データベースシステム 116－5
（1998. 7. 8）

自己組織化マップを用いた Web 文書の対話的分類とその視覚化

佐野 綾一 † 波多野 賢治 ‡ 田中 克己 ‡

†神戸大学大学院自然科学研究科情報知能工学専攻
‡神戸大学大学院自然科学研究科情報メディア科学専攻

本稿では，Kohonen の自己組織化マップ (Self-organizing Map) を用い，検索エンジンの出力結果を動的に分類・視覚化し，さらにユーザとの対話に基づく適合フィードバック (Relevance Feedback) 機能を持たせることにより，ユーザの視点を反映したマップ生成ができる Web 文書分類機構を提案する．

この機構では，検索エンジンによる検索結果によって得られるページをページ中の単語の頻度情報を基に特徴づけ，そうして得られた特徴ベクトルを自己組織化マップで学習させ表示させることで検索エンジンの検索結果の動的分類や曖昧検索可能な URL マップの生成が可能である．さらに，ユーザとの対話を行うことで適合フィードバックを行い「ユーザ本位のマップ」を生成することができる．これにより，検索エンジンの利用の際にかかるユーザの労力の軽減を図ることが可能となった．

Clustering and Visualizing of Web Documents using Self-Organizing Map

Ryouichi Sano † Kenji Hatano ‡ Katsumi Tanaka‡

†Division of Computer and Systems Engineering,
Graduate School of Science and Technology, Kobe University

‡Division of Information and Media Sciences,
Graduate School of Science and Technology, Kobe University

In this paper, we propose a clustering search engine for Web documents, based-on Kohonen's Self-Organizing Map(SOM). This system provides not only dynamically classified and visual results but also interactive operations on user's demand. Moreover, by including the relevance feedback ability in the prototype system, we propose a extension type, which is able to form visual map based on user's view point. This system have been provide to be quite effective and reduce user's labor.

1 はじめに

近年の WWW (World Wide Web) の発展は目覚しく，その規模は数千万ページに及んでいる．このような発展に伴い，莫大な規模となった情報の中から目的のページへとリンクを頼りにたどり着くことは実質不可能となっている．そこで，通常必要とする情報を収集する為に検索エンジンを利用し，キーワードを入力することで数千万ものページをフィルタにかけ，その中から自分の欲しい情報に該当するページを得ている．しかし，検索エンジンによってフィルタリングされた検索結果であっても何百件になることは日常茶飯事で，

その中にどのような情報が含まれているかを知るには 1 つ 1 つページを閲覧していかねばならず，ユーザにとって大変な労力を要する．これは検索エンジンの検索結果が，情報に該当するページの URL リストであるというインターフェースに問題があるのではないかと考えられる．

そこで本研究では，該当ページの URL のフィルタリングだけではなく，フィルタリングの結果に対して動的な分類や曖昧検索が可能なブラウジング機構を提供し，さらにユーザとの対話を行うことのできる適合フィードバックの機能を持ったユーザビュー機構を提案する．

本システムでは，検索エンジンの検索結果のページに含まれる単語の出現頻度を基に各ページの特徴ベクトルを生成し，それを自己組織化マップ (Self-Organizing Map，以下 SOM と記す) に学習させることで，検索エンジンの検索結果に対して動的な分類および曖昧検索を可能にする機能を実現している．

さらに，ユーザとの対話を行うことで適合フィードバックを行い「ユーザ本位のマップ」を生成することができる．これにより，検索エンジンの利用の際にかかるユーザの労力の軽減を図っている．

2 自己組織化マップ (SOM)

ニューラルネットワークの一種である自己組織化マップ (SOM) [3, 4] は教師無し競合強化学習モデルである．出力層の各セルが層の中で位置を持つという点が他の学習モデルと異なる．データに隠されているトポロジカルな構造を学習アルゴリズムにより発見し，通常 2 次元空間で表示するという特徴を持っているため，特徴のよく似たデータ同士は出力マップ上の近い位置に配置されるようになっている．生成されたマップはそれぞれのデータの位置関係によって類似しているデータかどうかが直観的に理解しやすいという点からシステムの視覚化に利用できる．SOM には様々な種類があるが，ここでは最も基本的なものについて述べる．

