知能と複雑系　132－17
(2003. 3. 14)

Decision Tree - Graph-Based Induction の機能拡張

ワロドム・ジアムサクン† 松田 喬† 吉田哲也† 元田 浩† 鷲尾 隆†

† 大阪大学産業科学研究所 〒567–0047 大阪府茨木市美穂ヶ丘 8–1
E-mail: †{warodom,matsuda,yoshida,motoda,washio}@ar.sanken.osaka-u.ac.jp

あらまし Graph-Based Induction（GBI 法）は，逐次的ペア拡張によってグラフ構造データから類型パターンを抽出する機械学習手法の一つである．一方，決定木はデータ分類に有効な手段であり，理解しやすいルールが得られるという利点があるが，決定木を構築するにはデータを属性 - 属性値として表現する必要がある．本稿は，GBI 法を用いて，グラフ構造データに対して分類器（決定木）を構築する手法，Decision Tree - Graph-Based Induction（DT-GBI 法）を提案する．この手法は，オンラインで属性（分類に有効な部分グラフ）を GBI 法により生成しながら決定木を構築するという特徴を持つ．DT-GBI 法の性能を UCI Repository からの DNA データセットに対する実験で評価し，DT-GBI 法がグラフ構造データに対して分類器を構築する効率的な手法であることを示す．

キーワード データマイニング，グラフ構造データ，Decision Tree - Graph-Based Induction (DT-GBI 法)

Functional Extension of Decision Tree - Graph-Based Induction

Warodom GEAMSAKUL†, Takashi MATSUDA†, Tetsuya YOSHIDA†, Hiroshi MOTODA†, and Takashi WASHIO†

† Institute of Scientific and Industrial Research, Osaka University
8–1, Mihogaoka, Ibaraki, Osaka 567–0047, Japan
E-mail: †{warodom,matsuda,yoshida,motoda,washio}@ar.sanken.osaka-u.ac.jp

Abstract A machine learning technique called Graph-Based Induction (GBI) efficiently extracts typical patterns from graph-structured data by stepwise pair expansion (pairwaise chunking). Meanwhile, a decision tree is an effective means of data classification from which rules that are easy to understand can be obtained. However, a decision tree could not be produced for the data which is not explicitly expressed with attribute-value pairs. In this paper, we propose a method of constructing a classifier (decision tree) for graph-structured data by GBI. In our approach attributes, namely substructures useful for classification task, are constructed by GBI on the fly while constructing a decision tree. We call this technique Decision Tree - Graph-Based Indution (DT-GBI). DT-GBI was tested against a DNA dataset from UCI repository. The results indicate the effectiveness of DT-GBI for constructing a classifier for graph-based data.

Key words Data mining, graph-structured data, Decision Tree - Graph-Based Induction (DT-GBI)

1. はじめに

近年，多数の新規化学物質が合成され，我々の生活の向上に役立っているが，一方でそれらの化学物質の有害性が問題となっている．そのため，化学物質の有害性を評価する必要があるが，これらを実験的に測定を行うためには長期の年月や多額の費用を必要とする．しかし，化学物質の特性は構造に深く関連しているために，構造から化学物質の特性を予測することは技術的に可能であり，かつ，意義が高いと考えられる．

化学物質はグラフ構造として表現することができる．このようなデータから知識を発見するにはグラフ構造データからのマイニングが望まれる．

グラフ構造データから知識発見（グラフマイニング）を適用できるその他の例として，類型的なウェブブラウジングのパターンのや化学物質の部分構造，DNA の部分構造を発見することや，患者の記録から診断上のルールを発見することが挙げられる．

Graph-Based Induction（GBI 法）[3], [9] は，隣接する 2 つのノードを逐次的に拡張すること（これを逐次ペア拡張と呼ぶ）によって一般のグラフデータから特徴的なパターンを発見する

