

(19) 日本国特許庁(JP)

## (12) 特許公報(B2)

(11) 特許番号

特許第6101650号  
(P6101650)

(45) 発行日 平成29年3月22日(2017.3.22)

(24) 登録日 平成29年3月3日(2017.3.3)

(51) Int.Cl.

G06N 99/00 (2010.01)

F 1

G06N 99/00 153

請求項の数 8 (全 29 頁)

(21) 出願番号 特願2014-37245 (P2014-37245)  
 (22) 出願日 平成26年2月27日 (2014. 2. 27)  
 (65) 公開番号 特開2015-162113 (P2015-162113A)  
 (43) 公開日 平成27年9月7日 (2015. 9. 7)  
 審査請求日 平成28年1月13日 (2016. 1. 13)

(73) 特許権者 000004226  
 日本電信電話株式会社  
 東京都千代田区大手町一丁目5番1号  
 (74) 代理人 110001519  
 特許業務法人太陽国際特許事務所  
 (72) 発明者 鈴木 潤  
 東京都千代田区大手町一丁目5番1号 日本電信電話株式会社内

審査官 多胡 滋

(56) 参考文献 特開2012-256198 (JP, A)

最終頁に続く

(54) 【発明の名称】システムパラメタ学習装置、情報処理装置、方法、及びプログラム

## (57) 【特許請求の範囲】

## 【請求項 1】

入力データに対する出力データのスコアを決定するためのスコア関数を用いて、入力データに対して所定の情報処理を行って出力データを出力する情報処理システムにおいて設定される、複数のシステムパラメタを学習するシステムパラメタ学習装置であって、

複数の入力データの各々と前記複数の入力データに対する複数の正解出力データの各々とのペアである教師データを受け付ける教師データ入力部と、

前記教師データ入力部により受け付けた前記教師データと、前記スコア関数とに基づいて、前記複数のシステムパラメタの各々の値が、所定個の実数値  $v_i$  ( $i = 1, \dots$ ) と実数値  $-v_i$  ( $i = 1, \dots$ ) と 0 とからなる離散値の集合に含まれる制約を満たし、かつ、最適化された前記複数のシステムパラメタを学習する学習部とを含む、システムパラメタ学習装置。

## 【請求項 2】

前記学習部は、以下の(1)式に従って、前記最適化された  $N$  個のシステムパラメタ  $w^i$  を学習する請求項 1 記載のシステムパラメタ学習装置。

【数1】

$$w^* = \arg \min_{w^*} \{O(w^*; \mathcal{D})\}, \quad O(w^*; \mathcal{D}) = \mathcal{L}(\Phi(\cdot), w^*, \mathcal{D}) + \Omega(w^*), \quad s.t. \quad w^* \in \mathcal{S}_\zeta^N$$

• • • (1)

$$\mathcal{S}_\zeta = \{yv | (v, y) \in \mathcal{R}_\zeta \times \sigma\} \cup \{0\}$$

$$\mathcal{R}_\zeta = \bigcup_{i=1}^{\zeta} \{v_i\}$$

$$\sigma = \{-1, 1\}$$

10

ただし、

【数2】

$$\mathcal{L}(\Phi(\cdot), w^*, \mathcal{D})$$

は、前記教師データ

【数3】

$$\mathcal{D}$$

20

の入力データ  $x_i$  に対してシステムパラメタ  $w^*$  を用いて前記所定の情報処理を行ったときに得られる出力データが、前記教師データ

【数4】

$$\mathcal{D}$$

の正解出力データ  $y_i$  とどの程度間違っているかを示す値を返すリスク関数であり、(  $w^*$  ) は、正則化項であり、

30

【数5】

$$\mathcal{S}_\zeta^N$$

は、前記  $N$  個のシステムパラメタの各々についての前記離散値の直積集合である。

【請求項3】

前記学習部は、以下の(2)式に従って、前記最適化された  $N$  個のシステムパラメタ  $w^*$  を学習する請求項2記載のシステムパラメタ学習装置。

【数6】

40

$$w^* = \arg \max_{\alpha^*} \arg \min_{w^*, u^*} \{O(w^*, u^*, \alpha^*; \mathcal{D})\},$$

• • • (2)

$$O(w^*, u^*, \alpha^*; \mathcal{D}) = \mathcal{L}(\Phi(\cdot), w^*, \mathcal{D}) + \Omega(w^*) + \frac{\rho}{2} \|w^* - u^* + \alpha^*\|_2^2 - \frac{\rho}{2} \|\alpha^*\|_2^2$$

$$s.t. \quad u^* \in \mathcal{S}_\zeta^N$$

ただし、 $\alpha^*$  は、ラグランジュ未定乗数であり、 $w^*$  は、予め定められたチューニングパラ

50

メタであり、 $u^\wedge$  は、補助パラメタである。

【請求項 4】

前記学習部によって学習された前記複数のシステムパラメタに基づいて、値が同一となるシステムパラメタでグループを構成し、各グループにインデックス番号を付与し、各グループについて、前記グループを構成するシステムパラメタの各々を、前記グループに付与された前記インデックス番号のシステムパラメタに変換し、前記インデックス番号のシステムパラメタを保存する重複パラメタ圧縮部を更に含む請求項 1 ~ 3 の何れか 1 項記載のシステムパラメタ学習装置。

【請求項 5】

前記複数のシステムパラメタは、入力データ  $x^\wedge$  及び出力データ  $y^\wedge$  に対する複数の素性抽出関数  $f_d(x^\wedge, y^\wedge)$  の重みであって、

前記重複パラメタ圧縮部は、各グループについて、前記グループを構成するシステムパラメタの各々を、前記グループに付与された前記インデックス番号のシステムパラメタに変換し、前記インデックス番号のシステムパラメタを保存すると共に、以下の (3) 式に示すように、前記スコア関数を定義して保存する請求項 4 記載のシステムパラメタ学習装置。

【数 7】

$$\Phi(x^\wedge, y^\wedge; w^\wedge) = w^\wedge \cdot f^\wedge(x^\wedge, y^\wedge) = \sum_{d=1}^N w_d f_d(x^\wedge, y^\wedge) = \sum_{k=1}^K v_k g_k(x^\wedge, y^\wedge) \quad \dots \quad (3)$$

$$g_k(x^\wedge, y^\wedge) = \sum_{i \in G_k} f_i(x^\wedge, y^\wedge)$$

ただし、 $(x^\wedge, y^\wedge; w^\wedge)$  はスコア関数であり、 $v_k$  は、グループに付与されたインデックス番号  $k$  のシステムパラメタである。

【請求項 6】

入力データを受け付ける入力部と、

前記スコア関数と、請求項 4 又は 5 記載のシステムパラメタ学習装置によって保存された各グループの前記インデックス番号のシステムパラメタとに基づいて、前記入力部において受け付けた入力データに対して、前記所定の情報処理を行って出力データを出力する情報処理部と、

を含む、情報処理装置。

【請求項 7】

教師データ入力部と、学習部とを含む、入力データに対する出力データのスコアを決定するためのスコア関数を用いて、入力データに対して所定の情報処理を行って出力データを出力する情報処理システムにおいて設定される、複数のシステムパラメタを学習するシステムパラメタ学習装置におけるシステムパラメタ学習方法であって、

前記教師データ入力部が、複数の入力データの各々と前記複数の入力データに対する複数の正解出力データの各々とのペアである教師データを受け付け、

前記学習部が、前記教師データ入力部により受け付けた前記教師データと、前記スコア関数とに基づいて、前記複数のシステムパラメタの各々の値が、所定個の実数値  $v_i$  ( $i = 1, \dots$ ) と実数値  $-v_i$  ( $i = 1, \dots$ ) と 0 とからなる離散値の集合に含まれる制約を満たし、かつ、最適化された前記複数のシステムパラメタを学習するシステムパラメタ学習方法。

【請求項 8】

コンピュータを、請求項 1 から請求項 5 の何れか 1 項記載のシステムパラメタ学習装置又は請求項 6 記載の情報処理装置を構成する各部として機能させるためのプログラム。

【発明の詳細な説明】

10

20

30

40

50

**【技術分野】****【0001】**

本発明は、システムパラメタ学習装置、情報処理装置、方法、及びプログラムに係り、特に、システムパラメタを学習するシステムパラメタ学習装置、情報処理装置、方法、及びプログラムに関する。

**【背景技術】****【0002】**

図13に示すような、音声認識、機械翻訳、文字認識、物体認識、DNAの構造予測などといった情報処理における識別問題は、図14に示すように、入力が与えられたときに、出力を予測するシステムとみなすことができる。

10

**【0003】**

これらのシステムは一般的に、実行フェーズと構築フェーズにわけることができる。構築フェーズとは、人手により事前にシステムを設計し、システムパラメタ等を決定する作業を指す。実行フェーズとは、構築フェーズで定義された設計に基づき入力を処理し、出力はシステムパラメタに依存して決定される。

**【0004】**

構築フェーズでは、様々な方法でシステムを構築することができる。例えば、人手により変換規則を記述しておいて、その規則に則って入力を出力へ変換し、それを出力する方法が考えられる。ただし、変換規則を人手により準備するのは網羅性や整合性を保持するためのコストが非常にかかるため、図15に示すように、データから自動的にシステムを構築する機械学習手法を用いてシステムを自動構築する方法を用いるのが近年では主流である。

20

**【0005】**

構築フェーズでは、まず、対象とするシステムの入力とそれに対応する出力のペアを用意する。これは、一般的に、正解データ或いは教師データとよばれる。教師データとは、教師データ中の入力がシステムに入力された際に、どのような出力がされるべきかを表したデータである。次に、この教師データを用いてシステムを構築する。必要な要件は、教師データ中の入力に対して、正しい出力が行えるシステムであることである。そこで、機械学習に基づく構築フェーズでは、教師データを用いて、教師データを正しく判別できるようなシステムパラメタの集合を学習することに帰着する。

