J-036

Web 履歴データを用いた可視化による リソースアクセス向上の研究

Research of Improvement in Resource Access by the Visualization Using Web History Data

正谷 英樹† Hideki Masatani
吉田 博哉† Hiroya Yoshida

1. はじめに

インターネットの利用において，再度訪問したい Web ページを発見した場合，Web ブラウザのブックマーク機能を利用して，訪問先サイトの URL を保存する事がある．一方，ブックマーク機能を利用しなかったため，URL が保存出来ていない Web ページに対し，再度訪問したい場合，Web ブラウザによって蓄積された閲覧履歴データを確認する方法がある．しかし，閲覧履歴データは，時系列で整理されているものの，無選別で蓄積されている事から，データが膨大な分量になると，必要な情報を瞬時に発見出来ないというアクセシビリティに関する問題が挙げられる．

そのため，Web ブラウザが蓄積した閲覧履歴データを解析し，分類する事によって，利用し易い形式で可視化する手法が検討されている．例えば，閲覧履歴データの可視化に関する研究として，サムネイル画像を表示する方法[1]や，閲覧履歴データから Web サイトの推薦情報を可視化する方法[2]がある．これらの研究からも，可視化を実現する事で，利用者のアクセシビリティ向上を促進すると言える．

一方，スマートフォンを始めとする携帯端末において，閲覧履歴データを解析し，分類するアプリケーションを実現する場合，一般のコンピュータと較べて，ハードウェア性能が劣っている事から，処理に時間がかかり，必要な情報を瞬時に表示出来ない問題が挙げられる．そこで本研究では，前述の問題を解決するため，携帯端末に蓄積された閲覧履歴データをサーバ上で解析し，分類結果を端末に送信するシステムを提案し，シームレスな可視化システムの実現を目指す．

2. アプローチ

2.1 提案システムの概要

（1）全体構成

本研究では携帯端末による Web 閲覧履歴の分類においてシームレスな可視化システムを目指す．その際，携帯端末で解析や可視化の処理を全て行うとオーバーヘッドが発生する．そこで，携帯端末のハードウェア性能を考慮した，ストレスフリーなシステムを設計する必要があると考え，**図 1** に示すシステム構成を採用した．

本研究では**図 1** に示すように，まず，クライアントである携帯端末に蓄積された閲覧履歴データを Web サーバに送信する．次に送信された閲覧履歴データを構文解析し，カテゴリに分類する．その後，分類結果を携帯端末に送信し，端末側で可視化する．なお，本システムのクライアントとして，iPhoneG4（iOS5.2）を採用した．また，Web サーバとして，さくらインターネット VPS1G（centOS5.5）を採用した．

（2）システム動作の流れ

携帯端末上で動作するアプリケーションは Web ページを閲覧した際に，その都度，閲覧ページの URL を Websocket 通信[3]によって，Web サーバへ送信する．この際，Web サーバ側で送信された URL の HTML を取得し，ページ内の meta 要素における name 属性の値が keyword と description に対する content 属性値をコーパスとして取得する．取得したコーパスは形態素解析を行い，トークンへ分割する．なお本研究では，Websocket 通信として，Node.js-v0.8.0 を採用した．また，形態素解析エンジンとして，Mecab-0.992 を採用し，データベースエンジンとして，スキーマレスデータベースである MongoDB-v2.0.4 を採用した．

その後，分割したトークンに対して単語の出現頻度から閲覧先 URL をカテゴリに分類する．最終的に，携帯端末上で「履歴」ボタンが押下されると，サーバ上に蓄積された分類済み URL を携帯端末へ送信する．携帯端末は受信したデータを用いて可視化処理を行う．

2.2 閲覧履歴データの分類手法

（1）データ分類の概要

本研究では，閲覧ページの URL を特定のカテゴリに分類する際にベイジアンフィルタという分類手法を採用した．ベイジアンフィルタとは，ベイズ統計を用いたフィルタアルゴリズムである．本フィルタは，スパムメールに対する自動分類の手法として実装されている．また，このフィルタは分類データを学習する事で次第に精度が高くなると言う特徴が挙げられる[4]．他にも分類手法として用いられる

