Web検索結果におけるキーワード出現相関の可視化と対話的な質問変換

吉田　大我† 　小山　聡†† 　中村　聡史†† 　田中　克己††

† 京都大学工学部情報学科 〒606-8501 京都市左京区吉田本町
†† 京都大学大学院情報学研究科社会情報学専攻 〒606-8501 京都市左京区吉田本町
E-mail: †{yoshida,oyama,nakamura,tanaka}@dl.kuis.kyoto-u.ac.jp

あらまし　Web検索においてページを探す場合，ユーザは目的のページを見つけるためのクエリを考える必要がある．しかし，適切なクエリを生成することは多くのユーザにとって困難である．本稿では，AND検索，OR検索，NOT検索を組み合わせた複雑な検索をユーザが意識することなく利用できるようなシステムを提案する．検索結果の提示手法として，検索結果ページ中に現れる重要語を話題語として抽出し，それらの出現傾向をグラフを用いて可視化することにより，検索結果ページの傾向を一覧的に表示することを考案した．そして，グラフ上に表示した話題語をユーザとの対話操作によって重み付けを行い，動的にクエリ修正および再ランキングをすることで検索支援を行った．また，このシステムを利用することによる検索効率について検証する．

キーワード　情報検索，可視化，Webとインターネット，対話操作，質問変換

Visualization of Correlations Between Keyword Occurrences in Web Search Results and Its Application to Interactive Query Transformation

Taiga YOSHIDA†, Satoshi OYAMA††, Satoshi NAKAMURA††, and Katsumi TANAKA††

† Department of Informatics and Mathematical Science, Faculty of Engineering, Kyoto University, Yoshida-Honmachi, Sakyo, Kyoto 606-8501 Japan
†† Department of Social Informatics, Graduate School of Informatics, Kyoto University Yoshida-Honmachi, Sakyo, Kyoto 606-8501 Japan
E-mail: †{yoshida,oyama,nakamura,tanaka}@dl.kuis.kyoto-u.ac.jp

Abstract　In a conventional Web search, a user must constitute keywords as a query to retrieve desirable Web pages. It is, however, difficult for many users to make a proper query. In this paper, we propose a way and a system that support a user when he or she executes a query. The system does not have users execute AND queries, OR ones, NOT ones, and so on. To enable a user to understand the tendency of search results, our system extracts important terms and topics from them, and constructs a graph from extracted terms. Moreover, we proposed a method to change dynamically the search results by weighting important terms displayed in the graph through the user interaction, and verified the usefulness of the system.

Key words　information retrieval, visualization, web and internet, interactive operation, query transformation

1. はじめに

ユーザが求めている情報が記述されるウェブページを探す方法は，大きく分けて2種類ある．1つは，求めている情報に関する分野のリンク集からページを辿る方法であり，もう1つはサーチエンジンを利用してページを検索する方法である．前者は一度人手によって分類され，ページの説明が加えられていることが多いため，効率的にページを探すことができる．ところが，そのようなリンク集をユーザが初めから知っていることはまれであり，そのリンク集自体をサーチエンジンによって検索することが多い．また，分類が大規模になると探せないという問題もある．

そのため，ある情報を知りたいと思ったとき，ユーザはサーチエンジンを用いることがほとんどである．

一般的なサーチエンジンは，クエリとなるキーワードを入力するテキストボックスと，検索を実行するボタンからなる．ユーザがテキストボックスにクエリを入力し検索を実行すると，結果がリストとして表示される．1つのキーワードのみをクエ

リとして検索すると，その結果ページ集合はさまざまな話題が混在したものとなる．したがってユーザは検索を行う際，求めている情報がうまく表示されるように適切な検索キーワードを選び，それらを AND や OR，NOT でつないだ検索式を組み立てる必要がある．しかし，多くのユーザにとって検索を行う際に，そうしたキーワードや検索式を生成することは困難である．

例えば，ユーザが「京都」の「東西線」について知りたいと思い，クエリとして「東西線」と入力したとする．表示される結果は「東京メトロ」の「東西線」に関するページが大半を占め,「京都」の「東西線」に関するページはほとんど表示されない．これは「東西線」という語が曖昧性を持つことが原因である．

曖昧性を回避するためには「京都」,「東京」,「札幌」,「仙台」,「JR」といった曖昧性を解消するキーワードをクエリに追加する必要がある．検索結果を眺め，目的のページが検索結果に表示されていないことに気がついたユーザは，どのようなキーワードをクエリに追加すれば検索結果が洗練されるのかを考える．このとき,「京都」を追加すると目的のページが見つかる可能性が高くなるが,「地下鉄」や「市営」といったキーワードを追加しても，他の「東西線」に関するページが同時に表示されてしまうため効果がない．また,「山科駅」などのキーワードを追加すると話題が限定されすぎてしまい，目的のページが結果中に表示されなくなってしまう可能性もある．

このような問題を解決する手法として，Google サジェスト [1] や Yahoo! Japan の関連検索ワード [2] などがある．これらのシステムでは，ユーザが入力したキーワードと同時にクエリとして利用されやすいキーワードを提示することで，ユーザの検索行為を支援する．しかし，このような手法ではクエリログに依存しており，実際に検索結果中に含まれる話題は考慮されていない．そのため，クエリ修正によってどのようなページが表示されるようになるのかは分からず，目的のページを効率的に探せないキーワードを提示してしまう可能性もある．

