DEIM Forum 2012 E7-5

デスクトップサーチシステムのためのWeb利用履歴の活用手法

徳野　達也† 　小林　隆志†† 　阿草　清滋††

† 名古屋大学 工学部　〒 464–8601 名古屋市千種区不老町
†† 名古屋大学 大学院情報科学研究科　〒 464–8601 名古屋市千種区不老町
E-mail: †tokuno@agusa.i.is.nagoya-u.ac.jp, ††{tkobaya,agusa}@is.nagoya-u.ac.jp

あらまし　全文検索システムでは，画像などのキーワードを含まないファイルを検索することが困難である．我々はこれまでに，ファイルアクセス履歴から求めたファイル間関連度を利用することで，キーワードを含まないファイルを検索することが可能である．本研究ではファイルアクセス履歴以外にも Web 利用履歴を活用する検索手法を提案する．また，提案手法を実装した検索システムを用いた実験を通じて，提案手法の有効性を議論する．

キーワード　デスクトップ検索，PIM，Web 利用履歴

1. はじめに

電子的な文書を管理する際に，紙文書を管理する際と同じフォルダの概念を導入するべきかどうかは以前より議論されている [1]．適切に分類ができれば効果的な管理ができるため，依然として現在の多くのオペレーティングシステムでは，フォルダの階層構造によって管理されている．しかしながら，個人が扱うファイル量が爆発的に増加しており，フォルダ階層によって適切に分類することは困難である．その結果，必要なファイルを探し出すことが困難となっている．

そのような背景から，ファイルシステムに対して全文検索を行う，デスクトップサーチシステムが用いられてきた．デスクトップサーチシステムは，全文検索に基づくため，テキストファイル，PDF ファイル，Microsoft Office の文書ファイル，電子メール等の検索に限られており，ファイル名にキーワードを含んでいない画像などのファイルは検索できない．検索したいファイルが全文検索の対象ファイルであっても，キーワードを含んでいなければ検索できない．

この問題に対し，汎用的な全文検索手法ではなく，ユーザの活動履歴を用いる PIM(Personal Information Management) に特化した手法が提案されてきた [2]．ユーザがいつ，どのファイルを使用したかという情報は，ファイル検索において有用であることが分かっており [3]，ユーザの活動履歴を用いたファイル検索手法がいくつか提案されている．ユーザの活動履歴を利用するファイル検索システム [4, 5] では，ユーザがコンピュータを使用した際の使用履歴を記録し，この履歴を検索に利用している．また，時間軸に着目して検索を行うことで, ファイルシステムの階層構造を解決したファイル検索システムも存在する [6–8].

我々もこれまでに，ファイルアクセス履歴からユーザの活動を抽出，分析し，潜在的なファイル間関係を自動的に発見・活用する FRIDAL [9] を提案してきた．FRIDAL は同時に開かれたファイル間に関係を定義しファイル検索に用いる．これによりキーワードによる全文検索結果に加えて，キーワードを含まないファイルを検索対象に含めることが可能になった．しかし，FRIDAL では発見することができないファイル間関係も存在する．そのため本研究では，これまでとは異なったアプローチにより FRIDAL の拡張を試みる．

本研究の目的は，Web 上でどのような情報を閲覧したか，またどのようなキーワードで検索したかなどの Web 利用履歴を活用することで，FRIDAL では発見できなかった関係を発見し，デスクトップサーチシステムにおけるファイル検索精度を向上させることである．Jensen らの研究 [10] でも，ユーザは Web 上の文書から多くの情報を取得していることが確認されており，Web 利用履歴も重要なユーザの活動履歴として利用できることが分かる．

本研究では，作業時に Web ページ上で検索していたキーワードに着目することで，先行研究では考慮されていなかった Web 利用履歴を活用する手法を提案する．ユーザは作業時に参考にしたい情報を発見するために，作業に関係のあるキーワードで Web 検索を行い，目的の情報を発見する．そのため Web 検索時に作業を行っていたファイルと，Web 検索したキーワードとの間に関係があると仮定する．提案手法では，ファイルアクセス履歴と Web 利用履歴を解析することで，ファイル使用情報と Web 検索情報を抽出し，Web 検索キーワードとファイル間の関連度を算出する．

本研究では，提案手法を実現する検索システムを実装した．ユーザの活動履歴を獲得するために，Web 利用履歴を記録する Firefox アドオンと，Samba のアクセスログおよび Windows のアクセス監査ログを解析するツールを実装した．また被験者実験によって，ファイルシステム検索における Web 検索情報による関連の影響の有無と，Web 利用履歴を用いた検索手法の有用性を評価する．被験者実験では，提案手法を実装したファイル検索システム，FRIDAL，及び全文検索システムの検索性能を 11 点平均適合率と上位 20 位の再現率と適合率から評価する．

以下に本論文の構成を述べる．2 章では先行研究を紹介する．3 章では提案手法である Web サイトのアクセス履歴を活用したファイル検索手法について説明を行う．4 章では実際に提案手法を実装したファイル検索システムについて説明し，5 章では被験者実験について説明し，提案手法の有用性について議論す

