論文 / 著書情報
Article / Book Information

論題(和文)	RMC操作に基づくタスクとタスク間関連度を考慮したファイル検索
Title(English)	Search File by Using Task and Intertask Relationships Based on RMC Operations
著者(和文)	呉怡, 渡辺陽介, 横田治夫
Authors(English)	Yi Wu, Yousuke Watanabe, Haruo Yokota
出典(和文)	電子情報通信学会論文誌 D, Vol. J96-D, No. 5, pp. 1166-1177
Citation(English)	IEICE Transactions on Information and Systems, Vol. J96-D, No. 5, pp. 1166-1177
発行日 / Pub. date	2013, 5
URL	http://search.ieice.org/
権利情報 / Copyright	本著作物の著作権は電子情報通信学会に帰属します。 Copyright (c) 2013 Institute of Electronics, Information and Communication Engineers.

RMC操作に基づくタスクとタスク間関連度を考慮したファイル検索

呉　　怡[†*]　　渡辺　陽介[††]　　横田　治夫[†]

Search File by Using Task and Intertask Relationships Based on RMC Operations

Yi WU[†*], Yousuke WATANABE[††], and Haruo YOKOTA[†]

あらまし　近年，ファイルシステム内に格納されているデータ量の急速な増大に伴い，膨大な数のファイルの中から，必要な情報を探し出すことは困難である．これまで，全文検索によるデスクトップ検索は主なアプローチとして用いられてきたが，検索キーワードを含まないファイルが検索できない．そこで我々は全文検索によるファイル検索の結果を改善するために，個々のファイルがもつ情報に加え，関連ファイル群間の相関関係を利用する手法を提案する．ファイル間関係の抽出においては，ユーザの操作を記録したファイルアクセスログを使用している．本研究はまず，同一作業に関連するファイルは頻繁に近い時間に使用される傾向があることから，このようなファイル集合を「タスク」として抽出する．続いてファイル間の改名・移動・コピー（RMC）操作を考慮し，タスク間関連度を数値化する．プロトタイプシステムによる被験者実験により，抽出したタスクとタスク間関連度を取り入れることで，ファイル検索の結果が大きく改善されたことを確認する．

キーワード　ファイルアクセスログ，デスクトップ検索，全文検索，タスクマイニング

1. まえがき

近年，個人が扱うファイルの数が日々増加しており，ファイルの内容及びファイル間の関係を把握することが困難である．膨大なデータの中から必要な情報を見つけるため，これまで数多くのデスクトップ検索ツールが開発されてきた．代表的なデスクトップ検索ツールとして，グーグルの Google デスクトップ [1] や，マイクロソフトの Windows デスクトップサーチ [2]，Mac OS X に搭載されている Spotlight [3] などがある．Google デスクトップでは，Windows だけでなく，Linux などのプラットホームにも対応している．メール，ファイル，ウェブの履歴などから抽出したインデックスを利用して，パソコン上のデータを高速に検索できる．それに対して，Windows デスクトップサーチや，Spotlight，Linux 向けに開発された Beagle [4] などは OS に特化したデスクトップ検索ツールである．これらのデスクトップ検索システムはどれも高速なインデックシングと検索を実現しており，テキスト情報のほかにファイル名や作成日時といった簡単なメタ情報を利用しているが，画像やビデオファイルのようなファイルを検索することが難しい．このように，全文検索のみ利用した検索の場合は，キーワードを含まないファイルが検索できないため，再現率の低下が回避できない問題がある．一方で，ファイルアクセスログを加味した研究も進められている．例として，Connections [5]，FRIDAL [6], [7] などがある．これらのシステムはアクセス共起に基づくファイル間の関連性を用いており，テキストを含まないファイルの検索に有効だが，偶然の同時アクセスによって間違った相関関係が抽出されることや，アクセス間隔が近くても共起しなければ相関関係が抽出できないといった問題点が存在する．更に，ファイルは名称が変更されたり，コピーされたりすると，元のファイルとそれのもつ相関関係が失われるまたは追跡できなくなるため，本来のあるはずのアクセス共起が発見できないことや，タスクの間の関係を発見できなくなってしまう．このため，アクセス共起だけで全文検索の結果を拡張しても，偶然の同時アクセスにより適合率が低下する可能

