グラフ時系列データからの頻出部分系列マイニング手法の性能評価

Performance Evaluation of Methods to Mine Frequent Subsequences from Graph Sequences

猪口 明博[*1*2] 鷲尾 隆[*1]
Akihiro Inokuchi Takashi Washio

[*1]大阪大学 産業科学研究所
The Institute of Scientific and Industrial Research, Osaka University

[*2]科学技術振興機構 さきがけ
PRESTO, Japan Science and Technology Agency

There are many real-world applications suitable to model objects by using graph sequences. For example, a human network is represented by a graph where each human and each relationship between two humans correspond to a vertex and an edge, respectively. If a person joins or leaves a human community, the numbers of vertices and edges in the graph increase or decrease. Similarly, a gene network consisting of genes and their interactions produces a graph sequence in their evolutionary history. We have proposed a method, call GTRACE, to mine frequent subsequences from graph sequence data. In this paper, we comparer GTRACE with Dynamic GREW which is a past graph subsequence mining algorithm to evaluate their efficiency by applying to a real world dataset.

1. はじめに

膨大なデータから有用な，あるいは興味のあるパターンを知識として発掘するデータマイニングの研究が盛んに行われている．有用性は人それぞれ異なるので定義するのは難しいが，一般に多くの事例を説明できる知識は有用と考えられる [5]．複数のアイテム集合のデータから頻出アイテム集合を列挙する Apriori アルゴリズムが提案されて以来，様々なデータ構造に対して頻出パターン列挙アルゴリズムが提案されている．近年では，頂点間連結関係と頂点や辺ラベルの情報からなるグラフ構造に頻出する部分グラフパターン [4] をマイニングする手法が提案されている．提案されているグラフマイニング手法は実用上非常に効率的であるが，部分グラフ同型問題が NP 完全であるため，より大きな部分グラフをマイニングするのに多くの計算時間を要する．従って，既存手法をグラフ系列のような複数グラフからなる大きなグラフに対して適用することは困難である．

しかしながら，グラフの系列によるモデル化が適している実世界の対象は多く存在する．例えば，人間関係ネットワークは人が頂点，関係が辺であるグラフで表現でき，人がコミュニティ（ネットワーク）に参加，脱退することで頂点や辺が増減する．同様に，遺伝子が頂点，相互関係が辺である遺伝子ネットワークは，進化の過程で遺伝子が新規獲得されたり，欠落，突然変異するグラフの系列で表現できる．

このようなデータ解析上のニーズを背景として，グラフ系列からグラフ部分系列をマイニングする手法 Dynamic GREW[1] が提案された．Dynamic GREW は，頂点数が変化しないグラフ系列を対象として，辺の変化をビット列で表すことで，グラフ部分系列のマイニングを可能にした．一方，我々は，グラフ系列をマイニングする手法 GTRACE (Graph TRAnsformation sequenCE mining) を提案した [3]．GTRACE は，頂点数が変化するグラフ系列の中で連続する 2 つのグラフの差異を変換規則で表すことで，グラフ部分系列を列挙する．更に，グラフ系列中でどの頂点同士が関連するかを定義する和グラフを導入し，関連性のある頂点のみから成るグラフ部分系列をマイニングする．本論文では，GTRACE と Dynamic GREW の計算時間，導かれるグラフ部分系列数，1 グラフ部分系列導出あたりの計算時間を比較した性能評価結果を報告する．

連絡先: 猪口 明博，大阪大学 産業科学研究所，大阪府茨木市美穂ケ丘 8-1，inokuchi@ar.sanken.osaka-u.ac.jp

2. GTRACE

図 1(a) は観測されたグラフ系列の例を表している．GTRACE は，図 1(a) に示すグラフ系列の集合から，それらに頻出する図 1(b) のような系列を列挙する手法である．GTRACE が対象とするグラフ系列は，以下を満たすグラフの系列である．

- 系列中でグラフの頂点数や辺数が増減する．
- 系列中で頂点ラベルや辺ラベルが変わる．
- 観測グラフ系列の中の連続する 2 つのグラフ $g^{(j)}$ と $g^{(j+1)}$ 間でその構造のごく一部のみが変化する．
- 各グラフは疎グラフである．