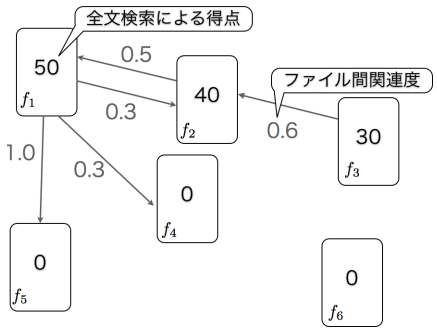

図 1　ファイル間関係と全文検索による得点の例

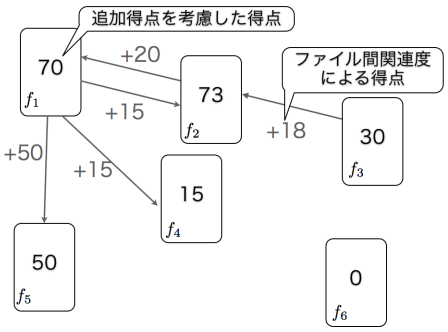

図 2　関連度を考慮した得点計算

る．6 章では関連研究の説明を行い，最後に今後の課題について述べる．

2. 先行研究: FRIDAL

我々はこれまでに，ファイルアクセス履歴を解析することで自動的に潜在的なファイル間関係を算出し，それらの関係を利用することでキーワード非含有ファイルをも検索可能にする手法 FRIDAL を提案してきた [9, 11].

2.1 潜在的なファイル間関係の発見

FRIDAL では，ファイルアクセス履歴からファイルのオープンとクローズ情報を取得し，ファイル毎の使用時間情報を抽出する．関連度の高いファイル同士は長期間において，各作業の度に同じタイミングで利用されるものと仮定し，抽出した各ファイルの使用情報の関係を基に潜在的なファイル間の関連度を計算する [9].

文献 [11] での拡張では，ファイルアクセス時の読み込みと書き込みを区別することで，同時に使用されるファイル間の関係の中でもより強い関係のみを抽出する．ある複数のファイルが同時に使用され，その使用中に書き込みが行われた場合に，そのファイル間に関連があると仮定し，書き込みが行われたファイルから他方のファイルへ関連付けを行う．

二つのファイルのファイル使用時間の重なりを “共起” と呼び，共起時間の累計 T，共起間隔の累計 D，ファイル使用開始パターンの類似度 P，書き込み回数 A をそれぞれ求め，それらを線形結合することで関連度を計算する．関連度要素の詳細については，文献 [9, 11] を参照されたい．

2.2 ファイル間関係を利用した検索

FRIDAL では，ファイル間関連度を利用して関連のあるファイルに全文検索による得点を伝播させることで，キーワードを含まないファイルをも検索可能としている．

例えば，図 1 に示すように，検索対象のファイル集合 $\mathbf{U}$ としてファイル $f_1, f_2, f_3, f_4, f_5, f_6$ があり，キーワード k で全文検索を行った結果のファイル集合を $F(k)$ がファイル f_1, f_2, f_3 であったとする．f_1, f_2, f_3 には全文検索による得点 $S_T(f, k)$ が与えられており, その得点を $S_T(f_1, k) = 50, S_T(f_2, k) = 40, S_T(f_3, k) = 30$ とする．

実験の結果から，参照方向と逆向きにも影響が発生することが分かっているため，FRIDAL では，前述の方法で計算された，ファイル f_i からファイル f_j へのファイル間関連度を $R_{in}(f_i, f_j)$ 用いて，正規化されたファイル間関連度 $\bar{R}$ を以下のように計算する．ω を用いて逆向きの関係も考慮している点に注意されたい．

$$\bar{R}(f_i, f_j, k) = \frac{R(f_i, f_j, k) + \omega \cdot R(f_j, f_i, k)}{\max_{f \in F(k), f' \in U}\{R(f, f', k) + \omega \cdot R(f', f, k)\}}$$

図 1 の例では，正規化されたファイル間関連度 $\bar{R}$ がそれぞれ, $\bar{R}(f_1, f_2, k) = 0.3, \bar{R}(f_1, f_4, k) = 0.3, \bar{R}(f_1, f_5, k) = 1.0, \bar{R}(f_2, f_1, k) = 0.5, \bar{R}(f_3, f_2, k) = 0.6$ であるとする．文献 [11] での拡張の結果，関係の向きを考慮するため，ファイル間には f_1 から f_2 と，f_2 から f_1 のように，関連度の異なる逆方向の関係が張られる場合がある．

全文検索におけるキーワードに対する各ファイルの得点付けおよび，検索キーワードに対する各ファイルの得点付けを行う．

ファイル f_i がファイル f_j との関連度により加点される点数 $S_R(f_i, f_j, k)$ は以下のように求める．

$$S_R(f_i, f_j, k) = \bar{R}(f_j, f_i, k) \cdot S_T(f_j, k)$$

これを用いて，f_i の得点 $S(f_i, k)$ を, 以下のように求める．

$$S(f_i, k) = S_T(f_i, k) + \sum_{f \in F(k)} S_R(f_i, f, k)$$

以上のように新たにファイルに与えられた得点を計算し，最終結果として提示する．

得点計算の結果を図 2 に示す．例えば，f_1 の得点は，f_2 との関連による追加得点が発生し，$S_R(f_1, f_2) = \bar{R}(f_2, f_1, k) \cdot S_T(f_2, k) = 0.5 \cdot 40 = 20$ である．よって, $S(f_1, k) = 50 + 20 = 70$ となる．またもともと全文検索では検索されなかった f_4 や f_5 にも点数が付与され，検索結果に含まれることになる．

複数キーワードの場合は, 各キーワードによる得点の和とする．複数キーワード集合を K とすると, f_i の得点は以下のようになる．

$$S(f_i, K) = \sum_{k \in K} S(f_i, k)$$

先行研究では，以上の実装を行い評価実験を行っている．評価実験の結果，既存研究や全文検索結果を含むフォルダの全ファイルを探索するなどの想定される人間の検索活動とくらべ,

精度が高いことを確認している [9]. また，参照ファイルにキーワードを含む場合と，被参照ファイルにキーワードを含む場合では影響が異なること，参照が行われない関係を除去することで多くの不要な関係を除去することができ検索精度と効率の双方が向上することを確認している [11].

