人　工　知　能　82－1
（１９９２．　５．　２６）

# 決定木の学習による文書データの分類と日本語キーワードの抽出

榊原 康文　　三末 和男
（株）富士通研究所 国際情報社会科学研究所

文書データを分類するための表現方法として，決定木を用いた方法，すなわち，文字列上の属性を扱う決定木，を提案し，それを帰納的に学習するアルゴリズムを示す．このアルゴリズムは，日本語処理に特有の分かち書き処理が必要ない，入力データに含まれるノイズに強い，などの特徴を持つ．そしてこの学習アルゴリズムは，キーワード自動抽出装置として用いることができることを示し，その実験結果を報告する．

# Classifying Document Data and Extracting Japanese Keywords by Learning Decision Trees

Yasubumi Sakakibara　　Kazuo Misue
*International Institute for Advanced Study of Social Information Science (IIAS-SIS)*
*FUJITSU LABORATORIES LTD.*
*140, Miyamoto, Numazu, Shizuoka 410-03, Japan*
E-mail : {yasu,misue}@iias.flab.fujitsu.co.jp

We introduce a class of representations for classifying document data based on decision trees, that is, decision trees over attributes on strings, and present an algorithm for learning it inductively. Our algorithm has the following features: it does not need Japanese segmentation processing, and it is robust for noisy data. We show that our learning algorithm can be used for automatic extraction of Japanese keywords. We also show some experimental results using our algorithm on classifying document data and extracting Japanese keywords.

## 1 はじめに

近年, 文書型データベースやフルテキストデータベースの普及に伴い, 大量の文書データが電子化され, これらを処理するための基本となる検索や分類などの文書ベースの情報検索技術の開発が要求されている. 現在までにも, コンピュータによるキーワード自動抽出や文書データの自動分類に関するいくつかの研究・開発[7, 8]が行なわれてきているが, まだまだ発展途上の段階と言える. またそこで使われている技術に共通して言えることは, 日本語処理などの言語処理技術にほとんど頼っていることであり, そしてそこで必要とされる辞書やシソーラスなどの構築は, 結局, 人手で行なっているのが現状である.

本稿は, 人工知能の分野で活発に行なわれている機械学習の研究や成果を, これらの情報検索における文書データの自動分類・日本語キーワードの自動抽出の問題に応用する, というまったく新しい方法を提案するものである. そしてその具体的なアプローチや結果, 可能性について示す.

まず, 機械学習の分野において分類を行なうための表現方法として良く用いられる決定木を基本にした, 文書データを分類するための表現方法, 正確には, 文字列上の属性を扱う決定木（これを, **文書分類木**と呼ぶ）, を導入する. 次に, 帰納的学習の方法を用いることによって, この文書分類木をすでに分類された文書群から自動的に構築するアルゴリズムを示す. この学習アルゴリズムは次のような特徴を持つ:

1. 日本語処理に特有の分かち書き処理が必要ない.
2. 入力データに含まれるノイズに強い.
3. 入力データの量が多いほど, 文書分類木の分類精度が高くなる.

さらにこの文書分類木を構築する学習アルゴリズムは, キーワード自動抽出装置として用いることができることを述べる.

またこの学習アルゴリズムを使って, 本の表題から本をいくつかの分類項目に分類する文書分類木を自動構築し, 日本語キーワードを抽出する実験を行なった. その結果を紹介し, 考察を行なう.

## 2 文字列上の属性を扱う決定木と文書データの分類

まず文字列上の属性を扱う決定木を導入し, それを文書の分類作業に応用する方法について示す.

決定木 (decision tree) は, 世の中の物（事例）が属性とその値の対の集合で定義されている場合に, これらをいくつかのクラスに分類するための規則を表現する方法の一つである.

$$\text{物} = \{(\text{属性}\,1, \text{値}), (\text{属性}\,2, \text{値}), \ldots\}$$

決定木の学習は, 医療診断システムなどの大量のデータを処理する分野などで用いられ, たとえば, そこでの学習問題は, 過去の患者の事例から病名を診断する決定木を学習するという問題になる.

