データベースシステム 109－1
（1996. 7. 24）

意味の数学モデルによる意味的連想検索の高速化アルゴリズム

宮原 隆行† 清木 康†† 北川 高嗣†

†筑波大学 電子・情報工学系 ††慶應義塾大学 環境情報学部

データベースを対象とした情報検索における基本操作は，データ間のパターン・マッチングによる連想検索である．本稿では，意味的な等価性，類似性に関する計算により情報検索を行う意味的連想検索方式の高速化アルゴリズムを示す．本アルゴリズムは，キーワード，および，それを説明する文脈語列を受けとることによって意味的に関連する情報を抽出するために用いられる．

このアルゴリズムの実現可能性，および，有効性を明らかにするために，基本英単語を対象とした意味的連想検索の実験を行った．実験結果により，提案アルゴリズムが意味的連想検索の高速化の実現に有効であることを明らかにする．

A Fast Algorithm for Semantic Associative Search by a Mathematical Model of Meaning

Takayuki Miyahara†, Yasushi Kiyoki†† and Takashi Kitagawa†

†Institute of Information Sciences and Electronics, University of Tsukuba
††Faculty of Environmental Information, Keio University

The basic operation for extracting information from databases is associative search based on the pattern matching between data items. In this paper, we present a fast algorithm of semantic associative search which realizes information extraction by the computation on semantic equivalence and similarity. This algorithm is used to extract semantically related information by receiving a keyword and a sequence of the context data items which explains the keyword.

We have perfromed several experiments in which the fast algorithm is applied to semantic associative search for the basic English word retrieval. Those experimental results show that the fast algorithm is effectively used to extract the potential ability of the semantic associative search method.

1 はじめに

データベース・システムにおける情報探索のための主要な基本操作は連想検索である．現行のデータベース・システムにおける連想検索は，パターン・マッチングによる検索であり，異なる表現形態であるが同一の意味をもつデータや近い意味をもつデータの検索を行うことはできない [5, 6, 10]．また，同一のデータがもつ多義性を取り扱うことはできない．データ間の意味的な関係の扱いについては，データ間の関係を静的かつ明示的に記述し，同一性，相異性を判定する方法が広く用いられてきた [1, 8]．しかし，その判定は，静的に与えられた関係を用いて，曖昧性を含んで行われる．例えば，シソーラスを用いて同義語を照会する方法があるが，その同義語は，シソーラスの設計時に静的に決定され，また，同義であることの定義には曖昧性を含んでいる．

我々は，データ間の意味的な同一性，相異性は，

静的な関係によって決定されるのではなく，文脈や状況に応じて動的に変化するものであり，その動的な要素を含んで決定しなければ，データ間の関係の曖昧性を排除することはできないものと考える．このような単語間の意味的な関係を文脈に応じて動的に計算するモデルとして，意味の数学モデルが提案されている [2, 3].

意味の数学モデルは，ほぼ無限通りの文脈や状況に応じた意味的な関係を動的に計算することを目的としたモデルである．このモデルに基づいた連想検索により，検索キーワードに意味的に近い検索対象語を連想検索することが可能となる．検索対象となるデータが大量となった場合には，意味の数学モデルによる意味的連想検索の高速化が必要になるが，単純なパターン・マッチングによる連想検索などで利用されているハッシングなどの高速化技法を，動的に関係を計算するモデルにそのまま適用することはできない．しかし，意味の数学モデルは，ある単語に意味的に近い単語を，大量のデータの中から高速に抽出する能力を潜在的に備えている [4]．意味の数学モデルにおいて，意味の近い 2 つの単語は，空間上における距離の近い 2 点として表されている．本稿では，この性質を利用して，意味の数学モデルによる意味的連想検索の高速化を実現する方式を示す．

2 意味の数学モデルによる意味的連想処理

2.1 概要

ここでは，意味の数学モデルの概要を示す．

(1) 前提: いくつかの単語を特徴づけたデータの集合が，m 行 n 列の行列（以下，“データ行列”と呼ぶ）の形で与えられているものとする．この行列において，m 個のそれぞれの単語（word）は，n 個の特徴（features）によって特徴づけされている．

