特　集　「アクティブマイニング」

多段階学習方式によるデータ収集と前処理の自動化

Data Gathering and Automated Preprocessing by a Multiple-Stage Learning Method

沼尾　正行　大阪大学産業科学研究所
Masayuki Numao　The Institute for Scientific and Industrial Research, Osaka University.
numao@sanken.osaka-u.ac.jp, http://www.ai.sanken.osaka-u.ac.jp

ナッティー チョラウィト（同　上）
Cholwich Nattee　cholwich@sanken.osaka-u.ac.jp, http://www.ai.sanken.osaka-u.ac.jp

市瀬　龍太郎　国立情報学研究所
Ryutaro Ichise　National Institute of Informatics.
ichise@nii.ac.jp, http://research.nii.ac.jp/~ichise/

Keywords: data gathering, preprocessing, multiple-stage learning.

1. はじめに

本計画研究では，A01 アクティブ情報収集班の中で，機械学習やプランニング技術を用いた前処理について研究を行った．従来からあるデータ処理技術では，分散した数多くのデータベースの中からあらかじめ必要なデータを選んで収集する作業を行う必要があった．これらの作業は「前処理」と呼ばれ，膨大な人手と時間を要していた．分散したデータベースの中からあらかじめ必要なデータを選んで収集し，データマイニングに適した形に整形する作業が前処理である．その手順は複雑であり，勘と経験に頼った職人芸とされてきた．

本研究では，データベース間の通信ネットワークを探索して，必要な収集および処理パスを発見し，情報収集と前処理の手順を半自動的に生成する手法として，次を開発した．(1) 前処理支援システムの構築: データマイニングシステム MUSASHI の前処理オペレータを組み合わせることにより，自動的に前処理を行う．(2) 伝言ゲーム型の情報収集: 人間関係の重みを調整することにより，必要な情報を収集する．

さらに，そうした前処理によって，マイニングの結果を改善する次の試みを行った．(3) 属性の重み付けによるマイニング過程の制御：文献データベースに基づいて属性を選択し，重み付ける．(4) 同一値を取る期間に基づく前処理: 間隔不定な時系列データを内挿する前処理により，マイニング結果を改善する．(5) 実例の重み付けによるマイニング過程の制御：実例の分布に基づいて，実例を選択し，重み付ける．本稿では，これらのうち (1), (4), (5), (2) について，概説する．

2. 前処理プランニング

プランニングシステムは初期状態，ゴール状態，オペレータセットを与えることで自動的にプランと呼ばれる行動列を生成する．これを利用して，メタデータを介して，自動的に前処理を実行する方法を提案した[城 04]．すなわち，既存のデータセットのメタデータを「初期状態」に，目標データのメタデータを「ゴール状態」に，メタデータに対する操作を「オペレータセット」とすれば，図 1 のように，前処理のプランニングを行うことができる．

プランナとして GraphPlan をベースとし，型情報や限量演算子を利用できる IPP[Brenner]を，最終的に出力されるコマンドとしては MUSASHI[Mus 04]を利用した．したがって，入力としても MUSASHI のコマンドを記述することになる．

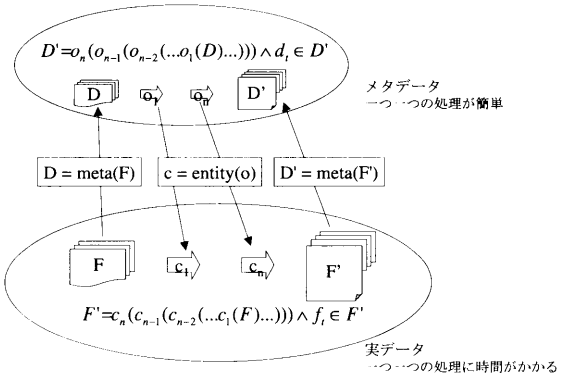


図 1　メタデータを介した前処理

2・1 前処理例 1

肝炎に関する医療データ（共通データ）を利用して，プランニングを試みた．ただし，実装の都合上日本語の属性名が利用できないため属性名を英語に変換した．まず，ベースファイル labo-base.xt に検査項目として GOT, GPT, TTT, ZTT の検査値を結合し，データのクリーニングを行う場合についての前処理コマンド列を生成する．

まずはクリーニング用オペレータセット $O_{cleaning}$ として表 1 を作成した．xtsed は属性値の文字列を置き換えるコマンドで属性値の末尾についている余計な文字を取り除く操作である．

そして，データ作成に必要なほかのオペレータのセットとして表 2 のような O_{step1} を作成した．

xtagg は-k で指定した属性をキーにして-f を指定した属性を集計し，xtdelnul は-F をつけることによって-f で指定したすべての属性が Null である行を削除する．これらのコマンドは必ず実行してほしいので ToDo を yes にし，xtagg は xtdelnul を実行した後に実行してほしいので（直後ではなくてもよい）Precondition に delnul1 を入れておく．

