3F7-2

情報利得値の上界に着目した特徴的部分グラフの効率的なマイニング

Pruning Strategy Focused on the Upper Bound of Information Gain in Mining Discriminative Subgraphs

高林健登 Kiyoto Takabayashi　原昌弘 Masahiro Hara　大原剛三 Kouzou Ohara　元田浩 Hiroshi Motoda　鷲尾隆 Takashi Washio

大阪大学産業科学研究所
The Institute of Scientific and Industrial Research, Osaka University

A graph mining technique called Chunkingless Graph-Based Induction (Cl-GBI) can extract discriminative subgraphs from graph-structured data by the operation called chunkingless pairwise expansion which constructs pseudo-nodes from selected pairs of nodes in the data. Because of the time and space complexities, it happens that Cl-GBI cannot extract subgraphs good enough to describe characteristics of data. Thus, to improve its efficiency, we propose a pruning method based on the upper-bound of information gain. Information gain is used as a criterion of discriminativity in Cl-GBI and the upper-bound of information gain of a subgraph is the maximal one that its super graph can achieve. The proposed method allows Cl-GBI to exclude unfruitful subgraphs from its search space by comparing the upper-bound of each subgraph with the best information gain at the moment. Furthermore, we experimenttally show the usefulness of the proposed method using both a real world dataset and artificial datasets.

1. はじめに

近年，大量に蓄積された電子化データから有用な知識を発掘するデータマイニングにおいて，複雑なデータをより柔軟に表現できるグラフ構造データを対象としたグラフマイニングが注目され，多くの手法が提案されている [Cook 94, Yoshida 95, Yan 02, Inokuchi 03]．その一手法である Chunkingless Graph-Based Induction（Cl-GBI）法 [Nguyen 05] は，従来手法である Graph-Based Induction（GBI）法 [Yoshida 95] と同様に，隣接する頂点対を逐次拡張（チャンク）することにより，グラフ中の特徴的な部分グラフを発見できる．ただし，Cl-GBI 法ではグラフ中の頂点対を 1 つの頂点に書き換えるのではなく，1 つの塊として捉え（擬似チャンク），新たな頂点として扱いつつもグラフ自体は書き換えない方法を採用している．これにより，Cl-GBI 法は GBI 法やその拡張である Beam-wise GBI（B-GBI）法 [Matsuda 02] では同時に抽出することが困難であった部分的に重複する部分グラフを抽出することが可能である．しかし，その一方で Cl-GBI 法の時間計算量，空間計算量は急激に増加する傾向を持ち，限られた計算時間，及び計算資源の下では，対象データの特徴を十分に表す部分グラフを抽出できない場合があった．

そこで本稿では，クラスが割り当てられた複数のグラフからクラス分類性能の高い部分グラフを抽出する問題に Cl-GBI 法を適用し，その場合における Cl-GBI 法の探索を効率化する情報利得の上界に着目した枝刈り手法を提案する．情報利得は，Cl-GBI 法において部分グラフのクラス分類性能の評価指標として用いられており，その上界を求めることが可能である [Morishita 00]．さらに本稿では，情報利得の上界による枝刈りを導入した Cl-GBI 法を人工データセット，および実データセットに適用し，その実行時間，得られた部分グラフのクラス分類性能などを従来の Cl-GBI 法と比較することで，提案手法の有効性を実験的に示す．また，その結果から予想される本提案手法の特性を様々な人工データセットを用いて検証する．なお，本稿では，擬似チャンクされた頂点対を擬似ノード，擬似チャンキングの対象となる頂点をノード，ノードの対をノードペア，もしくは単にペアと呼ぶ．

連絡先: 高林健登　〒 567-0047 大阪府茨木市美穂ヶ丘 8-1
大阪大学産業科学研究所 高次推論方式研究分野
電子メール: kiyoto_ra@ar.sanken.osaka-u.ac.jp