30

**【0006】**

以上の処理を数式的に表すと以下のようになる。まず、実行フェーズを示す。 $x^{\wedge}$ を一つの入力を表すこととし、 $\mathbb{X}$ を、システムが受け付けるとり得る全ての入力 $x^{\wedge}$ の集合とする。なお、記号に付された「 $\wedge$ 」は、当該記号が行列、多次元配列、又はベクトルであることを表している。同様に、 $y^{\wedge}$ を一つの出力を表すこととし、 $\mathbb{Y}$ を、システムが許容するとり得る全ての出力 $y^{\wedge}$ の集合とする。また、 $\mathbb{Y}(x^{\wedge})$ を、 $x^{\wedge}$ が与えられたときにとり得る全ての出力 $y^{\wedge}$ の集合とする。よって、 $x^{\wedge} \rightarrow \mathbb{Y}(x^{\wedge}) \subseteq \mathbb{Y}$ の関係が成り立つ。

**【0007】**

次に、 $w^{\wedge}$ をシステムパラメタの集合をベクトル表記したものとする。ここで、 $w_d$ をベクトル $w^{\wedge}$ の $d$ 番目の要素であり、同時に $d$ 番目のシステムパラメタとする。つまり、 $w^{\wedge} = (w_1, \dots, w_N)$ かつ $d = \{1, \dots, N\}$ の関係が成り立つ。ただし、システムパラメタ数は $N$ であり、 $w^{\wedge}$ は $N$ 次元ベクトルとする。

40

**【0008】**

このとき、入力 $x^{\wedge}$ が与えられたときに出力 $y^{\wedge}$ を返すシステムを下記(1)式に表すことができる。

**【0009】**

## 【数1】

$${}^*y^{\wedge} = \underset{y^{\wedge} \in Y(x^{\wedge})}{\arg \max} \Phi(x^{\wedge}, y^{\wedge}; w^{\wedge})$$

• • • • (1)

## 【0010】

ただし、 $(x^{\wedge}, y^{\wedge}; w^{\wedge})$  は、 $x^{\wedge}$  から  $y^{\wedge}$  へ変換する際のスコアを決定する関数であり、ここでは、単にスコア関数と呼ぶ。つまり、 $x^{\wedge}$  が与えられた際に得られる可能性がある全ての出力  $y^{\wedge}$  の中で、この変換スコアが最も高くなる  $y^{\wedge}$  が output として採用されることになる。そのため、 $w^{\wedge}$  は、どの出力が選ばれるかを制御するシステムパラメタであり、システム全体の性能を決定する要因といえる。よって、システムパラメタ  $w^{\wedge}$  をいかに精度よく求めるかという事が、構築フェーズの最大の要件となる。ここで、精度よくとは、あらゆる入力に対して可能な限り多くの正しい出力を行うことが可能な  $w^{\wedge}$  を求めることを意味する。なお、記号の前に付された「 $*$ 」は、当該記号が推定された値であることを表している。

## 【0011】

次に、構築フェーズについて説明する。実際に、あらゆる可能な入力に対して最良のパラメタ  $w^{\wedge}$  を求めるることは非常に困難を伴う。それは、実際に、あらゆる可能な入力を列挙することが事実上困難であることに起因する。そこで、パターン認識の分野では、実データに基づいて  $w^{\wedge}$  を決定する。まず、教師データを

## 【0012】

## 【数2】

$\mathcal{D}$

## 【0013】

で表す。教師データは入力  $x^{\wedge}$ 、出力  $y^{\wedge}$  のペアの集合で構成される。つまり、

## 【0014】

## 【数3】

$$\mathcal{D} = \{(x^{\wedge}_i, y^{\wedge}_i)\}_{i=1}^N$$

## 【0015】

このとき、 $x^{\wedge}_i$  を、教師データ中の  $i$  番目の入力データとし、 $y^{\wedge}_i$  を  $i$  番目の入力に対応する出力とする。システムパラメタの学習は、下記(2)式の最適化問題を解くことで得られる。

## 【0016】

10

20

30

40

【数4】

$${}^*w^{\wedge} = \arg \min_{w^{\wedge}} \{ \mathcal{O}(w^{\wedge}; \mathcal{D}) \},$$

• • • (2)

ただし、

10

$$\mathcal{O}(w^{\wedge}; \mathcal{D}) = \mathcal{L}(\Phi(\quad), w^{\wedge}, \mathcal{D}) + \Omega(w^{\wedge}),$$

【0017】

このとき、

【0018】

【数5】

$$\mathcal{L}(\Phi(\quad), w^{\wedge}, \mathcal{D})$$

20

【0019】

は、リスク関数や損失関数とよばれ、教師データ内の入力に対してどの程度正しい出力を得られるかといった値を返す関数である。現在のパラメタ  $w^{\wedge}$  を用いて、実際に上記(1)式を用いて判別を行ってみて、より多く間違える場合には、より大きな値となるような関数を用いる。 $(w^{\wedge})$  は、一般に正則化項とよばれ、教師データが有限個しかない状況で、教師データに現れないデータに対してもより正しく判別できるように、システムパラメタが教師データに過適応しないように、ペナルティを与える項である。例えば、パラメタの  $L_2$ -ノルムがなるべく小さくなるような制約を課すことで、パラメタが極端に大きな値をとらないように制限するといったことが、よく用いられる。最終的に、上記(2)式で得られる  ${}^*w^{\wedge}$  は、教師データを最もよく識別することができるパラメタの集合といえる。

30

【0020】

以上が、本発明で対象とする情報処理システムの実行フェーズと構築フェーズを数式的に定義したものである。

【0021】

上記(2)式に基づいたシステムパラメタの獲得は、パターン認識では教師あり学習と呼ばれる。このとき、学習後のシステムパラメタ  ${}^*w^{\wedge}$  は、実数値で表される。よって、構築フェーズ終了時に  $w^{\wedge}$  の値をファイルなどに書き出しておき、実行フェーズでは、書き出したファイルを読み込んでスコア関数を計算し、出力を得る。つまり、パラメタ数が多くなるほど、その情報を保持するために必要なファイルサイズは大きくなる。

40

【0022】

ファイルサイズは、そのまま実行時のメモリ占有量と同じとなる。メモリ占有量は、携帯端末等の限定されたリソースしか持たない計算環境で、非常に大きな問題となる可能性がある。また、一般的な計算機上での実行時にも、近年のマルチコアな計算機上で同時に複数実行する際や、他のプログラムになるべく影響を与えないという観点で、メモリ占有量は極力少ないとされる。つまり本質的に、プログラム実行時の必要リソース量(ファイルサイズ、メモリ専有量など)は、どのような計算環境であれ少ないとされるがよりよい、ということがいえる。この考えに基づいて学習後のモデルサイズを、学習の工程で圧縮するという課題が近年盛んに取り組まれている。

【0023】

50

最も単純な方法として、 $L_1$  正則化項の効果を用いて学習後のモデル圧縮を行う方法がある（非特許文献1）。この方法の原理は、 $L_1$  正則化項が学習時にパラメタが極力ゼロとなるようにする効果があるため、パラメタがゼロになった場合は、そのパラメタに関わる項目をモデルから消去できることから、モデルサイズを削減できる。

#### 【0024】

また、モデル圧縮が可能な方法として、システムパラメタ学習後に複数のパラメタが同じ値を取れば、この重複情報を利用して、実際に保持しておかなくてはいけない情報量を減らすことが可能であるという原理を利用する方法がある（非特許文献2）。このように、情報処理システムの精度を極力落とさずに学習モデルの圧縮を行うことは、実用上非常に大きな課題であり、様々な工夫により改善がなされている。

10

#### 【先行技術文献】

#### 【非特許文献】

#### 【0025】

【非特許文献1】Lin Xiao. Dual Averaging Methods for Regularized Stochastic Learning and Online Optimization. *Journal of Machine Learning Research*, 11:2543-2596, 2010.

【非特許文献2】Jun Suzuki and Masaaki Nagata. Supervised Model Learning with Feature Grouping based on a Discrete Constraint. *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, 18-23, 2013.

#### 【発明の概要】

20

#### 【発明が解決しようとする課題】

#### 【0026】

しかし、非特許文献1記載の技術は、値をゼロにするのは、真にそのシステムパラメタに関わる項目に意味がない場合を除き、システムの精度を低下させる恐れがあるため、値をゼロにする効果とシステムの精度はトレードオフの関係になる。つまり、なるべくゼロを多くしてモデルサイズを圧縮したいが、あまりにゼロを多くしすぎる設定で学習を行うと、今度はシステム精度が不十分になるという問題点が存在する。

#### 【0027】

また、非特許文献2記載の技術は、非常に効果が高いことが示されているが、一方、実際にモデルを作成する際には、システムの精度を落とさずにどこまでモデルを圧縮可能かは自明ではなく、実際に試行錯誤的に試してみて、開発データなどで最も良い結果を最終的に選択するといった方法がとられる。実際にこの作業は、学習データ量が多くなればなるほど、非常に高コストになり、運用上問題が発生する。結果として、試行錯誤のコストを割くことが出来なかつたがゆえに、本来得られる可能性があった高精度かつ高圧縮なモデルを発見できず、結果として、高圧縮なモデルを用いることができなかつたり、逆に、精度を落とす結果となつたりすることがしばしばおこるという問題点がある。

30

#### 【0028】

本発明では、上記問題点を解決するために成されたものであり、自動で適切な高圧縮モデルを獲得することができるシステムパラメタ学習装置、方法、及びプログラムを提供することを目的とする。

40

#### 【0029】

また、実行時に必要とされるリソースを削減することができる情報処理装置、及びプログラムを提供することを目的とする。

#### 【課題を解決するための手段】

#### 【0030】

上記目的を達成するために、第1の発明に係るシステムパラメタ学習装置は、入力データに対する出力データのスコアを決定するためのスコア関数を用いて、入力データに対して所定の情報処理を行つて出力データを出力する情報処理システムにおいて設定される、複数のシステムパラメタを学習するシステムパラメタ学習装置であつて、複数の入力データの各々と前記複数の入力データに対する複数の正解出力データの各々とのペアである教

50

師データを受け付ける教師データ入力部と、前記教師データ入力部により受け付けた前記教師データと、前記スコア関数とに基づいて、前記複数のシステムパラメタの各々の値が、所定個の実数値  $v_i$  ( $i = 1, \dots, n$ ) と実数値  $-v_i$  ( $i = 1, \dots, n$ ) と 0 とからなる離散値の集合に含まれる制約を満たし、かつ、最適化された前記複数のシステムパラメタを学習する学習部と、を含んで構成されている。

