Vol.2011-MPS-86 No.9
Vol.2011-BIO-27 No.9
2011/12/1

# ユーザフィードバックを用いた重み付き自己組織化マップ

久田大地[†1]　吉川毅[†1]　野中秀俊[†1]

ユーザにとって望ましいクラスタリング結果の定義は，ユーザそれぞれによって異なるものである．そのため，入力データの特徴のみを用いてクラスタリングを行っても，望ましいクラスタリング結果が得られない場合が多くある．そこで，本研究では教師無しクラスタリング手法である自己組織化マップに重みベクトルとユーザフィードバックを導入し，クラスタリングにユーザの意図を反映させる手法を提案する．

# Weighted Self-Organizing Map with User Feedback

Daichi Hisada,[†1] Takesi Yosikawa [†1]
and Hidetosi Nonaka[†1]

A clustering result which a user wants to obtain differs from user to user. A desirable clustering result will not be obtained only with the data features in clustering. We propose a new clustering algorithm which can reflect user's intention by introducing weight vectors and user feedback into Self-Organizing Maps. By using this algorithm the user can refine and adjust the clustering system interactively.

## 1. はじめに

### 1.1 背　景

互いに特徴が似ているデータの集まりをクラスタとよび，データを適切なクラスタに分けることをクラスタリングという．クラスタリングを行うシステムは，あらかじめ与えられた何らかの基準に基づいてクラスタリングを行い，この基準の下で最善のクラスタリング結果を出力する[1)]．そのため，どのような基準に基づいてクラスタリングするかによって，様々なクラスタリングを考案できることが指摘されている[2)]．

例えば，ある図形を入力データとしてクラスタリングすることを考える（**図 1**）．図形の辺の数と頂点数を基準としてクラスタリングを行うとクラスタリング A のようになり，図形の色を基準としてクラスタリングを行うとクラスタリング B のようになる．更に，特徴としては説明が出来ないが実験者の直感に基づいたクラスタリングとして，クラスタリング C のようなクラスタリングも考えられる．

上に述べた例は，基準の選び方によって，クラスタリング結果に差異が生じることを示しているが，他にも入力データ間の距離定義を変更したり，クラスタリングの手順を変える事により，様々なクラスタリング結果が得られる可能性がある．

入力データに対して，正しいクラスタリングが一意に定まらない原因として，クラスタの厳密な定義が存在しないことが指摘されている[1)2)]．そのためクラスタリング手法の評価実験などでは，正しいクラスタリング結果を実験者があらかじめ作成したり，あるいは既にクラスに分類されているデータを使用し，実験で得られたクラスタリング結果と比較することによって，手法の性能を評価することが一般的となっている．

このように，正しいクラスタリング結果やその定義がユーザの主観に依存するという問題点に対し，筆者らは，システムがユーザからクラスタリングに対する何かしらの情報を受け取り，ユーザの意図をクラスタリングに反映させることにより，望ましいクラスタリング結果が得られる可能性があることに着目した．

本研究では，自己組織化マップに Subset Clustering の手法を導入し，それぞれのクラスタに対する特徴の関連度を重みベクトルで捉えた Weighted SOM を提案する．更に，Weighted SOM が出力したクラスタリング結果に対してユーザフィードバックを与えることにより，特徴の関連度を変化させ，実験者の意図を考慮したクラスタリングを行う手法を提案する．

### 1.2 目　的

本研究ではユーザフィードバックを用いた重み付き自己組織化マップである Weighted SOM を提案する．Weighted SOM では，各クラスタに重みベクトルを付加することにより，それぞれのクラスタに対して支配的な特徴や各特徴の関連度を定量的に得ることが可能となる．また，Weighted SOM が出力したクラスタリング結果に対して，ユーザフィードバックを与え，重みベクトルに反映させることにより，ユーザの意図を考慮したクラスタリングを行うことが可能となる．

†1 北海道大学大学院情報科学研究科
Graduate School of Information Science and Technology, Hokkaido Unversity