ここでは, 文字列上の属性を考える. **文字列上の属性**とは, 文字列が持つ特徴・性質に関するものである. たとえば, 最も単純な属性としては, 文字列があるキーワードを部分文字列

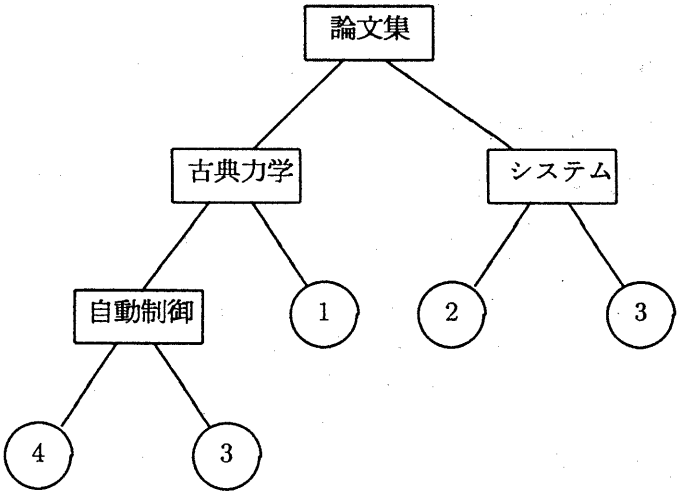


図 1: 本をその表題から分類する文書分類木の例.

として含むか否か, というものである. この属性を, **キーワード属性**と呼ぶことにする. より形式的には, $\Sigma$ を有限アルファベット, $\Sigma^*$ を $\Sigma$ 上のすべての有限の長さの文字列の集合とする. **キーワード**は, $\Sigma^*$ 中のある文字列である. キーワード属性とは, ある $w, v \in \Sigma^*$ に対して, 「文字列 $w$ は $v$ を部分文字列として含んでいるか？」という属性である.

このキーワード属性を扱う決定木を, **文書分類木**と呼ぶ. 形式的には, 文書分類木とは, キーワードが各内部ノードに, 文字列が分類されるクラス名が各葉にそれぞれラベル付けされた 2 分木である. この文書分類木は次のように文字列を分類し, その分類されるクラスを決定する. 1 つの文字列は, 文書分類木の根から葉への 1 つのユニークなパスを決定する: 各内部ノードにおいて, そのノードにラベル付けされているキーワードをその文字列が部分文字列として含んでいるならば右, そうでなければ左の枝をたどる. たどり着いた葉のクラス名が, その文字列が分類されるクラスとなる.

一般に, 世の中の文書はすべて, あるアルファベット上の文字列として考えることができる. したがって, 文書分類木は文書を分類する手段に使用することができる. 本をその表題からいくつかの分類項目に分類する文書分類木の例を, 図 1 に示す. この文書分類木により, たとえば, 「離散事象システム研究会講演論文集」という題の本は, 分類項目 3 に分類される.

## 3 文書分類木を学習するノイズに強いアルゴリズム

本節で文書分類木を帰納的に学習するアルゴリズムを示し, 次節でそれを使った実験結果を示す.

文書分類木を学習するアルゴリズムとは, すでに分類された文書群から, それをサンプルとして文書分類木を帰納的に構築するアルゴリズムである. 分類された文書とは, 文書とその分類クラスの組とする. たとえば, 図書のデータの場合には, 本の表題とその分類項目の組 (例, [遺伝子の発現と制御, 生物科学]) となる. 帰納的学習においては, この一つ一つの組を

例 (example), 組の集合（分類された文書群）をサンプルと呼ぶ.

我々の学習アルゴリズムは次の特徴を持つ.

1. 文書分類木を, 根から始めてトップダウンに構築する（TDIDT 法）.

2. 日本語処理に特有の分かち書き処理が必要ない.

3. 属性選択の評価関数として, Quinlan (ID3) [2] のエントロピー関数を使用する.

4. サンプル中のデータが含む誤分類（正確には, 分類ノイズ）に対処できる.

上記項目 2は, サンプル中の文書から, 指定された範囲内の長さのすべての部分文字列をキーワードの候補として生成し, それらすべてを学習する際の属性として扱うことによって, 実現されている. 後の実験によって示されるように, このような単純な方法を用いても, 日本語として正しく意味の通るキーワードが抽出される. またこの方法を採用することにより, 分かち書き処理のための辞書やパーザなどの大道具を用意する必要がなく, システムとして, 非常に軽量なものとなっている.

上記項目 4は, 榊原 [3] によって提案されたノイズを含むデータから決定木を学習するアルゴリズムに基づいて, 実現されている.

