ベイジアンフィルタを利用したWebページランキングシステムの提案とADMによる評価

庭野正義* マッキン ケネス ジェームス** 永井保夫**

あらまし Googleなどに代表される検索エンジンを用いてWebページを検索する場合、膨大な数のWebページのリストが検索される。さらに、そのリストは必ずしもユーザ個人に適した順序で表示されているとは限らない。大量の検索結果の中から必要なページを判断するにはかなりの労力が必要となる。本研究では、ベイジアンフィルタに興味状態の概念を導入し、「大量の検索結果の中から必要なページを判断する」という作業を自動化するWeb 推薦システムの検討を行ってきた。本論文では、前述のWeb推薦システムを基に、ベイジアンフィルタを利用したWebページランキングシステムを提案し、試作と実験による評価と考察を行った。評価尺度として、「システムが行った評価とユーザが行った評価がどれだけ近いか」という指標であるADM（Average Distance Measure）を採用した。その結果、ユーザの調べたい事が比較的大きく、一度の検索で十分な情報を得られない場合に、提案したWebページランキングシステムが有効である事を示す。

キーワード：Webページ、ランキング、嗜好情報、ベイジアンフィルタ、ADM

Web Page Ranking System Using Bayesian Filter and It's Evaluation by Using Average Distance Measure（ADM）

Masayoshi NIWANO*, Kenneth JAMES MACKIN**, and Yasuo NAGAI**

Abstract When the Web pages are retrieved by using the search engine such as Google search engine etc, a great number of Web pages are retrieved by the list form. In addition, these pages are not necessarily displayed in the appropriate order for the each users. It spends a lot of time in order to judge and search for a necessary page from among a large amount of retrieval result. In this research, the concept of the interest state for each users is introduced into the concept of Bayesian filtering, and the Web recommendation system automating the work that a necessary page is judged from among a large amount of retrieval result is considered. We proposed the Web page ranking system using the Bayesian filtering based on the Web recommendation system research. We evaluated the proposed Web page ranking system by adopting the ADM (Average Distance Measure) as a measure for evaluation showing that" how near are the evaluation of the system has done and the evaluation of the users have done ?" The experiment result shows that the effectiveness of the Web page ranking system when enough information cannot be obtained by once retrieval because the retrieval space is so huge.

Keywords：Web page, Ranking, Preference infomation, Bayesian filter, ADM

*東京情報大学 大学院 総合情報学研究科
Tokyo University of Information Sciences, Graduate School of Informatics
2010年 4 月よりアイコムシステック株式会社に所属
**東京情報大学 総合情報学部 情報システム学科
Tokyo University of Information Sciences, Faculty of Informatics, Department of Information Systems

1．はじめに

GoogleやYahoo!、goo、msnといった検索サイトで検索する場合、その検索結果は膨大であり、かつ、必ずしもユーザ個人に適した順序で表示されているとは限らない。そのため、ユーザは、大量の検索結果の中からタイトル、概要などを見て、ユーザ自身がそのページにアクセスするかどうかを判断することが必要になる。大量の検索結果に対してこの作業を繰り返すにはかなりの労力が必要であり、その問題点を解消するための研究が精力的に行われている[1][5][11]。

ユーザの嗜好に合ったWebページを推薦することができれば、「検索結果を見てアクセスするかしないかを判断する」部分を自動化させることができる。

しかし、協調フィルタリングの技術を用いて行われるWebページ推薦は、誰かが既に評価しているWebページしか推薦できないという問題点がある［4］。日常的な検索では誰も評価した事のないWebページが検索結果に含まれている事が多く、協調フィルタリングの技術を適用しにくい。

一方、ベイジアンフィルタは、ベイズ推定を利用して、対象となるデータを解析、学習し、分類するためのフィルタである。学習量が増加するとフィルタの分類精度が上昇するという特徴を持ち、文章の自動分類や、スパムメールの自動振り分けに利用されている［10］。スパムメールの自動振り分けでは、メールの文章を解析し、スパムメールかどうかを判断している。この作業は、「検索結果を見てアクセスするかしないかを判断する」作業と非常に似通っており、ベイジアンフィルタを応用する事により、対象にユーザが興味を持つかを判断できると考えられる。