そこで本研究では，検索結果がどのような話題を含んでいるかを一覧的に見ることを可能にするため，検索結果のスニペット中に出現する重要な語を話題語として定義し，出現傾向や共起関係によって分類した話題語間の関係をグラフを用いて可視化した．

可視化することにより，ユーザは求める情報と検索結果が合致しているかどうかを判断し，合致していないならばどのような語を重視することで目的の情報を含むページを見つけることができるかを知ることができる．

また，提案したシステムでは，複雑な検索式を生成する代わりにグラフ上に表示された話題語間の関係を操作することによって，クエリの修正および再ランキングを行うことができる．

本稿の 2 章では関連研究について述べ，3 章では提案システムの概要について述べる．4 章ではサーチエンジンの検索結果から話題語を抽出する方法について述べ，5 章ではユーザ操作による質問修正について述べる．6 章で検証および考察を行い，7 章でまとめと今後の課題について述べ，8 章で比較検索への応用について述べる．

2. 関 連 研 究

2.1 検索結果の可視化

ウェブページや情報を可視化する技術は多数存在する [3]. KeyGraph [4] はテキストデータ中の重要な語をノードとし，それらを共起関係によって結んだネットワーク図で可視化することで，文章構成のキーワード抽出を行うソフトである．KeyGraph は 1 つの文書を解析対象としているだけであり，Web 検索結果全体などは解析対象としていない．また，可視化したネットワークを対話的に操作することは想定されていない．

納豆ビュー [5] は，Web 空間を 3 次元グラフィックスにより可視化するもので，あるノードを持ち上げる操作により芋づる式にそのノードに関係するノードがすべて持ち上げることができるため，関係の深いノードを見ることが容易である．しかし，納豆ビューではグラフ操作により対話的にコンテンツをフィルタリングしたり再ランキングしたりすることは想定されていない．

2.2 対話的な質問修正

ユーザの操作を検索結果に反映させるものとして，Yahoo! Mindset [6] などが挙げられる．Yahoo! Mindset では，検索結果リストに表示される各検索結果について買い物（shopping）に適したページなのか，情報収集 (researching) に適したページなのかといったスコアリングを行っている．ユーザは，現在検索を行っている対象が買い物に関することなのか，情報収集に関することなのかによってスライドバーを操作することでシステムに指示する．システムはユーザの操作に基づき検索結果を動的に再ランキングする．

また，ユーザの検索結果による再ランキングを行うシステムとして 121r [7] がある．121r では，ユーザがレーダーチャート中に表示された多次元の軸を操作して変更することにより，検索結果の再ランキングを行うものである．例えば，宴会場を探す場合は,「リーズナブル」や「まんぷく度」といった軸がある．ユーザは値段の安い宴会場を探しているなら「リーズナブル」の値を増やし，高くてもいいなら値を減らすといった操作を行う．すると，システムは求める条件に合う店を上位に表示するよう再ランキングを行う．

これらのシステムは，ユーザの操作により検索結果の再ランキングを行う点は本研究と同じであるが，評価軸を動的に生成することを想定していない．本研究では，検索結果中から重要な語を抽出し，それらを評価軸の生成に利用する．

検索結果中から抽出した語を提示し，検索支援を行う研究として，松生らによる研究がある [8]．ここでは，検索結果ページの集合を複数のキーワードを組み合わせた検索式によって表現する．また，得られた検索式を決定木のような木構造により可視化することで，ユーザの知識発見と質問修正を支援する．しかし，生成される検索式の種類は限られていた．

本研究は，ユーザの操作を柔軟に検索に反映できる点で異なる．

3. 提案システムの概要

本研究では，ユーザが検索結果を一覧的に把握し，ユーザが求めるページの検索および再検索を支援するシステムを提案する．ユーザが検索結果中に含まれる話題を一目見て理解できるよう，ページのリストを表示するのではなく，検索結果ページ中に出現する重要語をグラフ上に視覚的に表示する．ここでは重要語を話題語として定義する．ユーザはグラフ上に表示した話題語を操作することで，質問式の修正および再ランキングを行い，求める情報を探すことになる．

3.1 話題語とは

サーチエンジンの検索結果は，スニペットと呼ばれる結果ページ中の文章を短く要約した文章のリストという形で表示される．検索結果中には様々な話題が存在しており，主要な話題はたくさんのスニペット中で言及されているはずである．しかし，ユーザが主要な話題を見つけるためには，リストから 1 件ずつ検索結果ページのスニペットを読む必要がある．

本研究では，そのようなことをしなくても，ユーザが検索結果集合における話題傾向を再発見することの支援を目的としている．スニペット中に出現する語の中には，検索結果中におけるある話題と強い相関性を示す語がある．例えば,「東西線」というクエリにおける「京都」や「メトロ」といった語がそれにあたる．そのような語を，話題を構成する語であることから話題語として定義し，抽出した話題語の出現相関をグラフとして可視化する．

また，話題語と関連性の強い語を話題語と同時に提示することにより，ユーザはより話題を発見しやすくなると考えられる．そこで本稿では，このような語のことを話題語に共起する語として定義し，可視化を行う．

