モバイル環境における地理情報の関連を利用したWebキャッシュ管理

Web Cache Management Using Geographical Relationships for Mobile Environment

五島一将[†] 李龍[†]
高倉弘喜[††] 上林弥彦[†]

Kazumasa Goshima,[†] Ryong Lee,[†] Hiroki Takakura[††]
and Yahiko Kambayashi[†]

モバイル環境では端末に重要なデータがキャッシュされるが、従来のキャッシュアルゴリズムはそのまま適用できない。例えば直前に訪れた場所の情報は、時間的に近くとも不要になる。また、Webページ以外にもURLやメタデータなど性質の異なるデータを同時にキャッシュしなければならない。本研究では、メタデータによるWebキャッシュ管理を効果的に用いたモバイル旅行ガイド支援システムを提案し、そのためのWebページ重要度計算手法を提案する。本システムにより、モバイルでのWebキャッシュ管理に加えて能動的ガイド機能なども提供できる。ページ重要度の判定には、我々が開発中の地域Web検索システムKyotoSEARCHを用いてWeb空間から抽出した地名・キーワード間の関連を用いる。ユーザ状態としての現在位置・行動履歴・目的・キーワード、および外部環境（天気など）を基礎に、キャッシュ内の各ページに関連する地名・キーワードを求め、これら相互間の関連の強いページほど高い重要度を与える。さらに、関連や状態を記述するメタデータの効率的なキャッシュ手法についても検討する。

1. はじめに

モバイル環境におけるWebからの情報収集が一般的になってきている。しかし、モバイル環境では未だ固定通信環境に比べて通信上の制約がある。したがって、モバイル端末上にWebデータを蓄えるWebキャッシュの重要性が増してきている。

キャッシュ管理における置き換えアルゴリズムの代表的なものにはLRU (Least Recently Used) やLFU (Least Frequently Used) などがある。しかしこれらは、いずれもデータに対するアクセス時刻やアクセス頻度のみを判断に用いている。これらの手法は本来、メモリデータの高速な出し入れのために用いられる手法であり、各キャッシュデータは等しいサイズでそれ単体では大した意味を持たないデータであることが前提にある。

しかしWebキャッシュはメモリキャッシュとは異なり、サイズもまちまちで、1ページあたり数KB～数十KBの大きさを持つ。そして何より、それ自体がユーザにとって重要な意味を持つデータであるという特徴がある。このように、キャッシュデータの内容や意味を利用したキャッシュ手法は、従来のキャッシュ管理手法data cachingに代わり、semantic cachingやcontent cachingと呼ばれ、近年の主要な研究領域となっている[1)~3)]。また、一旦キャッシュに保存されたWebデータのうち60%は再び使用されることはないという調査結果の示すように、Webキャッシュ特有の再利用率の低さもあり、従来のキャッシュの概念がそのまま適用できない。

そのため、Webページ自体の持つ性質や意味を積極的に利用したキャッシュ手法を用いる必要がある。特にモバイル環境では、ユーザの位置や興味、行動履歴や時刻などさまざまな状態を考慮しなければならない。例えば、ユーザが一旦訪れた場所に関するWebページは、そこについての情報を得る必要性は薄くなるため再度参照される可能性は低くなると考えられる。その他にも、現在地から近い場所や現在地に関連のある場所に関するWebページ、ユーザの興味ある事柄に関連したページなど、様々な要因が考えられる。そこで本研究では、地理情報をはじめとした様々なユーザ状態をメタデータとして利用したモバイルWebキャッシュの管理手法につ

† 京都大学大学院情報学研究科社会情報学専攻
Department of Social Informatics, Graduate School of Informatics, Kyoto University
{gossy, ryong, yahiko}@db.soc.i.kyoto-u.ac.jp
†† 京都大学大型計算機センター研究開発部
Data Processing Center, Kyoto University
takakura@rd.kudpc.kyoto-u.ac.jp

いて述べる。

本研究では、メタデータによるWebキャッシュ管理を用いたモバイルアプリケーションとして、能動的ガイド機能を備えた旅行ガイド支援システムを提案する。このシステムでは、出発前（プランニング）に検索に必要なメタデータを多くモバイル端末に保存し、屋外の検索場面（ガイド）でこれを活用する。実際のコンテンツをモバイル端末に格納するには多くの領域が必要となるが、 Webページを検索するためのインデックス情報をメタデータとして保存することにより、種々のユーザ状態を効果的に用いた効率のよい検索をすることができる。例えば、旅行中のプランニングを1日ごとに行うと仮定すれば、「夕方を過ぎたので、今日これまでに訪れたA寺とB公園の情報、そして閉館時間を過ぎたC美術館の情報はキャッシュから削除して記憶領域を広げる」というようなキャッシュ管理が可能になる。また、これらのメタデータを用いて、ユーザにとっての重要度が高いと判断されたページを提示することによる能動的観光ガイド機能や、ページの先読み機能などを実現することができると考える。

そこで本研究では、位置情報を持つWebページを対象に、位置情報どうしの関連の強さを用いたモバイルWebキャッシュのためのページ重要度判定アルゴリズムを提案する。位置情報の関連には、我々が現在開発を進めているWeb基盤空間情報検索システムKyotoSEARCH[4]により導かれた地名・キーワード間の関連を用いる。まず、キャッシュ内のWebページに対し、その位置情報および内容を特徴づける地名やキーワード（インデックス）に着目する。次にそれらのインデックスと、モバイルユーザの持つ現在位置や興味を表すキーワードとの関連をKyotoSEARCHより導く。そしてこれらの関連が強いものから順にユーザにとっての重要度が大きいページとして順位づけてゆく。最終的には地名・キーワードの関連以外の様々なパラメータを取り入れたモデルにまで拡張する。