SOM で用いられるネットワークは，セルを 2 次元に六角格子状に配置したものである．それぞれのセル i はセルの特徴ベクトル $\boldsymbol{m}_i(t) \in R^n$ (R は実数) を持っており (t は時間を表し，$\boldsymbol{m}_i(0)$ は適切な方法で初期化されている)，これらのセルの特徴ベクトルを；入力である特徴ベクトル $\boldsymbol{x}_j \in R^n (j = 1, 2, \cdots, d)$ に選択的に近づけることによって学習は進行する．このとき，SOM では入力となる特徴ベクトルに一番近いパターンを持つ出力セルおよびその近傍のセルの集合のみが入力ベクトルに近づくことができるようなアルゴリズムをとる．

SOM のアルゴリズムを以下に示す．

1. 各入力特徴ベクトルを生成し，その集合を X とする．

$$X = \{\boldsymbol{x}_j \mid \boldsymbol{x}_j \in R^n, j = 1,\ 2,\ \cdots,\ d\}$$

2. 出力層にある各ユニットの持つパターンを初期化する．

$$M = \{\boldsymbol{m}_i \mid \boldsymbol{m}_i \in R^n,\ i = 1,\ 2,\ \cdots,\ k\}$$

(ただし，$\boldsymbol{m}_i(0) = [\ 0,\ 0,\ \cdots,\ 0\ ]$ とした)

3. T をあらかじめ設定された学習回数とする．このとき，$t = 0, 1, \cdots, T$ について以下を繰り返す．

   (a) $\boldsymbol{x}_j$ に最も近いセル c を探す．つまり，$\|\boldsymbol{x}_j - \boldsymbol{m}_c(t)\|$ を最小にするセル c を求める．

   (b) 探し出したセル c の特徴ベクトル $\boldsymbol{m}_c$ を更新し，さらにその近傍のセルの集合 N_c も入力パターンに近づける．

$$\boldsymbol{m}_i(t+1) = \begin{cases} \boldsymbol{m}_i(t) + \alpha(t)[\boldsymbol{x}_k(t) - \boldsymbol{m}_i(t)] \\ \qquad (i \in N_c(t)) \\ \boldsymbol{m}_i(t) \quad (i \notin N_c(t)) \end{cases}$$

   N_c の中央はセル c である．N_c の半径は，学習の初期段階ではたいてい大きく，学習を繰り返していくうちに単調に減少させる．また，$\alpha(t) \in (0, 1)$ は「学習率」を表し，これもまた時間と共に単調に減少させる．

$$\alpha(t) = \alpha_0(t) exp(-\frac{\|\boldsymbol{r}_c - \boldsymbol{r}_i\|^2}{\sigma(t)^2})$$

   ただし，$\boldsymbol{r}_c$ と $\boldsymbol{r}_i$ はそれぞれセル c とセル i のもつベクトルを表す．$\alpha_0(t)$ や $\sigma(t)$ には単調減少の一次関数や指数関数がよく用いられる．

(c) $j = 1, 2, \cdots, d$ について上記 (a), (b) を繰り返す.

3 漸次的情報組織化

我々は，これまでに SOM を文書データベースのブラウジングツールおよび問い合わせインターフェースと見なして，自動的な構造化や分類を支援するシステムを開発してきた [1]. しかし，このシステムでは，

- 分類するデータはあらかじめユーザが収集して準備しておかなければならない.
- マップ上のキーワード同士の関連を見ながらの検索だけでは，目的の情報にたどり着くまでに時間がかかる.
- ユーザ側の意図や興味をマップに反映することができないため，ユーザ本位のインターフェースを提供できない.

といった問題点があった.

このような問題点を解決するために，システムに対してユーザのインタラクションを取り入れる研究が始まっている. Kohonen の研究グループでは，文書中に現れる単語のカテゴリ分類を行った上で文書の分類を SOM を利用して行っているが，カテゴリ分類の際に単語を選択したり，データをブラウズする際にマップ上でデータの詳細表示を行うといったユーザの操作がシステムに反映されるような工夫がなされている [2, 5]. これに対し，我々は，システムが生成したマップを WWW 検索エンジンとの連動機能を含めた適合フィードバックにより，ユーザとのインタラクションを行い，マップを再構成することでよりユーザに依存した検索を可能とし問題点を解決する.