話題語の抽出方法については，4 章で説明する．

3.2 可視化手法

抽出した話題語およびそれに共起する語は，単純に並べて表示するだけではそれらの語がどのように関連性を持っているのか分からない．複数の語を一覧的に表示する手法として代表的な手法であるタグクラウドは，重要度によって大きさを変えた複数の語を並べて表示する手法であり，flickr [9] などで用いられている．しかし，タグクラウドは語の間の関係は考慮されていないため，ユーザとのインタラクションを行うことが難しい．

そこで本研究では，検索クエリと検索結果に現れる話題語とそれらの共起語について，図 1 のようにクエリを中心とし，ネットワーク構造の形で可視化する．本稿では，この表示方式を相関グラフと呼ぶ．相関グラフ上において語を表すものをノードと呼び，ノード間を結ぶ線をエッジと呼ぶ．

相関グラフ上には，クエリとして入力されたキーワードを赤いノードとして表現し，これをクエリノードと呼ぶ．話題語のノードは青いノードとして表現し，トピックノードと呼ぶ．クエリノードとトピックノードはエッジで結ばれており，場所を移動することが可能である．話題語に共起する語は緑のノードとして相関グラフ上に表示し，トピックノードの周辺に表示する．

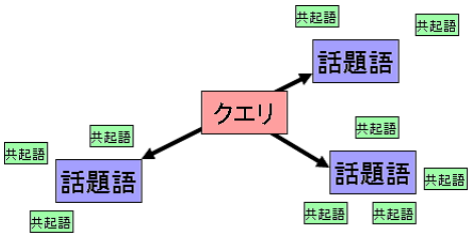

図 1　検索結果の相関グラフによる可視化（イメージ）

3.3 ユーザの操作と検索式の動的生成

従来の検索エンジンで目的のページが上位に表示されるような検索を行う場合，ユーザは複数のキーワード候補を組み合わせたクエリを作成する必要がある．

検索式とは，複数のキーワードを AND, OR, NOT で接続することによって検索の条件付けを行う論理式のことである．例えば,「京都 観光 not 研究」という検索式をクエリとして入力した場合，サーチエンジンは「京都」と「観光」という語を含み,「研究」という語を含まないページのスニペットのリストを返す．

一般的な検索では，多くても 3 つくらいのキーワードを AND でつないだクエリしか用いられておらず，複雑な検索式を用いて検索することはまれである．それは，AND 検索や NOT 検索を行うと，検索結果に表示される結果はキーワードを含むページのみ，もしくは含まないページのみに限定してしまうため，適切なキーワードを選ばなければ理想的な結果が得られないからである．OR 検索を用いることで回避できる点もあるが，適切な検索式を作ることは容易ではない．

本研究では複数のキーワード候補をユーザが操作し，ゆるやかな重み付けにより再ランキングした結果を表示するシステムを提案する．重み付けの値が一定以上大きい，もしくは小さいキーワードに関しては，検索式を修正することにより検索支援を行う．

ユーザは相関グラフにおいて，図 2 のようにクエリノードとトピックノードとの間の距離を変えることにより，システムは AND 検索，NOT 検索を切り替え，次のように検索式の生成および再ランキングを行う．

- 距離が近い場合　→　AND 検索
- 距離が遠い場合　→　NOT 検索
- その中間　→　距離に応じて再ランキング（5.1 節）

相関グラフ上には，AND 検索の対象となるノードへのエッジはピンク色，NOT 検索の対象となるノードへのエッジは水色，それ以外のノードへのエッジは黄緑色で表示する．

クエリが変わると検索結果中に存在する話題傾向も変わってくる．そこで，システムは生成した検索式に対する話題語およびそれに共起する語を再び求め，相関グラフを再描画する．

3.4 システムの動作イメージ

ユーザがシステムを用いて検索する流れは，以下のようになる．

（1） ユーザはシステムにクエリを入力

（2） システムが検索結果を相関グラフとして表示

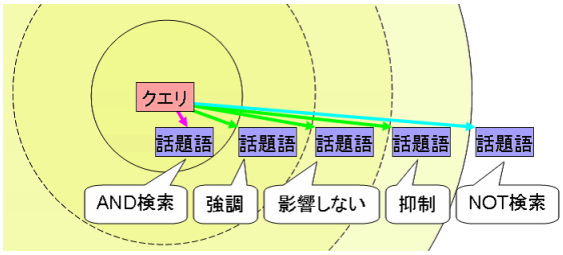

図 2　ノード間の距離と検索式の関係（イメージ）

（3）　ユーザは重視する話題語に応じて相関グラフ上のノードを操作する

（4）　システムはクエリノードとトピックノードの関係により再ランキングを行う

（5）　ユーザは検索結果を見て，求めるページが見つかるまで操作を繰り返す

本システムの実装画面を図 3 に示す．

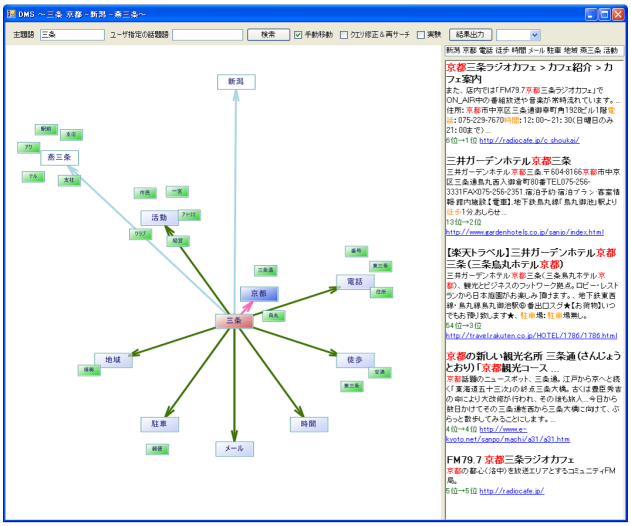

図 3　システム概要図

図 3 は，ユーザが「三条」で検索した場合の相関グラフである．「三条」の話題語である「京都」や「新潟」がトピックノードとして相関グラフ上に表示されていることが分かる．

