

PCにおけるユーザ操作を可視化する タスク想起支援手法の提案

松本光弘^{†1} 岡野真一^{†2} 森田哲郎^{†2}
沼尾正行^{†3} 栗原 聡^{†3,†4}

近年、個人レベルで大容量の記憶媒体が容易に使えるようになりつつあり、PC上には多くのファイルが蓄積されていく一方、有効に利用されていない。しかし、多くのタスクがPC上で行われる状況では、タスクに依存してファイル間に依存関係があったり、過去のファイルが現在のタスクにおいて重要である場合があったりするなど、すべてのファイルにはそのファイルに関するコンテキストが存在するはずである。そこで、ファイルに対する一連の操作などの時系列履歴やファイルとWebページ間の類似関係をネットワークとして抽出・可視化してユーザに提示することが、ユーザにとってそのときのコンテキストを思い出し、検索対象のファイルなどを従来の検索手法に比べて容易に見つけることが可能となることを示す。

Visualization Method of User's File Operation for Recalling Task on Desktop Environment

MITSUHIRO MATSUMOTO,^{†1} SHINICHI OKANO,^{†2}
TETSUO MORITA,^{†2} MASAYUKI NUMAO^{†3}
and SATOSHI KURIHARA^{†3,†4}

Users can use the huge storage of personal computers at a moderate price in late years, and they tend to store a large number of files on their computers but cannot make efficient use of them. On the other hand, when a task is carried out on the computer, some files related to the task may be generated or some old files may be important for the task. Therefore, all files must be associated with a task and some contexts corresponding to a file must exist. Hence, we will show that visualization of "file event network", which is made by both file access or web access history and analogous relationship with files or web pages, allows to recall the contexts according to a file and to retrieve the desired file or web page easier than the traditional approach.

1. はじめに

インターネットの発展とともにクラウドコンピューティングなどの登場で、もはやパソコン(PC)は単体での存在であると同時にネットワーク端末としての存在でもあり、より多様性のある使い方がされるようになった。仕事からプライベートに至る多種多様なタスク、たとえば報告書の作成から日記の執筆、家計簿の管理に至る様々なタスクがPC上で行われるようになり、タスクの処理に応じて生成されるファイルの種類も多様化しつつある。

一方、近年のPCの高性能化と、HDD、SSDなどの二次記憶装置の大容量化・低価格化により、個人レベルで大容量の記憶媒体が容易に使えるようになった。記憶容量が少ないときは必要なファイル以外は消去するしかなかったが、記憶容量が十分な状況となれば無理に消去する必要はなく、結果としてPC上には様々なファイルがひたすら蓄積されていく傾向にある。

通常我々は、あるタスクを処理している間はそのタスクに関するコンテキスト、すなわち、作成中のファイルがどのファイルを参照しているとか、どのファイルのどの部分を編集したなど、そのコンテキストに関わる自分の行動履歴を把握できている。よって、バックアップのために類似したファイル名でファイルを保存したり、一時的なフォルダを作成し、そこに重要なファイルを保存していても迷うことはない。

しかし、タスクが終了して時間が経過してしまえば、ほとんどの場合、そのタスクに関するコンテキストは忘れられてしまう。たとえば、我々はいきなり過去に生成されたファイルを見ても、そのファイルがいつどのタスクにおいて生成され、その生成過程においてどのファイルを参照したとか、どのファイルを編集したとかなどをなかなか思い出すことはできない。しかも、我々がPCで行う作業を見つめ直してみると、新規にゼロから文書の作成を開始し、しかも完成するまでいっさい他のファイルなどを参照しないような作業は希であ

†1 大阪大学大学院情報科学研究科

Graduate School of Information Science and Technology, Osaka University

†2 住友電気工業株式会社情報通信研究所

Information and Communications Labs., Sumitomo Electric Industries, Ltd.

†3 大阪大学産業科学研究所

The Institute of Scientific and Industrial Research, Osaka University

†4 独立行政法人科学技術振興機構

Japan Science and Technology Agency CREST

り、通常は、過去に作成したいくつかの文書やウェブページを参照したり、過去の文書を部分的に更新したりと、多くのタスクは過去のタスクに何かしら関連している。

よって、我々が新しいタスクへの取り組みを開始するときには、何らかの方法で過去のタスクを参照する必要性が発生するのであるが、現在我々ができる手段はファイル名やキーワードに基づくファイル単位の検索であり、基本的なしくみはインターネット検索と同じである。しかしながら、キーワード検索だけでは十分な検索を行うことができないことが知られている¹⁾。文献1)では、メールやローカルファイルやWeb検索において、人々がどのような検索を行うか検証しており、特定の情報においては42%、一般の情報においては55%がキーワードによって検索されたものであり、残りは別の手段を用いて検索されている。たとえば、ある教授室の電話番号を調べる際には、「教授名、電話番号」というキーワードで調べるのではなく、学科のページから該当教授のページに行き、そこから電話番号を得る。これは、キーワードを使って目的のページに飛ぶのではなく、過去に行ったプロセスを思い返しながらか、小さなステップを経て目的の情報にたどり着いた1つの例であり、文献1)では小さなステップを経ることで徐々に情報を思い出していく過程が検索において重要であることを示している。また、ファイル名での検索は、同一または類似するファイル名のファイルが複数検出されることがあり、1つ1つのファイルがそれぞれどのタスクに関連したものであるかまでは特定できず、日付や内容、保存されているディレクトリなどをいちいち確認しなければ所望のファイルを探すことはできない。

仮にタスクどうしが互いに独立した関係で個々のタスクが簡潔なものであれば、このような方法であってもファイルを特定することは比較的容易かもしれない。しかし、実際には、タスク自体も複雑であり、タスクを完了するまでに様々なファイルが生成されていたり、さらに異なるタスク間でファイルを共有していたり、あるタスクに関連するファイルが別のタスクのファイルを参照したりしている場合や、異なるタスクで使用されたにもかかわらず、類似した名前を持つファイルが複数存在していたりと、ファイルのみに着目する検索ではなかなか該当するタスクに関連するファイルを見つけることは難しい。しかも、上述するようにPCに保存されているファイル数は増加しつつある。

この状況において、もしも個々のファイルに関連するコンテキストを思い出すことができれば、個々のファイルとそれに対応するタスクを結びつけることができ、参照したいタスクとそのタスクに関わる目的とするファイルを見つけることが容易になると期待される。ここで、コンテキストについて考えてみると、コンテキストはタスクを処理する際の一連の行為そのものであり、なかでも、ファイルの生成・編集・コピーなど、タスクを処理する過程

でユーザがPCを操作した一連の履歴は、コンテキストにおける主要な要素であり、幸い、現在のPCでは過去のファイルも消去されずに保存される傾向にある。

そこで、本論文では単にファイル名で検索するのではなく、ファイルに対する一連の行為に関する時系列履歴を「ファイルイベントネットワーク」として抽出・可視化してユーザに提示することが、ユーザがそのときのコンテキストを思い出し、参照したいタスクとそのタスクに関連するファイルをより容易に見つけることができることを示す。

以下、2章において、関連研究における本研究の位置づけを述べ、3章で提案するファイルイベントネットワークを作成する方法について述べる。そして、4章では実装したシステムの動作検証と有用性について述べ、5章でまとめと今後の課題を述べる。

