ベクトル空間モデルに基づく次元削減による大規模文書データの検索と可視化

青野 雅樹，小林メイ

日本アイ･ビー･エム東京基礎研究所

近年ブロードバンドのインターネットの普及に伴い，巨大なデータの伝送や処理がネットワークを介して行うことが可能となってきた。同時に，横溢する巨大データに対する知的な処理（マイニング）の重要性が増してきた。

本報告では，ベクトル空間モデルでモデル化された大規模文書データの次元削減手法による，情報検索，クラスタリング，および可視化について述べる。コンテンツ解析や理解を助けるために開発した，自動推奨する3つの軸（基底ベクトル）に投影してデータ表示したり，この3次元空間での回転・拡大縮小，平行移動といったアフィン変換操作が可能な Prosciutto と呼ばれるシステムに関しても言及する。

Information Retrieval and Visualization of Massive Database using Dimensional Reduction based on Vector Space Model

Masaki Aono and Mei Kobayashi
aono@jp.ibm.com, mei@jp.ibm.com,
IBM Tokyo Research Laboratory

We present a novel system, Prosciutto, *for IR (Information Retrieval) and visualization of the contents of massive databases. The system has several notable features. One of the most useful is a similarity search based on vector space modeling. Another is a service to recommend three mutually perpendicular subspace coordinate axes in attribute space onto which document vectors can be projected and displayed for view to help users understand relationships between a query and database documents.*

1. はじめに

近年コンピュータシステムのディスク容量の増加やブロードバンドのインターネットの普及で大量のデータのやりとりが可能になってきた。これに伴い，ヘテロな文書データベースに対する情報処理のニーズが高まっている[33]。文書データは基本的に，作成する人によってフォーマットもまちまちなので，これを統一的にコンピュータシステムで理解させるのは容易なことではない。しかしながら，このように横溢する文書データの中には，きわめて貴重なものも潜伏している可能性がある。たとえば，カスタマーサービス用のコールセンターには，カスタマーからの要求やクレームなどが，毎日大量に蓄積されている。クレジットカードや保険会社には，カスタマーのデータが蓄積され，将来，新しいサービスを提供する場合の，優良カスタマー情報が存在している。

データマイニングの分野では，このような情報から，有用なデータを検索・抽出できる技術が数多く発表されている。われわれの以下で述べる技術は，このようなデータマイニング技術とは，多少主旨が異なる。まず，大抵のデータマイニングにおけるデータは，数値データのみを対象とする。これに対して，われわれはベクトル空間モデルでモデル化できるものであれば，数値データに限らず，画像データや音声データなども対象とすることができる。また，本論文では，大規模データに対する情報検索とマイニング技術（とくにクラスタリング）を組み合わせ，これにグラフィカルなユーザインターフェイスをつけたシステム構成をとり，データをシステムがリコメンド（推奨）する軸を選んでプロットできるという点が，これまでの研究との大きな違いである。

以下の節の構成は，次のようである。まず，2節で多次元データの可視化と情報検索に関する従来技術をサーベイする。3節では，われわれが用いているベクトル空間モデルと，多次元データの圧縮技法のひとつと位置づけされる次元削減手法を述べる。4節では，われわれが開発した Prosciutto(「プロシュート」と発音する)システムに関して述べる。Prosciutto は，大規模データのコンテンツの解析や理解を目的とした GUI を提供するものである。

2. 多次元データの可視化と情報検索

多次元データの可視化は，情報検索システムの扱うデータベースが大規模になるにつれ，また非科学技術計算を行うユーザ数が増えるにつれて，ますます重要なトピックになってきた[6,32,34]。情報検索のインターフェイスとして，これまで幾つかの表現構造（たとえば，チャート図，木構造，アノテーションなど）が提案されている[3,4,9,14,16,19]。ここでは，我々の開発した Prosciutto システムに関連するデータマイニングおよび情報検索のための可視化技術をレビューする。