ユーザは重要視するトピックノードをクエリノードに近づけ，必要のないトピックノードを遠ざけることで質問修正および再ランキングが行われる．図の下部分が検索結果を再ランキングしたものであり，各ページのタイトルやスニペット，URL などが表示されている．

4.　検索結果中の話題語の抽出

本章では，検索結果ページ集合中に出現する重要語を見つけ，それらを話題語として抽出する方法について説明する．

4.1　話題語候補の抽出

話題語を抽出する前処理として，まず話題語の候補となる語をスニペット中から抽出する．

検索を行うサーチエンジンとして Google [10] を用い，検索結果上位 100 件におけるスニペットを形態素解析する．解析結果のうち以下の品詞に該当するものを話題語候補とした．なお，形態素解析には茶筌 [11] を用いた．

- 名詞-一般
- 名詞-固有名詞
- その他，数詞・非自立・接頭語・接尾語以外の名詞
- 未知語のうち，ひらがな・カタカナ・漢字に該当するもの

4.2　話題語候補の選別

1 つのページのスニペットを 1 つのドキュメントとみなすと，Google の検索結果 100 件は 100 個のドキュメントとみなすことができる．そのため，各話題語候補に対して，100 件のスニペット中において，その語の出現するスニペットの数を DF 値として求めることができる．

話題を構成する語が持つ特徴を考えると，まずある程度出現頻度の高い語であるということが挙げられる．しかし，頻出単語を話題語として選択するだけでは「人」や「もの」といった，話題の構成には関与しない一般的な語まで抽出してしまう．

これを回避するため話題語が持つ他の特徴を考えると，クエリに対してその語を追加することで，検索結果中に存在する話題を限定することができる語であると考えることができる．

そこで，本研究では以下に示す方法により話題語を抽出することを試みた．

（1）　n 個の話題語候補それぞれについて，DF 値の上位から順に $T_i(i = 1 \sim n)$ と名前をつけ，各 T_i に対して (2)〜(5) の操作を話題語が 10 個になるまで繰り返す

（2）　100 個のスニペットを話題語候補 T_i を含むものと含まないものに分け，含む集合を正例 P_i，含まない集合を負例 N_i と呼ぶこととする

（3）　正例スニペット中における話題語候補全ての DF 値のリスト $\overrightarrow{\mathrm{df_P}} = \{df_{P_i} \mid \mathrm{i} = 1 \sim \mathrm{n}\}$ と，負例スニペット中における話題語候補全ての DF 値のリスト $\overrightarrow{\mathrm{df_N}} = \{df_{N_i} \mid \mathrm{i} = 1 \sim \mathrm{n}\}$ を求める

（4）　$\overrightarrow{\mathrm{df_P}}$ と $\overrightarrow{\mathrm{df_N}}$ に対し，両ベクトルのコサイン類似度を計算する

（5）　コサイン類似度の値が閾値 $\theta = 0.6$ 以下ならグラフにトピックノードを追加する

コサイン類似度とは，ベクトル X_1 と X_2 に対し，以下のように求められる値 $cos\theta$ のことである．

$$cos\theta = \frac{< X_1, X_2 >}{||X_1||||X_2||}$$

ただし，$< X_1, X_2 >$ は，X_1 と X_2 の内積，$||X_1||$ と $||X_2||$ はそれぞれ X_1 と X_2 のノルム $||x|| = \sqrt{< x, x >}$ である．

4.3　話題語に共起する語の抽出

クエリに対して話題語をキーワードとして追加すると検索結果中における話題傾向は大きく変わるはずである．つまり，検索結果ページのスニペット中に出現する語も大きく変化することになる．

話題語に共起する語とは，クエリ中にその語を加えることで，どのような語が同時に増えるのかを提示し，ユーザがその話題語を重視するべきか否かを決定することを支援するためのものである．話題語に共起する語は以下に示す方法により選択した．

なお，話題語候補として抽出したものと同じものを共起語候補として用いた．

（1） 各話題語に対して，Google の検索結果 100 件のスニペットを話題語を含むものと含まないものに分ける

（2） 話題語を含むスニペット（正例）の総数を n_P，含まないスニペット（負例）の総数を n_N とする

（3） 話題語に共起する語を調べるため，各共起語候補 C_i に対して，正例ページ集合中における C_i の DF 値 df_{PC_i}，負例ページ集合中における C_i の DF 値 df_{NC_i} を求める

（4） n_P，n_N，df_{PC_i}，df_{NC_i} を用いて，共起語候補の出現頻度に応じてカイ 2 乗検定を行う

表 1 正例と負例のスニペット集合中における C_i を含むスニペット数と含まないスニペット数

	C_i を含む	C_i を含まない	合計
話題語を含む	df_{PC_i}	$n_P - df_{PC_i}$	n_P
話題語を含まない	df_{NC_i}	$n_N - df_{PC_i}$	n_N