本システムの全体構成を図1に示す. ここでは，SOM による検索エンジンの検索結果の漸次的情報組織化について述べていく.

3.1 マッチページの特徴ベクトルの生成

SOM を利用した検索エンジンの検索結果に対する情報組織化を行うには，検索エンジンの検索結果からキーワードにマッチしたマッチページの URL を得て，その HTML 文書のソースから各マッチページの特徴ベクトルを生成しなければならない.

以下に，その特徴ベクトルの生成過程を示す.

1. **検索エンジンの検索結果の獲得**

   本研究では，自己組織化をする検索エンジンの対象として goo [7] を使用した. 検索エンジンで検索を行うようにユーザはブラウザ上のテキストフィールドにキーワード $(k_1, k_2, \cdots, k_m)$ を入力すると，検索エンジンにより検索結果の URL が返され，マッチページの HTML ソースを得ることができる.

2. **特徴ベクトルの生成**

   得られたマッチページの HTML ソースファイル h から特徴ベクトル $\boldsymbol{F}(h)$ を生成する.

   (a) 得られたファイルから取り出された単語の出現頻度を全て調べる.

   (b) 出現頻度の高い順に並べ替えて，その上位 500 の単語 $(t_1, t_2, \cdots, t_{500})$ を特徴ベクトルの要素とする.

   (c) 要素 $(t_1, t_2, \cdots, t_{500})$ の各マッチページ h_l での出現頻度 $(f_1, f_2, \cdots, f_{500})$ を求め，各マッチページ内の全出現単語数 N_l で正規化することによって，各マッチページのベクトル

   $$\boldsymbol{F}(h_l) = (\frac{f_1}{N_l}, \frac{f_2}{N_l}, \cdots, \frac{f_{500}}{N_l})$$

   が生成される.

3.2 ユーザインターフェース

SOM は，入力データ群をトポロジカルマッピングの性質により 2 次元空間に表示することができるため，マップを生成するには大変有効な手段であると考えられる. そこで，我々は SOM に VRML ベースのインターフェースを採用し，マップ中のセルを選択することでそのセルに分類されたマッチページの URL とセルに含まれる 3 つのキーワードのデータが WWW ブラウザで参照できるように設計した.

さらに，今回これに適合フィードバックの機能を実現することで，より検索結果の URL マップの分類状況が把握しやすいようになっている.

以下にインターフェースの機能を示す.

