人文科学とコンピュータ　29－5
（1996．1．27）

構造マッチングによる文献の知的検索と結果の色空間表示

安部 隆之　佐藤 浩史　重松 修一　中島 誠　伊藤 哲郎
大分大学工学部 知能情報システム工学科
〒870-11 大分市大字旦野原700番地

ネットワークの広範な普及に伴い，自然言語文が扱え結果の適合判断が容易なインタフェースを備えた検索システムの構築に注目が集まっている．本報告では，この研究の流れに沿って，語の綴り，句の並びや構文構造の似ている割合を類似度として求める方法を構造マッチングと名づけて定式化し，自然言語文の扱える検索システムの構築を試みる．結果の扱いでは，図書館でのブラウジングの形態を念頭に置き，取り出された文献同士について構造マッチングをもとにして文献の空間を作り，視覚的なブラウジングを可能にすることを試みる．これらの方法の有効性は計算機実験を通じて調べる．

Intellectual Document Retrieval Using Structure Matching and Visualization of Retrieved Results on a Munsell Color Space

Takayuki ABE　Hiroshi SATO　Shuichi SHIGEMATSU　Makoto NAKASHIMA　Tetsuro ITO
Department of Computer Science and Intelligent Systems, Oita University
700 Dannoharu, Oita-shi, Oita 870-11, Japan

Recent researches in information retrieval focus on natural language query processing and on developing graphical user interface. This research report describes a method of computing similarities, called the structure matching procedure, between queries and document sentences under the consideration that two sentences semantically similar have many words, phrases and syntactic structures in common. As to the effective user interface design, a method of visualizing retrieved results on a Munsell color space, where the novices can browse the results as if they are in a library, is presented. The proposed methods were examined by the computational experiments.

1 まえがき

ワードプロセッサを使った文書作成や印刷文書の機械可読化並びにネットワークの普及に伴い，文献検索システムで扱うべきデータが膨大なものとなってきており，システムを利用する側も研究者以外の一般ユーザが多くなってきている．このような状況で，キーワードをベースに探索を進め結果をリストの形で提示する既存の様式に頼っていては，効率の高い検索は困難であるため，自然言語文が扱え [1, 3], 適合判断が容易なインタフェースを備えた [8] 検索システムの構築に注目が集まっている．

自然言語文によると要求のより細かな表現が可能になる．ただし，文解析に通常の仕方を採用すれば，概念辞書整備のための大きなコストを要する．検索では，質問文と内容的によく似ていると思われる文を含む文献が取り出せれば良い．これに注目して，語の文中での大まかな役割とシソーラスを通じた語間の関連を使った検索や [1], 実例による機械翻訳にヒントを得て，文中に含まれる文字列同士のマッチングによる検索が提案されている [6]．結果の出力に関しては，ユーザが適合判断のためのブラウジングを効率的に進めて行けるよう，単純な文献リストでなく，質問との関連性の高さに従いランクづけたり視覚的なグラフを使ったりする方法の開発が行われている [2, 8].

本報告では，この研究の流れに沿って，簡単な構文解析を通じて得た文の構造をもとに，その構成要素同士の「見た目の重なり具合」を類似度として求める方法を構造マッチングと名づけて定式化し，自然言語文が扱える検索システムの構築を試みる．また，結果表示の仕方としては，取り出した文献同士について構造マッチングで求めた類似度をもとに，関連の高い（低い）ものはできるだけ近く（遠く）になるような線形性を保ちながら，結果全体を 3 次元距離空間に配置することを試みる．

構造マッチングの特徴は，語の綴り，句の並びや構文構造を勘案しながらこれらが多少違っていても，また概念辞書を用いずとも，文同士の関連性の高さを類似度の形で求められるところにある．また，空間表示は文献の線形配置を基本にしているため，適合判断に際しユーザは空間中の近くにある文献は同じと認識しながら，ブラウジングして行ける．

以下，2 章で構文解析・構造マッチング法，3 章で結果の表示法について述べる．4 章では計算機実験を通じてこれら方法の有効性について評価する．

```
drive taro . car . . . . . parents
drive taro . car . . . . . parents .
```

図 1. 解析結果

2 構文解析と構造マッチング

2.1 構文解析

種々の文の扱いを可能にするには，複雑な意味処理を要しない形で自然言語文解析する必要がある．ここでは，ネットワークを通じて入手可能で，格構造での結果を出力できる LangLAB[11] を用いる．LangLAB の特徴を次にあげる．