そこで、本研究では、ベイジアンフィルタを利用し、Webページをランキングするシステムを提案する。スパムメールの自動振り分けでは、メールの文章を解析し、スパムメールである確率が閾値を超えた時にメールをスパムと判断する。一方、提案するWebページランキングシステムでは、検索システムから受け取った検索結果の文章（タイトル、概要、ホスト名）を解析し、ユーザが興味を持つ度合いを求め、その度合いの降順に検索結果を並び替える。ユーザが興味を持つ度合いの高い検索結果を上位に並べかえることで、「ユーザが検索結果を見てアクセスするかしないか判断する」手間を省き、検索の効率化を図る。

システムの評価尺度には、「システムが行った評価とユーザが行った評価がどれだけ近いか」という指標であるADMを採用した。

本論文の構成は以下の通りである。まず、2.章では、Webページランキングシステムの構成と概要について述べた後、どのように検索結果を推薦し、嗜好情報の記録が行われているかについて説明する。次に、3．章では、評価実験の内容と結果、ならびに考察を述べる。最後に、4．章で、まとめと今後の研究について述べる。

2．ベイジアンフィルタを利用したWebページランキングシステム

2.1　試作システムの概要

スパムメールの自動振り分けでは、メールの文章を解析し、スパムメールかどうかを判断している。この作業は「受け取った大量の文章を2つのクラスに分類する」という点で、「検索結果を見てアクセスするかしないかを判断する」作業と非常に類似している。この点に着目し、本研究ではベイジアンフィルタを応用する事により、検索対象にユーザが興味を持つかを判断できると考えることにする。本研究では、ベイジアンフィルタを利用し、Webページを順位付けするシステムを提案し、試作する。試作システムでは基本的に、スパムフィルタが行う「メールがスパムか非スパムかを判断する」という処理をそのまま「Webページを閲覧す

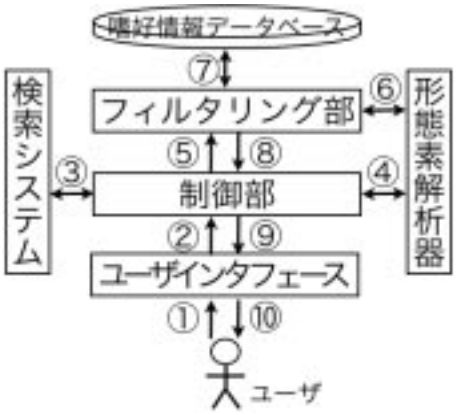

図1．システム構成

るかしないかを判断する」という処理に置き換える。スパムフィルタの場合は、スパムである確率に基づき、メールをスパムか非スパムかに分類する。一方、試作システムでは、検索結果をユーザの嗜好に合わせて並び替える事により、ユーザの嗜好に合ったページが上位に表示されるようになる。その結果、より早く目的のページにたどり着く確率が上がり、検索作業効率の向上が期待できる。

2.2 特徴

試作システムは、ユーザがWebページを閲覧する確率（以下、推薦度とする）を計算するために、次の情報を利用する。

- ユーザが検索時に入力した検索ワード
- 既存検索システムから返された検索結果のタイトル、概要、ならびにホスト名
- 返された検索結果から、そのWebページにアクセスしたか/しなかったかの情報

このような情報を利用する事で、以下のような利点が得られる。ユーザの作業を増やさない検索結果への評価値として、ユーザが検索結果のWebページにアクセスした/しなかったを1と0に対応させた2値を用いる。この値を用いることで、ユーザがWebページの評価をシステムに入力するという新たな手間が発生せず、Googleなどの既存の検索システムを利用する場合と変わらない作業量で検索を行える。

大量の嗜好情報が手に入るユーザの評価を、「ページのタイトル、概要、ホスト名を見て、実際にそのページへアクセスした/しなかった」を1と0に対応させた2値とする。それにより、評価値入力のための新たな作業が発生しないため、ユーザが評価したページ全てをシステムに入力できる。

全く新しいWebページでも適切に推薦できる検索結果のタイトル、概要、ホスト名を取得し、その文章の特徴と、ユーザの嗜好情報を比較する。そうすることにより、ユーザやシステムが初めて見たWebページでも、ある程度適切な評価を行う事ができる。