Vol.2011-MPS-86 No.9
Vol.2011-BIO-27 No.9
2011/12/1

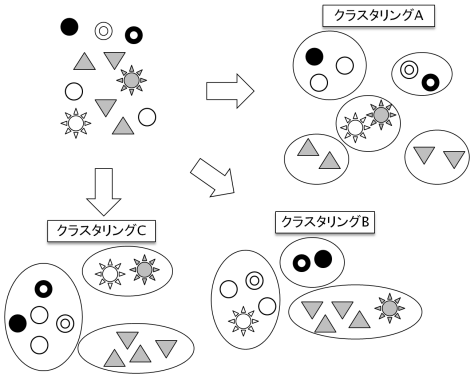


**図 1** クラスタリング結果の例
Fig. 1 Examples of clustering result

本研究では Weighted SOM によりクラスタリングされた結果に対し，ある入力データ$\boldsymbol{x_f}$が配置される望ましいクラスタ $k'_f$ を，他の入力データのクラスタリングを考慮してユーザが指定することをユーザフィードバックとする．

このユーザフィードバックによりシステムは，クラスタ $k'_f$ の周囲にクラスタリングされている入力データのうち，$\boldsymbol{x_f}$と似ている入力データを $k'_f$ に集めることで，入力データ$\boldsymbol{x_f}$がクラスタ $k'_f$ に配置されることが望ましいというユーザの意図をクラスタリングに反映させる．

自己組織化マップ (SOM:Self-Organizing Maps) は，Teuvo Kohonen が提案したニューラルネットワークを用いた教師無しクラスタリング手法である[3]．SOM をクラスタリングに用いると，高次元の入力データを 2 次元のマップに射影することができる．本研究で SOM を用いる理由は，クラスタリング結果を 2 次元のマップ上で表現でき，それぞれのクラスタがどのような関係を示しているかを視覚的に観察できるからである．また SOM では，2 次元のマップ上で入力データ間の距離関係が保存されているため，ユーザは他のクラスタを参考に望ましい入力データのクラスタを予想することができる．更に，SOM は教師無しクラスタリング手法であるため，教師データの一種であるユーザフィードバックがない場合でも，クラスタリングを行うことができる．

## 2. 関 連 研 究

### 2.1 ユーザフィードバックを用いるクラスタリング手法

Andreas Nurnberger ら[4] は，教師無しクラスタリング手法である SOM に，ユーザフィードバックを学習する重みベクトルを適用した手法を提案した．この研究では，ある参照ノードに属すると判断された入力データを，ユーザが別の参照ノードに移動させるというユーザフィードバックを SOM に導入している．このユーザフィードバックを用いて重みベクトルを更新し，入力データをクラスタリングしなおすことでユーザの意図に考慮したクラスタリングを行っている．この手法は本研究の提案手法とは異なり，ユーザフィードバックを受けて初めて重みベクトルが更新される．

山田ら[5] は，人間と知的システムが協調しながら問題解決を行う知的インタラクティブシステム実現のために，最小ユーザフィードバック (MUF: Minimal User Feedback) を提唱している．この研究では，知的システム全体のパフォーマンスを維持しつつ，ユーザが与えるフィードバックの計算論的コストと認知的コストを最小にすることを目的としている．計算論的コストを最小化するために制約つきクラスタリングを拡張した手法[6] を，認知的コストを最小化するために人の能動的学習を促進する GUI[7] を提案している．この両者では，主に制約付き $k$-means 法を改良する事を考えている．

### 2.2 重みベクトルを用いるクラスタリング手法

Hao Cheng ら[8] は，$k$-means 法に重みベクトルを適用した Locally Weighted Clustering(LWC) と，LWC に must-link と cannot-link といった制約情報を付加した，Constrained Locally Weighted Clustering(CLWC) を提案している．LWC と CLWC は LAC とは異なり，重みの影響度を表すパラメータが存在しないためパラメータの調整が必要ない．LWC と CLWC はいくつかのデータに対して LAC などの既存クラスタリング手法よりも，高いクラスタリング精度を示している．しかし，LWC と CLWC はどちらも $k$-means 法を拡張した手法であるため，正しいクラスタ数がわからない場合は，正しい分類が出来ない．