例えば，一度に大半の人間や遺伝子が入れ替わることはなく，更に各時点では個々の人間や遺伝子は他の一部としか関係を持たない人間関係ネットワークや遺伝子ネットワークのように，実世界の多くのグラフ変化は，これらの仮定を満たしている．

2.1 グラフ系列の表現形式

グラフ系列中で連続する 2 つのグラフのごく一部が変化するという仮定より，各グラフ $g^{(j)}$ をその全頂点，及びその間の辺で直接表す方法は冗長である．部分系列を効率よく探索するためには，計算コストと空間コストを抑えるためのグラフ系列の簡潔な表現が必要となる．そこで本節では，GTRACE が用いるグラフ系列の表現形式を説明する．

ラベル付きグラフ g を $g = (V, E, L, f)$ で表す．ここで，$V = \{v_1, v_2, \cdots, v_z\}$ は頂点集合，$E = \{(v, v') \mid (v, v') \in V \times V\}$ は辺集合，L は頂点と辺のラベル集合であり，$f : V \cup E \rightarrow L$ である．グラフ g の頂点集合，辺集合，ラベル集合を $V(g), E(g), L(g)$ と表す．また観測グラフ系列を $d = \langle g^{(1)}\, g^{(2)} \cdots g^{(n)} \rangle$ と表す．$g^{(j)}$ は j 番目に観測されたグラフである．$g^{(1)}$ を系列の先頭，$g^{(n)}$ を系列の末尾とする．グラフの各頂点 v はユニーク

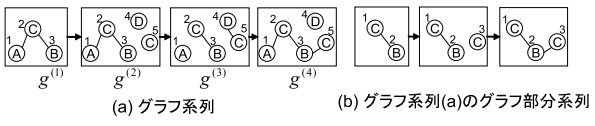

図 1: 観測グラフ系列とそのグラフ部分系列の例

表 1: グラフ系列データのための変換規則

頂点追加 $vi^{(j,k)}_{[u,l]}$	ラベルが l，ユニーク ID が u である頂点を $g^{(j,k)}$ へ追加し，$g^{(j,k+1)}$ へ変換
頂点削除 $vd^{(j,k)}_{[u,\bullet]}$	ユニーク ID が u である頂点を $g^{(j,k)}$ から削除し $g^{(j,k+1)}$ へ変換
頂点ラベル変更 $vr^{(j,k)}_{[u,l]}$	ユニーク ID が u である頂点のラベルを l に変更し，$g^{(j,k)}$ を $g^{(j,k+1)}$ へ変換
辺追加 $ei^{(j,k)}_{[(u_1,u_2),l]}$	ユニーク ID が u_1 と u_2 である頂点間にラベル l の辺を追加し，$g^{(j,k)}$ を $g^{(j,k+1)}$ へ変換
辺削除 $ed^{(j,k)}_{[(u_1,u_2),\bullet]}$	ユニーク ID が u_1 と u_2 である頂点間から辺を削除し，$g^{(j,k)}$ を $g^{(j,k+1)}$ へ変換
辺ラベル変更 $er^{(j,k)}_{[(u_1,u_2),l]}$	ユニーク ID が u_1 と u_2 である頂点間の辺のラベルを l へ変更し，$g^{(j,k)}$ を $g^{(j,k+1)}$ へ変換