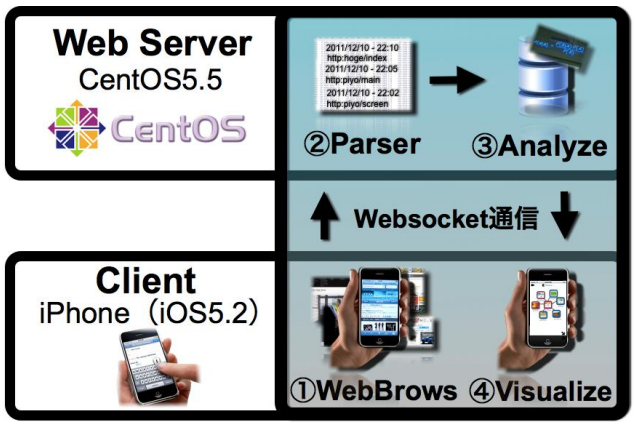

図 1 システム構成

† 神戸情報大学院大学，Kobe Institute of

アルゴリズムとして，SVM（Support Vector Machine）フィルタがある．なおベイジアンフィルタと SVM の性能比較を行った研究[5]では，SVM の方が高い正解率となった．一方，SVM を利用した場合，誤検出も多い結果となっている．そのため，本研究では，閲覧履歴のカテゴリに分類する際に，誤検出の少ない手法であるベイジアンフィルタを用いた分類処理を採用した．

（2）本システムで実装した解析手法

本システムで実装したベイジアンフィルタは，閲覧履歴データが複数のカテゴリに属するような分類方法を採用した．例えば，コンピュータゲームに関する Web ページを分類する際，利用者の目的に応じて，「ゲーム」や「プログラミング」といったカテゴリに分類する必要がある．一方，本研究では，利用者の目的を判別しない事から，Web ページを 1 つのカテゴリにのみ分類するのではなく，カテゴリに属しているか否かを判断するための閾値を設け，その閾値を越えた場合に限り，カテゴリに属しているとみなす．**図 2** に本研究で採用するベイジアンフィルタの処理フローに示す．

図 2 に示すように，閲覧した Web ページの HTML 文書をパースし，形態素解析エンジンを用いてトークンに分割する．このデータを用いて，各カテゴリに与えられたキーワード群（教師データ）における出現頻度を計算する．次に，算出した出現頻度を元に閾値を超えたカテゴリ群を確定する．最後に，Web ページ URL，閲覧日時，カテゴリ分類結果から構成される BSON データ型に変換の上，MongoDB へ格納する．

なお，本研究では，分類カテゴリとして，Yahoo!Japan のディレクトリ検索[6]に記載された 2 階層までのカテゴリを利用する．また教師データとして，Yahoo!Japan の各カテゴリに含まれる Web ページ群を利用した．

2.3 分類データの可視化手法

（1）可視化の概要

可視化とは，数値のみのデータを把握しやすくグラフィカルに表現する方法であり，グラフ表示や，地図といったポピュラーな手法として世間一般では認知されている．またデータの相関性や階層的なデータ構造を表現するためには，ツリー構造による可視化手法がよく用いられる．例えば，階層的なデータを 3 次元ツリー構造で表現し，奥行き

