		
40	係り元の単語	13,557 以上
41	係り元に一番近い文節の単語	10,818 以上
42	係り元に二番目に近い文節の単語	8,503 以上
43	係り元から最も遠い文節の単語	3,212 以上

第 6 章

実験

6.1 実験環境

EDR 日本語コーパス（EDR 1996）の 208,157 文のうち, 101,540 文を学習に用い, テストには, 2603 文を用いた. 実験に用いた Kernel 関数は, 日本語係り受け解析に適している式 4.7 の polynomial 関数を用い, 次数 d は, 工藤の実験でも日本語係り受け解析には最適とされた, $d = 3$ を適用した.

前章でも述べた通り, 係り先文節候補が *2* 文節である場合のモデルである ”*3* つ組モデル ”用の *SVM* モデルを一つ作成する. 候補が *3* 文節である場合のモデルである ”*4* つ組モデル ”を, *one vs. rest* 法を用いた多値分類用 *SVM* モデルを三つ作成する. モデル作成には, 学習コーパス中の文を, *SLUNG* で解析したのち, 係り先文節候補が *2* 文節である係り元文節には, 係り元文節の属性と係り先文節候補の *2* 文節の属性を一つのデータとして ”*3* つ組モデル ”を構成する. 係り先文節候補が *3* 文節以上ある係り元文節は, *3.3* 節で述べてた方法により *3* 文節までに制限し、係り元文節の属性と係り先文節候補の *3* 文節の属性を一つのデータとして ”*4* つ組モデル ”を構成する. 本実験では, これらのデータを, 工藤 *[8]* が作成した *SVM* のツールである *TinySVM* を使用して, 各文節の係り受け確率を推定する.

6.2 実験結果

EDR 日本語コーパスに対する, 文節正解率を表 *6.1* に示す.

表 6.1 EDR 日本語コーパスに対する文節正解率

提案モデル	文節正解率	87.72% (18,139/20,618)
(101,540 文)	(後ろから 2 番目の文節を除く場合)	85.99% (15,587/18,126)
従来のモデル	文節正解率	88.55% (23,078/26,062)
(192,778 文)	(後ろから 2 番目の文節を除く場合)	88.55% (23,078/26,062)

表 *6.1* には, 金山ら *[2]* が作成した, *ME* 法を用いて係り受け確率を推定している *3* つ組／ *4* つ組モデルで求められた文節正解率と, 提案モデルである, *ME* 法に変わって *SVM* を用いることにより係り受け確率を推定し, 係り受け解析の精度の向上を目指した *3* つ組／ *4* つ組モデルで求められた文節正解率の中で最も高い確率が示されている.

提案モデル, 従来のモデル共に, *EDR* コーパスを用いて学習を行なっている. 表 *6.1* では, 従来のモデルは *192,778* 文, 提案モデルでは *101,540* 文の学習コーパスを使用しており, 従来のモデルが提案モデルよりも高い精度を誇っている. しかしながら, 金山ら *citekanayama* の実験によると, 約 *10* 万文の学習コーパスによる学習でも従来のモデルは, *88.0%*以上の精度がでている. 提案モデルは, 従来のモデルの精度を越えることができなかった.

図 *6.1* に, 学習コーパスの量を変化させたときの文節正解率を示す.

図 *6.1* より文節正解率は, 学習コーパスの量に比例していることが判る. 本実験では, *EDR* コーパスの約 *10* 万文使用したが文節正解率は収束しておらず, 学習コーパス量を増加することによって, 精度の向上が見込める.

しかしながら, 学習コーパス量が *10* 万文を越えると *SVM* において学習に要する時間が, 学習コーパス量が増加するに従って爆発的に増大し現実的な時間内で学習する事ができなかった.