† 東京工業大学大学院情報理工学研究科計算工学専攻，東京都
Department of Computer Science, Graduate School of Information Science and Engineering, Tokyo Institute of Technology, 2–12–1 Ookayama, Meguro-ku, Tokyo, 152–8552 Japan
†† 東京工業大学学術国際情報センター，東京都
Global Scientific Information and Computing Center, Tokyo Institute of Technology, 2–12–1 Ookayama, Meguro-ku, Tokyo, 152–8552 Japan
* 現在，(株) エヌ・ティ・ティ・データ

性が大きく，また相関関係が十分に抽出できず，再現率の改善が少ない．

これに対して，我々はタスクの考え方を取り入れ，ファイルをタスク単位にまとめて，タスクとしての重みとタスク間の関連を利用した手法 SUGOI (Search by Utilizing Groups Of Interrelated files in a task) を提案している [8]．本論文は，文献 [8] に実験データと考察を追加し，提案手法の評価を更に詳細化した拡充版である．

SUGOI は，頻繁に近い時間にアクセスされたファイルは同一作業に関連することを利用している．ここで，作業とは書類作成のような複数のファイルをアクセスしながら行う論理的に一つの仕事を想定し，本論文ではある作業に関連するファイルの集合を「タスク」と表す．ファイルをタスク単位にまとめることで，ファイルシステム内に分散しているファイル間の関連性が分かり，検索する際の再現率向上につながる．「頻繁に」の制約を付けることにより，偶然の同時アクセスによる適合率の低下を防ぐことが可能になり，「同時」ではなく，「近い時間」内のアクセスに着目することで，多少の間隔があるアクセスでも相関の抽出が可能となる．更に，SUGOI では同一タスク内のファイル間の関連性だけでなく，タスク同士の関係まで考慮することで，検索結果を改善する．SUGOI はファイルの改名（Rename）・移動（Move）・コピー（Copy）（以降，RMC）操作の情報も使用することで，アクセス共起だけでは検索できないファイルも検索できることを特徴とする．過去の作業で使用したファイルを RMC して，新しい作業で再利用することはよくあるが，ファイル間の関連という面からみると，元のファイルに関連のあるファイル群と，RMC によって新しくできたファイルに関連のあるファイル群との間にも，相関関係があると推測できるが，アクセス共起のみ用いる手法では，このような相関関係は検出できない．RMC 操作を考慮して初めて，RMC 操作によって切れていたタスク間の相関も見つけ出すことができるようになる．更に，近い時間に RMC 操作される複数のファイルは互いにタスクとして関連していることがあるため，タスクを検出するためにも RMC 操作を利用することができる．つまり，SUGOI では RMC 操作を考慮してタスクのマイニングを行うと同時に，RMC 操作を利用してタスク間の関連度も算出している．タスク間の関連度をタスクとしてのスコアに反映させて，キーワード検索の結果のファイルを含む高いスコアのタスクを SUGOI の検索結果とすることで，キーワードに関連するファイルを提供する．なお，タスク間の関連度を算出する際には，RMC 操作が発生してからの経過時間，編集回数，ファイルサイズの変化等を考慮する方法も取り入れている．

評価実験では，我々の研究グループが日常的に使っているファイルサーバにおけるファイルアクセスログを使用し，提案手法は既存手法に比べ，適合率，再現率，F 値が大きく改善されたことを示す．

以下に論文の構成を述べる．**2.** ではデスクトップ検索とファイル整理に関する既存研究を紹介する．**3.** で提案手法である SUGOI の詳細を説明する．**4.** において評価実験と考察を記述し，**5.** にてまとめと今後の課題について述べる．

2. 関連研究

本章では，ファイルアクセスログを利用した既存研究について紹介する．

2.1 アクセスログを用いたファイル検索

Soules らが提案した Connections [5] はシステムコールのログを使用したファイル検索システムである．Connections は一定期間内におけるファイル操作から，ファイル間の参照・被参照関係の有無を判別し，ファイル間の関係を表す重み付グラフを作成する．検索する際は，エッジの向きと重みでノードであるファイルの重みを伝搬させ，全文検索の結果を拡張している．Connections では参照，書込みのほかに，コピーや改名といった操作も使われているが，その情報を関連度の算出に用いていない．更に，同一作業としてのタスクには注目しておらず，本論文で提案しているようなアプローチはとられていない．