ここで，システムを使って表 3 のような条件でプランニングを行った．結果が図 2 である．

これを見ると，大部分は問題なく生成できており，このまま実行することもできる．しかし，実際には xtjoin の際に-n オプションが必要である．このオプションは参照ファイルにデータがなくてもベースファイルのレコードを削除しないというオプションなのだが，xtjoin はシステムによって自動的に生成されたため，システムがオプションまでは判断できないからである．したがって，-n オプションを手動で追加すればコマンド列は完成である．

表 1 クリーニング用オペレータセット

ID	コマンド
sed1	xtsed -f GPT:GPTclean -c 'H#$ |H$' -v"
sed2	xtsed -f GOT:GOTclean -c 'H#$ |H$' -v"
sed3	xtsed -f TTT:TTTclean -c 'H$' -v"
sed4	xtsed -f ZTT:ZTTclean -c 'H$ |L$' -v"

表 2 前処理例 1 用オペレータセット

ID	コマンド
delnul11	xtdelnul -f GOTclean, GPTclean, TTTclean, ZTTclean
agg1	xtagg -k MID, testdate, testnumber -f GOTclean, GPTclean, TTTclean, ZTTclean -c sum

表 3 前処理例 1 の入力

入力ファイル	labo-base.xt
関連ファイル	GOT.xt,GPT.xt,TTT.xt,ZTT.xt
出力ファイル	case1.xt
出力属性	MID,testdate,GOTclean,GPTclean TTTclean,ZTTclean
オペレータセット	$O_{user} = O_{cleaning} \cup O_{step1}$

```
xtjoin -k MID,testdate,testnumber -f TTT
                                  -m ./attribute/TTT.xt
                                  -i labo-base.xt |
xtjoin -k MID,testdate,testnumber -f ZTT
                                  -m ./attribute/ZTT.xt |
xtjoin -k MID,testdate,testnumber -f GPT
                                  -m ./attribute/GPT.xt |
xtjoin -k MID,testdate,testnumber -f GOT
                                  -m ./attribute/GOT.xt |
xtsed -f GOT:GOTclean -c 'H#$|H$' -v '' |
xtsed -f GPT:GPTclean -c 'H#$|H$' -v '' |
xtsed -f ZTT:ZTTclean -c 'H$|L$' -v '' |
xtsed -f TTT:TTTclean -c 'H$' -v '' |
xtdelnul -f GOTclean,GPTclean,ZTTclean,TTTclean -F |
xtagg -k MID,testdate,testnumber
      -f GOTclean,GPTclean,TTTclean,ZTTclean -c sum |
xtcut -r -f testnumber -o case1.xt
```

図 2 出力コマンド列

表 4 前処理例 2 の入力

入力ファイル	clossed.xt
関連ファイル	なし
出力ファイル	case2.xt
出力属性	MID,testdate,GOTclean,GPTclean TTTclean,ZTTclean
オペレータセット	$O_{user} = O_{cleaning} \cup O_{step1}$

```
xtsed -f GOT:GOTclean -c 'H#$|H$' -v '' -i clossed.xt|
xtsed -f GPT:GPTclean -c 'H#$|H$' -v '' |
xtsed -f ZTT:ZTTclean -c 'fH$|L$' -v '' |
xtsed -f TTT:TTTclean -c 'H$' -v '' |
xtcut -r -f UA |
xtcut -r -f I-BIL |
xtcut -r -f ALP |
xtdelnul -f GOTclean,GPTclean,ZTTclean,TTTclean -F |
xtagg -k MID,testdate,testnumber
      -f GOTclean,GPTclean,ZTTclean,TTTclean -c sum|
xtcut -r -f testnumber -o case2.xt
```

図 3 出力コマンド列

2・2 前処理例 2

次に入力データの構成の変化への対応を見るために入力データとして別の方法を使ってすでに必要と思われる属性の結合を行ったデータ clossed.xt が提供されたとする．ただし，clossed.xt には必要のない属性 ALP, I-BIL, UA が含まれている．これに対して表 4 のようにケース 1 と入力ファイルと参照ファイルだけが違う条件でプランニングを行う．その結果が図 3 である．

これを見ると実行する必要のなくなった xtjoin コマンドは組み込まれず，代わりに必要のない属性を削除するための xtcut-f コマンドが組み込まれており，自動的に入力データの構造の変化に対応していることがわかる．

2・3 前処理例 3

オペレータセットを再利用することを考える．実際に利用したクリーニング用のオペレータセット $O_{cleaning}$ やデータを離散化するためのオペレータセットといったものを作成するにはデータについての知識や医学的な知識が必要である．したがってこれらのオペレータセットを知識をもったユーザが作成し，ほかの人がこれを簡単に再利用できることは知識の再利用ができるということにほかならない．

表5 前処理例3の入力

入力ファイル	GPT.xt
関連ファイル	なし
出力ファイル	case3.xt
出力属性	MID,testdate,GPTcode
オペレータセット	$O_{user} = O_{cleaning} \cup O_{step1}$

```
xtsed -f GPT:GPTclean -c 'H#$|H$' -v '' -i GPT.xt |
xtagg -k MID,testdate,testnumber
      -f GPTclean -c sum |
xtcut -r -f testnumber -o case3.xt
```

