Webページ間の特徴語類似度を用いた検索支援システム

上野研究室　中川　尊雄

本研究では，Webページ検索に習熟していないユーザの支援を目的とし，多くの正解ページが予想される検索について，形態素解析とTF-IDF法を用いて検索結果からWebページを推薦する手法を提案する．本研究では提案手法に基づいてシステムを実装し，被験者実験を行ってその有効性を確認する．

提案手法では，まず，正解の見本ページと検索結果に出現したページをそれぞれ形態素解析器にかけ，単語に分割する．次に，分割された単語からTF-IDF法によってそれぞれのWebページの特徴語を求め，最後に特徴語の類似度順に検索結果を並べ替えたリストをユーザに推薦する．

実装したシステムの有効性を確認するために行った被験者実験は，設定された検索目的に基づいて，被験者に検索結果から目的に合致するページを選択してもらい，その結果とシステムの推薦結果の比較を行った．

被験者実験の結果，システムの推薦するページとユーザが有用と考えるページの間には有意な関連がみられ，システムの検索結果が有用であることが示唆された．

目次

1. はじめに

近年，一般家庭へのコンピュータ普及とともに，インターネットの新規ユーザが過去 10 年間で 5 倍に増加している[1]．このような新規ユーザは一般的に検索に関する能力が低く，検索結果の絞り込みや，絞り込みの際のキーワードの選定が得意でない．

このようなユーザを支援すべく，これまでにも検索エンジンやユーザの検索支援の研究が行われている．中島らの研究ではユーザの検索行動について調査を行っており[2]，國貞らの研究では限定的な範囲での使用を想定した検索支援システムの提案と実験を行っている[3]．また，実働している検索エンジンに実装されている検索支援機能として，Google の類似ページ検索や関連キーワード推薦機能などがある．

しかし，これらの研究では，検索範囲を限定しない検索システムの提案が行われていない．また，実働している検索エンジン上の機能でも，ユーザが入力したキーワードを参照しないため，ユーザの意図と異なるページを推薦する場合がある．

これらの問題を解決するために，本研究ではユーザが選択した見本ページと，Google の検索結果をデータベース化したものから，見本ページとの類似度順に検索結果を並べ変えたリストをユーザに推薦するシステムを提案する．また，提案したシステムについて，広範囲な検索に対応できるかどうか，被験者実験により有効性を確認する．

提案システムのメリットとして，検索結果の並べかえによって Web ページの推薦を行うため，元の検索キーワードと関係のない Web ページを推薦しない点があげられる．

本研究では特に，ただひとつの答えを持つような検索ではなく，いくつかの正解と思われるデータが存在し，それらを多く収集したいユーザの支援を目的とする．

本研究では，提案したシステムを実装し，このシステムの有効性の確認するために被験者実験を行う．

2. 本研究の対象とする問題

本研究で解決すべき問題は下記の 2 つである.

・既存の検索支援の研究では検索範囲を限定しない検索システムの提案が行われていないこと.

・実働している検索エンジン上の機能については,ユーザが入力したキーワードを参照しないため,意図と異なるページを推薦する場合があること.

以下の節でそれぞれの問題について説明する.

2.1. 既存研究の問題点

例えば,中島らは,検索経験や検索対象への知識が検索に与える影響について調査を行い,検索対象に対する理解が深いユーザは,効率的に検索を行える専門的な検索キーワードの選定が得意であり,検索経験が豊富なユーザは検索のために次々とキーワードを追加して絞り込んでいくことを明らかにしている.また,適切なキーワードを用いた検索を行うシステムを作成する必要性についても述べているが[2],システムは実装されておらず,実装の必要がある.

國貞らは,検索エンジンの検索結果に示される二,三行程度の要約文の類似度を用いて Web 検索の支援を行うシステムを提案し,実装している[3].しかし,提案システムは医療分野に関する検索を対象としているため,広範囲な検索に対応させる必要がある.また,國貞らのシステムでは,ページの情報として要約文しか参照していないため,ページの持つすべての情報を用いていない問題もある.

2.2. 既存システムの問題点

Google で大和郡山市の中華店を探そうと「中華　大和郡山」というキーワードで検索を行った場合についての,関連キーワード推薦機能の出力例を図 1 にあげる.図 1 であげられた推薦キーワードとしては「大和郡山　殺人事件」や「大和郡山　イオン」など中華と直接関係がないキーワードがあり,元の検索キーワードにあった「中華」と言う情報が抜け落ちている.

また,同様の検索で,大和郡山の中華店のリストをあげているグルメサイトを類似ページ検索の入力とした場合の推薦例を図 2 にあげる.図 2 であげられたページとしては,他のグルメサイトのトップページなどがあるが,これはグルメサイトという情報だけが抽出されており,元の検索キーワードにあった「中華」や「大和郡山」と言った情報が抜け落ちている.

上にあげた二つの例のように,検索キーワードが推薦結果から抜け落ちる推薦では,ユーザが意図している結果と違う結果が推薦される問題がある.

図 1 Google「他のキーワード」機能の推薦例

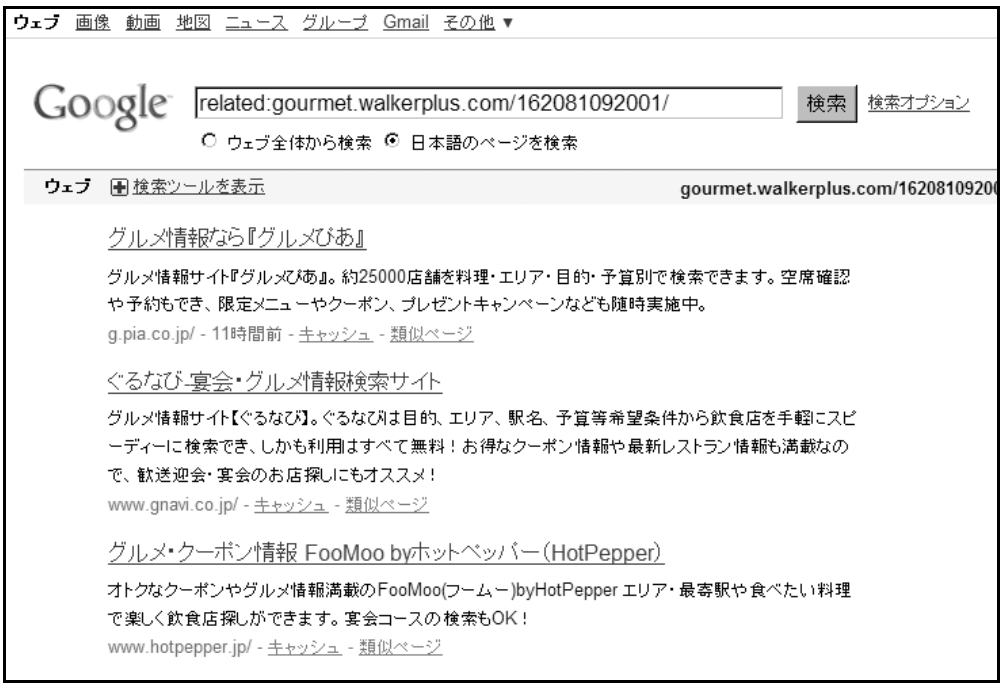


図 2 Google「類似ページ検索」機能の例

2.3. 解決方法

2.1 節と 2.2 節で述べた問題を解決する為に，本研究ではユーザが選択した見本ページと，Google の検索結果をデータベース化したものから，見本ページとの類似度順に検索結果を並べ変えたリストをユーザに推薦するシステムを提案する．