ID をもち，$id(v)$ と表す．頂点集合と辺集合に対するユニーク ID の集合を以下のように定義する．

$$ID(V) = \{id(v)|v \in V\}$$
$$ID(E) = \{(id(v), id(v'))|(v, v') \in E\}$$

グラフ系列を簡潔に表現するため，グラフ系列中の連続する 2 つのグラフ $g^{(j)}$ と $g^{(j+1)}$ の差異に着目する．

定義 1 観測グラフ系列 $d = \langle g^{(1)} g^{(2)} \cdots g^{(n)} \rangle$ の各グラフ $g^{(j)}$ を外部状態と呼ぶ．さらに，連続する *2* つのグラフ $g^{(j)}$ と $g^{(j+1)}$ の間を補間するグラフ系列を $s^{(j)} = \langle g^{(j,1)} g^{(j,2)} \cdots g^{(j,m_j)} \rangle$ で表し，各 $g^{(j,k)}$ を内部状態と呼ぶ．ただし，$g^{(j,1)} = g^{(j)}$ かつ $g^{(j,m_j)} = g^{(j+1)}$ とする．観測グラフ系列 d は補間系列 $d = \langle s^{(1)} s^{(2)} \cdots s^{(n-1)} \rangle$ で表される． ■

外部状態の順序は観測グラフ系列中のグラフの順序であるが，内部状態の順序は人工的に補間されたグラフの順序であり，$g^{(j)}$ と $g^{(j+1)}$ の間に様々な補間系列が考えられる．GTRACE は，グラフ系列マイニングの計算コストと空間コストを抑えるために，グラフ編集距離に基づき最短の補間系列を選択する．

定義 2 頂点や辺の追加，削除，ラベル変更を変換の最小単位とし，それらの変換を編集距離 *1* とする．内部状態系列 $s^{(j)} = \langle g^{(j,1)} \cdots g^{(j,m_j)} \rangle$ の連続する *2* つの内部状態の編集距離は *1* である．また，内部状態系列中の任意の *2* つの内部状態の編集距離は最小である． ■

本稿では，最小単位の変換を変換規則を用いて表す．

定義 3 $g^{(j,k)}$ を $g^{(j,k+1)}$ へ変換する変換規則を $tr^{(j,k)}_{[o_{jk},l_{jk}]}$ で表す．ただし，$o_{jk} \in ID(V) \cup ID(E)$，$l_{jk} \in L$ である．

- tr は頂点や辺の追加，削除，ラベル変更のいずれか．
- o_{jk} は変換される頂点や辺のユニーク *ID*.
- l_{jk} は変換される頂点や辺のラベル． ■

本稿では簡単化のため変換規則 $tr^{(j,k)}_{[o_{jk},l_{jk}]}$ を $tr^{(j,k)}_{[o,l]}$ と略記する．GTRACE が用いる 6 種の変換規則を表 1 に示す．例えば，j 番目の外部状態の k 番目と $k+1$ 番目の内部状態間で，ラベルが l でユニーク ID が u である頂点の追加を $vi^{(j,k)}_{[u,l]}$ で表す．頂点削除と辺削除はユニーク ID のみの指定で変換可能であるので，変換規則の引数 l はダミー辺数であり，'$\bullet$' で表す．

以上より，変換系列を以下のように定義する．

定義 4 内部状態系列 $s^{(j)} = \langle g^{(j,1)} g^{(j,2)} \cdots g^{(j,m_j)} \rangle$ を変換規則を用いて $seq(s^{(j)}) = \langle tr^{(j,1)}_{[o,l]} tr^{(j,2)}_{[o,l]} \cdots tr^{(j,m_j-1)}_{[o,l]} \rangle$ と表し，内部状態変換系列と呼ぶ．さらに，外部状態系列 $d = \langle g^{(1)} \cdots g^{(n)} \rangle$ を内部状態変換系列の系列である外部状態変換系列 $seq(d) = \langle seq(s^{(1)}) seq(s^{(2)}) \cdots seq(s^{(n-1)}) \rangle$ で表す． ■

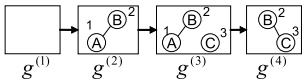

図 2: 外部状態系列

このような変換系列によるグラフ系列の表記は，グラフが徐々に変化するという仮定の下で，連続するグラフの差異のみに注目した表現形式であるので，グラフによる直接の系列表記に比べ簡潔である．また，如何なるグラフ系列も表 1 に示す 6 種の変換規則で表現可能である．

2.2 頻出変換部分系列のマイニング

本節ではグラフ系列の集合から頻出変換部分系列をマイニングする手法を示す．2.1 節で説明した外部状態の系列から頻出変換部分系列をマイニングするために，変換系列の包含関係を以下のように定義する．

