ContextRank: 語ベースフィードバックおよびそのコンテキストに基づく検索結果の再ランキング手法

山本　岳洋† 　中村　聡史† 　田中　克己†

† 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: †{tyamamot,nakamura,tanaka}@dl.kuis.kyoto-u.ac.jp

あらまし　本稿では，ユーザからの語ベースフィードバックおよびそのコンテキストに基づき検索結果を再ランキングする手法を提案する．我々はこれまでにユーザからの語ベースフィードバックの情報のみを利用して検索結果を再ランキングする手法を提案してきた．本稿では，ユーザの再ランキングの意図を，検索結果を様々な側面から閲覧しようとする散策型の意図，検索結果をある特定の話題に絞り込もうとする適合性重視型の意図の 2 種類に分けて考える．ユーザの語ベースフィードバックからそうした意図を推測し，再ランキングの意図に合わせた情報の提示やランキングを行う事で，より効率良くユーザの情報検索を支援できると考えられる．本研究では特に適合性重視の意図に基づく再ランキングの支援に注力し，語ベースフィードバックとそのコンテキストを利用した再ランキングアルゴリズム，*ContextRank* を提案する．評価実験の結果，*ContextRank* は少ないフィードバックで高い精度を達成できることが明らかになった．

キーワード　情報検索，語ベース適合フィードバック，検索意図，ユーザインタラクション

1. はじめに

近年，多くの人々が検索エンジンを利用して必要な情報を取得している．しかし，ユーザの求める検索結果はユーザの興味や嗜好，検索意図などによって異なり，同じユーザが同じクエリを入力したとしても，その時の検索タスクによって求める検索結果は同じであるとは限らず多岐にわたる．また，検索エンジンが返した検索結果を実際に閲覧していくことでも，ユーザの検索意図は移り変わっていくと考えられる．このように，ユーザの検索意図は多様で移り変わりやすいが，ユーザにとって自らの検索意図を表すクエリを作成することは非常に困難である [1] [2] [3]．そのため，現状の検索エンジンではユーザの検索意図を十分に反映した検索結果をユーザに提示することは難しい．

この問題に対して，我々はこれまでに，ユーザがウェブ検索結果閲覧中に，検索結果中の語に対してインタラクションを行うことにより，ユーザの検索意図をシステムに伝達する手法を提案してきた [4] [5]．ユーザは検索結果中の特定の箇所に対して削除操作を行うことにより「検索結果のこの語がいらない」，強調操作を行うことにより「この語を含むような検索結果をもっと上位に表示して欲しい」といった意図をシステムに伝えることができる．また，システムもその意図をくみとり，ある程度ユーザの意図に沿うよう検索結果を再ランキングすることができていた．一方，我々がこれまで提案してきたシステムは，適合フィードバックシステムの観点から見ると，語ベースの適合フィードバックシステムとして捉えることができる．つまり，ある語に対する適合・不適合をフィードバックとして受け取り，それに基づいて検索結果の再ランキングを行うというものである．既存の語ベースの適合フィードバックシステムは，システムが検索結果とは別途に提示した多数の語に対して，ユーザがチェックをつけていくといった間接的なインタラクションに基づくものが多い [6]．それに対して，我々のシステムではユーザは実際に検索結果を閲覧しながら，検索結果中の語に対して直接インタラクションを行うことができるというものであった．

これまで，我々のシステムはユーザの単語単位のフィードバックを単に “その単語に興味が有るか無いか” という単純な意図として扱っていた．しかし，単語に対する興味の有無の背後には，より複雑なユーザの再ランキングの意図が潜んでいる．例えば，あるユーザが “京都　観光” というクエリで検索し，その検索結果中で，“金閣寺” を強調→ “銀閣寺” を強調→ “清水寺” を強調，といった一連の再ランキングを行ったとする．この際のユーザの再ランキングの意図は，“京都の観光地に関する情報をどんどん閲覧していきたい” というものを考えることができる．それに対して，あるユーザが “山本岳洋” というクエリで検索し，その検索結果から “消防士” を削除した後 “京都大学” を強調するといった再ランキングを行ったとする．この場合は，“京都大学の山本岳洋さんに関する情報を集めたい” という再ランキングの意図が考えられる．

前者の再ランキングの意図は，“検索結果を様々な側面で閲覧したい” という**散策的な意図**であると考えられる．一方，後者の意図は “特定の話題に関する検索結果だけを閲覧したい” という**適合性を重視した意図**であると考えられる．本稿ではまず，ユーザの再ランキングの意図をこのような 2 つの性質から考え，語ベースフィードバックからどのようにユーザの意図を推測できるのかについて検討する．ユーザのこうした意図を考慮する事で，よりユーザの求める情報を効率良く提示できると考えられる．

散策型の意図に基づく再ランキングについて考えた場合，我々

がこれまで提案してきたシステムは，タグクラウドによる単語集合の提示や単語の出現情報みの検索結果を再ランキングする手法である程度支援できていると考えられる．一方で，適合性重視の意図に基づく再ランキングの支援を考えた際に，我々が現在行っている単語の出現情報のみに基づく再ランキングではユーザの求めるランキングを提示することは難しい．

そこで，本稿では特に適合性重視の再ランキングに着目し，その意図を効率よく反映するためのランキング手法，*ContextRank* の提案を行う．*ContextRank* は単語のフィードバック情報だけではなく，そのフィードバックのコンテキストを考慮する事によって，より適合性の高いランキングを行う事を目的としている．