以下、2章でまずアプリケーションとしてのモバイル旅行ガイド支援システムの提案を行う。 3章でWebにおける地名・キーワードの関連およびそれらを管理するKyotoSEARCHシステムについて説明する。 4章で関連を用いたキャッシュモデルとページ重要度判定アルゴリズムについて述べ、 5章で種々のユーザ状態パラメータを用いた重要度判定手法に拡張する。6章ではメタデータキャッシュの手法について議論する。最後に7章で重要度判定アルゴリズムのシミュレーションを行った結果と考察について述べる。

2. アプリケーション: モバイル旅行ガイド支援システム

本章では、各種メタデータによるWebキャッシュアルゴリズムを用いたアプリケーションとして、モバイル旅行ガイド支援システムを提案する。

モバイル環境では、

- 限られた記憶容量
- 小さなユーザインタフェースと不自由な操作環境
- 通信コスト

などの制約がある。このような制約下で、効率のよい情報検索を行うためには、Webキャッシュの有効利用が欠かせない。本システムの特徴としては、メタデータとしての関連情報をモバイル端末に取り入れ、屋外に携帯することにより、ディスク容量の有効利用を図る。その結果、関連を用いたページ重要度判定によるWebキャッシュ管理や、動的観光ガイド機能などが実現できる。

2.1 メタデータの携帯

モバイル環境においてコンテンツを利用するための形式として、次のパターンが想定される。

(1) 必要なコンテンツをモバイル端末にあらかじめ保存しておく
現在のように屋外通信インフラが整備されていなかった時代は、ノートPCを用いたこの形態のモバイルコンピューティングが主流であった。この形式では、コンテンツ保存のために多量の記憶領域を必要とするうえ、情報が常に最新の状態に更新されるというWebの優位性を享受することができない。

(2) クライアントはコンテンツを持たず、ほぼ全てのコンテンツを通信により取り入れる
現在急速に普及してきている、携帯電話によるWeb閲覧環境がこの形態を取っていると言える。この形式の利点は、クライアント側の記憶領域が非常に少なくてすむ点である。一方、目的の情報にたどり着くまでには、その途中に位置するコンテンツもいちいち取り込んで確認しなければならない。この点は、屋外でのモバイル環境では、操作性や通信コストの面を考慮すると、デスクトップ環境の場合と比べてコストが大きいといえる。

(3) コンテンツそのものではなく、検索を助けるメタデータをモバイル端末に格納しておく
これは(1)と(2)の中間に位置するとも言える形態である。具体的には、 Webページのコンテンツ自身ではなくURL部分をモバイル端末に保存

し、併せて目的とする URL をオフラインで探し出すために必要な情報をメタデータとして保存する。コンテンツを含まない URL 情報およびメタデータだけならば、比較的少ないデータ量で多くの有益な情報を得ることができる。また、通信コストも最終的な目的 URL をもとに 1 個のコンテンツをダウンロードするコストだけで済む。この形式は、数 MB ～数十 MB のメモリを持つ PDA に適した形式であると考える。

本システムでは、(3) のアプローチを基本として、対象地域に限定した Web ページの URL、それらのページに関連する地名、キーワードとそれらの関連リンクをモバイル端末に格納する。加えてユーザの訪問予定地順序や興味あるキーワードなどの情報も併せてメタデータとして携帯する。また、昨今の大容量記憶の技術進歩により、モバイルでもある程度大きなデータも扱えるようになっていることを考慮すると、(1) のアプローチも全くの不可能ではない。そこで、残りの記憶領域を用いて、必要となる可能性の高い Web ページのコンテンツはあらかじめ Web キャッシュに保存しておくことができる。このことの概念図を図 1に示す。

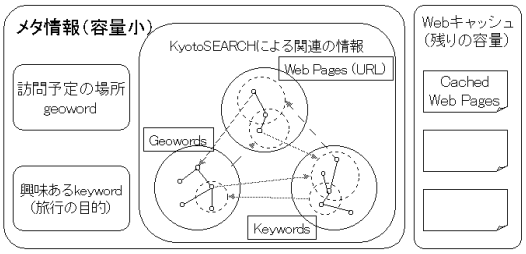

図 1 モバイルに格納・携帯するメタデータ

2.2 動的観光ガイド

メタデータを用いた Web キャッシュ手法は、より汎用的に応用することができる。その一例として、能動的な観光ガイド機能がある。

本稿で用いられるページ重要度判定では、モバイルユーザの現在位置やその他の状況を総合的に考慮したうえで、ユーザにとって最も興味があり有用であると思われるページを高重要度となるように算出している。また、キャッシュ内のページだけでなく、メタデータとして保存されている URL にも同様のアルゴリズムを用いて重要度を求めることができる。

そのため、最も重要度の高い Web ページをシステムが能動的にユーザに提示することで、「この場所を訪れてみてはどうですか」というような観光ガイドの役割を提供することができると考える。さらに、この機能を Web ページの先読み（プリフェッチ）に適用することで、モバイル環境における通信速度や通信コストの制約をカバーすることができる。

2.3 システム概要

本システムは、後述の KyotoSEARCH より得られる関連情報やユーザ状態などのメタデータを用いて、モバイル環境での Web 検索におけるキャッシュ管理および動的ガイド機能を提供する。