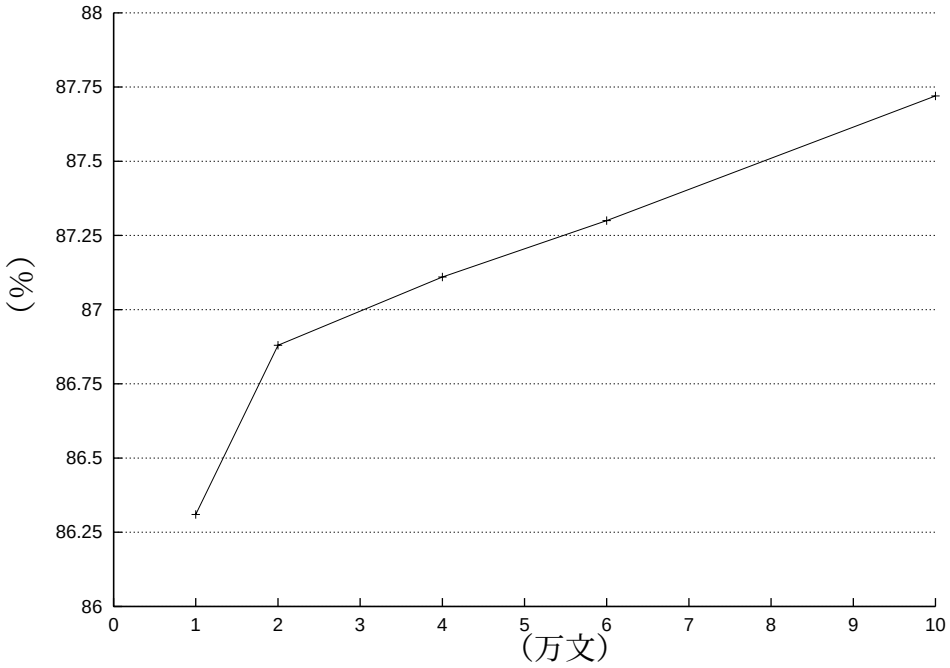

図 6.1 学習コーパスの量と文節正解率との関係

第 7 章

考察

7.1 「3 つ組／4 つ組モデル」と SVM との併用について

本節では, 金山ら *[2]* が提案した「*3* つ組／*4* つ組モデル」の係り受け確率推定アルゴリズムに *SVM* を利用する利点について述べる.

現在, 日本語係り受け解析において, 決定木や最大エントロピー（*ME*）法などの統計的学習モデルや, *Support Vector Machine*（*SVM*）や *Boosting* 等の統計的機械学習モデルに基づく様々な手法が提案されている.

これらの学習モデルは, それぞれ単独で高い精度を出している. 工藤ら *[3]* の *SVM* を使用した日本語係り受け解析は, 非常に少ない学習データにもかかわらず, 文節正解率が *89.09%* という高い解析精度が報告されている.

しかしながら, この係り受け解析手法は，日本語係り受け関係の制約をうまく利用し，文末からのビームサーチをしながら解析する手法である．この手法では，最適なビーム幅の設定方法が確立されておらず，さらに，係り元文節と係り先文節候補の素性と，一部の他文節の素性情報のみで係り受け解析が行われている.

そこで，ビームサーチの換わりに「*3* つ組／*4* つ組モデル」を利用する上での利点である ”係り先文節候補が，係り元文節から一番近い文節・二番目に近い文節，係り元から最も遠い文節の *3* 文節に係る確率が高い ”という，ヒューリスティクスに基づく方法を使い，さらに,「*3* つ組／*4* つ組モデル」は，係り元文節と係り先文節候補の *3* 文節を同時に考慮する確率計算を行う．この計算で使う素性集合を *SVM* に与え学習することによって,「*3* つ組／*4* つ組モデル」の多くの素性を扱う必要が有り，従来の「*3* つ組／*4*

つ組モデル」の係り受け確率推定に使用していた ME 法の人手により素性の組み合わせを適切に選択しないと過学習を起こしてしまう問題点を，SVM を使うことにより解決することができる．