提案システムでは，検索結果の並べ替えによってページの推薦を行うため，もとの検索キーワードの情報と関係のないページの推薦を行わないという利点があり，これによって既存システムの問題点を解決する．

また，広範囲な検索が可能なシステムを実装し，被験者実験を用いてシステムの評価を行うことで，既存研究の問題点を解決する．

ただし，本研究ではただひとつの答えを持つような検索の支援は対象とせず，

いくつかの正解と思われるデータが存在し,それらを多く収集したいユーザの支援を目的とする.

例をあげると,本研究では「大和郡山市のおいしい中華店を探したい」など,複数の中華店のページやグルメサイトが結果に出るような検索の支援を対象としている.一方で,「ある出版社が出版した書籍の一覧」など,出版社のページ(正解ページ)を見ればそれで他のデータが必要ない検索に対する支援は対象としない.

3. システム

本研究では，TF-IDF 法と形態素解析を利用して Web ページを特徴づける単語を抽出，ページ間類似度を測定して類似ページを推薦する．以下でそれらの手法について説明する．

3.1. 日本語形態素解析

文書に含まれる単語の出現回数を数えるためには，その文書から単語を抽出しなくてはならない．本研究では Web ページに含まれる文書から自動で単語を切り出すために，日本語形態素解析を用いる．形態素解析とは，自然言語で書かれた文章を言語の中で意味を持つ最小単位である形態素に分割し，それぞれの品詞を特定する[4]手法である．

例として，日本語形態素解析エンジン MeCab[1]によって，「吾輩は猫である．名前はまだない．」という文章を形態素解析した結果を図 3 に示す．

本研究では，Web ページから単語を切り出すために，CRF を用いた形態素解析手法[5]を用いた日本語形態素解析エンジンである MeCab を用いる．同様のエンジンには，このほかに，ChaSen[2]や KAKASI[3]などがあるが，平均動作速度および精度の点で MeCab が優れている[5]とされるため，MeCab を選択した．

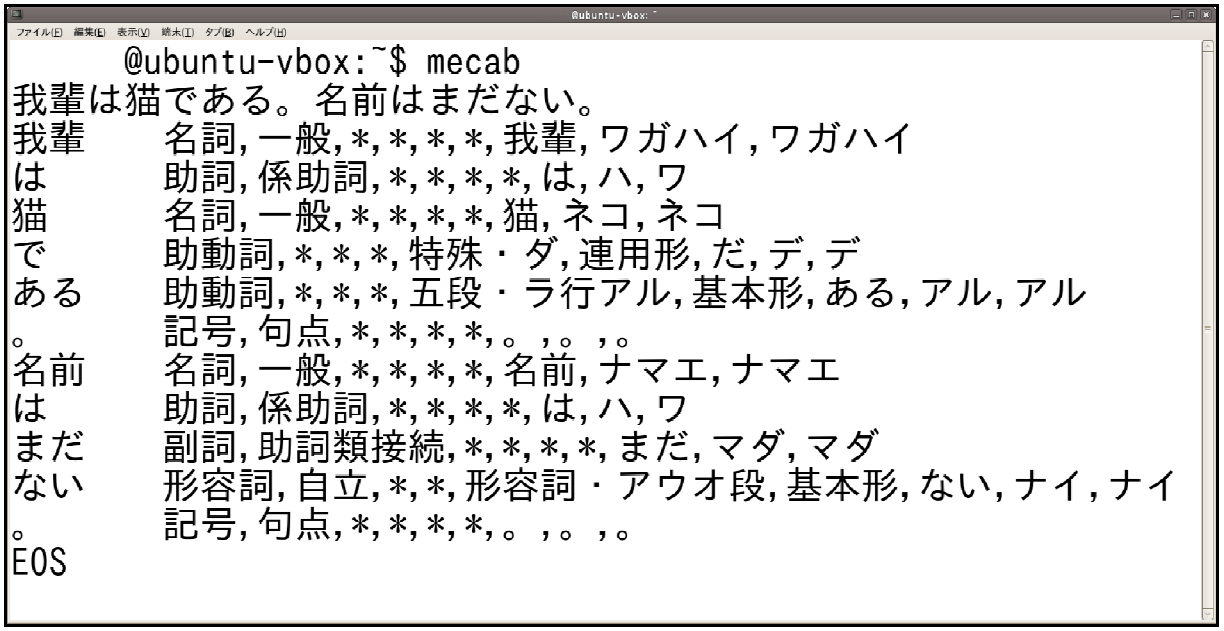


図 3 日本語形態素解析の実行例

[1] http://mecab.sourceforge.net/
[2] http://chasen-legacy.sourceforge.jp/
[3] http://kakasi.namazu.org/

3.2. TF-IDF 法

TF-IDF 法とは，ある文書の集合（文書セット）の中に含まれる一つの文書に注目したとき,その文書が文書セットの中でどういった単語(term)で特徴付けられるかを調べる手法[6]である.

この手法では，注目している文書にある単語が何度出現するかを表す TF(Term Frequency)と，その単語が文書セット中のいくつの文書に含まれているかを表す DF(Document Frequency)の逆数の対数をとった IDF(Inversed DF)の二つの値に注目して，ある文書中に含まれる単語ごとの特徴度(その単語が特定の文書をどの程度特徴付けているかを示す値）を算出する.

ある単語の TF が高ければ，その単語は文章中に多く登場することを，また同様に IDF が高ければ高いほど文書セット中においてみられないことをあらわす. これらの単語が文書を特徴付ける上で重要であるとし，式(1)によって単語ごとの特徴度を求める.

$$tf-idf_{term,document} = tf_{term,document} \times \log\left(\frac{N}{df_{term}}\right) \tag{1}$$

$tf\text{-}idf_{term,document}$:文書 document の中における単語 term の TF-IDF 値
$tf_{term,document}$:文書 document の中における単語 term の出現回数
df_{term}:文書セット全体に含まれる，単語 term が含まれる文書数
N:文書セットに含まれる文書の総数

この式で算出される TF-IDF の値は，ページにおけるその単語の特徴度の高さを示している.

本研究ではこの TF-IDF 法を用いて，複数の Web ページ間の類似度を測定する.