本システムでは、動作の場面を出発前に自宅で行う「プランニング」と、旅行中の「検索・ガイド」の 2 つのフェーズに分けて考える。プランニングでは

- Web ブラウジングによる目的地エリアの情報収集
- 訪問地や訪問順序の意思決定
- メタデータのモバイル端末への格納

を支援する。検索・ガイドでは

- ページの重要度に基づいた Web キャッシュ管理
- 重要度の高いページの提示による観光ガイド

を行う。

2.3.1 プランニング

プランニングは、自宅の PC など十分な固定通信環境での作業を想定している。プランニングでは、目的エリアの中でユーザが訪れようとする場所をある程度まで選んで決定する作業と、現地でのガイドに必要なオフラインデータをあらかじめダウンロードして PC 経由でモバイル端末に保存する作業を行う。

ユーザは、必要なメタデータをモバイル端末に格納する。必要なメタデータとは

- geoword, keyword, URL のリスト、およびそれらの関連を示したリスト
- 訪問予定地の geoword およびユーザが興味のある keyword
- ユーザ状態（所持金等）の初期値

を指す。これらを PC 経由で端末に保存する。

また、残りの記憶領域のうち Web キャッシュとして利用できるスペースには、コンテンツをダウンロードしておく。それにより、モバイルでの検索時に通信を行う回数を減らすことができる。

2.3.2 検索・ガイド

検索・ガイドのフェーズでは、各種メタデータを用いた Web キャッシュを用いて Web 検索を支援するとともに、動的観光ガイド機能を提供する。

動的観光ガイド機能では、キャッシュ内で重要度上位のページを自動的に選択し、ユーザに対し「ここを訪れてはどうか」というように、能動的に提示する。これにより、ユーザの位置・興味・その他の状態を反映した、動的な観光ガイド機能が行われる。

3. Web から導かれる関連

本章では、最初に Web 空間から抽出される地名・キーワードとウェブページの間の関連について述べ、次に我々が開発中の地域 Web 検索システム KyotoSEARCH についての説明を行う。

3.1 関連モデル

Web に存在する情報を名詞に着目して分類すると、地理情報を表す名詞（地名）とそれ以外の単語に分類される。本稿では前者を geoword(G)、後者を keyword(K) と呼ぶことにする。

これらの geoword および keyword 相互間には、多くの関連を発見することができる。それらの関連には (1)geoword と geoword を結ぶ関連（G-G 関連）、(2)keyword と keyword を結ぶ関連（K-K 関連）、(3)geoword と keyword を結ぶ関連（G-K 関連もしくは K-G 関連）の 3 種類がある。関連の具体例として考えられるもののひとつに、同一ページ中に同時に出現する共起関係による関連がある。

さらに、geoword とそれを含む Web ページの関連（P-G 関連）、keyword とそれを含む Web ページの関連（P-K 関連）も考えることにより、相互に関連しあう 3 つの集合から成る Web 情報空間のモデルを考えることができる。この概念図を図 2に示す。

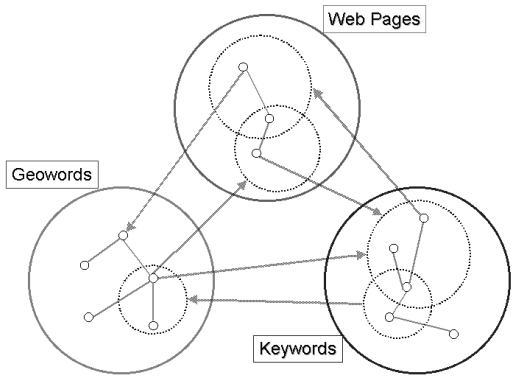

図 2 Web 空間の関連モデル

3.2 KyotoSEARCH

Web 基盤空間情報検索システム KyotoSEARCH は、特定の地域に関する情報検索・分析を行うシステムである[4]。KyotoSEARCH では、Web 空間を解析して、そこに含まれる地名と地名の関連、地名以外の単語（キーワード）どうしの関連、またそれら相互間の関連を抽出し、効率的な地域情報の検索・ナビゲーションに利用することを目指している。現在は、「京都」に関する Web ページを約 200 万ページ収集し、それらを対象に形態素解析を用いて名詞を抽出し、地名 (geoword) とそれ以外の名詞 (keyword) に分類し、共起関係などの関連を調査している。また、図 3に示すインタフェースを持ち、地図および関連相関グラフを用いた直感的で効率の良い地域情報検索ナビゲーションを提供する。

本研究では、KyotoSEARCH の持つ geoword, keyword および Web ページ相互の関連情報やそれらの管理機能を、次章以降で述べる Web ページ重要度判定アルゴリズムのために利用している。

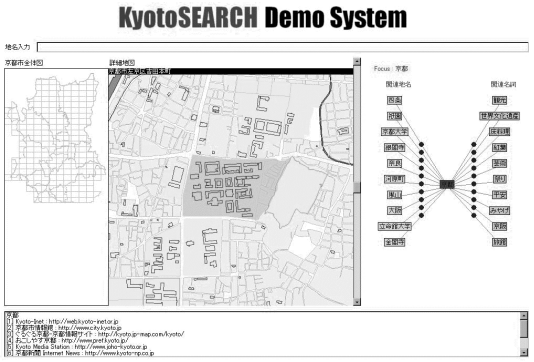

図 3 KyotoSEARCH インタフェース

4. 位置情報とキーワードの関連を用いたキャッシュモデル

本章では、モバイル環境において Web ページのコンテンツを蓄えている Web キャッシュに対し、ユーザ状態に基づく重要度を算出し、重要度の低いページから置き換え対象とするためのアルゴリズムについて論じる。前提条件として、モバイル端末にはその地域に関する多数の Web ページの URL と、種々の地名・キーワードとの関連が格納されていることを想定している（2章参照）。ユーザ情報には様々な要素（パラメータ）があるが、本章では最初に位置情報（地名）のみに基づくモデルを導入し、次にユーザの興味を表すキーワードを利用する段階にまで拡張する。