2. 関連研究

現在のPCでのファイル検索は、上述するようにファイル名やキーワードに基づく手法が利用されており、Google Desktop^{*1}などのサービスでは、PC内ファイルの各テキスト情報を高速に検索できるようにあらかじめインデックスを作成してファイルアクセスを支援している。ランキングをさらに工夫する手法としては文献2)、3)などが提案されている。また、ファイルに関するコンテキストに基づいてキーワード検索ランキングを決定する研究⁴⁾も存在する。Windows VistaなどのOSにおいてもインデックスを用いた高速な検索機能があらかじめ用意されている。このようなキーワード検索では適切なキーワードをユーザが選択する必要があり、文章や文書中の単語を覚えておかなければ、検索することはできない。さらに、内容が類似するファイルや編集された時刻などに着目した検出は可能であるが、お互いに内容が類似するファイルどうしがどのような経緯で作成されたのか、という作業のコンテキストまでを思い出そうとすることはなかなか困難である。また、検索対象のファイルについて、ユーザが知っているあらゆる情報に対応するため、メタデータやタグ取り付け機能などを用いた、より高度な検索インタフェースの提案もされている⁵⁾。これについても、現状ではメタデータを人手で付与する必要があり、つねに付与されているとは限らず、またファイルの内容によっては的確なメタデータの付与が困難な場合も考えられる。

ファイルのテキスト情報以外の情報も利用することで適切なファイル検索を行う方法としては文献6)があげられる。メールに添付されたファイルであれば、メールの送り主や件名との関係性を利用し、出版物であれば著者や会議名といったメタデータとGoogleやCiteseer

*1 Google Desktop, <http://desktop.google.com/en/>

によるランキングを基に適切なファイルのランキングを求める手法を提案している。また、メタデータにもファイルのコンテキストが含まれていることから、メタデータをオントロジを用いてスコアリングすることによって、精度の高いファイル検索が可能であるという立場をとる研究として、セマンティックデスクトップ⁷⁾があげられる。しかし、上述するように、メタデータについては、人手によるオントロジの構築が必要であるため、ユーザの負担が避けられない。たしかに、セマンティックデスクトップが実現すれば、ミーティングやプロジェクトといったカテゴリ別にファイル検索を行えるため、より効率良くファイル検索を行うことができると考えられるが、メタデータをユーザの負担なくして効率的に付与する仕組みの開発が必要不可欠である。これに対し、本研究ではメタデータを付与することは行わないことからその実現が容易である。また、文献 8) では、人々が何を手がかりに文書ファイルを探すのかを検証しており、ユーザはファイル名やファイルが格納されている場所はもちろんのこと、キーワードや利用時刻・文書全体の見た目といった様々な要因を思い出すことによって、ファイル検索を行っていることが示されており、ファイル検索においてファイルに関係するイベントは重要な手がかりであることが述べられている。

目的は異なるものの、本研究と同じく時間を手がかりとして検索を行う方法も提案されている。メール管理では、受信メールを時刻順に並べて、日時からメールを特定するという方法が有効である。Ringelらは公的、および私的なイベントを時系列インタフェースに表示することで、ユーザがイベントとの相対的な時間感覚を頼りに効率良くメールを特定できることを示した⁹⁾。大澤らは、ユーザのデータ参照時間や回数などから算出した着目度に基づいた時間軸インタフェースを実装している¹⁰⁾。また、アクセス時刻の共起時間を用いて、キーワード検索を拡張した研究もある^{11),12)}。これらの研究は、キーワードによる検索結果と、ファイル間のアクセス時刻に着目し、それぞれの結果を合わせることで検索結果の精度向上を図っている。文献 11) は、同時刻に利用されたファイルどうしを結びつけることによって、本来キーワードを含んでいない画像ファイルやデータファイルを紐付けされた文書ファイル内のキーワードを利用して検索できる手法である。これらは、ファイル検索を目的として提案された手法であるため、ファイル間の関係性を利用しているが、提案法はファイルのコンテキストを詳細に表すために、ファイルイベントの関係性をネットワークとして出力している点異なる。また、ファイル検索とは目的が異なるが、Rekimotoはコンピュータの作業履歴を蓄積し、時間移動によって過去の作業環境の再現を行うとともに、時間にとまなうPC環境の遷移を視覚的に表した¹³⁾。また、個人の経験に基づいたコンテンツ検索の研究として文献 14) があげられる。個人的な経験を経験要素表現に変換し、コンテンツ間で

経験要素がどれほど一致しているのかを用いてコンテンツ間にリンクを結ぶことで、そのリンク構造に基づいたコンテンツ検索結果のランク付けを行っている。これを実現するためには、自動的な経験要素表現の抽出方法が必須である。

PCの利用履歴から個々のタスクを抽出する研究として文献 15) がある。コピーや編集といった履歴を利用して一連の操作を可視化する手法を提案しているが、ファイル間の類似性は利用しておらず、「メールに添付された文書は編集され、再びメールに添付されて送信されることが多い」といった共通性のあるタスクのパターンを抽出することが目的であり、タスクの内容までを考慮したタスクの特定を行うことはできない。これらの研究は、ユーザの振舞いに着目した検索手法を提案している点では本研究と同じ着眼点を持つといえるが、これらが振舞いのみに着目しているのに対して、本研究では、個々のファイルの中身についても考慮する点が大きく異なる。

3. ファイルイベントネットワーク

我々は、PC上におけるユーザのコンテキストどうしの関連性をネットワークとして表示することによって、タスクがどのような手順で実行されたのかを、容易に把握できるようにすることを目的としている。

コンテキストを構成する要素としては、ファイルの生成・編集・参照・コピーから、ファイルの圧縮・解凍・添付・アップロードやマウスの軌跡やメール・ウェブアクセスなど、様々な要素が考えられるが、今回は、それらの中で、コンテキストをより効果的に表現すると考えるファイルの生成・編集・参照・コピー(リネーム)・削除・ウェブ閲覧に着目し、これらに関するユーザがPCを操作した一連の履歴を収集する。むしろ、他の振舞い情報を利用することでより詳細なコンテキストの同定が可能となることが推測されるが、まずは、これらに着目してファイルイベントネットワークとして可視化することで、振舞い情報の可視化がコンテキストの想起に有用であることを示す。

まず、ファイルイベントネットワークを、ファイル f と f がコピーや編集された時刻 t の対 (f, t) をノード V とし、そのファイルに対する操作である「更新」、「参照」、「コピー」、「リネーム」をリンク E とする有効グラフと定義する。各ノードはそれぞれ以下のようなプロパティを保有している。

- ファイルイベント(ファイルの書き込み, 読み込み, コピー, リネーム, 削除)
- 時刻
- ファイルの重要語

表 1 ファイルのアクセスログ
Table 1 File access log.