このように,「3 つ組／ 4 つ組モデル」と SVM を併用することにより相互の問題を解決できるという点で「3 つ組／ 4 つ組モデル」と SVM の併用は有効であるといえる．

7.2 kernel 関数の次数と解析精度

$10{,}170$ 文の学習データと,$Kernel$ 関数に $polynomial$ 関数を用いた時の次元数 d を $d = 2, 3, 4, 5$ と変化させたときの文節正解率を表 7.1 に示す．結果として，次元数 $d = 3, 4$ の時に精度が良かった．この結果はある意味，直感的に合致している．

何故ならば，本手法のベースである「3 つ組／ 4 つ組モデル」では，ME 法による係り受け確率推定のために，素性の組み合わせを明示的に示している．その組み合わせの大半は，2 組，3 組の素性の組み合わせであり一部 5 組の素性の組み合わせがある．本実験で使用した $kernel$ 関数である $Polynomial$ 関数は一般に, 次数が d 次である時, d 個までの組み合わせを含めた学習モデルとなることが知られている．

よって結果的に，$Polynomial$ 関数の次数を $d = 3, 4$ のときに，従来の「3 つ組／ 4 つ組モデル」により明示的に示された素性の組み合わせを網羅することができ，他の次数より良い精度がでたと考えられる．しかしながら，次数が $d = 5$ の場合，現実的にあり得ない素性の組み合わせまで学習してしまい次数 $d = 3, 4$ より精度が低下したと考えられる．

表 7.1 Polynomial 関数の次元数変化による文節正解率の変化

次元数	文節正解率
2	86.00%
3	86.31%
4	86.32%
5	86.19%

第 8 章

まとめ

本論文では, 文法を用いて係り受け解析をする統計モデルの改良について論じた. 実験には, 金山らが３つ組／４つ組モデルを構築したときに使用した素性のうち, 明示的に素性の組み合わせを与えられているものを除いた全ての素性と, 各文節（係り元文節, 係り元文節に一番近い文節, 二番目に近い文節, 最も遠い文節）毎に、主辞として学習コーパス中に現われた単語全てをリスト化し, SVM の素性として与えた. 学習およびテストには EDR コーパスを用い, それぞれ $101{,}540$ 文, $2{,}603$ 文を与えたが, 得られた精度は最大 87.7%であり，金山らが ME 法を使って達成した性能には及ばなかった.

金山らが ME 法を使って達成した性能には及ばなかった理由として以下の事柄を挙げることができる.

- 我々の学習コーパス量が $101{,}540$ 文であるのに対して, 金山らが使用した学習コーパス量は, 約倍の $192{,}778$ 文である点である. 統計モデルを使用した構文解析器は, 学習コーパス量が多ければ多い程に高い学習精度が望めるため, 金山らの達成した性能には及ばなかったと考えられる. 更に, 金山らと同じ学習コーパス量で学習を行なうと, 学習が現実的な時間で終了しないという問題も影響している.

- 3 つ組／ 4 つ組モデルの素性空間は, 工藤らの素性空間よりはるかに複雑である可能性がある. したがって, 工藤らが行なった以上の繊細なチューニングを、素性や $Kernel$ 関数に対して行なう必要がある可能性がある.

第 9 章

今後の課題

係り受け解析精度の向上する上で最も単純で効果的な方法は，学習コーパス量を増加することである．しかしながら，学習コーパス中に存在するすべての係り受け関係を用いるため本手法は，多くの計算量を必要としている．実際，本稿の実験で使用した学習コーパス量は *101,540* 文で，*EDR* コーパスの約半分である．*EDR* コーパスすべてを学習コーパスとした場合，*SVM* によるモデルの学習に一ヶ月以上かけても収束せず現実的に時間内での学習が行えなかった．

そこで，係り受け解析精度の向上を図るために学習コーパス量を増加するのではなく，素性を追加することを考え，鳥澤 *[7]* が提案している単語の意味クラスを導入することを考えている．

