自然言語処理 103－6
（1994．9．15）

共起行列の固有ベクトルを用いる単語クラスタリング法
～文書データベースの概要を表す単語クラスタの抽出～

津田 宏治, 仙田 修司, 美濃 導彦, 池田 克夫
京都大学 工学部
E-Mail: koji@kuis.kyoto-u.ac.jp

本論文では，類似度行列の固有ベクトルを用いたクラスタ抽出法について述べる．本手法の特徴は次の二つである．(1) 対象集合を排他的に分割せず，類似度の高い部分集合だけを抽出する．(2) 類似度行列の 0 でない要素の数を p とすると，本手法の計算量は p に比例する．従って，類似度行列が疎なほど高速にクラスタが求められる．「必要」，「程度」等，文書の内容を表さない「一般語」が文書中には多く存在するが，一般語はどの単語との類似度もあまり高くならないから，本手法は一般語を含まない単語クラスタを生成することができる．また，単語の類似度行列は一般に疎であるため，高速な処理が行える．

A Term Clustering Method Using Eigenvectors of Co-occurence Matrix

TSUDA Koji, SENDA Shuji, MINOH Michihiko, IKEDA Katsuo
Faculty of Engineering, Kyoto University
E-Mail: koji@kuis.kyoto-u.ac.jp

We propose a cluster extraction method using eigenvectors of similarity matrix. This method has two features. (1)It extracts several tight clusters. It leaves some objects unclustered. (2) The amount of computation is proportional to p, the number of non-zero elements of the similarity matrix. This method is suitable for term cluster extraction, since the similarity matrix of terms is sparse in most cases.

1 はじめに

クラスタとは，互いに類似した対象の集合である．対象間の類似性が類似度という値で表されているとき，対象集合をいくつかのクラスタに分割する操作をクラスタリングと呼ぶ．クラスタリングは，本来多変量解析の一手法であるが，近年では，機械学習 [1] や，パターン認識 [2] の分野でも盛んに応用が行われている．

本論文では，類似度行列の固有ベクトルを用いてクラスタを抽出する方法を提案する．本手法の特徴は次の二つである．

- ノイズをクラスタに含めない．

  ノイズとは，どの対象との類似度も大きくない対象を指す．一般のクラスタリング手法では，対象集合に含まれる全ての対象をいづれかのクラスタに分類するため，ノイズの多い対象集合を扱う場合，ノイズを多く含むクラスタを生成する．そのため，クラスタの密度が小さくなる（図 1(a)）．それに対し，本手法では，類似度が高い部分集合のみを抽出し，ノイズをクラスタに含めない（図 1(b)）．そのため，抽出されるクラスタの密度は，比較的大きくなる．

- 類似度行列の 0 でない要素の数を p とすると，計算量は p に比例する．

  類似度行列とは，対象が n 個あるとき，対象間の類似度を要素とする $n \times n$ 行列である．本手法は，類似度に 0 が多ければ多いほど高速にクラスタを抽出できる．

情報検索の分野では，文書は，その文書が含む単語の数を要素とするベクトルで表現される [3]．文書データベースに対する検索時には，質問に含まれる単語を多く含む文書をユーザに提示する．ここで，文書を単語を単位として表すのではなく，関連の深い単語を複数まとめた単語クラスタを単位として表せば，文書を表現するベクトルの次元数が小さくなる．また，質問中の単語を含まない文書でも，質問に関連のある単語を含むものは検索できるようになるので，検索の再現率が向上する．

上に挙げたクラスタ抽出法の特徴は，単語クラスタの抽出に非常に好都合である．その理由は次の二つである．

- 文書中の単語には，「一般語」が多い．

  文書中の単語には，ある特定の話題を扱う文書にのみ現れ，特定の事柄を端的に表す単語（「代表語」）と，話題に関わらず現れる単語（「一般語」）がある．例えば，固有名詞等は代表語であり，「必要」，「程度」などは，一般語である．文書の中には，代表語に比べて，一般語が非常に多い．一般語は，文書の特徴を表さない単語であるから，文書中に含まれるノイズであるとみなすことができる．一般に単語同士の類似度を測る指標として，二つの単語が同じ文書に現れた回数（共起頻度）を用いる．一般語は，多くの文書に少数ずつ分布しているので，どの単語との共起頻度もあまり高くならない．本クラスタ抽出法では，このような対象はクラスタに含まないので，一般語を含まないクラスタを抽出できる．