3.3. TF-IDF 法によるページ間の類似度の算出

本研究では,ユーザの提示した見本ページに類似したページを見つけるために，異なる二つの Web ページ間の類似度を求める方法を考える. そこで，前節の TF-IDF 法を用いてページ間の類似度を算出する.

TF-IDF 法によって得られる単語ごとの特徴度を利用して文書間の関係を求める手法は複数存在し,文書ごとに上位単語の特徴度をベクトル化した特徴ベクトルの距離を，特徴ベクトル空間で比較する方法などがある.

しかし，これらの手法ではベクトル空間の多次元化の際,計算量による限界があり，ベクトルの距離測定に用いる単語の数が限られる.

そのため，今回対象とするWebページに適用すると，Web広告などから抽出されるノイズにあたる特徴語に影響を受けやすいと考えられる．したがって，本研究では異なる文書それぞれのTF-IDFの上位25個に出た単語の中で一致するキーワードがいくつあるかを類似度の定義とした．

3.4. 類似度を用いたページ推薦

2.3節までに解説したそれぞれの手法を用いて，検索エンジンの検索結果とその中からユーザが選択した正解の見本データを利用して，類似ページの推薦を行うシステムを提案する．図4に提案するシステムの動作イメージを示す．本システムによる推薦の手順を以下に示す．

1. ユーザが行った検索の結果ページから正解の見本ページを選択させる．(図中①)
2. 検索結果に出現したページを入力用にデータベース化する．(図中②)
3. 検索結果のデータベースと，見本ページを入力とし，それぞれのWebページに対してTF-IDF法による特徴語抽出を実行する．(図中③)
4. 特徴語が一致している数によって，昇順にデータベース中のページを並び替え，このリストをユーザに推薦する．(図中④)

3.5. 作成したシステムについて

前節までに説明した手法に基づいてシステムの実装を行った．実装したシステムの構造を，図5に示す．なお，システムはRubyで書かれた231行のプログラムで，Linux上で動作する．

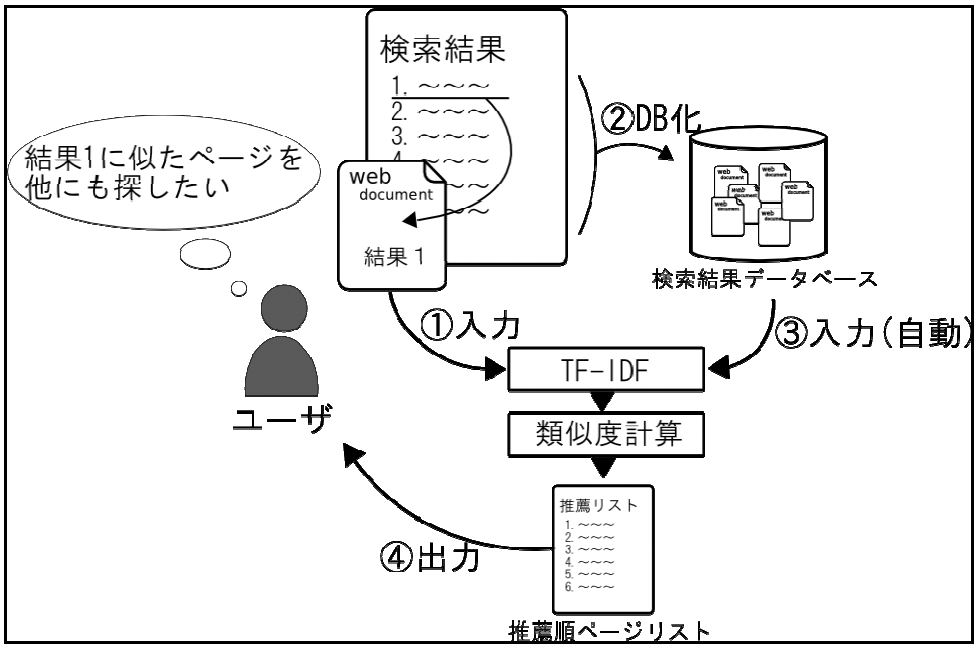


図4 提案システムの動作イメージ

システムは図 5 の中に示したモジュールを図中の A)→B)→C)→D)→E)の順番に実行する.

システムはまず,最初に入力として検索キーワード(Querykey)と正解の見本データ(Query)を受け取る.Querykey は A)Google 検索・結果解析部に送られ,このキーワードを元に Google 検索が行われる.

結果解析部は,検索結果のページから,検索結果一覧の URL から HTML ファイルを取得し,データベース(DB)として保存する.

次に,DB に格納された HTML ファイルは,それぞれ B)HTML タグ切り離し部に送られ,ファイルごとに HTML タグを切り離された文字列のデータベース(plainDB)として出力される.

その後 plainDB は,C)単語出現回数カウンターに送られる.単語出現回数カウンターは,plainDB 中の各 HTML ファイルから作成された文字列を plainText として,形態素解析エンジンに送り,形態素解析されて戻ってきた文書(segmentedText)に出現する名詞の出現回数を数え上げ,単語と出現回数の対応表(DBTF)を出力する.

一方,Query は HTML タグ切り離し部に送られ,DB と同じく,HTML タグを切り離され,単語出現回数カウンターに送られた後,単語と出現回数の対応表(QueryTF)として出力される.

QueryTF と DBTF は,同じタイミングで D)TF-IDF 計算モジュールに送られ,それぞれの URL ごとに区切られた単語と TF-IDF 値の対応表(tf-idf)の形にされる.