4.1 ユーザ状態

本稿で扱うユーザ状態には、現在位置、ユーザの興味を表すキーワード、訪問履歴、現在時刻、所持金、天候、などのパラメータが考えられる。そのうち本章で扱う基本モデルでは、KyotoSEARCH より得られる関連情報を用いるため、地名とキーワードを中心に考える。そこで、ユーザ状態 US を次のように定義する。

$$US(g_U, k_U, PARAM_1, \cdots, PARAM_n)$$

g_U はユーザの現在位置を表す geoword、k_U はユー

ザの興味を表す keyword である。$PARAM_i$ は、その他のユーザ状態を表す各種パラメータであり、次章で詳しく述べる。

4.2 キャッシュ構造・アルゴリズム

4.2.1 G-G モデル

まず、簡単のために、K-K 関連・G-K 関連（K-G 関連）は使用せず、G-G 関連（および P-G 関連）のみを用いたモデルを考える。このモデルを G-G モデルと呼ぶことにする。

G-G モデルは、図 4に示されるグラフ構造を持つ。ノードは、３つのグループに分類される。１つ目のグループであるページキャッシュ P は、保存されている Web ページ p の集合である。２つ目のグループである geoword 集合 G は、各 p と関連している geoword g の集合である。この第２グループを P のインデックスと呼ぶことにする。３つ目のグループはユーザ状態 US による単一のノードである。G-G モデルでは US の要素のうち g_U のみに着目する。

g_U と各インデックス g の間には、重み付き G-G 関連が存在する。便宜上、G-G 関連が存在しない対については重み 0 の G-G 関連が存在するとして扱う。

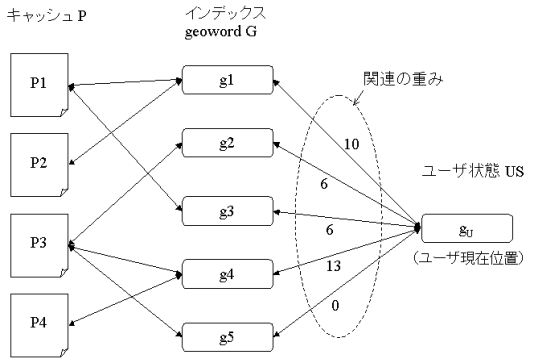

図 4 G-G モデル

4.2.2 G-G モデルにおけるアルゴリズム

本アルゴリズムの基本方針は、キャッシュ P に対し、geoword 集合 G をインデックスとしてユーザ状態（現在位置）g_U と最も関連の強いページ p を調べ、それを最も重要度の高いページとして判断し順位づけることにある。そのため、まずインデックス G の各要素とユーザ状態 g_U との関連の重みを調査する。それらの関連およびその重みは、KyotoSEARCH より得られるものとする。その結果最も重みの大きい geoword が g_1 であれば、g_1 と関連するページを最も重要度の高いページとして取り出す。以降、残りのページ集合に対して同様の操作を実行することにより、キャッシュ P 内の全ページについて重要度の順位が決定される。

ただし、ここではっきりしない問題点が生ずる。ある 1 個の geoword に関連するページは必ずしも 1 個ではなく、複数存在する場合がほとんどである。これらのページ群に対しても、同順位ではなく順位の大小を明らかにしたい。この場合は、対象ページに対して張られる他の geoword からの関連を考慮して決定するのが一般的である。したがって他の geoword からの関連の影響も考慮した判断手順を明確にする必要がある。

しかし、ここで図 5のような問題が生ずる。g_U に最も関連のある g_4 と関連するページは p_2 であるが、p_1 の方が関連の総和が大きいため、p_1 の重要度が高いようにも見える。どちらのパターンをより重要度が高いと判断するのが妥当であろうか。

本研究では「関連の重み」が具体的にどのような状態を表しているかには言及していないため、どちらの重要度が高いかについて的確な結論を出すことは難しい。そこで、仮に geoword 間の G-G 関連における重みの内容を「実距離の近さ」または「性質の類似度」と仮定する。そして具体例として京都の地名を用いて、

$g_U =$ 河原町, $g_1 =$ 大原, $g_2 =$ 鞍馬,
$g_3 =$ 嵐山, $g_4 =$ 祇園

と対応させて考える。ここで河原町および祇園は、どちらも飲食店等の多い繁華街であり、地理的にもほぼ隣接している。一方残りの大原・鞍馬・嵐山は、河原町や祇園とは離れており、繁華街とは性質を異にする観光地である。

つまり、遠くてあまり似ていない複数の場所について書かれた Web ページと、近くて性質の類似した場所について書かれた Web ページとを比較してみる。この場合、河原町にいるユーザは近くてよく似た感じの街である祇園に関連したページを求めることが多いと考えるのが自然である。もっと極端な例として、もし p_1 が遠く離れた町の電話帳だったらどうだろうか、ということも考えてみれば、p_2 より p_1 の重要度が高くなるということはほぼ考えにくい。そこで、本研究では重みの総和ではなく最高値を重要度判定のうえで重視するという立場で議論する。

4.2.3 アルゴリズム

前節の基本方針に基づき、本アルゴリズムを記述するため、まず以下のような定義を導入する。

定義 1 geoword g_1, g_2 の間に存在する G-G 関連の重みを、$Weight(g_1, g_2)$ と表す。g_1 と g_2 の間に G-G 関連が存在しない場合は、$Weight(g_1, g_2) = 0$ とする。