図4 出力コマンド列

この再利用性について考えるために次のケースを考える．

GPTの変化をグラフ化することによって観察したいため属性としてMID, testdate, GPTcleanをもつデータを作成する．このデータを作成するためにはGPTのクリーニングおよび集計が必要である．この中で集計だけは今までに作成したことがない操作なのでO_{gptagg} ={xtagg-k MID, testdate, testnumber -f GPTclean -c sum, ToDo = yes}とする．これらを利用し入力を表5のようにして出力されたコマンド列が図4である．

これを見ると$O_{cleaning}$には必要のないオペレータも含まれているにもかかわらず，必要なものだけが選択されて適切な順番で適用されていることがわかる．このように，本システムでは，余計なオペレータが含まれていてもその中で適切なものを選んでプランを生成する．ただし，似たような操作を行うオペレータが複数含まれていた場合は，オペレータを誤ったプランが生成される場合がある．このようなときは，誤って現れたオペレータにIgnore情報を付加し，再プランニングを行えばよい．

このようなオペレータの再利用を効率的に行うには，機能ごとにオペレータセットを分けるなどの工夫が必要である．もし，履歴から自動的にオペレータセットの自動作成ができるような方法があれば，より効率的な再利用が可能になる．

本研究では前処理にプランニングシステムの技術を適用して自動化を進めるアイディアについて説明し，その実現方法について考察，実験を行った．その結果，現状ではまだ不十分な点もあるが半自動的なコマンド列の生成が可能であるとともに，入力データの構成の変化への対応や知識の再利用という面でも有効なシステムとなることを確認した．

3. 時系列データの前処理

3・1 アプローチ

時系列データはさまざまな属性の値が時点によって順序づけされた値が並ぶデータである．こうした時系列データからデータマイニングを行う手法は数多く存在する．(例:[Agrawal 93, Agrawal 95, Rafiei 97, Shatkay 96, Yazdani 96].) しかし，既存のデータマイニング手法のほとんどは，この時系列データの各レコードはほかのレコードと明確に分離されているものとして扱っている．すなわち，ある時点で発生した現象は，その時点でのみ起きたものとしている．

ここで提案する手法では，各現象が起きた時点ではなく，各現象が起きている期間に着目した手法を提案する．実際のレコードは不定の間隔で並んでいるため，この間隔からある属性値の持続期間を推定する．IDが同じレコードすべてについてこの操作をすることで，時系列データを属性値が同一の値を取る期間を基本構造としてもつデータに変換することができる．

その後，同一の値を取る期間をそれぞれ比べることにより，複数のIDのデータが共通の振舞いをする期間を見つけることができる．インターフェロン（以下IFN）投与以前の血液検査データは時系列データであるため，最終的にはこの期間を基本単位としてデータマイニングを行い，IFN療法の有効・無効性を予測する規則の作成を目指した[本山 05]．

3・2 前処理の方法

本研究では，データが共通の振舞いをする期間を見つけるために，同一の値を取る期間に着目した前処理を提案した．前処理の具体的な手順は表6のとおりである．

§1 属性値の持続期間の推定

あるIDをもつ時系列データは図5のような形状をしている．ここで時間軸上の値が負の値をしているのは，ある基準時より前のデータだけを扱っているためである．

図中に記された四つの黒丸はある属性値の実際のレコードを表しており，間隔は不定である．これらのレコード

表6 時系列データの前処理手順

入力：表形式の時系列データ
出力：データが共通の振舞いをする期間を基本構造とするデータ
処理手順：
(1) 同じIDをもつレコードを比較し，その間隔から各レコードのもつ属性値の持続期間を推定する．
(2) 異なるIDのデータを比較し，(1)で求めた持続期間が重複する期間を，該当する複数のデータが共通の振舞いを示す期間として求める．

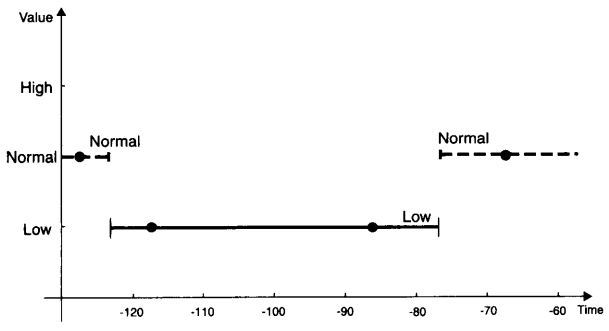


図5 属性値の持続期間を推定したデータ

表7 規則発見アルゴリズム

(1) もし，正例の集合が空ならば，規則集合を返す
(2) 属性を組み合わせることによって，正例を満たす規則を生成
(3) 得られた規則によって説明できる正例を正例集合から取り除く

を比較することによって，各レコードのもつ属性値の持続期間を推定することができる．

推定は次の二つの規則で行った．

- 隣接するレコードが異なる属性値のとき：両レコードの中央の時点までをその属性値とする．
- 隣接するレコードが同じ属性値のとき：両レコードの範囲をその属性値とする．