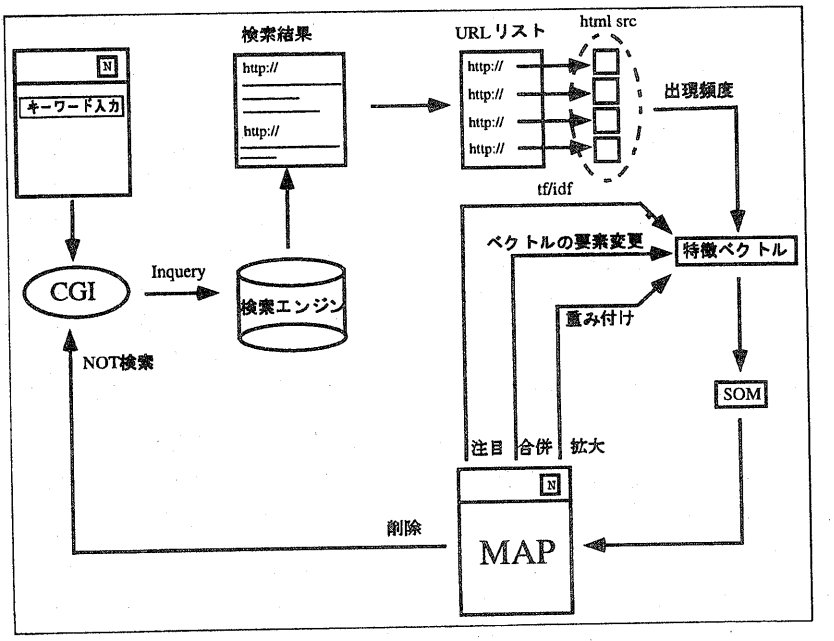

図 1: システムの全体構成

- **マップの3次元表示**

  出力セルが円柱型になっており，円柱の高さはそのセルに分類されたマッチページの数に比例している．これにより，外観からマッチページの数を把握できる．

- **各出力セルのキーワード**

  各出力セルの特徴ベクトルの要素中，最も大きな値を持つ要素，すなわちその出力セルのベクトルに含まれる単語群の中で最も出現頻度が高いものをそのセルのキーワードとする．同様に，出現頻度が 2 番目に高いもの，3 番目に高いものを選び各セルを 3 つのキーワードでラベルづけする．

- **マップの領域分割**

  各セルにつけられたキーワードの内，第 1 キーワードが同じものを 1 つの領域とし，色分けすることで各セルの区別をすることができる．

- **マップの詳細表示**

  第 1 キーワードによって色分けされた領域 (以下 この領域を詳細マップと呼ぶ) を選択し，1 つの領域のみの表示もできる．詳細マップは，第 2 キーワードの同じものが同色で色分けする．これにより，選ばれた領域をさらに細かい領域で見ることができる．

- **適合フィードバック機構**

  得られた URL のマップに対して，ユーザの意図や興味を反映させる機能は，ユーザが欲している情報を効率的に検索する上で大変重要な機能である．本システムにおいては，適合フィードバック機構を実現するために以下の機能を付加している (図1参照)．

  - **領域の拡大**

    マップ上の領域を指定することにより，その領域を表すキーワード k の要素 t_k が最も大きい特徴ベクトルの URL が分類されているセル c を探し出し，そのセルを中心に特徴ベクトルの要素 t_k に同心円状に重み付けの再計算

    $$\boldsymbol{F}(h_l) = (\frac{f_1}{N_l}, \frac{f_2}{N_l}, \cdots, w(r)\cdot\frac{f_k}{N_l}, \cdots, \frac{f_{500}}{N_l})$$

    $$w(r) = \frac{w(r-1)+1}{2}, \quad w(0) = 5, \quad r \geq 1$$

(ただし r は c からの半径) を行い，SOM によって学習させることことにより，キーワード k の重要度が高いマッチページをマップ上に浮かび上がらせることを目的としている．

- **領域の削除**

  ユーザがマップ上の不要な領域を指定することにより，その領域を表すキーワード k を含んだマッチページの削除，すなわち WWW との連動により いわゆる NOT 検索ができる機能を持たせている．この場合，ユーザは意識することなく新たな質問 q が生成され，検索エンジンに問い合わせが行われるようになっている．そして，再び特徴ベクトルを生成する．

$$q = k_1 \wedge k_2 \wedge \cdots \wedge k_m \wedge \neg k$$

- **領域の合併**

  マップ上の複数の領域 (キーワード) $K = (k_0, k_1, \cdots, k_n)$ を 1 つにまとめたい場合に用いる方法．例えばキーワードに "red" や "white" など色に関するキーワードが数多く出ている場合，これらを "color" などと 1 つのキーワードにまとめて見やすいマップを再構成することを目的としている．

$$\boldsymbol{F}(h_l) = (\frac{f_1}{N_l}, \frac{f_2}{N_l}, \cdots, \sum_{j=0}^{n} \frac{f_{k_j}}{N_l}, \cdots, \frac{f_{500}}{N_l})$$

- **領域の注目**

  ユーザはマップ上の必要な領域のキーワードを選択することにより，選ばれた領域内にあるページの特徴ベクトルを再生成し，そのページだけでのマップを生成する．その際に，単純に単語の出現頻度によって特徴ベクトルを生成するのではなく，tf/idf 法[1]と呼ばれる方法 [6] を用いることで特徴ベクトルを再生成する．

$$T = (t'_1, t'_2, \cdots, t'_{500}), \quad j = 1, 2, \cdots, 500$$

$$tf_j = \frac{f'}{N_l}, \quad idf_j = log(\frac{M}{df_j})$$

$$\boldsymbol{F}(h_l) = (f''_1, f''_2, \cdots, f'_{500}), \quad f''_j = tf_j \cdot idf_j$$