#### 【0031】

第2の発明に係るシステムパラメタ学習方法は、教師データ入力部と、学習部とを含む、入力データに対する出力データのスコアを決定するためのスコア関数を用いて、入力データに対して所定の情報処理を行って出力データを出力する情報処理システムにおいて設定される、複数のシステムパラメタを学習するシステムパラメタ学習装置におけるシステムパラメタ学習方法であって、前記教師データ入力部が、複数の入力データの各々と前記複数の入力データに対する複数の正解出力データの各々とのペアである教師データを受け付け、前記学習部が、前記教師データ入力部により受け付けた前記教師データと、前記スコア関数とに基づいて、前記複数のシステムパラメタの各々の値が、前記所定個の実数値  $v_i$  ( $i = 1, \dots, n$ ) と実数値  $-v_i$  ( $i = 1, \dots, n$ ) と 0 とからなる離散値の集合に含まれる制約を満たし、かつ、最適化された前記複数のシステムパラメタを学習する。

10

#### 【0032】

第1及び第2の発明によれば、教師データ入力部により、複数の入力データの各々と複数の入力データに対する複数の正解出力データの各々とのペアである教師データを受け付け、学習部により、受け付けた教師データと、スコア関数とに基づいて、複数のシステムパラメタの各々の値が、所定個の実数値  $v_i$  ( $i = 1, \dots, n$ ) と実数値  $-v_i$  ( $i = 1, \dots, n$ ) と 0 とからなる離散値の集合に含まれる制約を満たし、かつ、最適化された複数のシステムパラメタを学習する。

20

#### 【0033】

このように、複数の入力データの各々と複数の入力データに対する複数の正解出力データの各々とのペアである教師データを受け付け、受け付けた教師データと、スコア関数とに基づいて、複数のシステムパラメタの各々の値が、所定個の実数値  $v_i$  と実数値  $-v_i$  と 0 とからなる離散値の集合に含まれる制約を満たし、かつ、最適化された複数のシステムパラメタを学習することによって、自動で適切な高压縮モデルを獲得することができる。

30

#### 【0034】

第3の発明に係る情報処理装置は、入力データを受け付ける入力部と、前記スコア関数と、第1の発明のシステムパラメタ学習装置によって保存された各グループの前記インデックス番号のシステムパラメタとに基づいて、前記入力部において受け付けた入力データに対して、前記所定の情報処理を行って出力データを出力する情報処理部と、を含んで構成されている。

#### 【0035】

また、本発明のプログラムは、コンピュータを、上記のシステムパラメタ学習装置又は情報処理装置を構成する各部として機能させるためのプログラムである。

#### 【発明の効果】

40

#### 【0036】

以上説明したように、本発明のパラメタ学習装置、方法、及びプログラムによれば、複数の入力データの各々と複数の入力データに対する複数の正解出力データの各々とのペアである教師データを受け付け、受け付けた教師データと、スコア関数とに基づいて、複数のシステムパラメタの各々の値が、所定個の実数値  $v_i$  と実数値  $-v_i$  と 0 とからなる離散値の集合に含まれる制約を満たし、かつ、最適化された複数のシステムパラメタを学習することによって、自動で適切な高压縮モデルを獲得することができる。

#### 【0037】

また、本発明の情報処理装置によれば、システムパラメタ学習装置によって保存された各グループのインデックス番号のシステムパラメタとに基づいて、入力データに対して、所

50

定の情報処理を行って出力データを出力することにより、実行時に必要とされるリソースを削減することができる。

【図面の簡単な説明】

【0038】

【図1】本発明の実施の形態を適用する問題の例を示す図である。

【図2】素性とシステムの定義づけの一例を説明するための概念図である。

【図3】本発明の実施の形態に係るシステムパラメタ学習装置の構成を示すブロック図である。

【図4】素性抽出関数の一例を説明するための第1の概念図である。

【図5】素性抽出関数の一例を説明するための第2の概念図である。

【図6】教師データに基づいて、システムパラメタ値の学習処理を説明するための概念図である。

【図7】超平面への写像問題の例を示す図である。

【図8】一次元のk-meansと等価の処理の例を示す図である。

【図9】本発明の実施の形態に係る情報処理装置の構成を示すブロック図である。

【図10】本発明の実施の形態に係るシステムパラメタ学習装置におけるシステムパラメタ学習処理ルーチンの内容を示すフローチャートである。

【図11】本発明の実施の形態に係る情報処理装置における情報処理ルーチンの内容を示すフローチャートである。

【図12】本実施の形態を用いた場合における実験結果を示す図である。

10

【図13】従来技術の概要を説明するための第1の説明図である。

【図14】従来技術の概要を説明するための第2の説明図である。

【図15】従来技術の概要を説明するための第3の説明図である。

【発明を実施するための形態】

【0039】

以下、図面を参照して本発明の実施の形態を詳細に説明する。

【0040】

<本発明の原理>

【0041】

本発明に係る実施の形態は、非特許文献2によるモデル圧縮の枠組みに、学習データから自動で高精度かつ高圧縮モデルを構築する処理装置を追加する。

20

【0042】

まず、非特許文献2のモデル圧縮処理装置を持った機械学習装置の仕組みを説明する。基本的な考えは、システムパラメタ学習後に複数のパラメタが同じ値をとれば、この重複情報を利用して、実際に保持しておかなくてはいけない情報量を減らすことが可能であるという原理を利用している。例えば、パラメタ数が5の時に、上記(2)式で最終的に得られたパラメタ集合が ${}^*w^* = (0.3, 0.3, -0.6, -0.6, 1.0)$ だったとする。すると、 ${}^*w^*$ では、0.3と-0.6が2回重複して出現するため、1番目と2番目のパラメタを合わせて、また、3番目と4番目のパラメタを合わせて、 ${}^*w^* = (0.3, -0.6, 1.0)$ という3つの情報が最低限あれば同じ結果が得られる。このように、重複した値が多ければ多いほど等価な情報を情報量を減らして保持することが可能であることがわかる。

40

【0043】

つまり、上記(2)式のシステムパラメタの学習を行う際に、「システムパラメタの値がなるべく多く同じ値をとるようにする」という制約を追加して、システムパラメタの学習を行うことで、結果として得られるシステムパラメタの値の重複がより多くなるようにする。

【0044】

そして、次に、得られたパラメタに対して、同じ値を持つパラメタを一つにまとめることで、保持すべき情報量を削減する。最後に、削減したパラメタを用いて、実際にシス

50

ムを実行する。まとめると、従来の方法は、大きく分けて以下の1～3の3つの処理で構成される。

【0045】

1. 第1の処理として、通常のシステムパラメタ学習時に最終的に得られるパラメタの値をなるべく同じになるようなシステムパラメタの獲得処理。

2. 第2の処理として、得られたシステムパラメタの重複する値を一つにまとめて保持しておくべき情報量を最小限にするシステムパラメタの圧縮処理。

3. 第3の処理として、最小限に圧縮したシステムパラメタを用いて、システムを実行フェーズで動作させる処理。

【0046】

より具体的には、第1の処理は、上記(2)式のシステムパラメタ学習問題を、下記(3)式の問題に置き換える。

【0047】

【数6】

$$w^* = \arg \min_{w^*} \{ \mathcal{O}(w^*; \mathcal{D}) \}, \quad \mathcal{O}(w^*; \mathcal{D}) = \mathcal{L}(\Phi(\cdot), w^*, \mathcal{D}) + \Omega(w^*), \quad s.t. \quad w^* \in \mathcal{S}^N$$

◆◆◆(3)

【0048】

上記(2)式と上記(3)式との違いは、単純に制約項

【0049】

【数7】

$$w^* \in \mathcal{S}^N$$

【0050】

が増えただけである。この制約は、システムパラメタ  $w^*$  がある任意の離散集合

【0051】

【数8】

$$\mathcal{S}^N$$

【0052】

の要素となる場合にだけ解として認めるということを意味している。つまり、上記(2)式と同等の最適化問題を解くが、解は制約を満たしている必要がある。この制約は、パラメタの値をなるべく重複してとるように設計するために、有限個の離散値

【0053】

【数9】

$$\mathcal{S}$$

【0054】

の直積集合

【0055】

10

20

30

40

【数10】

$\mathcal{S}^N$

【0056】

で構成する。

【0057】

第2の処理として、同じ値になったパラメタを一つにまとめる。これは、構築フェーズの後処理的な位置づけになる。これは、例えば、 $w_i = w_j = w_k$  となったと仮定すると、つまり、 $i, j, k$  番目のシステムパラメタ（パラメタベクトルの要素）が同じ値となつたとき、 $w_i, w_j, w_k$  を削除し、新たに $w_1$  とするといった処理に相当する。ただし、新たに追加した $w_1$  は $w_i$  などと同じ値であり、インデックス $i, j, k$  はインデックス1に新たに振りなおされたとみなすことと等価である。このように、同じ値を保持していても冗長な情報なので、それらを一つにまとめて新たに割り振ったインデックスを、元の値のインデックスが指すことで、同じ情報を得られるようにする。こうすることで、従来と同じ形式のシステムパラメタでありながら、システムパラメタの値が重複した分を減らした、システムパラメタの集合を獲得することができる。

【0058】

第3の処理は、得られたシステムパラメタを用いて実際にシステムを動作させる。この処理は、第2の処理で、従来と同じ形式のシステムパラメタを得るようにすることで、処理としては、従来と同等の上記（1）式が使える。

【0059】

本発明に係る実施の形態は、主に上記第1の処理を変更する。従来の方法では、制約

【0060】

【数11】

$w^N \in \mathcal{S}^N$

【0061】

に現れる集合

【0062】

【数12】

$\mathcal{S}^N$

【0063】

は手動で決める必要がある。つまり、この集合の定義がシステム精度に大きく影響を与える。また、この定義は、データや対象タスクによって変わるものであり、事前に最適な定義を与えることは困難である。場合によっては、人間の事前知識等により、高精度なモデルを構築可能な定義を導出できる場合もありえるが、一般論としては、非常に難しくセンシティブな作業となるため、人手コスト等の面で大きな課題となる。本発明に係る実施の形態では、この定義を決める部分も最適化問題に含めて考えることで、データから自動的に決定する。