3. 提 案 手 法

3.1 概　　要

本研究では，先行研究で利用したファイルアクセス履歴だけでなく，ファイル使用時の Web 利用履歴を考慮することで，先行研究では得られなかった新たな関係を発見する．Web 利用履歴として，閲覧した Web サイトの URL とタイトルおよび，閲覧時間，そのタブが最前面であった時間を利用する．

ユーザは作業する際．Web 上の情報を参考にする．作業に必要な Web ページへたどり着くために，作業と関係あるキーワードで Web 上で検索することが多い．そのため Web 上で検索したキーワードと，その検索時に開いていたファイルとの間に関連があると仮定して，ファイルと Web 検索キーワードの関連付けを行う．

提案手法では，URL から検索サイトかどうかを判断し，URL およびタイトルから Web 検索キーワードを抽出する．これらの情報とファイルアクセス履歴とを合わせて解析することで，Web 検索キーワードとファイルの関連度を算出する．そして先行研究で用いたファイル間関連度と今回新たに導入するファイルと Web 検索キーワード間関連度を用いて各ファイルの得点付けを行う．

3.2 ファイルと Web 検索キーワード間関連度の算出

ファイルと Web 検索キーワード間関連度を求める要素として，キーワードの検索回数 C_k，キーワードの検索時間の累計 T_k，ファイルにおける検索キーワードの重要度 M_k，検索時間内でのファイル書き込み回数および使用パターンの類似度 P_k を用いる．

3.2.1 関連度要素

各関連度要素の説明を行う．あるキーワードの各検索時間を $\{t_1, t_2, \ldots, t_n\}$ とし，各検索開始時刻からファイル書き込み時間までの時間を $\{s_1, s_2, \ldots, s_n\}$ とし，各検索時間内のファイル書き込み回数を $\{w_1, w_2, \ldots, w_n\}$ とし，ファイルオープンと検索開始，ファイルクローズと検索終了の時刻の差を $\{p_1, p_2, \ldots, p_{2n}\}$ する．

a） **キーワード検索回数 C_k**

あるファイルを開いている際に，同じキーワードでの検索回数が多いほどキーワードとの関連度が大きいとし，C_k を以下のように定義する．

$$C_k = n$$

b） **キーワードの検索時間の累計 T_k**

ある検索キーワードで検索している時間が長いほど，キーワードとの関連度が大きいとし，T_k を以下のように定義する．対象 URL の Web 検索ページが最前面のタブであった時間の累計を検索時間とする．この最前面とはブラウザ上のタブの中で

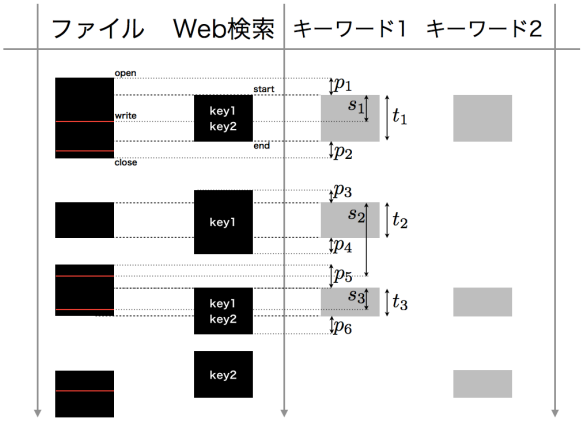

図 3　関連度の計算に用いる

のことである，

$$T_k = \sum_{j=1}^{n} t_j$$

c） **キーワードの重要度 M_k**

Web 検索を行い，何か Web サイトから情報を得た際にファイルへの書き込みを行う可能性が高いと判断した．そのため検索開始時刻からその検索開始時刻以降のファイルへの書き込みまでの時間が短いほど関連度が大きいとし，M_k を以下のように定義する．

$$M_k = \begin{cases} 1 & n = 0 \\ (\sum_{i=1}^{n} m_i)^{-1} & o.w. \end{cases}$$

d） **検索時間内でのファイル書き込み回数 W_k**

検索時間内でのファイルへの書き込み回数が多いほど，キーワードとの関連度が大きいとし，W_k を以下のように定義する．

$$W_k = \sum_{j=1}^{n} w_j$$

e） **ファイル使用パターンの類似度 P_k**

ファイルのオープンと検索開始，ファイルのクローズと検索終了，それぞれの時間の差が小さいほど関連度が大きいとし，P_k を以下のように定義する．

$$P_k = \begin{cases} 1 & n = 0 \\ (\sum_{i=1}^{n} p_i)^{-1} & o.w. \end{cases}$$

3.2.2 ファイル Web 検索キーワード間関連度

提案するファイルと Web 検索キーワード間関連度の算出方法を説明する．Web 検索キーワードと検索時に開いていたファイルの間には関係があるとし，Web 検索キーワードからファイルに対して重みを付けた関連度を求める．この考えに基づき，ファイルと Web 検索キーワード間関連度を以下のように定義する．Web 検索キーワード k からファイル f への関連度を R_k とし．キーワード k で全文検索を行った結果のファイル集合を $F(k)$ とする．まずは以下のように関連度要素を最大値 1 に正規