そして,E)類似度計算リスト並び替えモジュールで,tf-idf を用いて特徴語

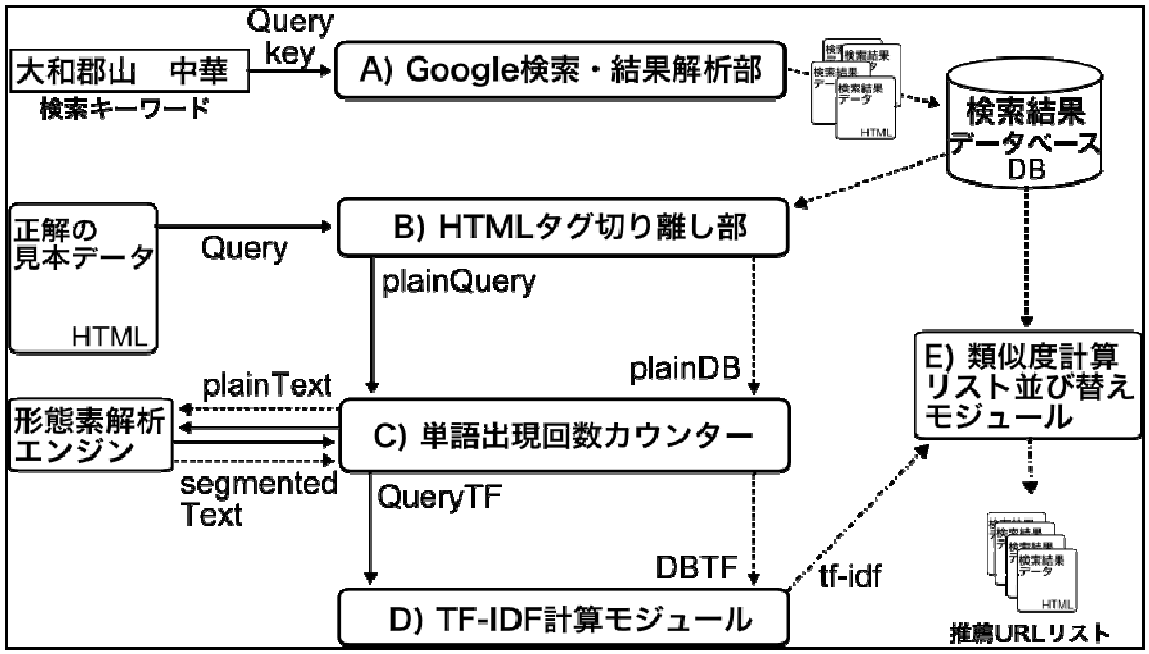


図 5 実装したシステムの構造

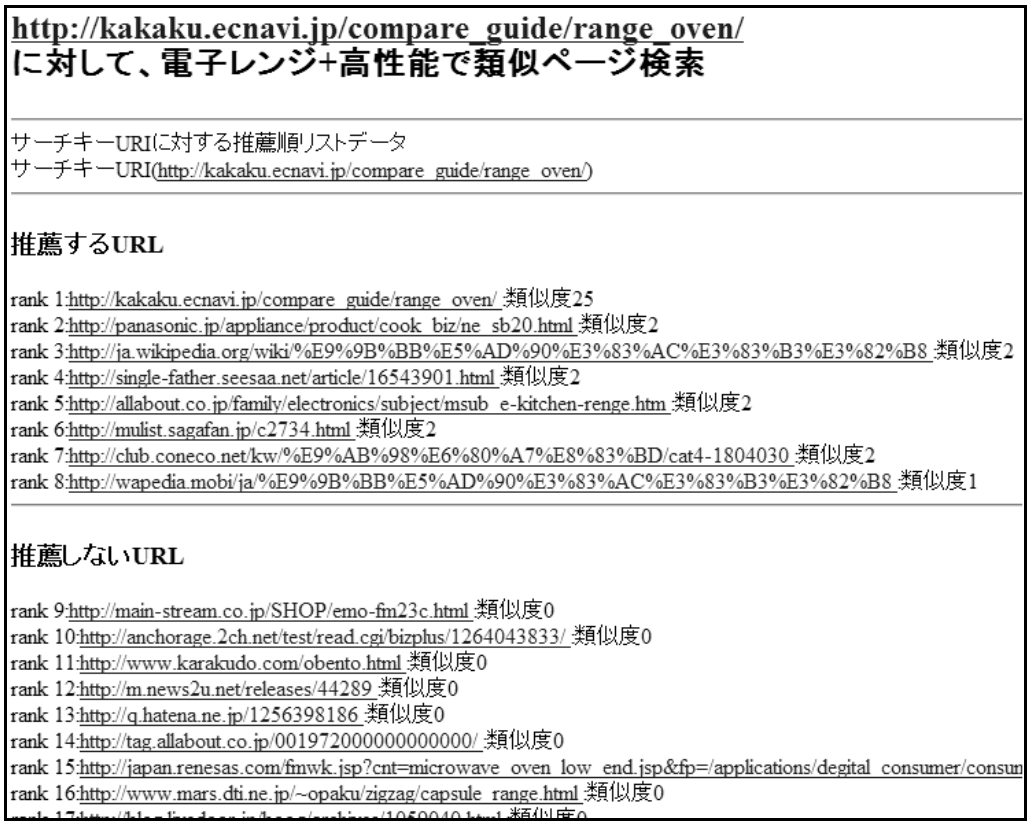
の比較を行い，DB 内の各ページの見本データとの類似度を計算する．最後に，システムは類似度を元に，DB の中の URL を類似度の高い順に並べ替え，推薦 URL リストとして出力する．

図 6 に，システムに Querykey として「電子レンジ+高性能」を，Query として「電子レンジの選び方―比較ガイド/EC ナビ[4]」を与えたときのスクリーンショットを示す．

図の最上段に示されたアドレスが，Query のアドレスで，「推薦する URL」以下 8 件がページの類似度順に並べ替えられた推薦ページリスト，「推薦しない URL」以下は類似度が 0 だったページリストとなっている．

Query は目次として「代表的なタイプと特徴」や「商品選びのポイント」などの語が含まれた Web ページである．

また，表 1 には Query から抽出した特徴語のうち TF-IDF 値が上位 10 件までの語について，TF，DF，TF-IDF をそれぞれ計算した結果を示す．表は，電子レンジの選び方を知るのに必要なキーワードである「タイプ」「選び方」「オススメ」などを抽出していることを示している．



http://kakaku.ecnavi.jp/compare_guide/range_oven/
に対して、電子レンジ+高性能で類似ページ検索

サーチキーURIに対する推薦順リストデータ
サーチキーURI(http://kakaku.ecnavi.jp/compare_guide/range_oven/)

推薦するURL

rank 1:http://kakaku.ecnavi.jp/compare_guide/range_oven/ 類似度25
rank 2:http://panasonic.jp/appliance/product/cook_biz/ne_sb20.html 類似度2
rank 3:http://ja.wikipedia.org/wiki/%E9%9B%BB%E5%AD%90%E3%83%AC%E3%83%B3%E3%82%B8 類似度2
rank 4:http://single-father.seesaa.net/article/16543901.html 類似度2
rank 5:http://allabout.co.jp/family/electronics/subject/msub_e-kitchen-renge.htm 類似度2
rank 6:http://mulist.sagafan.jp/c2734.html 類似度2
rank 7:http://club.coneco.net/kw/%E9%AB%98%E6%80%A7%E8%83%BD/cat4-1804030 類似度2
rank 8:http://wapedia.mobi/ja/%E9%9B%BB%E5%AD%90%E3%83%AC%E3%83%B3%E3%82%B8 類似度1

推薦しないURL

rank 9:http://main-stream.co.jp/SHOP/emo-fm23c.html 類似度0
rank 10:http://anchorage.2ch.net/test/read.cgi/bizplus/1264043833/ 類似度0
rank 11:http://www.karakudo.com/obento.html 類似度0
rank 12:http://m.news2u.net/releases/44289 類似度0
rank 13:http://q.hatena.ne.jp/1256398186 類似度0
rank 14:http://tag.allabout.co.jp/001972000000000000/ 類似度0
rank 15:http://japan.renesas.com/fmwk.jsp?cnt=microwave_oven_low_end.jsp&fp=/applications/degital_consumer/consum
rank 16:http://www.mars.dti.ne.jp/~opaku/zigzag/capsule_range.html 類似度0