(1) ユーザが辞書を整備してゆく（比較的簡単な操作で済む）ことで，入力文の構文構造が生成可能．

(2) 文の構成要素の移動に伴う根拠処理が可能．

(3) 単数・複数形，現在・過去・過去分詞形などの標準処理が可能．

LangLAB が扱う格は全部で 14 種類ある．図 1 に，英文 "Taro drives a car with his parents." に対する解析の結果を示す（2 つある）．

LangLAB は辞書を参照しながら可能な数だけの解析結果を構文木の形で出力するが，ここでは，文間の類似度が求められればよいため，構文木をそれぞれの格に文中の語や句を割り当てた形のリストに変換して用いる．以後解析結果といえば，語や句が現れる場所が格に対応した図 1 の形でのリストとする．入力文が複文の場合には，単文ごとの結果の組合せとなる．

2.2 構造マッチング

同様の事柄に言及している文は同じような言い回しがされていると考えると，2 つの文が内容的に似ているかどうかを判断する指標に，これらの文を解析した結果が構造的にどれくらい似ているかを示す類似度を使うことができる．文の解析結果は，複文なら単文の集まりとして，単文は格が割り当てられた句，句は単語，そして単語は文字から構成された形になっている．文間の類似度を計算するには，図 2 に示すように，まず 2 つの文中にある単語同士を文字列と捉えて類似度を求め，この値を句同士の類似度に，それをさらを単文同士，文全体の類似度にと反映させてゆけばよい [7].

具体的に文の各構成要素間の類似度（最大値は 1.0）は次の測度 Sim によって求める．

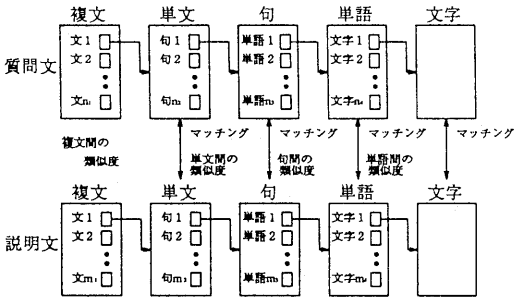

図 2. 構造マッチング

$$Sim(d_i, d_j) = (u_1 + u_2)/(n + 2(n-1))$$

例えば，句 d_i と句 d_j の類似度を求める場合，u_1 は d_i と d_j に含まれる単語を組み合わせて求めた類似度のうちの最大値，u_2 は句中で隣合う単語を一組として求めた最大値である．また分母に現れる n は，句 d_i 中の単語の総数である．単語間や複文間の類似度もこの式で求める[1]．ただし，単文間の類似度に関しては，句それぞれに格が割り当てられているため，u_1 は同じ格あるいは同じと見なしてよい格（ここでは行為者格と主語に相当する格，動詞の補語に対応する格と時間格）の位置にある句を取り上げる．このように，単語綴り，句の構成や構文構造の多少違いを考慮に入れながら，文同士の類似度を計算する操作を構造マッチングと呼ぶ．もし，シソーラスが準備されていれば，単語綴りのマッチングの際に，[10] でのように，シソーラス上での概念的関連を数値で表現した値に置き換えることもできる．

測度 Sim の分母として，d_i と d_j 中の構成要素数の合計とするのが一般的であるが，ここでは d_i だけの要素数とした．これは，d_i を質問文と考えたとき，それの構成要素が文献の説明文中にどれくらいの割合で含まれているかだけを見ることで呼出率の高い検索を目指す便法である．

辞書を厳密な形で準備する必要性がないことの裏返しとして，LangLAB による解析結果は複数でてくることがある．この場合には，可能な解析結果ごとに構造マッチングを試し，そのうちの最も大きな類似度を採用する．文献同士の類似度計算では，文献は複数の説明文の集まりと解釈し，複文と同様に扱う．もし辞書項目が不十分であったり入力文に誤りが含まれていれば，LangLAB は解析を放棄してしまう．この問題点への対処には，文全体を句と解釈して処理するようにする．

[1] 現在の所，d_i や d_j が複文や単文のとき u_2 は 0 で，分母は n としている

2.3 類似行列

質問文と文献中の文に対する構造マッチングの結果を用いれば，質問との類似度の大きさに従う文献のランクづけ出力が可能になる．この形式では，上位の方に関連が高いと思われる文献が出現するため，ユーザの適合判断に大きく役立つとされてきた [9]．しかしながら，出力文献相互間の関連性が明示的でないため，これだけでは十分な方式とはいえない [2]．相互関連が分かれば，それを利用してより迅速で正確な適合判断が可能になる．