以降，2 章では 2 種類の再ランキングの意図ついて述べる．3 章では *ContextRank* について説明し，4 章で手法の評価を行う．5 章で考察を行い，6 章で関連研究を述べた後，全体のまとめを行う．

2. 語ベースフィードバックによる再ランキングの意図

2.1 語ベースフィードバックに基づく再ランキング

我々はこれまでに，ユーザが検索結果に対して直接インタラクションを行うことによって検索結果の再ランキングを行うシステムを提案してきた [4] [5]．このシステムでは，ユーザは強調・削除操作という 2 種類の操作を通じて語ベースのフィードバックを行うことにより，意図に応じて検索結果を再ランキングできる．図 1 に実際にシステム(注1)の動作例を示す．システムはユーザからクエリを受け取ると，検索結果および検索結果中に出現する語の頻度に基づいたタグクラウドをユーザに提示する．ユーザは実際に検索結果を閲覧しながら，検索結果中の任意の語を選択やクリック等で直接指定することで，強調・削除を行う．システムは，その情報を語ベースのフィードバックとして受け取り，検索結果の再ランキングを行った後ユーザに再度提示する．この仕組みにより，ユーザは再ランキング操作を通じて検索結果を自らの興味に合わせて最適化していくことが可能となる．

2.2 再ランキングの意図

ユーザの再ランキングの意図は大きく分けて**散策型** と**適合性重視型**という 2 つの性質に分けて考えることができる．

散策型： 1 章で述べたように，京都の観光に行きたいが，どこに行くのか具体的に決まっていないため，とりあえずいろいろな観光地に関する情報を閲覧したいという場合，ユーザは興味を持った観光地に関する単語で強調を繰り返し，様々な観光地に関する情報を把握するだろう．また，豚肉とピーマンを使った料理のレシピが欲しいが，実際にどんな料理を作るのかを具体化していないユーザは，“豚肉　ピーマン” というクエリで検索を行い，検索結果に出現するいろいろな食材名に対して削除や強調を行いレシピを次々に探していく．このように，散策型の意図に基づく再ランキングは，ユーザの情報欲求が具体化されておらず曖昧なキーワードでしか検索意図を表現できない場合に用いられると考えられる．ユーザは散策型の再ランキングを繰り返すことによって，様々な検索結果を閲覧し自らの情報欲求を具体化させていく．従って，散策型の意図に基づく再ランキングは，検索結果の適合性よりもむしろユーザが興味を持ちそうな話題を次々と発見・閲覧していくことの方が重視される．

図 1　ユーザインタラクションに基づく検索結果の再ランキングシステム

適合性重視型： 例えば，1 章で述べたように，ある特定の人物に関する情報を集めたいと思っているユーザが，その人物名をクエリとしてシステムに入力し，検索結果を閲覧する場合を考える．もしもユーザが検索結果の精度に満足できない場合，ユーザは求めている人物とは関係のない単語を削除したり，目的の人物に関係が深いであろう単語を強調していくだろう．その際のユーザの再ランキングの意図は，ある人名のクエリの検索結果集合を，特定の人物の検索結果だけに絞り込みたいという意図であるといえる．このように，検索結果をある特定の集合に絞り込んでいくような再ランキングは，自らの情報欲求に適合する検索結果のみを閲覧したいという，適合性重視の再ランキングである．

散策型，適合性重視型の性質を図で表すと図 2 のようになる．ユーザがあるクエリ q で検索した検索結果を閲覧しているとする．ここでは簡単のため，検索結果がA，B，C という 3 つの話題に分けられるとする．散策型の意図に基づいた再ランキングは検索結果を様々な側面から閲覧しようとする．従って，図 2 のように，様々な話題に関する検索結果を閲覧するような再ランキングを繰り返しながら検索結果を閲覧していくと考えられる，一方，適合性重視型の意図を持ったユーザが，話題Aに関する検索結果だけを閲覧したいと考えた場合，図 2 のように，検索結果を特定の話題に関するものだけに絞り込もうとするような再ランキングを繰り返していくと考えられる．

一般的に，ユーザはこの散策型と適合性重視型の意図を行き来しながら自らの情報欲求を明確化し，時には変化させながら，最終的に自らの求める情報を得ようとしていくと考えられる．例えば，あるユーザが “京都　観光” というクエリにおいて，いろいろな観光地に関して情報を閲覧するために，“清水寺” を強調→ “銀閣寺” を強調→ “金閣寺” を強調，という一連のイン