2.3 システム構成と処理概要

Webページランキングシステムの構成は図1のようになっている。以下では、提案したWebページランキングシステムの処理概要について説明する。

Step 1 検索ワードを受け取る　Googleなどの一般的な検索システムと同じように、ユーザは、ユーザインタフェースを通してシステムに検索語を入力する（①）。入力された検索語を、制御部が受け取る（②）。

Step 2 既存検索システムでの検索　制御部は、受け取った検索語をそのままGoogle AJAX Search API［3］へ送信し、検索結果の集合を受け取る（③）。今回は、Google AJAX Search API から最大32個の検索結果を取得している。

Step 3 検索興味状態の取得　制御部は、入力された検索語を形態素解析器へ送り、形態素集合を受け取る（④）。受け取った形態素集合から名詞、動詞、未知語のみを抜き出し、それらを興味状態とする。

Step 4 推薦度の計算　制御部は、興味状態と検索結果をフィルタリング部へ送り（⑤）、検索結果毎に推薦度を取得する（⑥、⑦、⑧）。

Step 5 ユーザへの提示　制御部は、検索結果集合を、取得した推薦度の降順で並び替え、上位から順番にユーザに提示する（⑨、⑩）。

Step 6 ユーザの嗜好情報の取得　制御部は、提

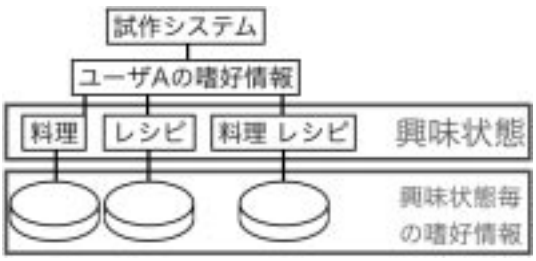

図2．興味状態と嗜好情報

示された推薦結果のWebページを、ユーザが実際にアクセスしたか/しなかったかという情報を受け取り、フィルタリング部へ送る（①、②、⑤）。

Step 7 嗜好情報に基づき、データベースを更新する　フィルタリング部は、受け取ったユーザの嗜好情報を基に、データベースを更新する（⑥、⑦）。

以上のStep 1からStep 7を繰り返す事で、検索結果の再順位付けによる推薦とユーザの嗜好情報データベースの更新が行われる。

2.4　興味状態の導入と取得

試作システムは、「ユーザがどのような項目を調べたいか」ということを興味状態として表現する。

検索結果の再順位付けを適切に行うために、ユーザがどのような興味状態であるかを把握した上でWebページの推薦度を求めなければならない。例えば、普段、料理について調べ、料理のレシピが記述されたページに興味を持つことがわかっていたとする。その場合に本を検索をしている時に、料理のレシピが書いてあるWebページを上位に表示するのは、検索時のユーザの興味を反映していないと考えられる。したがって、検索を行うたびに変化するユーザの興味を、興味状態として表現し、管理する必要がある。そこで、試作システムでは、ユーザの興味を興味状態として表現し、興味状態毎に別々の嗜好情報を記録する方法をとった。検索語が興味状態を表していると仮定し、「検索語の形態素の中から名詞、動詞、未知語のみを抜き出したものと、それらの中から2つの形態素を組み合わせたものの和」を興味状態として定義する。形態素とは、文章の中で意味を持つ最小の単位である。例えば、「料理レシピ」という検索語で検索した場合の検索結果は、「料理」、「レシピ」、「料理レシピ」という3つの興味状態に属しているとみなされ、ここで取得した嗜好情報は、図2のように興味状態別に記録される。図2は、試作システムがユーザAの中でも、興味状態毎に別々に嗜好情報を記録していることを表している。このように興味状態を取得し、取得した興味状態毎に嗜好情報の記録を行うことで、検索時のユーザの興味を反映できようになるため、より適切な再順位付けを行える。

2.5　推薦度の計算

本節では、トークンの定義を説明した後、2.3節のStep 4において推薦度を求める手順を説明する。

2.5.1　トークン

ここでは、トークンを「文章を分割する単位」と定義する。今回は、2種類のトークンの取得方法を採用し、それぞれの評価を行った。1つ目は、一般的なトークン取得方法と同じで、「形態素解析器により分割された1つの形態素」を1つのトークンとしてデータベースに記録する方法である。2つ目は、ある程度トークン同士の関係に注目するようにした方法である。ここでは、「形態素解析器により分割された1つの形態素」に加え、連続する5 つの形態素のうち2つの形態素を組み合わせたものを1つのトークンとして扱う。

