

筑波大学大学院博士課程

システム情報工学研究科修士論文

検索結果の対話評価に基づく
リランキングインタフェース

軽部 孝典

(コンピュータサイエンス専攻)

指導教員 田中 二郎

2008年3月

概要

近年、Web 上には多くの文章が存在し、その数はなお増え続けている。それに伴い Web で目的の情報を得るための検索技術が重要になってきている。Web 検索で一般的に用いられる有効な手段として Web 検索エンジンがある。しかし、Web 検索エンジンから得られる検索結果にはユーザの検索要求に合致したページとそうでないページが混在することが多いため、ユーザは閲覧によってそれを見分ける作業、検索クエリを修正する作業を強いられるという問題がある。

そこで検索結果の一つのページを検索キーとして追加指定したり、検索結果のページに含まれる単語を追加選択することによって検索結果を順位付けし直すリランキングと呼ばれる手法が多く提案されている。しかし、それらのリランキング手法を用いたインタフェースは、検索者の検索における閲覧行動について考慮されたインタフェースでなく、リランキングに重点を置いている。そこで我々は、検索者の検索要求というあいまいなものを表現するための適切で柔軟な検索要求の表現を検討し、インタフェースを設計した。検索者は、欲しい情報が検索結果のランキングの上位に集めることができたなら満足いく検索ができるという考えの基、検索結果ページや検索結果ページにランキングされるページ内の文章片を対話評価することによりリランキングを行うインタフェース Ensemble Search を実装した。本インタフェースの特徴は、検索要求を文章片で構成する点、及び検索要求の構成をインタラクティブに行える画面設計である。また Ensemble Search の評価を行うため、既存のリランキングインタフェースとの比較実験を行って、本研究のアプローチの有効性を確認した。

目次

第1章	序論	1
1.1	背景	1
1.1.1	検索要求と検索結果	2
1.2	Web 検索行動と考察	2
1.3	Web 検索行動における問題点	4
1.4	研究の目的	5
1.5	本論文の構成	5
第2章	関連研究	6
2.1	ランキング	6
2.1.1	PageRank	6
2.2	適合フィードバック	7
2.3	リランキング手法	7
2.3.1	ページを検索キーとして追加指定するリランキング手法	7
2.3.2	単語を検索キーとして追加指定するリランキング手法	8
2.4	既存のリランキングシステムについての考察	8
2.4.1	ユーザのページ閲覧行動への対応	8
2.4.2	検索要求の表現	10
第3章	リランキングインタフェースの設計	12
3.1	文章片によるリランキングインタフェース	12
3.2	インタラクティブな操作と応答	12
3.3	一画面性を持ったインタフェース	12
3.4	徐々に検索要求を満たす表現	14
3.5	検索要求の再構成の支援	14
第4章	リランキングインタフェース：Ensemble Search	16
4.1	本リランキングインタフェースで使用する要素技術	16
4.1.1	形態素解析	16
	形態素解析器 Sen	16
4.1.2	tf-idf 法	17
4.1.3	ベクトル空間法	17

4.2	Ensemble Search の特徴	18
4.2.1	インタラクティブに取捨選択した適合文章片によるリランキング . . .	18
4.2.2	全リランキング操作が一画面で行えるインタフェース	19
4.3	文書ベクトル生成法	19
4.4	検索要求ベクトル生成法	20
4.5	類似度の計算	21
4.6	検索要求の再構成	21
4.7	利用シナリオ	24
第 5 章	実装	25
5.1	検索語入力部	26
5.2	ページ閲覧部	27
5.3	検索結果リスト表示部	27
5.4	検索要求構成部	27
5.5	検索要求再構成部	27
第 6 章	評価	31
6.1	実験内容	31
6.2	評価指標	32
6.2.1	リランキング結果上位 20 件の適合率	32
6.2.2	結果を見てから単語または文章を評価するまでに要した時間	33
6.3	実験結果	33
6.3.1	リランキング結果上位 20 件における適合率の推移	33
6.3.2	単語または文章の評価に要する時間	36
6.4	評価実験において得られた意見	38
第 7 章	考察	39
7.1	文章片を用いたリランキングインタフェースについて	39
7.1.1	いくつかの文章を一度に追加しリランキング	39
7.1.2	類似した文章の提示	40
第 8 章	結論	41
	謝辞	42
	参考文献	43

目次

1.1	Web 検索サイトの例	1
1.2	一般的な Web 検索の流れ	3
2.1	Editable Search のインタフェース	9
2.2	Flickr のタグクラウド	10
2.3	Rerank.jp のインタフェース	11
3.1	提案手法処理	13
3.2	選択文字列	13
4.1	形態素解析の例	16
4.2	リランキングイメージ	18
4.3	ベクトル空間の例	20
4.4	文書ベクトル	20
4.5	ベクトル空間上の文書ベクトルと検索要求ベクトルの関係	22
4.6	検索要求再構成を行う操作	23
5.1	システム構成図	25
5.2	提案インタフェース画面構成	26
5.3	文章の評価方法	28
5.4	検索要求再構成部	29
5.5	検索要求再構成部の表示	30
6.1	検索結果のベン図	32
6.2	リランキング結果上位 20 件における適合率の推移 (ユーザ 1)	33
6.3	リランキング結果上位 20 件における適合率の推移 (ユーザ 2)	34
6.4	リランキング結果上位 20 件における適合率の推移 (ユーザ 3)	34
6.5	リランキング結果上位 20 件における適合率の推移 (ユーザ 4)	34
6.6	リランキング結果上位 20 件における適合率の推移 (ユーザ 5)	35
6.7	リランキング結果上位 20 件における適合率の推移 (平均)	35
6.8	単語または文章の評価に要した時間 (ユーザ 1)	36
6.9	単語または文章の評価に要した時間 (ユーザ 2)	36
6.10	単語または文章の評価に要した時間 (ユーザ 3)	37

6.11 単語または文章の評価に要した時間（ユーザ4）	37
6.12 単語または文章の評価に要した時間（ユーザ5）	37
6.13 単語または文章の評価に要した時間（平均）	38

第1章 序論

本章では、本研究の背景、目的、本論文の構成について述べる。

1.1 背景

近年、インターネットの普及は目覚しく自宅やオフィス、外出先でまでインターネットを使用するようになってきている。その利用目的は様々であるが、中でも情報収集、ネットショッピングなど多くの情報元から欲しい情報を探す場面が増えてきている。その際に利用するのが検索エンジンである(図 1.1)。



図 1.1: Web 検索サイトの例

検索の中でもキーワード検索は最も一般的で最も多く使用されている。一方、個人でも Web 上に多くの情報を載せる事が増えてきている現在では、Web 上には多くの文書が存在する。情報検索技術の向上に伴いそれら多くのデジタル文書から欲しい情報を探するのは容易になりつつあるが、欲しい情報をいくつかのキーワードで表現し、目的の情報に辿り着くためには

検索対象分野に対する知識や検索技術、検索知識が必要な場合が多い。よって検索対象分野に対する知識や検索技術、検索知識を持たないユーザは多くの情報の中から欲しい情報のみを見つけることが困難な状況にある。例えば、プログラミング言語の『Java』についてそれが何であるか全く知らない人にとって『java』とはどのようなもので関連する技術を調べる際、まず検索エンジンに『java』という単語を検索クエリとして入力し検索をした場合、検索結果の中にはそのユーザが知らない単語が並ぶ。そこで見た検索結果からうまくキーワードを見つけ検索クエリを修正し再検索するといったことが難しいと思われる。仮に検索結果の中にある程度欲しい情報がいくつか含まれていたとしても、本当に欲しい情報が下位にランキングされている場合、ランキングの上位のみを見て再検索をしたり、検索をあきらめてしまい問題の本当の解決に至らない場合も多い。フレッシュアイ (<http://www.fresheye.com/>) の調査によると約八割のユーザはキーワードを一語のみ用いて検索している。また三割は検索結果を絞り込むことができないなどの理由をあきらめていることがわかっている。しかし、どうしても欲しい情報である場合上位からランキングを見ていき、欲しい情報と欲しくない情報を判断しながら、欲しくない情報をいくつも確認していかなければならない。このようなユーザにかかる負荷は、ユーザが入力した検索クエリのみでユーザの検索要求を判断する検索エンジンでは解消することはできていない。ここで重要になるのがユーザにとって必要な情報が上位にランキングされることである。そこで、検索エンジンから得られた結果を様々な方法でリランキングし、ユーザが欲しいであろう情報を提示するという手法が研究されている。

1.1.1 検索要求と検索結果

情報検索を行うユーザは何らかの検索要求を持って検索をする。その検索要求は検索エンジンに検索クエリとして単語、もしくは複数の単語で表現し投げられる。しかし検索エンジンが返す結果は同じクエリである場合いつも同じような結果である。そのため同音異義語や、複数の意味を持つ単語を検索クエリとして投げた場合全てのユーザに同じ結果を返してしまう。よって検索結果の中には欲しい情報と欲しくない情報が混在することが多い。ユーザはその検索結果の中から欲しい情報を閲覧や再検索を繰り返し、探していくという作業を強いられる。またユーザが求める情報はユーザによっても検索の度にも違うため、その時欲しい情報を検索クエリとして表現することが難しい場合がある。特に検索対象分野の知識があまりないユーザにとってこれは顕著である。そのため検索結果のページをキーとした検索などが考案されている。しかし、検索対象分野の知識があまりないユーザにとって検索要求というあいまいなものは単語で表現するのは難しく、またページで表現することはページ内に多くの話題を含んでいる場合有効でないと我々は考える。

1.2 Web 検索行動と考察

一般に、Web 検索を行うユーザは以下に示す図 1.2 の流れで検索を行っている。

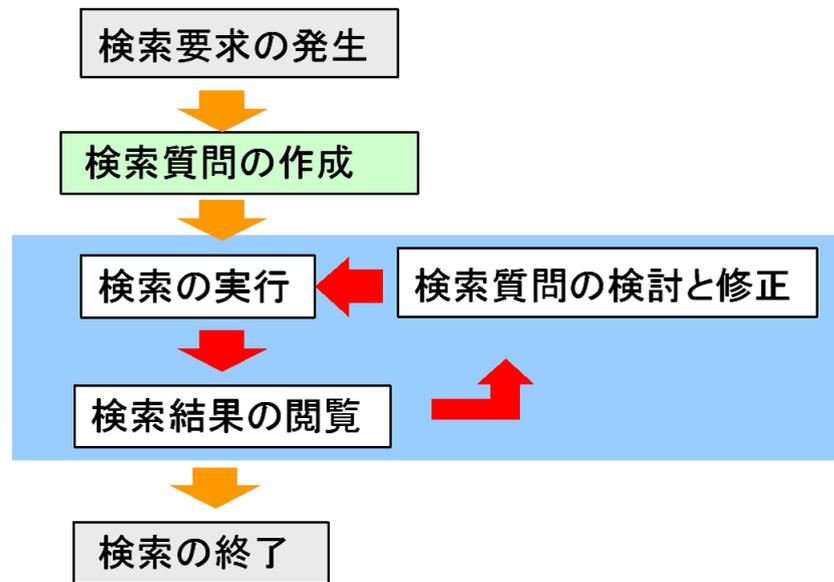


図 1.2: 一般的な Web 検索の流れ

検索要求の発生 ユーザに情報要求が生まれ、Web 検索を行いたいという要求が発生する。検索要求とはユーザがある目的を達成するために現在持っている知識では不十分であると感じている状態で、検索要求は細かく分けると Taylor の研究 [10] によると以下の 4 つの階層に分類される。

- Q_1 : 直感的要求 現状に満足していないことは認識しているが、それを具体的に言語化してうまく説明できない状態。
- Q_2 : 意識された要求要求 頭の中では問題は意識できるが、あいまいな表現やまとまりのない表現でしか言語化できない状態。
- Q_3 : 形式化された要求 問題を具体的な言語表現で言語化することができる状態。
- Q_4 : 調整済みの要求 問題を解決するために必要な情報の情報元が同定できるくらい問題が具体化された状態。

検索質問の作成 検索要求に応じて単語、もしくは複数の単語を用いて検索質問を作成し、検索システムに入力する。

検索の実行 検索質問を入力したら、検索ボタンを押すなどの検索実行のため必要な操作を行う。

検索結果の閲覧 ユーザは検索結果ページにランキングされるページをについて、タイトルやスニペットなどの情報から自分の欲しい情報であるかどうかをある程度判断し、リンクされているページを閲覧する。その際、無意識にこの情報は要る、この情報は要らないと評価しながら閲覧している。

検索質問の検討と修正 閲覧の際目にした情報によってインスピレーションを受け、検索質問を検討し、修正し、状況によっては以前の検索結果と比較を行うなどして結果の可否を判断する。

検索の終了 ユーザが検索結果の内容を閲覧した結果内容の質的、量的に十分と思われる情報が得られたと判断した場合、もしくは現在使用中の検索システムで検索を継続することをあきらめた場合、その時点で検索を終了する。

1.3 Web 検索行動における問題点

一般に我々が何らかの問題を抱えた時、前節において述べた4つのどの状態にあるかによって問題解決までの戦略も異なる。例えば必要な書籍の名前までわかっている場合(Q4の段階)には、図書検索システムを使って検索をする、もしくは司書に相談するなどして欲しい本を探すことが可能である。以下にQ1からQ4の段階までの説明をする。Q1の段階は、今の状態には満足しておらず、問題があることは認識できるが、それがどういう問題なのかを言語化できない、とにかく何か足りない、という非常に困った状態である。Q2の段階は、Q1の段階が少し具体化されて、何が問題であるかは何となくわかるが、まだ明確には言語化できない状態である。つまり、あいまいな表現、まとまりのない表現でしか言語化できない状態である。このような時、完全に言語化できなくても言語化する努力をすることによって何が問題かが明確になることがある。この状態では他人からの直接的な助言を得ることよりも、まず自分が言語化を試みることが重要である。Q3の段階は問題が整理できて、明確に言語化できる状態である。この段階ではどうやって問題が解けるかが具体的にわかっているわけではなく、何が問題か明確になっただけである。問題が言語化できたからといってすべての問題が解けるわけではない。Q4の段階は、問題が同定できて、問題解決のために具体的にどういう手段があるかが見えてきた状態である。例えば、身近に使える情報検索システムでどういうキーワードで検索すれば自分の必要な情報が得られるかがわかっているような段階である。したがって、Q4の段階では実際に利用可能な情報検索システムの制約も考慮しなければならない。例えば、Q3の段階で明確に言語化できても、その情報検索システムが単純なキーワードしか使えない場合は、Q3の段階で言語化できた情報がすべてそのシステムで利用できるとは限らない。

