クラスタリングアルゴリズムを利用した WWW情報整理システムの作成と評価

中村 順一(京大, 九工大) 村井 幸一 馬場 博巳 甲斐 郷子(九工大)

1 はじめに

インターネットの普及に伴い, WWW(World-Wide Web)の利用が一般にも広がってきた. WWWシステムには, 複数のサーバ中から目的の情報を探す手段が用意されていないという問題がある. そこで, 本研究室ではWeb情報探索を支援するため, Web上の任意のページからリンクで辿れる一定数の情報を自動的に整理し, 利用者が目的の情報に簡単に辿りつけるようにするWeb情報整理システム[1]の作成を行なっている.

従来のシステムでは, 情報整理の方法はT.Kohonen[2]が提案した自己組織化マップアルゴリズムの手法を用いている. Kohonenアルゴリズムには, 情報整理をおこなう学習時間が長く, 学習を収束させる判断がつきにくいという問題があった. そこで本研究では, K-mean(K-平均)クラスタリグアルゴリズムをKohonenの手法に組み合わせることで情報整理の精度を落とさずに実行時間の短縮をおこなえるようにして, これらの問題の解消を行なった.

本稿では, アルゴリズムの概要と, 従来の手法と比較するためにおこなった評価実験の結果を述べる.

2 情報整理アルゴリズム

本システムでは, 情報整理の対象となる情報をHTMLファイルのみとしている. ユーザからの指示により, 情報整理を行う情報(HTML)を自動的に収得する.

2.1 情報のベクトルへのコーディング

収得した情報に対して, $tf \times idf$ 法[3][4]を利用して各ページをベクトルパターン化することで情報の特徴を数値化する.

情報 H_i 中に現れる単語を日本語形態素解析システムJUMAN[5]により品詞分類を行ない, 普通名詞と固有名詞を抜き出し利用単語 t_j とする. ここで, t_j の H_i に対する出現頻度を F_{ij}, H_i の総数を N, t_j が含まれている情報ページの総数 N_j, t_j の文字数を L_j とすると, t_j の H_i に対する重要度 w_{ij} は式1で求める.

$$w_{ij} = F_{ij} \times \log\left(\frac{N}{N_j}\right) \times \log L_j \qquad (1)$$

重要度の高い一定数の単語を利用単語とすると, H_i は式2のようなベクトルへコーディングできる.

$$H_i = (w_{i1}, \ldots, w_{ij}, \ldots, w_{in}) \qquad (2)$$

各 H_i についてベクトル化を行い, システム全体として式3のパターンを得る.

$$X = \{H_1, \ldots, H_i, \ldots, H_N\} \qquad (3)$$

X をアルゴリズムへの入力パターンとする.

2.2 自己組織化アルゴリズム

従来の情報整理の手法には, Kohonenが考案した多次元ベクトル化した情報群を二次元マップに整理して配置する学習モデルである自己組織化マップを用いた. 入力パターンと同次元のベクトルを持つユニットの集合である二次元マップとの間で繰り返し学習を行なうことで, マップの各ユニットに特徴を持たせる[6].

学習が終わると, 各HTMLファイルは自分の特徴(入力パターン)と最も似た特徴(パター

ン)を持つユニットに配置されるため,結果として情報内容の近いものどうしがマップの一部分にまとまる.その結果,ユーザーは欲しい情報がありそうなユニット,及びその近傍のユニットを見ることで,目的の情報に近い情報を収得できる.

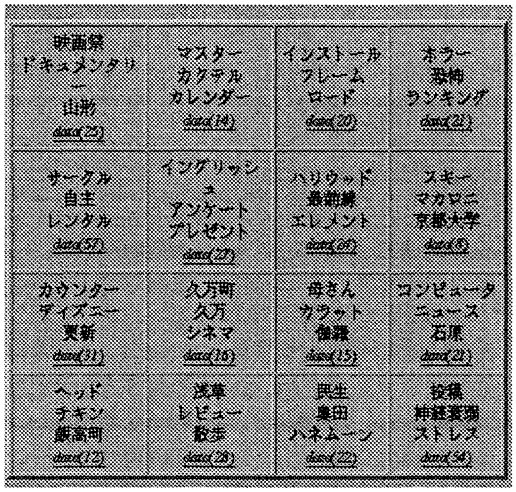

図 1: 結果表示画面

整理結果は図1のようにHTML形式のファイルとして出力される.マップの各ユニットにはそのユニットの最大の特徴を表す単語の上位三つ,及びそのユニットに配置されたページ数が表示されている.ここでユーザが目的の情報に関するページが配置されていそうなユニットをクリックすると,今度はそのユニットに配置されたページのタイトル一覧が表示されているページに切り替わる.ユーザはリストされたタイトルから目的の情報に関するタイトルを選び,そのタイトルの部分をクリックすることで,目的の情報にアクセスすることができる.