検定とは，帰無仮説と呼ばれる仮説をたて，その仮説がどの程度正しいといえるのかを統計的に検証する手法である．本稿では，検定の手法には 2 つの二項分布の比較手法 [12] を用いた．

正例における共起語候補 C_i の出現頻度と負例における共起語候補 C_i の出現頻度が等しいと仮定し，カイ 2 乗値を計算した．それが閾値以上ならば，帰無仮説が棄却できる，つまり正例と負例における共起語候補 C_i の出現頻度は異なるということが示せたこととなる．

各共起語候補のカイ 2 乗値 S_i は以下のように求められる．

$$S_i = \frac{n\{df_{PT_i}(n_N - df_{NT_i}) - (n_P - df_{PT_i})df_{NT_i}\}^2}{df_{PT_i}(n_P - df_{PT_i})df_{NT_i}(n_N - df_{NT_i})}$$

分割表において，値が 5 以下になるセルがある場合にはイェーツの補正を行った．

$$S_i = \frac{n\{|df_{PT_i}(n_N - df_{NT_i}) - (n_P - df_{PT_i})df_{NT_i}| - \frac{n}{2}\}^2}{df_{PT_i}(n_P - df_{PT_i})df_{NT_i}(n_N - df_{NT_i})}$$

ただし，$n = n_P + n_N$ とおいた．

4.4 話題語のクラスタリング

検索結果から抽出した話題語は，それぞれが何らかの話題を構成する語であると考えられる．しかし，上述の手法では 1 つの話題に対して 1 つの話題語のみが提示されるというわけではない．そこで，同じ話題に関すると考えられる語をクラスタリングすることを試みた．

各話題語 T とそれに共起する語 $C_i(i = 1〜10)$ について，C_i も話題語であれば同じ話題に関するものとし，相関グラフの描画の際，近い位置に表示するようにした．

5. ユーザ操作による質問修正

相関グラフに表示されたノードをユーザが操作することにより，システムは再ランキングおよびクエリの修正を行う．本章では，それらの手法について説明する．

5.1 再ランキング

本システムでは，リスト中に表示されている検索結果について，ノード間の距離に基づき検索結果ページを再ランキングする．ユーザはクエリノードと各トピックノードの距離を調整することにより再ランキングを行う．

クエリノードと各トピックノード $t_i(i = 1〜n)$ 間の距離を d_i とすると，スニペット s_j が t_i を含むとき $x_{ji} = 1$，含まないとき $x_{ji} = 0$ とおくと，スニペットのスコア S_j を

$$S_j = \sum_{i=1}^{n} \left(\frac{x_{ji}}{d_i} - \theta \right)$$

と定義した．閾値 θ の値には今回は 0.003 という値を用いた．

つまり，話題語 T_i に対するトピックノード t_i をクエリノードの付近に配置すると，T_i を含むスニペットのスコアが上がり，クエリノードから離れた位置に配置するとスコアが下がることになる．

そして，このスコアが大きいものが上位に来るように再ランキングを行う．すなわち，リストに表示された検索結果のうち，クエリノードとの距離が近い話題語を多く含むページが上位に表示される．

図 3 において,「京都」というトピックノードをクエリノードに近づけると京都の「三条」に関する話題が上位に表示される．このとき,「新潟」のノードを遠ざけると，より効果的な再ランキングが行える．

5.2 再ランキングの拡張

検索において検索式を作るとき，いくつかのキーワードを思いつくが，その全てを単純に AND で接続して検索しても検索件数が少なすぎ，かといって下手にキーワードを限定してしまうとノイズが増えてしまうという場合がある．

例えば,「スピード」という映画の DVD に関する情報を探すとき,「スピード DVD」というクエリを入力するとまったく関係のないページが多く表示される．しかし，主演である「キアヌ・リーブス」をクエリに追加すると検索件数は減るものの，欲しかったページも一緒に消えてしまうかもしれない．

他の例として，数人の出演者がいたドラマのタイトルを思い出したいが，何人かの出演者の候補は思い出せるものの，それら全ての人が出演していたかどうかの記憶が曖昧であるといった場合がある．このとき，それらの出演者の組み合わせを全てクエリとして試すとすると，出演者候補が n 人いたとして $2^n - 1$ 通りものクエリを作成する必要がある．

このような場合に対応するため，提案したシステムでは一度検索したスニペットを保存しておき，新しい検索式による検索結果と統合してリストに表示している．そのため，相関グラフ上でノードを操作していくことで，複数のクエリを横断的に検索できる．

5.3 検索式の再生成

ユーザが相関グラフ上のノードを操作すると，システムは検索式を修正し，生成された検索式をクエリとして Google で再検索する．再検索の結果の上位 100 件を，それ以前に検索された結果と統合してリストに表示する．

生成される検索式は，クエリとして入力されたキーワードに対し，図 2 における AND 検索，NOT 検索の位置にある話題語をキーワードとして追加したものである．ただし，NOT 検索の位置にある話題語については NOT 検索を行う．

図 3 を例とすると,「三条」というクエリに対して「京都」が AND として検索式に追加され,「新潟」,「燕三条」が NOT として追加される．その他のトピックノードは検索式の生成には関与せず，再ランキングに用いられる．このようにして,「三条 京都 not 新潟 not 燕三条」という検索式が生成される．