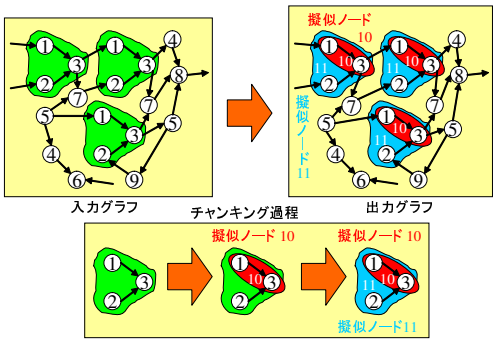

図 1: Cl-GBI 法における擬似チャンキングの例

2. Cl-GBI 法

図 1 は，入力グラフ中のノード 1，2，及び 3 からなる特徴的な部分グラフが擬似チャンキングにより抽出される過程を示している．まず，入力グラフ中で特徴的なノード 1 と 3 からなるペアが擬似チャンクされ，擬似ノード 10 として登録される．その後，擬似ノード 10 とノード 2 からなるペアが擬似チャンクされることで，擬似ノード 11 が生成される．この擬似ノード 11 が，前述の特徴的な部分グラフに相当する．

Cl-GBI 法のアルゴリズムを図 2 に示す．Cl-GBI 法では，ビーム幅 b，最大繰り返し数 N，及びペアが満たす必要条件 θ を与え，これらにより探索空間が制御される．また，アルゴリズムの **Step1.**〜**Step3.** までの一連の流れを一つの段階と考え，これをレベルと呼び，レベルは擬似チャンクすべきペアがある限り，0〜$N-1$ まで繰り返される [Nguyen 05]．なお，**Step 2.** における擬似チャンクする際のペアを選定基準として，GBI 法と同様にペアの出現頻度の他に対象ドメインに応じて情報利得 [Quinlan 86]，情報利得の上界，Gini Index[Breiman 84] など様々な特徴条件を用いることができる．

Input. グラフのデータベース D，ビーム幅 b，最大レベル N，擬似チャンクするペアの選定基準 C，ペアが満たす必要条件 θ

output. 特徴的な部分グラフの集合 S（最初は空集合）

Step 1. D のグラフ中の隣接する 2 つのノードから成る全てのペアを抽出する．また，レベル 2 以降については，2 つのノードのうち少なくとも片方は新しく登録された擬似ノードからなる全てのペアを抽出する．

Step 2. **Step 1.** で抽出されたペアのうち，θ を満たさないペアを探索から排除し，C に従って b 個のペアを θ を満たすペアの中から選ぶ．b 個の選ばれたペアをそれぞれ抽出部分グラフとして S に加える．この時，ペアを構成するノードが擬似ノードであれば元の部分グラフに復元してから S に加える．この際，擬似チャンクすべきペアがなければ終了する．また，レベルが N の場合もここで終了する．

Step 3. **Step 2.** で選ばれたペアにそれぞれ新しいラベルを割り当てるが，グラフは書き換えない．そして，**Step 1.** に戻る．

図 2: Cl-GBI 法のアルゴリズム

3. 情報利得の上界に基づく枝刈り

従来の Cl-GBI 法がしばしば用いる特徴条件は部分グラフが与えられたグラフデータ中に現れる頻度である．この特徴条件の下では，擬似チャンキングの導入により考慮すべきノード数が増加するため，GBI 法や B-GBI 法ではチャンクする度に減少していたペアが指数的に増加し，その結果，空間計算量と時間計算量も増加する．これに対して本研究では，探索過程において生成する部分グラフ g について，g を含むグラフ g' が取り得る情報利得の上界を求めることで g を拡張する価値があるか否かを判定し，拡張する価値のない部分グラフを探索から排除することで探索を効率化する．

3.1 情報利得

情報利得は，グラフ集合に対してある部分グラフを含むものと含まないものに分割した際に，その分割前後のグラフ集合の情報量の差として計算される．直感的には，グラフ集合がどのクラスに該当するのかというあいまいさが分割によりどれだけ減少したかを数値化したものであり，分割の基準に用いる部分グラフがあるクラスに特徴的であるほど，その数値は高くなる．2 つのクラス A，B のいずれかに属するグラフ集合 G を部分グラフ g を含むか否かで分割した場合の情報利得 $Gain(g,G)$ は以下の式で定義される．なお，G_g と $G_{\bar{g}}$ はそれぞれ，部分グラフ g を含む，含まないグラフ集合を表す．

$$Gain(g,G) = Ent(G) - \sum_{i\in\{g,\bar{g}\}} \frac{|G_i|}{|G|} Ent(G_i) \qquad (1)$$