日時	事象	事象コード	プロセス ID	スレッド ID	読み込みサイズ	書き込みサイズ	ファイル名	先ファイル名	プロセス名
2010/11/16 17:42:29:062	IO_FIL_MODIFY	0x00000040	0x17ac	0x1834	0	6	C:\test.txt		NOTEPAD.EXE
2010/11/16 17:42:41:984	IO_FIL_COPY	0x00000020	0x688	0x1648	0	6	C:\test.txt	C:\コピー~ test.txt	Mame5.exe

時刻 t_1 に生成されたファイル f_1 が時刻 t_2 に編集された場合、ノード (f_1, t_1) からノード (f_1, t_2) にリンクが張られる。また、時刻 t_1 にファイル f_1 からファイル f_2 にコピーされたとすると、ノード (f_1, t_1) からノード (f_2, t_1) にリンクが張られる。さらに、時刻 t_1 にファイル f_2 を参考にファイル f_1 を作成した場合、ノード (f_1, t_1) とノード (f_2, t_1) にリンクが張られる。リンクの生成については、3.1 節で詳細に述べる。ファイルイベントネットワークの辺は重み付きであり、コピーやリネームはファイルの内容自体は同一であるが、参照関係にあるファイルや、一部がコピーされた関係にあるファイルどうしなどにおいては、お互いの類似性が高いほど重みが大きくなるように可視化する。

このように、個々のファイルに対するファイルイベントネットワークを構築することで、ファイルが作成されるまでの過程や、あるファイルが起点となって様々なファイルが生成された過程などを可視化することができる。たとえば、あるファイル f_i から複数のファイルが生成されていれば、 f_i は、 f_i に関するタスクにおける根幹となったファイルであることが分かる。

また、ある文書を作成する過程で、バックアップのためのコピーや内容を修正するための編集を複数回繰り返した経緯もファイルイベントネットワークとして可視化できることから、どのファイルが最新であり、いつ編集やコピーされたのかも容易に把握することができる。また、提案法では、ファイルの参照関係もファイルイベントネットワークとして可視化することから、あるファイルが作成されたときに、どのファイルを参照したのかも容易に把握することができる。

先行研究として、我々はファイルイベントネットワークの情報を用いて、個人と複数人間の PC のコンテキストを描画する手法を提案している¹⁶⁾。先行研究では、1 つのファイル（種ファイル）に対するイベントとそのファイルに係るファイルのイベントを時系列上に表示する手法を提案しているが、この手法では、種ファイルの作成過程を知ることは可能であるが、コピーやリネームによって作成された類似ファイルのイベント情報を同時に閲覧することができないため、最新バージョンのファイルや不要ファイルを検出するといったことが不可能であった。提案法では、コピーやリネームによって作成されたファイルを同一

表 2 ウェブのアクセスログ
Table 2 Web access log.

日時	事象コード	URL	タイトル
2010-11-10-Wed-16-15-28	0	http://www.google.co.jp/	iGoogle - Mozilla Firefox
2010-11-11-Thr-16-25-53	1	http://www.yahoo.co.jp/	Yahoo! JAPAN - Mozilla Firefox

画面に表示することで、上記のことを可能とし、さらに、ブラウジング機能により、関連ファイルのファイルイベントネットワークも同時に閲覧できるため、1 つのタスクに関する個人の PC での活動内容を詳細に閲覧することができる。

以下の節でファイルイベントネットワークを構築する手順について述べる。

3.1 コンテキスト抽出

まずは、ユーザがファイルに関する一連の操作履歴を収集する必要があるが、我々は ManagementCore^{*1}を利用している。実際に ManagementCore で収集されるログを表 1 に示す。

ファイルに関する事象は以下の 5 種類が存在し、ファイルに関するユーザの操作を取得できる。

- IO_FIL_MODIFY：ファイルの作成・編集
- IO_FIL_READ：ファイルの閲覧
- IO_FIL_COPY：ファイルのコピー
- IO_FIL_RENAME：ファイルのリネーム
- IO_FIL_DELETE：ファイルの削除

表 1 の 1 行目は、“test.txt” ファイルが作成または編集されたことを示しており、2 行目は test.txt ファイルが“コピー~ test.txt”にコピーされたことを示している。

また、ファイルのアクセスログとは別にウェブ閲覧のログを収集する必要があるため、firefox のアドオンを作成し、ウェブ閲覧の履歴を収集した。実際に得られたログを表 2 に示す。

*1 IT 資産管理システム ManagementCore, <http://mcore.jp/>

事象コードは以下の3種類が存在し、ウェブに関するユーザのアクセス履歴を取得することができる。

- 0: 新規ページの読み込み
- 1: タブの切替え
- 2: ウェブページを閉じる

次に、内容が類似するファイルやウェブページは同一のタスクに関するリソースとして参照関係にあったり、お互いに依存関係があるタスクどうしに関するリソースであったりと、ファイルどうしのコンテンツの類似性はファイルイベントネットワークを構築する際に必須であることから、PC上の全ファイルと閲覧したウェブページに対して、重要語の抽出を行う。MS-OfficeファイルやPDFファイルのような特有のフォーマットファイルに対しても、xdoc2txtなどのバイナリ文書からテキストを抽出するツールを用いてテキストを抽出した。形態素解析にはSen^{*1}を用い、これと松尾らの重要語抽出法¹⁷⁾を用いて、重要語の抽出を行った。株に関する文章から重要語を抽出すると、「株主、換金、いつ、性、最大、優待、期待、配当、売買」などの重要語を抽出することができる。これにより、ファイル・ウェブアクセスと重要語から、ファイルの操作過程と関連するリソースを抽出することで、ファイルのコンテキストを獲得する。コンテキストとは人がそのタスクを実行するに際して行った行動履歴であり、そのコンテキストを思い出すことで、所望のファイルやそのファイルに関するタスクを思い出すことができる。このことから、我々はコンテキストを想起する手がかりとして、ファイル・ウェブのアクセス履歴と、その内容の両方に着目する。

本論文で扱わなかった重要なファイルアクセスイベントとして、クリップボードのトレースが考えられる。これは、本論文で用いているManagementCoreがファイルのIOを監視するロガーであり、クリップボードを対象としていないからである。しかしながら、ファイルを作成する際に、ファイルの一部をコピーすることがあるため、クリップボードの情報はファイル作成過程において重要な情報である。クリップボードの情報を用いることは今後の課題である。

3.2 リンクの生成

ファイルイベントネットワークはアクセスイベントをノードとし、そのアクセスイベントに関係のあるイベントをリンクで結んだネットワークである。ここでは、ノードとノードを結ぶリンクの生成について述べる。

3.2.1 同一ファイルにおけるリンク生成

PCには複数のファイルが存在しており、その利用のされ方も様々である。一度作成されて、その後まったく利用されないファイルもあれば、たびたび利用されるファイルもある。たびたび利用されるファイルについては、そのファイル自体が編集されたり、コピー・リネームされたりするため、ファイルの利用過程を明示するためには、アクセス履歴から、ファイルの利用過程を可視化する必要がある。本論文では、同一ファイルのアクセスとコピー・リネームについて、ファイルアクセス間にリンク強度1のリンクを張る。これにより、1つのファイルの利用過程を容易に把握することができる。以下にリンク強度1のリンクが張られる場合を示す。

同一ファイルに対するアクセス 同一ファイルに対する前後のアクセスに対してリンクを張る。また、削除やリネームがされずに存在しているファイルについては、ノードなしのリンクを張る(図1(上))。

コピー、リネーム コピー、もしくはリネームイベントに対してリンクを張る(図1(下))。

3.2.2 参照ファイルにおけるリンク生成

ファイルが作成されたり、編集されたりする場合、他のファイルを参照する場合が多々ある。参照されたファイルは作成・編集されたファイルに関連のあるファイルであるため、作成・編集されたファイルと参照されたファイルの間にもリンクを張る。本論文の提案するファイルイベントネットワークは各リンクに重みのついたネットワークであるため、リンク強度を以下のように定義する。