【0064】

具体的には、制約

【0065】

【数13】

$$w^{\wedge} \in \mathcal{S}^N$$

【0066】

を、制約

【0067】

【数14】

$$w^{\wedge} \in \mathcal{S}_{\zeta}^N$$

10

【0068】

に変更し、上記(3)式を、下記(4)式のように再定義する。

【0069】

【数15】

$${}^*w^{\wedge} = \arg \min_{w^{\wedge}} \{\mathcal{O}(w^{\wedge}; \mathcal{D})\}, \quad \mathcal{O}(w^{\wedge}; \mathcal{D}) = \mathcal{L}(\Phi(\quad), w^{\wedge}, \mathcal{D}) + \Omega(w^{\wedge}), \quad s.t. \quad w^{\wedge} \in \mathcal{S}_{\zeta}^N$$

· · · · (4)

20

$$\mathcal{S}_{\zeta} = \{yv | (v, y) \in \mathcal{R}_{\zeta} \times \sigma\} \cup \{0\}$$

$$\mathcal{R}_{\zeta} = \bigcup_{i=1}^{\zeta} \{v_i\}$$

$$\sigma = \{-1, 1\}$$

【0070】

上記(4)式の意味は、制約として取り得る値は、0と $\pm v_i$ のみとなり、 $v_i$ は、種類の実数値となる。例えば、 $\zeta = 4$ を予め指定した場合は、モデル内のパラメタ各 $w^{\wedge}$ の取り得る値は零を含めて9種類となる。また、このときの取り得る値の制約 $v_i$ もデータから自動に決定される。

30

【0071】

以下では、図1に示すような、自然言語処理の固有表現抽出と係り受け解析に対する問題について、本発明の実施の形態を適用した場合を想定して説明する。これらの問題は、構造予測問題と呼ばれ、グラフ構造などに変換されたものを入力として受け取り、同じくグラフ構造で表されるものを出力とする問題とみなすことができる。

【0072】

以下で説明する、構築及び実行フェーズでは、図2に示すように、最初にそれぞれの入力と出力の特徴付ける素性の定義を人手にて与える。ここでは、素性は関数として定義することを想定するため、素性抽出関数の集合を定義することと等価である。また、ある入力が与えられたときに、どのような出力がなされるかといった、システムの動作定義も人手にて与える。これは、実際に解く問題、ここでは固有表現抽出や係り受け解析、の問題の定義にしたがって自動的に決まるものである。

40

【0073】

&lt;システムパラメタ学習装置のシステム構成&gt;

次に、本発明の実施の形態に係るシステムパラメタ学習装置の構成について説明する。図3に示すように、本発明の実施の形態に係るシステムパラメタ学習装置100は、CPUと、RAMと、後述するシステムパラメタ学習処理ルーチンを実行するためのプログラムや各種データを記憶したROMと、を含むコンピュータで構成することが出来る。このシステムパラメタ学習装置100は、機能的には図3に示すように教師データ入力部10

50

と、演算部 20 と、システムパラメタ記憶部 90 とを備えている。

【0074】

教師データ入力部 10 は、教師データの入力を受け付ける。ここで、教師データは、上記図 1 で示したように、予め定められた入力データと、当該入力データに対する正解出力データとの複数ペアである。また、教師データ入力部 10 は、後述する学習部 40 で用いるチューニングパラメタ の入力を受け付ける。なお、パラメタ は、人手によって予め与えられる。

【0075】

演算部 20 は、教師データベース 30 と、学習部 40 と、重複パラメタ圧縮部 60 とを備えている。

10

【0076】

教師データベース 30 には、教師データ入力部 10 により受け付けた教師データが格納される。

【0077】

学習部 40 は、教師データベース 30 に格納された教師データと、予め定められたスコア関数とに基づいて、複数のシステムパラメタを学習する。具体的には、学習部 40 は、教師データとスコア関数とに基づいて、複数のシステムパラメタの各々の値が、個数 が予め定められた実数値  $v_i$  ( $i = 1, \dots, n$ ) と実数値  $-v_i$  ( $i = 1, \dots, n$ ) と 0 とからなる離散値の集合に含まれる制約を満たし、かつ、最適化された複数のシステムパラメタを学習する。学習部 40 は、初期化部 50、システムパラメタ更新部 52、補助パラメタ更新部 54、未定乗数更新部 56、及び収束判定部 58 を備えている。

20

【0078】

学習部 40 における学習アルゴリズムでは、目的関数を最小化するシステムパラメタ  $w^ = (w_1, w_2, \dots, w_N)$  を学習する。具体的には、正解データと教師あり学習アルゴリズムを人手により決定し、それを構築フェーズの設定として与える。以下では、システムパラメタ  $w^$  を学習するための目的関数について最初に説明する。

【0079】

ここでは、素性を抽出するための素性抽出関数  $f$  を、入力データ  $x^$  と出力データ  $y^$  の組み合わせで定義される関数とする。個々の素性抽出関数は、 $f_d(x^, y^)$  の形式の関数であり、任意の実数値を返す関数である。ここで、 $w^$  と同様に、素性抽出関数の集合をベクトル表記  $f^ = (x^, y^)$  で表す。このとき、 $f_d(x^, y^)$  は、 $f^$  の  $d$  番目の要素を表す。素性抽出関数によって抽出される素性の例を、図 4、及び図 5 に示す。

30

【0080】

また、上記の素性抽出関数の数が、システムパラメタの数  $N$  となるように設計すると、スコア関数 ( ) は以下の (5) 式のように線形関数として定義できる。

【0081】

【数 16】

$$\Phi(x^, y^; w^) = w^ \cdot f^ = \sum_{d=1}^N w_d f_d(x^, y^)$$

40

• • • (5)

【0082】

つまり、システムパラメタ  $w_d$  は素性抽出関数  $f_d$  の重みであり、値が大きければ、素性抽出関数  $f_d$  がよりシステムの出力データの選択に影響をおよぼし、マイナス側に大きければ、システムが出力データ  $y^$  をより選ばないような重み付けをしたことに相当する。つまりここでのシステムパラメタは素性の重みを決める値である。学習部 40 では、図 6 に示すように教師あり学習アルゴリズムによってシステムパラメタの学習を行う。

【0083】

50

目的関数は上記(4)式で表わされるが、上記(4)式は、解に離散制約が入るので、基本的に離散最適化問題となり、厳密に解を求めるのが非常に困難な問題の系になる。しかし、双対分解に基づく方法を活用することで効率的な解法が得られる(例えば、非特許文献3: Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein, 「Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers」, 2011, Foundations and Trends in Machine Learning)。まず、双対分解に基づいて上記(4)式の制約を分解する。

【0084】

【数17】

$${}^*w^{\wedge} = \arg \min_{w^{\wedge}, u^{\wedge}} \{O(w^{\wedge}, u^{\wedge}; \mathcal{D})\}, \quad O(w^{\wedge}, u^{\wedge}; \mathcal{D}) = \mathcal{L}(\Phi(\cdot), w^{\wedge}, \mathcal{D}) + \Omega(w^{\wedge}),$$

$$s.t. \quad w^{\wedge} = u^{\wedge}, \text{ and } u^{\wedge} \in \mathcal{S}_{\zeta}^N$$

••••(6)

10

【0085】

ここで、 $u^{\wedge}$ は、補助パラメタである。これは、等式制約 $w^{\wedge} = u^{\wedge}$ を用いて従来の最適化問題と制約を分離したことになる。次に、上記非特許文献3の拡張ラグランジュ緩和を用いて制約を目的関数に代入する。

【0086】

【数18】

$${}^*w^{\wedge} = \arg \max_{\alpha^{\wedge}} \arg \min_{w^{\wedge}, u^{\wedge}} \{O(w^{\wedge}, u^{\wedge}, \alpha^{\wedge}; \mathcal{D})\},$$

••••(7)

20

$$O(w^{\wedge}, u^{\wedge}, \alpha^{\wedge}; \mathcal{D}) = \mathcal{L}(\Phi(\cdot), w^{\wedge}, \mathcal{D}) + \Omega(w^{\wedge}) + \frac{\rho}{2} \|w^{\wedge} - u^{\wedge} + \alpha^{\wedge}\|_2^2 - \frac{\rho}{2} \|\alpha^{\wedge}\|_2^2$$

$$s.t. \quad u^{\wedge} \in \mathcal{S}_{\zeta}^N$$

30

【0087】

$\alpha^{\wedge}$ はラグランジュ乗数である。上記非特許文献3にしたがって、初期化部50、システムパラメタ更新部52、補助パラメタ更新部54、未定乗数更新部56、及び収束判定部58における処理によって、上記(7)式に示す目的関数を最適化することにより、本実施の形態では、値が重複したシステムパラメタが得られる。

【0088】

リスク関数は、基本的にどのような関数を用いてもよいが、ここでは、凸関数であることに限定する。本実施の形態では、リスク関数として、以下の(8)式に示すヒンジ損失関数を用いる。

【0089】

【数19】

$$\mathcal{L}(\Phi(\cdot), w^{\wedge}, \mathcal{D}) = \sum_{(x^{\wedge}, y^{\wedge}) \in \mathcal{D}} \max \left( 0, \Phi(x^{\wedge}, {}^*y^{\wedge}; w^{\wedge}) - \Phi(x^{\wedge}, y^{\wedge}; w^{\wedge}) + E(y^{\wedge}, {}^*y^{\wedge}) \right)$$

••••(8)

40

$${}^*y^{\wedge} = \arg \max_{y^{\wedge} \in Y(x^{\wedge})} \Phi(x^{\wedge}, y^{\wedge}; w^{\wedge})$$

【0090】

ここで、 $E(y^{\wedge}, {}^*y^{\wedge})$ は、 $y^{\wedge}$ と ${}^*y^{\wedge}$ とがどの程度違っているかを示す関数で

50

ある。 $y^\wedge$  と  ${}^*y^\wedge$  との違いが大きい程、 $E(y^\wedge, {}^*y^\wedge)$  は 0 以上の大きい値となり、 $y^\wedge$  と  ${}^*y^\wedge$  とが同一である場合に  $E(y^\wedge, {}^*y^\wedge)$  は 0 となる。

【0091】

また、リスク関数と同様に、正則化項も凸関数であることとする。例えば、正則化項について、以下の(9)式に示す  $L_1$ -ノルム正則化項を用いる。

【0092】

【数20】

$$\Omega(w^\wedge) = \lambda_1 \|w^\wedge\|_1$$

10

• • • • (9)