次の記号の準備をして, 図 2に我々の学習アルゴリズム *LEARN* を示す. 例 (example) とは, 組 $(w, l)$ である. ここで, $w$ は $\Sigma^*$ 中の文字列, $l$ は分類クラスのラベルである. **サンプル (sample)** とは, 例の有限集合である. $S$ をサンプル, $v \in \Sigma^*$, $l$ をラベルとする. このとき, $S_1^v$, $S_0^v$, $Occur$ を

$$\begin{aligned} S_1^v &= \{(w,l) \in S \mid w \text{ は } v \text{ を部分文字列として含む}\} \\ S_0^v &= \{(w,l) \in S \mid w \text{ は } v \text{ を含まない}\} \\ Occur(S,c) &= |\{(w,l) \mid l = c\}| \end{aligned}$$

と定義する. $S_0^v$ と $S_1^v$ が共に非空である時, 文字列 $v$ は **informative** であるという.

今, $m$ 個の分類クラスがあると仮定し, それらのラベルを $l_1, l_2, \ldots, l_m$ とする. $X$ を例の有限集合とする. 学習アルゴリズム *LEARN* で使われる属性の評価関数 *Loss* は, 次のように定義される.

$$I(X) = -\sum_{j=1}^{m} \frac{Occur(X,l_j)}{|X|} \log_2 \frac{Occur(X,l_j)}{|X|}$$

$$Loss(v,X) = \frac{|X_0^v|}{|X|} I(X_0^v) + \frac{|X_1^v|}{|X|} I(X_1^v)$$

学習アルゴリズム *LEARN* は, 分類ノイズに対処するため, ノイズの割合に関する値 $nsrt$ と枝刈り値 $prnrt$ を入力として取り, それを副手続き *FINDS* の引数に持たせている. この二つの値は, *FINDS* の 1と 2における停止条件に使われる. これらがノイズを扱うために拡張された部分であり, 今までの決定木を学習するアルゴリズムと異なる点である. 二つの引数 $prnrt$ と $nsrt$ の値は, Valiant の PAC 学習モデル [5] 上で形式的に求めることができる. ただし, 次節の実験においては, 経験的にこの値を決めている.

ALGORITHM *LEARN*
*Input*:

- A sample $S$,
- Parameters $keyl1$, $keyl2$, $prnrt$, and $nsrt$.

*Output*:

A decision tree $T$.

*Procedure*:

1. Calculate the following:

$$Keywords = \{v \mid keyl1 \leq |v| \leq keyl2, \quad v \text{ is a substring of } w \text{ for some example } (w, l)\};$$

2. Let $T = FINDS(S, Keywords, prnrt, nsrt)$;
3. Output $T$ and halt.

*Subprocedure* $FINDS(S, Keywords, prnrt, nsrt)$:

1. If $(|S| - Occur(S, l_i))/|S| \leq nsrt$ for some $l_i$,
   stop and return the decision tree $T = l_i$;
2. If $|S| \leq prnrt$,
   stop and return the decision tree $T = l_i$ for a largest $Occur(S, l_i)$;
3. Else
   - 3.1. Calculate $Loss(v, S)$ for all $v \in Keywords$ that is informative for $S$;
   - 3.2. If there is no informative keyword in $Keywords$,
     then stop and return $T =$"bad";
   - 3.3. Choose a longest keyword $v_g$ that minimizes $Loss(v_g, S)$;
   - 3.4. Let $T_0 = FINDS(S_0^{v_g}, Keywords - \{v_g\}, prnrt, nsrt)$
     and $T_1 = FINDS(S_1^{v_g}, Keywords - \{v_g\}, prnrt, nsrt)$;
   - 3.5. Stop and return the decision tree with root labelled $v_g$, left subtree $T_0$ and right subtree $T_1$;

図 2: 文書分類木の学習アルゴリズム *LEARN*

00! 化学大辞典
00! 画像解析ハンドブック
00! 岩波情報科学辞典
00! 現代数理科学事典
00! 情報処理ハンドブック
00! 人工知能大辞典
00! 全国試験研究機関名鑑 '91-'92 第I巻
00! 創造開発技法ハンドブック
00! 生化学辞典
01! 曲線・グラフ総覧
01! 初等数学幾何講義
01! 数式処理と数学研究への応用
01! 代数幾何学入門
02! 1991 情報学シンポジウム講演論文集
02! OPEN LOOK スタイルガイド
02! SUN システム管理
02! TeX ブック
02! 画像と言語の認識工学
02! 情報処理学会第 39 回全国大会講演論文集

図 3: 入力したサンプルの一部分.