この推定を行うと，図中の実線および点線のように属性値の持続期間が推定される．すべてのデータについてこの推定処理を行う．

§2 データが共通の振舞いを示す期間

次に，属性値の持続期間が異なるIDの間で重複する期間を求める．データの特徴を捉えるには複数のIDのデータが共通の振舞いを示す期間を求めることが重要となる．

§3 知識発見アルゴリズム

上述の前処理によって変換された属性を用いて，表7のアルゴリズムによって，ルールを求める．ルールは分類する条件の連言で表すようにした．

3・3 実　　験

§1 データの前処理

各実験データに共通する前処理について記述する．まず血液検査データより，IFN投与1年前までの検査データを抽出した．次に，検査項目を列とし，患者番号(MID)および検査日をキーとして，各検査項目が個別の属性となるような表形式データに変換した．

検査項目については，特に重要であると思われるGOT, GPT, TTT, ZTT, D-BIL, I-BIL, T-BIL, ALB, CHE, TP, T-CHOを使用し，検査値を医師の作成した離散化指標をもとに4～7段階に離散化した．

このように前処理した検査データの中で，IFN投与後にウイルスが消滅した55例を正例，IFN投与後にウイルスが残っていた55例を負例とした．

§2 実験1：提案した前処理によるルールの最小単位の変化

まず，提案した前処理による効果を調べるために，ルールの最小単位の変化を調べた．もし，提案した前処理を行わない場合に出現する個々のルールの最小単位は次のようになる．

- 投与A日前に属性Bの値がCである．
- 投与D週間前に属性Eの値がFである（1週間単位でデータをまとめた場合）．
- 投与G月前に属性Hの値がIである（1月単位でデータをまとめた場合）．

一方，提案した前処理を行った場合には出現する個々のルールの最小単位は次のようになる．

- 投与A日前から投与B日前の期間常に属性Cの値がDである．

データの数が少ない場合やデータの間隔が一定でない場合には，支持度向上のために，ある一定の期間ごとにデータをまとめることが多い．ここでは，10-foldの交差検定法により前処理をした際の最小単位の個数の変化を調べ，潜在的に支持度がどの程度，向上する可能性があるかについて調べた．その結果，二週間単位，月単位でまとめる場合よりも，提案した前処理はより適切な形で時系列データの特徴を捉えることができた．

§3 実験2：治療が有効・無効を弁別するルールの作成

次に，IFN治療が有効・無効を弁別するルールを作成する実験を行った．まず，IFN投与患者の血液検査データを提案した前処理によって構造を変換した．次にこの変換したデータに対し提案したアルゴリズムによって，IFN治療が有効であるか無効であるかのルールを作成する実験を行った．ここで，有効とはHCV-RNA（肝炎ウイルスの有無の判定法）によりIFN投与後にウイルス消滅を確認したものとし，無効はIFN投与後にウイルス存在を確認したものとする．

10-foldの交差検定を行った際の正答率は表8のとおりである．ここで，正答とは，正例が得られた規則に当てはまる場合，あるいは負例が得られた規則に当てはまらない場合とした．各手法の違いは，提案した前処理をしたか，それとも一定の日数でデータをまとめたか，という点のみであり，知識発見手法は同一の条件で行った．

何れの場合も提案した前処理を行ったときの正答率が高くなっていることがわかる．これは提案した前処理を行うことで探索空間を広げることができるためだと考えられる．また，同じ理由でより支持度の高いルールが見つけられるのが確認できた．

次に，実際に得られたルールセット（表9）を紹介する．これらのルールの確信度はすべて100％である．このルールセットの正答率は81.8％，正例のカバー率は71.4％であった．

規則の意味は，ルール番号1番を例にとると，「属性GPTがIFN投与209日前から投与207日前までの期間中，常にUltra HIGHという離散値である患者はIFN投与が有効である」という意味である．また，ルール番号6，7，8のルールについては全条件が満たされている必要がある．

表8 正答率（平均）

まとめた日数	正答率[%]
1	39.09
14	47.27
30	68.78
提案した前処理	73.03

表9 実験で得られたルールセット

	属性	値	持続期間	支持度[%]
1	GPT	Ultra HIGH	-209 ～-207	21.8
2	GPT	Ultra HIGH	-194 ～-189	23.6
3	TP	HIGH	-139 ～-138	21.8
4	TP	HIGH	-132 ～-128	20.0
5	ZTT	Very HIGH	-171 ～-168	18.2
6	GPT	Ultra HIGH	-60 ～-48	30.9
	I-BIL	Normal	-58 ～-21	
	ALB	Normal	-34 ～-30	
7	GPT	Ultra HIGH	-35 ～-31	29.1
	D-BIL	Normal	-113 ～-59	
	I-BIL	Normal	-57 ～-21	
8	GPT	Ultra HIGH	-134 ～-133	27.3
	I-BIL	Normal	-44 ～-21	
	CHE	Normal	-135 ～-132	

肝炎の患者データに対して，IFN 治療に効果がある患者に共通する知識を発見する課題に取り組み，時系列データが同一の値を取る期間に注目し，そこからデータが共通の振舞いを示す期間を求めるという手法を提案した．この処理を施したデータを使用して知識発見を行う実験をした結果，肝炎データから，より高い支持度の規則が発見でき，正答率も向上することが示された．