2.3 K-mean法

K-mean法は全サンプルパターンのクラスタリングを,ある収束状態になるまで反復することをおこなう[7].入力パターンをX,分類したいクラスタ数をK,反復回数の上限をTとすると,K-mean法のアルゴリズムは以下のようになる.

1. Xから任意のK個を選んでクラスタ中心c_iとする.[1]

$$c_1(0), c_2(0), \ldots, c_K(0) \qquad (4)$$

2. 以下の処理を$t=0$より$t<T$までに対して反復する

   (a) 各H_iを最も近い$c_i(t)$にクラスタリング[2]

   (b) 新しいクラスタ中心$c_i(t+1)$を$c_i(t)$にクラスタリングされた全パターンの平均値で定める

$$c_i(t+1) = \frac{1}{N_i(t)} \Sigma x \qquad (5)$$

   $N_i(t)$は$c_i(t)$に所属されたパターンの個数,Σxは$c_i(t)$に所属されたパターンの和

   (c) クラスタ中心が一つも変化しない,すなわち

$$c_i(t+1) = c_i(t), \forall i \qquad (6)$$

   であるならば,終了.その時,$c_i(t)$の値を最終的なクラスタリングの結果とする.

   (d) t←t+1 として 2a へ

以上のようにアルゴリズムが単純で,一般に少ない反復回数で収束することが知られている.一方,初期クラスタ中心の決め方によってクラスタリングの結果が大きく変わるため,本研究ではその決め方に工夫を行なった.以下に手法を示す.

1. i=0 番目の中心$c_1(0)$を任意に選ぶ
2. $c_i(0)$に一番遠いものをi+1番目の中心$c_{i+1}(0)$とする

[1] c_iは反復回数tにより変化するため$c_i(t)$と表わせ,現段階は$t=0$

[2] クラスタリングはコサイン値による類似度計算を行なう

3. $c_i(0)$ と $c_{i+1}(0)$ の中間点のものを i+2 番目の中心 $c_{i+2}(0)$ とする

4. i←i+2, $i < K$ の間2～4を繰り返す

これにより, お互いの距離が遠いもの同士が選ばれる.

2.4 Kohonen と K-mean の融合

Kohonen アルゴリズムの問題を解消するため, K-mean 法を Kohonen に組み合わせる. 具体的には, 始めに K-mean 法でクラスタリングをおこない, その結果を次に Kohonen のマップのユニットの初期値にそのまま利用する. ある程度の収束状態から学習を始めるので, その間の時間を短縮でき, より精度の高い結果が得られる.

3 評価実験

Kohonen に K-mean を組み合わせる有効性を調べるため, 評価実験を行なった. 評価のもととなる情報ソースは, 検索システムの Yahoo! Japan[3] と Goo[4] で, キーワード「映画」と「料理」の二つのそれぞれの単語に対して検索した結果の 400 ページ前後を用いた. それらの情報ソースに対してマップサイズ 6×6 で, 以下の5つのアルゴリズムを適用する.

Kohonen(10)

Kohonen アルゴリズム学習回数 10 回

Kohonen(100)

Kohonen アルゴリズム学習回数 100 回

K-mean(先頭) → Kohonen

K-mean 法クラスタ中心の初期値を, 先頭から K 個選び実行した後, その結果を Kohonen の初期ユニット値とし学習回数 10 回で実行する.

K-mean(乱数) → Kohonen

K-mean 法クラスタ中心の初期値を, ランダムに K 個選び実行した後, その結果を Kohonen の初期ユニット値とし学習回数 10 回で実行する.

K-mean(距離) → Kohonen

K-mean 法クラスタ中心の初期値を, 前節で述べたように遠方のもの同士が選ばれるように K 個選び実行した後, その結果を Kohonen の初期ユニット値とし学習回数 10 回で実行する.