文献相互間の関連を示す方法として，よく似たもの同士は近くに，そうでないものは遠くになるよう線形に並べることを考える [4]．具体的には，構造マッチングにより文献間の類似度を求め，これを要素とした類似行列Sを作る．そしてSを対称律・推移律を満たすよう変更した推移行列Tを求める．最後に，行列Tでの類似度を使って文献を線形に並べる．この便法は，見方を換えれば，高い類似度を示す文献対を近くに集めるクラスタ化に相当する．

行列Tで対称律を満たすようにするとき，文献 d_i と d_j の類似度を $min(Sim(d_i,d_j),\ Sim(d_j,d_i))$ で測る．この便法は，厳密な値に従って文献を関連づけることで，近くの文献は同一と考えてよいという指標にし，ユーザによる出力文献の適合判断に役立たせるためである．近くにある文献については，そのうちの何れかをピックアップして調べればよく，少数の文献の検査で，多くを調べたのと同じ効果が得られると期待できる．推移律を満たすようにするには，$Sim(d_i,d_j) = max_k\ min(Sim(d_i,d_k), Sim(d_k,d_j))$ と類似度を変更すればよい．

図3に図1での質問に対する構造マッチングで高い類似度を示した文を線形に並べて取り出した結果を示す．質問文との類似度が高い文ほどできるだけ上位に，また互いに関連が高い文ほどできるだけ近くにあるようにしている [7]．この結果表示から，質問に真に関連あるものを取り出すのに，例えば，文 1,2,4,3,5,9 を調べればよい．

3 結果の空間配置

前節では推移行列Tを利用して結果を線形に並べる方法を示した．しかしながら，もとの行列Sでの類似度について考えてみると，値が小さいが並びで

(1) Taro drives a car with his parents.

(2) Miss Reed usually drives her car because she must sometimes come to Seattle.

(6) If Tom has a car, I will come to Seattle with him tomorrow.

(4) I must take care of my mother who is sixty years old.
(8) I take care of him, because he is old.

(3) She takes care of me, because my mother died a long time ago.
(7) If his mother died, I take care of him.

(5) I will go to Seattle, because my mother died and miss Brown will take care of me.

(9) My mother bought a bag because I have wanted it for Christmas present.
(10) I want a small car, but I can not buy it.

図 3. 結果 1

は近くになってしまう文献対がでてくる可能性がある．文献と併せて行列Sを提示しても，そこから文献間の関連を迅速に直接読みとることはむずかしい．これらの問題点の解決には，Sを3次元（以下の）空間での距離を反映した距離行列Cに変換し，これを用いて検索結果をグラフィックディスプレイ上に空間的に配置して，視覚に訴える形で表示をすることが考えられる．

3.1 遺伝的アルゴリズムの利用

配置に統計的な手法を利用するのも一案である．ただし，文献を3次元空間全体に分散させて配置するより，線形の並びを反映させて配置した方がインタフェースの向上が図れる．ここでは，遺伝的アルゴリズム GA[5][2]を利用して，Tでの線形性とSでの類似度の大小関係をできるだけ保ちながら，文献を3次元距離空間に配置することを考える．

手続きは大きく分け，線形性を反映させるステップ sp1,sp2 と距離空間上の座標を求めるステップ sp3 とからなる．

(sp1) 行列 T を使った文献の並びで行列 S の値を並び換える．

(sp2) 新しい S で，行列要素の対角線からの離れ具合に応じ，そこでの類似度を減少させる．より遠くにある類似度をより多く減少させる．

(sp3) sp2 で得た行列をもとに GA を使って3次元空間中に誤差が小さくなるよう文献を配置する．

[2]問題の解の候補を複数適当に求め，これらのうちから高い評価の与えられた解を組み合わせながら，よりよい解に進化させてゆく確率的アルゴリズム．

GA では配置された文献 d_i と d_j 間の距離（最大を 1.0 として正規化する）と非類似度（$1.0 - Sim(d_i,d_j)$）の差の絶対値の和の合計が小さいほどよいとして進化させる．

3.2 マンセル色空間の利用

上記のアルゴリズムにより，線形性を反映させた配置のための座標が得られる．グラフィックディスプレイ上への表示のための具体的な座標系としては，一般的な直交座標系でもよいが，本稿ではマンセル色空間（円周方向，半径方向，縦方向がそれぞれ色相，彩度，明度を示す）の利用を考える．マンセル色空間によれば，文献の関連状況が空間的な近さだけなく，色の似具合でも認識できる．また，個々の文献は色の違いで区別できることから，空間における文献の絶対座標が認識でき，ブラウジングに際し，これまで調べた文献を再度調べてしまうといった非能率性を避けれるようになる．