時刻 t_a にファイル f_a に対するアクセスイベント A が生じ、時刻 t_b にファイル f_b に対するアクセスイベント B が生じた場合のリンク強度の計算方法を以下に示す。ファイル間の類似度によるリンク強度の計算方法は以下のとおりである。

$$L_{A \rightarrow B} = \begin{cases} \frac{\text{num}(\text{KeyWord}_{f_a}(t_a) \cap \text{KeyWord}_{f_b}(t_b))}{\text{num}(\text{KeyWord}_{f_a}(t_a))} & t_a \cap t_b = \text{true} \\ 0 & t_a \cap t_b = \text{false} \end{cases}$$

$L_{A \rightarrow B}$ は、ファイル f_b に対するファイル f_a の類似度を示す。 $\text{KeyWord}_a(t_a)$ は時刻 t_a の時点でファイル a が持つ重要語であり、 $\text{num}(\text{KeyWord}_{f_a}(t_a))$ はその重要語数を示す。 $\text{KeyWord}_{f_a}(t_a) \cap \text{KeyWord}_{f_b}(t_b)$ は、時刻 t_a におけるファイル f_a と時刻 t_b におけるファイル f_b に、共通して見られる重要語であり、 $\text{num}(\text{KeyWord}_{f_a}(t_a) \cap \text{KeyWord}_{f_b}(t_b))$ はその共通の重要語数を示す。類似ファイルであっても、同時刻に利用されていない場合、参照されたファイルではないので、ファイルの利用時間帯 t_a と t_b が重なっていなければなら

*1 Sen, <http://www.mlab.im.dendai.ac.jp/~yamada/ir/MorphologicalAnalyzer/Sen.html>

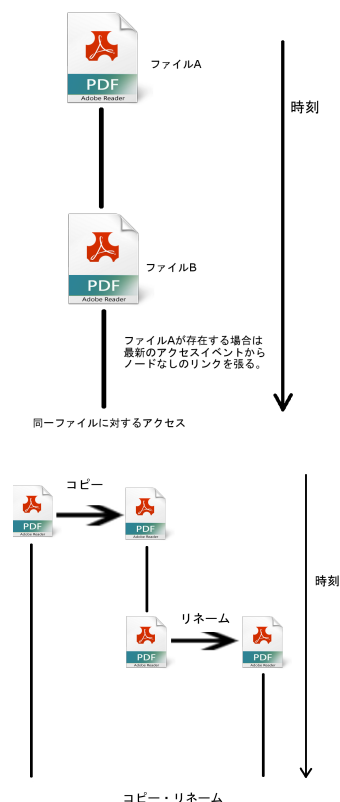


図1 リンクの生成例

Fig. 1 Examples of link generation.

ない。そのため、 $t_a \cap t_b = true$ (利用時間帯が重なっている) の場合にのみ、上式のリンク強度でリンクを張る。

上述したように、ファイルの関係を導くために、本論文ではファイルの中身を用いている。つまり、ファイルの中身が異なるファイルどうしが関連するファイルと見なされることはない。もちろん、ファイルの中身が異なるファイルどうしであっても関連するファイルである可能性もあるが、多くの場合、無関係なファイルであることのほうが多い。ファイル作成について考えると、あるファイルの中身を参考にファイルを作成することが多々あり、関

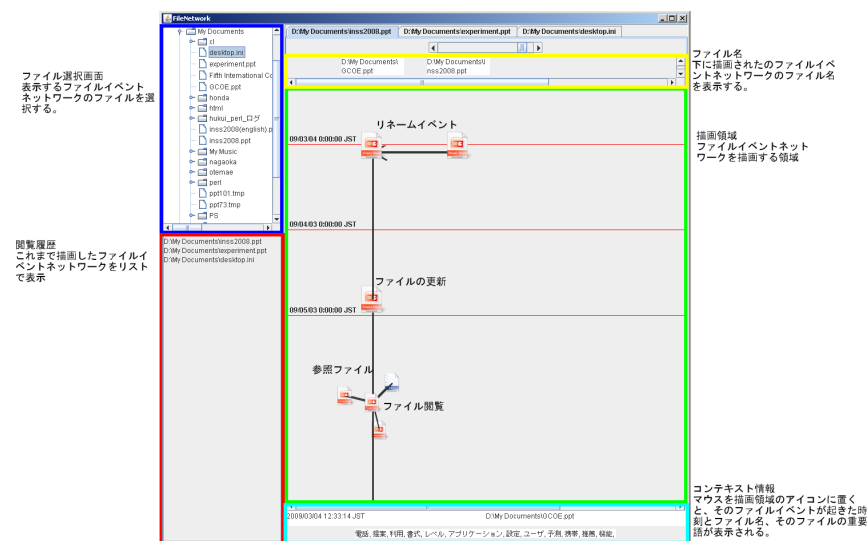


図2 ファイルイベントネットワークインタフェース

Fig. 2 File event network interface.

連するファイルの中身はある程度似た内容になっているものと考えられる。そのため、本論文ではファイルの中身の類似性を用いて同一のタスクかどうかを判断している。

3.3 ファイルイベントネットワークの可視化

作成したファイルイベントネットワークを時系列インタフェースに描画する。時系列インタフェースを用いることで、ファイルが持つ情報の遷移が明らかになる。たとえば、あるファイルに着目したとき、そのファイルが保有する情報が、他のファイルに分散していく様子が時系列インタフェースから取得できる。

図2に可視化インタフェースを示す。可視化インタフェースの左上部分がファイル選択画面になっており、その中の1つのファイル(種ファイル)を選択すると、種ファイルに関するファイルイベントネットワークが可視化される。可視化されたファイルイベントネットワークのアイコンの上にマウスポインタを置くと、そのファイルアクセス情報とそのファイルの重要語が画面下部に表示される。アイコンをクリックすると、そのファイルに関するファイルイベントネットワークが新たに表示される。これにより、ファイルイベントネットワークをたどっていくことで、所望のファイルにたどり着くことができる。

4. 実装と有用性の検証

提案法の有用性を示すために、以下の3つの実験を行った。

- (実験1) 長期間データを取得し、過去に作成したファイルに関するファイルイベントネットワークを可視化することによって、得られた知識の具体例を示し、タスク想起への効果を測る。
- (実験2) 被験者に複数のファイルを引用した新しいファイルを作成してもらい、3週間後に新しいファイル(種ファイル)を基に、引用されたファイルの検索やウェブ検索を行い、検索システムとしての有用性を測る。
- (実験3) ファイルの内容を考慮した場合と考慮しなかった場合におけるファイルイベントネットワークの違いを検証する。

4.1 パラメータと前処理

提案法は、ファイルに含まれる重要語とファイルのアクセス時刻を用いてファイルイベントネットワークを描画する。重要語については、得られる上位10個の単語を重要語とし、重要語が1つでも重複すれば関連するファイルと見なした。

ファイルのアクセス時刻に関しては、前後30分以内であれば、同時刻に利用していたものと見なし、30分以内に関連ファイルがアクセスされていれば、参照ファイルとしてファイルイベントネットワークに追加した。30分以内のアクセスに限って、同時刻に利用したと見なしたのは、本論文で用いたログ収集ソフトではファイルを閉じるというイベントを取得することができず、ファイルが参照されていた時間を取得することができないためである。そのため、1度開かれたファイルは30分程度は開かれているだろうと仮定した。ファイルを閉じるというイベントを取得することができれば、30分という前提を使う必要はなくなるが、30分が適切であるかどうかについてはコンテキストに依存する部分でもあり、検討する必要がある。