定義 5 $seq(s^{(j)})$, $seq(s'^{(h)})$ を内部状態変換系列とする．

$$seq(s^{(j)}) = \langle tr^{(j,1)}_{[o,l]} \cdots tr^{(j,k)}_{[o,l]} \cdots tr^{(j,m_j-1)}_{[o,l]} \rangle \quad (1)$$
$$seq(s'^{(h)}) = \langle tr^{(h,1)}_{[o,l]} \cdots tr^{(h,r)}_{[o,l]} \cdots tr^{(h,m_h-1)}_{[o,l]} \rangle \quad (2)$$
$$\forall tr^{(h,r)}_{[o,l]} \in seq(s'^{(h)}), \exists tr^{(j,k)}_{[o,l]} \in seq(s^{(j)}); tr^{(h,r)}_{[o,l]} = tr^{(j,k)}_{[o,l]}$$

上記の式を満たすとき，$seq(s'^{(h)})$ は $seq(s^{(j)})$ の部分系列と呼び，$seq(s'^{(h)}) \subseteq seq(s^{(j)})$ と表す． ■

内部状態変換系列は人工的に補間された系列であるので，$seq(s^{(j)})$ と $seq(s'^{(h)})$ 中の変換規則の順序は必ずしも一致しない．上記に加え，変換系列全体の包含関係を以下に定義する．

定義 6 以下の $seq(d)$ と $seq(d')$ を変換系列とする．

$$seq(d) = \langle seq(s^{(1)}) \cdots seq(s^{(j)}) \cdots seq(s^{(n-1)}) \rangle$$
$$seq(d') = \langle seq(s'^{(1)}) \cdots seq(s'^{(h)}) \cdots seq(s'^{(n'-1)}) \rangle$$

ただし，$seq(s^{(j)})$ と $seq(s'^{(h)})$ はそれぞれ式 *(1)* と式 *(2)* で与えられる． もし $h = 1, \cdots, n'-1$ に対して，$seq(s'^{(h)}) \subseteq seq(s^{(j_h)})$ であるような整数 $1 \le j_1 < \cdots < j_{n'-1} \le n-1$ が存在するなら，$seq(d')$ は $seq(d)$ の部分系列と呼び，$seq(d') \sqsubseteq seq(d)$ と書く． ■

例 1 図 *2* の外部状態系列は $seq(d) =$

$$\langle vi^{(1,1)}_{[1,A]} \underline{vi^{(1,2)}_{[2,B]} ei^{(1,3)}_{[(1,2),-]}} vi^{(2,1)}_{[3,C]} ed^{(3,1)}_{[(1,2),\bullet]} \underline{vd^{(3,2)}_{[1,\bullet]} ei^{(3,3)}_{[(2,3),-]}} \rangle$$

と表される．以下の系列 $seq(d')$ は $seq(d)$ の部分系列であり，$seq(d')$ は $seq(d)$ 中の下線部に対応する．

$$seq(d') = \langle vi^{(1,1)}_{[1,B]} ei^{(1,2)}_{[(1,3),-]} vd^{(2,1)}_{[3,\bullet]} ei^{(2,2)}_{[(1,2),-]} \rangle$$

グラフ系列の集合 $DB = \{d_i | d_i = \langle g_i^{(1)} \cdots g_i^{(n_i)} \rangle\}$ に対し，変換部分系列 $seq(d')$ の支持度 $\sigma(seq(d'))$ を

$$\sigma(seq(d')) = \frac{|\{d_i | d_i \in DB, seq(d') \sqsubseteq seq(d_i)\}|}{|DB|}$$

と定義する．最小支持度 σ' 以上の支持度を有する部分系列を頻出変換部分系列 (Frequent Transformation Subsequence: FTS) と呼ぶ．関連研究同様，$seq(d'_1) \sqsubset seq(d'_2)$ ならば $\sigma(seq(d'_1)) \ge \sigma(seq(d'_2))$ である支持度の逆単調性が成り立つ．以上の定義により，グラフ系列マイニングを以下のように定義する．