(2) イメージ空間 $\mathcal{I}$ の設定: データ行列から，特徴づけに関する相関行列をつくる．そして，相関行列を固有値分解し，固有ベクトルを正規化する．相関行列の対称性から，この全ての固有値は実数であり，その固有ベクトルは互いに直交している．このとき，非ゼロ固有値に対応する固有ベクトル（以下，“意味素”と呼ぶ）の張る正規直交空間をイメージ空間 $\mathcal{I}$ と定義する．この空間の次元 ν は，データ行列のランクに一致する．また，この空間は，ν 次元ユークリッド空間となる．

(3) 意味射影の集合 Π_ν の設定: イメージ空間 $\mathcal{I}$ から固有（不変）部分空間（以下，“意味空間”と呼ぶ）への射影（以下，“意味射影”と呼ぶ）の集合 Π_ν を考える．i 次元の意味空間は，$\frac{\nu(\nu-1)\cdots(\nu-i+1)}{i!}$, $(i = 1, 2, \cdots, \nu)$ 個存在するので，射影の総数は，2^ν となる．つまり，このモデルは，2^ν 通りの意味の様相の表現能力をもつ．

(4) 意味解釈オペレータ S_p の構成: 文脈を決定する ℓ 個の単語列（以下，“文脈語群”と呼ぶ）s_ℓ としきい値 ε_s が与えられたとする．このとき，その文脈に応じた意味射影 $P_{\varepsilon_s}(s_\ell)$ を決めるオペレータ（以下，“意味解釈オペレータ”と呼ぶ）S_p を次のように構成する．

- **(a)** 文脈語群 s_ℓ を構成する ℓ 個の単語を各々イメージ空間 $\mathcal{I}$ へ写像する．この写像では，ℓ 個の単語を各々イメージ空間 $\mathcal{I}$ 内でフーリエ展開し，フーリエ係数を求める．これは，各単語と各意味素の相関を求めることに相当する．
- **(b)** 各意味素ごとに，フーリエ係数の総和を求める．これは，文脈語群 s_ℓ と各意味素との相関を求めることに相当する．また，このベクトルは，ν 個の意味素があるため，ν 次元ベクトルとなる．このベクトルを，無限大ノルムによって正規化したベクトルを，以下，文脈語群 s_ℓ の意味重心と呼ぶ．
- **(c)** このとき，文脈語群 s_ℓ の意味重心を構成する各要素において，しきい値 ε_s を越える要素に対応する意味素を，単語を射影する意味空間の構成に用いる．これにより，意味射影 $P_{\varepsilon_s}(s_\ell)$ を決定する．

このオペレータは，文脈語群と相関の高い意味空間の自動的な選択を実現する．

(5) 意味空間における距離計算: 文脈語群 s_ℓ により，各意味素ごとに重みを定める．そして，意味空間において，その重みを考慮した単語間の距離計算を行う．これにより，文脈に忠実な単語間の関係の解釈が可能となる．

このモデルにより，文脈に応じた単語間の関係の解釈（意味空間の選択，およびその空間内における最良近似）が可能となる．

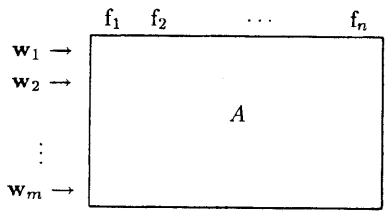

図 1: データ行列 A の構成

2.2 定式化

本節では，意味の数学モデルの定式化について述べる．

2.2.1 イメージ空間 $\mathcal{I}$ の設定

ここでは，m 個の単語について各々 n 個の特徴 $(f_1, f_2, \cdots, f_n)$ を列挙した各単語に対する特徴付ベクトル $\mathbf{w}_i (i = 1, \cdots, m)$ が与えられているものとし，そのベクトルを並べた m 行 n 列のデータ行列を A とする (図 1).