定義 2 ページ p_i と、p_i に関連する geoword g_j があ

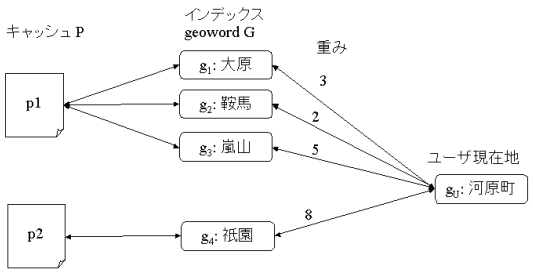

図 5　重みの総和と最高値の比較

る場合、g_j から得られる p_i の重要度を g_j による p_i のスコアと呼び、$Score[g_j](p_i)$ と表す。G-G モデルにおいては、$Score[g_i](p_i) = Weight(g_i, g_U)$ とする。

定義 3　ページ p_i に関連する geoword の総数が n 個であり、それらを $g_1, ..., g_n$ とする場合、集合 $\{Score[g_1](p_i), ..., Score[g_n](p_i)\}$ を p_i のスコア集合と呼び、$ScoreSet(p_i)$ で表す。また、$ScoreSet(p_i)$ の要素数を $|ScoreSet(p_i)|$ で表す。

定義 4　ページ p_i のスコア集合 $ScoreSet(p_i)$ の要素のうち、d 番目に大きいスコアを p_i の第 d スコアと呼び、$MaxScore(p_i, d)$ と表す。$(1 \leq d \leq |ScoreSet(p_i)|)$

以上の定義をもとに、G-G モデルにおけるページ重要度判定アルゴリズムを以下のように定義する。

まず、ページ集合 P および比較する対象スコアの深さ $depth$（第 $depth$ スコアで比較するか）を与えるとその中でスコアが最大となるページ p_{max} を返す関数 $GetMax(P, depth)$ を考える。$GetMax(P, depth)$ は次の再帰的手続きで表される。

```
GetMax(P, depth) {
  P' ← {p_i ∈ P | MaxScore(p_i, depth)が最大};
  if (P' がただ 1 つの要素 p' を持つ) p_max ← p';
    else p_max ← GetMax(P', depth + 1);
  return p_max;
}
```

この関数 $GetMax(P, depth)$ を用いて以下の手順を実行する。

手順 1　キャッシュ P の各 p_i について $ScoreSet(p_i)$ を求める。

手順 2　$GetMax(P, 1)$ を実行し、戻り値 p_{max} を $Ranking$ に追加する。

手順 3　$P \leftarrow P - \{p_{max}\}$ を実行。

手順 4　P が空ならば終了。空でなければ手順 2 に戻る。

このとき $Ranking$ に追加された順序が、ユーザにとっての重要度の高い順序である。本アルゴリズムによって得られた結果をキャッシュ置き換え手法として用いるためには、キャッシュ内のページのうち最も重要度が低いと判定されたページから順に置き換え対象とすればよい。

4.2.4　G-GK モデル

G-G モデルではユーザ状態 US のうち、現在位置を表す geoword g_U にのみ着目していた。本節では、着目する範囲をユーザの興味を表す keyword k_U にも広げた G-GK モデルに拡張する。

スコアの算出には、G-G 関連と G-K 関連の両方を用いる。G-GK モデルでは、定義 2 を次のように変更することで適用できる。

定義 2′　ページ p_i と、p_i に関連する geoword g_j がある場合、g_j から得られる p_i の重要度を g_j による p_i のスコアと呼び、$Score[g_j](p_i)$ と表す。G-GK モデルにおいては、

$$Score[g](p) = a \cdot Weight(g, g_U) + (1-a) Weight(g, k_U)$$
$$\text{ただし} 0 \leq a \leq 1$$

とする。

ここで a は (重みにおける)geoword 重視度を表す定数である。この値は、必要に応じて適宜定められる。ユーザが「このキーワードに非常に興味があり、多少遠い場所でも訪れてみたい」と考えているならば a を小さくすればよく、「キーワードにはあまりこだわらず、近い場所から効率よく回りたい」と考えているならば a を大きくすればよい。なお、$a = 1$ とすれば G-G モデルと同値となる。

このようにして求められたスコアに基づき、G-G モデルの場合と同じアルゴリズムで重要度順位を計算できる。

4.2.5　GK-GK モデル

前節まででは、インデックスとして geoword のみを用いてきた。ここでは、インデックスに geoword と keyword の両方を用いる GK-GK モデルを導入する。

GK-GK モデルでは前節の例に倣い、スコアにおける geoword 重視度 b $(0 \leq b \leq 1)$ なる定数を導入し、b により修正されたスコア $WeightedScore$ を次のように定義する。

定義 5　GK-GK モデルにおけるスコア $WeightedScore$ は、$c \in$geoword のとき

$$WeightedScore[c](p) = b \cdot Score[c](p)$$

$c \in$keyword のとき

$$WeightedScore[c](p) = (1-b)Score[c](p)$$

とする。ただし c はインデックス（geoword または keyword）である。

つまり、p のスコア集合のうち、geoword によるスコアには b を乗じ、keyword によるスコアには $(1-b)$ を乗じることで、各スコアに geoword 重視度を反映させる (図 6)。ここで、スコアにおける geoword 重視度 b は、重みにおける geoword 重視度 a と共通の値を用いてもよく、別の値を設定してもよい。