こういった状態で情報検索システムを使って検索をする場合どのようにシステムに自分の欲しい情報をうまく伝えるかということが困難である。またWeb検索において本当に探している文章がない場合も考えられるが、検索結果の中にはユーザが必要とする情報を含んでいるが、検索結果の量が多いため検索結果の下位までは確認せずに検索をあきらめてしまう場合が多々ある。事実オプトとクロスマーケティングの検索エンジンの利用状況に関する調査結果[24]によるとユーザが見る検索結果の閲覧ページ数は平均して4ページ目までしか見ない。このような場合下位に存在するそのユーザにとって必要な情報は一度もユーザの目に触れることなく終わってしまう。

1.4 研究の目的

本研究では前節で述べた Q2、Q3 の段階のユーザに焦点を当てる。このような段階のユーザが Web 検索エンジンから得られる検索結果にはユーザの検索要求に合致したページとそうでないページが混在することが多い。この原因は検索対象分野の知識があまりないユーザにとって検索要求を単語で表現することは難しい事、一つのページ内には検索ワードを含むが他の情報も含まれている事にあると我々は考えた。また、同じクエリでの検索でも求める情報はユーザによっても検索の度にも違うため、ユーザは自分が欲しい情報が検索結果のランキングの上位に集めることができれば満足はいく検索ができると考える。そこで我々は、既存のシステムの問題点を考慮に入れ、検索要求という単語で表現するの事やページで表現するには大雑把すぎるものを、ユーザが『欲しい情報である』、『欲しくない情報である』と判断した検索要求に適合（または非適合）した文章片（以下、適合文章片、非適合文章片と呼ぶ）を基に、その時のユーザの検索要求を構成し、ランキングを再構成し、検索を支援するためのインタフェースを設計、実装することを目的とした。また本研究のインタフェースは検索結果をインタラクティブに、かつ動的にリランキングを行えること、検索結果の確認がしやすいことを考慮し実装し、それらにより検索要求の表現を容易にし、検索結果の閲覧時におけるユーザのページ確認作業、再検索といった負荷を軽減し、その時のユーザの検索要求に最も合った検索結果を提示することが目的である。最終的にはランキングされたページを上位から順に閲覧していただくだけでユーザにとって最良の情報が得られるようにし、最小の負荷で検索におけるユーザの満足度を高めることが本研究の目的である。

1.5 本論文の構成

本論文の構成は以下のとおりである。第 2 章では、本研究の関連研究について説明し、リランキングインタフェースと検索要求の表現について考察し、その特徴と問題点について述べる。第 3 章では、リランキングインタフェースの設計について述べる。第 4 章では、実際に試作したリランキングインタフェース Ensemble Search について述べる。第 5 章では、Ensemble Search の実装について述べる。第 6 章では、Ensemble Search の有効性を示すために行った実験による評価の方法および結果について述べる。第 7 章では、前章で得られた評価結果から Ensemble Search の有効性と改善すべき点について述べる。第 8 章でまとめる。

第2章 関連研究

本章ではまず Web 検索エンジンにおける主要なランキング手法である PageRank について説明する。次にリランキングにおいて重要な要素である適合フィードバックについて述べ、本研究の関連研究を挙げ、それらのシステムについて考察する。

2.1 ランキング

近年の Web 検索エンジンでは、ウェブのリンク構造を解析し、ランキングを構成する方法が広く採用されている。一般的に検索要求が生まれた場合利用する検索エンジンの代表的なランキング手法 PageRank[4] について以下に説明する。

2.1.1 PageRank

Page[6] らによって提案された PageRank は、今日、Google などの代表的な検索エンジンに用いられている手法である。PageRank の基本的となるアルゴリズムは、「多くの良質なページからリンクされているページは、やはり良質なページである」という再帰的な関係の基、全てのページの重要度を判定する。そのページへのリンクをページへの支持投票と見なし、より重要度の高いページにリンクされればされるほどページとしての重要度が高くなる。したがって単純に被リンク数が多いだけではページとしての評価は高まらず、被リンクの価値も重要な要素となる。あるページの重要度が、被リンクの重みの和で定義されることは「良質なページにたくさんリンクされているページは良質である」という考えを反映している。またリンクの重みを求める方法は、リンクを出しているページの重要度を出力リンク数で割っている。これは厳選されたリンクほど良いリンクである、という判断を行っているためである。このような手法をとることリンクという形で他の人が良いページであるという評価をしたページが問題を抱えたユーザに提供されるシステムであるため検索エンジンは情報検索の際最も多く使われ、かつ有効な検索手法である。しかし、この PageRank はページに対するスコアによって決定されるため、検索クエリによって変化しない。またクエリユーザが異なっても同じクエリであれば同じ結果が返される。例を挙げると『Java』という単語で検索エンジンを用いて検索をした場合、検索エンジンは『java』という単語を含み PageRank 法などのランキング手法に沿ってランキングを形成し、ユーザに返す。この場合、検索エンジンは検索をかけたユーザがどのような『java』についての情報が欲しいかといった事は考慮に入れてくれない。つまりユーザはプログラミング言語の『java』の情報が欲しいのか、またはインドネ

シアにある『java』島についての情報が欲しいのか、はたまたコーヒー豆の『java』コーヒーについての情報が欲しいのか、どれかの状態にあったとすると、検索エンジンは『java』という単語を含むページを検索結果として返すだけなのでその場合ユーザにとって欲しい情報は多くの検索結果の中に埋もれてしまい、本当に欲しい情報を見つけるのが困難になる。

2.2 適合フィードバック

検索システムを利用するユーザが目的とするページにたどり着くためには、システムに自らの要求を適切に伝える必要があり、そのための手法として適合フィードバックが提案されている。この適合フィードバック (Relevance feedback) (適合性フィードバック、関連性フィードバックとも呼ぶ) はもともとは情報検索において確立された手法である [8]。これは出力された検索結果に対してユーザが評価を行い、その評価を基にシステムが再検索、または検索質問の再構成を行うというものである。つまりシステムにユーザの興味を伝え、そのフィードバックをシステムから得るという手法である。適合フィードバックの定義は、情報検索において検索結果として出力された文書の内容に基づいて、検索質問や検索戦略を修正することを目指す。適合フィードバックの例を一つ挙げると、goo などの検索エンジンで見られるように検索エンジンから得られた検索結果において興味のあるページをユーザが指定すると、そのページの内容に近いページを再度検索してくれるといったものである。適合フィードバックにおいて、システムがユーザから得る代表的な情報として、検索結果の適合性の判定がある。検索結果に順位付けのできるシステムでは、ユーザに上位の文書から適当な数の文書を提示し、ユーザにそれらが検索質問に適合しているかどうかの判定をさせるというものである。

2.3 リランキング手法

リランキングとは検索エンジンなどから得られたランキングを様々な要因によりランキングを再構成し提示する手法のことで、近年様々な研究がなされている。そのいくつかを紹介し、それぞれにおける問題点をあげる。

2.3.1 ページを検索キーとして追加指定するリランキング手法

goo[18] に代表されるいくつかのサーチエンジンでは検索結果の複数のページを選択し、それらをキーとし、再検索をすることができる。ページを検索キーとして追加指定するリランキング手法として平田らによる研究 [1] 松本らによる研究 [21] がある。平田らによる研究では、検索されたページ群の中からユーザがページを選択し、選択されたページの特徴ベクトルを用いて、基の質問ベクトルを修正し、リランキングする手法をとっている。この手法を使うことで似たようなページを集めたい場合にはシステムは有効である。しかし、検索結果の中から似たようなページを集める事は可能であるが、検索要求の表現を考えると、一つのページ内には複数の話題を含むページや、一つの単語についての説明のページであっても、その単

語が複数の意味を持つ場合などページをキーとして検索質問を再構成するのは検索要求の表現としては単位が大きすぎるといった問題点がある。例えば検索クエリとして『Java』、『コーヒー』という単語を入力し、検索を行った結果の中には、プログラミング言語の『Java』とコーヒーの『java』との関係を説明しているページが存在している。このようなページをキーとして検索をした場合、検索システムはどちらの『java』かわからずに両方に関するページを似ているページとして返してしまう。このように一つの単語で複数の意味を持つ多義語に関するページをキーとして検索をする場合、検索キーをページとすることはシステムにユーザの細かい検索要求を伝えることはできない。例で挙げた検索クエリに対してシステムはどちらの『java』に関するページを返せばいいのかといったことは考えることはできない。

2.3.2 単語を検索キーとして追加指定するリランキング手法

そこでページではできない細かい表現を単語を検索キーとして追加指定するリランキング手法も多く研究されている [3][19][23]。山本らによる研究 [3] では検索結果に対するユーザの操作を用いてリランキングを行うという点で本研究と似たアプローチを取っている。以下の図 2.1 に山本らによる検索システム Editable Search のインタフェースを示す。

山本らによる研究では、検索質問の修正の際、追加、削除されるのはタイトルかスニペット内の単語である。検索結果分類のためにフォルダを設け、検索結果ページに含まれるタイトル、スニペットの中から単語を追加することで、そのキーワードを含むページを分類することができるようになっている点も特徴的である。またタグクラウドと呼ばれる検索エンジンから得た検索結果のタイトルとスニペットから抽出した単語の出現頻度が高いものをタグとして提示し追加、削除するキーワードの決定支援を行っている。タグクラウドとは flickr[25] などで見られるタグの一覧表示 (図 2.2 参照) である。より頻出な単語ほど大きく表示されるのが利点である。山本らによる研究が本研究と異なる点は検索質問の修正に単語を用いるか、文章片を用いるかという点とインタフェースのデザインが異なる。

2.4 既存のリランキングシステムについての考察

本節では、既存のリランキングシステムについて考察を行い、本研究の位置づけを行う。

2.4.1 ユーザのページ閲覧行動への対応

ユーザは検索を行う際、ある程度検索結果のランキングページのタイトルスニペットを見て、自分の欲しい情報かどうかを判断しているが、その中で気に入った情報、気になったページがあれば検索結果のランキングページのリンクを辿りそのページを閲覧する。ここで既存のリランキングシステムのインタフェースを見てみると検索結果のリランキングに焦点を当てているため、ユーザのページ閲覧行動に対する備えは十分とは言えない。前節で述べた山本らによる研究のリランキングシステム Editable Search ではランキング結果、またはリラン



図 2.1: Editable Search のインタフェース

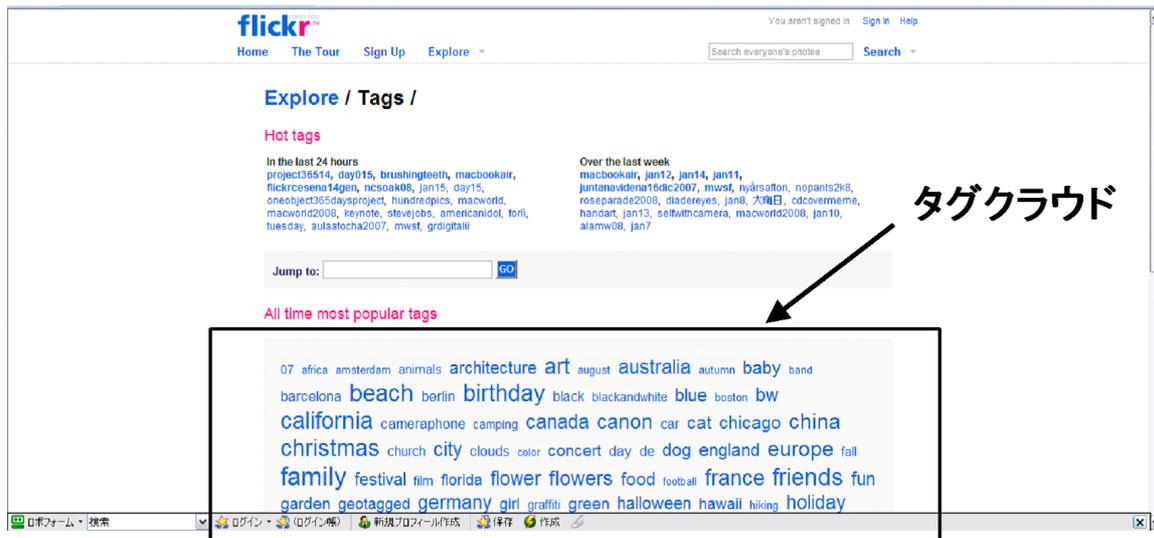


図 2.2: Flickr のタグクラウド

キング結果の中で気になったページを見たいとユーザが思った場合、検索結果ランキングページのリンクをクリックするとブラウザが立ち上がり、ページが表示される。しかし、Web 検索の際検索結果のリストページとそれにリンクされるページとを見比べ、行ったり来たりしながらページを閲覧する行動が予備実験でも多く確認されたためリランキングシステムとブラウザを別々のウィンドウで表示しているとユーザはアプリケーション間を行ったり来たりしなければならない。これはユーザにとって負担となる操作である。もう一つ Web 上で公開されている Rerank.jp についても考察を加える。以下の図 2.3 に Rerank.jp のインタフェースを示す。Rerank.jp においてリランキング結果のページにリンクされているページを見たい場合への対応は、リンクをクリックすると別ウィンドウ、あるいは別のタブでページを表示する。これについても検索結果とそれにリンクされるページを閲覧するユーザにとって、ウィンドウ間、タブ間を行ったり来たりする操作が強いられる。本研究では、このような検索結果ページと検索結果ページにランキングされるページの内容を見比べたり、行ったり来たりしながら閲覧するという行動への対応としてインタフェースの画面構成に一画面性という制約を設け、ランキングページとそれにリンクされているページの表示を並べて行っている。

2.4.2 検索要求の表現

既存のリランキングシステムでは、単語を検索キーとして追加指定し、リランキングを行う手法や、検索結果のページを検索キーとして追加指定し、リランキングする手法が多く提案されている。しかし、我々はユーザのあいまいな検索要求は、単語では指定することが難しいこと、またはページで検索要求を構成する場合には、ページ内に多種多様な内容、多義語を含むようなページではシステムに検索要求を伝えるには大きすぎる単位であると考える。