生成した検索式は AND 検索と NOT 検索のみから構成されており，OR 検索を含むクエリは生成されない．しかし，本研究におけるシステムでは検索式を生成して再検索すると同時に，クエリノードとトピックノードの距離に応じて検索結果の再ランキングを行っている．これにより，OR 検索を用いた複雑な検索式を生成せずとも，ユーザがノードを操作することで多様な検索が可能となっている．

再ランキングおよび検索式の再生成に関する操作の概要を図 4 に示す．

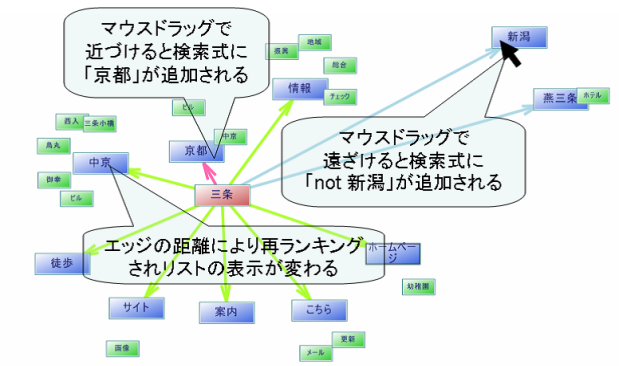

図 4　再ランキングおよび検索式の再生成

6.　検証および考察

提案したシステムの有効性を調べるため，システムを実装し，その有効性を検証した．

6.1　話題語の抽出

抽出した話題語が適切であるかどうかを検証するため，クエリに対して抽出した話題語とそれに共起する語が適切であるかどうかを調べた．比較対象とするため，DF 値のみを用いて話題語を抽出した場合の例とともに，表 2 に示す．

例：クエリ「東西線」

両手法で抽出された話題語を比較すると，提案手法では「地下鉄」,「検索」,「沿線」という語が話題語として抽出されていないことが分かる.「地下鉄」が抽出されないのは「東京」,「京都」,「札幌」,「仙台」の東西線が「地下鉄」であり,「地下鉄」という語だけではうまく話題を分離することができないためであると考えられる．また，他の 2 つに関しては,「検索」は一般的な語であるため,「沿線」というキーワードは話題を構成していないためであると思われる．

提案手法では，話題語とみなせる語は多く抽出できたが,「情報」といった一目ではどのような話題のページが検索されるか分からない語も含まれている.「情報」が話題語として抽出された理由は,「住宅情報」という形で情報という語を用いているページが多いため，正例と負例のコサイン類似度が低くなるためであると考えられる．

表 2　「東西線」の話題語抽出

提案手法		DF 値のみ	
話題語	コサイン類似度	話題語	コサイン類似度
東京	0.552	東京	0.552
メトロ	0.564	メトロ	0.564
情報	0.550	地下鉄	0.618
不動産	0.542	情報	0.550
マンション	0.493	検索	0.622
路線	0.523	沿線	0.634
アパート	0.489	不動産	0.542
住宅	0.509	マンション	0.493
市営	0.567	路線	0.523
物件	0.573	アパート	0.489

表 3　「東西線」の話題語に共起する語

話題語	話題語に共起する語
東京	メトロ
メトロ	東京
情報	アパート 検索 土地 不動産 マンション
不動産	アパート 住宅 情報 土地 マンション
マンション	アパート 住宅 情報 土地 不動産
路線	駅名 時刻
アパート	一戸建て 住宅 情報 土地 不動産
住宅	アパート 一戸建て おまかせ 土地 マンション
市営	札幌
駅名	御池 烏丸 クリック 時刻 選択

「東西線」というクエリに対して抽出された話題語は,「不動産」や「マンション」といった住宅情報に関係する語が多かった．それら住宅情報に関する語は，互いに共起しあっていることが表 3 から読み取れる．

しかし，本来話題語として相関グラフ上に表示すべきである「京都」や「仙台」という語は話題語やそれに共起する語として抽出することができなかった．それは，1 つの話題に対していくつもの話題語が抽出されてしまうためであると考えられる．

6.2　ユーザ操作によるフィードバック

ユーザが「京都」の「東西線」に関する情報を求めて「東西線」というクエリで検索したと想定し，相関グラフ上のノードの位置を操作することにより求めるページを検索することを試みた．

図 5 はクエリ「東西線」で検索した結果である.「京都」という話題語は含まれていないが,「東京」という話題語は求めているページとは異なると考えられるので，ユーザはクエリノードから「東京」を表すトピックノードを遠ざける．すると，図 6 のような相関グラフとなる．