(ただし，h の総数を M ，t_j を含む h の数を df_j) 実験によるとマップ全体を見渡す際には，出現頻度による特徴ベクトル，目的の情報を絞り込むという際には tf/idf 法がよい結果を得られているため [1]，ユーザの欲している情報を絞り込む際に有効となる手段である．

[1]高頻度で現れる，少ない数の文書にしか現れないといった文書の特徴によってキーワードの重要度を測る方法

4 実行例

実際に 20×20 のマップを用いてマップ生成を行った．"worldcup" という検索キーワードで得られた検索エンジンの検索結果 400 件を 3.2 節の方式に従って特徴ベクトルを生成し SOM で学習させると，セル数 400 のマップは 5 分程度で生成された．マップは，"world", "football", "france", "soccer", "description", "apr", "coupe", "french", "communication", "paris", "jun", "english", "ski", "korea", "main", "oct", "dec", "nov", "sep" のキーワードによって，19 の領域に分割された (図2 参照)．

3.2 節で述べたユーザインターフェース機能により，ユーザはこの 3 次元マップを検索エンジンの検索結果のブラウジングツールおよび問い合わせツールとして利用できる．マップ上のセルをクリックすると，3 つのキーワード (図2の場合は，"coupe", "football", "mondial" である)，適合フィードバックへのリンク，およびセルに振り分けられた URL のデータが WWW ブラウザに表示される．また，マップ上のラベルをクリックすると詳細マップを見ることができる．

「拡大」,「削除」,「合併」,「注目」の適合フィードバックの機能が選ばれると，マップ上のキーワードを選択する画面 (図3) が表示され，そのキーワードを選ぶことによりそれぞれの領域に対して適合フィードバックが行われる．

4.1 拡大

"worldcup" というキーワードによって生成されたマップ内の "france" というキーワードで,「拡大」の適合フィードバックを行ってみた．マップ

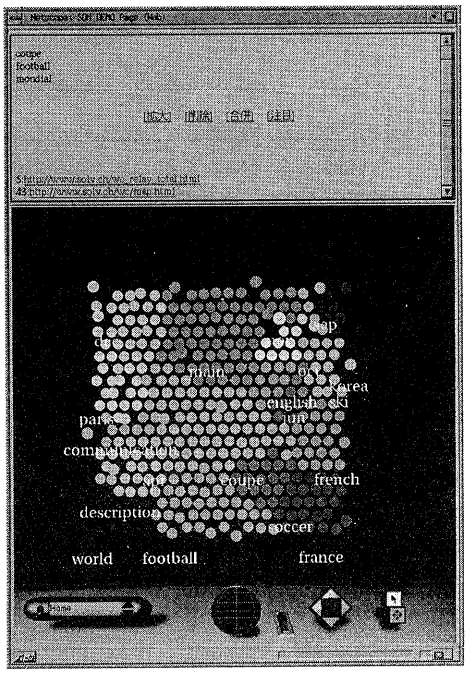

図 2: 実行例

上で “france” を第 1 のキーワード，第 2 のキーワード，第 3 のキーワードとして分類されているマッチページ数のそれぞれの変化は図 1 のような増加が見られた．

また “france” というキーワードの場合のみならず，他のキーワードの場合にも同じような変化が見られた．

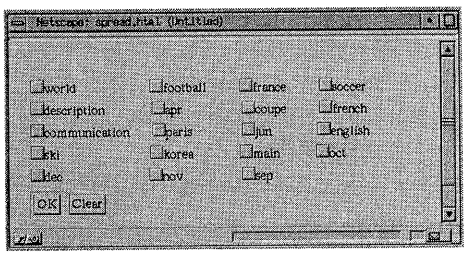

図 3: キーワード選択画面

表 1: 拡大によるマップ上の変化

	拡大前	拡大後
第 1 キーワード	27	32
第 2 キーワード	15	17
第 3 キーワード	1	9
合計	43	58

4.2 削除

“ski” というキーワードで,「削除」の適合フィードバックを行ってみた結果，図4のようなマップが生成された．マップは，“description”, “dec”, “jun”, “sep”, “nov”, “apr”, “french”, “soccer”, “jul”, “prev”, “paris”, “sports”, “world”, “message”, “board”, “france”, “main”, “usa”, “coupe”, “aus”, “groupe” のキーワードによって，21 の領域に分割され，“ski” というキーワードがマップ上には表れず，また特徴ベクトルの要素の中にも表れていないことから，正しく削除の機能が働いたことが分かる．