第3章 リランキングインタフェースの設計

本章ではユーザの Web 検索の際に行う行動とその考察をし、検索要求をシステムとユーザが対話的に満たしていくリランキングインタフェースの設計について説明する。

3.1 文章片によるリランキングインタフェース

本システムでははじめユーザが検索後入力部に検索語を入力し、検索を実行すると検索エンジンから得られた検索結果のタイトルとスニペットを形態素解析器 Sen[26] により解析し、単語に切り分け名詞と出現回数を抽出し、単語それぞれについて tf-idf 法を用いて重み付けを行い、ベクトル空間法を用いてページ一つの文書ベクトルを生成する。そしてユーザは次に検索結果のページまたは検索結果ページにあるリンクされているページを閲覧しながら自分の欲しい文章を探す。欲しい文章に似た文章、欲しい文章に近い文章、または欲しい文章を見つけた場合それを選択し、検索要求構成部にドラッグアンドドロップする。するとその文章に似た文章が書かれたページがリランキングによって上位にランキングされる。それを繰り返し行うことで自分好みの検索結果を得ることができる。本システムの処理の流れを図 3.1 に示す。

3.2 インタラクティブな操作と応答

ユーザはまず検索結果リストページまたは、検索結果リストページにランキングされリンクされているページの内容から自分の検索要求にマッチしていると判断する文章、あるいは検索要求にマッチしていないと判断する文章を選択文字列として選択する (図 3.2)。

次にその選択文字列をドラッグアンドドロップというインタラクティブな操作を用いて適合 / 非適合を評価する操作を行う。その操作を行うとシステムはインタラクティブな応答として文章片のドロップと同時にリランキング結果を表示する。このインタラクティブな表示をすることで、ユーザの検索行動における一連の動作を妨げることなく検索要求の表現をすることを可能とする。またユーザに負荷を与えることなく検索をするための支援をする。

3.3 一画面性を持ったインタフェース

関連研究の章の既存のリランキングシステムについての考察で述べたように既存のリランキングインタフェースでは、ユーザの検索行動において重要なページ閲覧行動に対する支援

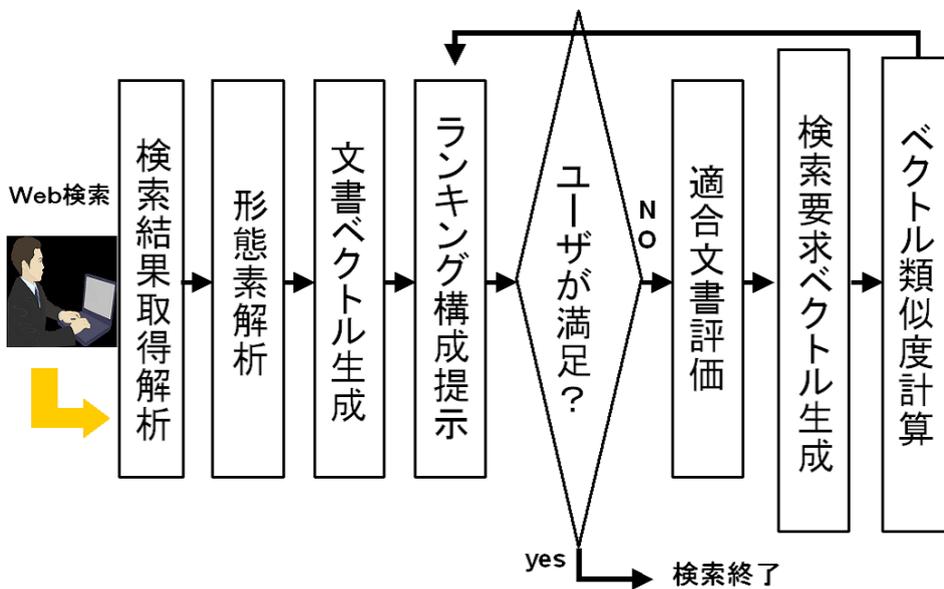


図 3.1: 提案手法処理



図 3.2: 選択文字列

が十分とは言えない。そこで提案するリランキングインタフェースは一画面でリランキング結果とページの閲覧が同時に、かつインタラクティブに表示をする。また既存のリランキングシステムでは行っていなかった、検索結果のリスト表示ページにリンクされているページ内の単語や文章の検索要求の構成への活用を行う。これによりユーザは検索結果のリスト表示ページとそれにリンクされているページ内の文章の両方から適合・非適合文章片を、ひとつのページの複数箇所、あるいは複数のページからいくつでも選択し追加可能となる。それによりユーザがシステムに対して伝える検索質問の柔軟な表現が可能となる。さらにユーザが検索結果のリストページとそれらに含まれるページの内容の閲覧を同時に行えることでユーザのページ間の行き来という負荷を軽減することができる。

3.4 徐々に検索要求を満たす表現

検索要求とは Web 検索行動と考察の節で述べたようにうまく言語化し表現することができない場合がある。そのような場合ユーザは検索結果を閲覧することでインスピレーションを受け、徐々に検索要求が具体化されていくものだ我々は考える。また検索要求が言語化できる場合についてもシステムにうまく自分の欲しい情報を伝えるためにユーザとシステム間の対話が重要だと考える。言語化された検索要求が多義語や同姓同名の人物に関する時など、検索システムはどの意味の単語で、どの人物に関するページが欲しいのかといった要求は現在の検索エンジンをはじめとするキーワード検索システムでは読み取ることができない。よってユーザはシステムに欲しい情報はこれだ、またはこれに似た情報をもっと欲しい、この情報は要らない、といった要求を伝えなければならないし、システムはそれを読み取り提示できることが望ましい。これらを考慮し、本研究では適合・非適合文章片として検索結果リストページまたは、検索結果リストページにランキングされリンクされているページの内容を評価することで、システムに欲しい情報、欲しくない情報を伝えることができるシステムを目指す。

3.5 検索要求の再構成の支援

大坪らによる論文 [11] によると人間の興味は固定化されたものではなく、その時点でユーザが目にするもの = システムが提示した情報によって影響を受け変化し続けるもので、「興味に合う情報」というのは必ずしもユーザの意識の中に最初から存在しているわけではない。またユーザは候補となりうる情報を全て知っているわけでもないので、「自分が知らなかった情報」を目にすることにより、そちらに興味に移行していくことが起こりうると考えられる。また頭の中で漫然と願望を抱いていたとしても、それを実際に候補としてシステムが提示した場合、それを見ることにより「気が変わる」ということも起こりうるということを述べている。一方で、ユーザはリランキングシステムのリランキングアルゴリズムを知っていることは稀であり、リランキング結果が意図に沿わない場合も考えられる。さらに間違っ文章を評価してしまう場合も考えうる。本研究ではそれらの事情を踏まえ、検索要求を再構成する

ための支援を行う。検索要求の再構成に用いる操作と表示は、インタラクティブな操作と応答で説明したインタラクティブなドラッグアンドドロップの操作と、動的なリランキング結果の表示を可能とするものである。また一画面性を保つため、検索要求再構成画面は、一画面に近似できるような表示方法を行う。

第4章 リランキングインタフェース： Ensemble Search

4.1 本リランキングインタフェースで使用する要素技術

4.1.1 形態素解析

本節では提案インタフェースで使用する形態素解析について説明する。形態素解析とは自然言語処理の最も基本となる処理である。形態素とはそれ以上分割できない語の単位であり、形態素解析とは自然言語の文中の単語を識別し、その語形変化を解析し、品詞を同定する処理。近年、自然言語で書かれた大量の文書が電子化され利用されている。例えば、Web 上では様々な情報をテキストや写真などで記述したページがある。より効率良く利用するためにはコンピュータによる文書の解析が必要である。コンピュータが文書を解析するためには、必要不可欠な処理である。形態素解析は応用分野が多くな漢字変換、音声合成、機械翻訳、情報検索などの分野で広く利用されている。そのため多くの形態素解析システムが開発されている。形態素解析システムでは、形態素辞書(単語辞書)と形態素解析接続規則を用いて、形態素に分割する。

形態素解析の例として『データ集合の中から目的のデータを探し出す』という文を形態素解析にかけた例を以下の図 4.1 に示す。

データの集合の中から目的のデータを探し出す

データ／集合／中／目的
2 / 1 / 1 / 1



図 4.1: 形態素解析の例

本研究では文から名詞のみを抽出するためこの例の場合、『データ』という単語は2回、『集合』、『中』、『目的』はそれぞれ一回文中に出現したという解析結果が得られる。

形態素解析器 Sen

本システムでは、形態素解析器の Sen を使用している。本インタフェースは java で実装しているため、Java で書かれた日本語形態素解析システムである Sen を用いた。Sen は C++ で

開発されている MeCab を Java に移植したもので辞書は MeCab、茶筌と同じ IPA の辞書を利用している。IPA 辞書は情報処理推進機構によって開発されている辞書である。

4.1.2 tf-idf 法

tf-idf 法は、文章中の特徴的な単語（重要とみなされる単語）を抽出するためのアルゴリズムであり、主に情報検索や文章要約などの分野で利用されている [9]。これは、TF 法と IDF 法を組み合わせる単語 t の特定の文書 D_i の重みを計算する手法であり、情報検索分野において索引語の重み付けに一般的に用いられている手法である。TF 法 (Term Frequency) は、索引語の頻度をもとに重み付けする方法であり、文書 D_i における単語 t の頻度で $tf(i, k)$ と表記する。これは文書中で出現頻度が高い単語はその文書において重要であると考えられるためである。しかし、多くの文書に出現する単語は、文書を特定する性質を持たないことが多いため、その単語がどのくらい特定性を持つかを重み付けに反映させているのが IDF 法 (Inverse Document Frequency) である。これは、索引語の特定性をもとに重み付けする方法であり、ある索引語 t が全文書中のどれくらいの文書に出現するかを表す尺度である。

tf-idf 法は、これらの二つの手法を組み合わせ単語の重み付けを次式のようにモデル化したもので、文書内で繰り返し使われ、かつ他の文書にはあまり見られないような単語は文書の内容をよく表しているという考え方に基づいている。

$$w(i, k) = tf(i, k) \cdot idf(k) \quad (4.1)$$

ここで tf とは文書 D_i における単語 W_k の出現頻度であり $idf(k)$ は以下の式で計算される。

$$idf(k) = \log(N/df(i)) + 1 \quad (4.2)$$

ここで N は全文書数を表し、 $df(i)$ は単語 W_k が一回以上生起する文書の数を示している。tf-idf 法は単語重要度に関する TF 法と IDF 法という 2 つの異なる観点を組み合わせ、単語重要度をバランスよく評価しようとする手法である。TF 法とは異なり出現する文書数が少ない単語は文書の絞り込みに役立つから重要であると考えられている。

4.1.3 ベクトル空間法

本節では本リランキングインタフェースで使用するベクトル空間法について説明する。ベクトル空間法とは、文書をベクトルによって表現し、個々の文書をベクトル空間上の点として表現する。また検索質問を単語、または複数のキーワードの集合を用いて文書と同様ベクトルで表現し、ベクトル間の類似度を計算する検索手法である。各文書はベクトル空間上の点であるため、文書間の距離を定義することが可能となり、自然言語で記述された文書の近さを定義することができる。また、ユーザの検索要求もベクトル空間で表現することによって、文書と検索要求の距離を求めることができる。文書をベクトルで表現するため、全文書に含まれる単語を要素とし、その単語に対する重みを tf-idf 法により計算して、文書をベクトル表

現する方法が一般的である。本稿で提案するリランキングインタフェース Ensemble Search ではリランキングにこの手法を用いる。

4.2 Ensemble Search の特徴

本インタフェース Ensemble Search は特徴として以下の二つを持つ。

- インタラクティブに取捨選択した適合文章片によるリランキング
- 全リランキング操作が一画面で行えるインタフェース

4.2.1 インタラクティブに取捨選択した適合文章片によるリランキング

本インタフェースによる検索結果のリランキングのイメージを図 4.2 に示す。ユーザは適

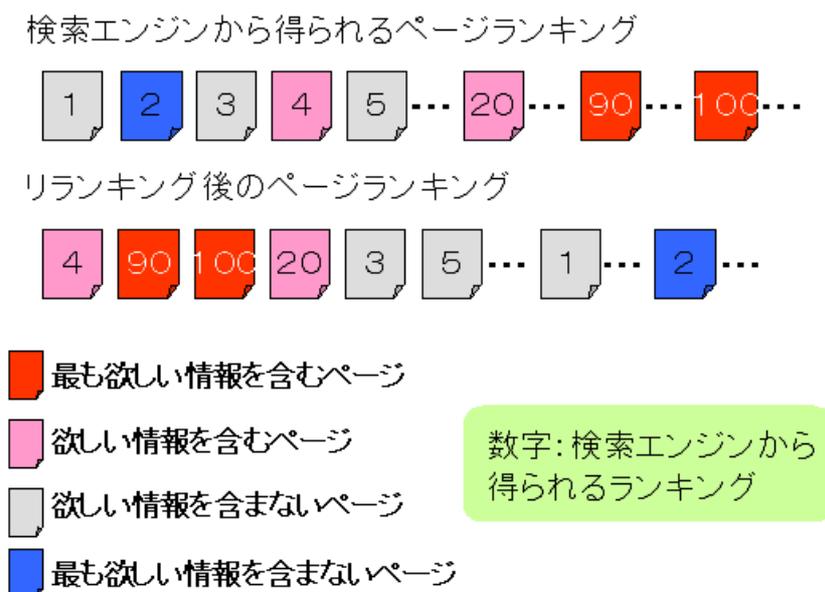


図 4.2: リランキングイメージ

合・非適合文章片を検索結果ページ内から選択すると、システムはその適合・非適合文書片を基にリランキングを行う。ユーザは適合文章片を追加してゆく事によって、より満足する検索要求を徐々に構築してゆく事が可能である。逆に、リランキング結果が悪くなったと感じた場合には、ユーザは先に追加した適合・非適合文章片を除けば検索要求の構成を戻す事も可能である。