渡部らが開発した FRIDAL [6], [7] では，ファイルの open・close ログから得られるファイルが使用される時間の間隔を用いている．その考え方は，同時に使用したファイルは互いに関連することに基づいている．ファイル間関連度の算出においては，共起時間や共起回数といった情報を用いている．検索の際は，キーワードを含むファイルのスコアをファイル間関連度によってキーワードを含まないファイルへと伝搬させることで，キーワード非含有ファイルの検索が可能となる．これに対して，本研究ではタスクに注目しており，頻繁に近い時間にアクセスされたファイル群をタスクとして抽出してから，RMC 操作を考慮したタスク間の関連度を検索に用いているため，FRIDAL では検

出できなかったファイルも検出できる．

Chen らが提案した iMecho [9] はシステムレベルのアクセスログと，専用のプラグインによって収集されるアプリケーションの操作ログを利用している．iMecho では，ファイルの内容と各種のログからファイル間に相関リンクを張らせており，その際，コピー操作やタスクマイニングの結果を用いているが，本論文のようにタスク間の関係を考慮していない．また，iMecho はランキングの精度改善のみを目的としているため，キーワードを含まないファイルは検索結果として提示されることはない．それに対して，本研究ではキーワードを含まない関連ファイルも検索可能である．

2.2 ファイルクラスタリング

小田切らは複数のディレクトリに分散している同一作業に関連するファイル群を発見する COFI (Clustering using Overlap of file-use time for Frequent Itemsets) 法 [10] を提案している．COFI 法では，同一作業に関連するファイルは頻繁に近い時間に使われるという性質を利用し，ファイルを作業単位にクラスタリングした仮想ディレクトリーを生成している．作業ごとに使用されたファイル群を求めるため，COFI 法は，頻繁に近い時間にアクセスされるファイルの集合を発見し，集合間のアクセス時間の重複度に基づき，階層的クラスタリングを行うことにより，仮想ディレクトリを生成している．これに対して，本研究におけるタスクマイニングは，ファイルクラスタリングのためではなく，キーワード検索の結果を改善するためである．また，COFI 法ではファイルの open・close ログのみを利用しているが，本研究では RMC 操作まで考慮し，共起情報では抽出できなかったタスクのマイニングや，タスク間関連度の算出を可能にしている．

3. 提案手法 SUGOI

提案手法では，従来のファイルアクセスログを用いてキーワードに関連するファイルを検出するアプローチでは考慮されてこなかった RMC 操作を利用して，タスクを抽出するとともに，タスク間の関連度を検索結果提示に利用する．そのために，SUGOI では，まずファイルアクセスログからタスクをマイニングし，タスク間の関連度を算出しておく．全文検索エンジンによってキーワードに関連するファイルに対して，あらかじめ算出しておいたタスク間関連度を用いたタスクのスコアリングを行って，関連ファイルを提示する．SUGOI を実現したシステムの構成を図 1 に示す．図に示すように，システムは，タスクとタスク間関連度を抽出する部分と，キーワードからファイルを検索する部分からなる．

本節において下記の処理について詳細を記述する．

（1） ログクリーニング (**3.1**)

（2） ファイル間共起及び RMC 操作を考慮したタスクマイニング (**3.2**)

（3） ファイルの重複度と RMC 操作を考慮したタスク間関連度の算出 (**3.3**)

（4） タスクとタスク間関連度を用いたファイル検索 (**3.4**)

なお，本手法適用の前提として，取得したファイルアクセスログには，アクセス時刻，クライアントユーザを特定するための情報，アクセスしたファイル，行われた操作といった項目を含むものとする．

3.1 ログクリーニング

ファイルに対するアクセスは，利用者の操作のほかに，ウイルス検査やアプリケーションの自動バックアップなどによる機械アクセスが多く存在する．それらのアクセスは作業とは直接関係がなく，利用者が意図しないものが多い．そこで，1 秒間におけるアクセス回数が TH_{sec} を超えた場合にその間のアクセスを全て機械アクセスとみなして除去する．それは，短期間において高頻度のアクセスは自動化されたプロセスによる大量同時アクセスによるものの可能性が高いためである．しかし，機械アクセスでも，処理の内容と対象ファイルのサイズによって比較的長く時間がかかることがあるため，本論文では TH_{sec} に加え，1 分間にお

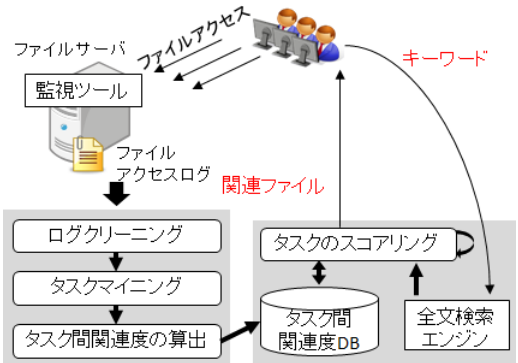

図 1 システムイメージ
Fig. 1 Overview of SUGOI.