- 単語の類似度行列は，一般に疎である．

  単語同士の共起頻度をそのまま類似度として用いるばあい，類似度行列の非零要素率は，非常に低くなる (5%程度) ことが知られている [4]．本クラスタ抽出法では，類似度行列が疎である方が高速にクラスタを抽出できる．

本手法で抽出される単語クラスタは，関連のある代表語をまとめたものである．このため，この単語クラスタは，文書データベースの概要を非常に良く表す．従って，単語クラスタをユーザに提示することによって，ユーザに文書データベースにどのような文書があるかを知らせることができる．本論文では，単語クラスタを文書データベースの「ブラウジング」に用いることについても考察する．

以後，2 章では，このクラスタ抽出法について述べる．3 章では，本手法を用いて単語クラスタを抽出する利点について述べる．4 章では，実際の文書データベースに対して本手法を適用する．5 章は結びである．

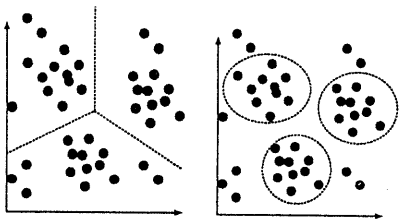
2 類似度行列の固有ベクトルを用いるクラスタ抽出法

2.1 クラスタ抽出法の概要

ここで提案するクラスタ抽出法とは，クラスタを対象集合から逐次的に抽出する操作である．対象集合の中から相互の類似度が大きい部分集合の抽出を行う．クラスタリングが全てのクラスタを一度に生成するのに対して，クラスタ抽出法では，クラスタを逐次的に一つずつ取り出していく．

クラスタリングは，事前にクラスタ数が決定しており，全ての対象を排他的に分類する用途 (ex. ベクトル量子化 [5] 等) に向いているのに対して，クラスタ抽出は，ノイズの多い対象集合から互いの類似度の高い部分集合を抽出する用途 (ex. 概念抽出 [1] 等) に向くと考えられる．

クラスタ抽出の対象は，パターンと非パターンに分類できる．対象がパターンであるか否かによって，ク



(a) クラスタリング　　(b) クラスタ抽出

図1　クラスタリングとクラスタ抽出の概念図

ラスタを求める方法は全く異る．対象が次の2条件を満たしているとき，その対象はパターンであるという．

- 対象が特徴ベクトルで表されている．
- 対象間の類似度が，特徴ベクトル間の距離である．

この場合，対象は特徴空間上の一点として表すことができ，K平均クラスタリング等の分割的クラスタリング手法 [6] を適用することができる．分割的手法の特徴は，ある目的関数を最小にする組み合わせ最適化問題として高速に解けることである．計算量は，対象数を n とすると，$O(n)$〜$O(n\log n)$ である [4]．

一方，上に挙げた2条件のどちらか一つでも満たさないとき，対象は非パターンであるという．例えば，文書は vector space model[3] においては，文書が含む単語の数を要素とする特徴ベクトルで表されるが，文書同士の類似度は，コサインなど，距離でない測度によって規定されるので，非パターンである（単語も同様）．非パターンには，分割的手法は適用できず，最大距離法，最小距離法などの階層的クラスタリングを適用する．計算量は $O(n^2)$〜$O(n^3)$ である [4]．

我々が提案するクラスタ抽出法では，非パターンを対象とする．さらに，クラスタ抽出を最適化問題に帰着させることによって，高速な処理を可能とする．本手法の計算量は，類似度行列の非零要素の数を p とすると，p に比例するので，階層的クラスタリングよりも高速である．従って，類似度行列の疎密が計算量に大きく影響する．本手法は，類似度行列が疎で，対象数の多い非パターンのクラスタ抽出に適している．

2.2　定式化

対象が n 個存在するとき，対象 $i,j(i \neq j, i = 1\cdots n, j = 1\cdots n)$ の間の類似度を e_{ij} とする．ここでは，$e_{ij} = e_{ji}, e_{ii} = 0$ とする．E を e_{ij} を第 i 列，第 j 行の成分とする $n \times n$ の対称行列とする．

本手法では，k 番目のクラスタ T_k を，このクラスタに対する対象 i の帰属度 $x_i^{(T_k)}(0 \leq x_i^{(T_k)} \leq 1)$ の n 次元ベクトル $\mathbf{x}^{(T_k)}$ で表現する．帰属度が大きいほど，対象が T_k に属する度合が大きいとする．