比較的規模の大きい多変量データを解析するために，適切な低次元あるいはそのスライスを見出してデータを表示するアイデアは 1960 年代に Kruskal [13]によって初めて提案された。最初の成功した実装として Friedman と Tukey [7] による「射影追跡」が有名である。彼らの目的は，多変量データの興味ある線形写像を自動的に見出して，線や平面にマッピングするというものである。「射影追跡」の概念を拡張して実装したものでは，Nason [17] による多次元データから情報量の多い 3 次元スライスを発見し，可視化する研究がある。

Prosciutto システムは，多次元データ（ここでは，文書-属性空間）から低次元（2次元または3次元）を切り出すという点で，「射影追跡」に似ている。我々のシステムでは，ユーザが興味のあるビューに近い座標軸を，システムが自発的にリコメンド（推奨）してくれる。

データベースのコンテンツを可視化するために属性空間の次元を減らす別のアプローチとしては，互いに直交していない幾つかの座標軸を用いるものがある。Inselberg [10] の開発した parallel coordinates 手法はその一例である。また，円弧に星状のプロットを行う手法も一例である。たとえば，Ankerst ら[1]は多色での円弧状のデータ表現を提案した。また，Kandogan [11] の「スター座標系システム」は，このような円弧状のプロットに焦点をあてたクラスター解析用の代表例である。上述の手法はいずれも直交座標系を用いない点で我々の Prosciutto システムと異なる。また，Prosciutto システムで提供しているような座標軸のリコメンド機能はない。

3．ベクトル空間モデルと情報検索

Salton ら[18] が約 30 年前に提案して以来，ベクトル空間モデルは，情報検索におけるデータベースの代表的なモデルとして確立されてきた。このモデルの利点のひとつは，ヘテロなフォーマットの文書の相関性のランキングが可能な点である。

ベクトル空間モデルに基づく情報検索システムでは，各文書はベクトルとしてモデル化される。このうち，*Boolean* モデルでは，各ベクトルの要素は0か1の値をとる。また，*Term weighting* モデルでは，属性（たとえばキーワードやキーフレーズ）の出現頻度を考慮するモデルであり，各ベクトルの要素は，0か正の実数値となる。

Prosciutto では，*Term Weighting* モデルの一種である TF-IDF (Term Frequency Inverse Document Frequency) モデル[15]を用いている。TF-IDF モデルでは，i 番目の文書の j 番目のキーワードの重み *weight(i,j)*は，キーワード頻度を $tf_{i,j}$ とし，各キーワードが出現する文書数を *dfj*としたとき以下の式で定義される。

$$\mathrm{weight}(i,j)=\begin{cases}(1+tf_{i,j})\log_2\dfrac{N}{df_j}, & \text{if} \quad tf_{i,j}\geq 1\\ 0 \quad , & \text{if} \quad tf_{i,j}=0,\end{cases}$$

ベクトル空間モデルの情報検索では，各クエリも，文書と同様に同じ属性空間からなるベクトルでモデル化される。検索結果の類似度ランク付けは，与えられたクエリとの「距離」に基づいて行われる。もっともよく用いられる「距離」としては，クエリベクトルと文書ベクトルとのなす角度で与えるものが頻繁に利用される [13]。

3.1．ベクトル空間モデルの次元削減

データベースが大規模になると，通常のベクトル空間モデルでの情報検索における類似度ランキングの計算量は膨大になり，実時間でのレスポンスが得られなくなる。大規模データベースにおける類似度ランキングのスケーラビリティの向上は，検索エンジンを利用するユーザにとっても，きわめて重大な問題である[8]。

この問題を解決するひとつのアプローチは，数学モデルの次元を削減することである。 すなわち，$M\times N$ の文書-属性行列 $\mathbf{A}$ が与えられたとき，この要素 $a(i,j)$ が i 番目の文書に対して，j 番目の属性に対する重み *weight(i,j)* を表すとする。ただし M は文書数を表し，N は属性数を表す。本論文で言う「大規模」データとは，M が最低でも 10 万以上で属性数 N がだいたい 5000 から 2 万程度の場合である。数学モデルの次元の削減とは，後者の属性数 N をそれより十分に小さい次元 k $(k \ll N)$ に削減することである。通常 k は N の 1/100 程度にする。こうすることで，情報検索における類似度ランキングをリアルタイムに計算しようというものである。