1. データ行列 A の相関行列 A^TA を作る．
2. A^TA を固有値分解する．

$$A^TA = Q \begin{pmatrix} \lambda_1 & & & \\ & \ddots & & \\ & & \lambda_\nu & \\ & & & 0 \ddots 0 \end{pmatrix} Q^T,$$

$$0 \leq \nu \leq n.$$

ここで行列 Q は，

$$Q = (\mathbf{q}_1, \mathbf{q}_2, \cdots, \mathbf{q}_n)^T$$

である．この $\mathbf{q}_i$ は，相関行列の固有ベクトル，つまり意味素である．

3. このとき，イメージ空間 $\mathcal{I}$ を以下のように定義する．

$$\mathcal{I} := span(\mathbf{q}_1, \mathbf{q}_2, \cdots, \mathbf{q}_\nu).$$

$(\mathbf{q}_1, \cdots, \mathbf{q}_\nu)$ は $\mathcal{I}$ の正規直交基底である．

2.2.2 意味射影集合 Π_ν の設定

P_{λ_i} を次の様に定義する．

$P_{\lambda_i} \overset{\text{d}}{\Longleftrightarrow} \lambda_i$ に対応する固有空間への射影，

i.e. $P_{\lambda_i} : \mathcal{I} \to span(\mathbf{q}_i)$.

意味射影の集合 Π_ν を次のように定義する．

$$\begin{aligned} \Pi_\nu := \{ \; & 0, P_{\lambda_1}, P_{\lambda_2}, \cdots, P_{\lambda_\nu}, \\ & P_{\lambda_1} + P_{\lambda_2}, P_{\lambda_1} + P_{\lambda_3}, \cdots, P_{\lambda_{\nu-1}} + P_{\lambda_\nu}, \\ & \vdots \\ & P_{\lambda_1} + P_{\lambda_2} + \cdots + P_{\lambda_\nu} \}. \end{aligned}$$

Π_ν の要素の個数は 2^ν 個であり，これは 2^ν 通りの意味の様相表現ができることを示している．

2.2.3 意味解釈オペレータ S_p の構成

文脈語群

$$s_\ell = (\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_\ell)$$

と，正数 $\varepsilon_s (\varepsilon_s > 0)$ が与えられたとき，意味解釈オペレータ S_p は，その文脈語群 s_ℓ に応じて，意味射影 $P_{\varepsilon_s}(s_\ell)$ を決定する．すなわち，$s_\ell \in T_\ell$, $\Pi_\nu \ni P_{\varepsilon_s}(s_\ell)$ とすると，意味解釈オペレータ S_p は，T_ℓ から Π_ν への作用素として定義される．また，$\{\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_\ell\}$ は，特徴付ベクトルであり，データ行列 A の特徴と同一の特徴を用いている．オペレータ S_p は次のように定義される．

1. $\mathbf{u}_i (i = 1, 2, \cdots, \ell)$ をフーリエ展開する．
   $\mathbf{u}_i$ と $\mathbf{q}_j$ の内積を u_{ij} とする．

$$u_{ij} := (\mathbf{u}_i, \mathbf{q}_j) \;, \; j = 1, 2, \cdots, \nu.$$

   ベクトル $\hat{\mathbf{u}}_i \in \mathcal{I}$ を次のように定める．

$$\hat{\mathbf{u}}_i := (u_{i1}, u_{i2}, \cdots, u_{i\nu}).$$

   これは，単語 $\mathbf{u}_i$ をイメージ空間 $\mathcal{I}$ に写像したものである．

2. 文脈語群 s_ℓ の意味重心 $\mathbf{G}^+(s_\ell)$ を求める．

$$\mathbf{G}^+(s_\ell) := \frac{\left(\sum_{i=1}^{\ell} u_{i1}, \sum_{i=1}^{\ell} u_{i2}, \cdots, \sum_{i=1}^{\ell} u_{i\nu}\right)}{\| \left(\sum_{i=1}^{\ell} u_{i1}, \sum_{i=1}^{\ell} u_{i2}, \cdots, \sum_{i=1}^{\ell} u_{i\nu}\right) \|_\infty}$$

   この $\| \cdot \|_\infty$ は，無限大ノルムを示す．

3. 意味射影 $P_{\varepsilon_s}(s_\ell)$ の決定

$$P_{\varepsilon_s}(s_\ell) := \sum_{i \in \Lambda_{\varepsilon_s}} P_{\lambda_i} \quad \in \Pi_\nu.$$

   但し $\Lambda_{\varepsilon_s} := \{ \, i \, | (\mathbf{G}^+(s_\ell))_i > \varepsilon_s \}$ とする．

2.2.4 意味空間における距離計算

単語 $\mathbf{x}$ と単語 $\mathbf{y}$ 間の距離 $\rho(\mathbf{x},\mathbf{y};s_\ell)$, $\mathbf{x},\mathbf{y} \in \mathcal{I}$ を次のように定める.