それぞれのクラスタについて，帰属度の2乗の総和（ベクトル $\mathbf{x}^{(T_k)}$ の長さ）は1であるとする．

$$|\mathbf{x}^{(T_k)}| = 1 \tag{1}$$

クラスタ T_k の密度 $Q^{(T_k)}$ を次のように定義する．

$$\begin{aligned} Q^{(T_k)} &= (\mathbf{x}^{(T_k)})^T E \mathbf{x}^{(T_k)} \\ &= \sum_{i=1}^{n}\sum_{j=1}^{n} e_{ij} x_i^{(T_k)} x_j^{(T_k)} \end{aligned} \tag{2}$$

また，クラスタ T_k, T_l の重なり $dup(T_k, T_l)$ を次のように定義する．

$$dup(T_k, T_l) = \mathbf{x}^{(T_k)}\mathbf{x}^{(T_l)} = \sum_{i=1}^{n} x_i^{(T_k)} x_i^{(T_l)} \tag{3}$$

本手法では，密度を最大化するように各対象の帰属度を定めることによってクラスタを求める．まず，第一クラスタ T_1 への帰属度 $\mathbf{x}^{(T_1)}$ は，次の最適化問題を解いて求める．

$$\begin{aligned} &\text{目的関数：} \quad Q^{(T_1)} \rightarrow \text{最大化} \\ &\text{制約条件：} \quad |\mathbf{x}^{(T_1)}| = 1, x_i^{(T_k)} \geq 0 \end{aligned} \tag{4}$$

$Q^{(T_1)}$ を大きくするためには，e_{ij} が大きい場合には $x_i x_j$ を大きくし，逆に e_{ij} が小さい場合には，$x_i x_j$ を小さくすればよい．従って，帰属度 $x_i^{(T_1)}$ の大きい対象間の類似度は比較的大きくなる．$x_i^{(T_1)}$ が大きい対象がクラスタを形成することは，このように直観的に推測できる．

第二クラスタ T_2 は，T_1 との重なりを μ 以下に保ちながら，$Q^{(T_2)}$ を最大化する解として求める．

$$\begin{aligned} &\text{目的関数：} \quad Q^{(T_2)} \rightarrow \text{最大化} \\ &\text{制約条件：} \quad |\mathbf{x}^{(T_2)}| = 1, x_i^{(T_2)} \geq 0 \\ &\qquad\qquad dup(T_1, T_2) \leq \mu \end{aligned} \tag{5}$$

μ は，重なりをどの程度許容するかを決めるパラメータである．同様にして，第 k クラスタ T_k は，$T_1 \cdots T_{k-1}$ との重なりを μ 以下に保ちながら，$Q^{(T_k)}$ を最大化する解として求める．

$$\begin{aligned} &\text{目的関数：} \quad Q^{(T_k)} \rightarrow \text{最大化} \\ &\text{制約条件：} \quad |\mathbf{x}^{(T_k)}| = 1, x_i^{(T_k)} \geq 0 \\ &\qquad\qquad dup(T_1, T_k) \leq \mu, \cdots, dup(T_{k-1}, T_k) \leq \mu \end{aligned} \tag{6}$$

上の最適化問題を，$Q^{(T_k)}$ がある閾値 t_q より小さくなるまで繰り返し解けば，密度が t_q 以上で，重なりが μ 以下のクラスタ集合を抽出することができる

2.3 類似度行列の固有ベクトルを用いる近似解法

クラスタを抽出するには，式6の最適化問題の大域的最適解が必要である．式6は，制約条件つきの非線形計画問題であり，大域的最適解を求めるのは，非常に困難である [7]．本章では，類似度行列の固有ベクトルを用いて式6の大域的最適解を近似的に求める方法について述べる．

Eの固有値を大きい順に$\lambda_1, \cdots, \lambda_n$とし，それに対応する固有ベクトルを$\mathbf{z}_1, \cdots \mathbf{z}_n$とする．固有値問題と最適化問題の関係について，以下のことが知られている [8]．

[定理1] 類似度行列の第k固有ベクトル$\mathbf{z}_k$は，次の最適値問題の大域的最適解として求められる．

$$\text{目的関数：} \quad \mathbf{z}_k^T E \mathbf{z}_k \rightarrow \text{最大化} \tag{7}$$

$$\text{制約条件：} \quad |\mathbf{z}_k| = 1$$

$$\mathbf{z}_1 \cdot \mathbf{z}_k = 0, \cdots, \mathbf{z}_{k-1} \cdot \mathbf{z}_k = 0 \tag{8}$$

式6と式7の最適化問題を比較すると，目的関数の部分は全く同じであり，非常に類似していることがわかる．式7は，式6の$x_i^{(T_k)} \geq 0$という制約を除き，$\mu = 0$と置いたものである．このような類似性から，簡単に計算できる固有ベクトルを用いて，式6の近似解を求める．具体的には，式7の最適解 (固有ベクトル) を，式6の制約条件を満たすように変形を加える．