4.2.2 全リランキング操作が一画面で行えるインタフェース

本インタフェースでは、検索結果のリランキング作業に必要な以下の操作を一画面で行うことが可能となっている。

- (a) 検索語入力
- (b) 検索結果リストの閲覧
- (c) 検索結果のページの閲覧
- (d) 適合文書片の取捨選択

ユーザが検索語入力部に検索語を与えて検索を開始すると、検索エンジンからの検索結果が検索結果リストに表示される。検索結果リストの項目を選択することによって、そのページをページ閲覧部において閲覧できる。ユーザは閲覧と同時に適合・非適合文章片の選択が可能である。表示されているページの一部をドラッグして選択し、続けて検索要求構成部へドロップすることにより適合・非適合文章片を検索要求に追加できる。この際、リランキングが行われる。なお、適合・非適合文章片は、ひとつのページの複数箇所、あるいは複数のページからいくつでも選択し追加可能である。適合・非適合文章片を検索要求から除くには、検索要求構成部にポインタを一定時間留める。すると画面構成が図 5.5 下のように変わり、検索要求構成部が大きく表示される。この状態の検索要求構成部にはそれまでに追加された全ての適合・非適合文章片がオブジェクトとして表示される。一度追加した適合・非適合文章片を取り除くには対応するオブジェクトを周辺領域までドラッグするれば良い。この時にもリランキングが行われる。マウスが検索要求構成部を離れると元の状態(図 5.5 上)に戻る。ユーザは上の (b) から (d) を高速に行えるため、検索要求を容易かつ漸次的に構築し自分の欲しいページを上位にリランクさせられる。

4.3 文書ベクトル生成法

本節では文書ベクトルの生成法について述べる。まず本インタフェースでユーザは検索エンジンにクエリを送る。そして受け取った検索結果を形態素解析器 SEN によって解析する。形態素解析器 SEN によって解析するのは検索結果のタイトルとスニペット(ページの要約文)であり、それぞれを単語に切り分け名詞のみを抽出する。今回抽出する品詞を名詞のみに限定したのは、最もそのページの特徴が現れる品詞が名詞であると考えたからである。またタイトルに含まれる名詞限ってはスニペットに含まれる名詞より重みを増して計算している。これはタイトルにはそのページの作者が最も強調したい部分であると我々は考えたからである。切り取った単語とそのページ内に出現した回数を計算し、それらを tf-idf 法によって全ての単語について重み付けをする。その重みを用いて N 次の文書ベクトルを作成する。ページをベクトル空間に配置するということは文書をベクトルによって表現し、個々の文書をベクトル空間上の点として表現する事であるが、例を挙げて説明すると、『Java』という検索語で検索

エンジンにクエリを投げた場合、検索結果の中には Java というワードを含んだ様々なページが含まれる。それはプログラミング言語である『Java』のページであったり、『Java コーヒー』のページであったり、『Java (ジャワ) 島』に関するページであったり多様な分野のページが混在する。これら 3 種類のページを三次元のベクトル空間で表現すると図 4.3 のようになる。

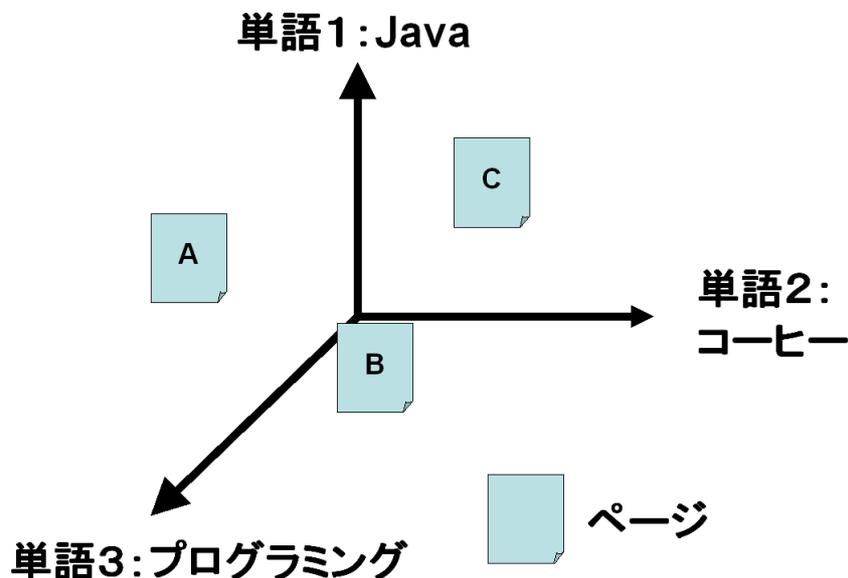


図 4.3: ベクトル空間の例

それぞれベクトル空間の軸は単語で表現され、文書（ページ）はそれぞれその空間上の点で表現される。文書ベクトルは図 4.4 のように表現する。

$$\vec{d}_i = (w(i,1), w(i,2), \dots, w(i,N))$$

単語一つの重み

図 4.4: 文書ベクトル

4.4 検索要求ベクトル生成法

検索要求ベクトルはベクトル空間法という検索質問のことである。検索要求ベクトルはユーザが検索要求に適合・非適合かを判断した文章を基に生成される。ベクトルの生成法は文書ベクトルと同じで、ユーザが評価した文章を形態素解析し、名詞を抽出する。抽出した名詞の出現頻度を用いて tf・idf 法を用いて抽出された名詞に重み付けをする。その重みを要素としたベクトルを生成するといった流れで生成が行われる。適合文章は追加の度に基の検索要求

ベクトルに加算し、非適合文書は追加の度に減算する。最初の検索要求ベクトルはユーザが評価した文章に含まれる N 個の名詞の重み $w(1) \sim w(N)$ を用いて以下の式で表現される

$$\text{検索要求ベクトル} = (w(1), w(2), \dots, w(N)) \quad (4.3)$$

そして次に追加された文章の検索要求ベクトルも同様に作成され元の検索要求ベクトルに、適合文章の場合加算し、非適合文章の場合減算する。よって検索要求ベクトルは以下の式のように構成されていく。まず適合文章を追加した場合は以下の式で検索要求ベクトルは更新される。

$$\text{検索要求ベクトル} = \text{元の検索要求ベクトル} + \text{適合文書ベクトル} \quad (4.4)$$

$$\text{検索要求ベクトル} = \text{元の検索要求ベクトル} - \text{非適合文書ベクトル} \quad (4.5)$$

これにより、ユーザが文章の適合・非適合を判断するたびに検索要求ベクトルは更新されていくため徐々にユーザの検索要求を満たす表現を可能とする。

4.5 類似度の計算

文書ベクトルと検索要求ベクトルの類似度の計算方法について述べる。まず適合・非適合文章が評価されるとシステムは検索要求ベクトルを作成、または更新を行い、ページベクトルと同じ N 次元空間上に配置する。そしてコサイン類似度を計算することによってランキング i 番目のページベクトル D_i と検索要求ベクトル r_k の類似度 $\text{sim}(r, D_i)$ をそれぞれ以下の式で計算する。

$$\text{sim}(r, D_i) = \frac{\sum_{k=1}^N w(i, k) \cdot r(k)}{\sqrt{\sum_{k=1}^N w(i, k)^2} \cdot \sqrt{\sum_{k=1}^N r(k)^2}} \quad (4.6)$$

この式で全ての文書ベクトルと検索要求ベクトルの類似度を計算し、類似度が高い、すなわち類似度が高いベクトルとして、類似度順にページを並び替えリランキングをする。その際ソートを行う方法はクイックソートを用いることで高速な並び替えを行っている。類似度の直感的な理解のため例を挙げ図を用いて文書ベクトル D_i と検索要求ベクトル q のベクトル空間上の関係を示す。例として単語『Java』、『プログラミング』、『コーヒー』という三次元のベクトル空間を図 4.5 に示す。図 4.5 における文書ベクトル D_i は Java コーヒーについて書かれたページのベクトルで、検索質問 q は『Java プログラミングに関する資料』とする。これらはベクトル空間上では図のように表現される。

4.6 検索要求の再構成

リランキングインタフェース Ensemble Search では、検索要求の再構成、つまり以前に得られたリランキング結果に戻す機能、検索要求ベクトルを構成する文章を検索要求ベクトルか

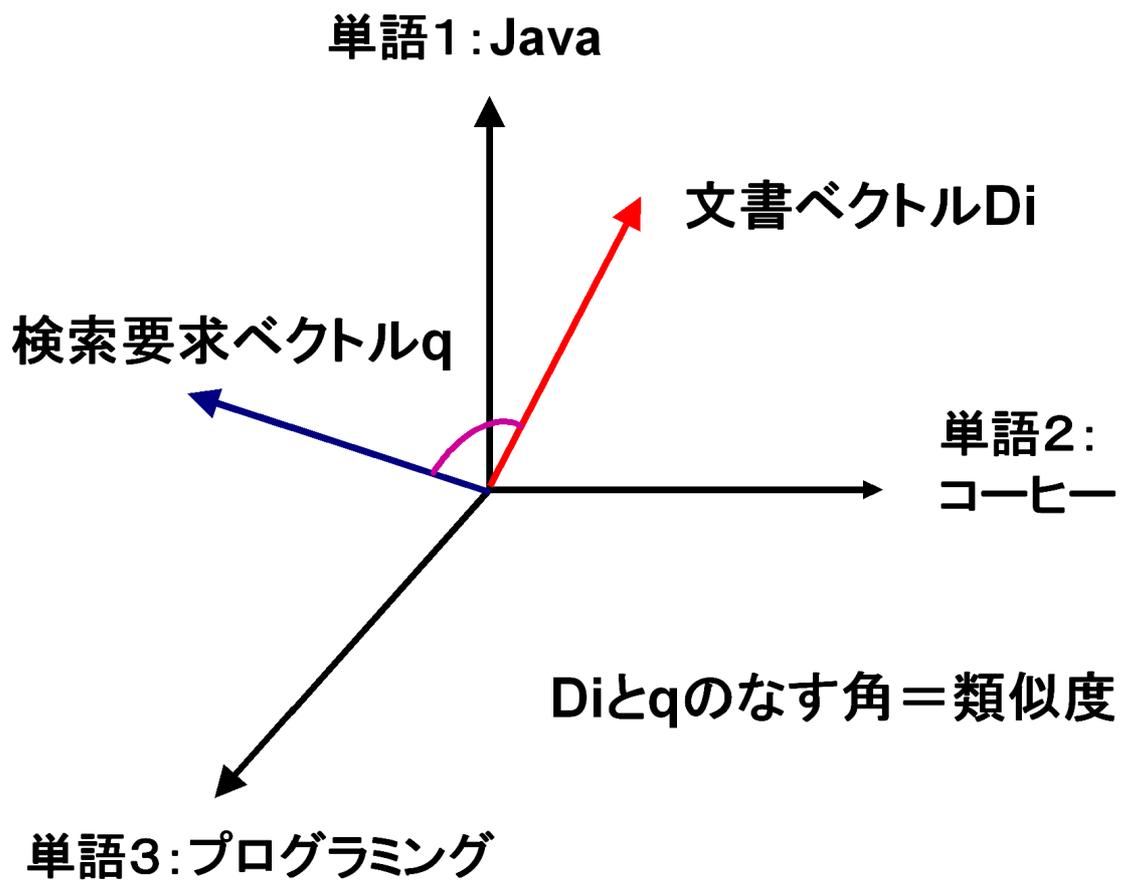


図 4.5: ベクトル空間上の文書ベクトルと検索要求ベクトルの関係

ら除外する機能を提供することを支援する。検索要求の再構成をどのような操作を用いて行うかについて以下に図 4.6 を用いて説明する。

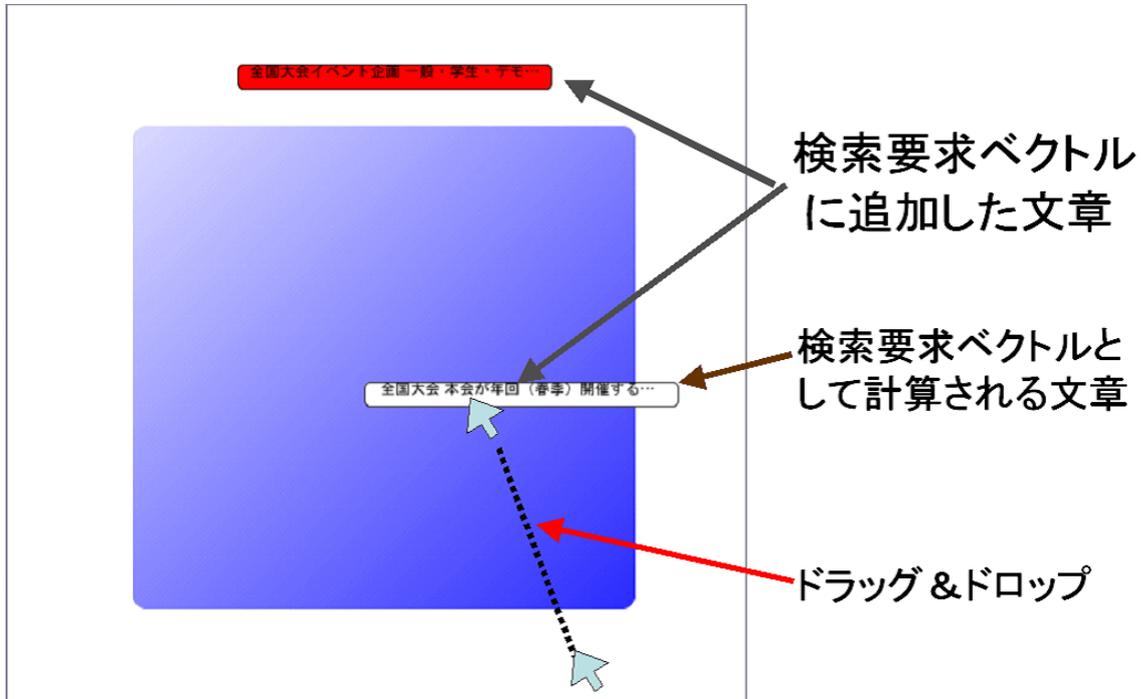


図 4.6: 検索要求再構成を行う操作

検索要求再構成に用いる操作もドラッグアンドドロップを利用する。ユーザは検索要求再構成部で今までに追加した文章を確認することができる。検索要求再構成部については実装の章で詳しく説明する。その文章片オブジェクトをドラッグアンドドロップにより検索要求ベクトルとして計算される領域とそうでない領域のどちらかに自由に動かす、検索要求ベクトルとして計算される領域は図 4.6 の内側に示される青の領域である。またその外側の白い領域は検索要求として計算しない領域になっているためユーザが追加した文章片を検索要求ベクトルから除外したいと思った場合は、文章片オブジェクトを図 4.6 の外側の領域にドラッグアンドドロップすることによって検索要求の再構成が可能である。また逆の操作も可能であり、一度除外したが気が変わりまた検索要求に加えたいと思った場合は、外の領域から内側の領域にドラッグアンドドロップすれば良い。文章片オブジェクトがどちらの領域内にあるかを確認するための視覚的なフィードバックとして、内側の領域内にある場合は文章片オブジェクトは白で、外側の領域にある場合は赤で表示される。この機能によりユーザの気が変わった、またはリランキング結果を元に戻したいと思った場合に備える。なお文章片オブジェクトがドラッグアンドドロップにより一方の領域からもう一方の領域に移動された場合、リランキング結果はドロップ操作の終了と同時に動的に変化する。また一画面性を損なわない表示方法も検索要求再構成部の特徴の一つである。