4.2 実験1: 可視化による効果

長期間データを取得し、普段のデスクトップ上での作業を取得するために、2台のPCを用意し、ファイルイベントネットワークを実際に構築し可視化する実験を行った。ユーザA: CPU Pentium(R) 4 3.60 GHz, メモリ 2 GB, OS Microsoft Windows XP と、ユーザB: CPU Core 2 Duo 3.06 GHz, メモリ 4 GB, OS Microsoft Windows XP である。ユーザAは7カ月間、ユーザBは約2カ月間にわたりログを収集した。ユーザAのログにおいて、解析の対象となったファイル数は2,440個である。

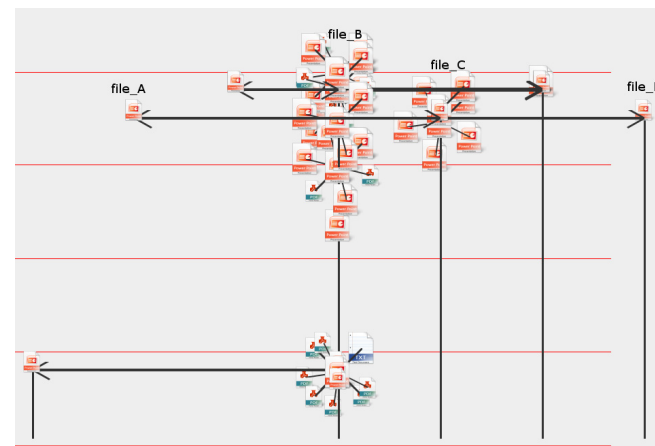


図3 可視化例1
Fig. 3 Example of visualization 1.

4.2.1 可視化例1

図3に被験者のPCにおけるファイルイベントネットワークの一例を示す。

図1に示すように、縦線は同一ファイルへのアクセスを示し、横の矢印はコピー・リネームを示す。図3はある発表資料file_Bのファイルイベントネットワークの全体図である。file_Bが最も古く作成されたファイルであるため、file_Bによって複数のファイルが作成されたことは分かるが、どのような手順で複製されたのかが分からない。そのため、コピーイベント部分を拡大することによって、file_Bの複製過程を詳細に示す。コピーイベント部分を拡大した図を図4に示す。

図4より、file_Bはfile_Aにコピーされ、file_Aはfile_Cというファイルに変更されており、file_Cは後にfile_Dにコピーされている様子が分かる。つまり、file_B → file_C → file_Dという流れでファイルが作成されていったことが分かる。

また、file_Bが作成された過程を見るために、ファイル作成部分を拡大したものを図5に示す。

図5より、file_Bは様々なファイルを参考にしながら、徐々に作成されていったことが分かる。また、図3より、file_Bは作られた時期が最も古いにもかかわらず、編集された時期が他の派生ファイルに比べて最も新しいため、file_Cやfile_Dよりも新しい情報が付け加え

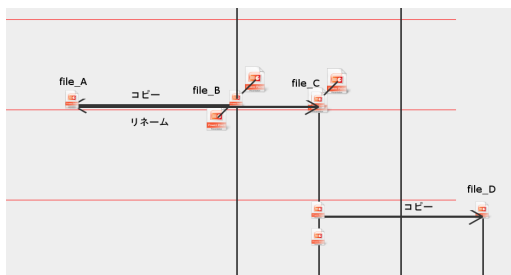


図 4 可視化例 1 の拡大図 (コピーイベント)

Fig. 4 Expanded figure for example operation 1 (the process of copying file_B).

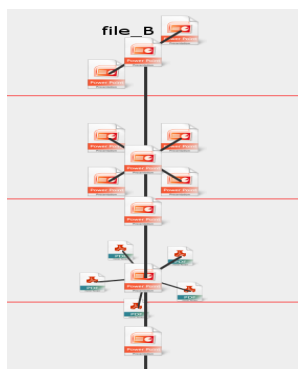


図 5 可視化例 1 の拡大図 (file_B の作成過程)

Fig. 5 Expanded figure for example operation 1 (the process of making file_B).

られた、つまり新しいバージョンのファイルである可能性が高い。このように、ファイルの編集時期を派生ファイルと簡単に比較することができるため、提案法はファイルのバージョン管理を行ううえでも有用であると考えられる。

4.2.2 可視化例 2

図 6 は提出資料 file_H を作成した際のファイルイベントネットワークを表している。file_H は 1 つのファイルを参考に作成されており、後に file_G にコピーされている。作成日時から数時間という短期間に作成された資料であることが分かる。また、作成されてから後に参照されていないことから、最近では利用されていないファイルであることが分かる。また、

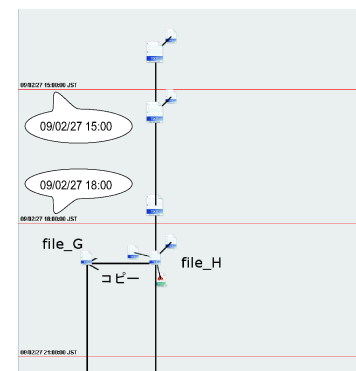


図 6 可視化例 2

Fig. 6 Example of visualization 2.

file_G にコピーされた後、file_G が編集されていないため、同一のファイルが PC 内に 2 つ存在していることが分かる。これにより、file_G がバックアップのためのファイルでないとすれば、file_G は不要なファイルであることが分かる。

4.3 実験 2: 検索システムとしての評価

本論文では、タスクを思い出すことを目的に、ファイルイベントネットワークの可視化を行っている。しかしながら、タスクが思い出せたのかどうかということについては、心理実験やアンケートによる評価を行う必要があり、定量的な評価を行うのは難しい。そこで、我々は、タスクを思い出す ⇨ ファイルを作成した際にどのファイルを参考にしたのかを思い出すこととした。そのため、参考にしたファイルをどれだけ早く正確に見つけることができるのか、比較実験を行った。参考にしたファイルをいち早く見つけ出すことが、いち早くタスクの内容を思い出すことにつながると考えている。

4.3.1 実験 2-1: ファイル検索の評価

ファイルを保管する際、適切なフォルダに割り振って保管することが多いが、ファイルの作成時期や内容によって、1 つの適切なフォルダにファイルを割り振ることができないことがある (たとえば、論文内に登場する実験結果の画像ファイルを学会フォルダに保管するか、実験データフォルダに保管するか迷う場合がある)。このような場合、同じファイルをコピーして別々のフォルダに保存したり、研究関連フォルダといった浅い階層のフォルダにまとめて保存しておいたりするといったファイル管理をすることがある。また、一時的に

デスクトップ上にファイルを置くようなこともあるため、1つのフォルダに複数のファイルが存在していたり、同一のファイルがPC内に複数存在していたりする状況が考えられる。我々は、1つのフォルダに複数のファイルが存在している状況において、適切にファイル検索を行うことができるかを検証した。実験のために、77個のファイルを用意し、1つのフォルダの中に格納した。77個のファイルの中身は株(10)、法律(13)、ネットワーク(14)、ニュース(12)、テクノロジー(15)、スポーツ(13)に関する記事で、カッコ内はその内容に関するファイル数である。ファイル名については、ファイルに書かれてある最初の一文(句読点まで)をファイル名とした。たとえば、株に関するファイルであれば、「株式投資で成功するかどうか」や「株式投資には魅力もあるがリスクもある」があり、お互いに内容の似たファイルも存在する。一方、法律関係では「裁判員制度の紹介」や「貸金返還請求事件」といったファイルが存在し、ファイルによっては、「昨年」、「現在」といったファイル名だけではファイルの中身が分からないファイルも存在する。実験の流れを以下に示す。