類似度行列の固有ベクトル$\mathbf{z}_k$の要素には，正負両符号のものが混在している．これは，式6の制約条件に反するので，正に統一する．$\mathbf{z}_k$の正の要素の総和と，負の要素の総和の絶対値を比較し，値が大きい方の符号を，固有ベクトル$\mathbf{z}_k$の極性と呼ぶ．固有ベクトルの極性と同符号の要素の集合をI_{k1}，異符号の要素の集合をI_{k2}とする．ここでは，I_{k1}に属する対象のみがクラスタを構成しているとして，I_{k2}に属する対象の帰属度を0にする．また，同時に正規化も行う．クラスタT_k'に対する要素iの帰属度$x_k^{(T_k')}$は次のように定める．

$$x_i^{(T_k')} = \begin{cases} |z_{ki}| / \sqrt{\sum_{i \in I_{k1}} z_{ki}^2} & (i \in I_{k1}) \\ 0 & (i \in I_{k2}) \end{cases} \tag{9}$$

上のようにして得たクラスタ集合τ'の中には，重なりがμを越えるクラスタも存在するので，このままでは式6の近似解であるとはいえない．そこで，クラスタの集合τ'から，互いの重なりがμ以下であるクラスタの集合τを抽出する．ここでは，τ'中のクラスタを，密度の大きい順にτに加えてゆき，既にτにあるクラスタとの重なりがμ以上ならば捨てるという方法をとる．

1 $\tau \leftarrow \phi$
2 $T_i \leftarrow \tau'$に含まれるクラスタで最も密度が大きいもの
3 if $\max_{T_j \in \tau}\{dup(T_i, T_j)\} \leq \mu$ then $\tau \leftarrow \tau \cup \{T_i\}$
4 $\tau' \leftarrow \tau' - \{T_i\}$
5 if $\tau' = \phi$ then 終了 else (2) に戻る

このようにして得たクラスタ集合τは，式6の制約条件を満たす．これを，式6の近似解であると考える．

2.4 クラスタの曖昧さの除去

抽出されるクラスタT_kは曖昧さを持っているが，実際の用途には曖昧さのない対象集合C_kの形に変換することが必要となる．ここでは，対象集合C_kのクラスタT_kに対する充足率$suf(C_k, T_k)$を定義し，これを用いて対象集合への変換を行う．

$$suf(C_k, T_k) = \sum_{i \in C_k} (x_i^{(T_k)})^2 \tag{10}$$

帰属度$x_i^{(T_k)}$の大きい対象から順にC_kに加えていき，充足率$suf(C_k, T_k)$が充足率の閾値σを越えた時点で終了する．クラスタT_kを対象集合C_kに変換する手順を以下に示す．

1 $C_k \leftarrow \phi$
2 $i \leftarrow C_k$に含まれない対象で$x_i^{(T_k)}$が最大のもの
3 $C_k \leftarrow C_k \cup \{i\}$
4 if $suf(C_k, T_k) \geq \sigma$ then 終了 else (2) に戻る

2.5 クラスタ抽出の計算量

近似解法を用いる場合には，固有ベクトルから直接クラスタを導き出すため，クラスタ抽出にかかる計算量は，固有値問題を解くアルゴリズムに依存する．ここでは，Lanczos 法 [9] を用いて固有値を求めている．Lanczos 法は，大規模で疎な実数値対称行列の固有値問題を解く場合に最も効率の良い方法であり，次の二つの特徴を備えている．

1. 固有ベクトルを固有値の大きい方から個数を指定して求められる．
2. 行列が疎な場合，非常に高速．

本手法では，固有値の大きい固有ベクトルのみが必要であるから，(1) の性質は，非常に好ましいといえる．また，Lanczos 法の計算量は，行列が疎な場合には，全ての固有値を求めても$O(n^2)$であり [9]，少数しか求めない場合にはさらに減る．

Lanczos 法は Lanczos step と呼ばれる操作を繰り返して，収束的に固有ベクトルを求める．Lanczos step で行われる主な処理は，類似度行列と，n次元ベクトルの積の計算であるから，類似度行列の非零要素数をpとすると，1回の Lanczos step に必要な計算量は$O(p)$である．収束するまでに必要な Lanczos step 数は，行列の固有値の分布の仕方などで異るため，Lanczos 法の計算量を解析的に定めることはできない．

表 1　対象数に対する Lanczos step 数の変化 (vは求める固有ベクトルの数)