図 6 システムのスクリーンショット

[4] http://kakaku.ecnavi.jp/compare_guide/range_oven/

表 1 Query から抽出された特徴語(上位十件)

rank	TF	DF	TF-IDF	term
1	8	2	19.879	コンベック
2	15	9	14.712	タイプ
3	15	9	14.712	庫
4	23	13	14.101	加熱
5	10	6	13.863	選び方
6	15	12	10.397	内
7	6	5	9.411	オススメ
8	5	4	8.958	代表
9	5	4	8.958	チェック
10	31	18	8.918	商品

4. 実験

この章では，システムの有効性を確認する実験について説明する．

4.1. 実験方法

実験は，5つのタスクを設定し，タスクに対するシステムの出力結果を被験者に評価してもらう．

4.1.1. タスク

本タスクで設定した目的は，質問内容に対して複数の答えが存在し，それらを総合的に収集するためのWeb検索である．タスク内容はテーマの偏りが起こらないように主観で選択した．以下に5つのタスクの概要を示す．

1. **電子レンジの性能の比較ポイントを調べるため，「電子レンジ+高性能」で検索した．**
2. **大和郡山にあるおいしいラーメンのある中華料理店を調べるため，「中華+大和郡山」で検索した．**
3. **ゲームソフト「ファイナルファンタジーXIII」の攻略サイトを探すために，「FF13」で検索した．**
4. **映画『AVATAR』の評判や情報を見るために「AVATAR」で検索した．**
5. **公害の概要やその害について調べるために，「公害」で検索した．**

4.1.2. 実験の手順

実験では，上記の5タスクについて，各タスクの説明文と，Google での検索結果の上位25件からエラーが発生するページを除去した検索結果のリスト，正解の見本ページを被験者に提示する．

そして，提示したURLが各タスクの検索目的に当てはまる有用なページか，当てはまらない不要なページかを「目的に強く当てはまる」「当てはまる」「余り当てはまらない」「全く当てはまらない」の四段階で回答してもらう．

被験者はPCとWebの利用経験が3年以上ある10人(男性9人，女性1人)で，実験環境は各自のPC(OSはすべてWindows)上で，HTMLデータの閲覧が可能なブラウザを用いて行った．

4.1.3. データ

被験者実験で収集したアンケート結果の例を，表2に示す．
表2に，あるひとつのタスクで検索結果に出現したページのうち上位のもの(表の例では22ページ)に対する被験者ひとりのアンケート結果を示す．

表 2 アンケート結果の例

検索結果	被験者の回答
1 ページ目	あまり当てはまらない
2 ページ目	全く当てはまらない
3 ページ目	当てはまる
4 ページ目	あまり当てはまらない
5 ページ目	全く当てはまらない
6 ページ目	当てはまる
7 ページ目	目的に強く当てはまる
8 ページ目	当てはまる
9 ページ目	あまり当てはまらない
10 ページ目	あまり当てはまらない
11 ページ目	あまり当てはまらない
12 ページ目	あまり当てはまらない
13 ページ目	当てはまる
14 ページ目	全く当てはまらない
15 ページ目	全く当てはまらない
16 ページ目	全く当てはまらない
17 ページ目	目的に強く当てはまる
18 ページ目	目的に強く当てはまる
19 ページ目	当てはまる
20 ページ目	目的に強く当てはまる
21 ページ目	目的に強く当てはまる
22 ページ目	あまり当てはまらない

表中では，被験者の回答が「目的に強く当てはまる」「当てはまる」ならば被験者がページを有用と判断したことを,「あまり当てはまらない」「全く当てはまらない」ならば被験者がページを不要と判断したことをあらわしている.

4.2. 結果

表 3 に全タスクにおける被験者の回答とシステムの推薦結果の集計結果を，表 4 から表 8 に各タスクの結果を示す.

表 3 から表 8 においては，簡略化のため被験者の回答は「目的に強く当てはまる」を 1,「当てはまる」を 2,「あまり当てはまらない」を 3,「全く当てはまらない」を 4 としている.

表は，4×2 のマトリクスで，それぞれのセルは，被験者の回答とシステム

表 3 全タスクのデータの集計

全タスク		被験者の回答			
		1	2	3	4
システム	推薦する	152 件	89 件	77 件	82 件
	推薦しない	105 件	129 件	210 件	356 件

表 4 タスク 1 のデータの集計

タスク 1		被験者の回答			
		1	2	3	4
システム	推薦する	27 件	17 件	16 件	20 件
	推薦しない	9 件	25 件	33 件	73 件

表 5 タスク 2 のデータの集計

タスク 2		被験者の回答			
		1	2	3	4
システム	推薦する	10 件	6 件	0 件	4 件
	推薦しない	48 件	51 件	77 件	84 件

表 6 タスク 3 のデータの集計

タスク 3		被験者の回答			
		1	2	3	4
システム	推薦する	47 件	13 件	6 件	14 件
	推薦しない	0 件	3 件	11 件	116 件

表 7 タスク 4 のデータの集計

タスク 4		被験者の回答			
		1	2	3	4
システム	推薦する	49 件	33 件	35 件	23 件
	推薦しない	35 件	21 件	45 件	39 件

表 8 タスク 5 のデータの集計

問題 5		被験者の回答			
		1	2	3	4
システム	推薦する	19 件	20 件	20 件	21 件
	推薦しない	13 件	29 件	44 件	44 件

の推薦結果に対応するページが何件あったかをあらわしている．結果で，被験者が有用であるとしたページ(1, 2)をシステムが多く推薦し，被験者が不要であるとしたページ(3, 4)をシステムが推薦していなければ，システムは意図したとおりに動作していると言える．

動作の確認のため，被験者が有用としたページを推薦した件数と，不要としたページを推薦しなかった件数の合計を正答数，被験者が不用としたページを推薦した件数と有用としたページを推薦しなかった件数の合計を誤答数とする．図 7 にシステム全体と全タスクについてタスクごとの誤答数と正答数の割合を 100%積み上げ棒グラフであらわす．

図 7 から，全タスクで推薦のほぼ 6 割以上が正答であるとわかる．また，タスク 3 では正答数の割合が非常に高いとわかった．この結果は，システムが有用なページを推薦できていることが示唆している．