例えば、「Web推薦システム」という文章を分割するとき、1つ目の方法では「Web」、「推薦」、「システム」という3つのトークンが得られ、2つ目の方法では「Web」、「推薦」、「システム」、「Web 推薦」、「Webシステム」、「推薦システム」というトークンが得られる。

2.5.2　Webページの推薦度

Webページの推薦度$P(D, w)$は式(1)で表される。ここでは、「検索ワードwを与えられた

時のWebページ（ドキュメント）Dの推薦度」を$P(D, w)$、「ユーザが入力した検索ワード」をw、「検索ワードwに含まれる興味状態の数」をn、「検索ワードwに含まれるi番目の興味状態（$1 \leqq i < \leqq n$）」をc_i、「興味状態c_iが与えられた時のドキュメントDの推薦度」を$P(D, c_i)$と表す。

2.5.3で説明する式(2)により、$P(D, c_i)$を求め、式(1)に代入し$P(D, w)$を求める。

$$P(D, w) = \frac{\prod_{i=1}^{n} P(D, c_i)}{\prod_{i=1}^{n} P(D, c_i) + \prod_{i=1}^{n} (1 - P(D, c_i))} \quad (1)$$

2.5.3 興味状態毎の推薦度

検索興味状態毎の推薦度$P(D, c_i)$は式(2)で表される。ここでは、c_iを2.5.2節で説明した式(1)と同じものとし、「興味状態c_iが与えられた時のドキュメントDの推薦度」を(D, c_i)、「ドキュメントDに含まれているトークンの数」をm、「興味状態c_iが与えられた時の、ドキュメントDに含まれるj番目のトークンt_jの推薦度$(1 \leqq i \leqq m)$」を$P(t_j, c_i)$と表す。2.5.4で説明する式(3)により、$P(t_j, c_i)$を求め、式(3)に代入し$P(D, c_i)$を求める。

$$P(D, c_i) = \frac{\prod_{j=1}^{m} P(t_j, c_i)}{\prod_{j=1}^{m} P(t_j, c_i) + \prod_{j=1}^{m} (1 - P(t_j, c_i))} \quad (2)$$

2.5.4 トークン毎の推薦度

トークン毎の推薦度$P(t_j, c_i)$は式(3)で表される。c_i, t_jは、2.5.3節で説明した式(2)のc_i, t_jと同じものとし、「興味状態c_iが与えられた時のトークンt_jにユーザが興味を持った回数」を$MC(t_j, c_i)$、「興味状態c_iが与えられた時のトークンt_jにユーザが興味を持たなかった回数」を$NC(t_j, c_i)$、「興味状態に属するトークン全てのユーザが興味を持った回数の合計」を$MC(c_i)$、「興味状態に属するトークン全てのユーザが興味を持った回数の合計」を$NC(c_i)$と表す。

$MC(t_j, c_i)$、$NC(t_j, c_i)$、$MC(c_i)$、$NC(c_i)$は、ユーザの嗜好情報が記録されているデータベース（2.6節参照）から取得する。データベースに

表1．嗜好情報データベース

	トークン	興味状態	選択回数	非選択回数
レコード1		web	10	25
レコード2	システム	web	3	5
	…	…	…	…

どのようにユーザの嗜好情報が記録されているかは2.6節で説明する。

$$P(t_j, c_i) = \frac{\frac{MC(t_j, c_i) + 1}{MC(c_i) + 1}}{\frac{MC(t_j, c_i) + 1}{MC(c_i) + 1} + \frac{NC(t_j, c_i) + 1}{NC(c_i) + 1}} \quad (3)$$

2.6 ユーザの嗜好情報

ユーザの嗜好情報は、表1のデータベースに格納される。第1フィールドにトークン、第2フィールドに興味状態が格納される。この2つのフィールドが主キーとなる。第3フィールドには、「第2フィールドの興味状態に属する第1フィールドのトークン」にユーザが興味を持った回数、第4フィールドには、「第2フィールドの興味状態に属する第1フィールドのトークン」にユーザが興味を持たなかった回数を格納する。