対象数	100	300	500	1000	2000	3000
v=1	10.3	8.0	7.0	7.0	6.0	6.0
3	17.6	24.0	36.3	34.3	40.6	49.0
5	20.6	31.3	33.3	41.6	57.6	58.6
7	24.0	37.6	40.3	48.0	67.6	74.0
9	24.6	40.0	48.3	58.0	73.3	83.3

対象数と求める固有ベクトルの数を変化させて，固有値を求めるまでに行われた Lanczos step の数を測定した (表1)．ここでは，類似度をランダムに設定した非零要素率 5%の類似度行列を用いた．また，クラスタ抽出では精度の高い解は必要でないため，精度は 4 桁とした．

対象数が増加するにつれて step 数が増加する傾向は認められるが，対象数の増加に対して step 数の増加は軽微である．例えば，5 個の固有ベクトルを求める場合に注目すると，対象数が 30 倍になっても，step 数の増加は 3 倍程度である．また，1 個の場合では，step 数は減少している．この事実から，固有ベクトル計算に必要な step 数は，現実的な対象数の範囲では，対象数にはほとんど依存しないと考えられる．従って，Lanczos 法に必要な計算量は p にほぼ比例するということができる．なお，文献 [10] においても，求める固有ベクトルの数が小さく，また，精度が低いときには，Lanczos step 数はほぼ一定であるという結論を出している．

本手法と，階層的クラスタリングとの実行時間の比較を行う．階層的クラスタリングを行うアルゴリズムには，様々なものがあるが，ここでは，stored matrix アルゴリズムを用いる．stored matrix は，最大距離法，群平均法など，すべての代表的な階層的クラスタリングを行うアルゴリズムであり，計算量は，$O(n^3)$ である．最小距離法に関しては SLINK という $O(n^2)$ の高速アルゴリズムが提案されているが，最小距離法の作るクラスタは，密度の低い，質の悪いものになることが知られている [4, 6] ため，ここでは比較の対象から除く．

Sun SS-10,50Mhz での実行時間を図 2に示す．ここでのクラスタ抽出数は 10 個であり，類似度行列には，非零要素率 10%のものを用いた．本手法は，数値計算を各ステップで行っているため，オーダーの差に比べると実行時間の差は少ないが，対象数が多くなるとクラスタ抽出法の方がかなり高速であることがわかる．

一般に，対象の数 n が多くなると，その間に定義される関係の数は ${}_nC_2$ より，かなり少なくなることが知られている．従って，このような対象の間に類似度を

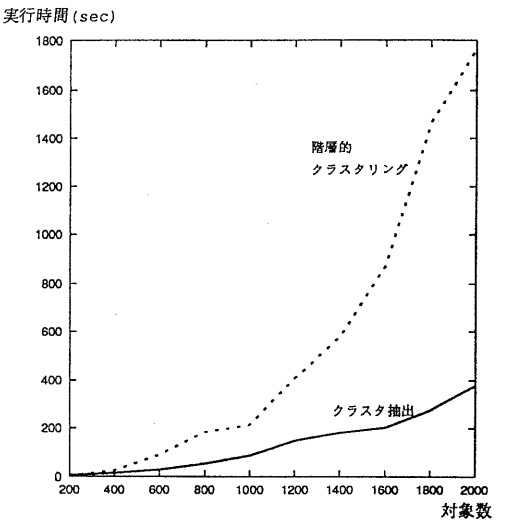


図 2　クラスタ抽出法と階層的クラスタリングとの実行時間の比較

設定すると類似度行列の非零要素率は低くなると考えられる．本手法は，類似度行列の非零要素率が低い場合に非常に高速であるので，計算量の点で階層的クラスタリング手法が適用できなかった大規模な問題に対する適用が期待できる．

2.6　クラスタ抽出例

二次元平面に配置した点を対象にクラスタ抽出を行った．ここでは，(0,0) を中心に半径 1 の範囲に 50 点，(2,0) を中心に半径 1 の範囲に 50 点，(0,2) を中心に半径 2 の範囲に 100 点を中心からの距離が正規分布を成すように配置した．

2 点 i, j 間の類似度 e_{ij} は，2 点間の距離を d_{ij} とすると，$e_{ij} = 1 - d_{ij} (d_{ij} < 1$ のとき$), 0 (d_{ij} \geq 1$ のとき) と設定した．第 1〜10 固有ベクトルからクラスタを抽出した後，$\mu = 0.1$ として選択を行い，3 個のクラスタを得た．クラスタの抽出結果を図 3に示す．図 3では，各クラスタを充足率$\sigma = 0.9$ で点の集合に変換し，□で表示している．ここでは，200 点のうち 78 点がクラスタとして抽出された．密度の高いクラスタの中心部だけが抽出され，周辺部が残されているのがわかる．