(注1)：Rerank.jp, http://rerank.jp

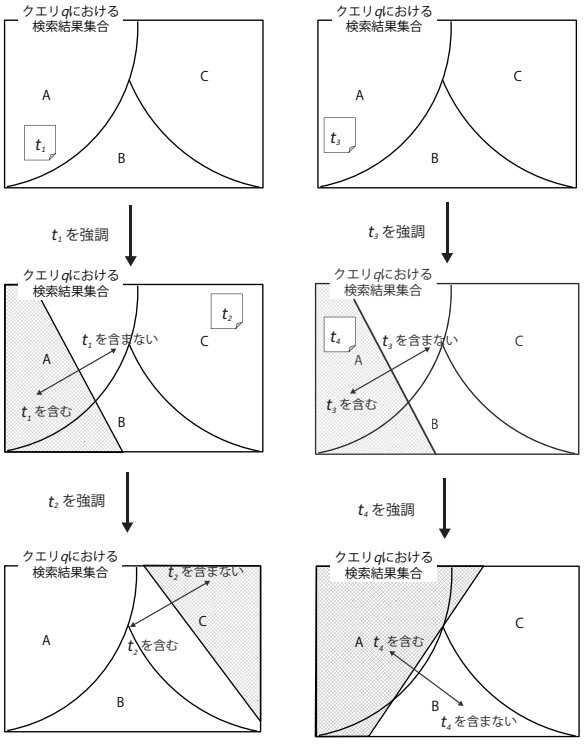

図 2　2種類の再ランキングの意図に基づく再ランキング例

タラクションを行ったとする．この時，そのユーザは散策型の意図に基づいて再ランキングを行っていると考えられる．ここで，ユーザが “金閣寺” に関する情報に興味を惹かれ，もっと金閣寺に関する情報を知りたいと考えたとする．すると，ユーザは金閣寺の検索結果中に現れる “銀閣寺” を削除したり，“鹿苑寺” を強調したりする．この様な再ランキングは “金閣寺に関する情報だけを閲覧したい” という適合性重視の意図に基づく再ランキングと捉えることができる．

2.3　語ベースフィードバックからの再ランキングの意図推定

ユーザの求める情報を効率よく提示するためには，前節で述べた再ランキングの意図をうまく捉え，それぞれの意図に基づいたランキングや情報の提示を行っていくことが極めて重要である．

ここで，ユーザが行った語ベースフィードバック１つのみからユーザの意図を推定することを考えてみる．しかし，“京都観光” において “鹿苑寺” を強調した，という情報のみからでは，ユーザは散策型の再ランキングを行っているのか，適合性重視型の再ランキングを行っているのかを判断することは難しい．このように，**１つの語ベースフィードバックのみからではユーザの再ランキングの意図を推定することは困難である．**

しかし，前節のように，**ユーザがどのような流れで再ランキングを行ったのか**を考え，一連の語ベースフィードバック間の関連性を考えることで，ユーザの再ランキングの意図を推定することが可能であると考えられる．例えば，“山本岳洋” といった人名のクエリにおいて，“消防士” と “京都大学” という２つの単語の関連について考えてみる．“消防士” と “京都大学” が “山本岳洋” の検索結果内で同時に出現する確率は極めて少ない．このような情報を用いると，“消防士” を削除し，“京都大学” を強調した場合は，京都大学学生であるような “山本岳洋” に関する検索結果だけが欲しいという適合性重視型の意図が妥当である．それに対して，別のユーザが “消防士” を強調し，“京都大学” を強調した場合，そのユーザの再ランキングの意図は，“山本岳洋という名前を持つ人物にはどういった人がいるのだろう” という，散策型の意図に基づいていると考えられる．

このように，一連の再ランキングにおいてどのような単語を削除・強調してきたのかという情報は，ユーザの再ランキングの意図を判断する上で重要な情報である．

2.4　再ランキングの意図反映

散策型の意図に基づく再ランキングは，検索結果を様々な側面から見てみたいという欲求に基づいている．従って，そうした意図を反映するには，例えば検索結果のクラスタリングや，分散性を考慮したランキング [7]，関連のある語の推薦といったアプローチが考えられる．散策型の意図を反映する手法のひとつとして，我々のシステムでは検索結果中に出現する頻度の高い語をタグクラウド形式で提示している．タグクラウドにより，クエリに関する話題を俯瞰的に閲覧することができ，タグクラウド中の単語を次々と強調・削除していくことで様々な話題を効率よく閲覧することが可能となっている．我々のシステムにおいて，ユーザがタグクラウド中の語を用いて再ランキングを行う割合は，全体の再ランキングの７割を占めている．このことからも，タグクラウドが散策型の再ランキングにある程度役立っていることが分かる．

一方で，適合性重視の意図に基づく再ランキングについて考えた場合，タグクラウドのような様々な語の提示や，クラスタリングといった検索結果の提示方法よりもむしろ，検索結果のランキングそのものがどれだけユーザの検索意図に適合しているかの方が重要であると考えられる．しかし，現在のシステムは単純に検索結果がユーザの指定した単語を含むか含まないかという情報だけで再ランキングを行っている．そのため，ユーザが求めるような適合性の高いランキングを達成するにはインタラクションを多数回行う必要がある．例えば，我々のシステムにおける再ランキングのログでは，人名で検索したユーザが自分とは関係の無いであろう単語をいくつも削除しているという記録が多く見受けられる．このことからも，単語の出現情報のみに基づく再ランキングでは，ユーザの求める適合性の高いランキングを実現することは難しいことが分かる．

従って，本稿では，適合性重視の意図を反映するためのランキングアルゴリズムについて検討を行う．

3.　ContextRank

本章では，適合性重視型の意図に基づく再ランキングのための検索結果の再ランキング手法，*ContextRank* について述べる．２章で述べたように，ユーザの再ランキングの意図は，１つの語ベースフィードバックのみから推測することは難しい．そのため，本稿ではユーザが２回の語ベースフィードバックを行い，システムがそのユーザの再ランキングの意図を適合性重視型であると判断したものと仮定して進めていく．

3.1 基本的なアイデア

これまでのシステムでは，ユーザがフィードバックとして指定した 1 つの単語の出現情報のみで検索結果の再ランキングを行っていた．そのため，ユーザが検索結果をある目的の話題に関するものに十分に絞り込むには，多数のインタラクションを行う必要があった．