第1フィールドであるトークンが空のレコード（表1のレコード1）には、その興味状態全体に対する嗜好情報（式(3)の$MC(c_i)$と$NC(c_i)$）が記録され、トークンがあるレコード（表1のレコード2）にはその興味状態に属するトークンに対するユーザの嗜好情報（式(3)の$MC(t_j, c_i)$と$NC(t_j, c_i)$）を記録する。

例えば、表1の場合、レコード1は、「web」という興味状態で検索された時の検索結果が合計35個であり、35個のうち10個の検索結果に興味を持った事を表している（式(3)の$MC(c_i) = 10$、$NC(c_i) = 25$）。レコード2は「web」という興味状態で検索された結果の中に、「システム」というトークンが合計8個含まれており、その8個のうち、3個の検索結果に興味を持ったという事を表している（式(3)の$MC(t_j, c_i) = 3$、$NC(t_j, c_i) = 5$）。

このようなデータベースを作成し、ユーザの嗜好の情報を記録しておく事により、2.3節のStep 4の推薦度の計算に必要なユーザの嗜好情報（式(3)で利用する$MC(t_j, c_i)$、$NC(t_j, c_i)$、$MC(c_i)$、$NC(c_i)$の値）を求める事ができる。2.3節のStep 7では、ユーザの嗜好情報を受け取り、データベースを更新する。

3. 実験による評価と考察

3.1 実験内容

実験は、以下に示す通りに行う。

(1) 被験者5名に、図3のWebページで検索を行ってもらう。被験者は、上部と下部に配置されているテキストフィールドに検索ワードを入力し、sendボタンを押す事で、検索を行う。sendボタンを押し、検索ワードを送信すると、中央に検索結果が4つずつ表示される。nextボタン、prevボタンを押す事で、前後の検索結果を表示する。この時、推薦度による並び替えは行わない。本実験では、表示された検索結果と実際のWebページの内容が一致していない場合は考えない事とする。つまり、検索結果の内容が、Webページの内容を正しく要約しているものと仮定している。

(2) 被験者が入力した検索キーワードと、どのような検索結果が返ってきたか、ユーザがどの検索結果を選択したか、という情報を記録する。

(3) 記録した情報を基に、Webページランキングシステムを使った場合と使わなかった場合それぞれで、1回の検索毎のADMを計算する。

記録したデータは、表2の形式で保存される。このデータを用いて、何も処理をしない場合と、「1トークン1形態素」の方法で嗜好情報を記録し推薦した場合、「1トークン2形態素」の方法で嗜好情報を記録し推薦した場合のシステムを評価する。

3.2 ADM

推薦システムの推薦性能を評価するために、正確性と網羅性の観点から適合率や再現率が評価尺度として利用されている［2］。適合率は、推薦システムが推薦した情報の中に、どれだけユーザの要求が満たされている情報を含んでいるかの割合を示す。一方、再現率は、推薦した情報で、ユーザの要求を満たしているもののうち、実際に推薦された情報の割合である。しかしながら、提案するWebページランキングシステムは、単純に「推薦する/しない」に分類するわけではなく、推薦度の降順でユーザに提示することで推薦を行っており、推薦度は0から1までの連続値である。このシステムを、適合率や再現率で評価するためには、推薦度に閾値をもうけ、閾値以上のものを推薦し、そうでないものは推薦しないという処理をしなければならない。そのため、検索結果の再順位付けによる推薦を行うシステムを評価する指標として、適合率や再現率は適切ではないと判断した。そこで、「システムが行った評価とユーザが行った評価がどれだけ近いか」という指標であるADMを使用し、提案するWebページランキングシステムを評価することにした。ADMとは、システムが行った評価とユーザが行った評価とが完全に一致するシステムが最良のシステムであるという仮定の下でシステムを評価する手法で、式(4)で表される。

ここでは、「検索結果集合」をR、「検索結果集合RのADM値」をADM_R、「検索結果集合Rに属する検索結果の数」をn、「検索結果集合Rに属する i 番目の検索結果」をr_i、「r_iに対するシステムの評価」を$SRE_R(r_i)$、「r_iに対するユーザの評価」を$URE_R(r_i)$と表している。

