1N-9

アクセスログに基づくファイルとWebページの関連性抽出手法

宋 強† 渡辺 陽介†† 横田 治夫†††

† 東京工業大学工学部開発システム工学科
†† 東京工業大学学術国際情報センター
††† 東京工業大学大学院情報理工学研究科計算工学専攻

1 はじめに

近年、インターネットもストレージ技術も日々進歩しており、Web を利用しながら、ファイル編集を行うことが日常的になってきている。例えば、授業の課題問題を解くために、Web 上から情報を探しながら、課題レポートを書いた場合を考える。後でレポートを修正しようとする際に、レポートファイルだけではなく、当時見た Web ページももう一度見たいという要望が考えられる。既存技術によりファイル単体または Web ページ単体であれば、個別に検索可能であるが、双方を関連づけて提示することはできない。ファイルと Web ページの間の関連性を抽出することは、両者を対象とした検索における結果の改善に非常に重要である。

本研究の目的は、同じ作業に属するファイル群と同時に参照した Web ページ群を関連づけることである。ここでの「作業」というのは、一つの仕事単位を意味する。提案手法では、ファイル編集と Web ページ参照が頻繁に共起する組合せほど関係が強いと考えて、アクセスログからそのような関係を抽出する。

2 提案手法

本稿では、ファイルの読み書き等を記録したファイルアクセスログと、Web ページのアクセスログ (http プロキシログ [1] と firefox ブラウザログ [2]) の 2 種類のログデータからファイル群と Web ページ群の関連性を抽出する二つの手法 (**Pre-Merge 法**と **Post-Merge 法**) を提案する。両手法の内部処理において、アクセスログをトランザクションに変換し、頻出アイテム集合を抽出する処理を行う。最後の結果を仮想ディレクトリの形式で、ユーザに提示する。ユーザが仮想ディレクトリを見ることによって、ファイルシステム内に存在する同一作業に属するファイル群と Web ページの URL 群を発見することができる。

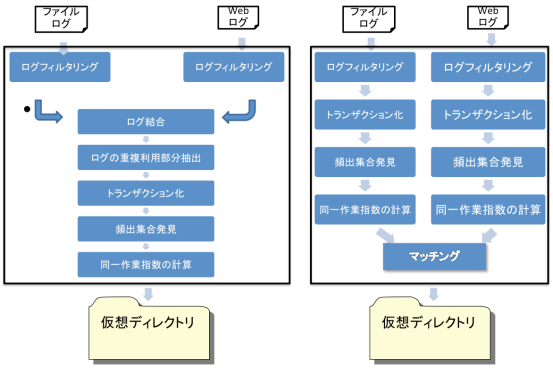

図 1: 処理手順 左 Pre-Merge 右 Post-Merge

2.1 Pre-Merge 法

Pre-Merge 法では、ログデータの段階で、ファイルアクセスログと Web アクセスログを統合し、統合されたアクセスログに対して頻出アイテム集合抽出を適用する。処理手順を図 1(左) に示す。

[ログフィルタリング] ファイルログに対し、特定の拡張子によるフィルタリングを行い、さらに一秒間に 4 ファイル以上の高頻度アクセスの記録を機械的な操作と判断して削除する。一方、Web アクセスでは、ユーザが目的のページに到着するまでに、いくつかのページを経由することが想像できる。途中で経由したページを見る時間は比較的短いと考えられるため、あるページへのアクセス時刻から 3 秒以内に他のページをアクセスしたら、このページを途中に経由したページだと判断して、解析対象から除外する。

[ログ結合] 両ログファイルのフォーマットを統一した上で、アクセス時間順にソートし、時間順にマージ

Extraction Methods for Relationship between Files and Web Pages based on Access Logs

Qiang SONG†, Yousuke WATANABE††and Haruo YOKOTA†††
†Department of International Development Engineering, Tokyo Institute of Technology
††Global Scientific Information and Computing Center, Tokyo Institute of Technology
†††Department of Computer Science, Graduate School of Information Science and Engineering, Tokyo Institute of Technology
{soukyou, watanabe}@de.cs.titech.ac.jp
yokota@cs.titech.ac.jp