しかし，ユーザが適合性重視の意図に基づき，ある特定の話題に関する検索結果だけを求めているという再ランキングの意図が分かれば，単純な単語以外の情報を再ランキングに用いることができる．本稿では，ユーザがどのような検索結果上で強調・削除操作を行ったのかというフィードバックのコンテキストに注目し，再ランキングのアルゴリズムに組み込む．また，フィードバックやコンテキストから得られた情報を元に，検索結果の適合性を評価する際に，検索結果間の類似度を再帰的に計算することで最終的な値を求める．これは，単純な類似度だけで再ランキングを行うと，ユーザの求める話題と関係の強い単語も関係の弱い単語も同様のスコアを割り当てられ，ランキングの精度低下に繋がるのではないかと考えられるからである．つまり，

- 適合する確率が高い検索結果と類似する検索結果は適合する確率が高い

という検索結果間の類似性を再帰的に評価することによって，よりユーザの求める検索結果を提示できるのではないかという仮定に基づいている．

まず，*ContextRank* は検索結果集合をノードと見なし，検索結果同士を内容の類似度に基づきエッジを張る．また，構築されたグラフ構造に対して，

- 語ベースフィードバックを考慮した遷移行列の修正
- フィードバックのコンテキストから得られる検索結果単位の重要度のグラフ構造を通じた伝播

の 2 つを実行し，検索結果集合に対して PageRank アルゴリズムを適用し，ユーザが求めているであろう検索結果の適合性を確率的に求める．

以降，まず PageRank アルゴリズムについて簡単に説明し，その後 *ContextRank* の具体的な手法について述べる．

3.2 PageRank アルゴリズム

PageRank [8] は，ある Web ページから別の Web ページに張られたリンクをページに対する支持とみなし，多くのページからリンクされているページは重要であるというアイデアのもとで各ページの重要度を確率的に評価するための手法である．ページ間の遷移行列を $\boldsymbol{T}$，全ページ数を n とすると，各ページの PageRank 値 $\boldsymbol{PR}$ は以下の数式で与えられる．

$$\boldsymbol{PR} = \alpha \cdot \boldsymbol{T} \times \boldsymbol{PR} + (1-\alpha) \cdot \boldsymbol{p},\ \ where\ \ \boldsymbol{p} = [\frac{1}{n}]_{n \times 1} \quad (1)$$

ここで，α は damping factor，$\boldsymbol{p}$ はある仮想的なユーザがページからのリンクを辿らない場合に，どの程度の確率で他のページにジャンプするかを表現した damping vector である．$\boldsymbol{p}$ の要素は非負値であり，その総和は 1 である．

通常の PageRank はランダムサーファーモデルを仮定しており，ユーザはある一定の等しい確率で他のページへとジャンプする．それに対して，この $\boldsymbol{p}$ の要素に対して一様でない確率を与えることによって，特定のページに対する重要度を恣意的に与え，その値をグラフ構造を通して各ページへと伝播させることができる．この考えを用いた研究として TrustRank [9] や topic-sensitive PageRank [10] がある．

3.3 ContextRank の流れ

まず，ユーザはクエリ q で検索を行う．システムは q に対する検索結果集合 R をユーザに提示する．その後，ユーザが提示された検索結果に対して 2 回の語ベースフィードバック $\{f_1, f_2\}$ をシステムに対して与える．

ContextRank による検索結果の再ランキングの流れは以下の通り．

（1） 検索結果集合 R からグラフ構造を構築し，検索結果間の類似度行列 $\boldsymbol{S}$ を計算

（2） 語ベースフィードバック f_1, f_2 に基づき類似度行列 $\boldsymbol{S}$ を修正し，検索結果間の遷移確率行列 $\boldsymbol{S^*}$ を計算

（3） f_1, f_2 に基づき damping vector $\boldsymbol{p}$ を計算

（4） （2），（3）で得られた $\boldsymbol{S^*}$，$\boldsymbol{p}$ を用いて下記の式に基づき検索結果の重要度 $\boldsymbol{CR}$ を計算

$$\boldsymbol{CR} = \alpha \cdot \boldsymbol{S^*} \times \boldsymbol{CR} + (1-\alpha) \cdot \boldsymbol{p} \quad (2)$$

（5） $\boldsymbol{CR}$の値の降順に検索結果 R を並び替える

（6） ユーザに再ランキングされた検索結果 R を提示

以降，主に（1），（2），（3）について説明する．

3.4 類似度に基づくグラフ構造の構築

まず，*ContextRank* は検索結果集合 $R = \{r_1, r_2, \ldots, r_n | r_i$ は i 番目の検索結果 $\}$ それぞれの要素をノードとみなし，各ノード間に仮想的なエッジを張ることによってグラフ構造を構築する．エッジの重みとしては，検索結果間の類似度を用いる．すなわち，検索結果間の類似度行列を$\boldsymbol{S}$とすると，$\boldsymbol{S}$は以下の式で表される．

$$S(i,j) = sim(r_j, r_i);$$

ここで，$sim(r_i, r_j)$ は 2 つの検索結果間の類似度を計算するための関数である．本論文では類似度関数として，検索結果のタイトルおよびスニペットから生成される，$tf \cdot idf$ で重み付けされた特徴ベクトル同士のコサイン類似度を用いた．なお，図 3 に示すように，ある検索結果 r_i から r_j へのエッジの重みと r_j から r_i へのエッジの重みは等しく，類似度行列$\boldsymbol{S}$には対称性が存在する．