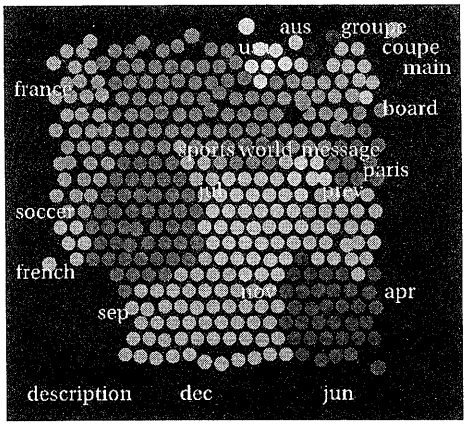

図 4: 削除の適合フィードバックの実行結果

4.3 合併

「合併」は，複数のキーワードの特徴ベクトルの要素を加えて，ある 1 つのキーワードに置き換える機能である．最初に生成されたマップ (図 2) の “apr”, “jun”, “oct”, “dec”, “nov”, “sep” の

キーワードを “month” というキーワードに置き換えて，特徴ベクトルを再生成し SOM に学習させてみたところ，“description”, “french”, “month”, “soccer”, “korea”, “paris”, “ital”, “usa”, “main”, “world”, “green”, “france”, “groupe”, “communication”, “coupe”, “ski”, “english” のキーワードによって，17 の領域に分割された．図5のマップにおいて，合併された “month” という領域がマップの右下の 4 分の 1 を占める大きな領域を作っているのが分かる．

この合併の機能によりユーザにとって見やすいマップを再構成することができる．

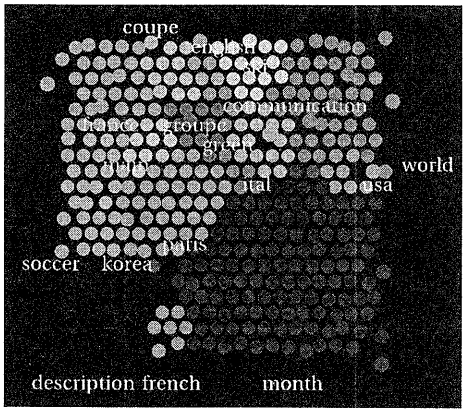

図 5: 合併の適合フィードバックの実行結果

4.4 注目

「注目」は，ユーザの欲している情報を最終的に絞り込むために使われる機能である．ここで，出現頻度によって「注目」の機能を実現した場合と tf/idf によって実現した場合のマップの分割状況を見てみると，出現頻度の場合は25程度，tf/idf を利用した場合は35程度の領域に分割された．

5 評価

まず，文献データベースの場合の実験 [1] では欲しいデータを絞り込むために有効な手段であった tf/idf 法であるが，検索エンジンの出力結果のような文書の集まりである本実験のような場合では，検索エンジンによってフィルタリングされたページの中で，ノイズになるような単語が重要度を増し，結果としてそういった単語によりマップが領域分割されてしまうためマップ上からの検索を行うのに有効なキーワードを取り出すことは出来なかった．

そこで，「注目」の適合フィードバックの機能の場合も，単語の出現頻度により特徴ベクトルを生成するものとした．

実際，行った実験がどれほどの結果なのかを判断するために，検索精度の評価に情報検索で一般的に用いられている「精度(適合率)」と「再現率」を測定する．「精度」と「再現率」は，検索されなかった適合情報を A，検索された適合情報を B，検索された不適合情報を C とした場合，精度は $B/(B+C)$，再現率は $B/(A+B)$ で表されるものであるが，本研究で利用したものは少し定義が異なる．ここで用いた「精度」と「再現率」は次のように定義する．