【0093】

リスク関数、及び目的関数が凸関数であるため、上記(2)式の最適化問題は基本的に凸最適化であり、大域的な最適解が必ず存在する。

【0094】

初期化部50は、最適化に用いる3種類のパラメタ  $w^\wedge$ 、 $u^\wedge$ 、 $\alpha^\wedge$  を全て 0 にセットする。以下繰り返し計算となるため、繰り返し回数を管理する変数を  $t$  とし、 $t = 0$  とする。

【0095】

20

システムパラメタ更新部52は、初期化部50で初期化された3種類のパラメタ  $w^\wedge$ 、 $u^\wedge(t)$ 、 $\alpha^\wedge(t)$ 、又は後述する各処理で前回更新された3種類のパラメタ  $w^\wedge$ 、 $u^\wedge(t)$ 、 $\alpha^\wedge(t)$  を用いて、以下に説明するように、システムパラメタ  $w^\wedge(t)$  から  $w^\wedge(t+1)$  へ更新する。

【0096】

反復計算  $t$  の時点で、 $u^\wedge(t)$  と  $\alpha^\wedge(t)$  を固定したときの  $w^\wedge$  の最適解は、目的関数

【数21】

$$\mathcal{O}(w^\wedge, u^\wedge(t), \alpha^\wedge(t); \mathcal{D})$$

30

【0097】

を最小にする  $w^\wedge$  を見つける問題の解である。

【0098】

【数22】

$$w^\wedge(t+1) = \arg \min_{w^\wedge} \{\mathcal{O}(w^\wedge, u^\wedge(t), \alpha^\wedge(t); \mathcal{D})\}$$

40

• • • • (10)

【0099】

定義に従って、上記(10)式から、 $w^\wedge$  に関係する項のみを取り出すと、下記(11)式となる。

【0100】

【数23】

$$w^{\wedge(t+1)} := \arg \min_{w^{\wedge}} \mathcal{L}(\Phi(\cdot), w^{\wedge}, \mathcal{D}) + \Omega(w^{\wedge}) + \frac{\rho}{2} \|w^{\wedge} - u^{\wedge(t)} + \alpha^{\wedge(t)}\|_2^2$$

• • • (11)

【0101】

この最適化問題は、上記(2)式に  $L_2$ -ノルム正則化項が追加された最適化問題とみなすことができる。この問題は、上記(2)式が凸最適化問題であれば、必ず凸最適化問題となるため、上記(2)式を解く方法を用いて容易に解くことができる。 10

【0102】

そこで、システムパラメタ更新部52は、初期化部50で初期化された3種類のパラメタ  $w^{\wedge}$ 、  $u^{\wedge(t)}$ 、  $\alpha^{\wedge(t)}$ 、又は後述する各処理で前回更新された3種類のパラメタ  $w^{\wedge}$ 、  $u^{\wedge(t)}$ 、  $\alpha^{\wedge(t)}$ を用いて、上記(11)式に従って、システムパラメタ  $w^{\wedge(t)}$  から  $w^{\wedge(t+1)}$  へ更新する。

【0103】

補助パラメタ更新部54は、システムパラメタ更新部52で更新された  $w^{\wedge(t+1)}$  と、初期化部50で初期化されたパラメタ  $\alpha^{\wedge(t)}$ 、又は前回更新されたパラメタ  $\alpha^{\wedge(t)}$ とを用いて、以下に説明するように、補助パラメタ  $u^{\wedge(t)}$  から  $u^{\wedge(t+1)}$  へ更新する。 20

【0104】

$w^{\wedge(t+1)}$  と  $\alpha^{\wedge(t)}$  を固定したとき  $u^{\wedge}$  の最適解は、目的関数

【数24】

$$\mathcal{O}(w^{\wedge(t+1)}, u^{\wedge}, \alpha^{\wedge(t)}; \mathcal{D})$$

【0105】

に制約

【0106】

【数25】

$$u^{\wedge} \in \mathcal{S}_{\zeta}^N$$

【0107】

が追加された最適化問題の解である。まず、 $u^{\wedge}$ に関連する項だけ集めると、下記(12)式に表す目的関数となる。

【0108】

【数26】

$$\mathcal{O}(w^{\wedge(t+1)}, u^{\wedge}, \alpha^{\wedge(t)}; \mathcal{D}) = \frac{\rho}{2} \|w^{\wedge(t+1)} - u^{\wedge} + \alpha^{\wedge(t)}\|_2^2$$

• • • (12)

$$s.t. \quad u^{\wedge} \in \mathcal{S}_{\zeta}^N$$

【0109】

まず、制約が無い場合を考えると、目的関数

【0110】

50

【数27】

$$\mathcal{O}(w^\wedge, u^\wedge, \alpha^\wedge; \mathcal{D})$$

【0111】

の  $u^\wedge$  に関する勾配が零ベクトルになる点である。その関係から下記(13)式及び下記(14)式の関係式が得られる。

【0112】

【数28】

$$\nabla_{w^\wedge} \mathcal{O}(w^\wedge(t+1), u^\wedge, \alpha^\wedge(t); \mathcal{D}) = \rho(w^\wedge(t+1) - u^\wedge + \alpha^\wedge(t)) = 0$$

•••• (13)

$$u^\wedge = w^\wedge(t+1) + \alpha^\wedge(t)$$

•••• (14)

【0113】

次に、制約を考慮すると、上記(12)式から、制約が無い場合の最適解  $u^\wedge = w^\wedge(t+1) + \alpha^\wedge(t)$  から自由度を持った超平面への写像問題と捉えることができる。よって、図7に示すように、 $u^\wedge = w^\wedge(t+1) + \alpha^\wedge(t)$  から最も近い超平面上の点を見つける処理を行えばよい。実際にこの処理は、図8に示すように、一次元の k-means と等価の処理になる(非特許文献4: Haizhou Wang and Mingzhou Song. Ckmeans.1d.dp: Optimal k-means Clustering in One Dimension by Dynamic Programming. The R Journal, 3(2):29-33, 2011.)。このことから、各パラメタに対して独立に処理が可能であり、非常に効率的に求めることができる。

【0114】

未定乗数更新部56は、システムパラメタ更新部52で更新された  $w^\wedge(t+1)$  と、補助パラメタ更新部54で更新された  $u^\wedge(t+1)$  と、初期化部50で初期化されたパラメタ  $\alpha^\wedge(t)$ 、又は前回更新されたパラメタ  $\alpha^\wedge(t)$  とを用いて、以下に説明するように、ラグランジュ未定乗数  $\lambda$  を更新する。

【0115】

$w^\wedge(t+1)$  と  $u^\wedge(t+1)$  を固定したときの最適値の方向は、目的関数

【0116】

【数29】

$$\mathcal{O}(w^\wedge(t+1), u^\wedge(t+1), \alpha^\wedge; \mathcal{D})$$

【0117】

の  $\lambda$  に関する勾配方向である。

【0118】

【数30】

$$\nabla_{\alpha^\wedge} \mathcal{O}(w^\wedge(t+1), u^\wedge(t+1), \alpha^\wedge; \mathcal{D}) = \rho(w^\wedge(t+1) - u^\wedge(t+1) + \alpha^\wedge) - \rho\alpha^\wedge$$

•••• (15)

【0119】

上記(15)式から下記(16)式の更新式を得る。

10

20

30

40

50

【0120】

【数31】

$$\begin{aligned}
 \alpha^{\wedge(t+1)} &= \alpha^{\wedge(t)} + \frac{1}{\rho} \nabla_{\alpha^{\wedge}} \mathcal{O}(w^{\wedge(t+1)}, u^{\wedge(t+1)}, \alpha^{\wedge}; \mathcal{D}) \\
 &= \alpha^{\wedge(t)} + w^{\wedge(t+1)} - u^{\wedge(t+1)}
 \end{aligned}
 \quad \cdots \cdots (16)$$

10

【0121】

以上より、未定乗数更新部56は、システムパラメタ更新部52で更新された $w^{\wedge(t+1)}$ と、補助パラメタ更新部54で更新された $u^{\wedge(t+1)}$ と、初期化部50で初期化されたパラメタ $\wedge(t)$ 、又は前回更新されたパラメタ $\wedge(t)$ とを用いて、上記(16)式に従って、ラグランジュ未定乗数 $\wedge(t)$ から $\wedge(t+1)$ へ更新する。

【0122】

収束判定部58は、予め定められた条件が満たされたか否かを判定し、当該条件が満たされるまで、システムパラメタ更新部52、補助パラメタ更新部54及び未定乗数更新部56による更新処理を繰り返す。具体的には、収束判定部58は、上記の各処理によって得られたシステムパラメタ $w^{\wedge}$ が最適値になっているか判定する。より詳細には、二つの小さな正の実数 $\varepsilon_1$ 、 $\varepsilon_2$ をあたえ、下記(17)式、及び下記(18)式を満たした際に収束したと判定する(上記非特許文献3参照)。

20

【0123】

【数32】

$$\|w^{\wedge(t+1)} - u^{\wedge(t+1)}\| < \varepsilon_1
 \quad \cdots \cdots (17)$$

$$\|u^{\wedge(t+1)} - u^{\wedge(t)}\| < \varepsilon_2
 \quad \cdots \cdots (18)$$

30

【0124】

そして、収束判定部58による収束判定で、収束していなかった場合は、 $t = t + 1$ としてシステムパラメタ更新部52に戻る。収束していると判定された場合は、後述する重複パラメタ圧縮部60に移る。

【0125】

重複パラメタ圧縮部60は、学習部40によって学習された複数のシステムパラメタに基づいて、値が同一となるシステムパラメタでグループを構成し、各グループにインデックス番号を付与し、各グループについて、グループを構成するシステムパラメタの各々を、当該グループに付与されたインデックス番号のシステムパラメタに変換し、当該インデックス番号のシステムパラメタを、後述するシステムパラメタ記憶部90に記憶する。具体的には、上記(5)式で定義したように、スコア関数には線形形式を仮定したので、 $w_i = w_j = \dots$ となるような、同じ値を取るシステムパラメタでグループを構築する。