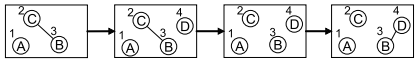

図 3: 関連性のない頂点を含む外部状態系列

問題 1 グラフ系列の集合 $DB = \{d_i | d_i = \langle g_i^{(1)} g_i^{(2)} \cdots g_i^{(n_i)} \rangle\}$ と最小支持度 σ' が入力として与えられたとき，DB 中の頻出変換部分系列を全て列挙する．

GTRACE は，FTS の末尾に深さ優先探索で変換規則を付加する FrefixSpan[6] を用いて，$seq(DB)$ から全 FTS を列挙する．

2.3 関連性のある FTS のマイニング

2.2 節では全ての頻出変換部分系列を列挙するアルゴリズムを示した．GTRACE は，実用性の観点から出力される系列中の頂点と辺が互いに関連がある系列 (Relevant FTS : rFTS) のみを列挙する．例えば，図 3 のグラフ系列では，ラベルが A でユニーク ID が 1 である頂点は，どの外部状態においても他の頂点と連結していないため，他の頂点と関連がないと考える．一方，頂点 2 と頂点 4 はどの外部状態においても直接は接続していないが，それらの頂点はラベル B をもつ頂点 3 と，1 番目の外部状態と 4 番目の外部状態でそれぞれ連結している．この場合，本稿では頂点 2 と 4 は頂点 3 を介して互いに関連があると考える．このように，図 3 における関連性のある系列の例として，頂点 2，3，4 を含み，頂点 1 を含まないものが考えられる．以上の外部状態系列の連結性の議論に基づいて，頂点と辺のユニーク ID の関連性を以下に定義する．

定義 7 外部状態系列 $d = \langle g^{(1)} g^{(2)} \cdots g^{(n)} \rangle$ に対し，ラベルを持たない d の和グラフ $g_u(d) = (V(g_u(d)), E(g_u(d)))$ を以下のように定義する．

$$V(g_u(d)) = \bigcup_{j=1,\cdots,n} \{id(v) | v \in V(g^{(j)})\}$$

$$E(g_u(d)) = \bigcup_{j=1,\cdots,n} \{(id(v), id(v')) | (v, v') \in E(g^{(j)})\} \quad \blacksquare$$

定義 7 に基づき，ユニーク ID 間の関連性を定義する．

定義 8 外部状態系列 d の和グラフが連結グラフであるとき，d のユニーク *ID* は互いに関連がある． ■

和グラフは変換系列に対しても同様に定義される．GTRACE は和グラフが連結である rFTS のみを列挙する．

GTRACE は効率良く rFTS を列挙するため，はじめに，定義 7 に基づいて DB 中のグラフ系列の和グラフを計算する．次に，和グラフの集合から AcGM[4] を用いて，頻出連結部分グラフを取り出す．頻出連結部分グラフ g_u が取り出されるたびに，2.2 節で述べた PrefixSpan を呼び出す．さらに，PrefixSpan により列挙された FTS の和グラフが g_u と同型ならば，それを rFTS として出力する．

3. Dynamic GREW

本稿では，Dynamic GREW[1] を改良した Dynamic GREW2 を用いて，GTRACE との比較実験を行う．Dynamic GREW2 が対象とするデータは，GTRACE と同様に グラフ系列の集合 $DB = \{d_i | d_i = \langle g_i^{(1)} \cdots g_i^{(n_i)} \rangle\}$ で表される*1．ただし，Dynamic GREW2 が対象とするグラフ系列と 2 節のはじめで述べた GTRACE が対象とするグラフ系列では，以下が異なる．

*1 オリジナルの Dynamic GREW が対象とするデータは，グラフ系列の集合ではなく，1 本のグラフ系列である．本稿では，GTRACE との比較のために Dynamic GREW を改良した Dynamic GREW2 を用いる．