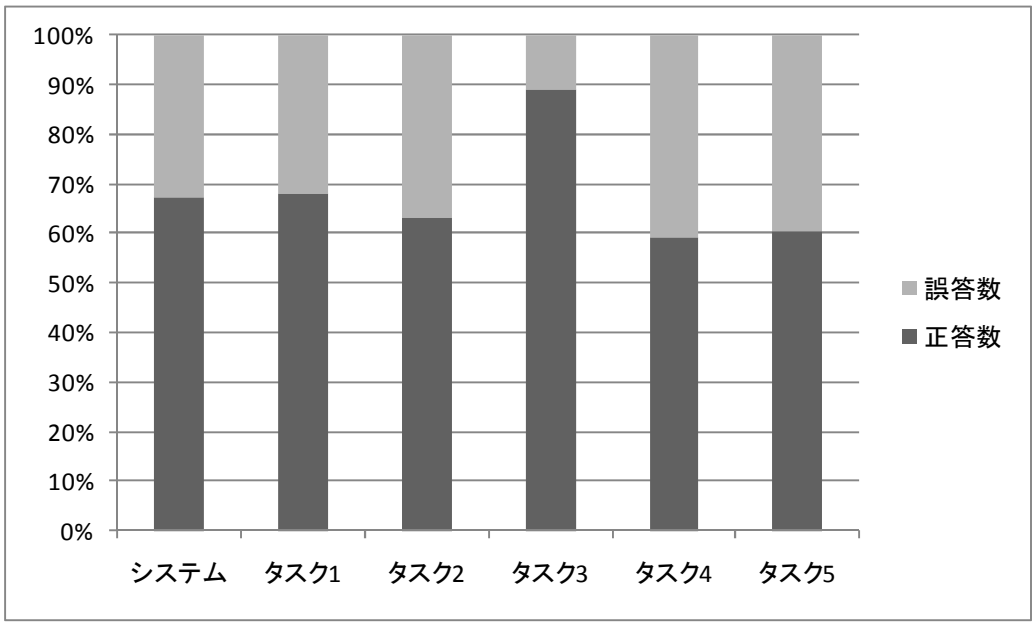


図 7 各タスクごとの誤答数と正答数の割合

5. 考察

この章では，前章での実験結果に対して統計的な分析を行い，その結果について考察する．まず分析手順について説明を行い，システム全体の場合とそれぞれのタスクの場合の結果について考察を加える．

5.1. 分析

まず,分析の容易化を計るため,前章で示した表 3 から表 8 のデータを「1,2」と「3,4」で「有用」と「不要」の二つにグループ化し，2×2 分割表にまとめる．表 9 にまとめた後の例を示す．

そして表 9 にあるそれぞれの値から， *accuracy*(正答率)，*error*(誤答率)，*precision*(精度)，*recall*(再現率)，*F1-value*(F1 値)[7]の 5 つの値を求める．これらの分析用変数について，表 10 に示した A，B，C，D，N の 5 つの変数を用いて説明する．

システムの回答がどの程度被験者の回答と一致しているかを表す値 *accuracy* は，A，D，N より，

$$accuracy = \frac{A + D}{N} \tag{2}$$

によって求められる．*accuracy* は 0～1 の範囲をとり，大きければ大きいほどシステムが被験者の回答と一致していることを示す．

表 9 2×2 分割表の例

	有用	不要	合計
推薦する	44	36	80
推薦しない	34	106	140
合計	78	142	220

表 10 分析用変数の算出方法

	良い	悪い	合計
推薦する	A	B	A+B
推薦しない	C	D	C+D
合計	A+C	B+D	N

システムの回答がどの程度被験者の回答と食い違っているかをあらわす値 *error* は，*accuracy* より，

$$error = 1 - accuracy \tag{3}$$

によって求められる．*error* は 0～1 の範囲をとり，小さければ小さいほど食い違った回答が少ないことを示す．

システムが回答した解答のうち何割が正しい答えかをあらわす値 *precision* は，A，B より，

$$precision = \frac{A}{A+B} \tag{4}$$

によって求められる．*precision* は 0～1 の範囲をとり，値が高ければ高いほどシステムが被験者の意見に反する結果を推薦していないことを示す．

被験者が有用と判断したページのうち何割をシステムが推薦できているかをあらわす値 *recall* は，A，C より，

$$recall = \frac{A}{A+C} \tag{5}$$

によって求められる．*recall* は 0～1 の範囲をとり，値が高ければ高いほどシステムが被験者の望むページを推薦していることを示す．

また，*precision* と *recall* はトレードオフの関係にあるので，仮に *precision* がよい値でも *recall* が極端に低い場合や，*recall* がよい値でも *precision* が極端に低い場合などが考えられる．このため，システムの検索性能について評価する指標として本研究では *F1-value* を用いる．

F1-value は *precision* と *recall* の調和平均をとった値で，

$$F1-value = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} \tag{6}$$

によって求められる．*F1-value* は 0～1 の範囲をとり，*precision*，*recall* が両者ともに高い場合に高い値を示す．また，*F1-value* は高ければ高いほど良い．

さらに，システムの推薦結果の有効性を統計的に確認するため，システムの推薦結果と被験者の回答の間に関連性が見られるという仮説を立てた．これを

検証する為に，帰無仮説 H0 と対立仮説 H1 を立てた．H0 と H1 を以下に示す．

H0:システムの推薦結果と被験者の回答の間に関連性はない
H1:システムの推薦結果と被験者の回答の間に関連性がないとはいえない

仮説の検証には，ピアソンの χ^2 独立性検定(以下，χ^2 検定とする)を用い，前述のそれぞれのタスクについての 2×2 分割表に対して検定を行った．

χ^2 検定は，2×2 分割表に対して，縦と横二つの事象が独立かどうかを検定する為に用いられる手法である[8]．同様の検定としてはフィッシャーの正確確率検定や二項検定などがあるが，これらの手法は標本数が多い時に精度が下がることが知られており，本実験では標本数が多いため，χ^2 検定を用いるのが適切と判断した．

表 11 に，各タスクとシステム全体についての分析結果と，χ^2 検定の結果を示す．この表から，χ^2 検定は各タスクとシステムについて，システムの判断と被験者の判断には，有意な関係があるとわかる．この結果は，提案システムが推薦した Web ページが，被験者にとって有用であり，システムが有効に機能することを示している．

5.2. 個別の考察

ここでは，前節の分析と検定の結果に基づいて，システムと各タスクに関して詳細な考察を加える．

5.2.1. システム

システム全体の分析結果では，*precision* は 0.603 で，推薦結果に占める正解データの割合は 60.3%であった．システムの適用前の検索結果では，結果全体に占める正解データの割合は 39.6%(129/356)であるので，*precision* には 1.5 倍の向上がみられる．

一方，*recall* は 0.507 で，正解データの半分を推薦していない．本システムの目的は，*precision* の高い推薦を実現することであり，*recall* は *precision* とトレードオフの関係にあるため，ある程度 *recall* が低減することはやむを得ない．

また，χ^2 検定では，$p<0.00$ で有意な関連がみられ，システムの推薦と被験者の回答には有意な関連性が認められた．

システムは，有用なページをすべて抽出できているわけではないが，システム適用前の結果に比べ不要ページを多く除去しており，多くの検索結果から有効なページだけを抽出するという本システムの目的を満たしている．今後は *recall* の改善と *precision* のさらなる向上が必要である．

また，ここでは比較対象がないため *F1-value* については議論しないが，こ

表 11 分析と検定の結果

	システム	タスク 1	タスク 2	タスク 3	タスク 4	タスク 5
accuracy	0.673	0.682	0.632	0.890	0.593	0.605
error	0.328	0.318	0.368	0.110	0.407	0.395
precision	0.603	0.550	0.800	0.750	0.586	0.488
recall	0.507	0.564	0.139	0.952	0.594	0.481
F1-value	0.551	0.557	0.237	0.839	0.590	0.484
χ^2 test	0.000*	0.000*	0.000*	0.000*	0.002*	0.017**

*有意水準 1%で有意な関連あり

**有意水準 5%で有意な関連あり

の後に示すそれぞれのタスクについての項目で,システム全体の動作との比較のために用いる.