40

【0126】

仮に、K個のグループができたと仮定すると、各グループに1からKまでのインデックス番号を付け

【0127】

【数33】

$$\{\mathcal{G}_k\}_{k=1}^K$$

【0128】

のようにK個のグループの集合を構築する。ここで、 $w_i = w_j = w_1$  の時に、

【0129】

【数34】

$$\mathcal{G}_k$$

10

【0130】

をシステムパラメタの元の番号の集合と考え、

【0131】

【数35】

$$\mathcal{G}_k = \{i, j, l\}$$

【0132】

20

と定義する。つまり、元のシステムパラメタ番号から、そのシステムパラメタと同じ値になるグループのインデックス番号への変換を、一時的に記憶しておくことを意味する。また、 $v_k$  を k 番目のグループの値とする、つまり、 $w_i = w_j = w_1$  で

【0133】

【数36】

$$\mathcal{G}_k = \{i, j, l\}$$

【0134】

のとき、 $w_i = w_j = w_1 = v_k$  である。

30

【0135】

また、重複パラメタ圧縮部60は、このインデックス番号の変換を素性関数  $f$  にも同じように適用する。このときに、新しく  $g_k$  を以下の(19)式のように、元の素性抽出関数の単純な線形結合で定義する。

【0136】

【数37】

$$g_k(x^\wedge, y^\wedge) = \sum_{i \in \mathcal{G}_k} f_i(x^\wedge, y^\wedge)$$

40

$$\dots \quad (19)$$

【0137】

すると、上記(5)式のスコア関数は、

【0138】

## 【数38】

$$\Phi(x^\wedge, y^\wedge; w^\wedge) = w^\wedge \cdot f^\wedge(x^\wedge, y^\wedge) = \sum_{d=1}^N w_d f_d(x^\wedge, y^\wedge) = \sum_{k=1}^K v_k g_k(x^\wedge, y^\wedge)$$

· · · · (20)

## 【0139】

という関係が成り立つ。ここで、複数のシステムパラメタ  $w^\wedge$  は、入力データ  $x^\wedge$  及び出力データ  $y^\wedge$  に対する複数の素性抽出関数  $f_d(x^\wedge, y^\wedge)$  の重みとなる。つまり、もともとの線形関数  $\sum_{d=1}^N w_d f_d(x^\wedge, y^\wedge)$  は、新たな線形関数  $\sum_{k=1}^K v_k g_k(x^\wedge, y^\wedge)$  と等価で変換できる。ただし、 $K$  はグループの数なので、 $N$  と比べると圧倒的に少ない数になっていることが想定できる。また、 $g_k$  はグループ内に属する素性関数の総和なので、事前に容易に計算できる。

## 【0140】

以上のように、重複パラメタ圧縮部 60 は、各グループについて、当該グループを構成するシステムパラメタの各々を、当該グループに付与された当該インデックス番号のシステムパラメタに変換し、当該インデックス番号のシステムパラメタを保存すると共に、上記の(20)式に示すように、スコア関数を定義する。

## 【0141】

よって、システムパラメタ更新部 52 により得られた重複が多くあるシステムパラメタの集合から、重複部分を融合し、等価だが無駄のない形式に圧縮することで、大幅にモデルサイズを削減することができる。また追加の処理として、 $w_d = 0$  になる場合、 $f_d$  は出力の選択に何も寄与しないので、素性関数の定義そのものを削除することができる。この処理もここで合わせて行う。

## 【0142】

システムパラメタ記憶部 90 には、重複パラメタ圧縮部 60 によって圧縮されたシステムパラメタ  $v_k$  と、新たに定義されたスコア関数  $(x^\wedge, y^\wedge; w^\wedge)$  とが格納される。

## 【0143】

<情報処理装置のシステム構成>

前述のシステムパラメタ学習装置 100 で得られた圧縮されたシステムパラメタを用いて、情報処理装置 200 によって、未知の入力データに対して所定の情報処理を行う。システムパラメタの圧縮を行う場合と、仮にシステムパラメタの圧縮をしなかった場合とで、処理結果は完全に一致する。よって、圧縮を行うことによって、モデルのサイズ（システムパラメタ自体のサイズ）を大幅に削減できる分、実行時に必要とされるリソースを削減できるというメリットだけを享受することができる。

## 【0144】

図9は、本発明の実施の形態に係る情報処理装置 200 を示すブロック図である。この情報処理装置 200 は、CPU と、RAM と、後述する情報処理ルーチンを実行するためのプログラムを記憶した ROM と、を備えたコンピュータで構成され、機能的には次に示すように構成されている。

## 【0145】

本実施の形態に係る情報処理装置 200 は、図9に示すように、入力部 210 と、システムパラメタ記憶部 220 と、情報処理部 230 と、出力部 240 とを備えている。

## 【0146】

入力部 210 は、入力データ  $x^\wedge$  を受け付ける。

## 【0147】

システムパラメタ記憶部 220 には、上記システムパラメタ学習装置 100 によって圧縮されたシステムパラメタ  $v_k$  と、新たに定義されたスコア関数  $g_k$  とが格納される。

## 【0148】

10

20

30

40

50

情報処理部 230 は、システムパラメタ記憶部 220 に格納された、インデックス番号  $k$  のシステムパラメタ  $v_k$  と、新たに定義されたスコア関数  $(x^k, y^k; w^k)$  とに基づいて、入力部 210 により受け付けた入力データ  $x^k$  に対して、所定の情報処理を行う。具体的には、情報処理部 230 は、入力部 210 により受け付けた入力データ  $x^k$  と、システムパラメタ記憶部 220 に格納された、インデックス番号  $k$  のシステムパラメタ  $v_k$  と、新たに定義されたスコア関数  $(x^k, y^k; w^k)$  とに基づいて、所定の最適化手法を用いて、スコア関数  $(x^k, y^k; w^k)$  が最大となる出力データ  $y^k$  を算出する。

#### 【0149】

出力部 240 は、情報処理部 230 によって算出された出力データ  $y^k$  を結果として出力する。 10

#### <システムパラメタ学習装置の作用>

次に、本実施の形態に係るシステムパラメタ学習装置 100 の作用について説明する。まず、教師データと、パラメタ  $w$  と、実数値の個数を表すパラメタ  $u$  とが、システムパラメタ学習装置 100 に入力されると、システムパラメタ学習装置 100 によって、入力された教師データが、教師データベース 30 へ格納される。

#### 【0150】

そして、システムパラメタ学習装置 100 によって、図 10 に示すシステムパラメタ学習処理ルーチンが実行される。 20

#### 【0151】

まず、ステップ S100において、初期化部 50 によって、最適化に用いる 3 種類の最適化変数  $w^k$ 、 $u^k$ 、 $v^k$  を全て 0 にセットし、初期化する。 20

#### 【0152】

次に、ステップ S102において、初期化部 50 によって、繰り返し回数を管理する変数  $t$  を、 $t = 0$  と設定し、初期化する。 20

#### 【0153】

ステップ S104において、システムパラメタ更新部 52 によって、上記ステップ S100 で初期化された 3 種類のパラメタ  $w^k$ 、 $u^k$ 、 $v^k$ 、又は後述する各ステップで前回更新された 3 種類のパラメタ  $w^k$ 、 $u^k$ 、 $v^k$  を用いて、上記 (11) 式に従って、システムパラメタ  $w^k$  から  $w^k$  へ更新する。 30

#### 【0154】

ステップ S106において、補助パラメタ更新部 54 によって、上記ステップ S104 で更新された  $w^k$  と、上記ステップ S100 で初期化されたパラメタ  $u^k$  、又は後述するステップ S108 で前回更新されたパラメタ  $u^k$  を用いて、上記 (14) 式に従って、 $u^k$  を計算し、計算された  $u^k$  から最も近い超平面上の点を見つけて、補助パラメタ  $u^k$  から  $u^k$  へ更新する。 30

#### 【0155】

ステップ S108において、未定乗数更新部 56 によって、上記ステップ S104 で更新された  $w^k$  と、上記ステップ S106 で更新された  $u^k$  と、上記ステップ S100 で初期化されたパラメタ  $v^k$  、又は本ステップ S108 で前回更新されたパラメタ  $v^k$  を用いて、上記 (16) 式に従って、ラグランジュ未定乗数  $v^k$  から  $v^k$  へ更新する。 40

#### 【0156】

ステップ S110において、収束判定部 58 によって、上記ステップ S104 で更新された  $w^k$  と、前回更新された  $w^k$  と、上記ステップ S106 で更新された  $u^k$  と、前回更新された  $u^k$  とに基づいて、上記 (17) 式、及び上記 (18) 式に示す収束条件を満たしているか否かを判定する。そして、収束していないと判定された場合には、ステップ S112 で繰り返しを管理する変数  $t$  をインクリメントして、ステップ S104 へ移行し、上記ステップ S104 ~ ステップ S108 の各処理を繰り返す。収束したと判定された場合には、ステップ S114 へ移行する。 50

## 【0157】

ステップS114において、重複パラメタ圧縮部60によって、上記ステップS104の更新処理によって最終的に得られたシステムパラメタ $w^{\wedge}$ の値に基づいて、値が同一となるシステムパラメタでグループを構成し、各グループについて、グループを構成するシステムパラメタの各々を、当該グループに付与された当該インデックス番号のシステムパラメタに変換し、当該インデックス番号のシステムパラメタをシステムパラメタ記憶部90に保存すると共に、上記の(20)式に示すように、新たなスコア関数を定義して、システムパラメタ記憶部90に保存して、システムパラメタ学習処理ルーチンを終了する。

## 【0158】

<情報処理装置の作用>

10

次に、本実施の形態に係る情報処理装置200の作用について説明する。まず、システムパラメタ学習装置100のシステムパラメタ記憶部90に記憶されているシステムパラメタ $v_k$ の各々と新たに定義されたスコア関数 $(x^{\wedge}, y^{\wedge}; w^{\wedge})$ とが、情報処理装置200に入力されると、システムパラメタ記憶部90に格納される。そして、対象としての入力データ $x^{\wedge}$ が情報処理装置200に入力されると、情報処理装置200によって、図11に示す実行処理ルーチンが実行される。

## 【0159】

まず、ステップS200において、入力部210によって、入力データ $x^{\wedge}$ を受け付ける。

## 【0160】

20

次に、ステップS202において、情報処理部230によって、システムパラメタ記憶部220に記憶されたシステムパラメタ $v_k$ の各々と、新たに定義されたスコア関数 $(x^{\wedge}, y^{\wedge}; w^{\wedge})$ とを読み込む。