していく。

[ログの重複利用部分抽出] ユーザがファイル編集作業を行っていないときに、単にWeb閲覧をする場合も考えられる。本研究ではそれらを解析対象外として、ファイルアクセスを行った時点の前後30分、計60分の間以外のWeb記録を除く。

[トランザクション化] アクセスログを一定期間 (パラメータ TransactionTime[sec]) ごとに分割し、その各区間で使われたファイル群を1つのトランザクションとして変換する。

[頻出集合発見] 頻出集合発見を行う。トランザクション群から、アプリオリアルゴリズム [3] を用いて、出現回数が一定数以上の組み合わせを抽出する。

[同一集合指数の計算] 頻出集合たちを極大化し、集合内の要素が類似する頻出集合同士を結合する。類似の度合いはDice係数を用いる。集合AとBに対する、集合同士の結合条件は以下の通りである。

$$\frac{2|A \cap B|}{|A| + |B|} \geq threshold$$

Dice係数がパラメータ thresholdの値以上の集合のみを結合する。

2.2 Post-Merge法

Post-Merge法では、ファイルアクセスログとWebアクセスログそれぞれに対して、個別に頻出アイテム集合抽出 [3] を適用し、最後に得られた頻出アイテム集合同士を時間情報に基づいて関連付ける。処理手順を図1(右) に示す。Post-Merge法では、一部の処理がPre-Merge法と共通であるからここでは省略し、最後のマッチングについてのみ説明する。

[マッチング] それぞれのログから頻出するものの集合を抽出した後、双方の集合で対応するものを求める。関連があるファイル群とWebページ群がほぼ同じ時間帯に利用されることに着目し、両方の集合の各要素のアクセス時間のオーバーラップ度合いに基づいて、マッチングを行う。2つの集合の各要素の任意のペアにおいて、同じトランザクションに共起した回数が一定回数以上であった場合、2つの集合を同一作業とみなして一つの集合に結合する。

3 比較実験

今回の実験では、1ユーザのアクセスログを用いた。ログの記録期間は2010/09/24 - 2010/11/21である。ファイルアクセスログのサイズは23,624,947byteで、ファイル数は472である。ブラウザログのサイズは2,036,315byteで、記録されたページ数は1731である。

表 1: ファイルログとブラウザログを用いた実験結果

手法	Precision	Recall	F 値
Pre-Merge	0.893	0.369	0.434
Post-Merge	0.190	0.375	0.236

表 2: ファイルログとプロキシログを用いた実験結果

手法	Precision	Recall	F 値
Pre-Merge	1.000	0.119	0.212
Post-Merge	0.208	0.030	0.052

プロキシログのサイズは52,332,311byteで、ページ数は3939である。その中から人手で3つの正解セットを作成した。ファイルアクセスログとブラウザログを用いた実験では、両手法の最適パラメータにおいてのPrecision, Recall及びF値が表1の通りとなった。処理時間はPre-Mergeが平均で48秒かかり、Post-Mergeが19秒かかった。ファイルアクセスログとプロキシログを用いた実験の結果が表2である。処理時間はPre-Mergeが80秒かかり、Post-Mergeが33秒かかった。これらの結果から、PostMerge法より、PreMerge法の性能が良いことが分かった。一方、処理時間から見ると、Pre-Merge法がPost-Merge法の約2.5倍の時間がかかることが分かった。

4 まとめと今後の課題

本論文では、ファイル群とWebページ群の間の関連性を抽出し、同一作業の両者を関連づけて、一つの仮想ディレクトリとしてユーザに提示するための二つの手法Pre-Merge法とPost-Merge法を提案した。

これからの課題として、Recallの向上、パラメータの自動決定方法と評価対象のユーザ数を増やすことなどである。

謝辞

本研究の一部は文部科学省科学研究費補助金特定領域研究 (#21013017) の助成により行われた。

参考文献

[1] Squid Cache, http://www.squid-cache.org/

[2] Firefox, http://mozilla.jp/firefox/

[3] R. Agrawal, T. Imieliński, and A. Swami, “Mining association rules between sets of items in large databases”, Proc. ACM SIGMOD, pp. 207-—216, 1993.