4. 構造データの前処理

データマイニングにおいて処理速度を向上させるため，前処理において例題のサンプリングを行うことがあるが，その場合でも，選択された例題はすべて同じ重みで処理されるのが普通である．このような前処理を一歩発展させて，重要な例題に2個分あるいは3個分の重みを付ける手法について研究した[Nattee 04]．

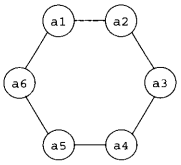
化学構造式の特徴を学習するには，あらかじめ特徴的な部分パターンを列挙して，各部分パターンを属性として，決定木やニューラルネットワークなどを用いることが多い．しかし，部分パターンは無数にあり，その中から有用なパターンを見つけるためには，職人的な勘に基づく前処理が必要であった．知識表現の基本に立ち帰れば，図6に示すような述語表現を用いて表現すれば，こうした悩みのほとんどは解消される．しかしながら，述語論理の学習法である帰納論理プログラミングシステムを駆使するには，その複雑なメカニズムに通暁する必要がある．

最もシンプルで見通しの良い帰納論理プログラミングシステムとして，FOIL[Quinlan 90]がある．このシステムは，決定木とよく似た次のような Gain ヒューリスティック関数で制御され，決定木の拡張として理解が容易である．

$$I(T_i) = -\log_2 \frac{|T_i^+|}{|T_i^+| + |T_i^-|}$$

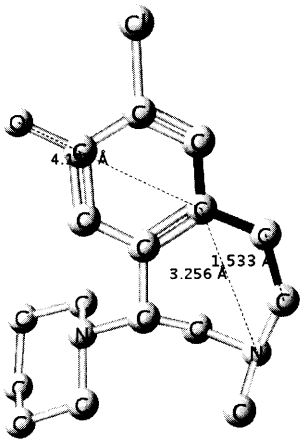
$$Gain(L_i) = |T_i^{++}| \times (I(T_i) - I(T_{i+1}))$$

FOIL は，トップダウン方式をとっているために，近視眼的な探索を行いやすいという欠点があった．ところが，




```
atom(c1,a1,c)     atom(c1,a2,c)
atom(c1,a3,c)     atom(c1,a4,c)
atom(c1,a5,c)     atom(c1,a6,c)
bond(c1,a1,a2,1) bond(c1,a2,a3,2)
bond(c1,a3,a4,1) bond(c1,a4,a5,2)
bond(c1,a5,a6,1) bond(c1,a6,a1,2)
```

図6 構造データの述語表現




```
d1(A) :-
 atom(A,B,C,D,E,F), E>=3.7, F=3.3,
 bond(A,L,B,H,M,N), bond(A,G,H,I,J,K),
 K=1.5, bond(A,O,B,P,Q,R), not_equal(H,P).
```

図7 発見された規則の例

複数インスタンス学習（multiple instance learning）で用いられている評価関数 *DD*（Diverse Density）に基づいて，次のように，例に重み付けを行うことにより，ボトムアップ手法よりも良い結果が得られることが示せた．

$$DD_s(T) = \sum_{T_i \in T} DD(T_i)$$

$$I(T_i) = -\log_2 \frac{DD_s(T_i^+)}{DD_s(T_i^+) + |T_i^-|}$$

$$Gain(L_i) = DD_s(T_i^{++}) \times (I(T_i) - I(T_{i+1}))$$

$DD(T_i) = 1$ に固定すると重み付けを行わない場合と等価になることに注意されたい．

以上の結果は，静的な前処理によって，マイニングの過程を大幅に改良できることを示しており，前処理を工夫することで，これまで顧みられなかった単純で見通しの良いマイニング手法が採用可能になると考えられる．

以上の手法を元に計画研究 A03-10（岡田）と協力して，化学薬品データについて，実験を行ったところ，専門家の納得する結果が短期間で得られた．学習されたルールの例と対応する化学構造を図7に示す．

5. 口コミによる伝言ゲーム型の情報収集

前処理の前半である情報収集作業に関して，伝言ゲーム型のシステムを構築した．このシステムは情報収集過程を半自動化するための推薦機構をもっている．

計算機ネットワーク上にはさまざまな情報が氾濫し，ユーザにとって有用な情報を獲得するのに大変な労力を必要とする．情報を発見するための最も身近なツールとして，Yahoo や Goo，Google[Brin 98]といった検索エンジンがあるが，最もカバー率の大きいものでも，せいぜい 4 ～ 6 割程度であるといわれており，広大な WWW 空間に氾濫した情報をすべて把握するのは，非常に困難である．

そこで，情報の内容を解析し，ユーザが文章中のどの部分に対して興味をもっているかを推定することで，同様の部分をもつ文書を有用な情報として提供する content-based filtering や，評価傾向の似ている他ユーザのもつ情報を参考にして，有用な情報を提供する collaborative-filtering などの情報フィルタリングの研究が盛んに行われている．

これらは結局，人をモデリングしていることになっており，かなり困難な問題を扱っている．人の内部を推定してモデル化するには，アンケートを取ったり，プロトコル解析するなどの観察に頼るか，脳波，MRI，光トポグラフィーなどを用いることになる．何れにせよ，リアルタイムでユーザをモデル化するには障害が多い．たとえ内部が完全にわかっても，社会レベルの活動への影響を解析するのは困難である．人にはいろいろな側面があり，ごく一部の側面のみしかモデル化できないからである．