3　クラスタ抽出法の単語への適用

1 章でも述べたとおり，単語クラスタの主な用途は，文書の表現の次元数を下げることにある．ここでの単語のクラスタリングは，パターン認識でいうと「特徴抽出」に相当する．単語の中には，ある特定の話題を

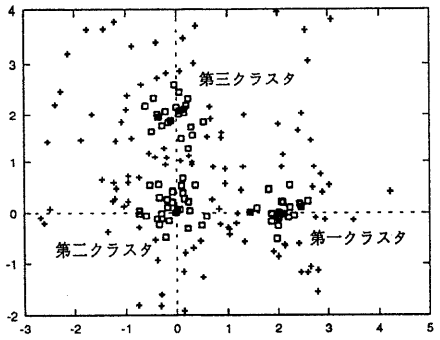


図 3　クラスタ抽出例

扱う文書に現れ，それを見ると特定の事柄を想起できる単語（代表語）と，話題に関わらず現れる，意味の抽象的な単語 (一般語) がある．特徴抽出という立場から，単語クラスタリングをとらえると，文書の特徴を良く表す代表語のみがクラスタに含まれ，一般語はノイズとして残されるのが望ましい．

統計的な観点から代表語と一般語を見てみると，代表語は，ある特定の話題を扱う文書にしか現れないため，少数の文書に集中して出現すると考えられる．一方，一般語は，多数の文書に分散して出現すると考えられる．本クラスタ抽出法は，図3に示したとおり，類似度が高い対象集合のみを抽出し，境界部分は残す性質がある．代表語は，ある特定の内容の文書にのみ集中的に現れるので，共通の事柄を代表する単語間の類似度（共起頻度）は非常に高くなり，クラスタを形成する．一方，一般語は，全ての文書に全体的に分布するため，どの単語との類似度もあまり高くならず，どのクラスタにも属さない境界部分を成す．従って，本手法を適用すると，一般語が除かれ，代表語のみからなるクラスタが抽出されると予想できる．

単語 i が代表語である度合を，単語の重要度 s_i と表す．ここでは，s_i として，情報検索において単語の重み付けに頻繁に用いられる $tf \times idf$ 測度 [11] を用いる．

$$s_i = \max_d TF(d,i) \times IDF(i) \tag{11}$$

$$IDF(i) = \log_2 \frac{D - a_i}{a_i} \tag{12}$$

この式で，$TF(d,i)$ は，文書 d に含まれる単語 i の個数である．また，a_i は i が含まれる文書の数，D は全文書数である．s_i は，a_i が少なく，また，$TF(d,i)$ の最大値が大きいほど大きな値となる．従って，少数の文書に集中して現れる単語の s_i は高くなる．4 章で行う実験では，抽出された単語クラスタ中の単語の s_i を評価する．

表 2　実験に用いる文書データベース

データベース名	Thesis
文書数	100
単語の種類	2329
単語総数	14473
文書毎の単語数	94〜234
文書の種類	卒論の内容梗概
カテゴリ数	21 (主観的な分類による)

単語クラスタの用途としては，先に述べた文書の表現の次元数を下げること以外に,「ブラウジング」が考えられる．文書データベースに含まれる文書中の共起頻度にもとづいて単語間の類似度を設定し，単語クラスタを抽出すれば，文書データベース中の代表語が数個ずつまとまった単語クラスタが得られる．この単語クラスタに含まれる語は互いに頻繁に共起する単語であるから，この単語クラスタを，データベースに含まれる文書のサンプルであると考えることができる．これをユーザに提示することによって，ユーザは，文書データベースにどのような文書があるのかをおおまかに理解することができる．現在，文書データベースのブラウジング方法としては，文書をハイパーリンクで結ぶなど，できるだけ多くの文書をユーザに見せることに力点が置かれているものが多い [12] が，多くの文書を見せるのはユーザにとって大きな負担となる．文書をそのまま提示するのではなく，その特徴を示す単語クラスタを提示することによって，ユーザのブラウジングにかける負担を減少させることができる．

4　実験と考察

4.1　実験に用いる文書データベース

実験に用いる文書データベースには，京大工学部情報工学科の過去の卒論，修論の内容梗概の第一ページを 100 部集めたものを用いる．この文書データベースを Thesis と呼ぶ．