$URE_R(r_i)$は、ユーザが r_iを選択した場合1となり、選択しなかった場合は0となる。$SRE_R(r_i)$は、2.5で説明した推薦度で、0から1までの数値で表される。ADM値が1に近づけば近づくほど、ユーザの行った評価とシステムの行った評価が近く、逆に、0に近づけば近づくほどユーザの行った評価とシステムの行った評価が離れている事を示す。

図3．実験用Webページ

表2．実験で記録したデータ

keyword:=: 検索ワード―
 No:=: 1―
  title:=: タイトル
  abstract:=: 概要
  host:=: ホスト名
  stats:=: ユーザーの評価情報
 No:=: 2―
…
…
…

$$ADM_R = \frac{1 - \sum_{i=1}^{n} |SRE_R(r_i) - URE_R(r_i)|}{|R|} \quad (4)$$

3.3 検索例の説明

「CUDA 環境導入」という検索ワードで検索した結果を表3に示す。表3は被験者の1人が行った検索に結果と、それらをWebページランキングシステムによって再順位付けした検索結果とを比較したものである。

表3の「*」マークは、その印のついたページにユーザがアクセスした事を示す。ユーザが

表3．実験1で使われていた検索語

順位	Google AJAX Search APIの検索結果	提案システムにより再順位付けされた推薦結果
1	*ひびろぐ ver.2 ― Windowsの無茶な環境でCUDAを使うための方法	*○○’s website
2	*○○’s website	*ひびろぐ ver.2 ― Windowsの無茶な環境でCUDAを使うための方法
3	CUDA技術を利用したGPUコンピューティングの実際（前編）	【GPGPU】NVIDIA CUDA 質問スレッド
4	www.cuda-powerdirector.com	*CUDA 開発環境のインストール
5	特定の環境 と Barracuda 7200.11に関する情報 -ZD-	Barracuda - FAQ： 富士通ソーシアルサイエンスラボラトリ
6	「cuda」を含むブログ-はてなキーワード	CUDA技術を利用したGPUコンピューティングの実際（前編）
7	Barracuda ATA 導入記	「cuda」記事検索 - gooニュース
8	Barracuda - FAQ： 富士通ソーシアルサイエンスラボラトリ	Barracuda ATA 導入記
9	*CUDA 開発環境のインストール	Mac OS X と環境とCUDAに関する記事 - builder by ZDNet Japan
10	東京工業大学、グローバルCOE「計算世界観」にて国内初のNVIDIA	お気楽なページ：TMPG がCUDA をサポートへ！
11	価格.com - 『CUDA といえば...』話題のキーワード検索 -	テクノロ散策：NVIDIA GPU プログラミング統合開発環境「CUDA」Mac版
12	ドスパラ - DOSPARAが語る IT 活用ブログ:nividia	TMGEEnc 4.0 XPress でCUDA 2.0 を試してみた

アクセスした「○○’s website」という名前のページが2位から1位へ、「CUDA開発環境のインストール」が9位から4位へ移動している事がわかる。

入れ替えにより1位となった「○○’s website」というアイテムを「1トークン1形態素」の方法で解析した結果の一部を表4に示す。表4のトークンとその推薦度の項目は、検索システムから受け取った「○○’s website」のタイトル、概要、ホスト名をひとまとめにした文章の解析結果を示している。「興味状態：

表4.「○○’s website」の解析結果

検索ワード：“CUDA 環境 導入
興味状態：“CUDA”,“CUDA 導入”,“CUDA 環境”,“環境”,“導入”,“環境 導入”
トークンとその推薦度（一部抜粋）： CUDA：CUDA = 0.406 CUDA：インストール = 0.63 CUDA：プログラミング = 0.73 CUDA：環境 = 0.47 CUDA：用意 = 0.45 環境：環境 = 0.63
興味状態毎の推薦度： CUDA = 0.85 CUDA 導入 = 0.50 CUDA 環境 = 0.50 導入 = 0.50 環境 = 0.63 環境 導入 = 0.50
全体の推薦度： 0.90

トークン＝推薦度」という書式で表されており、「CUDA：CUDA＝0.40」は「CUDA」興味状態に属する「CUDA」というトークンの推薦度が 0.40 である事を示す。