次に，このようなグラフ構造に対して検索結果間の遷移確率を考える．検索結果間の遷移確率は，類似度行列$\boldsymbol{S}$を列ベクトルごとに総和が 1 となるように正規化した行列として得られる．直感的には，この時点での遷移行列を用いて式（1）を適用すると，検索結果集合内で典型的な内容を持つ検索結果が高い PageRank 値を獲得する．

3.5 語フィードバックによる遷移確率の修正

前節で作成した遷移行列は類似度のみに基づき遷移確率を決定していた．これに加え，検索結果間の遷移確率を考える際に，ユーザが強調・削除した単語の重要度を考慮する．検索結果間

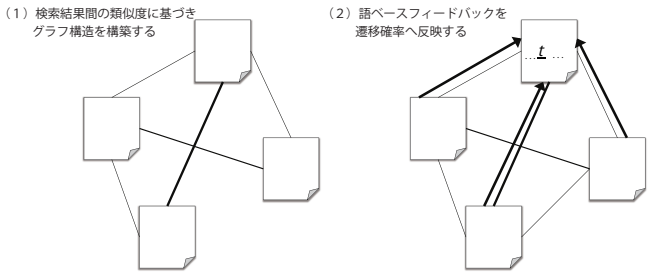

図 3　検索結果間の類似度に基づくグラフ構造の構築と，語ベースフィードバックに基づく遷移確率の修正

の遷移確率にバイアスを加えることによって，単語の重要度を PageRank アルゴリズムに反映させることができる．例えば，ユーザがある単語 t を強調したとする．その際に，ある検索結果から，他の検索結果への遷移確率を，t を含む検索結果へ優先的に辿るように修正することで，ユーザが強調した単語は重要であるということを表現することができる（図 3）．

ここで，ある語ベースフィードバック f でユーザが指定した単語を f_t，強調か削除かを f_o として表す．2 回の語ベースフィードバック $f \in \{f_1, f_2\}$ それぞれについて，遷移行列 $\boldsymbol{S}$ を次式に従って修正する．

$$S(i,j) = \begin{cases} \beta \cdot sim(j,i) & r_i\text{が } f_t\text{を含み，かつ } f_o = \text{強調} \\ \frac{1}{\beta} \cdot sim(j,i) & r_i\text{が } f_t\text{を含み，かつ } f_o = \text{削除} \\ sim(j,i) & otherwise \end{cases}$$

ここで，$\beta(\beta > 0)$ はユーザが強調・削除した単語の重要度を決める定数である．その後，*ContextRank* は修正された類似度行列$\boldsymbol{S}$を各列成分について総和が 1 となるように正規化した遷移行列 $\boldsymbol{S}^*$ を求める．

3.6　フィードバックのコンテキストに基づく検索結果単位の重要度反映

前節では，単語に対するユーザの興味を遷移確率に反映することにより，ユーザの単語単位の重要度を PageRank アルゴリズムに組み込んだ．しかし，フィードバックのコンテキストを考慮すると，ユーザが削除・強調した単語そのものだけではなく，どのような検索結果がユーザにとって重要なのかということを推測することができる．そうした検索結果単位の重要度を damping vector へと組み込むことで，よりユーザの意図を PageRank アルゴリズムに反映することができる．

本稿では，語ベースフィードバックの種類が強調か削除かに応じて，それぞれどのような検索結果がユーザにとって重要であるのかを考える．

3.6.1　強調の場合

あるユーザが検索結果 r_j 中で強調操作を行ったとする．この場合，そのユーザにとって r_j は他の検索結果よりも重要，つまり適合している確率が高いと考えられる．従って damping vector $\boldsymbol{p}$ を以下のように作成する

$$p(i) = \begin{cases} 0 & (i \neq j) \\ 1 & (i = j) \end{cases}$$

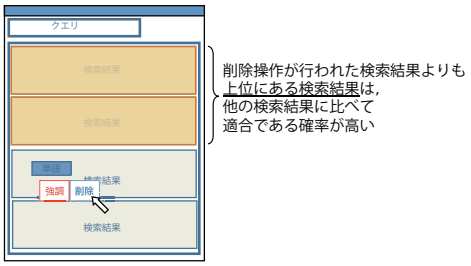

図 4　1 件目以外の検索結果上で削除操作が行われた場合から得られる検索結果の重要度

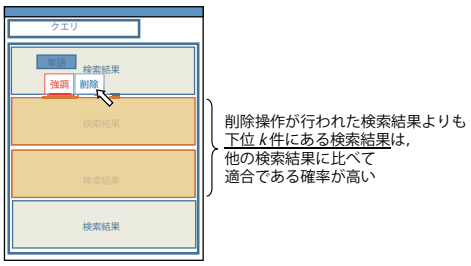

図 5　1 件目の検索結果上で削除操作が行われた場合から得られる検索結果の重要度

3.6.2　削除の場合

削除操作を行った場合は，削除操作が行われた検索結果の順位によって，どのような検索結果がユーザにとって重要であるかが大きく異なってくる．

（1）削除が行われた検索結果よりも上位に検索結果が存在する場合

一般的に，ユーザは検索結果をランキングの上位から順に閲覧していくことが知られている [11]．従って，ユーザは不適と判断した検索結果中で削除操作を行うとすると，ユーザが検索結果を閲覧したにも関わらず，削除操作を行わなかった検索結果というのはユーザに取って重要である確率が高いといえる（図 4）．つまり，あるユーザが検索結果 $r_j(j \neq 1)$ 中で削除操作を行ったとすると，damping vector $\boldsymbol{p}$ を以下のように作成する

$$p(i) = \begin{cases} \frac{1}{j-i} & (i < j) \\ 0 & (i \geqq j) \end{cases}$$

（2）1 位の検索結果中で削除が行われた場合

1 位の検索結果に対して削除が行われた場合，どのような検索結果がユーザに対して適合しているのかを事を推測することは難しい．本稿では，下位 k 件がユーザにとって適合している確率を与えた．

$$p(i) = \begin{cases} \frac{1}{k} & (2 \leqq j \leqq 2+k) \\ 0 & otherwise \end{cases}$$