図 6 のように，話題語として「京都」を表すトピックノードが表示される．このノードを近づけることにより，ユーザは

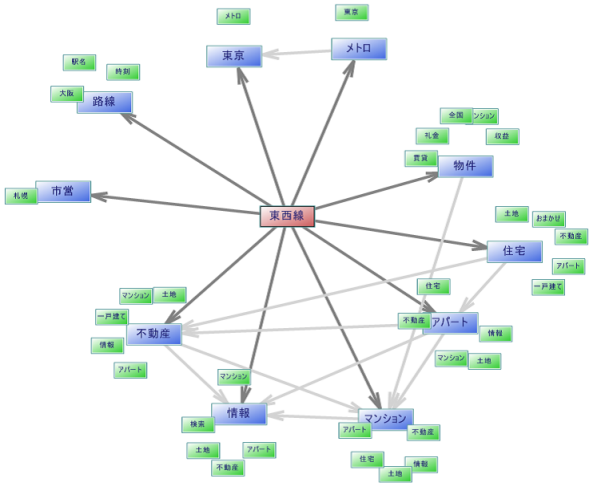

図 5 「東西線」の相関グラフ

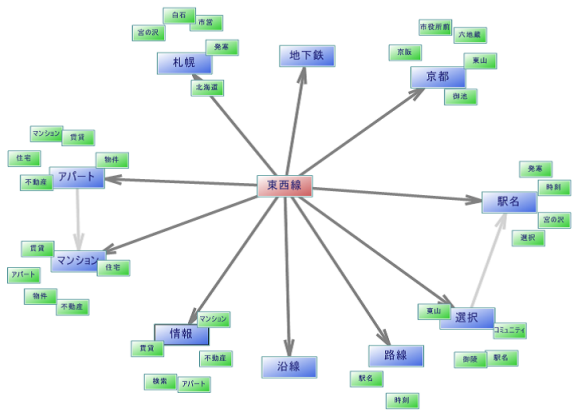

図 6 「東西線 not 東京」の相関グラフ

「京都」の「東西線」に関するページを見つけることができる．

「東西線 not 東京」というクエリにおいて，話題語「京都」について重み付けを行い再ランキングを行った結果上位 10 件のタイトルを図 7 に示す．

京都市営地下鉄東西線とは - はてなダイアリー
さあ でかけよう！ 鉄道からさがす(京都市交通局地下鉄東西線)
近畿(京都地下鉄東西線)の不動産投資物件 - 投資HOME'S
時刻表 京都市営地下鉄東西線 - OCN 路線
京都地下鉄東西線のマンション、アパート、一戸建て、土地、店舗、事務 ...
京都地下鉄東西線沿線の家賃相場情報／家賃のことならHOME'S家賃相場
京都市営地下鉄東西線 時刻表 | エキサイト乗り換え案内
えきから時刻表 [京都市営]東西線(六地蔵〜二条) [駅名]
京都市交通局 - 東西線 - 駅から探す - 京都府 - 飲食店情報 - Yahoo ...
地下鉄東西線の賃貸マンション、アパート、貸家、店舗、事務所探し ...

図 7 再ランキング結果

7. 検索結果の比較への応用

相関グラフによる可視化は，複数のクエリによる検索結果を同時に比較することに応用できる．本章ではその手法について述べる．

7.1 複数の対象を含むクエリ

複数の対象を含むクエリとは，「京都 大阪 観光」といったような，メインとなるキーワードを 2 つ以上含むクエリのことである．ユーザがこのようなクエリを入力した場合，ユーザの意図には次のような場合があると考えられる．

- 2 つ以上のものに共通する話題を探したい場合

例 東京 大阪 ラーメン屋（東京にも大阪にも店を持つラーメン屋を探したい）

- 2 つ以上のものを比較したい場合

例 Radeon GeForce（グラフィックボードを比較したい）

- 2 つ以上のものを同時に含む上位概念を探したい場合

例 レオナルド・ディカプリオ ケイト・ウィンスレット（両俳優が出演した作品を知りたい）

上に挙げた例は完全に分類できるものではない．例えば，「東京 大阪 ラーメン屋」というクエリを入力したユーザは「東京と大阪のラーメン屋の傾向を比較したい」のかもしれない．このクエリからだけでは，どのような意図でユーザがクエリを入力したのかを知ることはできない．

複数の対象を含むクエリを入力したユーザは，それらに関する共通点や相違点を検索したいのだと考えることができる．そこで，ユーザが明示的に複数の対象を相関グラフ上で比較できるようにするため，システムを拡張した．

7.2 共通の話題および個別の話題の発見

拡張したシステムでは，複数のクエリノードを 1 つの相関グラフ上に表示することができる．複数のクエリノードが同じ話題語を持つ場合，相関グラフ上には 1 つのトピックノードしか表示されず，複数のクエリノードがトピックノードを共有する．そして，相関グラフ上において，複数クエリに共通する話題語はそれらのクエリノードの中間に表示され，共通していない話題語はそれぞれのクエリノードの外側に表示される．このような表示をすることで，ユーザは複数のクエリに共通している話題語と，各クエリが独自に持っている話題語を一覧的に見ることができる．

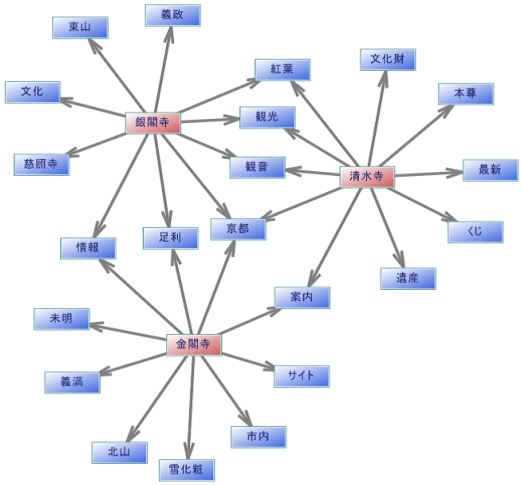

図 8 クエリごとの話題

図 8 は，「清水寺」，「金閣寺」，「銀閣寺」という 3 つのクエリ

ノードを同時に相関グラフ上に表示した例を示している．この例では,「金閣寺」と「銀閣寺」の共通の話題語である「足利」が両トピックノードの中間に，3 つのクエリノードの共通の話題語である「京都」は中心に表示されている．また，個別の話題である「北山」や「東山」といったトピックノードが，それぞれ「金閣寺」と「銀閣寺」の個別の話題語として表示されている．一方,「サイト」や「最新」といった話題を構成しない話題語も表示されている．話題語の抽出方法を改善していくことは今後の課題である．