化する．ただし N_k は実際には要素 T_k, C_k, M_k, W_k, P_k のいずれかである．

$$\bar{N}_k(f_i,k) = \frac{N_k(f_i,k)}{max_{f\in F(k)}N_k(f,k)}$$

さらに，定数 $\alpha,\beta,\gamma,\delta,\epsilon$ を用いて，ファイルと Web 検索キーワード間関連度 R_k を以下のように定義する．

$$R(f_i,k) = \alpha\cdot\bar{T}_k(f_i,k) + \beta\cdot\bar{C}_k(f_i,k) + \gamma\cdot\bar{M}_k(f_i,k) + \delta\cdot\bar{W}_k(f_i,k) + \epsilon\cdot\bar{P}_k(f_i,k)$$

3.3 得点方法

キーワード k で全文検索を行った結果のファイル集合を $F(k)$ とする．まず，R_k の最大値が 1 となるように正規化を行う．

$$\bar{R}_k(f_i,k) = \frac{R_k(f_i,k)}{\max_{f\in F(k)} R_k(f,k)}$$

また全文検索結果上位 N 位までの TF-IDF 法による得点の平均点を S_N とし，これを Web 検索キーワード k が持つ基本点とし，ファイルとの関連度によりファイルに点数を伝播させる．ファイル f_i の追加得点 $S_k(f_i,k)$ を，以下のように求める．

$$S_k(f_i,k) = \bar{R}_k(f_i,k)\cdot S_N$$

前章のファイル間関連度を用いた得点を $S_R(fi,k)$ と，ファイルと Web 検索キーワード間関連度を用いて各ファイルに与える付加得点を $S_k(f_i,k)$ を用いて，最終的な点数を計算する．

$$S(f_i,k) = S_R(f_i,k) + S_k(f_i,k)$$

以上の計算に従った得点 $S(f_i,k)$ をファイルに付与することで新たな最終結果を提示する．

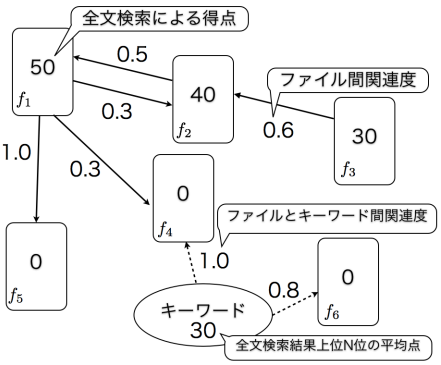

図 4　全文検索による得点

4. 実　　装

4.1 実装ツールの構成

本研究では，ファイルアクセス履歴として Samba のログファイルまたは Windows イベントログを，また Web 利用履歴として自作した Firefox アドオンの出力を解析し，それぞれの関連度を関連度 DB に格納する機能と，関連度 DB と全文検索エンジンを用いて，提案手法によってファイルを検索する機能を有するアプリケーションとして実装した．Samba のログの場合，FRIDAL では標準機能であるログ出力機能を利用しているが，Samba では vfs の追加機能である audit が利用可能であるため，audit のうち最も詳細な監査が可能な vfs_full_audit を使用する．Windows イベントログの場合は，イベントログのうちセキュリティログを使用する．実際には Window セキュリティログファイルはバイナリファイルであるため，LogParser 2.2 を用いて利用可能な形のデータとして取り出す必要がある．実装には Java 言語を使用し，関連度を格納する DBMS には MySQL 5.1.60 を，全文検索エンジンには Hyper Estraier 1.4.13 を用いた．

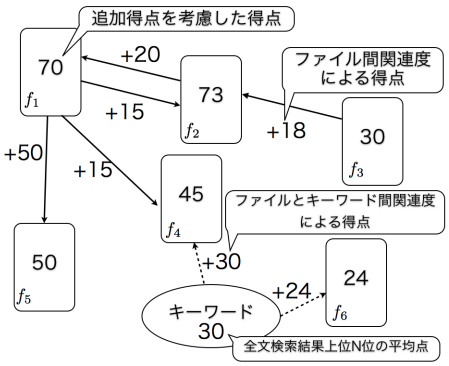

図 5　二つの関連度を考慮した得点

4.2 実装ツールの動作

準備段階の動作　準備段階は図 6 において破線で記載されている．ファイルアクセス履歴と Web 利用履歴の解析を行う．解析では，それぞれのログから使用情報を抽出し，関連度要素等の情報を関連度 DB に格納する．また事前に Hyper Estraier を用いてファイルシステムの全文インデックスを作成しておく．

検索段階の動作　検索段階は図 6 において実線で記載されている．検索プログラムにおいて，ユーザ名とキーワードを入力する．検索の際，オプションとして，検索結果をフィルタリングする拡張子と関連度算出の各種パラメータを指定できる．検索キーワードを用いて全文検索を行い，検索結果と関連のあるファイルとファイル間関連度要素を関連度 DB から取り出し，算出したファイル間関連度を用いて，追加得点の計算を行う．次に入力したキーワードと関係のあるファイルとファイルと Web 検索キーワード間関連度要素を取り出し，算出したファイルと Web 検索キーワード間関連度を用いて，追加得点を計算する．最後に得点の高いものから順にユーザに提示する．