上記の damping vector $\boldsymbol{p}$ をそれぞれのフィードバックについて求め，その和をとり正規化したベクトルを最終的な damping vector $\boldsymbol{p}$ として採用する．

以上により，生成された遷移行列$\boldsymbol{S}^*$，damping vector $\boldsymbol{p}$を用いて（2）式を適用することで，検索結果が適合している確率を計算し，検索結果の再ランキングを行う．

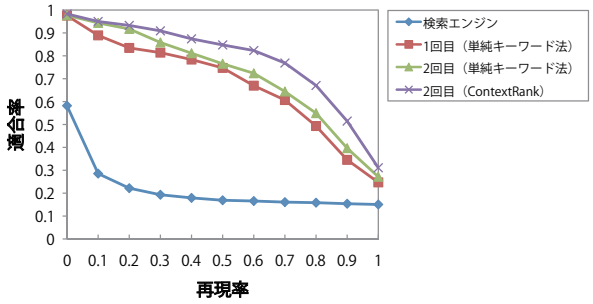

図 6　強調操作における 11 点平均適合率の比較

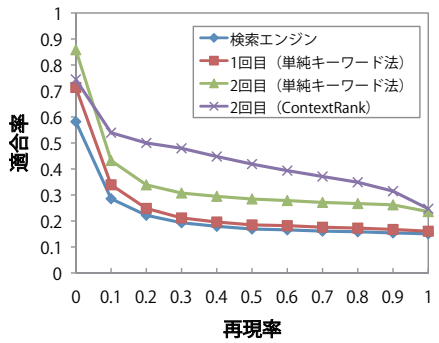

図 7　削除操作における 11 点平均適合率の比較

4. 実　　験

4.1 実験の設定

本稿で提案するランキングアルゴリズムの有用性を明らかにするため，再ランキングの精度を評価する実験を行った．実験として，著者らが [5] にて行ったユーザ実験で用いたタスクおよび被験者の操作ログを用いて仮想的に提案手法の精度評価を行った．著者らが [5] にて行ったユーザ実験の概要は，各タスクについてクエリと正解基準を被験者に伝え，上位２０件の検索結果がその正解基準を満たすよう強調もしくは削除のみを用いて検索結果を再ランキングしていくというものである．その際に行ったタスクおよび正解基準を表 1 に示す．表 1 にあるように，実験として用いられたクエリは全て曖昧性を含むような語である．このようなクエリは適合性重視の意図に基づく再ランキングが多いと考えられる．なお，検索エンジンには Google を用い，検索結果上位 500 件を再ランキングの対象としており，被験者の数は 5 人であった．

提案手法の評価には，ユーザが 2 回目の強調（削除）を行った際に，検索結果に対して仮想的に *ContextRank* を適用し，[5] と同じ正解セットを用いて適合率・再現率の評価を行った．なお，実験に用いたパラメータは予備実験から，$\alpha = 0.85$，$\beta = 10$，$k = 3$ を用いた．

4.2 結　　果

図 6 は強調のみを用いた際の，図 7 は削除のみを用いた際の，検索結果の 11 点平均適合率グラフの推移を示している．なお，単純キーワード法とは，直前のランキングを保持したまま，検索結果のなかでユーザが強調（削除）した語を含む検索結果を上位に（下位に）再ランキングする手法である．また，表 2 は強調・削除それぞれについて，各手法での 30 タスクにおける MAP(mean average precision) を示している．

図 6 より，強調操作のみから *ContextRank* を適用すると，選択キーワード法，つまりユーザが選択した語のみからランキングするよりも，検索結果をある程度閲覧していっても高い適合率を保っている事が分かる．語のみに基づく再ランキングでは，再現率が 8 割を超える程度の検索結果を閲覧する際の適合率は 5 割ほどであるが，*ContextRank* を用いることで 7 割程度の精度を保っている．しかし，本来適合ではない検索結果中の語に対して強調操作を行った場合，*ContextRank* は不適合な検索結果に対して重要であるという判断をしてしまうため，ランキングの精度が悪くなっていた．

一方，図 7 から分かるように，削除操作のみを用いた場合では *ContextRank* を用いることで大幅な精度の改善が見受けられた．単純な削除操作のみでは，一度に大量の検索結果を削除する事が出来ず，操作回数を増やしてもあまり大幅な精度の改善が期待できない．それに対し *ContextRank* はフィードバックのコンテキストからユーザが重要であると判断した検索結果に応じて検索結果の再ランキングを行うので，削除操作のみでも検索結果をダイナミックに再ランキングすることが可能である．しかし，3.6.2 項で述べた，1 位の検索結果中で削除操作が行われた場合に *ContextRank* を適用した場合，精度が悪いタスクがいくつか見受けられた．特に，正解結果が検索結果中にあまり出現しないような，マイナーな話題に関するタスクで低い精度となった．これは，削除が行われた検索結果のすぐ下部には正解のある検索結果が出現しないため，*ContextRank* が本来不適な検索結果に対して重要度を付加するためである．今後は，初期のランキングに [7] のような，検索結果の上位には様々なクラスタからの代表的な検索結果表示するといった手法を組み合わせることで，マイナーな話題の検索結果でも上位に表示される確率を高めるような手法が必要であると考えられる．