これに対して，大量の意見を要約して質の高い情報を提供するマスメディアと情報発信の自由やインタラクティブ性をもつ電子メディアを融合した新しいメディア[西田 99]や，コミュニティの可視化[藤田 00, 高橋 99, 舘村 99]など，人の心の動きや人間関係を考慮し，コミュニティの形成を支援する研究もある．しかし，情報フィルタリングのように，有用な情報獲得支援といったことまで考慮されていない．

本研究では，人間関係を電子コミュニティ上に再現することで，効率の良い情報収集や円滑なコミュニケーションを支援するシステムを提案した．そのうえに推薦機構を導入することにより，人を独立した個体としてモデル化するのではなく，現実に情報のやり取りが行われている人間関係をモデル化する．人の間のコミュニケーションには，ある程度の持続性があり，一貫したモードが存在する．それを解析したほうが，個体をモデル化するよりも容易だと考えるからである．

5・1 WAVE

本研究では，電子コミュニティ上において効率の良い情報収集や円滑なコミュニケーション支援をするシステムとして，WAVE（Word-of-mouth-Assisting Virtual Environment）*1 を提案した[Numao 02]．WAVE は，口コミを電子コミュニティ上に再現することで，人間のコミュニティそのものが，分散化された情報収集システムとして機能し，グローバルな情報交換ネットワークを形成する．また，WAVE 上でユーザが情報の発信，公開，閲覧，評価，取得をシームレスに行うことができるように，ユーザインタフェースにも工夫を行った（図 8）[伊藤 03]．これにより，従来よりも効率の良い情報収集や円滑なコミュニケーションを行うことができた．

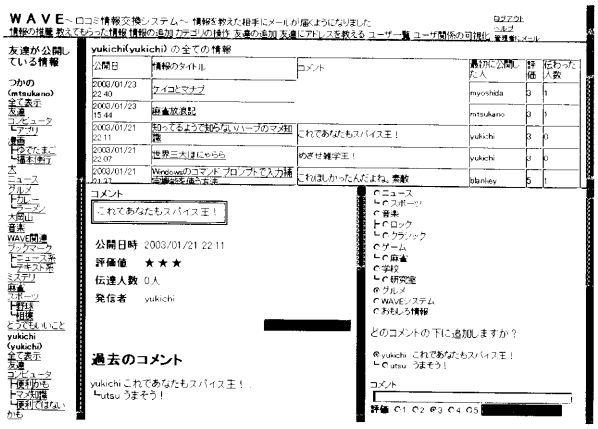


図 8　WAVE システムの画面

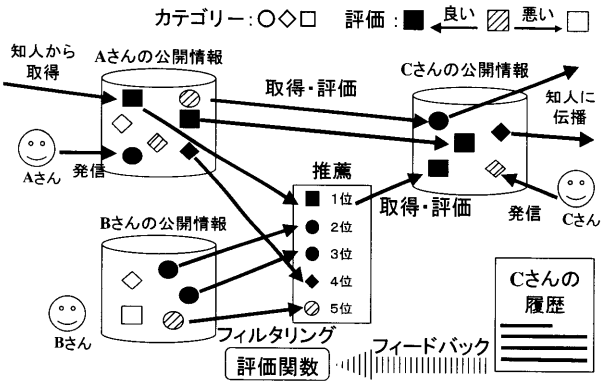


図 9　WAVE における口コミの再現

以下では，WAVE の仕組み（図 9）とその特徴について説明する．

§1 情報の発信

各ユーザは，自分がもっている新しい情報を発信することができる．発信する情報には Web ページや画像データの URL 情報を付加することができたり，情報を閲覧するユーザが情報の内容を判断しやすいように，情報を簡単に分類するためのカテゴリーを割り当てるようになっている．また，情報に対して 1 ～ 5（1 が一番悪く，5 が一

*1 WAVE には，口コミが波のように伝播していくというイメージと，このシステムが，WAVE を起こし，世界中の人々に使ってもらえるようなコミュニケーションメディアとなってほしいとの願いがこめられている．

番良い）の評価値を与える．発信した情報は，自分の情報として他ユーザに公開される．つまり，WWWやメーリングリストのように，自分なりの情報を多数の人々に向けて自由に発信することが可能である．

§2 情報の公開

自分が発信した情報や他ユーザから取得した情報は，自分の情報として公開される．公開されている自分の情報を閲覧することができ，情報の評価やコメントを修正したり，必要のなくなった情報を削除することもできる．また，ユーザのもつWAVE専用のアドレス（“ユーザ名@ホスト名:ポート番号”の形）をシステムに登録すれば，そのユーザの情報を閲覧することができる．

このとき，ユーザは情報提供者として信頼できるユーザを登録する．これによりWWWのような情報公開性をもちながら，電子メールにおける1対1のコミュニケーションのようにユーザどうしの人間関係が強く現れ，相手の専門分野や信頼性を判断することができるので，WWWなどに欠落していた情報源の信頼性を高めることできる．