3.2．2 つの次元削減アルゴリズム

前節で述べた，ベクトル空間モデルでの属性次元を削減する手法として以下の2つが知られている。

- 潜在的意味解析法(LSI 法)

● 共分散行列法(COV 法)

前者の潜在的意味解析法，すなわち LSI 法とは Latent Semantic Indexing の略である。一方，後者の COV 法とは Covariance matrix method の最初の単語の先頭3文字とった略である。

LSI 法の基本的なアイデアは，行列 $\mathbf{A}$ を $\mathbf{A}=\mathbf{U}\Sigma\mathbf{V}^{\mathbf{T}}$ のように，直交行列 $\mathbf{U}$ と $\mathbf{V}^{\mathbf{T}}$ および特異値の大きい順に並べた対角行列 Σ に特異値分解したとき，対角行列の要素の大きいほうから k 個の特異値 $(\sigma_1,...,\sigma_k)$ だけを選択し，これに対応する左特異ベクトル $\mathbf{U}$ の k 列と，右特異ベクトル $\mathbf{V}^{\mathbf{T}}$ の k 個の列をもとに生成できるランク k の行列 $\mathbf{A_k}$ で，$\mathbf{A}$ を近似しようというものである[6]。この行列 $\mathbf{A_k}$ はランク k の行列の中で Frobenius ノルムの意味で，$\mathbf{A}$ をもっともよく近似するという性質がある[2]。LSI 法は，特異値分解定理のこのような性質を利用した点で，ベクトル空間モデルの情報検索への道を切り開いたが，2つのボトルネックがある。ひとつは，データベースが大規模になったときに，特異値分解に膨大な時間がかかることである。もうひとつは，$\mathbf{U}$ は $\mathbf{A}$ と同じ行数(文書数) M をもつ長方形の行列であるので，メモリ空間も大量に必要とし，スケーラビリティに問題があることである。

COV 法では，$M\times N$ の文書-属性行列 $\mathbf{A}$ が与えられたとき，以下の式で共分散行列 $\mathbf{C}$ を定義する。

$$\mathbf{C}=\frac{1}{M}\sum_{i=1}^{M}\mathbf{d_i}\,\mathbf{d_i^t}-\bar{\mathbf{d}}\,\bar{\mathbf{d}}^{\mathbf{t}},$$

ここで，$\mathbf{d_i}$ は i 番目の文書ベクトルを表し，$\bar{\mathbf{d}}$ は，全文書ベクトルの平均ベクトルを表す[12]。すなわち，

$$\bar{\mathbf{d}}=\left[\bar{d}_1\ \cdot\ \cdot\ \cdot\ \bar{d}_N\right]^{\mathbf{t}};\ \mathbf{d_i}=\left[a_{i,1}\ \cdot\ \cdot\ \cdot\ a_{i,N}\right]^{\mathbf{t}},$$

ただし，$\bar{d}_j=\frac{1}{M}\sum_{i=1}^{M}a_{i,j}$ である。

一旦，共分散行列 $\mathbf{C}$ が構築できると，これは属性空間×属性空間次元の正方対称行列となるので，$\mathbf{C}=\mathbf{V}\Lambda\mathbf{V}^{\mathbf{t}}$ のように，固有値分解をおこなう。ここで $\mathbf{V}$ は，正方直交行列である。また，対角行列 Λ は，実数の固有値が大きい順に並んだものである。COV 法では，LSI 法と同様に，大きいほうから k 個の固有値 $(\lambda_1,...,\lambda_k)$ を Λ から，それに対応する k 個の固有ベクトルを $\mathbf{V}$ からとってきた行列 $\mathbf{C_k}$ で，$\mathbf{C}$ を近似する。COV 法は多次元データに対する主成分分析法の一種であるが，LSI 法でのボトルネックであった，特異値計算の計算時間が $O(MN^2)$ から対称行列の固有値計算に必要な時間 $O(N^3)$ に，また記憶領域も $O(MN)$ から $O(N^2)$ で済み，文書数 M に依存しないため，スケーラビリティの問題に対応できる。