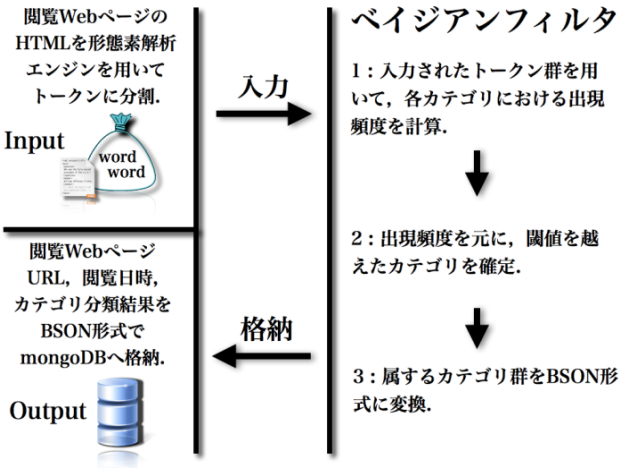

図 2　ベイジアンフィルタの処理フロー

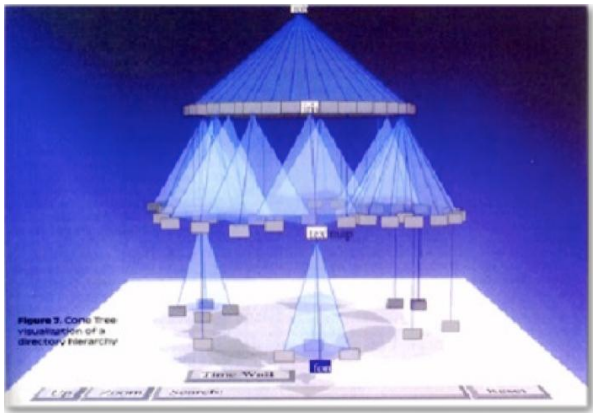

図 3　Corn Tree を用いた可視化の例

を持たせ，大量のデータを一度に表示する Corn Tree[7]がある．**図 3** にコンピュータのディスク階層を Corn Tree によって，可視化した例を示す．

一方，2 次元で同様の階層的なデータ表現方法として，2 次元平面で大きさ，色を使い分け，四角形で表現する Tree Map[8]や，相関性を 2 次元ツリー構造で表現する Topic Map[9]がある．特に，Topic Map は，主題（Topic）に基づく分類法として主題と関連（Associations）の集合によって表現する事が出来るため，インタラクティブな操作によって，膨大なデータを表現する事が出来る．Topic Map を利用した例として，**図 4** に示す著名人などの繋がりを示す，あの人検索[10]や，音楽アーティストのジャンル，関連性を示す discover music[11]などがある．

本研究では，携帯端末のハードウェア性能や画面サイズを考慮した結果．Topic Map を利用して，カテゴリ毎に分類した閲覧履歴を表現する．

（2）本システムで実装した可視化手法

本研究では，閲覧履歴データのカテゴリとの関連の見渡しやすさ，操作性やアプリの解りやすさから，可視化手法に階層的な可視化手法である Topic Map を採用した．その際，閲覧履歴ページを視覚的に把握しやすくする為に Web ページの魚拓サムネイルを使用する．本システムでは，閲覧履歴データが，Web サーバ上で管理されていることから，魚拓サムネイルも Web サーバ上で生成，保存する事で，ストレスフリーなアプリケーションを実現する．生成した魚拓サムネイルは，クライアント側で履歴表示の要求があった際に，Websocket 通信によって，クライアントに送信する．なお，Web サーバで魚拓サムネイルを生成する方法として，LinkLook[12]を使用する．

図 4　あのひと検索

3. アプリケーションの実装

3.1 実装アプリケーションの概要

本システムのクライアント端末は iPhoneG4（iOS5.2）を想定している．そのため，本アプリケーションは，iPhone アプリとして実装した．

本アプリケーションは，Web ブラウジングの機能を有しており，Web ページを閲覧する毎に指定の Web サーバへ閲覧履歴を送信する．送信されたデータは，随時解析され，カテゴリ毎に分類される．

なお，本アプリケーションは，利用者が履歴閲覧を確認したい場合，画面上の「履歴」ボタンを押下する事で「履歴確認」画面へ遷移する．「履歴確認」画面では，可視化方法として「カテゴリ」「検索」のいずれかを選択する．