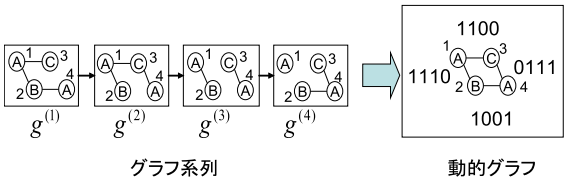

図 4: グラフ系列とその動的グラフ

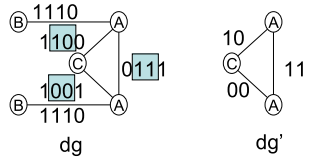

図 5: 動的グラフとその動的部分グラフ

- 系列中で辺数は増減するが，頂点数は増減しない．
- グラフの頂点のみがラベルをもち，辺はラベルを持たない．
- 系列中で頂点ラベルは変化しない．

グラフ系列 $d = \langle g^{(1)} g^{(2)} \cdots g^{(n)} \rangle$ が与えられたとき，d に対する動的グラフを $dg(d) = (V, E, L, l, es)$ と定義する．ここで，V はグラフ系列中のグラフの頂点集合で，$V = V(g^{(1)}) = \cdots = V(g^{(n)})$ である．すなわち，頂点数は系列中で変化しない．E は辺の集合であり，$E = \bigcup_{j=1,\cdots,n} E(g^{(j)})$ である．L は頂点ラベルの集合であり，系列中で頂点ラベルは変化しない．l は頂点にラベルを割り当てる関数である．es は辺 $(v_1, v_2) \in E(dg)$ に長さ n のビット列を割り当てる関数 $es : E \rightarrow \{0|1\}$ であり，$g^{(j)}$ に辺 (v_1, v_2) が存在するとき，辺 (v_1, v_2) の j ビット目に 1 を割り当て，辺が存在しないとき 0 を割り当てる．図 4 の $g^{(1)}$，$g^{(2)}$，$g^{(3)}$，$g^{(4)}$ では，ユニーク ID が 1 である頂点とユニーク ID が 3 である頂点の間には，それぞれ辺が存在する，存在する，存在しない，存在しないので，動的グラフでそれらの頂点間の辺には 1100 というビット列が割り当てられる．

2 つのグラフ $dg(V, E, L, l, es)$ と $dg'(V', E', L', l', es')$ が与えられ，$\forall v, v_1, v_2 \in V'$ に対して，以下を満たす単射 $\phi : V' \rightarrow V$ が存在するとき，dg' を dg の動的部分グラフと呼び，$dg' \sqsubseteq dg$ と表す．

1. $(\phi(v_1), \phi(v_2)) \in E$ if $(v_1, v_2) \in E'$ (グラフ g' の辺がグラフ g に存在)
2. $l'(v) = l(\phi(v))$ (頂点ラベルの一致)
3. $es'((v_1, v_2)) = substr(es((\phi(v_1), \phi(v_2))), i, k)$ (辺に対するビット列の包含関係)

ここで $substr(s, i, k)$ は，s の部分文字列であり，部分文字列は s の i 番目のビットから始まる長さ k のビット列とする．GTRACE が対象とする変換規則系列の包含関係では，部分系列は元の系列において連続している必要はない．一方，Dynamic GREW2 が対象とする動的グラフのビット列では，部分ビット列が元のビット列において連続している必要がある．図 5 の dg' は dg の動的部分グラフである．dg' の各辺のビット列は，dg の網がけされた部分に対応している．

動的部分グラフ dg' の支持度は，$\sigma(dg') = |\{d_i | d_i \in DB, dg' \sqsubseteq dg(d_i)\}| / |DB|$ と定義され，最小支持度以上の支持度をもつ動的部分グラフを頻出動的部分グラフと呼ぶ．Dynamic GREW2 が対象とする問題は以下である．

問題 2 グラフ系列の集合 $DB = \{d_i | d_i = \langle g_i^{(1)} g_i^{(2)} \cdots g_i^{(n_i)} \rangle\}$ と最小支持度 σ' が入力として与えられたとき，DB 中の頻出動的部分グラフを全て列挙する．