4.7 利用シナリオ

本稿で提案するリランキングインタフェース Ensemble Search が効果を発揮し、有効的に機能する場面について説明する。Ensemble Search は多義語や、同姓同名の人物の名前などを検索クエリとして検索をする場合などに効果を発揮する。そのような検索クエリで検索エンジンで検索を行った場合、検索結果にはその単語は含むが、複数の意味を持つ検索クエリのためシステムはどの意味の結果を返せばいいかわからず、ユーザにとって欲しい情報と欲しくない情報が混在してしまう。そこで Ensemble Search を用いることで、ユーザは対話的に検索結果に含まれる、または閲覧しているページの文章片の自分の検索要求への適合・非適合の評価をシステムに伝える事で、システムがユーザが欲しい情報を理解し、リランキングをすることにより欲しい情報のみを上位に集める。これによってユーザは検索結果に含まれる多くの要らない情報を閲覧することなく、欲しい情報のみを閲覧できる。例え不必要な情報を閲覧してしまっても、それを非適合文章としてシステムに伝えることにより、よりそのユーザ好みのリランキング結果を得ることができる。Ensemble Search が利用されるシナリオを以下に述べる。ユーザ A は冬になり、防寒具であるマフラーの購入を考えている。そこでネット上でオシャレなマフラーを探して購入することにした。しかしユーザ A はオンラインショッピングの経験は浅く、どのサイトで購入を検討したらいいかわからない状態にある。なのでユーザ A は検索エンジンで『マフラー』『通販』という検索語を入力し、検索を行った。しかし、検索結果を見てみると、防寒具である『マフラー』の通販サイトは含まれているものの、車やバイクの排気管である『マフラー』の通販サイトも同様に多く含まれていた。そこでユーザ A は二つの『マフラー』の差別化を計る検索クエリを考えしたが、なかなか良い単語が思いつかない。このような場合、ユーザ A は検索結果を見ながら、自分でこのサイトは欲しい情報、このサイトは欲しくない情報と見分ける作業をするか、二つのマフラーを差別化する単語を思いつくまで考えなければならない。そこで Ensemble Search を使うと、検索結果を見ながら判断した欲しい情報、欲しくない情報をシステムに伝える操作をするだけで、欲しい情報を含むページのみが上位にランキングされる。するとユーザは上位の結果のみを見て、マフラーの通販サイトを比較し、購入するサイトを検討しやすくなる。またあるニュースに関する情報を探しているユーザ B はある程度自分の欲しい情報が表示されているページを見つけたが、よりその情報について詳細に書かれたページを閲覧したいと感じた時 Ensemble Search を使うと、ユーザがある程度自分の検索要求を満たした情報（文章）を Ensemble Search で評価することにより、同じニュースについて書かれたページがリランキングによって上位に表示され、再検索やランキングの下位の方まで検索結果を閲覧することなく目的の情報を探すことができる。

第5章 実装

提案手法、提案インタフェースを実現するために実装を行った。図 5.1 にシステム構成図を示す。

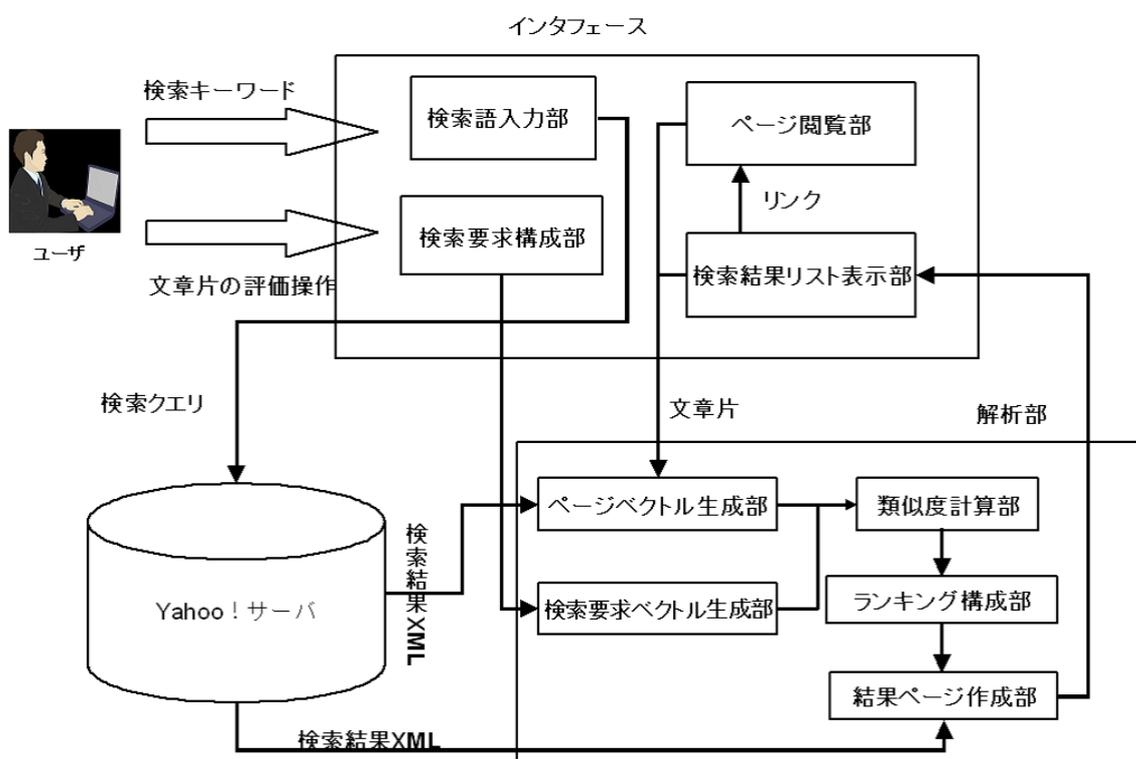


図 5.1: システム構成図

本システムはウェブ検索結果のリランキングに焦点を当てた。かつ日本語のページのリランキングに絞ってリランキングを行うことができる。ウェブ検索エンジンは yahoo! を用いている。本章では提案インタフェースの実装方法について詳しく説明する。提案インタフェースは 4 つの部で構成されている。提案インタフェースの概要を図 5.2 に示す。

提案インタフェースは以下の 4 つの部で構成される。

検索語入力部

ページ閲覧部



図 5.2: 提案インタフェース画面構成

検索結果リスト表示部

検索要求構成部

検索要求再構成部

以上の 5 つの部について以下に詳細を述べる。

5.1 検索語入力部

検索語入力部ではユーザが検索クエリをひとつ、もしくは複数の単語で表現し、検索エンジンに送る。その際 yahoo! のウェブ検索 Web サービスを使用し yahoo! から検索結果を XML 形式で取得する。そしてその XML を解析すると同時に検索結果リスト表示に表示するための HTML を作成する。加えて XML を解析し、ページそれぞれに対してベクトル空間法を用いて、文書ベクトルを生成し、N 次元空間上に配置する。ページ i の文書ベクトル d_i は以下の式で表現される。

$$d_i = (w(i, 1), w(i, 2), \dots, w(i, N)) \quad (5.1)$$

それぞれの要素は単語一つの重みである。ベクトル空間の次元数 N は取得した検索結果のタイトルとスニペットに含まれる全ての名詞の数である。

5.2 ページ閲覧部

ページ閲覧部では、検索結果のリスト表示部に表示されたページのリンクを受け取り、クリックされたページを表示する。また、ページ閲覧部内のリンクもページ閲覧部で受け取り、表示される。これにより、今までのリランキングインタフェースではできなかった一画面でページの閲覧、リランキング結果の確認を平行して行え、インタラクティブな評価とスムーズな検索作業が可能となる。加えてページ閲覧部に表示されるページ内の文章は後述する検索要求の構成に使用することもでき、より柔軟な検索要求の構成をすることができる。

5.3 検索結果リスト表示部

検索結果リスト表示部では、まずユーザが検索語入力部に入力し、検索ボタンを押す、または Enter キーを押すという操作後検索エンジンから得られた検索結果を解析し、検索結果ページを作成し、表示する。ユーザはこの検索結果ページ表示部を閲覧するという作業をする。そして検索結果にランキングされるページのタイトル、スニペット、URL からユーザ自身が気に入ったページのリンクをクリックする。すると検索結果リスト表示部はそのリンクをクリックしたというイベントとリンクの URL をページ閲覧部へ渡す、そしてページ閲覧部にリンクをクリックしたページが表示される。ユーザはこの作業を繰り返し行うことで、検索結果のページにランキングされるページを閲覧していく。既存のリランキングインタフェースでは、リランキングの手法については焦点を当てているが、ページを閲覧する場合、ブラウザが立ち上がったり、新しいタブが開き、ページを表示するという手法をとっているためリランキング結果のページと見比べ、確認するという作業をするためには、画面遷移を余儀なくされる。検索結果のページとページ内の文章を見るためには、このような画面遷移を繰り返し行う必要がありユーザにとっては大変な負荷になる。このような負荷を削減するため Ensemble Search ではページ閲覧部と検索結果のリスト表示部を並列にならべるといった画面構成、またリンクイベント、リンク URL の転送を行うという工夫をすることで、リランキングするだけでなく、ユーザのページ閲覧行動を一画面内で行うことを可能としている。

5.4 検索要求構成部

検索要求構成部ではユーザからの文章の適合・非適合の操作(図 5.3 参照)を受け取り、その文章を解析し、検索要求ベクトルの生成法で述べた手法で検索要求ベクトルを構成していく。この評価を繰り返し、漸次的に行うことで、徐々に検索要求を満たしていく。

5.5 検索要求再構成部

検索要求再構成部では、ユーザがリランキング結果を確認、閲覧した後の検索要求の変化への対応をする。ユーザは一度、もしくは複数回、文章を検索要求構成部へ追加し、リラン



図 5.3: 文章の評価方法

キングした後、その結果を確認し、閲覧をする。その時、思ったようなランキング結果は得られたが、ランキングされた結果の中の文章を読み、もっと良いランキングができると思われる文章、もしくは単語を見つける場合が考えられる。そのような時ユーザは以前のランキング結果をもう一度見て、違うランキングを行いたいと感じられる。検索要求再構成部はそのような要求が生まれた場合を想定し実装した。図 5.4 に検索要求再構成部の概要を示す。

大坪よる論文に書かれているように人間の興味は固定化されたものではなく、その時点でユーザが目にするもの、すなわちシステムが提示した情報によって影響を受け変化し続けるものである。検索要求再構成部はこのような「気が変わる」という起こりうる場合に備えた機能である。

Ensemble Search の特徴で述べたように、提案インタフェース Ensemble Search では、一画面内で検索要求を満たす表現が可能であるが、検索要求再構成部については一画面内という制約に基づき実装に工夫を加えた。ユーザは適合・非適合文章片を検索要求から除くため、検索要求構成部にポインタを一定時間留める。すると画面構成が図 5.5 下のように変わり、検索要求構成部が大きく表示される。この状態の検索要求構成部にはそれまでに追加された全ての適合・非適合文章片がオブジェクトとして表示される。一度追加した適合・非適合文章片を取り除くには対応するオブジェクトを周辺領域までドラッグするれば良い。この時にもリランキングが行われる。マウスが検索要求構成部を離れると元の状態図 5.5 上に戻る。