けるアクセス回数が TH_{min} 超えた場合のアクセスも機械アクセスとして除去する．そのほかに拡張子を用いたフィルタリングも行っている．

3.2 タスクマイニング

既存のファイルアクセスログを用いたファイル検索手法では，ファイル間の関連度を算出しているだけで，同一作業に使われるファイルの塊に注目して，その塊の間の関連に着目したアプローチはなかった．それに対して，本論文のアプローチでは，同一作業に使用されたファイル群をタスクとして抽出し，タスク間の関連を利用している．ここでは，下記 2 種類のタスクを抽出する．タスクマイニングにより，ユーザがアクセスした多くのファイルは少なくとも一つのタスクに属することになる．

FI タスク： 頻出アイテム集合 (Frequent Itemset) であるタスク

RMC タスク： RMC 操作に基づいて抽出されるタスク

3.2.1 FI タスク

FI(Frequent Itemset) タスクは同一作業に関連するファイルは頻繁に近い時間にアクセスされるという傾向に着目しいる．例えば，論文執筆の際に tex ファイルのほかに，pdf ファイルやグラフのファイルに対して，近い時間にアクセスを繰り返すことがよくある．我々はその特性を注目し，頻繁に近い時間にアクセスされる複数のファイルを FI タスクとして抽出する．

図 2 は FI タスクの抽出手順を示している．本手法では，頻繁に近い時間にアクセスされたファイル群のみ抽出するため，まずファイルアクセスログを一定の時間幅 ($TransactionTime$) でトランザクション単位に分割する．次に，既存の頻出アイテム抽出用アルゴリズム Eclat [11] を適用し，トランザクションにおける出現回数が $MinSuppCnt$ 回以上の頻出ファイル集合をタスク (T) として抽出する．ここでは，長期間にわたるファイルアクセスログを解析対象としているため，個々のファイルに対するアクセスの回数に比べ，トランザクションの件数がはるかに大きく，$MinSuppCnt$ を低く設定する必要がある．最後に，他集合の部分集合でない集合のみを FI タスクとする．

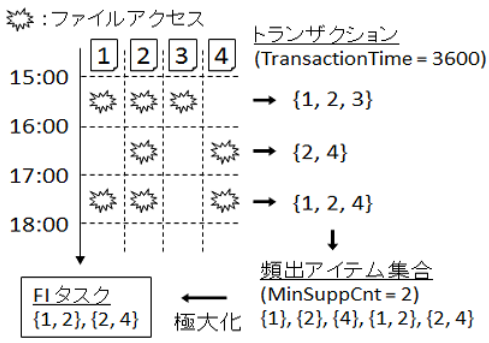

図 2 FI タスクの抽出
Fig. 2 Extraction of FI Task.

3.2.2 RMC タスク

過去の作業で使われた複数のファイルをコピーなどをして，他の作業で再利用することがよくある．そのため，複数のファイルに対して，短時間内にまとめて RMC 操作が行われた場合，それらのファイルは論理的に意味のあるまとまりであり，同一作業に由来するものである可能性が高い．そこで，本手法はこれまで考慮されていなかったこの特徴を利用して，一定時間 ($RMCTaskTime$) 以内に RMC された複数のファイルを RMC タスクとして抽出する（図 3）．

3.3 タスク間関連度の算出

SUGOI では，検索結果の再現率をより改善させるため，タスク間関連度を利用して，キーワードに対するタスクの得点を伝搬させることで，キーワードに関連するファイル群の対象を拡張する．このため，各タスクをノードとした際のノード間をつなぐエッジの重みであるタスク間関連度を算出する必要がある．