§3 情報の評価と取得

他ユーザの情報を閲覧する際に興味をもった情報があれば，その情報に関する詳しいデータを見ることができる．このときユーザは，その情報に与えられた評価に対して新しい評価をつけたり，新たな付加情報として，コメントを与えることができる．新しい評価やコメントを与えると，その情報は自動的に取得され自分の情報として公開される．そして，さらに別のユーザによって，その情報は評価・取得されるという繰返しになる．

つまり，WAVEでは情報の公開，閲覧，評価，取得をシームレスに行うことができる．情報をアップロードしたり，情報の存在を人々に知らせたりしなければならないといった，従来WWWがもっていた情報公開にかかる手間を軽減することができる．また，BBSやメーリングリストにおけるROM（Read Only Member）がもつような，積極的な参加への心理的抵抗感を軽減し，情報伝搬において重要な役割を果たすブリッジを維持することができる．

そして，情報の評価と取得が繰り返し行われることにより，人間を介しながら，良い情報だけが生き残り，ネットワーク上に広まる．同じような嗜好をもつ人間は集まりやすいということから，情報の流れは指向性をもち，ブリッジを介して，情報がひとたび自分の属するクリークに到達すれば，自分もその情報を得られる．つまり，WAVEは，口コミを電子コミュニティ上に再現することで，人間のコミュニティそのものが，分散化された情報収集システムとして機能し，グローバルな情報交換ネットワークを形成する．

カテゴリーを情報の発信時に付与し，取得時に変更することができる．さらに，ユーザごとに階層構造をもった独自の分類体系を設定できる．これにより，各ユーザ独自の分類が可能である．各ユーザごとに得意分野があるので，他ユーザにとっては，そのユーザの得意なカテゴリーのみを参照するのが便利である．

5・2 情報の推薦

他ユーザの公開している情報を閲覧する際に，情報参照ユーザの数が増加したり，ひとりのユーザが公開する情報の数が増加したりすることによって，すべての情報を閲覧するのは，ユーザにとって負担となってくることが予想される．

そこで，補助的な機能として，ユーザの閲覧履歴などをもとに，評価関数を動的に作成し，公開されている情報の中から有用であると思われる情報の一覧を表示する．これにより，ユーザ間での情報のやり取りが支援され，システム上で，より活発な情報交換を行うことができる．なお，推薦の評価関数は，以下の二つの項目について考慮した．

- ●情報提供者が与えた情報に対する評価
- ●ユーザの嗜好に基づく情報に対する評価

一般に，口コミにおいて，人は他人から聞いた評価を参考にする．WAVEでは，ユーザは，情報に対して1～5（1が一番悪く，5が一番良い）の評価値を与えており，それが参考になるが，ユーザはそれをそのまま評価とするわけではなく，情報提供者に対する信頼性や専門性も考慮する．そこで，ある情報の提供者の公開情報をどれくらい閲覧・取得したかや，その情報提供者に対してどういったカテゴリーの情報を閲覧・取得したか，回数を記録しておき，ユーザがほかの情報提供者と比べてその情報提供者にどれくらい依存しているかクリック率を計算し，評価関数に用いる．また，このとき，ユーザの嗜好や他ユーザとの人間関係，信頼性は，時間が経るにつれて変化していくので，その情報提供者の公開情報を最後に参照してから経過した日数から，最近その情報提供者にどれだけ依存しているかも調べる．

人は，他人から聞いた評価を参考にするだけでなく，自分自身の嗜好も合わせたうえで，その情報が有用であるかを判断する．そこで，ユーザの嗜好によって，その情報に対して，1～5の評価値でどれくらいの評価を与えるかを予測して，評価関数として計算する．ほかのカテゴリーと比べてどの程度興味があるか，この情報と同じカテゴリーの情報に対して与えてきた評価の平均値を計算したり，今までその情報と同じカテゴリの情報をどれくらい閲覧・取得しているか，回数を記録しておき，クリック率も計算する．

一般に多くの人々の間を伝搬してきた情報ほどユーザは好むので，情報の伝達人数についても考慮し，伝達人数が多いほど評価を高くする．さらに，ユーザは新しい情報を好むので，情報が公開されてからの日数が経ったものほど情報の評価は下がるようにする．

情報発信時だけではなく，情報取得の際にもカテゴリーを付与できるようになっているので，それを考慮するようにすれば，推薦精度が向上する．カテゴリーとして階層

構造を許すようにすると，階層構造内で何親等になるかという距離を定義することができ，カテゴリー間の類似度が計算できる．それに基づいてほかのカテゴリー中の情報への評価を活用すれば，カテゴリーに分類したことによるサンプル数減少を補える[沼尾 99]．

6. ま と め

計画研究 A01-04 の成果の概略を述べた．以上の一連の手法により，前処理作業のかなりの部分を省くとともに，マイニング結果の質の向上に貢献することができた．したがって，本計画研究は当初の目標を満たすとともに，期待以上の成果を上げることができたと考えている．

謝 辞

本稿は，数多くのスタッフおよび学生の共同作業によるものである．本計画研究の参加者およびアクティブマイニングプロジェクトの皆様に感謝する．

◇ 参 考 文 献 ◇

[Agrawal 93] Agrawal, R., Faloutsos, C. and Swami, A.: Efficiency Similarity Search in Sequence Databases, *Proc. Conference of Foundations of Data Organization 22* (1993)