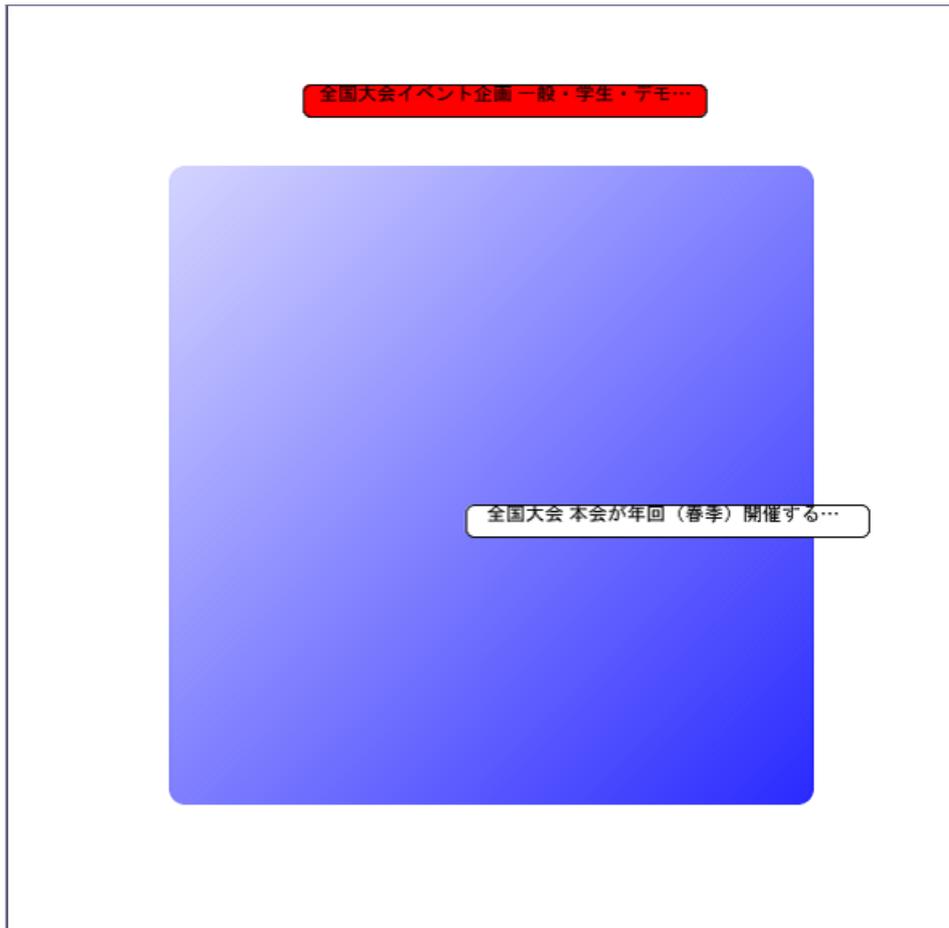


図 5.4: 検索要求再構成部

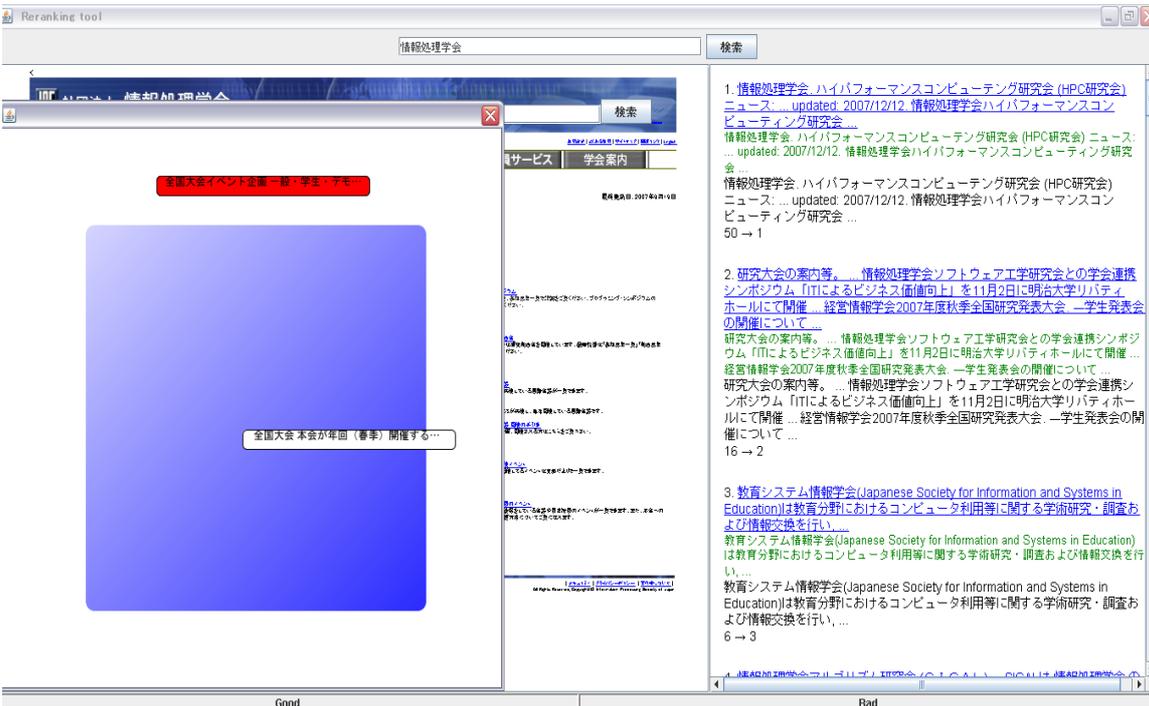
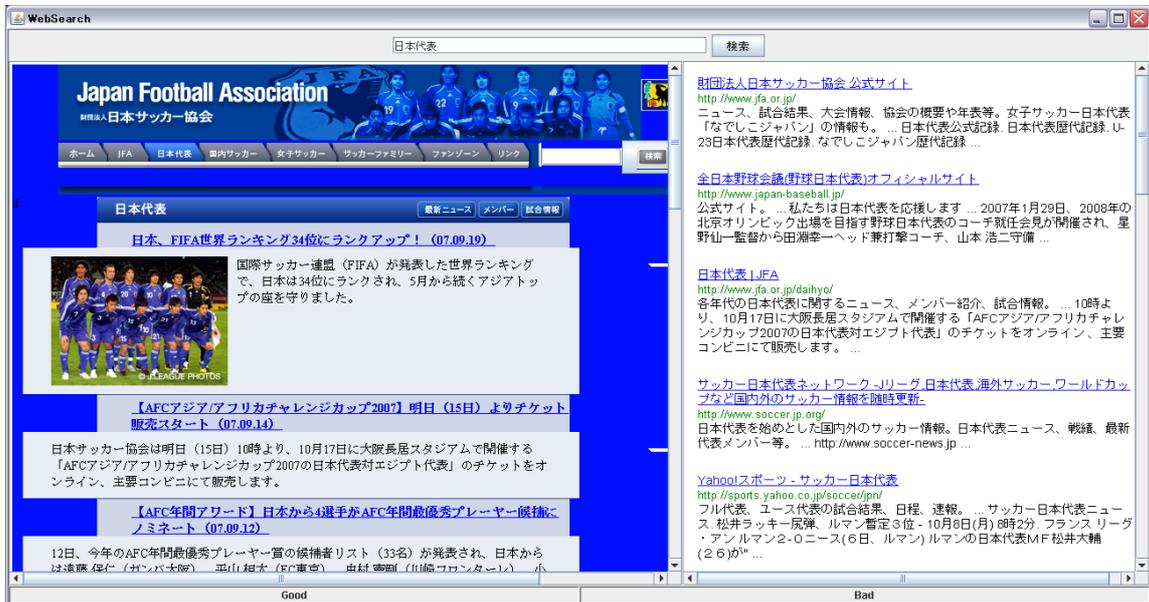


図 5.5: 検索要求再構成部の表示

第6章 評価

本章では、Ensemble Searchの有効性を示すためユーザによる被験者実験を行い、関連研究の章で紹介したリランキングインタフェース Rerank.jp との比較を行った。評価実験を通じて Web 検索エンジンから得られる検索結果をリランキングにより欲しい情報を上位へ集めることに対して有効であるか、また、かえってユーザへの負担になってしまっている部分はないかについて確認を行い、今後のインタフェースの改善へ役立てることとする。

6.1 実験内容

本稿で提案するリランキングインタフェース Ensemble Search において適合・非適合文章片を用いてリランキングを行った場合と関連研究の章で紹介した Web 上で公開されている Rerank.jp で単語をリランキングに用いる手法との比較実験を行った。実際に行ったタスクの流れを以下に示す。まず、ユーザは与えられたクエリの検索結果 100 件をシステムから提示される。検索結果を受け取ったユーザはあらかじめ決められた分類に従いリランキングを行う。今回実験に用いたタスクは、正解セット作成の容易さを考慮し、検索結果を二つに分類可能な多義語をタスクとして用いた。Ensemble Search におけるリランキングはドラッグアンドドロップを用いて行い、リランキング回数の増加によるリランキング結果上位 20 件の適合率の推移を調べた。Rerank.jp におけるリランキングは単語の強調操作、削除操作を用いて同様にリランキング回数の増加によるリランキング結果上位 20 件の適合率の推移を調べた。加えて Ensemble Search と Rerank.jp のリランキング回数の増加に伴うリランキング結果上位 20 件の適合率の推移を比較し、本リランキングインタフェース Ensemble Search の有用性を示す。また検索結果から単語を検索要求として追加する手法と文章を検索要求として表現する手法の差別化を行うため検索結果、リランキング結果を提示されてからユーザが単語、あるいは文章を評価するまでに要した時間をリランキングごとに取得し比較を行った。検索の終了条件としてリランキング結果上位 20 件の適合率が 90 % に達するか、リランキング回数が 8 回に達した時にタスクを終了することとする。なお、前提条件として検索エンジンで得られる結果上位 100 件以内にすべての解答セットが含まれているという仮定を設けた。今回用いた検索クエリは『マフラー』『通販』という二つの単語を用いて形成される。『マフラー』という単語には首に巻くファッションや防寒具としての『マフラー』ともう一つ車やバイクの排気の役目をする『マフラー』という二つの意味を持つ。検索エンジンから返される検索結果において『通販』という単語はどちらの『マフラー』についてもあり得る。よって検索結果の中には首に巻く『マフラー』の通販サイトと車やバイクの排気管の『マフラー』の通販サイ

トが混在する。この中からユーザには首に巻くほうの『マフラー』の結果のみをリランキングにより上位にランキングされるように検索要求を構成するというタスクを行ってもらった。

6.2 評価指標

本節では評価実験で用いた評価指標について説明する。

6.2.1 リランキング結果上位20件の適合率

リランキングにおいて重要なのは、ユーザの検索要求に適合しているサイトを上位にランキングさせることである。よってリランキング結果にはより多くの適合文書が含まれることを評価する指標が必要である。一般的に検索システムの評価に用いられる適合率 (Precision) について図 6.1 を用いて説明する。

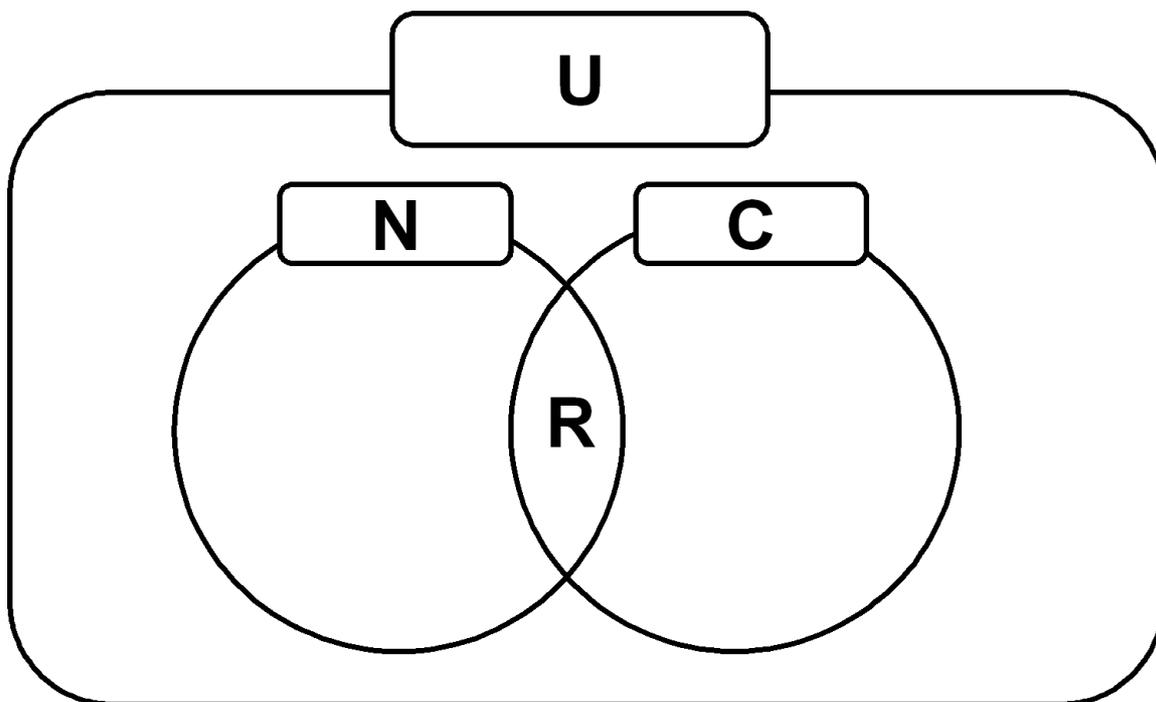


図 6.1: 検索結果のベン図

図 6.1 において U は検索対象となる文書すべて、N は検索結果の文書の数、C は全対象文書中の正解文書の数、R は検索された適合文書の数である。適合率は以下の式によって求められる。

$$\text{適合率} = \frac{R}{N} \quad (6.1)$$

しかし、今回の評価実験では、検索エンジンから得られる結果の上位100件以内にすべての解答セットが含まれているという仮定を設けているためこの指標を上位20件に適用し、評価を行う。20件としたのは、検索エンジンで一度に見られるページは20件であり、それらが検索要求に適合した文章で埋まれば検索要求を満たしたと判断することとし、上位20件に設定した。

6.2.2 結果を見てから単語または文章を評価するまでに要した時間

この評価指標は、試験的に行った実験を通して得られた、単語を検索キーとして追加する手法に比べ、本手法の文章片を検索キーとして追加する手法のほうが、適合・非適合の判定が楽であるという結果からそれを評価できる指標として検索結果、またはランキング結果をシステムが提示してから、ユーザが単語、または文章を評価するまでに要した時間を計測し評価の指標とした。

6.3 実験結果

本節ではユーザごとの実験結果に加え、被験者全ての平均値をランキングインタフェース Ensemble Search と Rerank.jp とで比較した結果を示す。

6.3.1 リランキング結果上位20件における適合率の推移

図 6.2 ~ 図 6.6 にユーザごとのランキング結果上位20件における適合率の推移のグラフを示す。

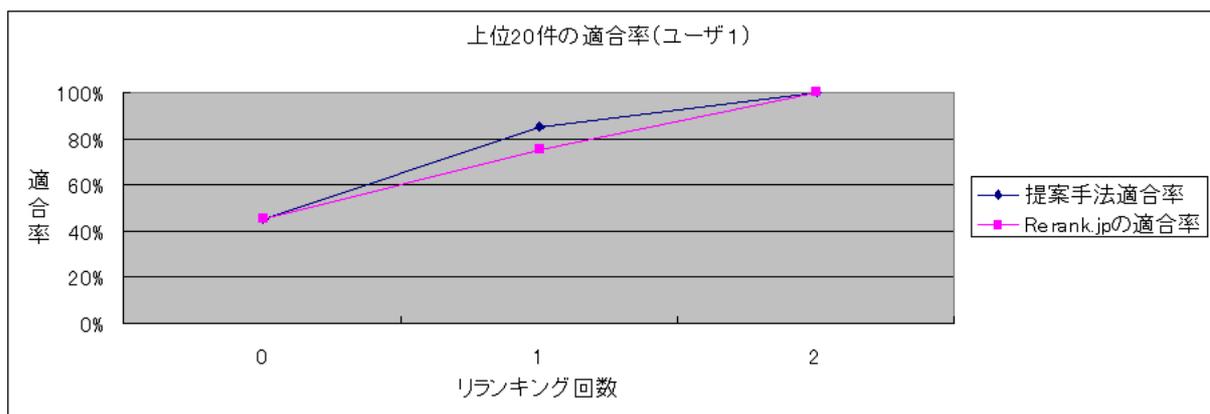


図 6.2: リランキング結果上位20件における適合率の推移 (ユーザ1)