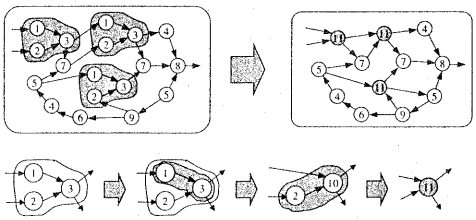

図1　GBI法の基本アイデア

ことを目的に考案された手法である．これはループ（自己ループを含む），有色／無色のノードとリンクを有するグラフデータを取り扱える．さらに，二つのノードの間にいくつのリンクがあっても構わない．欲張り探索 (greedy search) を用いるため，GBI法は高速に特徴的なパターンを抽出できるという利点がある．GBI法の逐次ペア拡張（チャンキング:chunking）はグラフ構造の情報を失うことなく，チャンキングの評価には頻度をはじめさまざまな評価関数を利用できる．逆戻りをしない欲張り探索を用いるため複数のノードに同じラベルがつけられるグラフ構造データには適していないが，それぞれのノードが個別のラベルを有するグラフ構造データ（例えば，WWWブラウジングのデータ）または特徴的な構造を持つグラフ構造データ（例えば，ベンゼン環を持つ化学物質など）からの特徴的なパターンの発見には有効である．

一方，決定木の構築[6],[7]はデータの分類および予測に広く使われる手法である．決定木の利点の一つは，理解しやすい「ルール」が導出されることである．しかし，大半のデータマイニング手法と同じように，決定木を構築する際には属性および属性値がデータに明示的に表現されている必要がある．グラフ構造データに対しては属性を事前に定義することが自明ではないため，グラフ構造データの分類に対して決定木として表現される分類器を構築することは困難である．

本稿では，GBI法を利用して属性を構築しながらに決定木を構築する手法を提案する．ここでは，複数のノードとそれらを結ぶリンクからなる「ペア」を属性とし，各グラフにペアが含まれるか否かを属性値とする．決定木を成長させながら，同時に分類に有効なペアをGBI法によって生成する．この手法をDecision Tree - Graph-Based Induction（DT-GBI法）と呼ぶ．

本稿の第2節でGBI法の概念を簡潔に紹介する．続いて第3節ではGBI法を利用したグラフ構造に対する分類器の構築する提案手法を紹介し，第4節では計算機上に実装した提案手法の実験的評価を報告する．最後にまとめと今後の課題を第5節に述べる．

2. Graph-Based Induction（GBI法）

図1に示すように，GBI法は逐次ペア拡張によって特徴的なパターンを抽出する．従来のGBI法では，特徴的なパターンは何らかの部分構造または構造を表し，特徴的か否かはパターンの頻度あるいは頻度に基づく任意の評価関数の値によって特徴付けられる．GBI法では，頻度の他に情報利得（information gain）[6]，利得比（gain ratio）[7]，ジニ指数（Gini index）[2]などの統計的指数を評価関数として用いることができる．

逐次ペア拡張では，図1に示すように，1,2,3番ノードからなり，黒色で強調されるパターンはグラフに3回も現れるため類型的とされる．GBI法は頻度に基づいて1→3というペアを発見し，このペアを10番という新しいノードにチャンクする．GBI法をもう1度繰り返すと，2→10というペアが発見され，11番という新しいノードにチャンクされる．以上の3つのステップを繰り返すことにより，様々なサイズの類型パターンを抽出することができる．欲張り探索のために逆戻りがないことに注意されたい．これは，ペアを列挙される際に一度チャンクされたペアが元のパターンに戻されないことを意味する．このため，グラフに現れる全ての類型パターンが必ずしも抽出されるとは限らない．全ての同形な部分グラフを抽出する問題がNP完全であることが知られているため，GBIは十分大きい類型パターンのみ抽出することを目指す．言い換えれば，GBIの目的は全ての類型パターンまたは頻繁パターンを見つけることではない．

前述の通り，GBI法では頻度に基づく任意の評価関数を使用できる．但し，逐次ペア拡張に基づくため，特徴的なパターンを見つけるためにはその部分パターンも特徴的なものでないといけないという条件がある．図1では，2→10ペアが類型的であるためには1→3ペアも類型的でないといけない．言い換えれば，1→3ペアがチャンクされない限り，2→10ペアが見つかることはない．頻度はこの単調性を満たす．評価関数がこの条件を満たさなければ，その評価値が最も高いペアを選んでチャンキングを数回繰り返しても良いパターンが得られない可能性がある．この問題を解決するために，著者らは2つの評価関数を同時に使うようにGBI法を改良した．評価関数の一つはチャンクするペアを選択するための頻度，もう一つはグラフ中に含まれるペアの中から特徴的な部分グラフを抽出するための任意の関数である．なお，二つ目の評価関数は単調性を満たさなくても良く，情報利得[6]や，利得比[7]，ジニ指数[2]などを用いることができる．