ここで，$Ent(G)$，$Ent(G_i)$（$i \in \{g,\bar{g}\}$）はそれぞれグラフ集合 G，G_i の情報量であり，次式により求められる．

$$Ent(G) = -\sum_{j\in\{A,B\}} \frac{|G_j|}{|G|} \log_2 \frac{|G_j|}{|G|} \qquad (2)$$

$$Ent(G_i) = -\sum_{j\in\{A,B\}} \frac{|G_{i_j}|}{|G_i|} \log_2 \frac{|G_{i_j}|}{|G_i|} \qquad (3)$$

ここで，G_{g_A}，G_{g_B} はそれぞれ G_g 中のグラフのうちクラス A に属するグラフ集合，クラス B に属するグラフ集合である．また，$G_{\bar{g}_A}$，$G_{\bar{g}_B}$ はそれぞれ $G_{\bar{g}}$ 中のグラフのうちクラス A に属するグラフ集合，クラス B に属するグラフ集合である．

上記のように定義される情報利得は凸関数であり，g の任意の拡大グラフ g' に関して，その上界を計算できることが知られている [Morishita 00]．具体的には，g' を含むクラス A に属するグラフの集合を $G_{g'_A}$，g' を含むクラス B に属するグラフの集合を $G_{g'_B}$ としたとき，$|G_{g'_A}| = |G_{g_A}|$ かつ $|G_{g'_B}| = 0$，もしくは $|G_{g'_B}| = |G_{g_B}|$ かつ $|G_{g'_A}| = 0$ のいずれかのとき

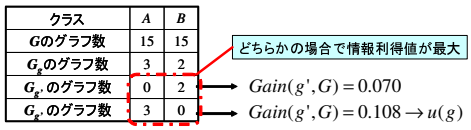

図 3: 情報利得の上界の計算例

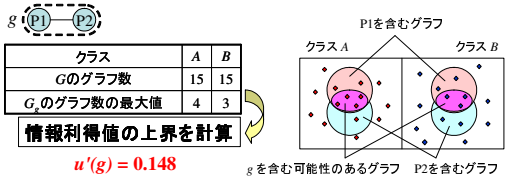

図 4: 事前枝刈りの例

に，$Gain(g',G)$ は最大値を取る．すなわち，この最大値が部分グラフ g を拡張した際の情報利得の上界となる．なお，g' は g の拡大グラフであるので，$|G_{g'_A}| \leq |G_{g_A}|$ であり，同様に $|G_{g'_B}| \leq |G_{g_B}|$ である．

図 3 に情報利得の上界の計算例を示す．部分グラフ g を含むグラフ数が $|G_{g_A}| = 3$，$|G_{g_B}| = 2$ であるとき，$|G_{g'_A}| = 0$，$|G_{g'_B}| = 2$，もしくは $|G_{g'_A}| = 3$，$|G_{g'_B}| = 0$ のときのいずれかで g の拡大グラフ g' の情報利得は最大となり，実際には後者の場合に $Gain(g',G) = 0.108$ という最大値を取る．以下では，部分グラフ g の拡大グラフが取り得る情報利得の上界を $u(g)$ とする．

3.2 Cl-GBI 法への枝刈りの導入

Cl-GBI 法のアルゴリズムの **Step 1.** において，頻度を数えると同時にその前後で情報利得の上界を用いた枝刈りを行う．ここで，それぞれ**事前枝刈り**，**事後枝刈り**と呼ぶ．**事前枝刈り**により頻度計算の負荷軽減ができ，**事後枝刈り**により次レベルにおける頻度計算の負荷とメモリ使用量の軽減が期待できる．

事前枝刈りは，ノードペアの頻度計算前にペアを構成する各ノードを共に含むグラフ数からノードペアの可能な最大の頻度を計算し，それに基づき情報利得の上界を計算する．図 4 に事前枝刈りの例を示す．ペア g の 2 つの親ノード $P1$，$P2$ を共に含むグラフ数は，図中に示すようにクラス A については 4 個，クラス B について 3 個である．そしてその値から計算した情報利得の上界 $u'(g)$ は 0.148 となる．したがって，それまでに抽出した部分グラフの情報利得の最大値を τ としたとき，$0.148 < \tau$ であれば部分グラフ g を拡張しても情報利得が τ を超えることがないので部分グラフ g は破棄される．このように，事前枝刈りは対象ペアの頻度計算が必要ない．