## 【0161】

ステップS204において、情報処理部230によって、上記ステップS200で受け付けた入力データ $x^{\wedge}$ と、上記ステップS202で読み込んだシステムパラメタ $v_k$ の各々及びスコア関数 $(x^{\wedge}, y^{\wedge}; w^{\wedge})$ とに基づいて、所定の最適化手法を用いて、スコア関数 $(x^{\wedge}, y^{\wedge}; w^{\wedge})$ が最大となる出力データ $y^{\wedge}$ を算出する。

## 【0162】

ステップS206において、出力部240によって、上記ステップS204で算出された出力データ $y^{\wedge}$ を結果として出力して、情報処理ルーチンを終了する。

30

## 【0163】

<実験結果>

次に、本実施の形態の実験結果を示す。実際にテキスト処理の問題では、識別する際に入力と出力を特徴付ける、いわゆる素性として単語等の離散シンボル的なものを扱うため、全体の素性数が数百万から、数十億程度まで利用されることが往々にしてあり得る。図12に、自然言語処理の構文解析と固有表現抽出問題において、本実施の形態で説明した手法を用いた際の効果を示す。

## 【0164】

図12中の横軸は、グループの数を表し、縦軸は、システムの解析精度を表している。この図からもわかるように、本実施の形態の方法を用いると、グループの数を2から8といった非常に少ない数で従来と同等の解析精度を得ることができる。

40

## 【0165】

次に、この実験データにおけるモデル学習の一回あたりの試行でおよそ6時間かかる。ここでの結果を得るために人手で調整した定義はおおよそ20回程度に及んだ。一方、本実施の形態の方法を用いると、特に試行錯誤をせずに、一回で同じ結果を出すことが可能である。これは単純計算で、モデル作成コストがおよそ20分の1にできたことと等価である。

## 【0166】

以上説明したように、本発明の実施の形態に係るシステムパラメタ学習装置によれば、

50

複数の入力データの各々と複数の入力データに対する複数の正解出力データの各々とのペアである教師データを受け付け、受け付けた教師データと、スコア関数に基づいて、複数のシステムパラメタの各々の値が、個数が予め定められた実数値  $v_i$  と実数値  $-v_i$  と 0 とからなる離散値の集合に含まれる制約を満たし、かつ、最適化された複数のシステムパラメタを学習することによって、自動で適切な高压縮モデルを獲得することができる。

【0167】

また、本発明の実施の形態に係る情報処理装置によれば、システムパラメタ学習装置によって学習された、圧縮されたシステムパラメタに基づいて、入力データに対して、所定の情報処理を行って出力データを出力することにより、モデルのサイズ（システムパラメタ自体のサイズ）を大幅に削減できる分、実行時に必要とされるリソースを削減できる。

10

【0168】

また、従来とほぼ同等の精度を保ちつつシステムパラメタ自体のサイズを大幅に削減することが可能となる。具体的には、システムパラメタ数 1 億のモデルの場合は、 $100,000,000 \times 8 \text{ byte} = 800,000,000 \text{ byte}$  であるので、800 MB の容量を必要とする。しかし、本実施の形態を用いることで、システムパラメタ値の種類数を、例えば、8 程度といった非常に小さい値にまで削減できる。この場合、 $8 \times 8 \text{ byte} = 64 \text{ byte}$  であり、計算上は、約 1250 万分の 1 程度に圧縮できたことになる。

【0169】

また、従来技術で示した方法との違いとしては、従来技術では、

20

【0170】

【数 39】

$$w^\wedge \in \mathcal{S}^N$$

【0171】

中に現れる集合

【0172】

【数 40】

$$\mathcal{S}$$

30

【0173】

の定義

【0174】

を入手により調整して得る。この時、試行錯誤により、よい定義を発見しなくてはいけないため、一般的には、数十回の試行を逐次または並列実行し、その中で最も開発セットの結果がよいものを選択する方法を用いる。一方、本実施の形態においては、入手による調整が不要であり、かつ、たった一回の試行でほぼ確実に従来技術と同レベルの精度と圧縮率を保ったモデルを獲得することが可能となる。これにより、モデル構築および選択の速度とコストが従来に比べて数から数十分の一程度に削減できるため、実用上非常に価値の高い効果を与えることができる。

40

【0175】

なお、本発明は、上述した実施形態に限定されるものではなく、この発明の要旨を逸脱しない範囲内で様々な変形や応用が可能である。

【0176】

例えば、システムパラメタ記憶部 90、220 及び教師データベース 30 は、外部に設けられ、システムパラメタ学習装置 100 及び情報処理装置 200 とネットワークで接続されていてもよい。

【0177】

50

また、上記実施の形態では、システムパラメタ学習装置100と情報処理装置200とを別々の装置として構成する場合を例に説明したが、システムパラメタ学習装置100と情報処理装置200とを1つの装置として構成してもよい。

#### 【0178】

上述のシステムパラメタ学習装置100及び情報処理装置200は、内部にコンピュータシステムを有しているが、「コンピュータシステム」は、WWWシステムを利用してい10る場合であれば、ホームページ提供環境（あるいは表示環境）も含むものとする。

#### 【0179】

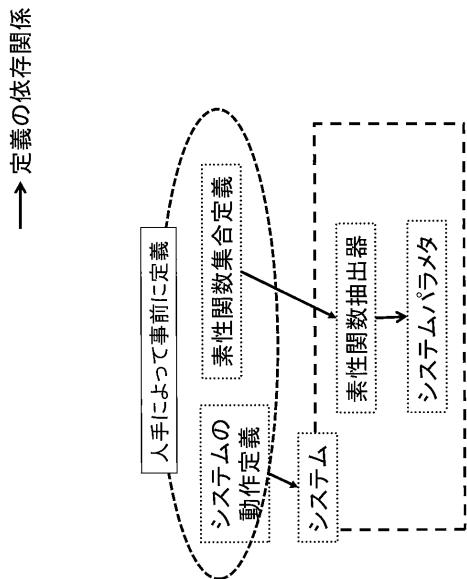
例えれば、本願明細書中において、プログラムが予めインストールされている実施形態として説明したが、当該プログラムを、コンピュータ読み取り可能な記録媒体に格納して提供することも可能である。

#### 【符号の説明】

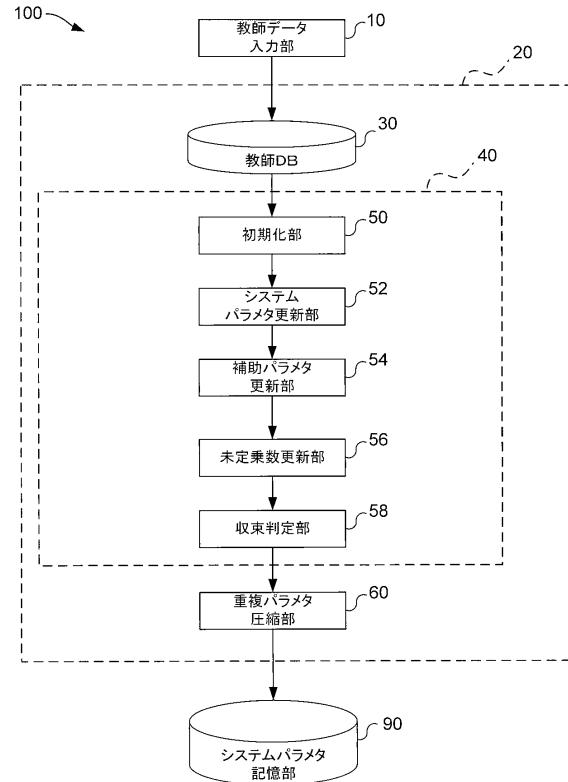
#### 【0180】

10	教師データ入力部	20
20	演算部	
30	教師データベース	
40	学習部	
50	初期化部	
52	システムパラメタ更新部	
54	補助パラメタ更新部	
56	未定乗数更新部	
58	収束判定部	
60	重複パラメタ圧縮部	
90	システムパラメタ記憶部	
100	システムパラメタ学習装置	
200	情報処理装置	
210	入力部	
220	システムパラメタ記憶部	
230	情報処理部	
240	出力部	30

【図2】



【図3】



【図4】

・推定したい出力の部分構造に対して、その部分構造と入力構造の組み合わせで素性を抽出

$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$
田中	一郎	は	陸上	連盟	の	会長	です
$y_1$	$y_2$	$y_3$	$y_4$	$y_5$	$y_6$	$y_7$	$y_8$
人名	人名	o	組織名	組織名	o	o	o

$y_4$ のラベル推定時に抽出される素性（前後2単語から素性を抽出する例）

$y_4$ の推定ラベル &  $x_2$   
 $y_4$ の推定ラベル &  $x_3$   
 $y_4$ の推定ラベル &  $x_4$   
 $y_4$ の推定ラベル &  $x_5$   
 $y_4$ の推定ラベル &  $x_6$

例

$y_4$ を「組織名」と推定する場合の素性

$i$ 番目のラベルがB-組織名& $i$ -2の単語が「一郎」  
 $i$ 番目のラベルがB-組織名& $i$ -1の単語が「は」  
 $i$ 番目のラベルがB-組織名& $i$ の単語が「陸上」  
 $i$ 番目のラベルがB-組織名& $i$ +1の単語が「連盟」  
 $i$ 番目のラベルがB-組織名& $i$ +2の単語が「の」

$y_4$ を「人名」と推定する場合の素性

$i$ 番目のラベルがB-人名& $i$ -2の単語が「一郎」  
 $i$ 番目のラベルがB-人名& $i$ -1の単語が「は」  
 $i$ 番目のラベルがB-人名& $i$ の単語が「陸上」  
 $i$ 番目のラベルがB-人名& $i$ +1の単語が「連盟」  
 $i$ 番目のラベルがB-人名& $i$ +2の単語が「の」

【図5】

素性抽出関数集合の例

素性10:  $i$ 番目のラベルが組織名& $i$ -2の単語が「一郎」のパラメタ  
 素性11:  $i$ 番目のラベルが人名& $i$ -2の単語が「一郎」のパラメタ  
 ...