3.2 カテゴリによる可視化

「履歴確認」画面より，「カテゴリ」を選択すると，**図 4 左**に示す「カテゴリ一覧」画面へ遷移する．「カテゴリ一覧」画面では，本システムで登録されたカテゴリ一覧がリスト形式で表示される．その後，カテゴリを選択すると，**図 4 右**に示す様に，ベイジアンフィルタリングによって分類された閲覧履歴データが Topic Map 形式で表現される．

本画面では，選択したカテゴリを示すアイコンが中心に表示され，その周囲に分類された閲覧履歴 URL のサムネイルが表示され，カテゴリの関連として表現される．

3.3 検索による可視化

「閲覧確認」画面より，「検索」を選択すると，**図 5 左**に示す「検索語句入力」画面へ遷移する．本画面において，テキストボックスに検索語句を入力し検索を行うと，検索語句のアイコンを中心に「1 日前」「7 日前」「30 日前」「その他」といった閲覧日時を表すアイコンが Topic Map 形式で表示される．これらのアイコンを選択すると，条件に合致した閲覧履歴データが Topic Map 形式で表現される．**図 5 右**に示す例では，検索キーワード「python」が含まれるカテゴリに絞り，かつ「7 日前」の履歴を表示している．なお，検索機能の一致判定は，各カテゴリの教師データによって与えられた Web ページのトークン群を利用する事で，比較的自由度の高い閲覧履歴を取得する事が出来る．

3.4 Topic Map の操作方法

Topic Map で可視化した画面では，以下の操作方法を実現した．まず，閲覧履歴を示すサムネイルをダブルタップする事で，選択サムネイルを中心に拡大表示する．さらに，サムネイルをタッチする事で閲覧履歴 URL をブラウザで表示する．さらにサムネイルを移動，操作する事も可能で，物理シミュレーション（バネモデルによる主題間の弾力性）による操作性を実現する．他にも，ピンチ操作によって，Topic Map 全体を拡大縮小する他，スライド操作によって視点位置を変更する事も出来る．これらの操作によって，Topic Map で表現した膨大な閲覧履歴データを全て確認する事が出来る．

4. 実証実験と考察

本研究で開発したアプリケーションを使用し，Web サイトを巡回する事で，閲覧履歴を蓄積した．その後，蓄積した閲覧履歴データを分類し，Topic Map で表現した画面の操作感を確認する事で，本システムの有効性を検証した．その結果，本システムを利用した方が，従来の閲覧履歴の利用方法よりも，アクセシビリティの向上が見られた．一方，検証を通じて，以下に示す課題が顕在化した．

図 4　カテゴリによる可視化

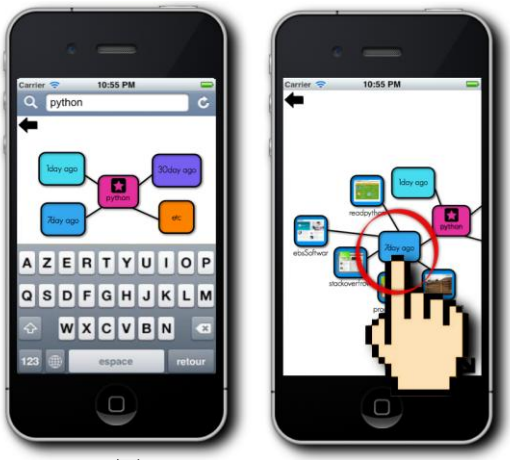

図 5　検索による可視化

- 閲覧履歴データのコーパスの質・量
- カテゴリの細分化，及びグルーピング化
- 画面の表示範囲制約
- Web 閲覧履歴データの機密性

本研究では，閲覧履歴データのコーパスとして，meta 要素の content 属性値の一部に限定している．meta 要素は，SEO 対策を施した Web ページに対しては，ページ内の内容とは直接関係の無い語句が含まれている場合がある．この事から，妥当性の高い分類が出来ていない可能性があると考えられる．また，本研究では，カテゴリを固定としており，生起データに対しての処理を行っていない状態である．カテゴリの細分化を行い，閲覧履歴データの分類の妥当性を向上する必要がある．