3.3. 次元削減空間での情報検索

LSI 法または COV 法により次元削減された文書-属性空間では，次元削減された文書ベクトル $\hat{\mathbf{d}}_\mathbf{i}$ は，k 個の右特異値ベクトル(あるいは固有ベクトル)を用いて

$$\hat{\mathbf{d}}_\mathbf{i}=[\mathbf{v_1},...,\mathbf{v_k}]^{\mathbf{t}}\,\mathbf{d_i}=\mathbf{V_k^t}\,\mathbf{d_i}$$

あるいは，Σ_k^{-1} (または Λ_k^{-1})を用いて

$$\hat{\mathbf{d}}_\mathbf{i}=\Sigma_\mathbf{k}^{-1}\,[\mathbf{v_1},...,\mathbf{v_k}]^{t}\,\mathbf{d_i}=\Sigma_\mathbf{k}^{-1}\,\mathbf{V_k^t}\,\mathbf{d_i}$$

と表現できる。情報検索におけるクエリベクトル $\mathbf{q}$ は，文書ベクトルと同様に，

$$\hat{\mathbf{q}}=[\mathbf{v_1},...,\mathbf{v_k}]^{\mathbf{t}}\,\mathbf{q}=\mathbf{V_k^t}\,\mathbf{q}$$

と表現できる。もっとも，よく用いられる類似度計算として，2つの k 次元ベクトル間の内積を用いると，

$$\text{similarity}(\hat{\mathbf{q}},\hat{\mathbf{d}}_\mathbf{i})=\hat{\mathbf{q}}\bullet\hat{\mathbf{d}}_\mathbf{i}$$

と表現できる。

LSI 法と COV 法の情報検索における質的な違いを表したのが図1である。

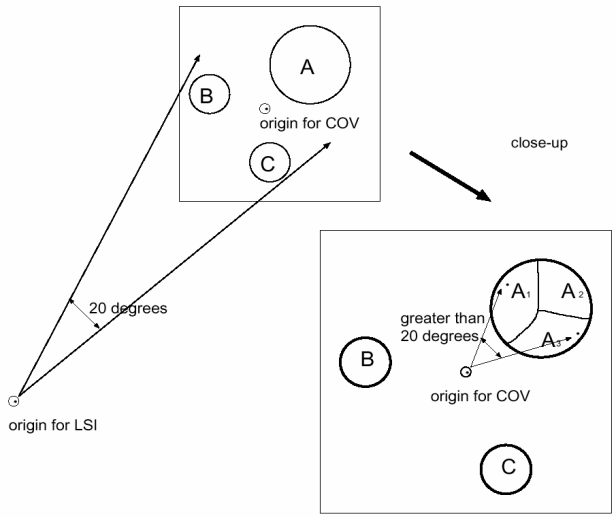

図1. LSI 法と COV 法による情報検索問題の文書-属性空間へのマッピングのイラスト。LSI 法は原点をシフトしないが，COV 法では，文書ベクトル全体の平均の位置に原点をシフトする。このため，原点中心に文書が等方向に分布する傾向があり，情報検索の精度は，一般に COV 法の方がよい。この図では，クエリベクトルが2次元平面上でだいたい(1,1)方向を向いており，そのうち角度 20 度の文書を類似文書として検索する様子を表している。LSI 法では，3つのクラスターA,B,C とも，この検索範囲に入ってしまうが，COV 法では，クラスターA だけをとらえている。これが検索の精度に反映される。

4. Prosciutto システム

Prosciutto は LSI 法と COV 法とを兼ね備えた大規模データに対する検索結果の可視化およびクラスタリングを理解するための支援システム用の GUI である。本節では，実際のニュースデータベースをもとに実装したシステムのコンポーネントや実行結果などを述べる。LSI 法と COV 法をシステムコンポーネントとして持つ検索可視化システム全体の構成は，図2に示すようである。