文書からの単語の抽出は，辞書に含まれる全単語を，全文探索 [13] で文書中から抽出することによって行った．単語抽出用の辞書には，Wnn の名詞辞書，計算機用語辞書を用いた．抽出される単語数を制限するため，一文字の単語，及び，ひらがなのみから成る単語を除いた．さらに，この名詞辞書には，専門的な単語は含まれていないため,「ＣＡＳＥ」,「スレッド」など情報工学関連の単語 90 個を加え，総単語数 19653 語の辞書を作成した．Thesis の概要を表 2に示す．

Thesis に，どのような文書が含まれているのかを

明らかにするため，クラスタの抽出に先だって，100文書を内容によって分類し，21個のカテゴリを作成した．この分類は，情報処理学会の全国大会の各セッションに文書を割り振ることによって行った．

4.2 単語間の類似度の設定

クラスタ抽出を行うには単語間の距離を何らかの指標で表す必要がある．単語間の類似度は，単語が同じ文書に現れた回数，即ち，共起頻度であるとする．単語 $i,j(1 \leq i,j \leq n)$ の共起頻度を c_{ij}とする．文書数を Dとし，文書 $d(1 \leq d \leq D)$ に単語 i が，$n_i^{(d)}$個含まれているとすると，

$$c_{ij} = \sum_{d=1}^{D} n_i^{(d)} n_j^{(d)} \qquad (13)$$

と定義する．

4.3 単語クラスタによるブラウジング

文書データベース Thesis から抽出された単語クラスタを表3に示す．ここでは，第1〜第6クラスタにあたる6個のクラスタの帰属度上位8個の単語を示す．表中，左に人手で分類したカテゴリを示し (() 内は分類された文書数)，右に抽出したクラスタを示す．21個のカテゴリ中，この6個の単語クラスタに対応しているものは，12個であった．また，100文書中，単語クラスタに対応する12個のカテゴリに含まれているのは，66文書であった．この6個の単語クラスタによって，文書データベースの全ての概要を示しているとはいい難いが，右の単語クラスタを見れば，左のカテゴリの内容がほぼ想像できる．各カテゴリを代表する単語が，単語クラスタとしてまとめられているので，本手法が抽出する単語クラスタによって，文書データベースの概要を知ることはある程度可能であるといえる．

但し，この単語クラスタは，単語の意味を全く考慮せずに統計的な性質のみに基づいているため，不可解な単語クラスタを生成することも多く，また，表4中の「広ま」のように，無関係な単語が混入する場合も多い．このような不可解なクラスタを生成しないためには，単語の意味を考慮した類似度の設定法が必要となると考えられる．この点については今後の課題としたい．

4.4 ノイズに対する強靭さ

本節では，単語クラスタには，一般語が含まれにくいことを示す．一般語の重要度 s_iは，代表語に比べて低いと考えられるので，ここでは，単語クラスタに含まれる単語の重要度の高さを示す．本実験では，前節に示した第1〜第6までの6個の単語クラスタを用いて，次の三つの単語集合に含まれる単語の重要度 s_iの平均を比較する．

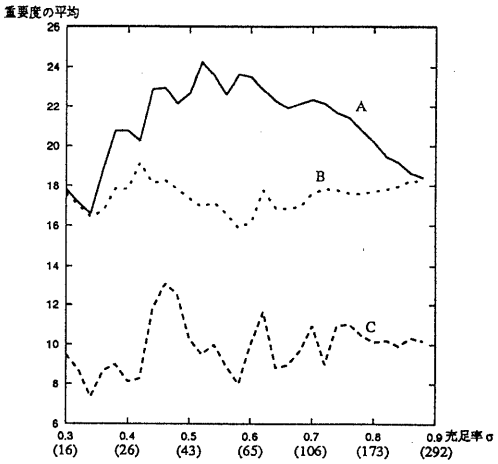


図4 抽出クラスタに含まれる単語の重要度 (充足率の下の数字は，各充足率での A が含む単語数 w)

- 単語集合 A: 抽出クラスタに含まれる全単語（個数を w個とする）
- 単語集合 B: 総出現数が多い順に w個の単語を集めたもの
- 単語集合 C: ランダムに w個の単語を集めたもの

単語集合 A に含まれる単語数 wは，クラスタを対象集合に変換する際の充足率σによって変化する．ここでは，充足率σを 0.3〜0.9 まで変化させて，重要度の平均を測定した．図4に，単語集合 A,B,C の重要度の平均の値を示す．全てのσに渡って Cより A の重要度が大きいことから，抽出クラスタには重要度の低い単語が含まれにくいことがわかる．また，Bより A の重要度の方が大きいことから，総出現数が多い単語であっても，重要度の小さい単語は抽出クラスタには含まれにくいことがわかる．