改良した逐次ペア拡張アルゴリズムを図2に示す．これらの4つのステッをが終了条件（通常は最低サポート）を満たすまで繰り返す．

GBI(G)
  Enumerate all the pairs P_{all} in G
  Select a subset P of pairs from P_{all} (all the pairs in G) based on typicality criterion
  Select a pair from P_{all} based on chunking criterion
  Chunk the selected pair into one node c
  G_c := contracted graph of G
  while termination condition not reached
    $P := P \cup \mathrm{GBI}(G_c)$
  return P

図2　GBI法のアルゴリズム

ステップ 1 全ての状態について，グラフに存在するペアを全て抽出する．

ステップ 2a ステップ 1 で抽出したペあのうち，評価関数により特徴的なペアを全て登録する．この時，ペアを構成するノードが既に書き換えられたノードであれば元のパターンに復元してから登録する．

ステップ 2b ステップ 1 で抽出したペアのうち，頻度によりチャンクすべきペアをある一定の数だけ選び，抽出パターンとして登録する．この時，ペアを構成するノードが既に書き換えられたノードであれば元のパターンに復元してから登録する．この際，チャンクすべきペアがなければ終了する．

ステップ 3 ステップ 2b で選ばれたそれぞれのペアに対し，ペアを一つのノードに置き換えることにより，それぞれにグラフを書き換える．この際，必要の応じて状態を分裂・消滅させる．ステップ 1 に戻る．

改良した GBI の出力はステップ 2a で抽出された特徴的なパターンの順位リストである．このリストに含まれるパターンは，使われた基準から見ると選ばれなかったパターンより特徴があるという意味で類型的といえる．

3. Decision Tree - Graph-Based Induction (DT-GBI 法)

3.1 GBI 法での要素の構築

決定木の表現は理解しやすいため，属性 - 属性値で表されるデータに対する分類器としてしばしば利用される．一方，グラフ構造データは通常ノードとリンクで表現され，属性と属性値に相当する要素を持たない．このため，グラフ構造データに対して直接決定木を構築することは困難である．しかし，部分グラフを属性とすれば，グラフ構造データを属性 - 属性値として表現できるため，決定木を構築することが可能になる．

分類に有効な部分グラフを予め抽出することは困難であるが，ペアを GBI 法により逐次的に拡張し，特徴的なものを決定木を構築しながらに拡張していくことで，決定木の構築と同時に分類のための属性に相当する特徴的なパターン（部分グラフ）を作ることができる．我々の手法では，属性および属性値は次のように定義される．

- 属性：グラフ構造データに含まれるペア（部分グラフ）
- 属性値：グラフ中でのペア（部分グラフ）の有無

決定木を構築する時，全グラフに含まれる全てのペアを数え上げ，その中から評価値の高い（分類に効果的と考えられる）ペアを一つ選ぶ．データ（構造データの集合）を二つのグループ，つまり，選ばれたペアが含まれるグループと含まれないグループに分ける．次に，前者に属する全てのグラフに対して，選ばれたペアを一つのノードに置き換えるチャンキングを行う．選ばれたペアが 1 枚のグラフに複数含まれる場合は全てチャンクする．これらの過程を決定木の各ノードで実行し，分類のための属性を作ると同時に決定木を成長させていく．以上の DT-GBI 法のアルゴリズムを図 3 にまとめる．

```
DT-GBI(D)
  Create a node DT for D
  if termination condition reached
    return DT
  else
    P := GBI(D) (with the number of chunking
      specified)
    Select a pair p from P
    Divide D into D_y (with p) and D_n (without p)
    Chunk the pair p into one node c
    D_yc := contracted data of D_y
    for D_i := D_yc, D_n
      DT_i := DT-GBI(D_i)
      Augment DT by attaching DT_i as its child along
        yes(no) branch
  return DT
```

図 3 Algorithm of DT-GBI

属性値が YES（指定のペアがある）と NO（指定のペアがない）の二つであることから，構築された決定木は二分木になる．

以上で提案した手法は，決定木を構築しながら分類のための属性（ペア）を構築するという特徴を持つ．データ分類のためのペアが選ばれる度にそのペアがチャンクされ，より大きな部分グラフに成長していく．従って，初期のペアに二つのノードとそのリンクしかなくても，数回チャンキングを適用することで分類に効果的なペアが徐々により大きなペア（部分グラフ）に成長する．提案した DT-GBI 法では分類に有効な属性（ペア）が次々と構築されることから，属性構築の手法の一つと考えられる．