[Agrawal 95] Agrawal, R., Psaila, G., Wimmers, E. L. and Zait, M.: Querying Shapes of Histories, *Proc. VLDB* (1995)

[Brenner] Brenner, M., Koehler, J. and Hoffmann, J.: IPP: http://www.informatik.uni-freiburg.de/~koehler/ipp.html

[Brin 98] Brin, S. and Page, L.: Anatomy of Large-Scale Hypertextual Web Search Engine, *Proc. 7th Int. World Wide Web Conference* (1998)

[藤田 00] 藤田邦彦, 亀井剛次, Jettmar, E., 吉田 仙, 桑原和宏：ネットワークコミュニティの可能性— Community Organizer 評価実験結果報告—, 第 3 回 CMCC 研究会シンポジウム (2000)

[伊藤 03] 伊藤雄介, 沼尾正行, 右田隆仁：口コミ支援システム WAVE へのプッシュ型情報交換の導入, 情報処理学会知能と複雑系研究会資料, No. 2003-ICS-132, pp. 81-86 (2003)

[城 04] 城 敦, 沼尾正行：データマイニングのための前処理プラニング, 人工知能学会知識ベースシステム研究会資料, No. 64, pp. 31-36 (2004)

[本山 05] 本山真也, 市瀬龍太郎, 沼尾正行：間隔不定な時系列データからの知識発見, 人工知能学会知識ベースシステム研究会資料, No. 69 (2005)

[Mus 04] MUSASHI: Mining Utilities and System Architecture for Scalable processing of HIstorical data (2004) http://musashi.sourceforge.jp/

[Nattee 04] Nattee, C., Sinthupinyo, S., Numao, M. and Okada, T.: Learning First-order Rules from Data with Multiple Parts: Applications on Mining Chemical Compound Data, *Proc. 21st Int. Conference on Machine Learning*, pp. 606-614 (2004)

[西田 99] 西田豊明, 畦地真太郎, 藤原伸彦, 角 薫, 福原知宏, 矢野博之, 平田高志, 久保田秀和：パブリック・オピニオン・チャンネル, 第 2 回 CMCC 研究会シンポジウム (1999)

[沼尾 99] 沼尾正行, 横山 甲：階層化された知識の継承による情報フィルタリング, 情報処理学会知能と複雑系研究会, Vol. 99-ICS-116, pp. 43-48 (1999)

[Numao 02] Numao, M., Yoshida, M. and Ito, Y.: Data Mining on the WAVEs — Word-of-mouth-Assisting Virtual Environments, *Frontiers in Artificial Intelligence and Applications*, Vol. 79, pp. 11-20 (2002)

[Quinlan 90] Quinlan, J. R.: Learning Logical De nitions from Relations, *Machine Learning*, Vol. 5, pp. 239-266 (1990)

[Rafiei 97] Rafiei, D. and Mendelzon, A.: Similarity-Based Queries for Time Series Data, *SIGMOD Record* (1997)

[Shatkay 96] Shatkay, H. and Zdonik, S. B.: Approximate Queries and Representations for Large Data Sequences, *Proc. 12th Int.l Conference on Data Engineering* (1996)

[高橋 99] 高橋正道, 北山 聡, 金子郁容：ネットワーク・コミュニティにおける組織アウェアネスの計量と可視化, 情報処理学会論文誌, Vol. 40, No. 11, pp. 3988-3999 (1999)

[舘村 99] 舘村純一：協調型情報探索を支援する仮想評者とその視覚化 (1999)

[Yazdani 96] Yazdani, N. and M. Ozsoyoglu, Z.: Sequence Matching of Images, *Proc. 8th Int. Conference on Scientific and Statistical Database Management* (1996)

2005 年 1 月 28 日 受理

著 者 紹 介

沼尾 正行 (正会員)

1982 年東京工業大学工学部電気電子工学科卒業．1987 年同大学院情報工学専攻博士課程修了．工学博士．同年より東京工業大学工学部情報工学科，1994 年より同大学院情報理工学研究科計算工学専攻勤務．2003 年大阪大学産業科学研究所教授．現在に至る．1989 ～ 90 年スタンフォード大学 CSLI 客員研究員．人工知能，機械学習，関数型言語などの研究に従事．情報処理学会，日本認知科学会，日本ソフトウェア科学会，AAAI 各会員．

ナッティー チョラウィト (正会員)

1998 年チュラロンコーン大学工学部卒業．2001 年東京工業大学大学院情報理工学研究科計算工学専攻修士課程修了．2004 年同大学院博士課程修了．博士 (工学)．2004 年より大阪大学産業科学研究所助手．現在に至る．

市瀬 龍太郎 (正会員)

2000 年東京工業大学大学院情報理工学研究科計算工学専攻博士課程修了．博士 (工学)．同年より国立情報学研究所知能システム研究系助手．2001 年から 2002 年までスタンフォード大学言語情報研究所客員研究員．機械学習，知識発見，知識共有などの研究に従事．AAAI，電子情報通信学会，情報処理学会，日本認知科学会各会員．