5.2.2. タスク 1(電子レンジの比較方法)

タスク 1 では, *precision* は 0.550 で, タスク 1 におけるシステム適用前の検索結果に比べて 1.55 倍向上した. *precision* 自体はシステム全体の結果に比べて低いが, *precision* の向上の面から見ると有効な動作をしていた. また, *recall*, *F1-value* もシステムの結果より高く, このタスクではシステムの結果より良い結果が出ているとわかる. また, χ^2 検定では, $p<0.00$ で有意な関連がみられた.

precision が下がっている理由としては, システム適用前の検索結果で, 全データに対する正解データの割合が小さかったことがあげられる.

システムが特徴語として抽出した語としては,「比較」や「選び方」など目的をあらわす語から「特徴」や「出力」,「容量」など比較する項目をあらわす語までを広く抽出しており,これが良い結果を生む要因になったと考えられる.

5.2.3. タスク 2(ラーメン屋の探索)

タスク 2 では, *precision* は 0.800 で, システム適用前の検索結果に比べて 1.94 倍向上した. これはシステム全体の結果に対しても大きな値であり, システムはこのタスクにおいて *precision* の面ではよい動作をしている. また, χ^2 検定では, $p<0.00$ で有意な関連がみられた.

しかし, *recall* は 0.139 で, *F1-value*(0.237)とともに最低の値である.

recall が極端に低い原因として, 正解の見本ページに含まれていた, ひらがな表記の「らーめん」という語に対する誤検出があげられる.

このタスクにおいて重要と思われる「らーめん」というキーワードが, 形態

素解析エンジンによって「らー」と「めん」という二つの形態素に分割された．一方，検索結果に出現した，カタカナ表記の「ラーメン」はそのままひとつの形態素として抽出されており，そのため，正解ページの「らーめん」と検索結果中の「ラーメン」に対して類似度の算出が行われなかった．

本来一つの形態素として抽出されるべき単語が，複数の形態素に分割される問題は日本語形態素解析に広く見られ，改善には形態素解析の精度を向上させるか，形態素として識別するキーワードのリストを別に用意する必要がある．

また，この項目による正解データの取りこぼしはシステム全体の分析にも大きな影響を与えており，全タスクでシステムが取りこぼした推薦するべきページのうち半数近くはこのタスク 2 で取りこぼしたデータである．

5.2.4. タスク 3(ゲームソフトの攻略法)

タスク 3 は，総てのタスクの中でもっとも良い結果が出た例で，*precision* は 0.750 で，システム適用前の結果にくらべて 2.5 倍と大きく向上していた．また，*recall* も 0.952 と，正解データの取りこぼしがほとんどなかった．*F1-value* も全タスク中で最大の値(0.839)で，*precision*, *recall* 両方の面から優れた動作をしたことがわかる．また，χ^2 検定では，$p<0.00$ で有意な関連がみられた．

ユーザが有用としたページや正解の見本ページから抽出された単語にはゲームの進行に用いられる「章」や攻略情報をあらわす「攻略」，ゲームシステムに関する「地図」「鎧」など，攻略ページ特有の語が多く見られた．システムはこれらの単語が含まれない宣伝目的のページやゲームのストーリーについて解説するレビューページなどが除去できており，これが *precision* 向上の理由と考えられる．

5.2.5. タスク 4(映画の評判)

タスク 4 では，precision は 0.586 で，システム適用前に比べて 1.18 倍の向上がみられた．これは，システムの平均から見ると低い向上であるが，*recall* は 0.594 と比較的高い値を示しており，*F1-value* はシステムの平均より高い．また，*accuracy* は全タスク中で最も低かった．

accuracy が最低であるのに *F1-value* がシステムの平均より高い理由としては，元の検索結果に含まれる正解ページの量が多いことが考えられる．元の検索結果に正解ページが多いと，ランダム抽出に近いような推薦を行っても *recall* や *precision* が高い値を示す．したがって，システムの出力としては *precision* の向上率が低く，望ましくない結果が出たが，*F1-value* は高くなっている．

また，*precision* の向上率が低かった理由としては，正解の見本ページに対

して TF-IDF で特徴語抽出を行った結果に「県」や「さん」,「みたい」など,どのような文章でも見られる語が入っていたことが考えられる.

このように,一般的に頻出する語を特徴語として抽出すると,特徴語抽出の質が低下するため,通常はストップワードリスト(特徴語としてカウントしない単語のリスト)を作成しておく事が多い.本システムでもこれを実装する事で,更なる性能の改善を望める.

5.2.6. タスク 5(公害の調査)

タスク 5 の *precision* は 0.488 で,システム適用前と比べて 1.25 倍の向上が見られたが,システムの平均と比べると少ない値だった. *recall* も 0.481 と高い値ではなく,それによって F1-value も平均を下回る値になっている.

このタスクで *precision* が低かった理由としては,タスクで用いる見本ページに Wikipedia の記事を選んだことが考えられる.このケースでは,システムが Wikipedia 特有の頻出語である「ノート」や「出典」などの単語を特徴語として抽出したため,システムが類似度の測定に使える有効な単語が少なくなり,*precision* が下がったと考えられる.

この動作は,「公害」に関する特徴語を抽出しようとしたが,「Wikipedia」の特徴語が抽出されたと解釈できるため,今後は HTML タグの解析などから,メインの文章以外(他ページへのリンクやメニュー部分など)を読まない工夫が必要である.

また,データベースとして用いられた検索結果の多くは政府機関や企業の宣伝ページなどで,公害の調査に有効と考えられる資料や事例を掲載した Web ページがわずかしか結果に含まれていなかった.結果として,関係ない文書の除去は可能でも,正解ページを選択することが出来なかったと考えられる.特に,企業のページでは,環境問題に対する自社の取り組みについて,公害に関する単語を用いて解説しているページもあり,*precision* を下げる原因になったと考えられる.