3.2 DT-GBI 法の適用例

DT-GBI 法の適用例を挙げる．この手法が図 4 の左上隅のようなグラフセットを受けたとする．このデータには次の 13 種類のペアがある；a→a, a→b, a→c, a→d, b→a, b→b, b→c, b→d, c→b, c→c, d→a, d→b, d→c. a→a ペアがクラス A とクラス B のグラフに存在し，クラス C のグラフに存在しない．この段階でのペアの有無が表 1 に示すような属性 - 属性値表に変換される．その後，最も評価値の高いペアはデータを 2 つ

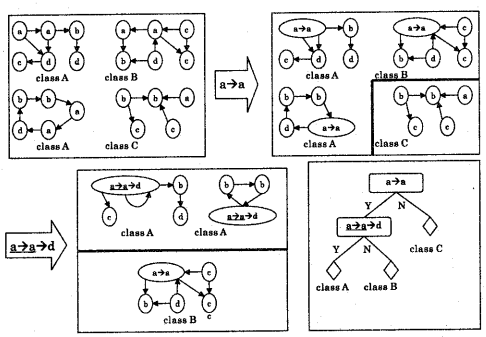

図 4 DT-GBI 法の基本アイデア

表 1　第 1 段階での属性 - 属性値表

Graph	a→a	a→b	a→c	a→d	b→a	b→b	b→c	b→d	c→b	c→c	d→a	d→b	d→c
1 (class A)	1	1	0	1	0	0	0	1	0	0	0	0	1
2 (class B)	1	1	1	0	0	0	0	0	0	1	1	1	0
3 (class A)	1	0	0	1	1	1	0	0	0	0	0	1	0
4 (class C)	0	1	0	0	0	1	1	0	1	0	0	0	0

表 2　第 2 段階での属性 - 属性値表（右下のグラフを除く）

Graph	a→a→b	a→a→c	a→a→d	b→a→a	b→b	...	d→c
1 (class A)	1	0	1	0	0	...	1
2 (class B)	1	1	0	0	0	...	0
3 (class A)	0	0	1	1	1	...	0

表 3　最終段階での属性 - 属性値表

Graph	a→a→d→a→a→d	a→a→d→b	a→a→d→c	b→a→a→d	b→d
1 (class A)	1	1	1	0	0
2 (class A)	0	1	0	1	1

のグループ（選ばれたペアが含まれるグループと含まれないグループ）に分けるものとして選ばれる．ペアが含まれるグループで現れる全ての分岐ペア（選ばれるペアそのもの）がノードにチャンクされ，グラフが書き換えられる．このグループに対応する属性 - 属性値表が表 2 である．DT-GBI 法は，分類に有効な属性（ペア）を生成すると同時に決定木を構築するために以上の過程を繰り返す．図 4 の最終段階での属性 - 属性値関係を表 3 に示す．

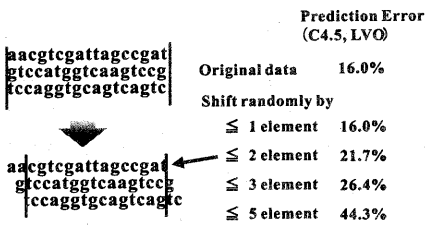

図 5　誤った予測

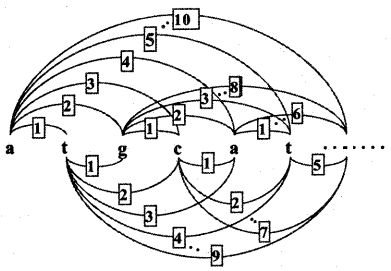

図 6　DNA 配列からグラフへの変換

4. DT-GBI 法の実験的評価

前節で示したアルゴリズムを実装し，評価実験として DNA の塩基列データに適用した．ドメインは DNA の塩基列データより，クラス分類に適したパターンを抽出することである．用いたデータセットは UCI Machine Learning Repository [1] により提供されている Promoter データセットである．Promoter データセットは塩基を表す A（アデニン），T（チミン），C（シトシン），G（グアニン）からなる長さ 57 の文字列データであり，クラスはその塩基列が “Promoter”（DNA を鋳型に mRNA 合成を開始する DNA 上の特定の塩基列）を含むことを示す Promoter と “Promoter” を含まないことを示す Non-promoter の 2 つである．データセット中の事例数は 106 個で，クラス Promoter とクラス Non-promoter のデータがそれぞれ 53 個である．