表 2: 最小支持度を変化させたときの実験結果

最小支持度	計算時間 [sec]		頻出部分系列数		1 頻出部分系列導出あたりの計算時間		t_2/t_1
	GT	DG	GT	DG	GT	DG	
20%	73.547	0.108	3,346	190	0.0220	0.00057	82.0%
21%	46.703	0.077	2,827	177	0.0165	0.00044	79.6%
22%	25.313	0.090	2,081	153	0.0122	0.00059	69.4%
23%	15.485	0.110	1,754	141	0.0088	0.00078	58.0%
24%	5.001	0.094	1,495	132	0.0033	0.00071	51.8%
25%	3.921	0.062	1,257	126	0.0031	0.00049	0%
30%	1.140	0.078	524	67	0.0022	0.00126	0%
35%	0.280	0.046	196	39	0.0014	0.00127	0%
40%	0.156	0.031	106	19	0.0015	0.00163	0%

表 3: ユニーク ID 数を変化させたときの実験結果

ユニーク ID 数	計算時間 [sec]		頻出部分系列数		1 頻出部分系列導出あたりの計算時間	
	GT	DG	GT	DG	GT	DG
80	0.016	0.015	12	0	0.0013	–
100	0.141	0.015	26	0	0.0054	–
120	0.562	0.046	206	26	0.0027	0.0018
140	0.813	0.047	254	35	0.0032	0.0013
160	14.750	0.093	790	71	0.0187	0.0013
182	87.140	0.109	1,376	82	0.0633	0.0013

4. 実験，及び考察

GTRACE と Dynamic GREW2 を比較するために，エンロン社電子メールデータ [2] を用いた．このデータは，1998 年 11 月 15 日 (日) から 2001 年 3 月 25 日 (土) までの 123 週の 182 人の人間間の電子メールやり取りのデータである．182 人それぞれが固有の名前，すなわち，ユニーク ID を持ち，ある 2 人が 1 日の間に電子メールでコミュニケーションをとると 2 頂点間に辺を張り，ある 1 日に対応するグラフ $g^{(j)}$ を生成した．また，各頂点には，CEO, Director, Employee, Lawyer, Manager, President, Trader, Vice President のいずれかが頂点ラベルとして割り当てられている．各週を単位として各々 1 つのグラフ系列を生成した．すなわち，入力となるグラフ系列数は 123 である．

表 2 はランダムに選択した 100 個のユニーク ID から構成されるグラフ系列の集合を対象として，最小支持度を変化させたときの計算時間，導出された頻出部分系列数，1 頻出部分系列導出あたりの平均計算時間を示している．表 3 は，最小支持度を 50% とし，選択するユニーク ID 数を変化させたときの実験結果である．GT 及び DG はそれぞれ，GTRACE と Dynamic GREW2 の結果を表している．最小支持度を減少させたとき，あるいはユニーク ID 数を増加させたとき，GT と DG が導出する頻出頻出部分系列の数が増加し，GT と DG の計算時間も増加する．2 つの手法により導出される頻出頻出部分系列が異なるため，1 頻出部分系列あたりの平均計算時間で両者を比較すると，DG の計算時間のほうが小さい．これは，DG では，部分ビット列が元のビット列において連続したビット列であるために，共通する部分ビット列を，接尾辞木を用いて入力ビット長に対して線形時間で列挙することが可能であるからである．一方，GTRACE が列挙する FTS は，元の変換規則系列で必ずしも連続ではないので，入力系列長の線形時間で解けないためである．