これに対して事後枝刈りは，実際に計算されたノードペア g の頻度を用いて情報利得の上界 $u(g)$ を計算し，$u(g) < \tau$ であれば部分グラフ g を破棄する．図 5 に事後枝刈りの例を示す．実際に計算されたノードペア g の頻度は，クラス A が 3，クラス B が 1 であり，それから得られる情報利得の上界 $u(G)$ は 0.108 となる．したがって，$0.108 < \tau$ であれば部分グラフ g は破棄される．これら 2 つの枝刈りを取り入れた Cl-GBI 法のアルゴリズムを図 6 に示す．**Step 2.1**〜**2.4** が図 2 のアルゴリズムの **Step 2.** に相当する．

4. 評価実験

本実験では，情報利得の上界を用いた枝刈りを行わないもの，事前枝刈りのみ行うもの，事後枝刈りのみ行うもの，両方の枝刈りを行うものの合計 4 種類のアルゴリズムが利用可能な

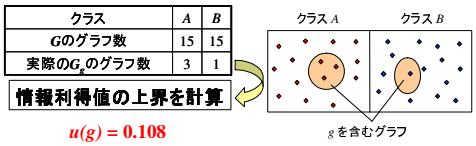

図 5: 事後枝刈りの例

Input. グラフ構造のデータベース D，ビーム幅 b，最大繰り返し数 N，擬似チャンクするペアの選定基準 C，ペアが満たす必要条件 θ，情報利得の上界の閾値 τ

Output. 特徴的な部分グラフの集合 S（最初は空集合）

Step 1. D のグラフ中の隣接する二つのノードから成る全てのペアを抽出する．また，レベル 2 以降については，二つのノードのうち少なくとも片方は新しく登録された擬似ノードからなるペアの全てを抽出する．

Step 2.1. （事前枝刈り）ノードペアの頻度計算前にペアを構成する各ノードを共に含むグラフ数からノードペアの可能な最大の頻度を計算し，それに基づき情報利得の上界を計算する．その値が τ より小さいペアは削除する．

Step 2.2. θ を満たさないペアは削除する．また，抽出されたペアの情報利得を計算し，これまでに抽出した情報利得の中で最大なものがあれば τ を更新する．

Step 2.3. （事後枝刈り）実際に計算されたノードペアの頻度を用いて情報利得の上界を計算し，その値が τ より小さいペアは削除する．

Step 2.4. C に従って b 個のペアを **Step 2.3.** の枝刈り後に擬似チャンキング候補として残っているペアの中から選ぶ．b 個の選ばれたペアをそれぞれ抽出部分グラフとして S に加える．この時，ペアを構成するノードが擬似ノードであれば元の部分グラフに復元してから S に加える．この際，擬似チャンクすべきペアがなければ終了する．また，レベルが N の場合もここで終了する．

Step 3. **Step 2.4.** で選ばれたペアをそれぞれ新しいラベルを割り当てるが，グラフは書き換えない．そして，**Step 1.** に戻る．

図 6: 情報利得の上界を用いた枝刈りを取り入れた Cl-GBI 法のアルゴリズム

Cl-GBI 法を計算機（CPU: Athlon XP 2100+，Memory: 3 GB, OS: Red Hat Linux 8.0）上に C++を用いて実装し，慢性肝炎データセットおよび人工データセットに適用した．具体的には，Cl-GBI 法のパラメータのうち，ビーム幅 b を 5 に固定し，最大繰り返し数 N を 5，10，15 に変化させ，擬似チャンクする際のペアの選定基準にグラフ中の部分グラフの出現頻度，または情報利得を用い，ペアの頻度が正の数であることをペアが満たす必要条件とし，各アルゴリズムに関して，計算時間，抽出された部分グラフの種類，抽出された部分グラフの情報利得の最大値を観測した．