5. 考　　察

本稿では，ユーザが与える削除・強調というフィードバックからユーザの再ランキングの意図を推定し，検索結果のランキングを行うという手法を提案した．我々はこの考え方は既存の Web 検索にも適用可能であると考えている．

一般の Web 検索において，ユーザが初めに入力するクエリは曖昧で非常に短く，ユーザは検索結果を閲覧しながら自らの検索意図の変化に合わせてクエリを修正を行う．クエリ修正の方法の中には，AND や NOT を用いて元のクエリにキーワードを追加したり，あるいは検索エンジンが推薦したクエリをクリックすることによって再検索を行う．ここで，この新しく追加されたキーワードを，ある種の語ベースのフィードバックとして捉えることができる．また，フィードバックのコンテキストとして，ユーザが入力したクエリの一連の流れや，どのような検索結果をクリックしたのか，クリックしたページではどのような行動を取っていたのかといった情報を考えることができる．

このような捉え方をすることによって，Web 検索においても

表 1　実験に用いたクエリと正解 Web 検索結果の基準

クエリ	正解基準	クエリ	正解基準	クエリ	正解基準
田中克己	京都大学教授	手塚太郎	京都大学助教	かぐや姫	古典物語
田中克己	日経	手塚太郎	関西大学創立者	かぐや姫	フォークグループ
田中克己	ピアニスト	ジャガー	漫画	NEO	NHK 番組
田中克己	詩人	ジャガー	車ブランド	NEO	株式市場
東西線	東京	ジャガー	ミシン	三条	新潟三条市
東西線	京都	ジャガー	動物	三条	京都市三条通
東西線	札幌	ライスシャワー	馬名	ZOO1	漫画
東西線	仙台	ライスシャワー	ブライダル	ZOO1	小説
ライオン	生活用品メーカ	アマゾン	アマゾン川	アルク	出版社
ライオン	動物	アマゾン	通販サイト	アルク	メガネ店

表 2　各手法間での MAP の比較

	MAP
強調 2 回（単純キーワード法）	0.714
強調 2 回（ContextRank）	0.778
削除 2 回（単純キーワード法）	0.312
削除 2 回（ContextRank）	0.419

散策型や適合性重視型というユーザの意図を考えることができる．例えば，“京都　観光” というクエリで検索したユーザが，検索結果から金閣寺に関するページをクリックし，そのページを閲覧した後，“京都　観光 -金閣寺” というクエリ修正を行ったとする．この場合のユーザのクエリ修正の意図は，“金閣寺に関する情報はもう十分なので，金閣寺以外の観光地に関する情報をいろいろ見てみたい” という散策型の意図をもってクエリ修正を行ったのではないかと推測できる．散策型の意図に基づく場合，例えばユーザには金閣寺以外の観光地に関する情報が色々分散して表示されるようなランキング [7] を用いることで，よりユーザの求める検索結果を提示できる可能性がある．

それに対して，“京都　観光” というクエリで検索したユーザが，検索結果の中から金閣寺に関するページを閲覧した後，銀閣寺や清水寺といった他の観光地に関するページは閲覧しなかったり，ページをクリックしてもすぐに検索結果ページに戻ってしまったとする．そのユーザが “京都　観光　金閣寺” というクエリ修正を行ったとすると，この場合におけるユーザのクエリ修正の意図は，“他の観光地に関する情報はいらないから，もっと金閣寺だけに関する情報が欲しい” という適合性重視型の意図ではないかと推測される．このようなクエリ修正の意図が推測できれば，直前のユーザの検索結果の閲覧行動から，ユーザがどのような検索結果を求めているかを推測して，今回我々が提案したような検索結果のランキングを行う事ができる．

近年，探索型検索 (exploratory search) という，ユーザが曖昧な情報欲求から，情報欲求を具体化していき，最終的にユーザの求める情報にたどり着くまでの過程を包括的に扱った情報検索が注目されている．既存の Web 検索エンジンにおいても，このようにユーザが散策型，適合性重視型のどちらに重きを置いているのかを推測し，上手くランキングを生成してやることで，ユーザの情報欲求の変化や具体化に効率良く応えることが出来るのではないかと考えられる．

6. 関 連 研 究

6.1　コンテキストを考慮した情報検索

Cao らは Web 検索時のユーザのコンテキストを考慮したクエリ推薦手法 [12] を提案している．ユーザがあるクエリを入力した際に，そのユーザが直前にどのようなクエリで検索していたのかというコンテキストに基づいて，推薦するキーワードを求めている．Cao らはユーザが入力してきた一連のクエリを検索行動のコンテキストとして扱っている．それに対して我々はユーザが行ってきた一連の再ランキング操作や，再ランキング対象の周辺の検索結果をコンテキストとして捉えている．また，彼らはクエリ推薦によりユーザの情報検索を支援することが目的なのに対して，我々はランキングそのものを変更することが目的であり，彼らの研究とは異なっている．しかし，ユーザが入力してきた一連のクエリを考慮する事は，散策型・適合性重視型というユーザの再ランキングの意図を推測する上で重要な情報となると考えられる．また，パーソナライズド検索 [13] も，ユーザプロファイルというコンテキストに基づいた情報検索であると捉えることができる．しかし，Cao らの手法やパーソナライズド検索はすべて，あらかじめユーザの行動をマイニングしたり，ユーザプロファイルを作成したうえでクエリ推薦やランキングを行っている．それに対して我々は，ユーザ 1 人 1 人のその場の検索行動に応じて検索意図を推測し，ランキングを変えるということを目的としており，彼らとは異なっている．