## 3. 背 景 知 識

### 3.1 Subspace Clustering

一般的なクラスタリング手法は入力データの全ての次元を用いてクラスタリングを行っている．しかし，高次元の入力データにはクラスタリングに不必要な次元が多く，その全てを用いてクラスタリングを行うとクラスタリング精度が落ちる場合がある．また，入力デー

Vol.2011-MPS-86 No.9
Vol.2011-BIO-27 No.9
2011/12/1

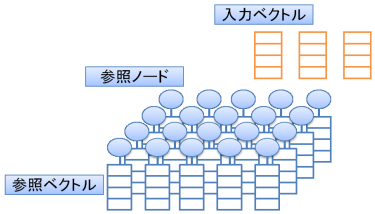


**図 2** SOM の概念図
Fig. 2 Conceptual diagram of Self-Organizing Map

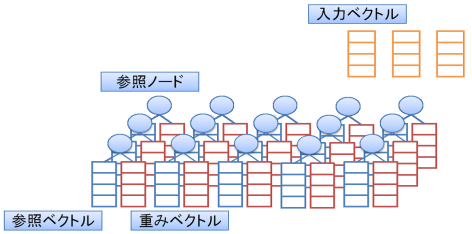


**図 3** Weighted SOM の概念図
Fig. 3 Conceptual diagram of Weighted Self-Organizing Map

タが高次元になると入力データ間の距離が等距離になってしまい，データ間の類似性を正しく求めることが出来なくなってしまうという指摘がある[9]．そこで Subspace Clustering では，各クラスタに重みベクトルを付加して，クラスタ毎に次元の重みを変化させ，クラスタに関連性のある次元を用いてクラスタリングを行うことでクラスタリング精度の向上を図っている．

### 3.2 自己組織化マップ

SOM は Teuvo Kohonen により提案された教師無しクラスタリング手法である．SOM は多次元の入力データを 2 次元平面に配置された参照ノードに分類する (**図 2**)．このとき，ある入力データに対して特徴が似ている他の入力データは 2 次元平面上で近くの参照ノードに分類され，特徴が似ていない入力データは遠くの参照ノードに分類される．

SOM には Online 型 SOM と Batch 型 SOM がある．提案手法では，学習効率が良いとされている Batch 型 SOM を用いる．

## 4. 提 案 手 法

### 4.1 概　　要

提案手法である Weighted SOM では，Teuvo Kohonen が提案した Batch 型 SOM[3] の各参照ノードに重みベクトル$\boldsymbol{w_k}$を適用し，参照ベクトル$\boldsymbol{m_k}$と入力データ$\boldsymbol{x_i}$を用いて重みベクトル$\boldsymbol{w_k}$を更新する．SOM に重みベクトルを導入することにより，参照ベクトル毎に各次元の重みを変化させクラスタリング能力の向上を図る（**図 3**）．本研究では，重みベクトル$\boldsymbol{w_k}$の更新には LWC で用いられている重みベクトルの更新式を拡張した式を採用した．更に，Weighted SOM から得られたクラスタリング結果にユーザフィードバックを与えることにより，重みベクトル$\boldsymbol{w_k}$の値を変化させ，ユーザの意図に合ったクラスタリングを行う (**図 4**).