| 分類項目 | 内　容 |
|---|---|
| 00. 参考 | ハンドブック, 辞典, 辞書, 説明書など. |
| 01. 数理科学 | 数学, 数理科学など. |
| 02. 情報 | 情報全般, 情報科学, 計算機科学など. |
| 03. 言語科学 | 言語学全般, 記号論, 統語論など. |
| 04. システム科学 | システム理論, 制御工学, ロボティックスなど. |
| 05. 生物科学 | 生物工学, 神経回路, DNA/ 遺伝子など. |
| 06. 人文科学 | 哲学, 心理学, 認知科学など. |
| 07. 社会科学 | 社会科学全般, 政策, 経営など. |
| 08. 環境科学 | 環境科学など. |
| 09. 教育 | 教育学, 図書館学など. |
| 10. 工学全般 | 電気工学, 電子工学, 電波工学, 機械工学など. |
| 11. 物理科学 | 物理科学全般（但し, 2. 〜 5. に含まれるものは除く） |

図 4: 分類項目のリスト.

学習アルゴリズム *LEARN* において, 変数 *Keywords* は, サンプル中の文字列の指定された長さ *keyl*1 から *keyl*2 までのすべての部分文字列（キーワードの候補, と呼ぶ）のリスト（キーワードリスト, と呼ぶ）を表す. また *LEARN* では, いくつかのキーワードの候補が同じ評価関数の値を持つ時は, 最大の長さのものを採用する.

# 4　実験と考察

## 4.1　実験

本実験では, 本のその表題からいくつかの分類項目に分類する文書分類木を, 前節の学習アルゴリズムを使って構築する問題を扱った. 本実験に使われたサンプルは, ある図書室に保管されている日本語の本の表題とその分類項目からなるデータである. 図 3は, 入力したサンプルの一部分である. ここで, フォーマットは“分類項目の番号！ 本の表題”となっている. 図 4は, その図書室で使用されている分類項目のリストとそれに関する説明である.

そこで, これらのすでに分類された文書（すなわち, 本の表題とその分類項目番号）群か

| | | |
|---|---|---|
| システ | システムシ | プログ |
| システム | システムシン | プログラ |
| システムの | システムシンポ | プログラミ |
| システムの研 | システムシンポジ | プログラミン |
| システムの研究 | システムシンポジウ | プログラミング |
| | システムシンポジウム | |

図 5: 生成されたキーワード候補の一部分.

ら, 学習アルゴリズム *LEARN* によって文書分類木を構築し,

1. 出力された文書分類木の内部ノードに現れるキーワードと図4の分類項目リスト中のキーワードを比較する,
2. 出力された文書分類木が未知の本をその表題からどれくらい正しく分類できるか（すなわち, 予測可能性）をテストする,

二つの実験を行なった.

実験 1では, サンプル中の例の数（実験に使う本の数）を 163 冊, キーワード候補の文字列の長さを 3 文字以上 12 文字以下, ノイズの割合に関する値 $nsrt$ を 0.2, 枝刈り値 $prnrt$ を 5, とした. 生成されたキーワード候補の一部分を図 5に, 構築された文書分類木の一部分を図 6に示す. 文書分類木のレイアウトには, 当研究所で開発中の図的発想支援システム D-ABDUCTOR [6] のグラフ描画機能を用いた.

実験 2では, サンプル数を 446 冊, キーワード候補の文字列の長さを 3 文字以上 8 文字以下, ノイズの割合に関する値 $nsrt$ を 0.05, 枝刈り値 $prnrt$ を 3, とした. サンプル中に含まれない別の本 306 冊の表題とその分類項目からなるテストデータを, 出力された文書分類木に入力した結果, 約 72% の本を正しく（すなわち, その本に付けられた分類項目と同じ分類項目に）分類した.

## 4.2 考察

実験 1において出力された文書分類木を見ると, 分かち書きなどの日本語処理を行なっていないにもかかわらず, 日本語として正しく意味の通るキーワードが抽出されていることが分かる. 図 7は, サンプル中の本で, その表題が図 6の文書分類木の一番右端の（C:4 とラベル付けされた）葉にたどり着くものであるが, 最初の本と最後の本はあきらかに分類項目 4 番に分類されるべきであり, 元の分類はエラーである. この結果から, 分類ノイズがうまく処理されていることが分かる.