$$\rho(\mathbf{x},\mathbf{y};s_\ell) = \sqrt{\sum_{j\in\Lambda_{\epsilon_s}} \{c_j(s_\ell)(x_j - y_j)\}^2},$$

ここで, $c_j(s_\ell)$ は, 文脈語群 s_ℓ に依存して決まる重みであり, 次のように定義する.

$$c_j(s_\ell) := \frac{\sum_{i=1}^{\ell} u_{ij}}{\| \left(\sum_{i=1}^{\ell} u_{i1}, \cdots, \sum_{i=1}^{\ell} u_{i\nu}\right) \|_\infty}, \quad j \in \Lambda_{\epsilon_s}.$$

3 意味的連想検索の高速化アルゴリズム

意味の数学モデルによる意味的連想検索は, 文脈を表す単語列 (文脈語群) に対応する意味空間 (部分空間) を選択し, その意味空間において, 検索キーワードに最も近い意味の単語を検索対象語群の中から選ぶことによって, 実現される.

ある文脈語群が与えられ, その文脈において, 検索キーワードに最も意味の近い検索対象語を求めるとき, 全ての検索対象語との距離を計算するのでは高速な応答を行うことができない. そこで, 次のアルゴリズムによって距離計算を行うことにより, 全ての検索対象語を対象とした距離計算を行うことなく, 検索キーワードと意味の近い順に, 任意の個数の検索対象語を求めることができる.

本アルゴリズムでは, あらかじめ, 各意味素ごとに, その要素の大きさに応じて, 検索対象語がソートされていることを前提とする.

検索キーワード x のベクトルを

$$\mathbf{x} := (x_1, x_2, \cdots, x_\nu) \ , \ \mathbf{x} \in \mathcal{I}$$

とし, 検索対象語のベクトル $\mathbf{y}_i$ を

$$\mathbf{y}_i := (y_{i1}, y_{i2}, \cdots, y_{i\nu}) \ , \ \mathbf{y}_i \in \mathcal{I} \quad , i = 1, 2, \cdots, m$$

とする.

(1) 文脈に応じた部分空間 $P_{\epsilon_s}(s_\ell)\mathcal{I}$ を形成する意味素の中から, 意味重心 $\mathbf{G}^+(s_l)$ に最も関連が強い意味素 $\mathbf{q}_j$ を選ぶ.

(2) 検索キーワードからの距離の小さい順に並べられた解の候補リスト $\mathcal{Z}$ を $NULL$ に初期化する. 検索対象語のベクトルの集合 $\mathcal{Y}$ を, 全検索対象語として初期化する.

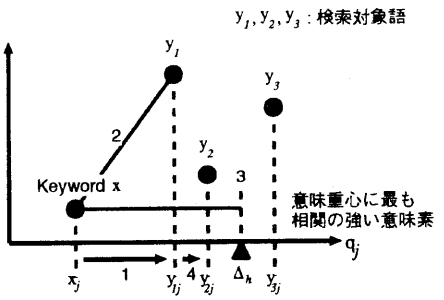

図 2: 意味的連想検索の高速化アルゴリズム

(3) 解の候補リスト $\mathcal{Z}$ の先頭要素を, 解の候補 $\mathbf{z}$ とする.

(4) 解の候補 $\mathbf{z}$ が $NULL$ ならば, 範囲変数 Δ_h を ∞ に初期化する. 解の候補 $\mathbf{z}$ が $NULL$ でなければ, 範囲変数 Δ_h を, 検索キーワードと解の候補 $\mathbf{z}$ との部分空間 $P_{\epsilon_s}(s_\ell)\mathcal{I}$ 上での距離 $\rho(\mathbf{x},\mathbf{y}_z)$ とする (図 2-3) .