3.1 評価方法

評価方法は, 結果のマップでのカテゴリごとの配置状況を調べるため, 検索単語である「映画」と「料理」について二つのカテゴリに注目し, 各情報に対し 0～3 点の得点付けを行なった. それらの情報がマップのどのユニットに配置されているか調べ, ユニットの得点をそこに配置されたページの得点の合計点で与える. マップ上での得点の配置状況を 8 連結手法でグループ化を行い, 各グループがマップの一部にまとまっているかをグループのモーメント値を計算することで評価した.

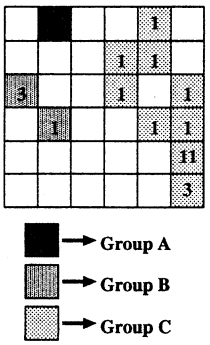

図 2: グループ分け

モーメント値の計算方法は, グループでの重心点を求め, 重心点から, グループ内の各ユニット i の座標 (x_i, y_i) までの距離を d_i, i の得点を

[3] http://www.yahoo.co.jp/

[4] http://www.goo.ne.jp/

p_i とすると, モーメント値 M は式7で計算する.

$$M = \sum_{i=1}^{n} d_i p_i \qquad (7)$$

モーメント値 M が小さな値を示すグループほど, マップの一部にまとまって収束していることになる.各グループのモーメント値の合計が, そのマップの評価値となる.

図2の例では, グループ A, B, C, の三つに分けられる. A のモーメント値が 0, B が 2.12, C が 26.37 である. それらの合計値 28.49 がこのマップの評価値である.

3.2 実験結果

アルゴリズムごとにグループのモーメント値を合計して, それらのモーメント値と, グループ数の平均値を求めたものを, 表1に示す.

表 1: アルゴリズムごとの平均値

アルゴリズム	モーメント値	グループ数
Kohonen(10)	32.87	3.25
K-mean(先頭) → Koho	26.95	3.25
K-mean(乱数) → Koho	22.73	4.25
K-mean(距離) → Koho	23.06	4.0
Kohonen(100)	25.67	4.0

グループ数は, 平均3～4個の間で大きな差はなく同じようなカテゴリ分けがされた. モーメント値が一番悪いのは, Kohonen(10) がただ一つ 30 ポイント台である. その中でも良い値を示したのが順に, K-mean(乱数), K-mean(距離), 次いで Kohonen(100) が続く.

K-mean(乱数) について, 乱数で初期値を決めるのは偶然性によるところが大きく, 常に同じ様な結果になるとは考えにくい. これは, 初期値が先頭でも同じことである. その点, K-mean(距離) は偶然性に負うところは少ないため, K-mean(距離) を利用する方が有効的であると考える.

Kohonen(100) の実行時間は, 今回の実験で 50 分程度かかる[5]. それに対し, 本研究で考案した K-mean を利用したタイプでは約 10 分程度で終了する. モーメント値では, k-mean を利用した場合と Kohonen のみで長時間学習をさせた結果には大差がなく, 二つの間に遜色はみられない. 同じ結果を得られるなら, 実行時間がより短い方が良いわけだから, K-mean を組み合わせた利点が確認できた.

参考文献

[1] 中村 順一, 中尾 学: "自己組織化マップを利用した Web 情報整理システムの作成と評価", 言語処理学会第2回年次大会, pp.425-428(1996)

[2] T.Kohonen: "The Self-Organizing Map", Proceedings of the IEEE, Vol.78, No.9, pp.1464-1480 (1990).

[3] 有田 英一, 安井 照昌, 津高 新一郎: "単語集合の自動構造化機能を持つ「情報散策」方式", 電子情報通信学会技術研究報告, 95-NLC-17 (1995).

[4] Zechner: "Fast Generation of Abstracts from General Domain Text Corpora by Extracting Relevant Sentences", COLING96 Vol 2, pp986-989

[5] 松本裕治, 伝 康晴, 宇津呂 武仁, 妙木 裕, 長尾 真: 日本語形態素解析システム JUMAN 使用説明書 (1994), 奈良先端科学技術大学院大学.

[6] 銭 晴, 史 欣, 田中 克己: "自己組織化マップと語彙索引を用いたデータベースの抽象化機構", 情報処理学会データベースシステム研究報告, 99-DB-22 (1994).

[7] 鳥脇 純一郎: "認識工学 - パターン認識とその応用 -", コロナ社

[5] SUPARC station 20, 160MB, SunOS4.1.3, で実行させた. ネットワークのアクセス時間は除外している.