ユーザは，まずクエリを入力する。次に， COV 法か LSI 法かを選択する。これは，前処理でどちらの手法で文書データ(ここでは Los Angeles Times ニュースデータ)を次元削減したかによって選択する。

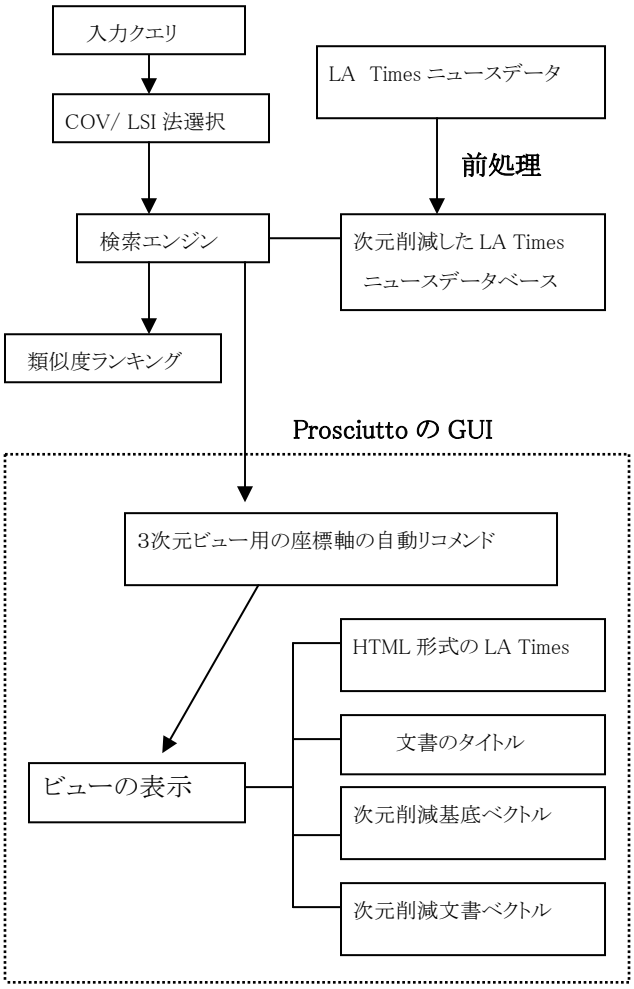

図2. 検索エンジン，次元削減(前処理)と Prosciutto システムの GUI 部分の概略図。

この後，検索エンジンにより，クエリに対する類似度ランキングを計算する。この結果は通常の検索エンジンのように，ランキング順の(表)形式で表示すると同時に，Prosciutto システムの GUI で，k 次元空間から，ユーザのクエリに対する類似度ランキングの上位のデータをもっともよく捉える3つの座標軸を自動的にリコメンド(推奨)して，その3次元空間に投影して表示する。

ユーザは，システムのリコメンドする座標系でデータをブラウズすることも出来るし，3つの次元をマニュアルで選択することも出来る。ここで，注意すべきは，もともとの文書データの数が非常に大きいので，文書データをすべて，選ばれた3次元空間に投影表示すると，この後のアフィン変換に代表される3次元空間のナビゲーションのストレスが大きくなってしまう。そこで，Prosciutto では，クエリに対して，関連性のある上位数百程度の文書データだけを絞り込んで表示するようにしている。この数も，ユーザが制御することが出来る。