重要度順位を求めるアルゴリズムのためには、スコア集合 $ScoreSet(p)$ を求める際に、$Score[c](p)$ の代わりに $WeightedScore[c](p)$ を利用する。なお、GK-GK モデルでも同様に、$b=1$ とすることで G-GK モデルを表すことができる。

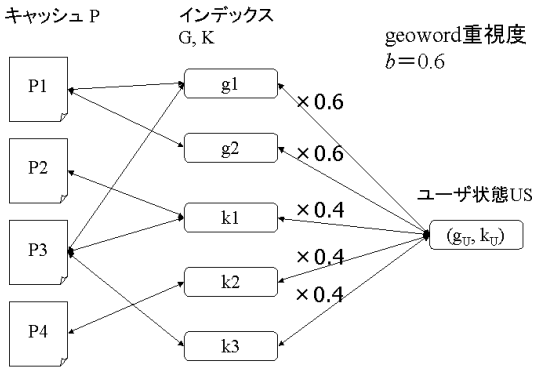

図 6 GK-GK モデル

5. ユーザ状態の反映

前章までの議論で、ユーザ状態のうち Kyoto-SEARCH により関連が与えられる geoword と keyword を用いた計算モデルを扱ってきた。しかし、よりユーザの需要を反映した重要度判定のためには、これらの関連を考慮するだけでは不十分である。

例えば、ユーザが次に訪れる場所を決めようと思って周辺の場所についての情報を Web ページから得ようとしている場面を考える。ユーザはシステムに対し、現在位置および興味のあるキーワードに関連の深い場所についての Web ページを問い合わせる。ここで返ってきたページを見てみると、ついさっき訪れたばかりの場所についての Web ページであった、ということも充分あり得る。このような場合、ユーザはまだ訪れたことのない場所についての情報をより強く求めるのが自然である。すなわち、Web ページの重要度は、ユーザの訪問履歴によっても変化する。

他にも、例えば正午近くなって昼食をとる場所を探すような時間帯になると、飲食店の多い繁華街のような場所に関する Web ページは重要度が増すと考えられる。さらに、今後は携帯電話などを利用した電子的な決済が普及してくることが考えられるが、そのようになればユーザの所持金情報をもとに、高級な店の多い地域や安い店の多い地域を区別して重要度を変えることも可能になる。

このように、真にユーザの状態を反映した重要度判定を行うには、様々なパラメータを用いることが必要になる。本章では、これら関連以外のパラメータをページ重要度判定に導入するための手法について議論する。

5.1 パラメータによるスコア操作

スコア $Score[c](p)$（GK-GK モデルでは $WeightedScore[c](p)$）の値のパラメータによる操作を行うため、まずパラメータ $PARAM$ を以下のように定義する。

定義 6 スコア増減パラメータ $PARAM$ は、3 つ組 (V, f, α) で表される。

V は、スコア増減の判断基準となる値の集合である。訪問履歴や所持金などユーザの状態を表す値や、時刻・天候など外部環境を表す値がこれに該当する。f は、インデックス c（geoword または keyword）と値集合 V より c についてのスコア増分を与える関数 $f(c, V)$ である。α は、$PARAM$ の重視度を表す定数である。

パラメータ $PARAM$ は、同時に複数設定することができる必要がある。そのため各パラメータを $PARAM_i(V_i, f_i, \alpha_i)$ と表すことにする。

このように定義された各 $PARAM_i$ により調節されたスコア $FixedScore$ を次の式で定義する。

定義 7 ページ p のインデックス c によるスコア $FixedScore[c](p)$ は

$$FixedScore[c](p) = Score[c](p) + \sum_i \alpha_i \cdot f_i(c, V_i)$$

優先度計算アルゴリズムを実行する際には、各 $ScoreSet(p)$ の要素として $Score$ の代わりに $FixedScore$ を用いる。つまり、インデックス c と状態値集合 V により決まる増減値をスコアに加算することにより、各パラメータに基づくスコア操作すなわち重要度操作を実現する (図 7)。その際、そのパラメータ $PARAM_i$ がどれだけの影響力を持つべきかの度合いが $PARAM_i$ の重視度 α_i により示される。

パラメータによる操作の代表的な具体例を以下に示す。

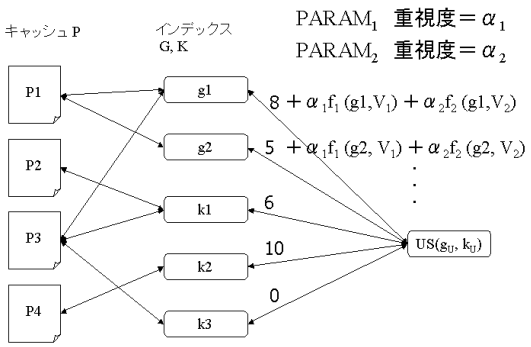

図 7 パラメータによるスコア操作

5.1.1 訪問順序

ある時刻におけるユーザにとって、過去に訪問したことのある場所に関する情報は、重要度が低くなることが多い。しかも、より最近（現在に近い時刻）に訪れた場所ほど、再度訪れようとする動機はより低くなり、重要度もより低下すると考えられる。一方、もしユーザの訪問順序があらかじめ決まっているならば、現在以降に訪れる予定の場所に関する情報は重要度が高くなり、その中でも直後に訪れる場所に関する Web ページの重要度ほど高くなると考えられる。このことを図に表すと図 8のイメージになる。そこで状態値集合 V に訪問予定順序を表す geoword 列 $\{g_1, g_2, \ldots, g_n\}$ および訪問履歴 $\{g_{-1}, g_{-2}, \ldots, g_{-m}\}$ を追加する。関数 f の値は図 8のように、直後の訪問地が最高値、直前の訪問地が最低値を取り、現在から遠ざかるに従い 0 に近づくように設定する。例えば、現在時刻を t_0，時刻 t に訪れる（訪れた）場所を g_t として、