このデータセットの説明および分析の詳細は文献 [8] に参照されたい．このデータは，ある点を参照に塩基が整列されるように準備されたものであり，属性 - 属性値表現に従って n 番目の属性に n 番目の塩基を割り当てることができる．ある意味で，このデータセットはドメイン知識に沿って符号化されたものと言える．このことは次の実験で実証される．C4.5 [7] と Leave-One-Out (LVO) による評価実験では予測誤り率が 16.0%である．ランダムに配列を 3 つシフトすると予測誤り率が 21.7%，5 つシフトすると予測誤り率が 44.3%となってしまう（図 5 に参照）．データが正しく整列されなければ，C4.5 など属性 - 属性値表現を使う標準的な分類木は図 5 に示すような問題を解決できない．

グラフ表現の利点の一つとして，特定の点にデータを整列しなくても良いことが挙げられる．本稿では，この文字列を図 6 のように塩基をノードラベルとし，一つの塩基から両側 1〜10 個の塩基にそれぞれ 1 から 10 のラベルをつけたリンクを張ることで一つのグラフに変換し，GBI 法への入力とした．一つのグラフのサイズはノード数 57 個，リンク数 515 本と，かなり大きなグラフとなる．

実験においては，GBI でチャンクするペアの選択には頻度，GBI 法が出力したペアの中で最も分岐に有効なペアを選択するには情報利得 [6] を利用した．決定木は次の 2 つの方法で構築した：1) 根ノードでは n_r 回チャンキングし，その他のノードでは 1 回しかチャンクしない，2) 決定木の各ノードで n_e 回

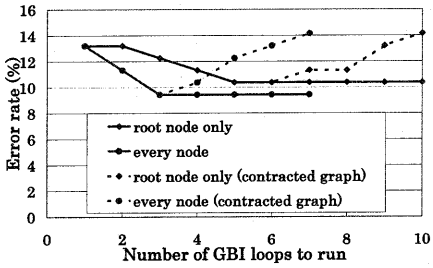

図 7　Promoter データセットを用いた実験の結果

チャンクする．実験 1) では n_r が 1〜10 に，実験 2) では n_e が 1〜7 に設定した．図 3 に示す DT-GBI 法の終了条件を D 内のグラフ数が 10 以下とした．DT-GBI 法に沿って構築された決定木の予測誤り率を，両実験ともに 5-fold cross-validation で評価した．その結果を図 7 に示す．この図の実線はチャンキング後に残るペアおよびチャンキングによって消滅したペア（即ち，その時点までに作られた全てのペア）の中から分岐ペアが選ばれる場合の誤り率を指し，点線は最後に残っているペアの中からしか選ばない場合の誤り率を指す．点線での誤り率の増加は，書き換えられたグラフに残ったペアしか選ばれない場合でできた決定木がチャンキングを繰り返すほど誤り率が悪くなってしまうことがあるを示す．ペアが大きくなり過ぎると分類能力（情報利得など）が低下してしまうと考えられる．

文献 [4] は Promoter データセットに対する決定木を構築する別のアプローチを報告している．探索能力を高めるために GBI 法にビーム探索が組み入れられる B-GBI 法により予め抽出したパターン（部分グラフ）を属性とし，C4.5 を用いて決定木を構築して予測精度を評価している．文献 [4] によると，C4.5 の予測誤差は 16.04%であり，B-GBI 法の予測誤差は 11.32%であった．それに対し，本稿の DT-GBI 法の最良の予測誤差は実験 2) で n_e=3 とした場合の 9.43%であった．n_e=3 とした場合の決定木を図 8 に示す．

しかし，同じ著者らは後に文献 [5] で B-GBI 法におけるビーム幅を 10 として C4.5 と組み合わせることで LVO による精度誤差が 2.8%となることを報告している．この際に構築された決定木と B-GBI 法におけるビーム幅を 1 として C4.5 により構築した決定木を比較したところ，決定木のノードに使用される属性（部分グラフ）が異なっていた．GBI 法は欲張り探索に基づいて高速にパターンを抽出するが，探索の不完全性により全ての特徴的なパターンは抽出できないという課題がある．B-GBI 法ではビーム探索を導入して探索空間を広げることによりこの問題の低減を図っており，ビーム幅を増加することでより多くの特徴的なパターンを抽出できるという利点がある．現状の DT-GBI 法でのチャンキングは B-GBI 法でビーム幅を 1 とした場合に対応するが，ビーム幅を増やして抽出したパターンを属性とした場合の方が予測精度が向上していることから，現状のままでは n_r や n_e を増加しても分類に効果的なパターンを抽出することは困難である．このため，DT-GBI 法の予測精度を向上させるためには各ノードでのチャンキングにおいてビーム探索を導入して探索空間を広げることが必要と考えられる．