なおリランキング回数の0回目は検索エンジンから得られたランキング結果の上位20件の適合率である。提案ランキングインタフェース Ensemble Search と Rerank.jp を比較し

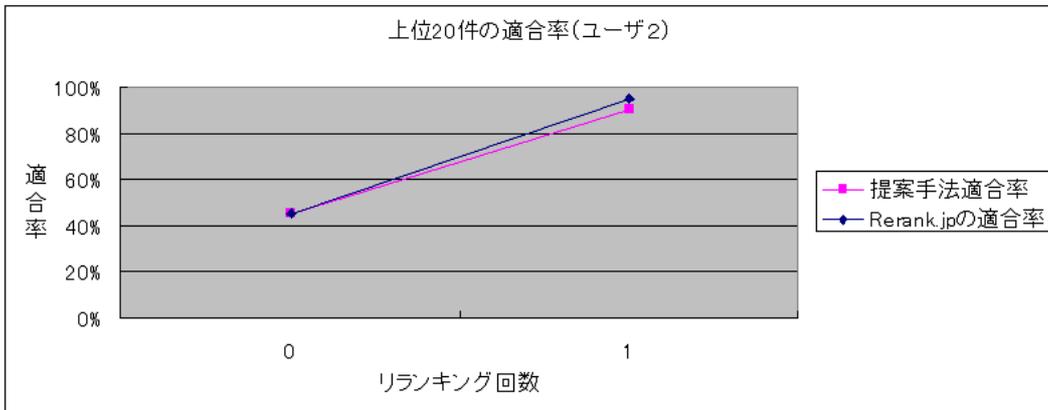


図 6.3: リランキング結果上位 20 件における適合率の推移 (ユーザ 2)

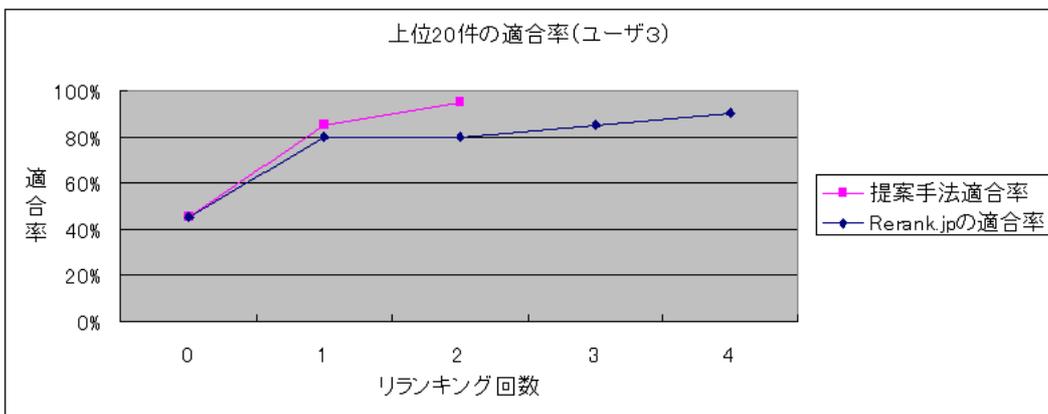


図 6.4: リランキング結果上位 20 件における適合率の推移 (ユーザ 3)

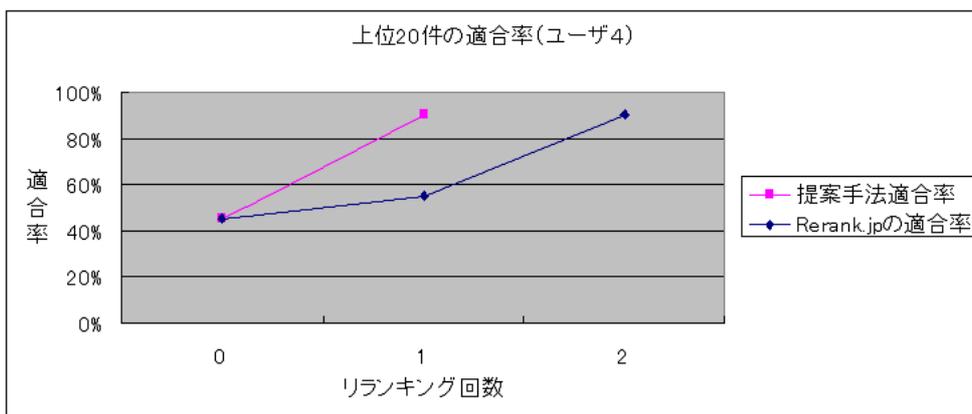


図 6.5: リランキング結果上位 20 件における適合率の推移 (ユーザ 4)

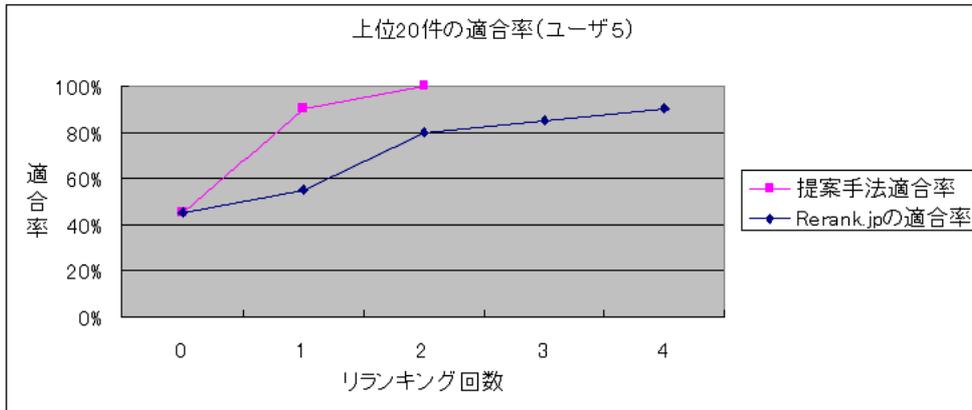


図 6.6: リランキング結果上位 20 件における適合率の推移 (ユーザ 5)

た結果としてユーザごとに多少のばらつきはあるが、Rerank.jpのリランキング結果の上位20件の適合率に比べリランキングインタフェース Ensemble Search の上位20件の適合率が同等かそれ以上の適合率を達成していることが見て取れる。さらに5人のユーザのリランキング結果の上位20件の適合率の平均値の推移グラフを図6.7に示す。

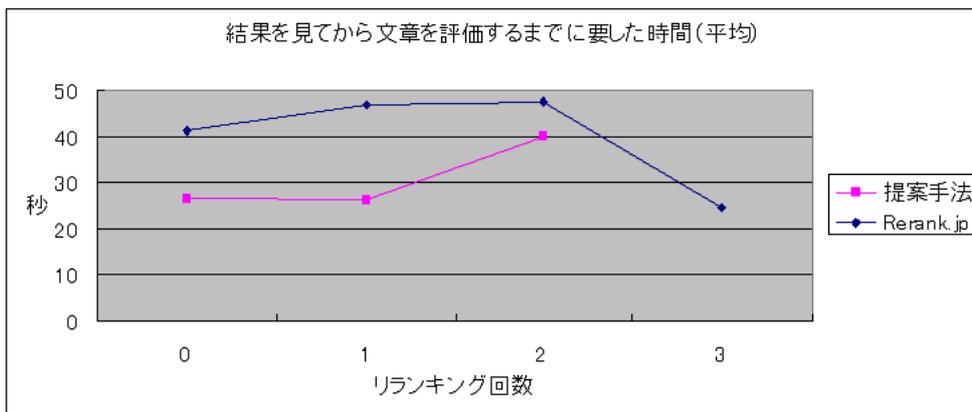


図 6.7: リランキング結果上位 20 件における適合率の推移 (平均)

図 7.6 からも見取れるように提案インタフェース Ensemble Search によるリランキング結果上位20件の適合率の平均値はRerank.jpの結果を上回る結果になっている。このことから提案リランキングインタフェースによるリランキングのほうがよりユーザの検索要求を適切に表現可能であり、よりユーザの検索要求に合った検索結果を少ないリランキング回数で多く得ることが可能であることが証明できた。

6.3.2 単語または文章の評価に要する時間

本節では単語を検索質問として追加し、リランキングする手法に比べ、本提案手法の文章を検索質問として追加し、リランキングすることの有効性を示すため、単語、または文章の評価に要する時間について調査した。これは検索結果、またはリランキング結果をユーザが目にしてからどの単語、またはどの文章が自分の検索要求に適合しているかを判断するまでにかかる時間で、検索要求として単語、文章のどちらが検索要求として追加しやすいのかを示す指標となる。結果として5人のユーザの単語、文章の評価に要した時間を図 6.8～図 6.12 に示す。

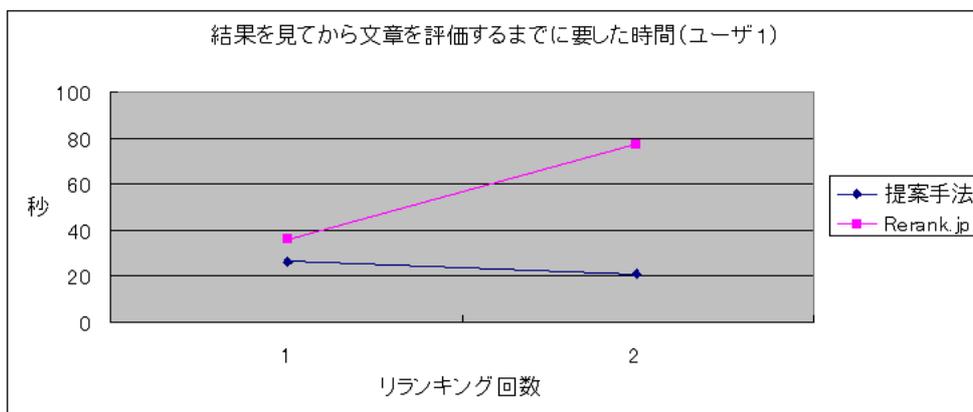


図 6.8: 単語または文章の評価に要した時間 (ユーザ 1)

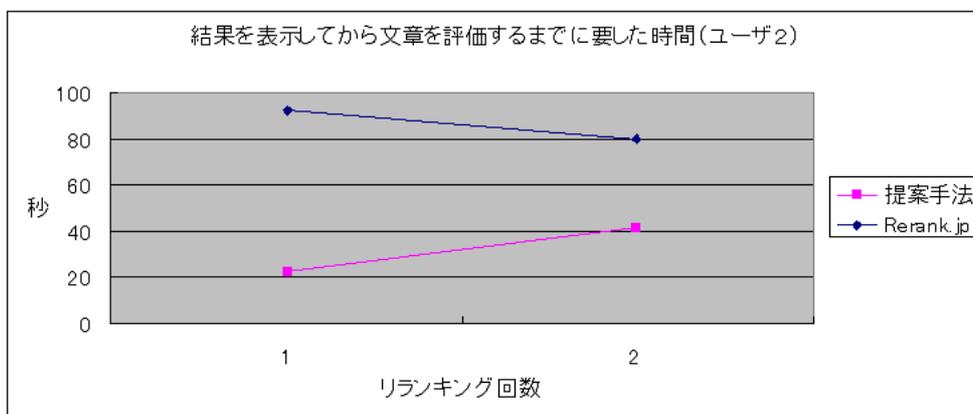


図 6.9: 単語または文章の評価に要した時間 (ユーザ 2)

それぞれのユーザによる単語、または文章の評価に要した時間はユーザ 3 を除いては提案インタフェースを用いて文章を評価する場合のほうが短いことがわかった。さらに5人のユーザの単語、または文章の評価に要した時間の平均値のグラフを図 6.13 に示す。

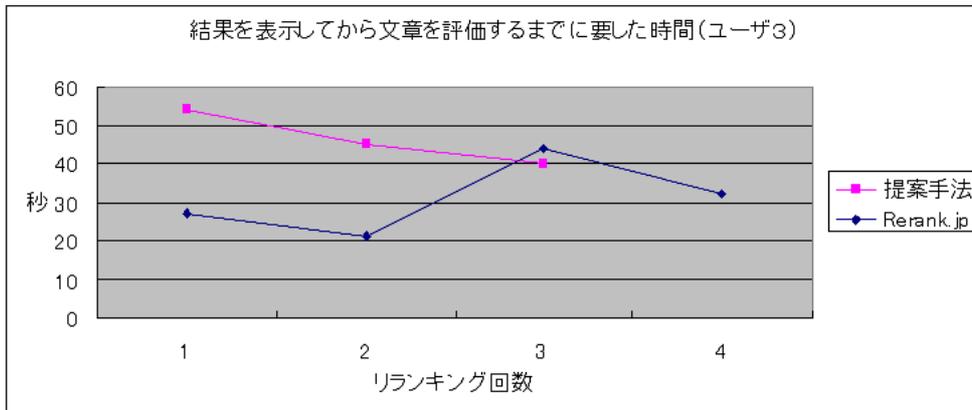


図 6.10: 単語または文章の評価に要した時間 (ユーザ 3)

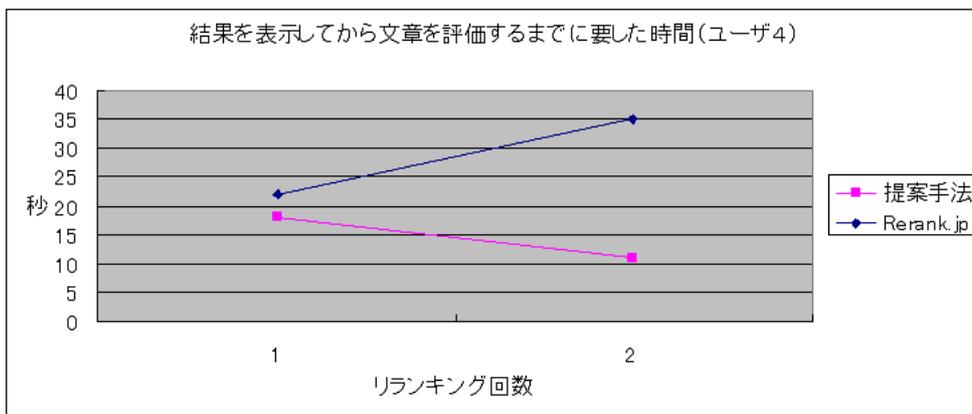


図 6.11: 単語または文章の評価に要した時間 (ユーザ 4)