(5) 集合 $\mathcal{Y}$ の要素から, $x_j \pm \Delta_h$ の範囲内にあり, かつ, 意味素 $\mathbf{q}_j$ 上において, 検索キーワードのフーリエ係数 x_j に最も近いフーリエ係数 y_{kj} $(1 \le k \le m)$ を持つ検索対象語のベクトル $\mathbf{y}_k \in \mathcal{Y}$ を探す. (図 2-1 : この検索は, 検索対象語を意味素 $\mathbf{q}_j$ 上における値の大きさの順にソートしていることにより, 高速に実行できる.) もしも $\mathbf{y}_k$ が存在するならば, 集合 $\mathcal{Y}$ から取り除く. もし, $\mathbf{y}_k$ が存在しなければ, 解の候補リスト $\mathcal{Z}$ から $\mathbf{z}$ を取り除き, $\mathbf{z}$ を解とする. 解の個数が指定数未満ならば, (3) に行く. 指定個数の解が得られたならば, 終了する.

(6) 文脈に応じた部分空間上で, 検索キーワードのベクトル $\mathbf{x}$ と検索対象語のベクトル $\mathbf{y}_k$ との距離 $\rho(\mathbf{x},\mathbf{y}_k)$ を求める (図 2-2) . 解の候補リスト $\mathcal{Z}$ に $\mathbf{y}_k$ を追加する.

距離 $\rho(\mathbf{x},\mathbf{y}_k)$ が, 範囲変数 Δ_h より大きければ, (5) に行く. 距離 $\rho(\mathbf{x},\mathbf{y}_k)$ が, 範囲変数 Δ_h より小さければ, ベクトル $\mathbf{y}_k$ を解の候補 $\mathbf{z}$ とする. そして, (4) に行く.

4 実験

高速な意味的連想検索アルゴリズムの有効性の検証のために，全ての検索パターンを試行し，検索時の距離計算回数の測定を行うことが考えられる．しかし，意味の数学モデルによる連想検索では，検索キーワードと文脈の組合せにより，無限に近いパターンの検索を行うことが可能である．したがって，意味的連想検索の高速アルゴリズムの有効性を検証するにあたって，全ての検索パターンを試行するのは不可能である．そこで，Longman Dictionary of Contemporary English[7] における基本英単語 2328 語を対象として，それぞれの単語が 1 回ずつ解となることを想定した実験を行った．

4.1 実験環境

実験では，限られた数の基本英単語のみを使用して単語の定義を行っている英英辞典を 2 つ使用した．具体的には，Longman Dictionary of Contemporary English における基本英単語 2328 語を，The General Basic English Dictionary[9] の定義を用いて定義し，単語の活用形を基本形に戻す filter を通して 2328×874 の行列を作成し，イメージ空間を構成した．検索対象単語群として，上述の基本英単語 2328 語を用いた．上の行列において，同じ見出し語を持つ単語群の定義を合成した行列を使用して検索キーワードとなる単語群の定義を行った．

実験に使用した計算機は Sun4/ELC．OS は SunOS 4.1.4 である．

4.2 実験方法

実験を次の方法によって行った．

(1) 検索対象語群から，1 単語を選び出し，それを $\mathbf{d}_i$ とする．

(2) $\mathbf{d}_i$ の定義に使用されている語句を，文脈語群とする．

(3) $\mathbf{d}_i$ と同じ綴の単語を，検索キーワードとする．

(4) 検索対象語群の中から，与えた文脈において，検索キーワードに近い順に，10 単語検索する．

検索対象語 2328 単語の中から，検索キーワードに近い順に 10 単語を検索し，その時に行った距離計算を測定する．解と想定した 2328 語の各々について，意味的連想検索を行い，その処理に必要な平均距離計算回数を求めた．また，意味空間のしきい値 ε_s を，0.0 から 0.9 まで 0.1 きざみで変化させ，それによる影響を調べた．

4.3 実験結果

実験結果を 図 3 に示す．各グラフは，1 回の検索において抽出するデータ数に対応しており，(1) は第 1 位のデータだけを抽出する場合，(2) は第 1 位及び第 2 位の，(10) は第 1 位から第 10 位のデータを抽出する場合を示している．また，“failure” に対応するグラフは，解に想定した検索対象データ以外のデータが第 1 位のデータとして抽出された数を示している．“all” は高速化のアルゴリズムを使用しなかった時の計算回数であり，キーワードと全検索対象語との距離を計算した時の計算回数に等しい．横軸は ε_s の設定に対応している．

検索キーワードに最も近い検索対象語だけを求めた場合と比較すると，2 番目，3 番目に近い検索対象語，すなわち，2 個，3 個の解を求めた時には，より多くの距離計算が必要であったことがわかる．しかし，その場合においても，全ての検索対象語との距離計算は必要ないことがわかる．