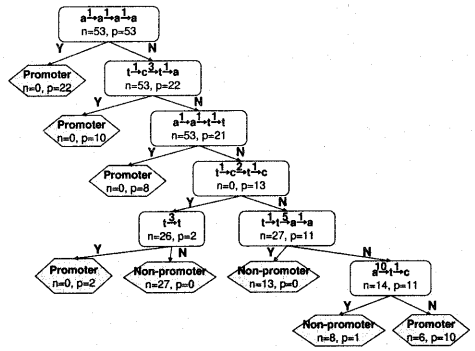

図 8　各ノードで 3 回チャンクする際の決定木

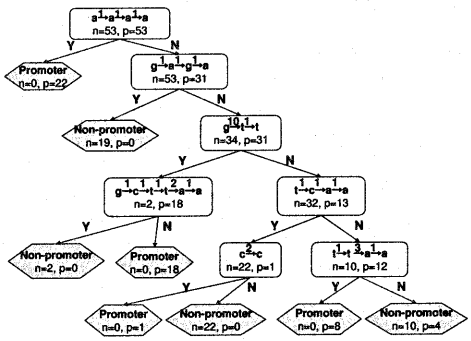

図 9　根のみで 5 回チャンクする際の決定木

なお，決定木の根ノードのみで複数回チャンクする実験 1) における最良の予測誤差は n_r=5 とした場合の 10.4%であった．根ノードの情報利得が大きいものの，それ以外のノードでは 1 回しかチャンクしないため分類に効果的なペアが現れず，全体的な情報利得が小さいまま決定木の構築が終了してしまうと考えられる．n_r=5 とした場合の決定木を図 9 に示す．

5. ま と め

本稿ではグラフ構造データに対して GBI 法を用いて分類器（決定木）を構築する DT-GBI 法を提案した．DT-GBI 法では，GBI 法で属性（分類に有効な部分グラフ）を生成しながら決定木を構築する．DT-GBI 法の予測精度を DNA プロモータ配列の分類問題で評価し，特徴的なパターンを抽出しながら決定木（分類器）を構築できることを示した．

現状では各ノードでのチャンキング回数 (4. 節での n_r や n_e) をパラメータとしているが，今後はチャンキングにより得られるペア（部分グラフ）の情報利得の変化率に基づいてチャンキング回数の自動化に取り組む．また，文献 [5] ではビーム探索を用いることでより分類に効果的なパターン（部分グラフ）が抽出されることが報告されているため，決定木の各ノードでチャンキングをする際にビーム探索を取り入れるように拡張する．

6. 謝　　辞

本研究の一部は文部科学省科研費特定領域研究「情報洪水時代におけるアクティブマイニングの実現」 (No.13131101, No.13131206) の補助による.

文　　献

[1] C. L. Blake, E. Keogh, and C. J. Merz. Uci repository of machine learning database, 1998. http://www.ics.uci.edu/ mlearn/MLRepository.html.

[2] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees.* Wadsworth & Brooks/Cole Advance Books & Software, 1984

[3] T. Matsuda, T. Horiuchi, H. Motoda, and T. Washio. Extension of graph-based induction for general graph structured data. In *Knowledge Discovery and Data Mining: Current Issues and New Applications, Springer Verlag, LNAI 1805*, pages 420–431, 2000.

[4] T. Matsuda, H. Motoda, T. Yoshida, and T. Washio. Knowledge discovery from structured data by beam-wise graph-based induction. In *Proc. of the 7th Pacific Rim International Conference on Artificial Intelligence, Springer Verlag, LNAI 2417*, pages 255–264, 2002.

[5] T. Matsuda, T. Yoshida, H. Motoda, and T. Washio. Mining patterns from structured data by beam-wise graph-based induction. In *Proc. of the 5th International Conference on Discovery Science*, pages 422–429

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

[7] J. R. Quinlan. *C4.5: Programs For Machine Learning.* Morgan Kaufmann Publishers, 1993.

[8] G. G. Towell and J. W. Shavlik. Extracting refined rules from knowledge-based neural networks. *Machine Learning*, 13:71–101, 1993.

[9] K. Yoshida and H. Motoda. Clip:Concept learning from inference pattern. *Journal of Artificial Intelligence*, 75(1):63–92, 1995.