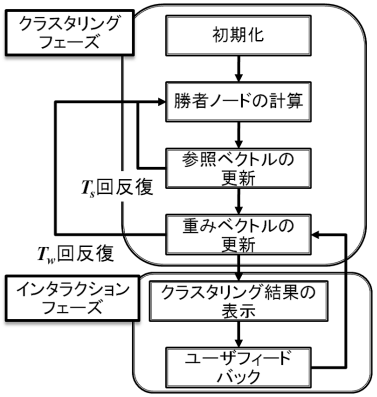


**図 4** 提案手法の流れ図
Fig. 4 Flow diagram of proposed clustering algorithm

### 4.2 勝者ノードの計算

提案手法では，LWC[8] で用いられている重み付きの距離関数 $L_{2,\boldsymbol{w_k}}$ を拡張した重み付き距離 $D_{\boldsymbol{w_k}}$ を使用し，入力データ$\boldsymbol{x_i}$に対して最も距離が小さい参照ノードを勝者ノード $k_i^*$ とする（式 (1)）．勝者ノード $k_i^*$ は入力データ$\boldsymbol{x_i}$と最も距離が小さい参照ノードのことであり，入力ベクトル$\boldsymbol{x_i}$は参照ノード $k_i^*$ に属すると判断される．

Vol.2011-MPS-86 No.9
Vol.2011-BIO-27 No.9
2011/12/1

$$k_i^* = \arg\min_k D_w(\boldsymbol{x_i}, \boldsymbol{m_k}) \tag{1}$$

$$D_w(\boldsymbol{x_i}, \boldsymbol{m_k}) = \sqrt{\sum_{j=1}^{M} w_{kj} |m_{kj} - x_{ij}|^2} \tag{2}$$

### 4.3 参照ベクトルの更新

提案手法では Batch 型 SOM の参照ベクトル更新式を用いる．前節で求めた勝者ノードと入力ベクトルを用いて参照ベクトルの更新を行う（式 (3)）．ここで，$N_{k,r(t)}$ は参照ノード $k$ の近傍半径 $r(t)$ 内に存在する参照ノードを勝者ノードとした入力データの集合である（式 (4)）．$Dist(k_1, k_2)$ は，参照ノード $k_1$ と参照ノード $k_2$ の 2 次元マップ上でのユークリッド距離であり，$n(N_{k,r(t)})$ は $N_{k,r(t)}$ の要素数である．

$$\boldsymbol{m_k} = \frac{\sum_{\boldsymbol{x_i} \in N_{k,r(t)}} \boldsymbol{x_i}}{n(N_{k,r(t)})} \tag{3}$$

$$N_{k,r(t)} = \{\boldsymbol{x_i} \mid Dist(k_i^*, k) < r(t)\} \tag{4}$$

### 4.4 重みベクトルの更新

参照ベクトルを $T_s$ 回更新した後，各参照ノードに分類された入力データと参照ベクトル $\boldsymbol{m_k}$を用いて重みベクトル$\boldsymbol{w_k}$の更新を行う．参照ノード $k$ を勝者ノードとする入力データの集合を $N_{k,0}$ とする．提案手法では LWC で用いられていた重みベクトルの更新式におけるクラスタの中心点$\boldsymbol{c_k}$を，参照ベクトル$\boldsymbol{m_k}$に置き換えた式を重みベクトルの更新に使用する（式 (5)）．重みベクトルを更新し，再び勝者ノードの計算と参照ベクトルの更新を $T_s$ 回行う．重みベクトルが $T_w$ 回更新された後に，インタラクションフェーズに移行する．ここで $M$ は入力データの次元数，$N$ は入力データの数を表す．

$$w_{kj} = \frac{\lambda_k}{\sum_{\boldsymbol{x_i} \in N_{k,0}} |x_{ij} - m_{kj}|^2} \tag{5}$$

$$\lambda_k = \{\prod_{j=1}^{M} \sum_{\boldsymbol{x_i} \in N_{k,0}} |x_{ij} - m_{kj}|^2\}^{\frac{1}{M}} \tag{6}$$

### 4.5 クラスタリングと結果の表示

各入力データはそれぞれの勝者ノードに割り当てられることにより，クラスタリングされる．提案手法では分類されたデータ間の関係がわかりやすいように，2 次元平面で碁盤の目状に各参照ノードと分類された入力データをディスプレイに描画する．