4.1 データの仕様

本実験では，千葉大学医学部附属病院から提供を受けた慢性肝炎データセット [山口 02]，および特定の条件を満たすように作成した人工データセットを用いて実験を行った．慢性肝炎データセットに関しては，文献 [Geamsakul 05] と同様にインターフェロン投与の効果があった患者のクラスを R（Response），効果のなかった患者のクラスを N（Nonresponse）として，24 個の検査項目を属性として用いた．グラフ構造データへの変換の詳細は文献 [Geamsakul 05] を参照されたい．また，人工データセットに関しては，文献 [Nguyen 06] と同様に 2 クラスからなるデータセットをランダムに生成し，一方のクラス *positive* にのみ情報利得が最大となるような特徴的な部分グラフ（以下，基本部分グラフと呼ぶ）をいくつか埋め込んだ．各データセットのグラフサイズを表 1 にまとめる．

4.2 実験結果と考察

実験結果を表 2，3 に示す．表はそれぞれチャンキング指標を頻度と情報利得のいづれかにした場合の計算時間と抽出した部分グラフ中の情報利得の最大値（MaxIG）の関係を示している．慢性肝炎データセットでは，両方の枝刈りを行う手法は枝刈りを行わない手法に比べて，同じ情報利得を得るまでの計算時間が，チャンキング指標が頻度の場合は約 20%，情報利得を用いた場合は最大 80%削減されている．また，人工データセットでは，チャンキング指標に頻度を用いた場合は最大 90%，情報利得を用いた場合は最大 75%計算時間が削減されている．特にチャンキング指標に頻度を用いた場合に慢性肝炎データセットよりも大幅に計算時間を削減している．

表 1: 各データセットのグラフのサイズ

データセット	慢性肝炎データセット		人工データセット	
クラス	$N90$	$R90$	*positive*	*negative*
グラフ数	56	38	150	150
平均頂点数	112	104	50	50
頂点数の合計	6,296	3,944	7,524	7,502
頂点ラベル数	12		20	
平均辺数	117	108	498	495
辺数の合計	6,577	4,090	74,631	74,198
辺ラベル数	30		20	

チャンキング指標に頻度を用いた場合，事前枝刈りの効果は薄く，慢性肝炎データセットに対しては事前枝刈りの効果が全くなく，逆に条件判定のためにより多くの計算時間を要している．この原因として，事前枝刈りは（擬似ノードを含む）ノードの頻度が低くなければ効果が出にくい点が上げられる．単一ノードは元々頻度が高い上に，頻度の高い順に擬似チャンキングされるため擬似ノードの頻度も高くなり，このような結果となったと考えられる．人工データセットに対して事前枝刈りが有効であったのは，人工データセットの頂点ラベル数と辺ラベル数が，グラフ 1 枚あたりの平均頂点数と平均辺数に比べて大きいため，ノードの頻度が低くなったためと考えられる．

事後枝刈りの効果の差に関しても同様の原因が考えられる．しかし，事前枝刈りが頂点ラベル数のグラフ 1 枚あたりの平均頂点数に対する比率が重要だったのに対し，ノードペアを構成する各ノードに比べてノードペアの頻度が低くなると事後枝刈りされやすいという観点から，辺ラベル数のグラフ 1 枚あたりの平均辺数に対する比率が重要であると思われる．

4.3 検証実験

ここで，頂点ラベル数と辺ラベル数の違いが枝刈りの効果にどれだけの影響を与えるかを検証すべく表 4 に示す人工データセットを用いて検証実験を行った．人工データセット α（D_α）はラベル数の少ないデータセットであり，人工データセット β（D_β）は逆にラベル数の多いデータセットである．この 2 つのデータセットに対して前述の実験と同様の設定で，$N = 5$，及び $N = 10$ の場合について実験を行ったところ結果は表 5，6 のようになった．結果より，事前枝刈りは D_α において効果がないが，D_β では有効であることや，計算時間の削減率も D_β に対しての方が大きいことから，前節における考察を裏付けていると言える．また，D_α に対してチャンキング指標を情報利得とし，$N = 10$ で探索したものの計算時間が非常に大きなものになっているが，これは枝刈りによって頻度の低いペアが探索から除外され，代わりに頻度の高いペアが擬似チャンキングされたことにより，次のレベルで生成されるペアの数が増加したことが原因であると考えられる．