全体の推薦度を求める手順は、以下の通りである。まず、タイトル、概要、ホスト名に使われているトークン全ての推薦度を求める。次に、トークンの推薦度を利用し、興味状態毎の推薦度を求める。最後に、文章の推薦度を求める。このように、検索ワードから求めた興味状態と、ページ情報に使われているトークン、ユーザの嗜好情報を利用して推薦度を求める。その結果、「○○’s website」の推薦度は 0.90 となり、1位へ再順位付けされた。Google AJAX Search APIの検索結果では 4 位となっている「www.cuda-powerdirector.com」の推薦度は 0.07となり、25位へ再順位付けされた。

表 4 を見ると、同じ「環境」というトークンでも、興味状態毎に推薦度が違っているのがわかる。長期的にユーザの嗜好情報を取得し記録していった時、興味状態を導入しない場合では、この興味状態毎の推薦度の差が平均化されてしまう。例えば、表 4 の「環境」というトークンは、あくまで「CUDAの環境」であり、それ以外の環境（「環境問題」の「環境」など）とは意味が異なる。本システムの興味状態取得方法は、今後「環境問題」という検索ワードで検索した時にも、「CUDAの環境」で検索した時の嗜好情報と区別ができるため、精度の高い推薦が期待できる。ただし、この例のように、2つの形態素の組み合わせを興味状態とした場合、興味状態の数が非常に多くなってしまう。1度の検索毎に興味状態の数だけ文章の解析と学習を繰り返さなければならないため、学習、推薦時の処理量と、ユーザの嗜好情報の記憶量が増加してしまう。

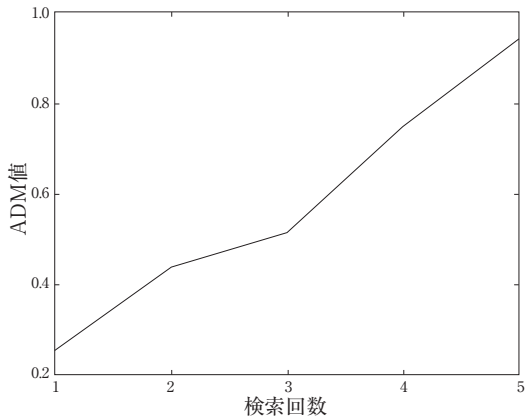

図4．検索回数とADM値の推移

表5．各検索での検索ワード

検索回数	検索ワード
1	仮想化とは
2	仮想化技術
3	仮想化の歴史
4	仮想化の歴史
5	仮想化ソフトの歴史

3.4　実験結果

被験者 5 名に 3.3 節で示した検索を繰り返し、合計 620 回の検索を行った。1つのトークンを1つの形態素で構成する方法では、平均ADM値が 0.65 となり、1つのトークンを2つの形態素の組み合わせで構成する方法を用いて

も、同等の結果となった。

実験結果の一部のグラフを図4に示す。図4は横軸が検索回数、縦軸が検索のADM値を表すグラフである。このグラフを見ると、検索を重ねる毎にADM値が上昇している事がわかる。表5は、図4の各検索毎に、どのような検索ワードが使われているかを示している。

例えば、1度目の検索では、「仮想化とは」という検索ワードで検索し、その時のADM値が 0.25 であるという事を示している。このように、1つ又は2つの検索キーワード（表5の場合は「仮想化」）を固定し、それ以外のキーワードを追加、変更しながら調べる場合、検索回数を重ねる毎にADM値が上がっていく傾向が見られた。今回の実験ではこの他に、「CUDA」「Objective-C」「物理演算」について調べているときに、検索を重ねる毎にADM 値の上昇が見られた。

3.5 考察

1つ又は2つの検索キーワードが固定で、それ以外のキーワードを追加、変更しながら調べる場合、ADM値が右上がりで上昇している傾向見られた。このような検索を行う場合は、一度の検索で十分な情報を得ることができないために検索を繰りかえしている場合が多い。そのため、再順位付けを行わない場合と比べ、本システムを使用した場合の方がより早く目的のページにたどり着ける可能性が高いと思われる。

同じトークンでも興味状態が異なる場合には、推薦度が違うことがわかった。このことから、ユーザが対象を変えて複数回の検索を行った場合、興味状態を導入した方がより精度の高い推薦が行えると考えられる。