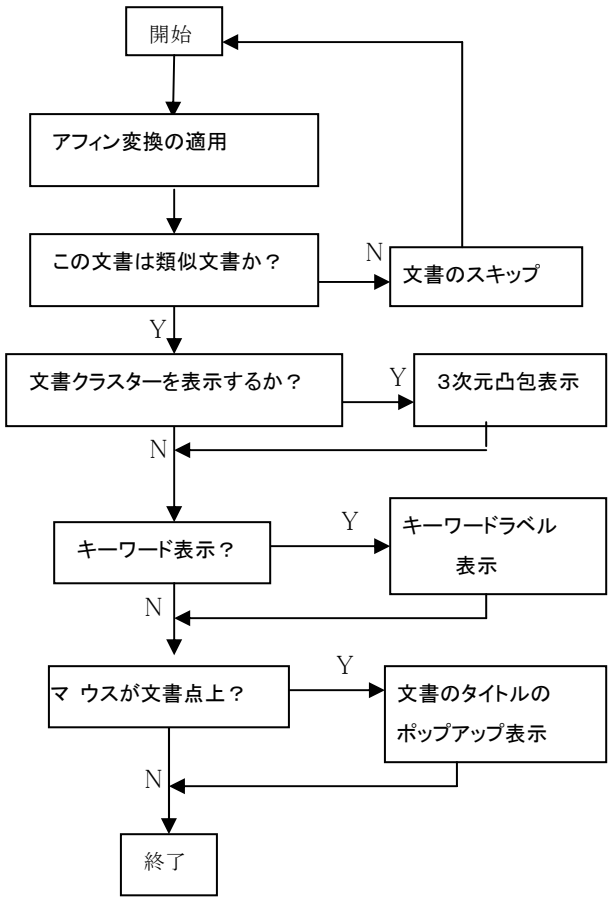

図3. Prosciutto の GUI 処理のフローチャート。これを各文書に対して適用する。

図3は Prosciutto の GUI 部分の流れ図である。また，図4は，Prosciutto システムのスクリーンダンプである。ここでは，入力として，“baseball 3 game 1 basketball 2”という文字列をクエリとして入力している。ここで，“3”とか“1”は，それぞれのキーワードの重みを表す。このキーワードと重みのペアから，TF-IDF モデルにより，単語の頻度情報を加味したクエリベクトルが生成される。

図4の左側のパネルには，このクエリに対する類似度の高い文書が表形式でリストアップされており，一方，右側のパネルでは，システムがリコメンドした3つの座標軸（ここでは，2次元，109次元，45次元）が選択され，関連度の高い数百の文書がこの空間に射影されて表示されている。文書は，色が赤いほど，また，文書点のサイズが大きいほど，より入力クエリとの類似度が高いことをあらわしている。