実験 2において, 学習された文書分類木の正解率は 72% であったが, 一般に帰納的学習において, その予測正解率は 8 〜 9 割が望ましいとされている. 実験 2の正解率は, 次のような理由によるものと考えている:

1. 例の数が絶対的に少ない (446 冊).
   少なくとも数千のオーダーが必要で, できれば数万のオーダーが望ましい.
2. 各分野に特有の専門的な本が少数ずつあり, 例が偏っている.

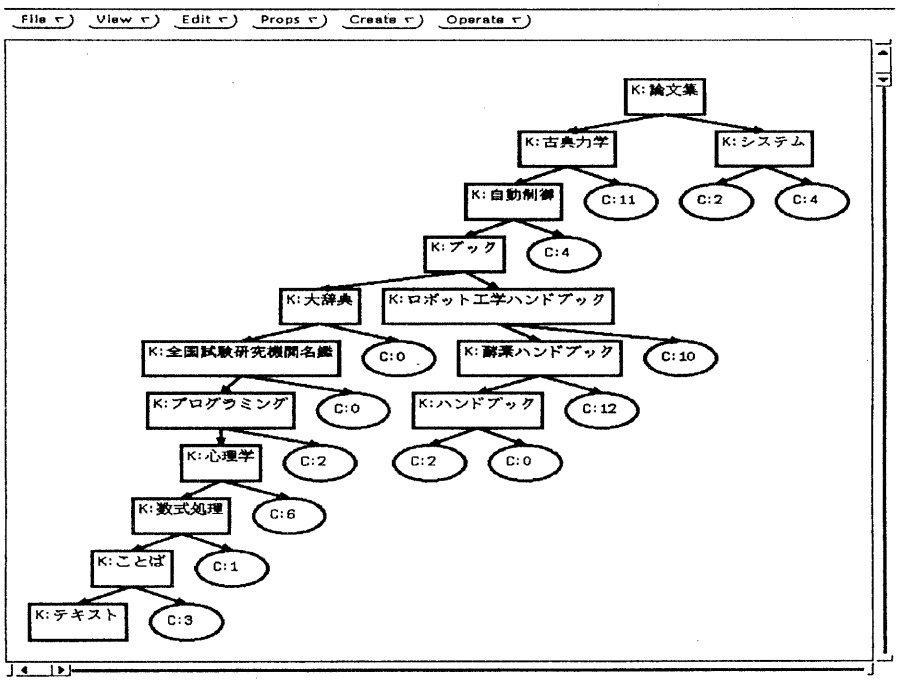


図 6: 構築された文書分類木の一部分.

3. その本に固有のキーワードがある / 多い （[7] を参照）.
4. 表題のみで分類することに限界がある.（要約や本文の一部も利用できると良い.）

## 5 キーワード自動抽出としての文書分類木の学習

本節では, 文書分類木を構築する学習アルゴリズムがそのまま, キーワード自動抽出に応用できるということについて述べる.

文献などの文書型データの情報検索においては, 「キーワード」を用いて検索を行なう方法が最も一般的である. しかし効率の良い検索を行なうための適切なキーワードを見つけ出すことは, 高度に専門的な行為であり, かなりの労力を要する. そこでこの作業をコンピュータにより支援しようとする, キーワード自動抽出技術の研究が盛んに行なわれ, ある程度の貢献をしている [7, 8].

コンピュータによるキーワード自動抽出は, 対象とする文書の中からキーワードを見つけ出し, 抽出する. 従来のキーワード自動抽出において用いられている方法は, たとえば, 発生頻度の高いものをキーワードにするというものである. この方法は, 重要な単語ほど文書中で使用される頻度が高いという前提に基づいている. しかし発生頻度が高いものには, 一般的でキーワードに適さない単語, たとえば「問題」とか「影響」など, が多く, 逆に発生頻度が低くても重要である単語は少なくない（特に, 専門分野においてはこの傾向が強い）.

我々は, 「キーワードを使って文献検索を行なうことは, 取り出したい文献とそうでない文献を分類することである」という考えを仮定することによって, キーワード自動抽出の問題に対する新しい見方・方法を提案する.

02! 大型プロジェクトパターン情報処理システム研究開発成果発表会論文集
04! 第 14 回システムシンポジウム講演論文集
04! 第 33 回システム制御情報学会研究発表講演会講演論文集
04! 第 35 回システム制御情報学会研究発表講演会講演論文集
04! 第 3 回離散事象システム研究会講演論文集
04! 第 4 回離散事象システム研究会講演論文集
04! 第 6 回離散事象システム研究会講演論文集
04! 第 7 回離散事象システム研究会講演論文集
10! 第 3 回ロボティクス・自動化システムシンポジウム講演論文集

図 7: 文書分類木の一番右端の（C:4 とラベル付けされた）葉にたどり着く本の表題.