5. 評 価 実 験

5.1 概　　要

提案手法の有用性を評価するために，以下の 3 つの観点でそれぞれ実験を行った．

- Web 検索情報はファイル検索に影響を与えるか
- 提案手法は，検索精度を向上させるか
- Web 検索キーワードの基本点の算出方法は適切か

以降，ユーザがあるファイルを検索したいと考え，入力するキーワードを “表現キーワード” と定義する．

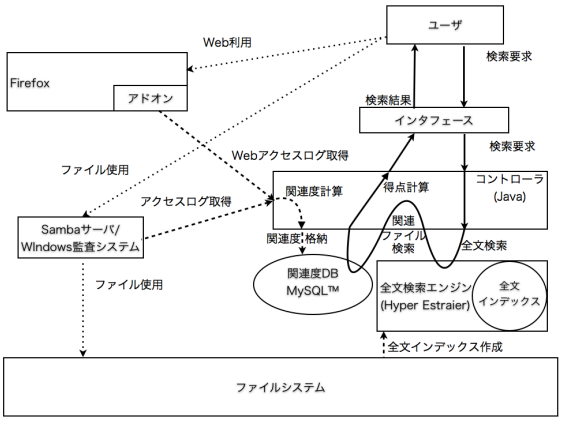

図 6　実装の構成と動作

5.2 実験環境

本研究の実験では，二名の被験者 A，B のアクセスログを採取した．各被験者のアクセスログの情報を表 1 に示す．またユーザに表現キーワードで表現されるファイルをファイルシステム内の全てのファイルから選んでもらい，それを正解セットとする．用いた表現キーワードと正解セット数を表 2 に示す．

表 1　各被験者のアクセスログ情報

	ログ採取日数	ファイルログ行数	OS	Web 検索キーワード総種類数	のべ Web 検索回数
被験者 A	65	4,252,431	OS X Lion	102	302
被験者 B	119	9,262,085	Windows 7	298	940

表 2　使用する検索キーワードと正解ファイルの情報

	キーワード	正解ファイルの概要	正解ファイル数
Q_1	HyperEstraier	被験者 A の卒業研究に関するファイル	108
Q_2	MATLAB	被験者 B の卒業研究に関するファイル	93
Q_3	simulink	被験者 B の卒業研究に関するファイル	91
Q_4	model-driven development	被験者 B が勉強会で使用したファイル	19

Hyper Estraier は xdoc2txt フィルタを使用して Office 文書及び pdf をインデックス作成の対象に加える．また抽出するファイルの拡張子は`.bib`，`.doc`，`.docx`，`.gif`，`.htm`，`.html`，`.jpg`，`.jpeg`，`.mpg`，`.mpeg`，`.ppt`，`.pptx`，`.pdf`，`.png`，`.txt`，`.tex`，`.xls`，`.xlsx`，`.c`，`.cpp`，`.cs`，`.java`，`.key` に限定した．また対象とする Web サイトは `Google`，`Yahoo!`，`Bing` に限定した．

本研究では，Web 検索キーワードによる関連の影響を考慮するために，Web 利用履歴から抽出した Web 上で検索したキーワードの中から表現キーワードを選んでいる．Q_1 は被験者 A が卒業研究で開発したシステムに使用した全文検索ソフトの名前である．また Q_2 と Q_3 はどちらも被験者 B の卒業研究に関するツールの名前であるが，Q_2 は主に実験を行う前の作業に関するものであり，Q_3 は実験を行う際やそれ以降の作業に関するものである．また Q_4 は被験者 B がある勉強会で使用した本に登場する単語である．本研究では，この 4 つの表現キーワードの傾向が異なると考え，実験で使用することを妥当であると判断し，以降この 4 つの表現キーワードを用いて実験を行う．

検索性能の評価実験では，適合率と再現率を求め．F 尺度と 11 点平均適合率を用いて，各検索システムの性能比較を行う．

5.3 評価実験 1：Web 検索情報がファイル検索に与える影響を評価

5.3.1 実験方法

実際に各キーワードに対して Web 検索キーワードによる関連があるか，正解ファイルであるか，FRIDAL による関連があるかを調べた．

5.3.2 実験結果

Web 検索キーワードによる関連，FRIDAL による関連，及びどちらかに関連がある場合における正解ファイルの適合率を表 3，再現率を表 4 に示す．

表 3　適　合　率

		Q_1	Q_2	Q_3	Q_4
1	Web 関連のあるファイルの適合率	48/54 (0.89)	7/17 (0.41)	13/29 (0.45)	3/3 (1.00)
2	FRIDAL 関連のあるファイルの適合率	49/59 (0.83)	27/108 (0.250)	25/84 (0.298)	4/7 (0.57)
3	どちらかに関連があるファイルの適合率	62/73 (0.84)	29/113 (0.257)	28/92 (0.304)	7/10 (0.70)
4	Web 関連のみのファイルの適合率	13/14 (0.93)	2/5 (0.4)	3/8 (0.38)	3/3 (1.00)

表 4　再　現　率

		Q_1	Q_2	Q_3	Q_4
5	Web 関連のあるファイルの再現率	48/108 (0.44)	7/93 (0.075)	13/91 (0.14)	3/19 (0.16)
6	FRIDAL 関連のあるファイルの再現率	49/108 (0.45)	27/93 (0.29)	25/91 (0.27)	4/19 (0.21)
7	どちらかに関連があるファイルの再現率	62/108 (0.57)	29/93 (0.31)	28/91 (0.32)	7/19 (0.37)