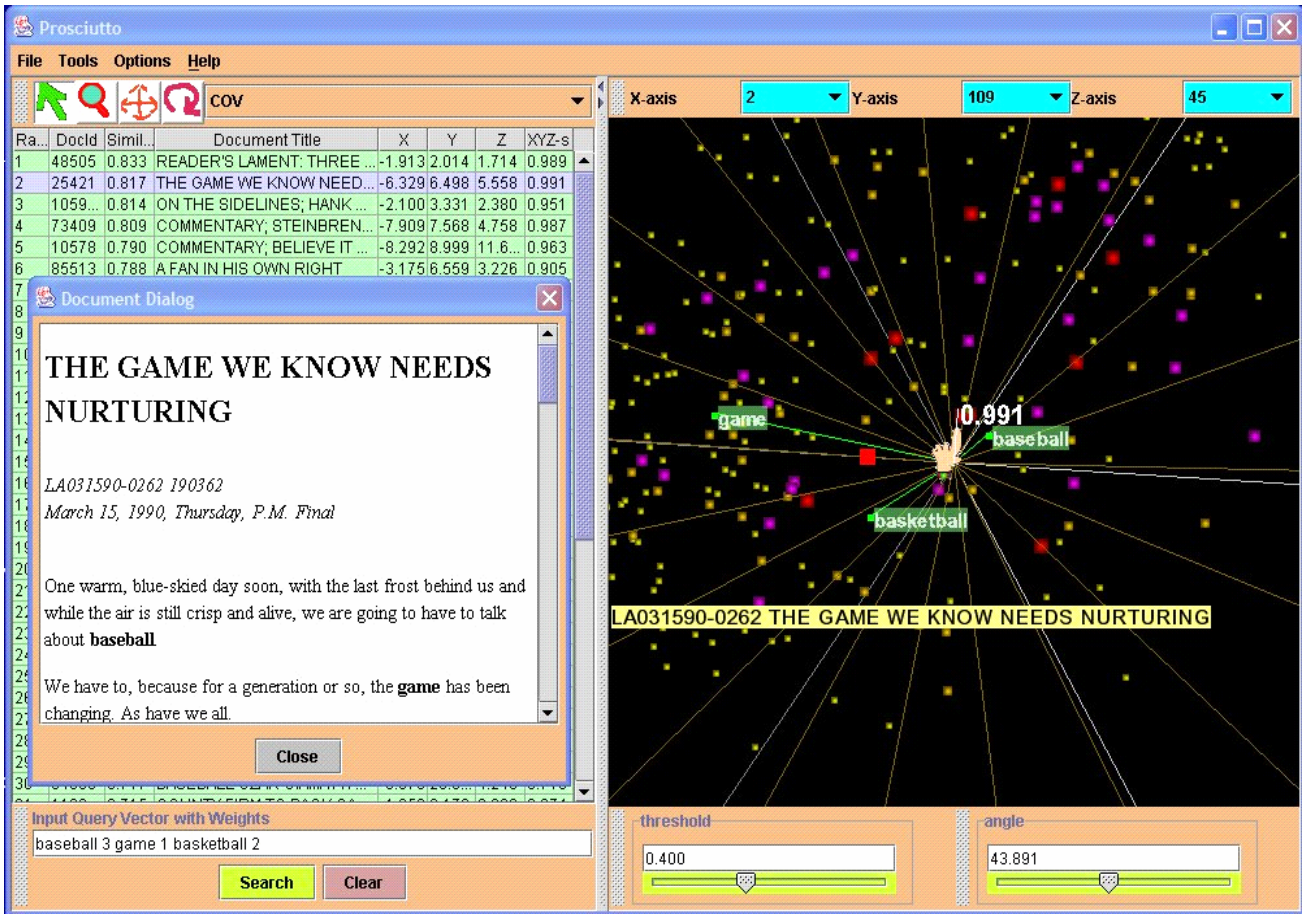

図4. Prosciutto システムの情報検索のための GUI を示すスクリーンダンプ。このスクリーンダンプでは，マウスが丁度，類似度ランキング2番目の文書（タイトルが”THE GAME WE KNOW NEEDS NURTURING”）の上にあり，「指」を表すアイコンが，その文書位置を表している。“baseball”，“game”，“basketball”という各キーワードも，この選択された3次元空間に射影されたベクトル方向に表示されている。

ここで用いているデータは，図3に示すように Los Angeles Times のニュースデータで，約13万件のニュースデータからなる。この中から，前処理でキーワードを約1万抽出し，これを200次元に COV 法により次元削減した文書データをシステムに読み込ませている。

図4の右側のパネルに示している3次元スライス上で，マウスが適当な文書上にくると，その文書に対応するニュース記事のタイトルがポップアップする。この状態で，マウスの右ボタンをクリックすると，さらにその文書の中身にアクセスでき，左側のパネル上に表示されているような，文書の中身を見ることが出来るようになっている。ここで，入力クエリにあった，“baseball”や“game”は，太字で表示されるようになっている。

5. まとめ

本論文では，ベクトル空間モデルを用いた情報検索の手法と，それを実装した Prosciutto システムを紹介した。Prosciutto システムの特長をまとめると，以下のようである。

- ベクトル空間モデルに基づく，スケーラブルな情報検索システム
- あるクエリに対する，もっとも関連度の高い3つの座標軸を自動的に選択し，表示する機能を有すること
- リコメンドする次元の選択と，ユーザによるマニュ

アルでの3軸を決定できること

- 大量の文書データであっても，ユーザクエリに関連する文書だけを絞り込んで表示することによるGUI でのナビゲーション・ストレスの低減
- クエリに対してリコメンドされた次元で表示された文書のタイトルや中身にアクセスできるポップアップ情報提示機能
- オプションとして，入力された各キーワードに対する3次元凸包を表示する機能