また，全体的な傾向として，意味空間のしきい値 ε_s の増加に伴い，距離計算回数が減少している．これは，意味空間のしきい値 ε_s の増加に伴い，距離計算を行う空間の次元数が減少していることにより，本アルゴリズムの有効性が，より強く現れることによる．しかし，ε_s の増加に伴い，想定した解と異なる解が選択される回数 （“failure” に対応するグラフ ） も増加を始めるので，ε_s に関しては，0.5 以下に設定するのが望ましいと考えられる．なお，“failure” に関しては，意味の数学モデルにおける学習機構を適用することで改善が可能である．

本アルゴリズムにより，求める解の個数に応じて距離計算回数は多くなるが，本アルゴリズムを用いない場合との比較において，本アルゴリズムにより距離計算回数を軽減できることが明らかとなった．

5 おわりに

本稿では，データベース・システムにおける情報探索のための高速な意味的連想処理方式の実現方式について述べた．この方式により，大量データの中から意味的に関連するデータを文脈に応じて動的かつ効率的に検索することが可能となる．高速な意味

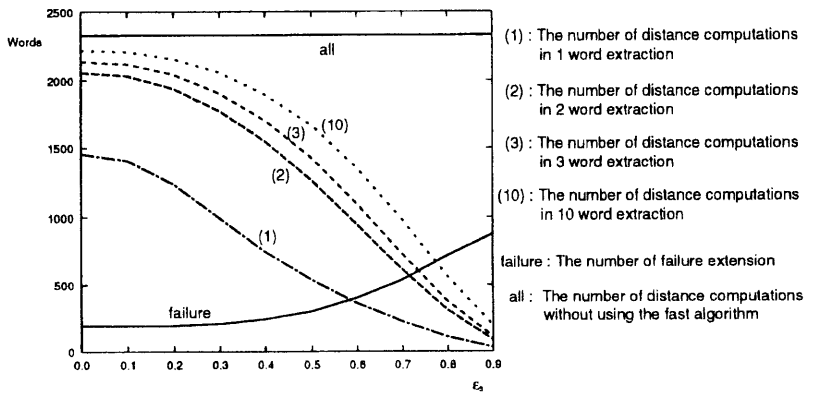

図 3: 基本英単語に関する実験結果

的連想検索に関する実験結果を示し，高速な意味的連想検索アルゴリズムの有効性を確認した．

今後は，本稿で述べた高速な意味的連想検索アルゴリズム，および，意味的連想検索のための学習機構の実現を行っていく予定である．

参考文献

[1] David, R., and Lenat, D.B., *"Knowledge-based systems in artificial intelligence,"* MaGraw-Hill Book Co., 1982.

[2] Kitagawa, T. and Kiyoki, Y., *"The mathematical model of meaning and its application to multidatabase systems,"* Proceedings of 3rd IEEE International Workshop on Research Issues on Data Engineering: Interoperability in Multidatabase Systems, pp.130-135, April 1993.

[3] Kiyoki, Y.,Kitagawa, T. and Hitomi, Y., "A fundamental framework for realizing semantic interoperability in a multidatabase environment," Journal of Integrated Computer-Aided Engineering, Vol.2, No.1, pp.3-20, John Wiley & Sons, Jan. 1995.

[4] Kiyoki, Y.,Kitagawa, T. and Miyahara, T.,"A fast algorithm of semantic associative search for databases and knowledge bases," Information Modelling and Knowledge Bases (IOS Press), Vol. VII, 4.1-4.16,1995.

[5] Kolodner, J.L., *"Retrieval and organizational strategies in conceptual memory: a computer model,"* Lawrence Erlbaum Associates, 1984.

[6] Krikelis, A., Weems C.C., "Associative processing and processors," IEEE Computer, Vol.27, No. 11, pp.12-17, Nov. 1994.

[7] *"Longman Dictionary of Contemporary English,"* Longman, 1987.

[8] *"Natural language processing,"* Comm. ACM, Vol.39, No.1, Jan. 1996.

[9] Ogden, C.K., *"The General Basic English Dictionary,"* Evans Brothers Limited, 1940.

[10] Potter J.L., *"Associative Computing,"* Frontiers of Computer Science Series, Plenumn, 1992.