- (1) 77個のファイルから、筆者が指定した株に関する4つのファイル(元ファイル)を参考に、1つのファイル(種ファイル)を作成してもらう。4つのファイル名を以下に示す。
 - ドルコスト平均法で買い付け
 - 株式投資を始めて株主になるメリット
 - 金融商品を選ぶには
 - 連続したローソク足で株価の動きを見てみよう
- (2) 数日後、種ファイルのバックアップを作成してもらい、さらに、元ファイルとは別のファイル(ファイル名は「アナリストの銘柄診断は要チェック」)を用いて、種ファイルに内容を追加してもらう。
- (3) ファイル作成から3週間後、種ファイルを参考に4つの元ファイルをすべて探し出してもらう(制限時間5分)。

元ファイルの中で見つけることができたファイルの数や検索に要したクリック数と時間から提案法の有用性を検証する。検索方法は以下の3つである。

- フォルダ内のファイルをファイル名を手がかりに検索
- キーワードによる検索(Windowsのファイル検索機能)
- ファイルイベントネットワークインタフェースを用いて検索

本論文では、ファイル数77個、実験期間3週間という実環境より小規模の環境で実験を行っているため、実環境よりファイルを見つけやすい環境での実験といえる。しかし、一時

表3 見つけることができたファイル数(見つけるまでに要したクリック数と時間(秒))

Table 3 No. of files that was able to be bound (No. of clicks and time (seconds) which was required by finding the files).

被験者	フォルダ	キーワード	提案法
A	3 (22, 95)	3 (55, 300)	4 (18, 58)
B	3 (35, 163)	3 (25, 97)	3 (30, 131)
C	2 (60, 300)	3 (78, 300)	4 (22, 73)
D	3 (45, 182)	2 (40, 105)	2 (30, 60)
E	4 (19, 91)	3 (90, 300)	4 (26, 83)
F	4 (46, 180)	2 (70, 300)	4 (36, 205)
G	3 (47, 208)	4 (30, 113)	4 (21, 258)
H	4 (63, 275)	4 (100, 245)	4 (20, 50)
I	4 (50, 216)	3 (100, 300)	4 (35, 47)

的にファイルを作成してもらう行為としては、ファイルが77個あり3週間ほど時間がたてば、人間がファイル操作を忘れるのに十分だと考えて、このような実験設定にした。実験結果を表3に示す。

表3に示された数値は元ファイルの中で、被験者が見つけることができたファイルの数であり、括弧内の数値はそれぞれファイルを見つけるまでに要したクリック数(左)と時間(秒数)(右)である。太字の数値は3つの検索方法のなかで、最も良い結果を示した数値である。表3から、提案法が最も正確にファイル検索を行えることが分かった。被験者BとDは、すべての元ファイルを見つけることができなかったが、これは、元ファイルに含まれるキーワードが作成されたファイルの中に存在しなかったためである。また、被験者Bにおいて、提案法による検索に時間がかかった原因は、種ファイルのイベント間隔が短かったために、目的のファイルアイコンをなかなか見つけることができなかったためである。インタフェースとしての改良は必要であるが、ファイルイベントネットワークを示すことによって、ファイル作成時のコンテキストを思い出させることができ、所望のファイルを検索できたことが実験により示された。

被験者Aのファイルイベントネットワークを図7に示す。

図7より、ファイルが作成された際に、複数のファイルが参照されていることが分かる。また、ファイルが作成された後に、バックアップのためにファイルがコピーされていることも分かり、バックアップファイルを探すのも容易であることが分かる。

4.3.2 実験 2-2: ウェブ検索の評価

文書ファイルを作成する際、ローカルファイル以外にもウェブページを参考に文書を作成

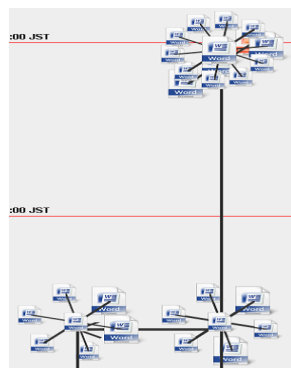


図7 被験者 A のファイルイベントネットワーク
Fig. 7 File event network for examinee A.

することがあるが、そのようなウェブページはブックマークされることなく、閲覧後閉じられることが多い。そのため、文書を編集したり、新たに似たようなファイルを作成したりする際に、参考にしたウェブページを閲覧する必要があり、そのためキーワード検索を用いて再度ウェブ検索を行うことがある。しかしながら、時間が経過している場合、簡単なキーワード検索だけでは目的のウェブページを見つけることができないこともあり、そのようなウェブページに対して、複数または特定のワードを入れて検索したり、ウェブの閲覧履歴から探したりするが、いずれの場合も多くの時間を費やすことが考えられる。そこで、ファイルに関係するウェブページをファイルイベントネットワークに表示することで、ウェブページを参考に作成されたファイルから、参考にしたウェブページを効率的に検索することができるか検証を行った。実験の流れを以下に示す。

- (1) 「先物市場」とは何かをウェブページを参考にファイルにまとめてもらう（参考にするウェブページは被験者の任意であり、参考にしたウェブページ数は被験者によって異なる）。
- (2) ファイル作成から3週間後、種ファイルを元に参考にしたウェブページを探し出してもらう（制限時間5分）。

参考にしたウェブページの中で、見つけることができたウェブページ数や検索に要したクリック数と時間から提案法の有用性を検証する。検索方法は以下の2つである。

- キーワードによる検索（被験者が普段使い慣れている検索エンジンを使用する）

表4 検索できたウェブページ数/参考にしたウェブページ数（見つけるまでに要したクリック数と時間（秒））
Table 4 No. of Web pages that was able to be bound/No. of Web pages that was referred (No. of clicks and time (seconds) which was required by finding the files).

被験者	キーワード	提案法
A	2/2 (12, 39)	2/2 (9, 17)
B	1/1 (10, 34)	1/1 (4, 20)
C	3/3 (30, 125)	3/3 (12, 38)
D	1/1 (12, 38)	1/1 (4, 7)
E	4/4 (65, 215)	4/4 (40, 114)
F	4/5 (44, 300)	5/5 (34, 123)
G	2/2 (12, 52)	2/2 (14, 40)
H	3/3 (55, 182)	3/3 (40, 167)
I	2/2 (30, 57)	2/2 (16, 37)

● ファイルイベントネットワークインタフェースを用いて検索

実験結果を表4に示す。

表4に示された数値は検索できたウェブページ数/参考にしたウェブページ数であり、括弧内はクリック数と時間（秒）である。ほとんどの被験者がどちらの手法でもすべてのウェブページを見つけることができた。これは、ほとんどの被験者が Wikipedia や Yahoo 知恵袋といった検索ランキング10位以内のウェブページを参考にファイルを作成したためである。検索時間を比較すると、より多くのウェブページを参考にした被験者に対して、提案法は有効であることが示された。これは、キーワード検索が1つ1つのウェブページに対して、適切なキーワードを選んで逐一検索する必要があるのに対して、提案法では、1つのファイルイベントネットワークの中に参考にしたウェブページ候補がすべてあげられているため、簡単に検索できたのだと考えられる。また、被験者Fにおいて、キーワード検索で見つけることができなかったファイルを、提案法では検索することができている。

被験者Hのファイルイベントネットワークを図8に示す。

4.3.3 実験2-3：アンケートによる評価

被験者9名に提案法の検索システムについて、無記名のアンケートに答えてもらった。アンケート内容と回答は以下のとおりである。

- (1) 3つの検索方法でファイル検索を行いました。簡単にファイルを探せた順に並べてください。

- フォルダ内のファイルを手探りで検索

– 1位 1名



図 8 被験者 H のファイルイベントネットワーク
Fig. 8 File event network for examinee H.