### 4.6 ユーザフィードバック

クラスタリングフェーズによりクラスタリングされた結果を見たユーザは，自分の意図と違った参照ノードに配置されている入力データ$\boldsymbol{x_f}$に対して，適切だと思われる参照ノード $k_f^{'}$ を指定する．いくつかの入力データに対して参照ノードの指定を受けた後，重みベクトルを更新する．以降のクラスタリングにおいて，指定を受けた入力データ$\boldsymbol{x_f}$の勝者ノード $k_f^*$ を $k_f^{'}$ に変更する．入力データ$\boldsymbol{x_f}$は $k_f^{'}$ に分類されているものとして重みベクトルの更新と参照ベクトルの更新を行う．ある入力データに対して，直接最適な配置を指定することにより，cannot-link や must-link といった制約条件よりも，ユーザにとって直感的なフィードバックを与えることができる．

## 5. 実　　験

重みベクトルを付加しない Batch 型 SOM と提案手法のクラスタリング結果を比べ，重みベクトルを SOM に付加することにより，どのようなクラスタリングがなされるかを調べる．また，ユーザフィードバックを受けたことにより重みベクトルと参照ベクトルに変化が起こり，クラスタリング結果にどのような影響が起きるのか調べる．

### 5.1 実 験 方 法

提案手法では入力データの各次元の関連度を重みベクトルで表現する．そのため，次元の大きい入力データにおいて重みの影響が鮮明になると考えられる．そこで，提案手法のクラスタリング結果を調べるために，筆者が先行研究[10] で行ったソフトウェアのクラスタリングを行う．ソフトウェアのクラスタリングでは，単語の出現頻度を元に入力データを作成するため，次元が大きい入力データを扱う．また，ソフトウェアのクラスタリングにおける正しいクラスタリング結果はユーザの好みに大きく影響されるため，ユーザフィードバックがより重要になる．

ソフトウェアのクラスタリングでは，それぞれのソフトウェア名で Web 検索し，検索結果スニペット上位 50 件分から取得した単語の出現頻度を入力データとしてクラスタリングを行う．また，単語の出現頻度を tf-idf[11] を用いて変換することにより，全ての文書に共通して出現する単語や，特定の文書にのみ出現する単語の影響を抑える．今回の実験では上記の手順で作成した 7666 次元のソフトウェアベクトル 29 個に対してクラスタリングを行う．SOM の参照ノードは $6 \times 6$ の碁盤の目状に配置し，近傍半径 $r(0)$ を 2 としている．また，参照ベクトルの更新回数 $T_s$ を 100，重みベクトルの更新回数 $T_w$ を 10 としている．

Vol.2011-MPS-86 No.9
Vol.2011-BIO-27 No.9
2011/12/1

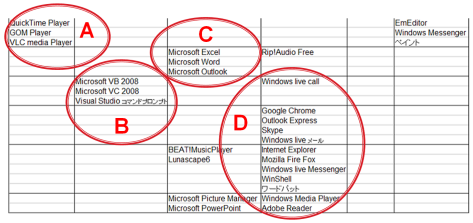


**図 5** Batch 型 SOM でのクラスタリング結果
Fig. 5 Clustering result of Batch version of SOM

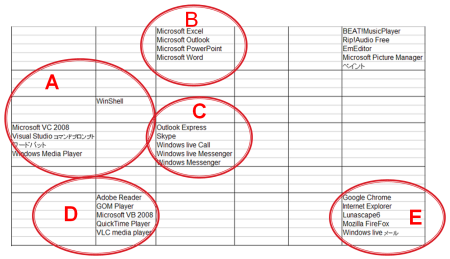


**図 6** Weighted SOM でのクラスタリング結果
Fig. 6 Clustering result of Weighted SOM

### 5.2 実験結果

**図 5** は，Batch 型 SOM を用いてソフトウェアのクラスタリングを行った結果である．この結果から，領域 A に音楽・動画プレイヤー，領域 B に Microsoft Visual Studio，領域 C に Microsoft Office，領域 D にネットワーク関係のソフトがそれぞれ分類されていることがわかる．