表4に，出現個数の多い17個の単語を示す．表中，クラスタとして抽出されなかった5個の単語には，下線が引かれている．また，式12の IDF の値も同時に示す．IDF の値が小さいほど多くの文書に現れていることを示している．「必要」，「内容」といった典型的な一般語が，出現個数が多いにも関わらず除かれていることが分かる．

表 3　抽出された単語クラスタ

主観的分類によるカテゴリ (文書数)	単語クラスタ
プロトコル (2)	(証明 認証 知識 ゼロ知識証明 相互 検証 広ま 通信)
画像処理 (2) 画像認識 (4) グラフィクス (2)	(処理 画像 並列 認識 研究 システム 情報 論理)
音声認識 (7)	(言語 単語 母音 記述 認識 候補 音声 識別)
ソフトウェア開発環境・手法 (8) ソフトウェア設計支援 (6) 仕様記述・プログラム合成 (5)	(ソフトウエア 管理 作業 開発 情報 仕様 ＣＡＳＥ 支援)
並列型プログラミング言語 (5) 並列マシン (18)	(並列 論理 実行 プログラム 言語 開発 回路 プロセス)
論理回路 (3) 論理検証・テスト (4)	(回路 関数 論理 設計 故障 検証 表現 決定 二分決定図)

表 4　Thesis に最も多く含まれる単語 (下線はクラスタとして抽出されなかった単語)

単語	処理	並列	画像	研究	システム
出現数	238	160	139	132	131
IDF	0.46	1.50	1.82	-1.15	0.23
情報	内容	認識	論理	文字	必要
125	107	105	103	102	89
1.36	-4.24	2.18	2.09	3.33	-0.17
モデル	梗概	計算機	言語	開発	対象
83	83	81	80	79	71
1.22	-2.28	0.89	1.91	0.89	0.83

5　おわりに

本論文では，類似度行列の固有ベクトルを用いてクラスタ抽出を行う方法について述べた．また，本手法を文書データベース中の単語に適用し，抽出される単語クラスタには一般語が含まれない傾向があること，及び，抽出クラスタによって文書データベースのブラウジングが可能であること，を示した．本手法が，類似度行列が疎であるほど高速であるという性質，また，一般語をクラスタに含めにくいという性質は単語のクラスタ抽出に適している．単語クラスタの用途として，ブラウジングを提案したが，今後は，情報検索等への応用をはかりたい．

参考文献

[1] G. Bisson: "Conceptual clustering in a first order logic representation", 10th Europian Conf. Artif. Intell. Proc., pp. 458–462 (1992).

[2] Z. Wu and R. Leahy: "An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation", IEEE Trans. on Pattern Anal. & Mach. Intell., **15**, 11, pp. 1101–1113 (1993).

[3] W. B. Frakes and R. Baeza-Yates Eds.: "Information Retrieval: Data Structures & Algorithms", Prentice Hall (1992).

[4] P. Willet: "Recent trends in hierarchic document clustering: A critical review", Information Processing & Management, **24**, 5, pp. 577–597 (1988).

[5] N. Ueda and R. Nakano: "A competitive & selective learning method for designing vector quantizers", 1993 IEEE Int. Conf. Neural Netw., pp. 1444–1449 (1993).

[6] A. K. Jain and R. C. Dubes: "Algotithms for Clustering Data", Prentice Hall (1988).

[7] 茨木, 福島：“最適化の方法”, 共立出版 (1993).

[8] 志賀：“固有値問題 30 講”, 朝倉書店 (1991).

[9] J. Cullum and R. A. Willoughby: "A survey of lanczos procedures for very large real 'symmetric' eigenvalue problems", J. Comput. Appl. Math., **12–13**, pp. 37–60 (1985).

[10] A. Pothen, H. D. Simon and K.-P. Liou: "Partitioning sparse matrices with eigenvectors of graphs", SIAM J. Matrix Anal. Appl., **11**, 3, pp. 430–452 (1990).

[11] 徳長, 岩山：“重みつき idf を用いた文書の自動分類について”, 情処研報, **NL100-5**, pp. 33–40 (1994).

[12] R. H. Thompson and W. B. Croft: "Support for browsing in an intelligent text retrieval system", Int.J.Man-Machine Studies, **30**, pp. 639–668 (1989).

[13] S. Senda, M. Minoh and K. Ikeda: "Fast string searching in a character lattice", IEICE Trans. Information and Systems, **E77-D**, 7, pp. 846–851 (1994).