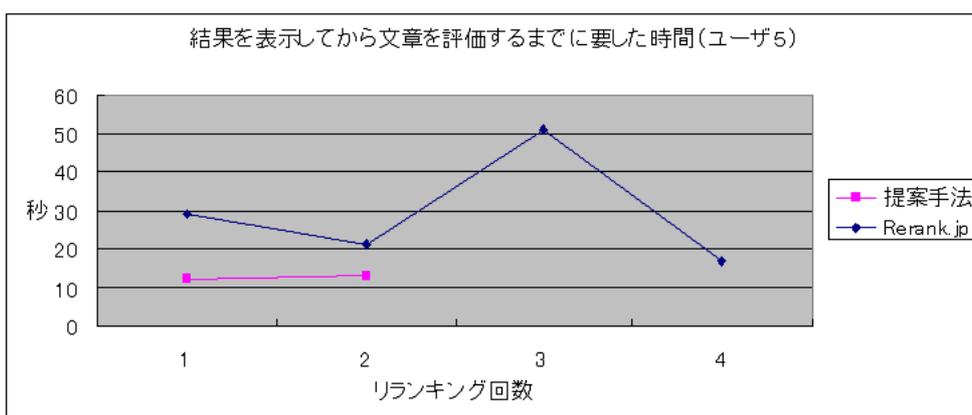


図 6.12: 単語または文章の評価に要した時間 (ユーザ 5)

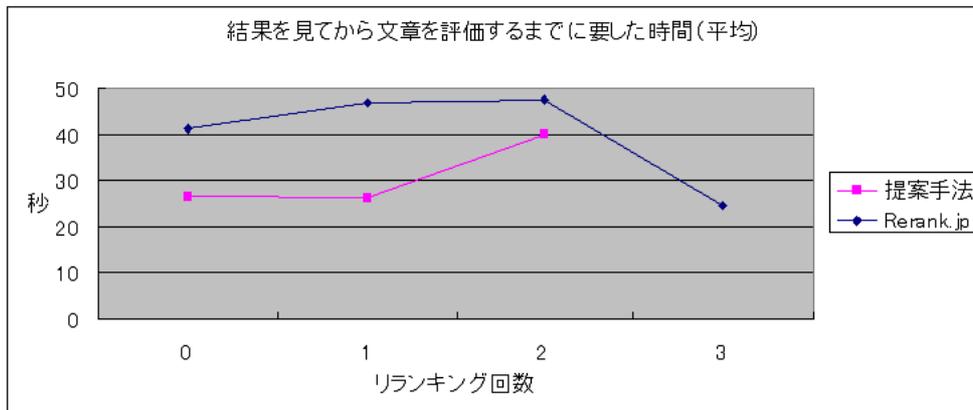


図 6.13: 単語または文章の評価に要した時間 (平均)

この結果より平均的に提案インターフェース Ensemble Search を用い、文章を評価するほうが Rerank.jp で単語を評価するのに比べて短い時間で評価していることがわかる。

6.4 評価実験において得られた意見

以下に評価実験を通じてユーザから得られた意見を示す。

- 画面構成が直感的でわかりやすい
- 文章片を評価する操作がリランキング結果が動的に反映されるためシステムとの対話性を感じられる
- 複数の文章片を一度に選択してリランキングしたらどうか
- 類似した文章をわかりやすく示して欲しい

今後はこれらの点について考察を深め、改善点、さらなる機能追加を検討していく。

第7章 考察

評価実験から得られた結果から Ensemble Search の利点とユーザからの意見を基に今後の改善点について考察する。

7.1 文章片を用いたリランキングインタフェースについて

被験者を使った評価実験では、単語の評価に基づきリランキングを行う Rerank.jp との比較を行い、文章片を用いたリランキングインタフェースである Ensemble Search を用いてリランキングを行う本手法のほうがリランキング結果上位 20 件の適合率が平均的に高いことを示した。リランキングでは上位にユーザの意図した結果が集めユーザに検索結果の閲覧や比較を行いやすくすることが望ましい結果であることからリランキング結果上位 20 件の適合率が高いことは、よりユーザの意図に合った検索結果が得られていること、検索要求の表現として文章片による表現が有効であることが示すことができた。一方では、単語を検索質問に加えていくリランキング手法より = A 文章を検索質問に追加していく本手法のほうが単語、または文章片の適合評価にかかる時間が短いこともわかった。これは単語を追加する場合、追加しようと考えている単語が 2 つに分類された検索結果の両方の分類に共通して現れる単語であった場合にそれを追加するか熟考する必要があることが理由として挙げられるのではないかと考えられる。しかし、文章片を検索要求に適合・非適合の評価をする場合、2 つの分類に共通に出現する単語があったとしても文章であるため、他の単語により検索要求が特徴付けられるので追加するべきかどうか熟考することなくユーザが文章の適合・非適合を評価ができる。またドラッグアンドドロップによる操作、画面構成についても直感的でわかりやすいという意見を評価実験のユーザから得ることができた。さらに提案インタフェースの画面構成についても同様の意見を得た。リランキングに馴染みがないユーザがほとんどであったが、このような意見を得られたということは、検索要求をうまく表現できないユーザに使用してもらうことを想定して設計した本研究のアプローチの意図した通りの結果となった。しかし本インタフェースを改善するに当たり考慮すべき意見も得ることができた。その点について以下に考察をする。

7.1.1 いくつかの文章を一度に追加しリランキング

Ensemble Search では、適合・非適合文章片を一つ追加するごとに動的にリランキングし、結果を提示する。この手法に関して有効だということは評価の章で述べたが、リランキング

する際に一度に複数の文章を追加し、一気にリランキングをしたらどうかという意見を得た。これについては試験的に行った実験でも同様の指摘を受けていた。例えばリランキングをする前に表示されていた検索結果のリストページで、この文章とこの文章の二つを追加したいと考えていたとしても、一つの文章を追加するとリランキングされてしまい、片方の文章を見つけるのが手間になってしまう。このような問題を解決するためには複数の文章片をペンのマーカーのように選択し、リランキングするような手法に改善することが考えられる。しかし、そこで問題になるのはシステムとユーザ間の対話である。そのような手法を適用するためにはリランキングをどのタイミングで行えばいいかといったことが問題である。もしボタンを解すような操作をキーとしてリランキングをするならば、システムとユーザ間の対話的なやりとりのスムーズさが損なわれる。よって今後はどのようなタイミングでユーザとシステム間の対話性を損なわずに複数の文章を指定してリランキングする手法を考えていきたい。この手法が実現することができればより少ないリランキング操作でユーザの検索要求に適合したランキングが作れるのではないかと思う。

7.1.2 類似した文章の提示

現在の実装では検索要求構成部に追加した適合・非適合文章片は、リランキングのためだけに用いている。しかし、文章単位での類似度の計算を行い、評価された文章片を有効活用し、それに類似した文章を提示することで、より早く望みの検索結果に辿り着くことを支援することが可能になると考える。具体的には、関連研究の章で述べたタグクラウドの文章版のようなものが考えられる。文章を提示するだけに提示方法は考えなければならない点だが、検索結果のリスト表示部において類似する文章の色を変更するなどの表示も考えられる。今後、類似した文章をうまく提示するための提示方法を検討したい。

第8章 結論

本研究では Web 検索行動と、Web 検索エンジンによる検索結果について考察し、さらに既存のリランキングインタフェースの問題点についても考察を加え、それに基づきユーザが評価した文章片を用いたリランキングインタフェース設計した。そして、検索要求を検索結果のページ内の文字列を適合文章片として選択することにより表現する手法 および検索要求を検索要求ベクトルに追加するインタラクティブな手法、インタラクティブ性を実現するための表示方法を特徴とするインタフェース Ensemble Search を実装した。そして本研究のアプローチの有効性を確かめるため評価実験を行い、既存のリランキングシステムとの比較を行った。その結果を基に本研究の有効性、今後の展望について考察を行い Ensemble Search の利点、および今後の改善点、検討すべき点を述べた。Ensemble Search は、既存のリランキングインタフェースと比較した結果、より少ないリランキング回数で高い適合率を得ることができた。これはリランキングに用いるユーザの操作回数の削減にもつながり、ユーザにとってはより少ない操作でそのユーザの検索要求に適合した情報が得られるという利点があることが明らかになった。また、リランキングにおいて単語を検索キーとして追加指定する手法に比べ、短い時間で適合・非適合の評価ができることがわかった。このことでユーザは追加指定するキーを熟考することなく追加することができることが証明できた。よって本研究のアプローチの有効性が示せた。しかし、評価実験を通じてユーザから得られた意見を深く検討し、さらなる手法の改善、機能追加を行っていきたい。また評価実験についても多くの場面について評価をしていくつもりである。

謝辞

本研究の着手および進行，本論文の作成にあたり，終始御指導いただいた筑波大学システム情報工学研究科田中二郎教授に心より感謝致します．先生には貴重な研究資料，快適な研究環境に加え研究発表の機会を提供していただき，また貴重な助言をしていただくことにより本論文をまとめることができました．そして筑波大学システム情報工学研究科志築文太郎講師には、研究の着手から、研究の内容、進捗状況、論文の執筆など研究活動全般に渡り、きめ細かに御指導いただいたことに心より感謝致します．さらに筑波大学システム情報工学研究科三末和男准教授、高橋伸講師には、研究の背景や内容のご指導、また研究に関する資料など様々なご助言、ご指導頂きました。心から感謝いたします。WAVE チームでは、ゼミの時間を通じて、研究に対する活発な議論を交わし、ゼミ以外の時間にいたっても貴重なご意見を頂いたことを厚く御礼申し上げます。また評価実験にご協力頂いた研究室のメンバーにも改めて御礼申し上げます。そして最後に自分を支えてくれた両親や、全ての友人にも感謝の意を表したい。本当に有難うございました。

参考文献

- [1] 平田 陽一, 松倉 健志, 田島 敬史, 田中 克己: Web 検索における意味的適合フィードバック機構, 情報処理学会研究報告, Vol.2000, No.69, pp. 137-144, 2000.
- [2] 酒井 浩之, 大竹 清敬, 増山 繁: 絞り込み語提示による一検索支援手法の提案, 言語処理学会第7回年次大会 発表論文集, pp. 185-188, (2001).
- [3] 山本 岳洋, 中村 聡史, 田中 克己: 編集操作を用いたウェブ検索結果の最適化, 第17回データ工学ワークショップ (DEWS2007) 論文集, L4-3, (2007) .
- [4] Page L, Brin S, Motwani R, Winograd T: The PageRank Citation Ranking, Bringing Order to the Web, Online manuscript(1998).
- [5] Jon M. Kleinberg, Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, Andrew S. Tomkins: The Web as Graph, Measurements, Models, and Methods, COCOON 99, pp. 1-17, (1999).
- [6] The PageRank Citation Ranking: Bringing Order to the Web Lawrence Page, Sergey Brin, Rajeev Motwani, Terry Winograd, (1998).
- [7] Yahoo!ウェブ検索 Web サービス <http://developer.yahoo.co.jp/search/web/V1/webSearch.html>
- [8] Meadow, C : Text Information Retrieval Systems, Academic Press(1992).
- [9] Salton, G : Automatic Text Processing , AddisonWesley , MA(1989) .
- [10] Taylor, R.S : Question-negotiation and information seeking in libraries , College & Research Libraries, 29(3) , pp . 178-194 , (1968) .
- [11] 大坪五郎 , : Gards - 変化し続ける興味に対応する情報推薦 , Proceedings of the 13th Workshop on Interactive Systems and Software, pp. 31-36, (2005).
- [12] J. Xu and W. Croft : Query expansion using local and global document analysis , Proc of SIGIR1996 , pp. 4-11 (1996).
- [13] D. Hand, H. Mannila and P. Smyth: Principles of Data Mining, Bradford Book (2001).
- [14] H. Zeng, Q. He, Z. Chen, W. Ma and J. Ma: Learning to cluster web search results, Proc of SIGIR2007, pp. 210-217 (2004).

- [15] M. Bilenko, R. Mooney, W. Cohen, P. Ravikumar and S. Fienberg: Adaptive name matching in information integration. *Intelligent Systems, IEEE*, 18, 5, pp. 16-23 (2003).
- [16] Yumoto, T. and Tanaka, K.: Page Sets as Web Search Answers, *Proceedings of The 9th International Conference on Asian Digital Libraries (ICADL2006)* pp. 244-253 (2006).
- [17] Oyama, S. and Tanaka, K.: Query Modification by Discovering Topics from Web Page Structures, *Proceedings of the Sixth Asia Pacific Web Conference (APWEB 04)*, pp. 553-564 (2004).
- [18] goo <http://www.goo.ne.jp/>
- [19] 河重 貴洋, 小山 聡, 大島 裕明, 田中 克己: 質問修正と再ランキングを用いた文脈依存 Web 検索, 第 17 回データ工学ワークショップ (DEWS2006) 論文集, 3C-i14, (2006) .
- [20] 田 馳, 手塚 太郎, 小山 聡, 田島 敬史, 田中 克己: 質問キーワードの意味的関連と近接性に着目したウェブ検索の精度改善, 第 17 回データ工学ワークショップ (DEWS2006) 論文集, 5A-o1, (2006)
- [21] 松本 裕二, 馬野 元秀, 瀬田 和久: 類似性に基づいたウェブページ選択インタフェース, 電子通信学会第二種研究会資料、(2005).
- [22] Takehiro Yamamoto, Satoshi Nakamura, and Katsumi Tanaka: Rerank-By-Example: Efficient Browsing of Web Search Results, *Proceedings of the 18th International Conference on Database and Expert Systems Applications (DEXA2007)* pp. 801-810 (2007).
- [23] Rerank.jp <http://rerank.jp/>
- [24] オプトとクロスマーケティング検索エンジン利用状況実態調査
<http://www.opt.ne.jp/news/news2006.html>
- [25] flickr. <http://www.flickr.com/>
- [26] Sen project. <http://ultimania.org/sen/>