今後の研究テーマとしては，Prosciutto システムを拡張して，3つの座標軸より多くの軸を用いて，人間の認知能力を利用したより効果的な方法で，大規模データを表現できるかどうかの研究があげられる。また，3軸への射影であっても，文書データをよりわかりやすくレンダリングする手法や表現形式も今後の課題である。

謝辞

本研究にあたって，Michael Houle 氏，寒川光氏，野美山浩氏，武田浩一氏に大変有益なアドバイスをいただきました。また，IBM ワトソン研究所の Eric Brown 氏には，TREC データを提供していただきました。ここに感謝の意を表します。

参考文献

[1] M. Ankerst, D. Keim , H.-P. Kriegel, “Circle Segments: a technique for visually exploring large multidimensional data sets”, *Proc. IEEE Visualization*, (Hot Topics Session), 1996.

[2] M. Berry, S. Dumais and G. O’Brien, “Using linear algebra for intelligent information retrieval”, *SIAM Review*, vol. 37, no.4, pp. 571-595, Dec. 1995.

[3] S. Card, et al.(eds.), *Readings in Information Visualization*, Morgan Kaufmann, CA, 1999.

[4] J. Cugini, S. Laskowski and C. Piato, Document clustering in concept space: the NIST information retrieval visualization engine (NIRVE), *manuscript*, NIST, 2002.

[5] J. Cugini, S. Laskowski and M. Sebrechts, “Design of 3-D visualization of search results”, *manuscript*, NIST: *www.itl.nist.gov/iaui/vvrg/cugini/uicd/nirve-paper.html*

[6] S. Deerwester et al., “Indexing by latent semantic analysis”, *J. American Soc. Info. Science,* vol. 41, no. 6, pp. 391-407, 1990.

[7] J. Friedman and J. Tukey, “A projection pursuit algorithm for exploratory data analysis”, *IEEE Trans. on Computers*, vol. c-23, no. 9, pp. 881-890, 1974.

[8] Graphics, Visualization, and Usability Center of Georgia Institute of Technology (GVU), *GVU’s (Web) users’ survey:* www.gvu.gaetch.edu/user_surveys

[9] M. Hearst, “User interfaces and visualization”, Chapter 10 in ref [9].

[10] A. Inselberg, “Parallel coordinates: a guide for the perplexed”, *Proc. of IEEE Visualization*, pp. 35-38, 1996.

[11] E. Kandogan, “ Visualizing multi-dimensional clusters, trends, and outliers using Star Coordinates”, *Proc. KDD*, San Francisco, CA, pp. 107-116, 2001.

[12] M. Kobayashi, L. Malassis and H. Samukawa, “Retrieval and ranking of documents from a database”, *patent*, filed, IBM Corp., June 2000.

[13] J. Kruskal, “Toward a practical method which helps uncover the structure of a set of multivariate observations by finding the linear transformation which optimizes a new `index of condensation”, in R. Milton and J. Nelder (eds.), *Statistical Computation* , Academic Press, NY, 1969.

[14] M. Lesk, *Practical Libraries*, Morgan Kaufmann, San Francisco, CA, 1997.

[15] C. Manning and H. Schütze, *Foundations of Statistical Natural Language Processing*, MIT Press, Cambridge, MA, 2000.

[16] M. Maybury, W. Wahlster (eds.), *Readings in Intelligent User Interfaces*, Morgan Kaufmann, San Francisco, CA, 1998.

[17] G. Nason, *Design and Choice of Projection Indicies,* Ph.D. Thesis, Univ. of Bath, UK, 1992.

[18] G. Salton (ed.), *The Smart Retrieval System,* Prentice-Hall, Englewood Cliffs, NJ, 1971.

[19] R. Spence, *Information Visualization*, Addison-Wesley, NY, 2000.

[20] C. Ware, *Information Visualization*, Morgan Kaufmann, San Francisco, CA 2000.

[21] I. Witten, A. Moffat, and T. Bell, *Managing Gigabytes,* second edition, Morgan Kaufmann, San Francisco, CA, 1999.

[22] P. Wong, “Visual data mining”, *IEEE CG & A,* pp. 20-21, Sept./Oct. 1999.