5.3.3 考　　察

表 3 の 1 と表 4 の 5 より Web 関連のみでも正解ファイルを発見できていることからファイル検索に影響を与える可能性が高いと考えられる．また表 3 の 4 より Web 関連のみの場合に，キーワードによって適合率に差はあるが，FRIDAL では発見できない正解ファイルをある程度発見することができている．表 3 の 1 と 2，表 4 の 5 と 6 を用いて，Web 関連のみと FRIDAL 関連のみを比較すると，Web 関連のみの方が適合率が高いため，Web 関連は FRIDAL 関連より発見したファイルが正解である可能性が高いことが言える．しかし Web 関連は再現率は低いため，Web 関連のみでは FRIDAL 関連ほどファイルを発見できないことが分かる．そこで，FRIDAL 関連と Web 関連の二つ

の関連を組み合わせて考慮すると，表 3 の 2 と 3，表 4 の 6 と 7 から実際に FRIDAL 関連のみより適合率と再現率は上がっていることが確認できる．そのため，FRIDAL による検索に Web 検索キーワードによる関連を考慮することで，検索精度が向上できるのではないかと考えられる．

5.4 評価実験 2：検索性能の評価

5.4.1 実験方法

提案手法，FRIDAL，Hyper Estraier それぞれで被験者二人の 4 つの表現キーワードを用いてファイル検索を行い，11 点平均適合率を用いて評価を行った．FRIDAL の各パラメータは最も良い結果が出たとされる $(\alpha,\beta,\gamma,\sigma) = (1.0, 0.5, 0.5, 1.0)$ を用いた．提案手法の各パラメータは事前に行った予備実験を行って求めた最適値 $(\alpha,\beta,\gamma,\sigma,\epsilon) = (0.5, 1.0, 1.0, 0.5, 0.5)$ を、また表現キーワードによる全文検索結果上位 N 位の平均点には $N = 5$ を用いた．

5.4.2 実験結果

被験者二人の 4 つのキーワードに対する検索結果の 11 点平均適合率の平均を図 7 に示す．また検索結果上位 20 件における再現率と適合率及び F 尺度の値の平均を表 5 に示す．

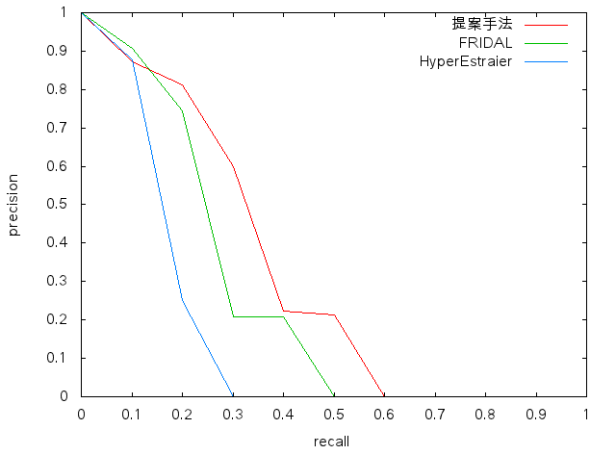

図 7　11 点平均適合率

表 5　検索結果上位 20 件における再現率, 適合率及び F 尺度の平均

	再現率	適合率	F 尺度
提案手法	**0.2155**	**0.6875**	**0.2943**
FRIDAL	0.1637	0.5875	0.2352
HyperEstraier	0.1586	0.5625	0.2267

5.4.3 考察

表 5 より他の手法と比べ，再現率，適合率ともに高い値を示していることから他の手法と比べ提案手法が検索結果上位 20 件において正解ファイルを多く含んでいることが分かる．これより Web 検索情報と作業ファイルの間には関係があると言える．これらから実験 1 の想定通り，Web 検索情報によって実際に検索精度を向上させることができたと言える．

図 7 より，再現率 10%までは提案手法は他の手法と比べて，適合率が低い．このことから提案手法は他の手法と比べ検索結果の上位に不正解ファイルが位置していることが分かる．これは Q_2 において，Web 検索情報の利用でいくつかの不正解ファイルが上位に表示されてしまったからである．検索結果から，ほとんどが新たに不正解ファイルが上位に表示されたのではなく，元々FRIDAL でもファイル間関連によって上位に表れていたが，提案手法の Web 検索キーワードとの関連によってさらに上位に表示されてしまっていたことが分かった．実際にこの不正解ファイルを確認すると eclipse のプラグイン内のファイルであった．Q_2 は卒業研究に関するファイルであり，主に卒業研究の実験を行う前の作業に関するものである．卒業研究の実装に eclipse を使用しているため，eclipse の起動時に被験者 B の意図しないアクセスがプラグイン内のファイルに発生し，ファイル間関連や Web 検索キーワードとの関連が生まれ、点数が高くなったことで上位に表示されたと考えられる．そのため，提案手法における低い再現率での適合率の差につながっていると考えられる．

また図 7 より，提案手法は他の手法と比べると再現率が高いため，正解ファイルを他の手法に比べて多く発見できていることが分かる．しかし，Web 検索キーワードによる関連のみの正解ファイルは提示される順位が低い傾向にあった．

5.5 評価実験 3：N の値の変化によってファイル検索に与える影響を評価

5.5.1 実験方法

本研究では，Web 検索キーワード k の持つ基本点を，キーワード k による全文検索結果上位 N 位の得点の平均と設定している．本実験では，N の値を変化させることで，検索結果に影響を与えるかどうか検索結果上位 20 件の再現率，適合率及び F 尺度から評価を行った．各パラメータ実験 2 と同様に $(\alpha,\beta,\gamma,\sigma,\epsilon) = (0.5, 1.0, 1.0, 0.5, 0.5)$ を用いた．