以上の結果と考察をまとめると，チャンキング指標に関しては頻度より情報利得の方が枝刈りの効果が得られる場合が多く，グラフの構造に関しては頂点ラベル数と辺ラベル数がグラフ 1 枚あたりの平均頂点数と平均辺数に比べて大きい方が枝刈りの効果が大きくなるといえる．

5. まとめ

本稿では，ある部分グラフの拡大グラフが取る情報利得の上界が計算可能な点に着目し，Cl-GBI 法の探索を効率化するた

表 2: 肝炎データにおける計算時間と抽出した部分グラフ中の情報利得の最大値（MaxIG）

チャンキング指標	頻度						情報利得					
	$N=5$		$N=10$		$N=15$		$N=5$		$N=10$		$N=15$	
	時間 [秒]	MaxIG	時間 [秒]	MaxIG	時間 [秒]	MaxIG	時間 [秒]	MaxIG	時間 [秒]	MaxIG	時間 [秒]	MaxIG
枝刈りなし	924	0.1034	11,253	0.1139	48,254	0.1139	66	0.1890	534	0.1890	1,894	0.1890
事前枝刈り	953	0.1034	11,403	0.1139	48,556	0.1139	42	0.1890	380	0.1890	1,430	0.1890
事後枝刈り	718	0.1034	8,831	0.1139	37,802	0.1139	35	0.1890	140	0.1890	362	0.1890
両枝刈り	743	0.1034	8,968	0.1139	38,117	0.1139	34	0.1890	137	0.1890	353	0.1890

表 3: 人工データにおける計算時間と抽出した部分グラフ中の情報利得の最大値（MaxIG）

チャンキング指標	頻度						情報利得					
	$N=5$		$N=10$		$N=15$		$N=5$		$N=10$		$N=15$	
	時間 [秒]	MaxIG	時間 [秒]	MaxIG	時間 [秒]	MaxIG	時間 [秒]	MaxIG	時間 [秒]	MaxIG	時間 [秒]	MaxIG
枝刈りなし	4,728	0.0421	17,486	0.2042	30,738	0.2042	1,286	0.2042	2,850	0.2042	3,701	0.2042
事前枝刈り	4,712	0.0421	17,211	0.2042	21,963	0.2042	1,242	0.2042	2,048	0.2042	2,467	0.2042
事後枝刈り	1,429	0.0421	3,013	0.2042	3,185	0.2042	597	0.2042	788	0.2042	1,183	0.2042
両枝刈り	1,377	0.0421	2,507	0.2042	2,719	0.2042	602	0.2042	675	0.2042	1,007	0.2042

表 4: 検証用人工データセットの概要

データセット		人工データセット α		人工データセット β	
クラス		P_α	N_α	P_β	N_β
グラフ数		150	150	150	150
平均頂点数		50	50	50	50
頂点数の合計		7,469	7,493	7,469	7,493
頂点ラベル数		10		20	
平均辺数		367	366	365	366
辺数の合計		54,989	54,948	54,739	54,968
辺ラベル数		10		20	
基本部分グラフの	平均頂点数	4	0	4	0
	頂点ラベル数	10		20	
	平均辺数	3	0	3	0
	辺ラベル数	10		20	

表 5: 人工データセット α の検証実験結果

	$N=5$		$N=10$	
	時間 [秒]	MaxIG	時間 [秒]	MaxIG
枝刈りなし（頻度）	8,934	0.3328	89,792	0.3328
事前枝刈り（頻度）	9,055	0.3328	90,806	0.3328
事後枝刈り（頻度）	3,501	0.3328	21,258	0.3328
両枝刈り（頻度）	3,555	0.3328	20,311	0.3328
枝刈りなし（情報利得）	210	0.3328	459	0.3328
事前枝刈り（情報利得）	94	0.3328	104	0.3328
事後枝刈り（情報利得）	543	0.3328	4,194	0.3328
両枝刈り（情報利得）	423	0.3328	2,146	0.3328