全てのページのの中から，マップ上において領域を作っているキーワード k を特徴として表すページの数を $similar(k)$ とし，k の領域に分類されているページの数を $neighbour(k)$ で表わすとする．この下で精度は，

$$\frac{|neighbour(k) \cap similar(k)|}{|neighbour(k)|}$$

で表わされ，また再現率は，

$$\frac{|neighbour(k) \cap similar(k)|}{|similar(k)|}$$

で表わされる．

今回の様に検索エンジンの出力結果というものは，「情報が適合する，しない」という評価は主観的であり，ユーザによって異なるものであるので，ここでは第一著者の基準でページの内容を評価した．検索エンジンの出力結果に適合フィードバックを施す前と適合フィードバックを施した後の精度と再現率の変化を表2 に示す．

表 2: 精度と再現率の評価

	精度(%)	再現率(%)
前	32.47	42.49
後	59.27	36.28

精度は，領域内に分類されているページがキーワードに適合している割合を示し，再現率は，マッ

プ内のページがどれだけ正しく領域に分類されているかを示す．このことから考えてみると，適合フィードバックを行うことにより精度が上昇しているので，マップ上のキーワード同士の関連を見ながらの検索において，目的の情報にたどり着くまでのユーザの労力が軽減できているといって良い．また，一般の情報検索では，およそ 20% の再現率の時 40% くらいの精度[8] であるといわれていることからも良好な結果が得られていると思われる．

6 おわりに

我々は SOM を用い，検索エンジンの出力結果を動的に分類・視覚化し，さらにユーザとの対話を行うことによりユーザの視点に基づいたマップの生成機構，つまり適合フィードバックの機能を持った Web 文書分類機構を提案し，検索エンジンを利用した Web 文書の検索の際に生じるユーザの労力の軽減を図った．
本システムの利点として

- 3次元上のマップにより，全ページの相対的な状態を見渡すことが可能
- 検索時に漠然としたキーワードしか持っていなくても情報の絞り込みが可能
- 情報を収集するのために必要な新しいキーワードの発見
- 適合フィードバックを施すことによる，ユーザの労力の軽減とマップ上からの曖昧検索の時間短縮

が挙げられる．また今後の課題として

- *tf/idf* 法による特徴ベクトル生成の際に生じるノイズの除去を考慮した特徴ベクトル生成法の確立
- Web 文書のリンク先との関係やリンク先の情報を付加した特徴ベクトルの生成
- 1 ページ単位での分類ではなく，複数のページを 1 つの単位として分類するための分類法の提案

などが考えられる．

謝辞

この研究は，一部，日本学術振興会未来開拓学術研究推進事業における研究プロジェクト「マルチメディア・コンテンツの高次処理の研究」および文部省科学研究費重点領域研究「高度データベース(No.275)」(課題番号 08244103) による．ここに記して謝意を表します．

参考文献

[1] K. Hatano, Q. Qian, and K. Tanaka. A SOM-Based Information organizer for text and video data. In *Proc. of the 5th International Confarence on Database Systems for Advanced Applications (DASFAA'97)*, pp. 205–214. World Scientific, Apr. 1997.

[2] T. Honkela, S. Kaski, K. Lagus, and T. Kohonen. WEBSOM - self-Organizing maps of document collections. In *Proc. of the Workshop on Self-Organizing Maps (WSOM'97)*, Jun. 1997.

[3] T. Kohonen. The self-organizing map. *Proceedings Of The IEEE*, Vol. 78, No. 9, pp. 1464–1480, 1990.

[4] T. Kohonen. *Self-Organizing Maps*. Springer, Berlin, 1995.

[5] K. Lagus, T. Honkela, S. Kaski, and T. Kohonen. Self-organizing maps of document collections: A new approach to interactive exploration. In *Proc. of the 2nd Int. Conf. on Knowledge Discovery and Data Mining (KDD-96)*, pp. 238–243, 1996.

[6] G. Salton, J. Allan, and C. Buckley. Automatic Structuring and Retrieval of Large Text Files. *Communications of the ACM*, Vol. 37, No. 2, pp. 97–108, Feb. 1994.

[7] NTT ヒューマンインターフェース研究所. goo パワーサーチ. http://www.goo.ne.jp/.

[8] 情報処理学会. 情報処理ハンドブック. 株式会社オーム社, 11. 1997.