$$f(g_t, V) = \frac{1}{t - t_0}$$

というような式で表すことができる。

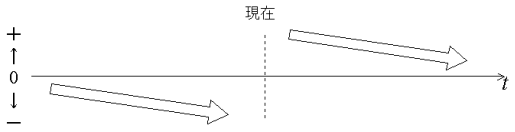

図 8 訪問順序と重要度の増分

5.1.2 時間帯

飲食店の多い繁華街の地名などは、他地域との関連に関わらず一般に昼や夜などの食事時間帯には需要すなわち重要度が高くなる。また観光施設などではオープンしている時間帯や定休日などの影響を考えなくてはならない。その他、祭りなどの期間限定のイベントなどもこのパラメータに該当する。そこで、状態値集合 V として現在時刻 t を設定する。増分関数 f に関しては、別データベースにより geoword と重要度を高くする時間帯との対応表を用意してそこから導く。時間帯により優先度を変化させることができる。

5.1.3 所持金

携帯電話等を用いた電子的な決済が、近い将来一般的に利用されると期待されている。このような環境では、モバイルによりユーザの所持金を管理することが可能になるため、このデータも Web ページ重要度判定のパラメータとして利用できる可能性がある。これにより、多くのお金を所持しているユーザには高級店の建ち並ぶ地域や入場料の高い観光施設などを優先的に案内し、逆に所持金の少ないユーザには使う金額の少なくてすむ地域や施設に誘導するといった使い方が可能になる。

この場合、状態値集合 V にはユーザの現在の所持金額 m をセットし、増分関数 f としては時間帯をパラメータにする場合と同様、 geoword と金額帯の対応データベースを用意する。

5.2 ユーザ状態入力の自動化

前節でユーザ状態を表す種々のパラメータについて議論した。これらパラメータの値を獲得するにはユーザが値を直接入力するのが最も単純な方法であるが、ユーザの負担を少しでも軽減するため、データ入力の自動化を考える必要がある。現在位置については GPS 等を用いることで解決するが、その他のパラメータについても可能な限り自動化することが望ましい。

ユーザ状態を判断する基準のひとつとして行動履歴の利用が考えられる。行動履歴は、 GPS 等により獲得した値を記録することにより簡単に得られる。行動履歴を解析することにより、例えば「レストランに 12 時 30 分から 1 時間滞在しているので、おそらく昼食を取ったのだろう」という推測ができる。それにより、所持金の値を 1,500 円程度減らしたり、それ以降は飲食店の重要度を下げるようにしたりといったユーザ状態操作が可能になると考える。

6. メタデータキャッシュ

前章までで述べた手法では、ページ重要度判定のための様々なメタデータを使用する。なかでも、最も重要なのは KyotoSEARCH より得た geoword, keyword, Web ページ間の関連情報である。これらの関連情報は、実際の Web コンテンツを含んでいないため、実際に Web コンテンツを収集したものと比べてかなりのデータ量が圧縮されている。加えて、昨今の記憶装置の小型化・大容量化により、モバイル機器といえども PDA で

数 MB ～数十 MB のメモリ、小型ノート PC ならば数百 MB のメモリと数十 GB のハードディスクを備えるまでになっている。そこで、関連情報などのメタデータは、その全てをモバイル端末に格納してオフラインで参照できることが通信コストなどの面からも理想である。

しかし、現在 KyotoSEARCH により収集されている京都に関する Web ページは、およそ 200 万ページにのぼる。仮に地名を「吉田本町」などの町名とし、関連を同一ページ内の共起関係としてその全てを解析、抽出する実験を行ったところ、その関連を記述したファイルはギガバイト単位の大きさとなった。対象となるページがこれだけの量に及ぶと、そのメタデータのみを抽出したと言っても無視できないほど大きなデータ容量となってしまう。

そのため、これらの関連情報の中から、必要と思われる情報を選別し抽出するメタデータキャッシュを行う必要性が出てくる。具体的には、geoword, keyword, Web ページによるノードとこれらの関連を表すリンクにより構成されるグラフの適切な部分グラフを求めることが必要となる。

部分グラフの求め方としては、様々な手法が考えられるが、ここでは次の手法について検討する。

対象地域をさらに限定する方法 KyotoSEARCH の対象とする地域は、「京都」全域である。しかし、ユーザの目的によっては、京都の中でも「左京区」「東山区」あるいは「洛北（京都市北部）」「洛西（京都市西部）」など、限られた地域だけを対象にすればいい場合がある。

このような場合には、地域を限定することで KyotoSEARCH のメタデータ容量を減らすことができる。まず、全ての geoword 集合の中から GIS の知識を用いて対象となる地域に含まれる geoword を抽出する。仮に「左京区」について地域を限定すると、その下位レベルにあたる地名である「吉田」「聖護院」やさらに下位に位置する「吉田本町」「聖護院西町」、または地理的に左京区内に位置する「京都大学」「下鴨神社」などが対象 geoword として抽出される。抽出した geoword 集合を G とする。

次に、G と関連する Web ページ集合 P を抽出する。この P が、対象となる地域 G に関連した Web ページ群である。この P をもとに、P と関連する keyword 集合 K を求める。最後にこれらをノードとする部分グラフを求める。これにより、地域 G に限定した Web ページとそれに関連のある geoword、keyword により形成される部分グラフが抽出される。