次に**図 6** は，提案手法である Weighted SOM を用いてソフトウェアのクラスタリングを行った結果である．この結果からは，領域 A に統合開発環境，領域 B に Microsoft Office，領域 C にチャットソフト，領域 D に音楽・動画プレイヤー，領域 E にインターネットブラウザが分類されていることがわかる．また，クラスタリング後の Weighted SOM の参照ベクトルと重みベクトルの値を調べると，参照ベクトルの値が 0 の次元の重みが大きくなっていることが分かった．

更に，図 6 のクラスタリング結果を出力した Weighted SOM にユーザフィードバックを与えた結果が**図 7** である．ここでは，ユーザフィードバックとして EmEditor の分類を変更している．その結果，フィードバックを与えられた参照ノード付近に存在していた Microsoft Office 関係のソフトウェアが EmEditor と同じ参照ノードに移動し，移動した EmEditor の周辺参照ノードに存在した WinShell がプログラミングのクラスタに移動している．また，Weighted SOM にユーザフィードバックを与える前後の参照ベクトルと重みベクトルの値を比較すると，両者の値が変化していることが確認できた．

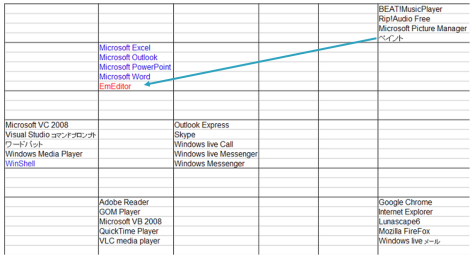


**図 7** Weighted SOM にフィードバックを与えた場合のクラスタリング結果
Fig. 7 Clustering result of Weighted SOM with user feedback

## 6. 考　　察

まず，Batch 型 SOM によるクラスタリング結果（図 5）と Weighted SOM によるクラスタリング結果（図 6）を比べる．両者のクラスタリング結果には音楽・動画プレイヤーのクラスタや Microsoft Office のクラスタなど，共通して出現しているクラスタが存在している．これは，Weighted SOM における重みベクトルが Batch 型 SOM により得られた参照ベクトルを用いて更新されているため，Weighted SOM のクラスタリング結果が Batch 型 SOM の分類の影響を受けているからである．両者に同じクラスタが存在する一方で，Batch 型 SOM ではネットワークソフトとしてクラスタを構成していたソフトウェアが，Weighted SOM を用いたクラスタリング結果ではチャットソフトとインターネットブラウザのクラスタに分割されていることがわかる．これは，重みベクトルを用いることで各次元の関連度が変化したことが原因と考えられる．以上の結果から，Batch 型 SOM のクラスタリング結果を踏襲しつつ，分類能力を向上することができたと考えられる．

今回実験で使用したソフトウェアベクトルはその作成手順から，単語の出現頻度ベクトルであり，要素の大部分は 0 である．また，式 (5) から参照ノード $k$ を勝者ノードとする入

Vol.2011-MPS-86 No.9
Vol.2011-BIO-27 No.9
2011/12/1

力ベクトル集合 $N_{k,0}$ 間で値が近い次元の重みが大きくなるように重みベクトルが更新されることがわかる．このことから，参照ベクトルの値が 0 の次元の重みが大きくなっていると考えられる．つまり，Weighted SOM によるソフトウェアクラスタリングでは，あるクラスタ内に出現しない単語を重視してクラスタリングを行っていることがわかる．

次に，Weighted SOM でのクラスタリング結果にユーザフィードバックを与える前後のクラスタリング結果（図 7）を比べる．今回の実験ではユーザフィードバックとして EmEditor の位置を変更している．この結果では，EmEditor という文書エディタに関係のある Microsoft Office のソフトウェアが EmEditor の近くに集まっていることが分かる．また，TEX の統合開発環境である WinShell は，文書エディタより統合開発環境と関係性が重みベクトルにより強調されて EmEditor と離れる方向に移動したと考えられる．このことから，ユーザフィードバックを与えることにより，クラスタリング結果に変化を与えることができたといえる．