素性51:  $i$ 番目のラベルが組織名& $i$ -1の単語が「は」のパラメタ  
 素性52:  $i$ 番目のラベルが人名& $i$ -1の単語が「は」のパラメタ  
 ...

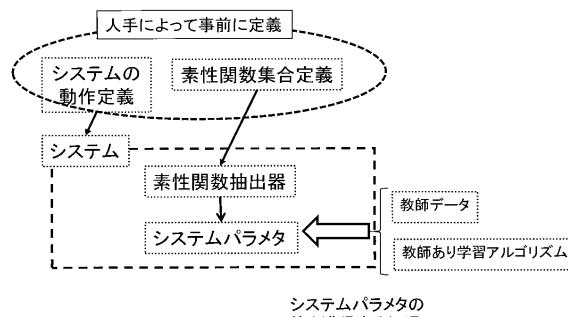
素性122:  $i$ 番目のラベルが組織名& $i$ の単語が「陸上」のパラメタ  
 素性123:  $i$ 番目のラベルが人名& $i$ の単語が「陸上」のパラメタ  
 ...

素性145:  $i$ 番目のラベルが組織名& $i$ +1の単語が「連盟」のパラメタ  
 素性146:  $i$ 番目のラベルが人名& $i$ +1の単語が「連盟」のパラメタ  
 ...

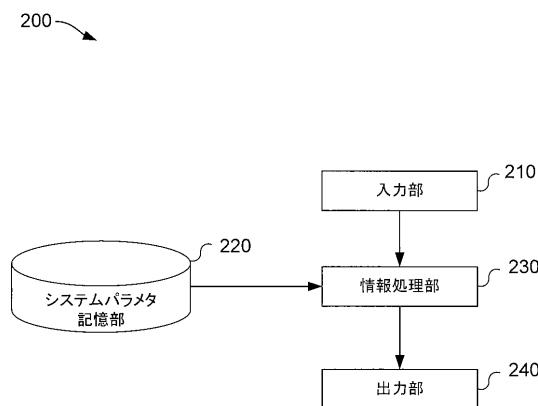
素性221:  $i$ 番目のラベルが組織名& $i$ +2の単語が「の」のパラメタ  
 素性222:  $i$ 番目のラベルが人名& $i$ +2の単語が「の」のパラメタ

【図6】

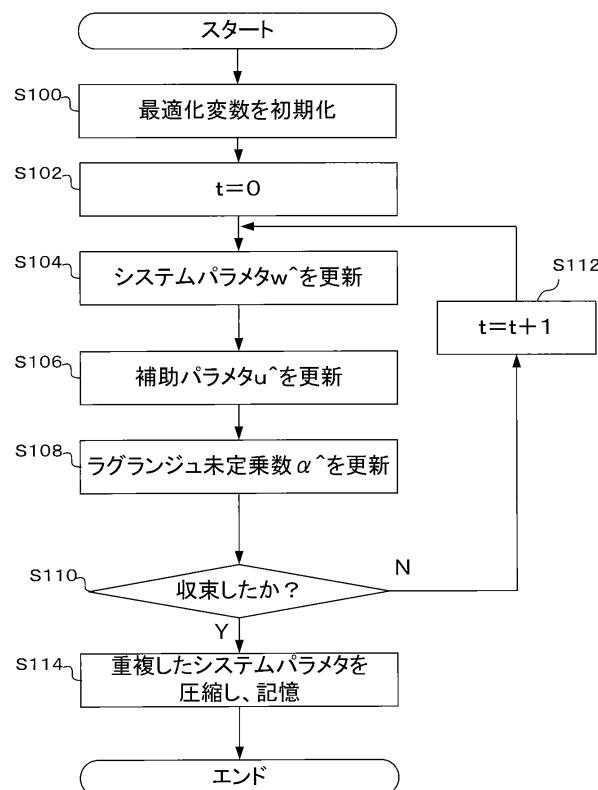
→ 定義の依存関係



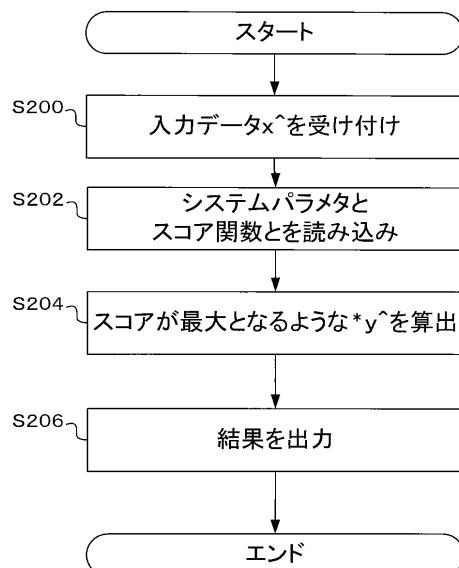
【図9】



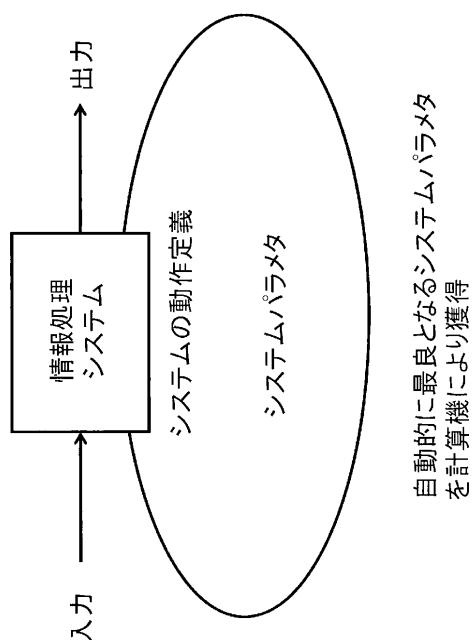
【図10】



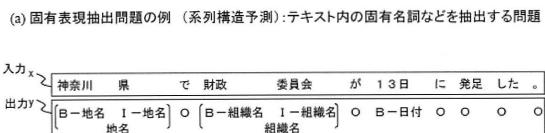
【図11】



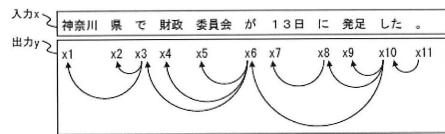
【図14】



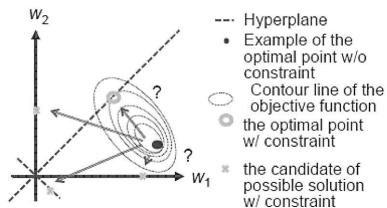
【図1】



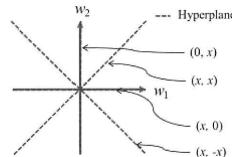
(b) 係り受け解析問題の例 (木構造予測): テキスト内の依存構造を解析する問題



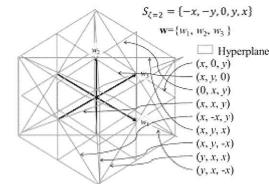
【図7】



2次元で考えた場合の例

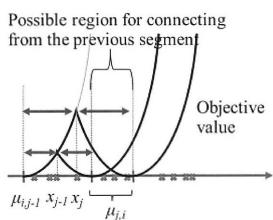


3次元で考えた場合の例

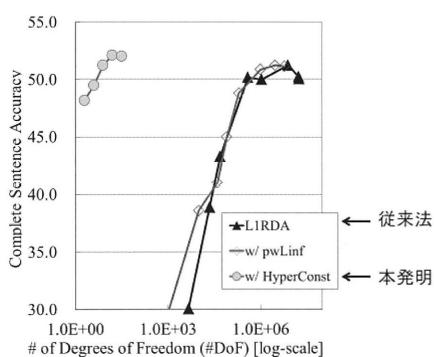


【図8】

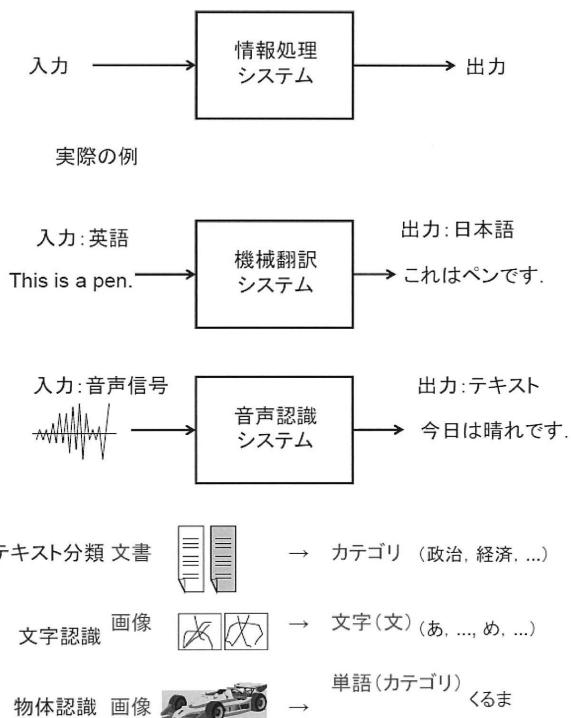
数直線上に値をプロットしてそれを5個の集合に分割する問題



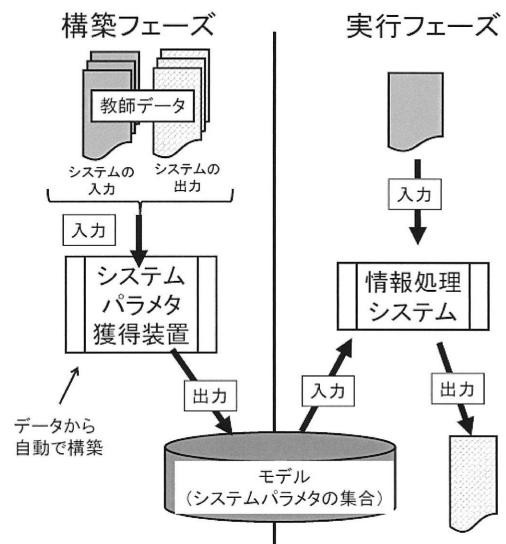
【図12】



【図13】



【図15】



---

フロントページの続き

(58)調査した分野(Int.Cl., DB名)

G 0 6 N 3 / 0 0

G 0 6 N 9 9 / 0 0