5.5.2 実験結果

N の値を 1，5，10，20 と変化させた被験者二人の 4 つのキーワードに対する検索結果上位 20 件における再現率と適合率及び F 尺度を表 6 に示す．

表 6　検索結果上位 20 件における再現率，適合率及び F 尺度の平均

	再現率	適合率	F 尺度
N=1	0.2155	0.6875	0.2943
N=5	0.2155	0.6875	0.2943
N=10	0.2182	0.7	0.2987
N=20	0.2182	0.7	0.2987
N=30	0.2182	0.7	0.2987
N=80	0.2131	0.6750	0.2903

5.5.3 考察

表 6 より，$N = 80$ のように N の値を大きくしすぎると，Web 検索キーワードによる関連が小さくなり再現率，適合率，F 尺度が少し落ちていることが分かる．このまま N を大きくし続けると Web 検索キーワードによる関連が限りなく小さくなり，FRIDAL 関連のみと同じファイル検索結果となる．

N を 5 以下に小さくすると，Web 検索キーワードによる関連が大きくなり，再現率，適合率，F 尺度が落ちていることが分かる．実際に Q_2，Q_3 においては Web 検索キーワードによる関

連によって新たに発見できるファイルと不正解ファイルの数にあまり差が見られなかったため，N を小さくすることでノイズが増えたと考えられる．

また N を最大値である 1 としても $N = 5$ と結果が変わらなかった．これより全文検索結果による最大得点を用いても，FRIDAL による検索に大きな影響を与えるほど基本点が高くないことが分かる．実際に Web 検索キーワードによる追加得点は，ファイル検索したキーワード全てと関係があっても検索キーワード数の分しか得点が加算されない．それに比べ，FRIDAL による関連はファイル間を伝播するため，複数のファイルと関係があった場合，その関係のあるファイル数の分だけ得点が追加されるため，Web 検索キーワードによる追加得点よりも高い追加得点となることが多いのではないかと予想できる，そのため，本研究で設定した基本点では Web 検索キーワードによる関連を適切に制御することができないことがわかった．基本点を FRIDAL による得点と大きな差が生まれないような値に設定することによって，Web 検索キーワードによる関連のみのファイルの順位を上げることができるが，Q_2，Q_3 のような Web 検索キーワードによる関連のみによる適合率が低いキーワードの場合は，同時にノイズが増えることに繋がってしまう．そのため長期間ログを採取し，少ないキーワード検索回数と関連しているファイルを取り除くなど，ノイズを減らす方法を考えることが重要であると考えられる．

6. 関 連 研 究

Soules 氏らの Connections [12] では read() や write() 等のファイル操作に対するシステムコールを取得し，その情報を基にファイルをノード，関連度をエッジとする有向グラフを作成する．検索を行う際は，まず従来の全文検索を行い，その結果のファイル群と関連の高いファイルを検索結果として提示する．目的とアプローチは本研究と同じだが，本研究ではファイルアクセスログのオープン，クローズ情報を用いて算出したユーザのファイル使用情報と，書き込み情報から算出した参照関係を利用しているのに対し，Connections ではシステムコールの read() と write() を用いてファイル間の参照関係を算出しようとしており，ファイルの使用時間等の情報を用いていないという違いがある．また，検索結果の各ファイルに対する得点の計算方法も大きく異なる．

Chirita らの研究 [13] では，open()，create() 等のファイル操作に対するシステムコールに加え，ファイル名等の属性情報を用いて，ファイル間にリンク構造を作成し，ベクトル空間モデルに基づくファイル検索を行う．システムコールから抽出したファイル操作履歴を用いて，一定時間内にアクセスが行われたファイルに対し関連を付けており，本研究のファイル関係の算出方法とは異なる．

Aharony らの研究 [14] では, 検索を行った際に，ユーザが選択したファイルに対してファイル属性としてクエリ情報を付加することで，過去の検索結果による学習を行い，検索性能を向上させている．

森田らは OS のイベントログと，ブラウザなどのアプリケーションに組み込んだプラグインから，ユーザの行動履歴を抽出し，それを基に Web 閲覧時の記憶を想起させるツール Memory-Retreiver [4] を提案している．

Chen らは 連想メモリに基づきファイル間関係を活用する検索システム iMecho を提案している [15]．提案手法では，活動履歴から，“jump_to”, “save_as”, “same_task” といったファイル間関係を抽出している．Chau らも，E-mail やオフィス文書の間に関連を定義し，関連グラフを探索することで，必要な情報の検索を支援する Feldspar [16] を提案している．

暦本らによる Time-Machine Computing [7] は，ファイルの階層構造を用いず，ファイルはデスクトップ上に配置され, 時間と共に消えていくシステムである．検索には時間を用い, その日時におけるデスクトップの状態を再現することができる．

大澤らの俺デスク [5] は，OS とアプリケーションのイベントから抽出したユーザ行動履歴を基にデータ間関連度とデータ着目度を算出し，関連データの検索やタイムラインビューアによるユーザの情報想起を目的とするツールである．関連データ検索ではキーワードによる検索ではなく，指定したファイル又は Web ページと関連のあるデータを提示する．また，関連データ検索に用いる関連度の計算は，参照時刻のみを利用しており，本研究とは異なる．

7. おわりに

7.1 ま と め

従来の検索システムでは，キーワードによる検索要求に対して，キーワードと関連があっても，キーワードをファイルの内容に含んでいるファイルでなければ検索することができなかった．この問題に対し，先行研究である FRIDAL はファイルアクセス履歴から抽出したファイル間関連度を用いることで，従来の方法では検索できなかった検索キーワードを含まないが，ファイル間の関連があるファイルを発見できるようになった．しかしファイル間の関連のみでは発見できないファイルも存在した．