人気度の高い Web ページを中心に抽出する方法 メタデータを限定するための他の手法として、Web ページの人気度を用いることが考えられる。これは、Web ページの情報の質をもとにデータを絞り込もうという方針で、質の高いページは人気度も高いだろうと考え、ある一定値以上の人気度を持つページとそこから構築される部分グラフのみをメタデータとして使用する。

Web ページの人気度は、そのリンク（ハイパーリンク）構造によって算出される場合が多い。[5] では、Web ページの人気度をリンク数で割り、その値をリンク先の人気度に加算し、これを再帰的に繰り返すという手法でページの人気度を求めている。KyotoSEARCH 上に保存された Web ページについても、この手法により人気度を求めることができる。

ここで、地域を限定したページ人気度の計算について、リンク数が限られるために正しい人気度計算ができなくなるという問題が井上らにより指摘されている[6]。彼らは、ページ集合 W に対し、ある w_i から w_j にリンクを n 回遷移してたどり着く確率を p_{ij}^n として、重み $\sum_{i=1}^{n} p_{ij}^n$ となる有向グラフを構成することで十分な数のリンクを確保して人気度計算に用いるという手法を提案している。

7. シミュレーション

本稿で提案したページ重要度順位判定手法の妥当性を確かめるために、ユーザの現在位置を仮定して Web ページの順位づけをする実験を行った。順位づけ対象となる Web ページには KyotoSEARCH が収集した Web ページより 100 ページ、500 ページ、1000 ページを無作為に選び、現在地には「祇園」と「銀閣寺」の 2 種類を仮定し、計 6 パターンについて試行した。順位づけのアルゴリズムには、地名間の関連のみを対象とする G-G モデルを用いた。

その結果、主に現在地やその周辺について述べられているページを上位に順位づけさせることに成功した。しかし、各試行において以下のようなページが共通に見られた。

- 京都のニュース一覧（例：京都新聞の記事バックナンバー）
- レストランガイド（例： Yahoo! グルメの京都エリア）
- 京都全域の観光地一覧や寺社一覧

すなわち、現在地やその周辺を含み多くの地名が記述さ

れているページは、我々の手法ではどうしても上位に来てしまうことが判明した。このようなページが、現在地付近にエリアを絞って有益な情報を記載しているページを排除してしまう危険性がある。

このことを解決するためには、Web ページの示す地域的な広さや内容の深さ[7] なども考慮し、ページの持つ意味により深く踏み込んだ解析が必要となる。

8. おわりに

本研究では、モバイル環境における Web 検索を支援するための新しいキャッシュ手法について提案した。

まず、この手法を用いたモバイルアプリケーションとして、能動的ガイド機能を有する旅行ガイド支援システムを提案した。本システムでは、ユーザが求める Web ページを検索するために必要なメタデータとして、地理情報の関連やユーザ状態などの情報などをモバイル端末に格納し、携帯する。このことにより、モバイル環境における種々の制約の中で効率のよい Web 検索を実現する。あわせて重要度の高い Web ページを能動的に提示することで、ユーザ状態を反映した観光ガイド機能を可能にする。

そのために、ユーザ状態の基づき Web ページの重要度を算出する手法について議論した。地域情報検索システム KyotoSEARCH により得られる地理情報やキーワードの関連を利用する。本手法では Web キャッシュとその中の Web ページに関連する地名・キーワード集合（インデックス）、およびユーザ状態の三者から成るモデルを考え、ユーザ状態とインデックス間の関連の重みをスコアとして用いることで重要度の順位付けを行う。また、関連以外のパラメータにより得られる値をスコアに加算することで様々な要因を重要度に反映させる。

今後は、システムを実装して実際に使用した場合の利用性検証を行うほか、 7章に示した問題点への対策として、地域情報の詳細度について「詳しい情報」「大まかな情報」を判別し、ユーザがどのような性質のページを求めているかが反映できるようにして判定精度の向上に取り組む予定である。

参 考 文 献

1) Q. Ren, M.H. Dunham, “Using Semantic Caching to Manage Location Dependent Data in Mobile Computing,” The Sixth Annual International Conference on Mobile Computing and Networking (MobiCom’00), August 2000.
2) B. Zheng, D. L. Lee, “Semantic Caching in Location-Dependent Query Processing,” LNCS 2121, pp. 97-113, 2001.
3) K. Cheng, Y. Kambayashi, “Multicache-based Content Management for Web Caching,” Proc. of the 1st International Conference on Web information Systems Engineering (WISE2000), Vol. 1, pp. 42-49, June 2000.
4) 李龍, 高倉弘喜, 上林弥彦, “地域ウェブ情報を利用した地域情報検索と地域分析,” 空間 IT ワークショップ, SIT01-2-2, Dec. 2001.
5) S. Brin, L. Page, “The Anatomy of a Large-Scale Hypertextual Web Search Engine,” Computer Networks and ISDN Systems, Vol. 30, pp. 1-7, 1998.
6) 井上陽介, 李龍, 高倉弘喜, 上林弥彦, “地域ウェブ情報検索のための対象を限定した Web ページの人気度,” DBWeb2001, Dec. 2001.
7) 山田直治, 李龍, 高倉弘喜, 上林弥彦, “地域的網羅度と詳細度を用いた新たな WEB 検索手法の提案,” 空間 IT ワークショップ, SIT01-2-3, Dec. 2001.
8) T. Tezuka, R. Lee, H. Takakura, Y. Kambayashi, “Web-based Inference Rules for Processing Conceptual Geographical Relationships,” WGIS Workshop at the 2nd International Conference on Web Information Systems Engineering, Dec. 2001.