1つのトークンを複数の形態素の組み合わせで構成する手法を用いた場合も、1つのトークンを1つの形態素で構成する手法を用いた場合も、ADM値はほぼ等しい事がわかった。これは、検索システムから得られる「タイトル、概要、ホスト名」という情報が、文章としては量が少なく、かつ、細切れな文章になっている事が多いことに起因すると思われる。普通の文章の場合、言葉の係り受けや、文章の流れなどがあるが、文章が細切れになると、その関係が壊れることが多く、その結果、トークン同士の前後の位置関係の重要性が低下する結果になったと考えられる。実験では1つのトークンを1つの形態素で構成する方式が処理時間の面で効率が良いことがわかった。また、検索結果の概要が、Webページの内容をうまく要約し、細切れになっていない文章の場合には、1つのトークンを複数の形態素の組み合わせで構成する方式が精度が高くなる可能性も考えられる。今後、検索システムをGoogle AJAX Search APIから別の検索システムのAPI に変更して処理効率と精度に関する実験を行うことを考えている。

検索ワードに使われているトークンの組み合わせによる検索結果の分類は、長い検索ワードが入力されたときなどに、処理量の増加、記録する情報の増加が問題となる。さらに、本来同じ興味状態に分類したい「同じ対象を表している別の言葉（表記揺れなど）」や、「複数の対象を包含する概念的な言葉」を、全く別の興味状態と認識してしまうことも問題である。そのため、別の検索結果の分類方法も検討し、改良する必要がある。

4. おわりに

スパムメール自動振り分けにも使われているベイジアンフィルタに、興味状態の概念を導入し、Webページランキングシステムに適用し、実験を行うとともに、ADMを用いてWebページランキングシステムを評価した。ADMを用いることで、再順位付けによる推薦を行うシステムを「システムが行った評価とユーザが行った評価がどれだけ近いか」という指標で評価することができた。その結果、ユーザが複数の対象について検索を行った場合、興味状態を導入した方がより精度の高い推薦が行えることを明らかにした。さらに、提案したWebページランキングシステムは、ユーザの調べたい項目が

多岐にわたり、一度の検索で十分な情報を得ることが難しい対象を調べる場合に有効であることを明らかにした。

今後は、同義語、表記揺れへの対応、より処理効率の高い興味状態取得方法の検討、協調フィルタリングの手法を取り入れるなどの改良をするとともに、さらに実験データを収集し、評価していく予定である。

【文献】

[1] 天野環，中里秀則，中村隆史：ベイズ推定を用いたWebマイニング，電子情報通信学会技術研究報告Vol.104，No724（2005），pp.43-48.

[2] Christopher，D. M. Prabhakar，R. and Hinrich，S：Introduction to Information Retrieval，Cambridge University Press, New York, 2008.

[3] Google AJAX Search API - Google Code ：http://code.google.com/intl/ja/apis/ajaxsearch/.

[4] 石川徹也，宇田隆幸：情報フィルタリングの利用システム：情報推薦システム（<特集>情報のフィルタリング），情報の科学と技術Vol.59, No10（2006），pp.458-463.

[5] 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，第21回人工知能学会全国大会，Miyazaki，JSAI，2007.

[6] Mizzaro, S：A new measure of retrieval effectiveness(Or : What's wrong withprecision and recall)，International Workshop on Information Retrieval，Oulu，IR，2001.

[7] 庭野正義，Kenneth James Mackin，永井保夫：ベイジアンフィルタを利用したWeb推薦システムの試作と評価，電子情報通信学会2009総合大会D-8-13，Ehime，IEICE，2009.

[8] 庭野正義，K.J.Mackin，永井保夫：ベイジアンフィルタを利用したWeb推薦システム，日本ソフトウェア科学会第26回大会3A-2，Shimane，JSSST，2009.

[9] 庭野正義，K.J.Mackin，永井保夫：ベイジアンフィルタを利用したWebページランキングシステム，社会システムと情報技術研究ウィーク，Hokkaido，SIG-AI，2010.

[10] POPFile - Automatic Email Classification - Trac ：http://getpopfile.org/pp.115-120.

[11] 高須賀清隆，丸山一貴，寺田実：閲覧履歴を利用した協調フィルタリングによるWebページ推薦とその評価，電子情報通信学会技術研究報告Vol.107，No131（2007），pp.115-120.