本研究では，先行研究では考慮されていない Web 利用情報の中でも Web ページ上で検索していたキーワードに着目することで，作業時の Web 利用履歴を活用する手法を提案した．また実際に Firefox プラグインを実装して，Web 利用情報を取得し，Samba や Windows のファイルアクセスログとともに解析することで，提案手法を実現する検索システムを実装した．また被験者実験により，Web 利用履歴を活用することの検索システムへの影響を明らかにし，提案手法の有用性を評価した．実験の結果より Web 検索キーワードによる関連はファイルシステム検索へ影響を与えることが予測でき，実際に FRIDAL や全文検索エンジンと比較して，検索結果上位において再現率と適合率がともに改善されたことを確認した．また先行研究では考慮されていなかった新たな関係に着目することで，従来の手法では発見できなかったファイルも一部発見することができた．

7.2 今後の課題

ファイルと Web 検索キーワード間関連度要素の見直し

Web 検索情報を考慮することで，新たな正解ファイルを発見

できたが，そのファイルの順位が低い場合が多かった．そのため，今回設定した関連度要素より，参考にしたWebページとの関連の強さをより反映したものへ改善する必要がある．

実験規模の拡大

今回の評価実験では，学生を対象として行ったため，特定の検索に対する評価のみ実施した．そのため，本研究よりも大人数かつ様々なユーザを対象とした評価実験を行う必要があると言える．

他のWeb利用情報の活用

本研究では，Web利用情報としてWeb検索情報のみを利用したが，他のWeb利用情報がデスクトップサーチシステムに活用できないか今後さらに考えていきたい．

謝　　辞

本研究の一部は科研費 (20300009, 22240005) の助成による．本研究に関して有益な助言を頂きました，東京工業大学の横田治夫教授と名古屋大学の山下訓昭氏に感謝致します．

文　　献

[1] Scott Fertig, Eric Freeman, and David Gelernter. ”finding and reminding” reconsidered,. *ACM SIGCHI Bulletin*, Vol. 28, No. 1, pp. 66–69, 1996.

[2] Edward Cutrell, Susan T. Dumais, and Jaime Teevan. Searching to eliminate personal information management. *CACM*, Vol. 49, No. 1, pp. 58–64, 2006.

[3] Tristan Blanc-Brude and Dominique L. Scapin. What do people recall about their documents?: Implications for desktop search tools. In *Proc. Intl' Conf. on Intelligent User Interfaces(IUI2007)*, pp. 102–111, 2007.

[4] 森田哲之, 日高哲雄, 田中明通, 加藤泰久. 記憶想起支援ツール『memory-retriever』. In *IPSJ Transactions on Database*, 第 48 巻, pp. 1197–1208, 2007.

[5] Ryo Ohsawa, Kazunori Takashio, and Hideyuki Tokuda. Oredesk: A tool for retrieving data history based on user operations. In *Proc. Eighth IEEE International Symposium on Multimedia (ISM'06)*, pp. 762–765, 2006.

[6] Susan Dumais, Edward Cutrell, J. J. Cadiz, Gavin Jancke, Raman Sarin, and Daniel C. Robbins. Stuff i've seen: A system for personal information retrieval and re-use. In *Proc. SIGIR2003*, pp. 72–79, 2003.

[7] Jun Rekimoto. Timemachine computing: A timecentric approach for the information environment. In *Proc. ACM UIST'99*, 1999.

[8] Yasuko Matsubara and Ichiro Kobayashi. Development of a desktop search system using correlation between user's schedule and data in a computer. In *Proc. WI-IATW'07*, pp. 235–238, 2007.

[9] Tetsutaro Watanabe, Takashi Kobayashi, and Haruo Yokota. Fridal: A desktop search system based on latent interfile relationships. In *Proc. ICSOFT 2010 (Springer CCIS 170)*, pp. 220–234, 2012.

[10] Carlos Jensen, Heather Lonsdale, Eleanor Wynn, Jill Cao, Michael Slater, and Thomas G. Dietterich. The life and times of files and information: A study of desktop provenance. In *Proc. CHI2010*, 2010.

[11] 山下訓昭, 小林隆志, 横田治夫, 阿草清滋. ファイルアクセス履歴から抽出した参照関係に基づくファイル検索手法. 第 4 回 Web とデータベースに関するフォーラム (WebDB Forum 2011) 論文集, 2011.

[12] Craig A.N. Soules and Gregory R. Ganger:. Connections: Using context to enhance file search,. In *Proc. ACM Symposium on Operating Systems Principles*, pp. 119–132, 2005.

[13] Paul A. Chirita and Wolfgang Nejdl. Analyzing user behavior to rank desktop items. In *Proc. Intl' Symp. on String Processing and Information Retrieval(SPIRE)*, pp. 86–97, 2006.

[14] Sara Cohen, Carmel Domshlak, and Naama Zwerdling:. On ranking techniques for desktop search. *ACM Transactions on Information Systems*, Vol. 26, , 2008.

[15] Jidong Chen, Hang Guo, Wentao Wu, and Wei Wang. iMecho: An associative memory based desktop search system. In *Proc. CIKM2009*, pp. 731–740, 2009.

[16] Duen Horng Chau, Brad Myers, Andrew Faulring, and Human Computer. What to do when search fails: Finding information by association. In *Proc. CHI2008*, 2008.