4 実験的考察

本稿で提案した構造マッチング法と結果の空間配置法の有効性を調べるため，計算機実験を行った．一般的な文献を扱うと，質問それぞれに対して合致する文献を客観的な手段で判断しなければならない．ここでは，実用的な検索システムに発展させてゆくための基礎データを得ることを目的として，実験者が 2.1 で示したような英文を 100 を用意し，1つ1つを文献の代わりと見なして蓄積しておいてから実験を進めた．

4.1 実験 1

実験1では質問文に関連した文を取り出す上での構造マッチングの働きを調べた．3つの検索方式を比較した．1番目は，従来から取られてきたもので，文をキーワードのベクトルに変換し，質問文と共有するキーワードの数の割合が高い文を上位にランクする方式である．2番目は文全体を1つの句と見なし，検索には構造マッチングを用いるものである．残りは LangLAB を用いて解析した構文構造について構造マッチングを用いる方式である．（マッチングの際，冠詞や前置詞など不要語と思われるものは除いた．）

質問文として，蓄積した文から選んだもの，その中の名詞に些細な誤りが含まれている，名詞と動詞に些細な誤りが含まれているものそれぞれについて

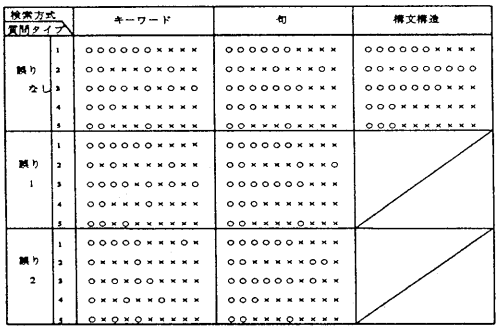

図 4. 検索結果の比較

5組の合計15文を用意した．ただし，入力に誤りが含まれているとLangLABでの解析が不可能になるため，3番目の検索方式では最初の質問のみを取り上げた．

ランク10位まで出力し，各文が質問に関連あるか否かを調べた．質問ごとの結果を図4に示す．出力を左から数えた位置にランクされた文が少しでも関連あると見なされた場合は「○」，それ以外は「×」とした．関連ある文をできるだけ多くかつ連続的に上位にランクした方法が最もすぐれている．この結果から方式2や3を利用すると，呼出率の高い検索システムが実現できると考えられる．

2番と5番目の質問に対しては，方式3がすぐれていた．これは質問中に蓄積文とは時制の違う不規則動詞が現れていたり，質問中の名詞の綴りが蓄積文中の動詞の綴りの一部と一致したためである．方式2でも小規模のシソーラスを準備したり，文を少し多めに取り出し，空間配置操作で整理するようにすれば（実験2参照）この欠点を補うことができる．ただし，方式3では，入力誤りがあったり，扱う文が多くなったりした場合，検索が不可能になったり構文解析にコストがかかってしまうという問題点が生じる．これらの考察から，検索時には，質問文を句と見なし，簡単なシソーラスを準備した状況で，構造マッチングを通じて上位にランクされたものを少し多めに取り出せばよいと考えられる．

4.2 実験2

上の方式2で取り出した文の集まりを，推移行列を使って線形に並べる実験を行った．取り出された文には入力誤りが含まれていないため構文解析が可能である．また，その数も少ない．この実験2では，文全体を，1つの句からなると見なす（方式A），LangLABを用いて解析した構文構造とする（方式B）の2つの場合について取り上げた．

(1) Taro drives a car with his parents.

(10) I want a small car, but I can not buy it.

(3) She takes care of me, because my mother died a long time ago.
(5) I will go to Seatlle, because my mother died and miss Brown will take care of me.

(4) I must take care of my mother who is sixty years old.

(7) If his mother died, I take care of him.
(8) I take care of him, because he is old.

(6) If Tom has a car, I will come to Seattle with him tomorrow.

(2) Miss Reed usually drives her car because she must sometimes come to Seattle.

(9) My mother bought a bag because I have wanted it for Christmas present.