8. まとめと今後の課題

本研究では，検索結果の可視化およびユーザ操作による質問修正を行った．本章では，その有効性について考察し，今後の課題について述べる．

8.1 候補語の選択

一般的にユーザがサーチエンジンで検索するクエリ中のキーワードは，ほとんどが名詞である．そこで，今回のシステムでも話題語として名詞のみを抽出した．

しかし，ユーザの検索意図として，ある商品の評判を知りたいといった場合もある．この場合には形容詞も話題語となりうる．また，観光地をクエリとして調べている場合などは，観光客がどこで何をしているのかを示すため，動詞を話題語として採用することも有効であると考えられる．

名詞以外の品詞を話題語とすると，それらの語は出現頻度が低いと考えられるので，話題語として採用するためには今回の手法とは異なる手法が必要であると考えられる．

8.2 検索結果の可視化

従来のサーチエンジンでは，自動的ないし人手によってつけられたページの説明をリストとして表示することが一般的であったが，相関グラフとして検索結果中に出現する話題語を提示することによって，検索結果がどのような話題を含んでいるのかを一覧的に見ることができた．話題語をそれに共起する語に応じて分類したことで，クエリが持つ曖昧性をユーザに指摘し，質問修正を行う支援にもつながった．

話題語を提示する上で重要なことは，できる限り全ての話題をユーザに提示することである．しかし，本稿におけるシステムではコサイン類似度を用いて話題を構成する語を抽出することを試みてはいるが，DF 値が話題語の選別に大きく作用している．結果として出現頻度の高い話題が優先的に表示されてしまい，頻度の低い語は表示されないという問題点があり，更なる改良を行う必要がある．

8.3 ユーザ操作によるフィードバック

本システムでは，ユーザが相関グラフを操作することにより，対話的な検索支援を行うことを可能とした．本提案では，ユーザは検索式を組み立てる必要がないため，複雑な検索式を必要とする検索において，サーチエンジンの使い方に慣れていない人でも，必要としているページを発見できるようになると期待できる．

今後は，より直感的に操作可能なシステムを実現するため，ユーザには必要な部分のみを提示し，必要に応じて詳細を提示していくといった手法を取り入れていく予定である．

謝辞 本研究の一部は，文部科学省 21 世紀 COE 拠点形成プログラム「知識社会基盤構築のための情報学拠 点形成」（リーダー：田中克己，平成 14〜18 年度），文部科学省研究委託事業「知的資産の電子的な保存・活用を支援するソフトウ ェア技術基盤の構築」，異メディア・アーカイブの横断的検索・統合ソフトウェア開発（研究代表者：田中克己），文部科学省科学研究費補助金特定領域研究「情報爆発時代に向けた新しい IT 基盤技術の研究」，計画研究「情報爆発時代に対応するコンテンツ融合と操作環境融合に関する研究」(研究代表者：田中克己, A01-00-02, 課題番号 18049041），文部科学省科学研究費補助金特定領域研究「情報爆発時代に向けた新しい IT 基 盤技術の研究」，計画研究「情報爆発に対応する新 IT 基盤研究支援プラットフォームの構築」（研究代表者：安達淳，Y00-01，課題番号：18049073），および文部科学省科学研究費補助金若手研究 (B)「参照の同一性判定に基づく複数 Web ページの検索閲覧方式の研究」（研究代表者：小山聡，課題番号：16700097）によるものです．ここに記して謝意を表するものとします．

文　献

[1] Google サジェスト. http://www.google.co.jp/webhp?complete=1&hl=ja.
[2] Yahoo! 関連検索ワード. http://search.yahoo.co.jp/.
[3] C. CHEN: “Information Visualization”, Splinger (2004).
[4] 大澤幸生, N. E. Benson, 谷内田正彦：“KeyGraph：語の共起グラフの分割・統合によるキーワード抽出”, 電子情報通信学会論文誌 J82-D-1, No.2, pp. 391–400 (1999).
[5] 塩澤秀和, 西山晴彦, 松下温：“「納豆ビュー」の対話的な情報視覚化における位置づけ”, 情報処理学会論文誌, Vol. 38, No. 11, pp. 2331–2342 (1997).
[6] Yahoo! mindset. http://mindset.research.yahoo.com/.
[7] 121r(one to one ranking system). http://www.kbmj.com/service/products/121r.html.
[8] 松生泰典, 是津耕司, 小山聡, 田中克己：“検索結果の概要を表すキーワード式生成による質問修正支援”, 電子情報通信学会第 16 回データ工学ワークショップ (DEWS2005) 1C-i9 (2005).
[9] flickr. http://www.flickr.com/.
[10] Google. http://www.google.com.
[11] 形態素解析システム茶筌. http://chasen.naist.jp/hiki/ChaSen/.
[12] 東京大学教養学部統計学教室：“自然科学の統計学”, 東京大学出版会 (1992).