## 7. 終わりに

本研究では，SOM の各参照ノードに重みベクトルを付加することにより，各次元の重要度を調節し，クラスタリング性能を向上させる Weighted SOM を提案した．また，Weighted SOM にユーザフィードバックを与えることにより，ユーザの意図をクラスタリング結果に反映させる手法を提案した．実験により，Weighted SOM では各クラスタの入力ベクトルの値に応じて重みベクトルが変化していることが分かった．また，Weighted SOM においてインタラクションフェーズでユーザフィードバックを与えることにより，参照ノードの重みベクトルを再学習させることができた．重みベクトルの再学習によって，ユーザフィードバックを与えた周囲の参照ノードに分類されていた入力データのクラスタリングを変化させることができた．

今後の課題として，ソフトウェアのクラスタリングだけでなく，他のクラスタリングに提案手法を適用し，手法の有効性を調べる必要がある．更に，ユーザフィードバックには様々なユーザの意図が考えられるため，それらに対応することができる重みベクトルの更新手法を考える必要がある．また，提案手法では入力データが何も分類されていない参照ノードの重みは更新されていない．しかし，SOM ではある参照ノードの近傍に存在する参照ベクトルの値は互いに似ているため，近傍参照ノードにおける次元の重要度も似ている可能性がある．そのため，周囲の重みベクトルを考慮して重みベクトルを更新する手法も考えられる．

## 参 考 文 献

1) 大橋靖雄：“ 分類手法概論 ”，計測と制御，Vol.24，No.11，1985
2) Vladimer Estivill：“ Why so many clustering algorithms: a position paper ”， ACM SIGKDD Explorations Newsletter， Vol.4 Issue 1， 2002
3) Teuvo Kohonen：“ The self-organizing map ”，Neurocomputing，Vol.21，pp.1-6，1998
4) Andreas Nurnberger，Marcin Detyniecki：“ Weighted Self-Organizing Maps: Incorporating User Feedback ”， In Proceeding of the joined 13th International Conference on Artifcial Neural Networks and Neural Information Processing， pp.883-890 ， 2003
5) 山田 誠二，岡部 正幸，高間 康史，小野田 崇：“ 最小ユーザフィードバックの枠組みとその要素技術 ”，知能と情報（日本知能情報ファジィ学会誌），Vol.23，No.1，pp.80-85，2011
6) Masayuki Okabe，Seiji Yamada：“ Learning Similarity Matrix from Constraints of Relational Neighbors ”，Journal of Advanced Computational Intelligence and Intelligent Informatics，Vol.14，No.4，pp.402-407，2010
7) Masayuki Okabe，Seiji Yamada：“ An Interactive Tool for Human Active Learning in Constrainted Clustering ”，Journal of Emerging Technologies in Web Intelligence，Vol.3，No.1，2011
Discovery Journal 14，2007
8) Hao Cheng， Kien A. Hua， Khanh Vu：“ Constrained Locally Weighted Clustering ”， The Proceedings of the VLDB Endowment， 2008
9) Lance Parsons，Ehtesham Haque，Huan Liu：“ Subspace Clustering for High Dimensional Data: A Review ”，ACM SIGKDD Explorations Newsletter，Vol.6，pp.90-105，2004
10) 久田大地，吉川毅，野中秀俊：“ 自己組織化マップを用いたソフトウェア分類表示手法 ”，第 26 回ファジィシステムシンポジウム，2010
11) Donna Harman：“ An Exerimental Study of Factors Important in Document Ranking ”，SIGIR ’86 Proceedings of the 9th annual international ACM SIGIR conference on Research and development in information retrieval，pp.186-193，1986