めにその上界を利用した枝刈り手法を提案した．提案手法は，Cl-GBI 法の部分グラフ抽出過程における頻度計算の負荷を軽減するとともに，不要な部分グラフを探索過程で破棄することでメモリ消費量も大幅に軽減できる．また，提案手法の特性として，頂点ラベル数と辺ラベル数がグラフ 1 枚あたりの平均頂点数と平均辺数に比べて大きいデータセットに対して非常に効率的な探索を行うことができる．

今後の課題としては，枝刈りの効果がデータに依存するため，さらに多様な人工データを用いたより詳細な特性解析が必要である．また，実データにおいてより高い情報利得をもつ部分グラフを効率的に探索するためには，今後，擬似チャンキングするペアの選定基準に関しても検討を重ねる必要がある．

表 6: 人工データセット β の検証実験結果

	$N=5$		$N=10$	
	時間 [秒]	MaxIG	時間 [秒]	MaxIG
枝刈りなし（頻度）	871	0.3438	3,867	0.3438
事前枝刈り（頻度）	185	0.3438	239	0.3438
事後枝刈り（頻度）	159	0.3438	158 *	0.3438
両枝刈り（頻度）	158	0.3438	157 *	0.3438
枝刈りなし（情報利得）	283	0.3438	401	0.3438
事前枝刈り（情報利得）	230	0.3438	237	0.3438
事後枝刈り（情報利得）	158	0.3438	159 *	0.3438
両枝刈り（情報利得）	159	0.3438	159 *	0.3438

* レベル 5 にて探索終了

参考文献

[Breiman 84] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*, Wadsworth & Brooks/Cole Advanced Books & Software (1984).

[Cook 94] D. J. Cook and L. B. Holder, *Substructure Discovery Using Minimum Description Length and Background Knowledge*, Artificial Intelligence Research, Vol.1, pp.231–255 (1994).

[Geamsakul 05] W. Geamsakul, T. Yoshida, K. Ohara, H. Motoda, H. Yokoi, and K. Takabayashi, *Constructing a Decision Tree for Graph-Structured Data and its Applications*, Fundamenta Informaticae, IOS Press, Vol.66, No.1-2, pp.131–160(2005).

[Inokuchi 03] A. Inokuchi, T. Washio, and H. Motoda, *Complete Mining of Frequent Patterns from Graphs: Mining Graph Data*, Machine learning, Vol. 50, No. 3, pp. 321-354 (2003).

[Matsuda 02] T. Matsuda, H. Motoda, T. Yoshida, and T. Washio, *Mining Patterns from Structured Data by Beam-wise Graph-Based Induction*, Proc. DS 2002, pp. 422-429 (2002).

[Morishita 00] S. Morishita and J. Sese, *Traversing Itemset Lattices with Statical Metric Pruning*, In Proc. of the 19th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, pp226–236 (2000).

[Nguyen 05] P. C. Nguyen, K. Ohara, H. Motoda, and T. Washio, *Cl-GBI: A Novel Approach for Extracting Typical Patterns from Graph-Structured Data*, Proc. of PAKDD 2005, pp.639–649 (2005).

[Nguyen 06] P. C. Nguyen, K. Ohara, A. Mogi, H. Motoda, and T. Washio, *Constructing Decision Trees for Graph-Structured Data by Chunkingless Graph-Based Induction*, Proc. of PAKDD 2006, pp.390–399, (2006).

[Quinlan 86] J. R. Quinlan, *Induction of decision trees*, Machine Learning, Vol. 1, pp. 81–106 (1986).

[山口 02] 山口高平, 慢性肝炎データセットのクレンジングとマイニングの試み, 平成 13 年度科学研究費補助金 特定領域 (B) 研究成果報告書, 情報洪水時代におけるアクティブマイニングの実現, pp. 205–221 (2002).

[Yan 02] X. Yan and J. Han, *gSpan:Graph-Based Structure Pattern Mining*, In: Proc. of the 2nd IEEE International Conference on Data Mining(ICDM2002), pp.840–847 (2002).

[Yoshida 95] K. Yoshida and H. Motoda, *CLIP: Concept Learning from Inference Patterns*, Artificial Intelligence, Vol. 75, No. 1, pp. 63–92 (1995).