図 9 ファイルの中身を用いなかった場合
Fig. 9 A file event network without a file's contents.

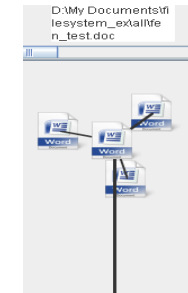


図 10 ファイルの中身を用いた場合
Fig. 10 A file event network with a file's contents.

- 2位 4名
- 3位 4名

● キーワードによる検索

- 1位 0名
- 2位 4名
- 3位 5名

● 提案法による検索

- 1位 8名
- 2位 1名
- 3位 0名

(2) ファイルイベントネットワークを見ることで、ファイル作成の流れを思い出すことができましたか？

- まったく思い出せなかった。0名
- 少しは思い出せた。2名
- かなり思い出すことができた。7名

(3) ファイルイベントネットワークインタフェースは、検索システムとして、有用だと思いますか？

- 有用だとは思わない。1名
- 少しは有用である。2名
- かなり有用である。6名

ファイル検索を行った際に、提案法が最も簡単にファイルを探すことができた理由について、参考にしたファイルが絞られていたため、簡単にファイルを検索することができたとい

う回答が最も多かった。提案法を2位にあげた被験者は、ファイル名を覚えており、フォルダ内を探すほうが簡単だったと回答している。インタフェースについては、操作性の面での改良が必要だという意見を多くいただいたが、直感的に操作できれば、有用であるという回答を得ることができた。

4.3.4 実験3：ファイルの内容の重要性

本論文では、ファイルのアクセスタイム以外にもファイルの内容を利用した点が他の手法と異なる。ファイルの内容を用いた場合と用いなかった場合（アクセス時刻のみ）について、比較を行った。種ファイルに関連のあるファイルは3つである。結果を、図9、図10に示す。

図9、図10に示すように、ファイル内容を用いることで、タスクに関係のあるファイルアクセスだけにリンクが張られていることが分かる。

提案法はファイルの中身の類似度を重要語によって判定している。そのため、重要語が共起しなければ、関係のあるファイルとは見なされず、重要語が1つでも共起していれば関係のあるファイルと見なされる。そこで、重要語の数を5~30個に5つずつ増やすことで、どれほど関係のあるファイルを抽出することができ、どれほど無関係なファイルを抽出してしまうのかについて検証を行った。対象としたファイルは実験2で用いた77個のファイルであり、実験2で作成したファイルが株関係のファイル10個とどれほど共起するかについて検証している。図11に結果を示す。

重要語数が増えれば増えるだけ、関連していると見なされるファイルの数は増え、無関係

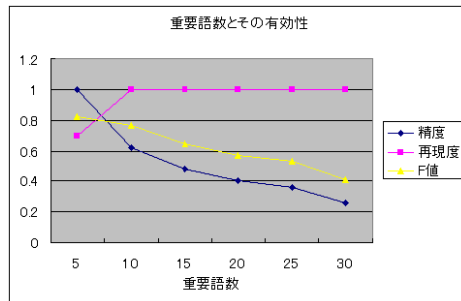


図 11 重要語数と関連ファイル数の関係

Fig. 11 Relationship between number of keywords and number of related files.

なファイルまで関連ファイルと見なされてしまうため、精度が落ちる。一方、重要語数が少なければ、関連のあるファイルが関連のないファイルであると見なされてしまうため、再現率が落ちる。本実験においては、すべての関連ファイルを網羅し、かつ精度が高い重要語数 10 という値が最も適切な値であると考えられる。しかし、ファイルによってはこの値が適切であるかどうか分からない。そのため、インタフェース上でユーザが関連度の閾値を変更できるように改良する必要がある。

4.4 ブラウジングインタフェース

被験者実験により、ファイルイベントネットワークを可視化することによって、ユーザがタスクの内容を思い出し、所望のファイルを検索することが可能であることが示された。しかしながら、提案法は種ファイルに関するファイルイベントネットワークを作成するものであり、種ファイルを探すことができなければ、所望のファイルを見つけることができない。そこで、我々は関連のありそうなファイルのファイルイベントネットワークをたどることによって、所望のファイルを検索することができるブラウジング機能をインタフェースに追加した(図 12 参照)。ブラウジング機能により、個々のファイルアイコンをクリックすることで、そのファイルを中心とするネットワークが随時展開される。

図 12 のように、file_A についてのファイルイベントネットワークを可視化することによって、file_A は file_B を参照して作成されたファイルであることが分かる。さらに、file_B のアイコンをクリックすることによって、同一のインタフェース上に file_B のファイルイベントネットワークを追加することができる。このように、関係のあるファイルについて、ファイルイベントネットワークを展開していくことで、各ファイルのコンテキストが明らかにな

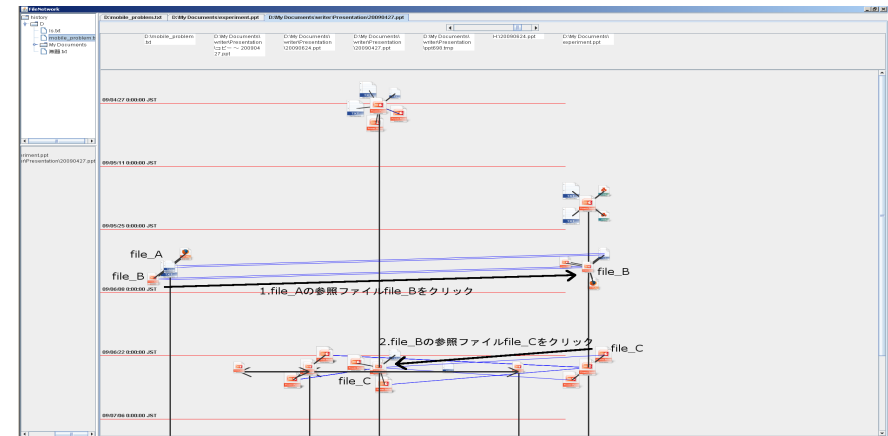


図 12 ブラウジングインタフェース

Fig. 12 Browsing interface.