また，本研究のクライアント端末として利用した iPhone は，表示範囲の制約から，可視化画面において，大量の閲覧履歴データが表示された場合，グラフィックが重なり合い把握しづらくなるため，表示データ数を限定する必要がある．

最後に，本研究では，携帯端末のハードウェア性能を考慮した，ストレスフリーなシステム設計を行った結果，閲覧履歴データを Web サーバへ送信し分類処理を行っている．閲覧履歴データは，個人の趣味嗜好を表す内容である事からも，細心の注意を払う必要がある．

5. 今後の展望

今後の展望として，まず，コーパス量に関する課題は，HTML データ全体を取得対象とし，HTML タグの排除を行い Web ページ内全文に対して形態素解析を行ってコーパスの質・量の向上を努める．

次に，カテゴリに関する課題は，履歴閲覧データを逐次学習する事で生起データを生成できる実装を行い，カテゴリの細分化を行う．その方法として，特徴語句クラスタリングによる自動分類を施したベイジアンフィルタを実装する事で，精度の高い分類結果を算出する．

そして，画面の表示範囲に関する課題は，現在の Topic Map 形式の表現手法を改良し，奥行きを持たせる事によって，iPhone の画面範囲でも大量のデータを表現できるような 3 次元可視化手法を検討し，実装する．

最後に，機密性の課題は，閲覧履歴データの通信において，暗号化通信を行い，認証処理として Websocket 通信ではセッションキー，RSA キーを用いた暗号化通信を行う．また Web サーバでは厳重なセキュア体制，バックアップシステムの導入，データベースの認証設定を施す．

6. おわりに

本研究では，携帯端末に蓄積された閲覧履歴データをサーバ上で解析し，分類結果を端末に送信する事で，シームレスな可視化システムを実装し，その有効性を検証した．その結果，Topic Map を使用した可視化手法によって，Web 閲覧履歴が把握しやすく，見渡しやすい形となり，アクセシビリティの向上を図る事が出来た．一方で，課題として挙げられる項目から，現時点では，本研究で開発したアプリケーションを一般利用として公開するまでには至っていない．今後，解決策を検討の上，改善を行う．

参考文献

[1] 新美礼彦, 片山悠樹, 小西修 : サムネイル表示によるブラウザ履歴情報の可視化, 第 22 回ファジィシステムシンポジウム, 日本知能情報ファジィ学会, pp.273-278, 2006.
[2] 大森慎吾, 宇野達也, 大野成義 ：閲覧履歴を利用した Web ページ推奨の SOM による可視化, 第 2 回データ工学と情報マネジメントに関するフォーラム, 2010.
[3] Websocket : http://www.html5.jp/trans/w3c_websockets.html
[4] 田端利宏 ：SPAM メールフィルタリング:ベイジアンフィルタの解説, 情報の科学と技術, Vol.56, No.10, pp464-468, 2006.
[5] 北村祐貴, 狩野均 ：事前処理に k-means 法を利用したスパムフィルタの開発, 数理モデル化と問題解決研究会, 情報処理学会, Vol.2009-MPS-76, No.12, 2009.
[6] Yahoo!Japan ディレクトリ検索 : http://dir.yahoo.co.jp/
[7] George G. Robertson, jock D. Mackinlay, and Stuart K. Card : Cone Trees:Animated 3D Visualization Of Hierarchical Information, Proceeding CHI '91 Proceedings of the SIGCHI conference on Human factors in computing systems, pp.189-194, 1991.
[8] newsmap : http://newsmap.jp/
[9] Topic map : http://www.topicmaps.org/
[10] あのひと検索 spysee : http://spysee.jp/
[11] Discover music : http://discovr.info/
[12] LinkLook : http://kawika.org/jquery/linklook/