すなわち, 文献検索の目的に適したキーワード群全体を, すでに分類された文書群から, 文書分類木を学習するアルゴリズムを使って, 完全に自動的に抽出するというものである. 学習アルゴリズムが出力した文書分類木中のノードに現れたキーワード全体が, 情報検索のために用いると良いキーワード群となる, と考える. 我々の方法の特徴は,

1. キーワードをどうやって獲得するか?
   **我々の方法:** 分類のための必要性に基づき学習によって獲得する.
   **従来の方法:** 出現頻度などの重み等に基づいて獲得する.
2. 日本語の分かち書きをどうするか?
   **我々の方法:** 学習によって獲得する.
   **従来の方法:** 辞書やパーザなどの道具を使う.
3. キーワードは分類する (他と区別する) ために獲得されたものなので, 文献検索などに効率良く使える.

## 6 まとめ

最後に, 今後の課題と応用可能性について, 簡単に述べる.
今後の課題としては, 次のような問題を考えている:

1. 現在の学習アルゴリズムでは, すべての部分文字列を候補としてるので時間がかかる. たとえば, 実験 1で生成されたキーワードの候補数は 5017 個であった. そこで, キーワードの候補を生成する際に, 多少の日本語の前処理を行なう. (たとえば, 句読点, 助詞, 接続詞, などの処理.)
2. 扱う属性のキーワードに “第*講演集” のようなワイルドカードを使ったパターンを考える. (類似の方法 [1] が, 遺伝子情報処理に応用されている.)
3. 基本属性の論理積 / 論理和を扱う. たとえば, 「システム ∧ 情報」など.
4. 文字列上のエラー (ノイズ) を扱う.
5. approximate pattern matching を用いた属性の評価.

本稿で示した学習アルゴリズムは, 分類ノイズは扱えるが, 文書中の文字列上のエラーは扱えない. そのような文字列上のエラーとしては,

削除：$aa\underline{b}c \rightarrow aac$, 挿入：$aabc \rightarrow aab\underline{c}c$, 置換：$a\underline{a}bc \rightarrow a\underline{b}bc$,

が考えられる. この文字列上のエラーによるノイズモデルは, 次のように定義される:「サンプルが与えられる時に, ある小さい確率 $\eta$ で, 上の文字列上のエラーがサンプル中の文書上で起きる」. このエラーに対処することにより,「メンテナンス」と「メインテナンス」などの, ひらがな・カナ表記のばらつきも扱えるようになる. このノイズモデルの理論的解析に関するいくつかの結果は, すでに得られている [4].

文書分類木を帰納的に学習する我々の方法の応用可能性としては,

- 新聞記事や特許出願などのフルテキストデータベースの分類・検索への応用,
- シソーラスの自動構築への応用,
- 大規模データベースからのサンプリングによる帰納的知識獲得への応用,

などが考えられる.

## 参考文献


[1] S. Arikawa, S. Kuhara, S. Miyano, Y. Mukouchi, A. Shinohara, and T. Shinohara. A machine discovery from amino acid sequences by decision trees over regular patterns. Technical Report 44, RIFIS, Kyushu University, 1991.

[2] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1:81–106, 1986.

[3] Y. Sakakibara. *Algorithmic Learning of Formal Languages and Decision Trees.* PhD thesis, Department of Information Science, Tokyo Institute of Technology, 1991. Research Report IIAS-RR-91-22E, IIAS-SIS, FUJITSU LABORATORIES LTD.

[4] Y. Sakakibara and R. Siromoney. A noise model on learning sets of strings. to appear in COLT'92, 1992.

[5] L. G. Valiant. A theory of the learnable. *Communications of the ACM*, 27:1134–1142, 1984.

[6] 三末和男. 図的発想支援システム D-ABDUCTOR における図の操作機能について. Research Report IIAS-RR-91-1J, IIAS-SIS, FUJITSU LABORATORIES LTD., 1991.

[7] 杉山, 山口, 田村. 自然語による索引語自動抽出システムとその索引語分析. 情報処理学会情報学基礎研究会報告書, 12-2, 1989.

[8] 内山, 中村. 重要キーワード抽出方式とその活用方法. 情報処理学会データベースシステム研究会報告書, 84-19, 1991.