GTRACE の全計算時間を t_1 とし，GTRACE で呼び出される複数回の PrefixSpan のうち，rFTS を出力しない PrefixSpan を実行するのに要する計算時間を t_2 とする．表 2 の最後の列は，t_1 に対する t_2 の割合を表している．2.3 節の最後で述べたように，GTRACE は，DB 中のグラフ系列の和グラフの集合を得て，AcGM を用いてその和グラフの集合から頻出連結部分グラフを取り出す．さらに，頻出連結部分グラフ g_u が取り出されるたびに PrefixSpan を呼び出し，PrefixSpan により列挙された FTS の和グラフが g_u と同型ならば，それを rFTS として出力する．各 PrefixSpan の実行で列挙される FTS が必ずしも，g_u と同型となるとは限らないので，rFTS を 1 つも出力しない場合がある．最小支持度を下げると AcGM により出力される頻出連結部分グラフの頂点数は大きくなる．一方，g_u の頂点数が多くなると，和グラフが g_u と同型になる変換規則系列の種類は多くなるので，各変換規則系列の支持度は小さくなる．このため，最小支持度を下げると，PrefixSpan が 1 つも rFTS を出力しない場合が起こる．このような PrefixSpan の実行を避けるような探索空間の枝刈りが可能となれば，計算時間の短縮が可能となる．例えば，表 2 で用いたデータで，最小支持度を 20%に設定したとき，最後の 1 回の PrefixSpan の実行を回避する探索空間の枝刈りが可能となれば，計算時間は 73 秒から 13 秒程度に短縮できる．

以下の変換規則系列は GTRACE によって列挙された FTS であり，図 6 はこの FTS を図示したものである．

$$\langle vi^{(1,1)}_{[1,CEO]} vi^{(1,2)}_{[2,VicePre]} vi^{(1,3)}_{[3,CEO]} ei^{(2,1)}_{[(1,2),1]} ei^{(2,2)}_{[(1,4),1]} ed^{(3,1)}_{[(1,3),1]} \rangle$$

上記の系列で $VicePre$ は Vice President を表している．この系列は，26 個のグラフ系列に出現していた．すなわち，123 週のうち 26 週でこのようなコミュニケーションがとられていた．図 6 では，ユニーク ID が 4 の頂点が $g^{(1)}$ で追加された（出現した）ように描かれているが，この FTS には頂点を追加する変換規則 $vi^{(j,k)}_{[4,l]}$ が含まれていないために，実際はそれ以前の場合もありうる．また，同様の理由で，変換規則 $vi^{(j,k)}_{[4,l]}$，あるいはこの頂点のラベルを変更する変換規則 $vr^{(j,k)}_{[4,l]}$ が含まれていないため，この頂点のラベル l（職位）はこの FTS からは特定できない．この FTS より，ユニーク ID が 1 の CEO がその他の 3 人を結びつけるハブの役割を果たしており，この CEO はユニーク ID が 3 の CEO からの情報をユニーク ID が 2 の Vice President や 4 に伝えている可能性があるといえる．

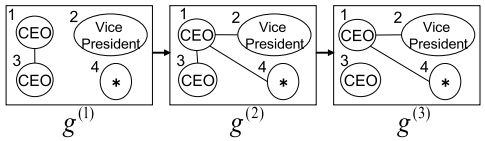

図 6: エンロンデータから列挙された FTS のグラフ表現

5. まとめ

本論文では，エンロンデータを用いて，GTRACE と Dynamic GREW2 の性能比較を行った．実験の結果，FTS の和グラフが大きくなると GTRACE の性能が悪くなる原因が明らかになった．系列長が長い FTS をマイニングするために，今後，FTS の探索方法を改善していく予定である．

参考文献

[1] K. Borgwardt, et. al. Pattern Mining in Frequent Dynamic Subgraphs. *Proc. of Int'l Conf. on Data Mining*, pp. 818–822, 2006.

[2] Enron Dataset, `http://www.cs.cmu.edu/~enron/`

[3] A. Inokuchi and T. Washio. A Fast Method to Mine Frequent Subsequences from Graph Sequence Data. *Proc. of Int'l Conf. on Data Mining*, pp. 303–312, 2008.

[4] A. Inokuchi, et. al. A Fast Algorithm for Mining Frequent Connected Subgraphs. *IBM Research Report*, RT0448, 2002.

[5] 元田浩. 明示的理解に魅せられて. 人工知能学会学会誌 pp.615-625,1999.

[6] J. Pei, et. al. PrefixSpan: Mining Sequential Patterns by Prefix-Projected Growth, *Proc. of Int'l Conf. on Data Eng.*, pp. 2–6, 2001.