6.2　テキスト情報に対する PageRank アルゴリズム

本論文では，ウェブ検索結果集合から仮想的にグラフ構造を構築し，PageRank を適用することによって，ユーザに適合する確率の高い検索結果を確率的に計算する手法を提案した．本研究と同様にテキスト情報から仮想的にグラフ構造を構築し，PageRank を適用した研究として LexRank [14] や TextRank [15] がある．LexRank や TextRank はドキュメントの要約生成やキーワード抽出のために PageRank アルゴリズムを利用している．それに対して我々の手法はユーザの求める検索結果をランキングするために使用しており，彼らとは目的が異なる．

7. ま と め

本稿では，我々がこれまで提案してきた語ベース適合フィードバックシステムにおいて，ユーザの再ランキングの意図に基づいて検索結果を再ランキングする手法を提案した．まず，ユーザの再ランキングの意図を散策型と適合性重視型の 2 種類に分けて考え，それぞれの持つ性質と 2 つの意図の関連について述べた．また，実際に適合性重視型の意図に基づいた再ランキングのためのランキングアルゴリズム，*ContextRank* を提案した．提案手法の有効性を明らかにするために，30 のタスクについて検索結果の再ランキングの精度評価を行った．実験結果から，*ContextRank* は少ないインタラクションで検索結果の精度を向上できることが明らかになった．

今後は，ユーザの再ランキングのコンテキストから再ランキングの意図を推測する手法について考える必要がある．また，今回は適合性重視型の意図の支援方法について検討したが，散策型の意図に基づく再ランキングを支援するための情報の提示方法やランキングに関しても取り組む予定である．

謝辞 本研究の一部は，グローバル COE 拠点形成プログラム「知識循環社会のための情報学教育研究拠点」，NICT 委託研究「電気通信サービスにおける情報信憑性検証技術に関する研究開発」，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、課題番号 1809041）によるものです．ここに記して謝意を表します．

文　献

[1] H. Cui, J. Wen, J. Nie and W. Ma: “Probabilistic query expansion using query logs”, Proceedings of the 11th international conference on World Wide Web, pp. 325–332 (2002).

[2] J.-R. Wen, J.-Y. Nie and H.-J. Zhang: “Clustering user queries of a search engine”, Proceedings of the 10th international conference on World Wide Web, New York, NY, USA, ACM, pp. 162–168 (2001).

[3] R. White and D. Morris: “Investigating the querying and browsing behavior of advanced search engine users”, Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 255–262 (2007).

[4] T. Yamamoto, S. Nakamura and K. Tanaka: “Rerank-By-Example: Efficient Browsing of Web Search Results”, Proceedings of the 18th International Conference on Database and Expert Systems Applications, pp. 801–810 (2007).

[5] 山本岳洋, 中村聡史, 田中克己：“Rerank-By-Example：編集操作の意図伝播によるウェブ検索結果のリランキング”, 情報処理学会論文誌（トランザクション）データベース, **49**, SIG7(TOD37), pp. 16–28 (2008).

[6] B. Tan, A. Velivelli, H. Fang and C. Zhai: “Term Feedback for Information Retrieval with Language Models”, Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 263–270 (2007).

[7] R. Agrawal, G. Sreenivas, A. Halverson and S. Leong: “Diversifying Search Results”, Proceedings of the Second ACM International Conference on Web Search and Data Mining, ACM, pp. 1–10 (2009).

[8] L. Page, S. Brin, R. Motwani and T. Winograd: “The pagerank citation ranking: Bringing order to the web” (1998).

[9] Z. Gyöngyi, H. Garcia-Molina and J. Pedersen: “Combating web spam with trustrank”, Proceedings of the Thirtieth international conference on Very large data bases, VLDB Endowment, pp. 576–587 (2004).

[10] T. H. Haveliwala: “Topic-sensitive pagerank”, Proceedings of the 11th international conference on World Wide Web, New York, NY, USA, ACM, pp. 517–526 (2002).

[11] T. Joachims, L. Granka, B. Pan, H. Hembrooke and G. Gay: “Accurately interpreting clickthrough data as implicit feedback”, Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, New York, NY, USA, ACM, pp. 154–161 (2005).

[12] H. Cao, D. Jiang, J. Pei, Q. He, Z. Liao, E. Chen and H. Li: “Context-aware query suggestion by mining click-through and session data”, Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, New York, NY, USA, ACM, pp. 875–883 (2008).

[13] J. Teevan, S. Dumais and E. Horvitz: “Personalizing search via automated analysis of interests and activities”, Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 449–456 (2005).

[14] G. Erkan and D. Radev: “LexRank: Graph-based Lexical Centrality as Salience in Text Summarization”, Journal of Artificial Intelligence Research, **22**, 2004, pp. 457–479 (2004).

[15] R. Mihalcea, P. Tarau, D. Lin and D. Wu: “TextRank: Bringing Order into Texts”, Proceedings of EMNLP 2004Association for Computational Linguistics, pp. 404–411 (2004).