図 5. 結果2

入力誤りを含む5種の質問文に対し，取り出したそれぞれの結果について，方式AとBでの並べられ方の違いを調べた[3]．質問とより関連の高いものができるだけ上位になるようかつ関連あるものがクラスタとしてまとめられればよい．

方式Aによれば，互いに関連の低い文同士も大きな1つのまとまりとされてしまうことが多かった．また，動詞が名詞の一部と一致するような場合には，うまく線形に並べられなかった．方式Bによれば，関連の高い文同士がまとまり，また質問5のような状況にもうまく対処できた．方式Aでの結果を図5に示す．方式Bでの結果は図3と同様になった。これより，方式Bによれば，重要と思われる格に重みをつけることで関連の高い文をうまくまとめることができると考えられる．

4.3 実験3

図6に実験2で求めた方式Bでの結果の空間配置の例を示す．色のついた球が文献を示している（現在の段階では，まだ文献それぞれを識別するための手段は取られていない）．ただし，マンセル色空間をそのまま用いると円周方向の距離感覚がつかみにくくなるため，円筒を切り開いた形で表示している．

この図からは手前に描かれた球（色は赤）に対応する文が関連あると視覚的に容易に認識できる．残

[3]ここで扱った文では代名詞を多く含んでいたため，同様の代名詞が現れているだけで類似度が大きくなってしまう．それゆえ，方式Bでは動詞の格での句のマッチングに相対的に大きな重みを設定した．

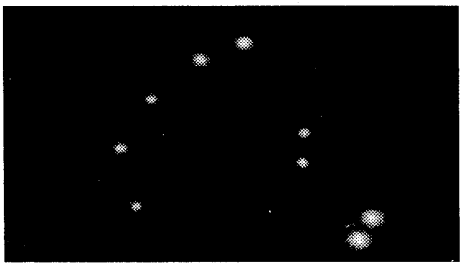

図 6. 結果の空間配置

りについては，２～３の小さな（色は緑や青で描かれた）クラスタが遠くには離れた状況で存在するため，これらについての適合判断をする必要はないと判断できる．図３のような推移行列を使った結果表示では文献間の関連を必ずしもうまく反映できていないため，ユーザはこのことを想定しながら適合判断を進めて行かなければならない．

5 むすび

文の詳しい内容解析を避けながら，質問文と似ていると思われる文献を取り出す方法を構造マッチングと名づけて定式化した．インタフェースの向上を図る一環として，関連の高い文献は近くにという線形性を保持した形で，検索結果を３次元グラフィックディスプレイ上に表示する方法についても述べた．実験的考察より，質問文の扱いとしては，キーワードベクトルより，文や句（処理コストを考えれば句）のように構造をもったものとした方がよいことが分かった．結果の表示については，構文構造のマッチングを通じて関連を求めておく方がすぐれていることが分かった．

質問の扱いでは比較的ゆるめられた形で高速の検索を行い，取り出された小規模の結果に関しては厳密な扱いとすれば，システム全体として，呼出率・適合率が共に高くまたコスト的にも効率的な検索が行えるようになると思われる．このようなシステム形態は，また，ネットワークを通じた世界的規模での検索システムに素直に組み入れられると考えられる．

今後の方向として，日本語文を含めた種々の分野の文献，メモ書きやマニュアルなどを対象として，ここでの方法の有効性を確かめて行きたい．また，空間表示については，最近の仮想現実や電子図書館に係わる研究の流れの中で，現実的に利用できるものに改良して行きたい．

参考文献

[1] Chakravarthy, A. S., and Hasse, K. B.: NetSerf: Using semantic knowledge to find internet information archives. Proc. of 18th ACM SIGIR Conference, Seattle, Washington, pp.4-11 (1995).

[2] Dublin, D.: Document analysis for visualization. Proc. of 18th ACM SIGIR Conference, Seattle, Washington, pp.199-204 (1995).

[3] Folts, P. W.: Using latent semantic indexing for information filtering. Proc. of ACM Conference on Office Information Systems, Boston, Massachusetts, pp.40-47 (1990).

[4] Ito, T., and Kizawa, M.: Hierarchical file organization and its application to similar-string matching. ACM Trans. on Database Systems, Vol. 8, No. 3, pp.410-433 (1983).

[5] 北野宏明 編: 遺伝的アルゴリズム, 産業図書 (1993).

[6] Morita, M., and Shinoda, Y.: Information filtering based on user behavior analysis and best match text retrieval. Proc. of 17th ACM SIGIR Conference, Dublin, Ireland, pp.272-281 (1994).

[7] 永野他: 構造マッチングをもとにした事例の知的検索, 電気関係学会九州支部連合会大会論文集, p. 827 (1995).

[8] Rao, R., Pedersen, J. O., et al.: Rich interaction in the digital library. Commun. ACM, Vol.38, No.4, pp.29-39 (1995).

[9] Salton, G., and J.McGill, M.: Introduction to Modern Information Retrieval. McGraw Hill, New York (1983).

[10] 佐藤理史: MBT2: 実例に基づく翻訳における複数翻訳令の組合せ利用, 人工知能学会誌, Vol.6, No.6, pp.861-871 (1991).

[11] 田中: LangLAB User's Manual (1989).