単語の意味クラスとは，鳥澤 *[7]* が提案した *Expectation Maximization(EM)* をベースとした教師なし学習で生成したものである．*EM* は，学習コーパス中のデータにより推定した推定確率を，それらの相対確率で重み付けしたものを繰り返し計算をする．この鳥澤 *[7]* の提案した手法を利用することで，人が文章を読む上で暗黙の内に前提となっている意味的要素を単語の意味クラスにより表現できる利点がある．これを本研究で素性として使用していた主辞の単語を元に，係り元文節と各係り受け文節候補間の意味的要素を，素性として反映させることによって，係り受け解析の精度が向上が見込めるのではと考えている．

謝辞

主テーマ指導教官の鳥澤健太郎助教授には, 研究の全般において多大なご意見, ご助言を頂きました. 心より御礼申し上げます. 東条敏教授には多くのご意見を頂きました. 深く感謝いたします. 永田祐一助手, 山田寛康助手には多くのご助言を頂きました. 深く感謝いたします. 知識工学講座の皆さんには, 公私に渡りとてもお世話になりました. ありがとうございます.

参考文献

[1] *C.Cortes and V.Vapnik. Support Vector Networks. Machine Learning, Vol.20, pp.273-297, 1995.*

[2] 金山博，鳥澤健太郎，光石豊，辻井潤一．３つ以下の候補からかかり先を選択する係り受け解析モデル．自然言語処理，*Vol.5 No.5 pp71-91. Nov,2000.*

[3] 工藤拓，松本裕治．*Support Vector Machine*による日本語係り受け解析．*SIG-NL-128. 2000.*

[4] 内元清貴, 関根聡, 井佐原均. *ME*による日本語係り受け解析. 自然言語処理, *1998*

[5] *CHRISTOPHER J.C.BURGES. A Tutorial on Support Vector Machines for Pattern Recognition*

[6] 山田 寛康, 松本裕治．*Support Vector Machine*の多値分類問題への適用について. 自然言語処理, *No.146-006, 2001.11.20*

[7] *Kentaro Torisawa. An Unsupervised Method for Canonicalization of Japanese Postpositions.in Proceedings of the 6th Natural Language Processing Pacific Rim Symposium (NLPRS 2001), pp. 211-218, December, 2001.*

[8] 工藤拓. *TinySVM. http://cactus.aist-nara.ac.jp/̃taku-ku/software/TinySVM/*

[9] *EDR(1996). "EDR(Japan Electronic Dictionary Research Institute, Ltd.) Electronic Dictiionary Version 1.5 Technical Guide.". Second edition is available via http://www.iijnet.or.jp/edr/E_TG.html.*

[10] *Thorsten Joachims. SVM light. http://ais.gmd.de/̃thorsten/svm_light/*

[11] 松本 裕治, 黒橋 禎夫, 妙木 裕, 新保 仁, 長尾 眞, "利用者定義可能な日本語形態素解析システム *JUMAN* 使用説明書", 京都大学工学部長尾研究室, *1991.*

[12] Pollard, C. and Sag, I. A. Head-Driven Phrase Structure Grammar. University of Chicago press. 1994

[13] 山田 寛康, 工藤 拓, 松本 裕治. *Support Vector Machine* を用いた日本語固有表抽出. 情報処理学会論文誌, *Vol.43*, *No.1*, *p44-53*, *2002.1*

[14] Mitsuishi, Y., Torisawa, K., and Tsujii, J. (1998). " HPSG-Style Underspecified Japanese Grammar with Wide Coverage."In Proc. COLING-ACL '98, pp.876-880.

[15] Takaki Makino, Kentaro Torisawa, and Jun-ichi Tsujii. LiLFeS - practical programming language for typed feature structures. In Proceedings of the 4th Natural Language Processing Pacific Rim Symposium, pp.239-244, Phuket, Thailand, 1997

[16] LiLFeS Home page http://www-tsujii.is.s.u-tokyo.ac.jp/lilfes/index.html