5.3. まとめ

分析の結果,今回作成したシステムは,システム適用前の検索結果に対して 1.5 倍程度 *precision* の高い推薦リストを作成できており,多くの検索結果から有効なページを抽出するというシステム本来の目的を満たしている.

しかし,タスク 2 やタスク 4,タスク 5 の分析においては,いくつかの問題点が明らかになった.

タスク 2 では,「らーめん」という単語が日本語形態素解析エンジンによって「らー」と「めん」に分解され,推薦可能なページが著しく減少した.これは,平仮名表記の「らーめん」が形態素解析エンジンの辞書に登録されていな

かったことが原因と考えられる.

この問題は,形態素解析エンジンの辞書に単語を追加するか,別な辞書を用いるなどの方法で改善をはかれる.

タスク4では「県」「さん」「みたい」など,一般的にどんな文書でも用いられるような単語を特徴語として抽出したため,関連性の低い文書を推薦する問題が起きた. これは, TF-IDF のスコアリングの問題であるが, ストップワードリストの作成や TF-IDF によるスコアリングの式の変更などによって改善出来る.

タスク5では,検索結果に含まれる有用なWebページの数が少なく,*precision*が下がる問題がみられた. そこで,今後の研究では検索対象にするページの数についても検討する必要がある.

また, 今回は TF-IDF 法を用いて文書中の特徴語のスコアリングを行った. しかし, 特徴語の抽出アルゴリズムには他にも種類があり, TF-IDF 法については文書内での単語の出現回数(TF)に重みをつけすぎると言う見方[7]もある.

TF-IDF 法以外の特徴語抽出アルゴリズムとしては,TF-IDF の TF を対数化したアルゴリズムや,ページの長さを考慮した計算式などもあるため,今後の研究ではこれらと比較して考察を行えば, システムの性能向上に繋がる.

6. おわりに

本研究では，複数の正解ページを持つ検索の支援と，既存の検索に関する研究やシステムの問題点の解決を目的とし，問題点の解決のためには，TF-IDF と形態素解析によって，ユーザの提示した見本ページに類似した Web ページを推薦するシステムを提案，実装し，被験者実験による有効性の確認を行った．

提案手法では，ユーザの提示した正解の見本ページと検索結果から，形態素解析と TF-IDF 法を用いて特徴語を抽出し，検索結果を正解の見本ページとの類似度順に並べ替える．提案システムでは，検索キーワードの情報と関係のないページの推薦を行わないという利点があり，これによって既存システムの問題点を解決が出来る．また，広範囲な検索に対応した既存研究がない点については，分野を絞らないタスクを５つ設定し，被験者実験を行いシステムの評価を試みた．

被験者実験の結果，本システムでは複数の正解ページの存在が予想されるような検索について，適切な正解の見本ページを用意すれば，正解ページを 5 割程度見落とすかわりに，もとの検索結果に比べて 1.5 倍程度精度(*precision*)の高い推薦リストを提示できることがわかった．

システムは，Web 検索でのキーワードによる絞り込みになれていないユーザの支援目的に作ったもので，複数の正解ページの存在が予想され，なるべく多くの情報を集めたい場合の検索に適用することで，元の検索結果より高い精度で情報収集を行う事ができる．

今後の課題として，本研究では，実験に著者が主観で定めたタスクを用いた．そのため，広範囲な目的の検索について，網羅的に対応できているか客観的に評価できないという問題があった．今後，検索エンジンの有効性確認用の文書セットデータベースである NTCIR のテストコレクション[9] [10]を用いた実験を行うことで，より有効性の高い評価が可能となる．

また，本研究では日本語形態素解析の限界によって推薦があまり上手く働かなかったタスクや，ストップワードリストを実装していなかったことによる無用単語の拾い上げなどの例も見られ，今後の研究ではこれらの点も改善する必要がある．

謝辞

本論文の執筆および研究を進めるにあたって，多くの方に多大なお力添えをいただきました．この場を借りて感謝の意を表明させていただきます．ありがとうございました．

指導教員である上野秀剛助教には，お忙しい中，研究のスケジューリングから論文のチェック，提出まで，様々な方面から丁寧で的確なご指導を頂きました．厚く御礼申しあげます．

査読教員である松尾賢一准教授からは，査読コメント，卒研発表会の両方で，的確で鋭いご質問を賜り，本論文の修正の上でいくつも気づかされる点がありました．先生のご質問は大変参考になりました．厚くお礼申し上げます．

本研究の中間発表会でご質問くださった近藤勝也特任教授には，当方の説明の至らない点を厳しく指摘していただき，その後の資料の作成について非常に参考になる意見を頂きました．ありがとうございます．

また，山口智弘教授にも，同発表会で質問をいただきました．その際には類似分野からの適切で鋭いアドバイスを賜り，論文の構成を決める段階でご意見を参考にさせていただきました．ありがとうございます．

同研究室の先輩，同輩の皆様，ならびに情報工学科 5 年クラスメイトの皆様や専攻科の先輩方にも，励ましや助言を頂き，また相談もさせてもらいました．ありがとうございます．

また，研究の進行において，お忙しい中被験者実験にご協力くださった皆様，まことにありがとうございます．

参考文献

[1]総務省 情報通信白書(平成 21 年度版) 図表 4-1-1-3.
[2]中島悠, 土方嘉徳, 西田正吾:"検索経験と領域知識の WWW 情報検索行動に与える影響,"ヒューマンインタフェース学会論文誌, Vol.7, No.2, pp.131-141(2005).
[3]國貞 暁, 山本 けい子, 田村 哲嗣, 速水 悟: "要約情報の類似度を用いた WEB 検索支援システム," 人工知能学会 2007 年全国大会, 3H7-1, 2007.
[4]長尾真: "言語の機械処理," 三省堂(1984).
[5]工藤拓, 山本薫, 松本裕治: "Conditional Random Fields を用いた日本語形態素解析," 自然言語処理研究会報告 Vol.2004, No.47, pp.89-96(2004).
[6]G. Salton, M.J.McGill: "Introduction to modern information retrieval," McGraw-Hill, New York(1983).
[7]北研二, 津田和彦, 獅子掘正幹: "情報検索アルゴリズム," 共立出版(2002).
[8]山田剛史, 杉澤武俊, 村井潤一郎: "R によるやさしい統計学" オーム社(2008).
[9]Eguchi,K., Oyama,K., Aizawa,A., Ishikawa,H.: Overview of the Informational Retrieval Task at NTCIR-4 WEB; Proceedings of the Fourth NTCIR Workshop on Research in Information Access Technologies Information Retrieval, Question Answering and Summarization (2004).
[10]Oyama,K., Eguchi,K., Ishikawa,H., Aizawa,A.: Overview of the NTCIR-4 WEB Navigational Retrieval Task 1; Proceedings of the Fourth NTCIR Workshop on Research in Information Access Technologies Information Retrieval, Question Answering and Summarization (2004).