る。これにより、ユーザは各ファイルの関係性とその操作内容を同時に見ることができ、より鮮明にユーザのタスクを思い出すことができる。また、ネットワークを展開していくことで、所望のファイルにたどり着くことが可能となる。

5. ま と め

本論文では、従来のファイル名やキーワードによる検索と異なり、ファイルに対する一連の操作に関する時系列履歴を、「ファイルイベントネットワーク」として可視化する方法を提案した。さらに、被験者実験を行うことによって、ファイルイベントネットワークをユーザに提示することで、ファイルに関するコンテキストを思い出し、効率的にファイルやウェブ検索を行うことができることを示した。

今後の課題としては、メールの送受信は重要なコンテキストであり、図や音楽といった文書ファイル以外のファイルについても、ファイルイベントネットワークを構築する必要があると考えている。図や音楽ファイルには、文書ファイルのようなファイルの内容を表す要素がないため、ファイルイベントとして、文書ファイルへの図の添付情報の取得が必要となる。また、ファイルが開いている時間を取得することで、ファイルの同時利用を抽出することが可能であるため、より精度良く参照ファイルを抽出することができる。上述のようなログを取得することができれば、PC 上におけるユーザの振舞いをファイルイベントネット

ワークによって正確に表現でき、1つのタスクに関するユーザの行動を容易に把握できる。

また、現在は単一のPCのファイルのみを対象としているが、複数PCや複数人への拡張も可能である。たとえば、PC間でのファイルの移動を把握したり、メールに添付されることで人から人にファイルが移動するイベントを把握したりすることも可能である。これにより、組織内の情報の流れや、メンバの活動などを把握でき、たとえばシステム開発におけるプロジェクトの進捗状況をリアルタイムに把握することも可能である。

参 考 文 献

- 1) Teevan, J., Alvarado, C., Ackerman, M. and Karger, D.: The Perfect Search Engine Is Not Enough: A Study of Orienteering Behavior in Directed Search, *CHI '04: Proc. ACM Conference on Human Factors in Computing Systems*, pp.415–422 (2006).
- 2) Cohen, S., Domshlak C. and Zwerdling, N.: On ranking techniques for desktop search, *Transactions on Information Systems (TOIS)*, Vol.26, Issue 2, pp.11:1–11:24 (2008).
- 3) Zobel, J. and Moffat, A.: Inverted Files for Text Search Engines, *ACM Computing Surveys*, Vol.38, Issue 2, pp.1–56 (2006).
- 4) Shah, S., Soules, C., Ganger, G. and Noble, B.: Using Provenance to Aid in Personal File Search, *Proc. USENIX Annual Technical Conference (USENIX 2007)*, pp.1–14 (2007).
- 5) Cutrell, E., Robbins, D., Dumais, S. and Sarin, R.: Fast, flexible filtering with PHLAT — Personal search and organization made easy, *CHI '06: Proc. SIGCHI Conference on Human Factors in Computing Systems*, pp.261–270 (2006).
- 6) Nejdil, W. and Paiu, R.: I know I stored it somewhere — Contextual Information and Ranking on our Desktop, *8th International Workshop of the EU DELOS Network of Excellence on Future Digital Library Management Systems*, pp.1–5 (2005).
- 7) Sauermann, L., Bernardi, A. and Dengel, A.: Overview and Outlook on the Semantic Desktop, *Proc. 1st Workshop on the Semantic Desktop at the ISWC 2005 Conference*, pp.1–18 (2005).
- 8) Blanc-Brude, T. and Scapin, D.: What do people recall about their documents?: Implications for desktop search tools, *IUI '07 Proc. 12th International Conference on Intelligent User Interfaces*, pp.102–111 (2007).
- 9) Ringel, M., Cutrell, E., Dumais, S. and Horvitz, E.: Milestones in time: The value of landmarks in retrieving information from personal stores, *Proc. Interact 2003*, pp.184–191 (2003).
- 10) 大澤 亮, 高汐一紀, 徳田英幸: 俺デスク: ユーザ操作履歴に基づく情報想起支援ツ

ル, 情報処理学会第47回プログラミング・シンポジウム(2005).

- 11) 渡部徹太郎, 小林隆志, 横田治夫: キーワード非含有ファイルを検索可能とするファイル間関連度を用いた検索手法の評価, データ工学ワークショップ(DEWS2008), E10-6 (2008).
- 12) Soules, C.A.N. and Ganger, G.R.: Connections: Using context to enhance file search, *Proc. 20th ACM Symposium on Operating Systems Principles*, pp.119–132 (2005).
- 13) Rekimoto, J.: Time-Machine Computing: A Time-centric Approach for the Information Environment, *UIST '99: Proc. ACM Symposium on User Interface Software and Technology*, pp.45–54, ACM (1999).
- 14) 牛尼剛聡, 渡邊豊英: 経験コンテンツ検索における意味的リンク構造を利用したランキングに関する考察, 日本データベース学会 Letters, Vol.6, No.2, pp.73–76 (2007).
- 15) Shen, J., Fitzhenry, E. and Dietterich, T.: Discovering frequent work procedures from resource connections, *The 13th International Conference on Intelligent User Interfaces (IUI '09)*, pp.277–285 (2009).
- 16) 福井秀徳, 森田哲郎, 岡野真一, 沼尾正行, 栗原 聡: ファイルネットワークに基づいた情報の抽出と可視化, 人工知能学会全国大会(第22回)論文集, CD-ROM (2008).
- 17) 松尾 豊, 石塚 満: 語の共起の統計情報に基づく文書からのキーワード抽出アルゴリズム, 人工知能学会誌, Vol.17, No.3, pp.213–227 (2002).

(平成22年11月18日受付)

(平成23年2月1日再受付)

(平成23年3月7日採録)



松本 光弘 (学生会員)

1982年生。2008年大阪大学大学院情報科学研究科情報数理学専攻修士課程修了。同年大阪大学大学院情報科学研究科情報数理学専攻博士課程入学。人物行動マイニングの研究に従事。人工知能学会学生会員。



岡野 真一

1973年生。1999年大阪大学大学院基礎工学部システム工学科修士課程修了。同年住友電気工業(株)入社。ネットワークシステムの研究開発に従事。



森田 哲郎

1981年大阪大学大学院工学研究科電子工学研究科前期博士課程修了。同年住友電気工業株式会社入社。横浜研究所，工業技術院光技術共同研究所出向，オプトエレクトロニクス研究所，通信放送機構神戸多段接続リサーチセンター出向を経て，システムエレクトロニクス研究開発センターに所属。現在，住友電気工業情報通信研究所主席。符号理論，エージェント技術の研究に従事。2005年より情報通信技術委員会(TTC)次世代ホームネットワーク専門委員会委員。2007年より大阪市立大学大学院工学研究科電気情報系非常勤講師。著書『光電子集積回路の基礎技術』(オーム社，共著)。『ユビキタス技術「ホームネットワークと情報家電」』(オーム社，共著)。工学博士。電子情報通信学会会員。



沼尾 正行(正会員)

1982年東京工業大学工学部電気電子工学科卒業。1987年東京工業大学大学院理工学研究科情報工学専攻博士後期課程修了。工学博士。同年東京工業大学工学部助手。1989年から1990年スタンフォード大学言語・情報研究所客員研究員。2003年大阪大学産業科学研究所教授，現在に至る。情報処理学会知能と複雑系研究会主査，代表会員，人工知能学会評議員，理事を歴任。2006年4月から2009年3月まで日本学術振興会学術システム研究センター主任研究員を併任。



栗原 聡(正会員)

1992年慶應義塾大学大学院理工学研究科計算機科学専攻修士課程修了。同年日本電信電話株式会社入社。基礎研究所を経て未来ねっと研究所に所属。1998年から慶應義塾大学大学院政策・メディア研究科専任講師(有期)。現在，同大学環境情報学部非常勤講師。2004年から大阪大学産業科学研究所知能システム科学研究部門准教授(同大学大学院情報科学研究科情報数理学専攻准教授兼務)。マルチエージェント，ネットワーク科学等の研究に従事。著書『社会基盤としての情報通信』(共立出版，共著)。翻訳『スモールワールド』(東京電機大学出版局，共訳)。編集『Emergent Intelligence of Networked Agents』(Springer in Computational Intelligence Series)等。博士(工学)。人工知能学会，日本ソフトウェア科学会，電子情報通信学会，人間情報学会，ACM，ESHIA 各会員。