タスク間関連度は，タスク間関連の強さを数値化したもので，エッジに付与される重みでもある．タスク間関連度の算出においては，タスク同士が含むファイルの重複を考慮したタスク間類似度 $sim_t(T_m \rightarrow T_n)$ と，ファイル間 RMC 操作に着目したタスク間 RMC 関連度 $rmc_t(T_m \rightarrow T_n)$ を用いる．特に，タスク間 RMC 関連度は異なるタスクにまたがるファイル間の RMC 操作の種類や回数と，RMC 操作が発生してか

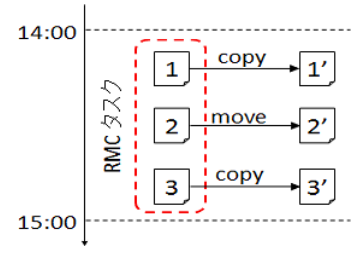

図 3 RMC タスクの抽出 ($RMCTaskTime = 3600\,s$)
Fig. 3 Extraction of an RMC Task.
($RMCTaskTime = 3600\,s$)

らの経過時間，編集回数，ファイルサイズの変化による関連度の減衰についても考慮している．式 (1) はタスク間関連度 $R(T_m \to T_n)$ の算出式である．

$$R(T_m \to T_n) = \theta * sim_t(T_m \to T_n) + (1-\theta) * rmc_t(T_m \to T_n) \quad (1)$$

ただし，θ はタスク間類似度とタスク間 RMC 関連度を考慮する度合を調節するためのパラメータで，$(0 \leq \theta \leq 1)$ の値をとる．

3.3.1 タスク間類似度

多くの共通ファイルを使った作業間の関係がより強いと推測できるため，各タスクに含まれるファイルの重複の度合に基づくタスク間類似度を算出する．タスク m，n 間のタスク間類似度 $sim_t(T_m \to T_n)$ は式 (2) によって定義される．ただし，T_m と T_n はタスク A とタスク B に含まれるファイルの集合を表す．

$$sim_t(T_m \to T_n) = \frac{|T_m \cap T_n|}{|T_m|} \quad (2)$$

3.3.2 タスク間 RMC 関連度

過去の作業で使用されたファイルをコピーなどをして，他の作業で再利用することがよくある．そのため，RMC 操作のあったタスク間はより強い関連をもつと考えられる．そこで，我々はまず式 (3) を定義し，ファイル f_i からファイル f_j へ RMC 操作が発生したことによって得られるファイル間 RMC 関連度を表す．ここでは，α_1，α_2，β_1，β_2，γ_1，γ_2 は全てパラメータである．

$$rmc_f(f_i \to f_j) = \begin{cases} \alpha_1 & \text{if } f_i \text{ was renamed to } f_j, \\ \alpha_2 & \text{if } f_i \text{ was renamed from } f_j, \\ \beta_1 & \text{if } f_i \text{ was moved to } f_j, \\ \beta_2 & \text{if } f_i \text{ was moved from } f_j, \\ \gamma_1 & \text{if } f_i \text{ was copied to } f_j, \\ \gamma_2 & \text{if } f_i \text{ was copied from } f_j, \\ 0 & \text{otherwise.} \end{cases} \quad (3)$$

このように定義されたファイル間 RMC 関連度は定数になる．しかし同じ RMC 操作でも，より最近に RMC 操作のあったファイル間の関連が強く，編集が重なることにより，RMC 操作のあったファイル内容に差分が大きくなることで，関連が弱くなる可能性があると考えられる．そこで，本手法では経過時間（式 (4)），編集回数（式 (5)），ファイルサイズの増減（式 (6)）に基づく減衰の度合を算出する．

$$T(f_i, f_j) = \Delta_{time}(f_i, f_j)^{-\tau} \quad (4)$$

$$E(f_i, f_j) = \Delta_{edit}(f_i, f_j)^{-\epsilon} \quad (5)$$

$$S(f_i, f_j) = \Delta_{size}(f_i, f_j)^{-\sigma} \quad (6)$$

ただし，$\Delta_{time}(f_i, f_j)$ はファイル f_i，f_j 間で RMC 操作が発生してから経過した時間で，τ は経過時間による影響の度合を調節するためのパラメータである．$\Delta_{edit}(f_i, f_j)$ はファイル f_i，f_j 間で RMC 操作が発生してから両ファイルに対する書込み操作の回数の和を表し，ϵ は編集回数による影響の度合を調節するためのパラメータである．$\Delta_{size}(f_i, f_j)$ はファイル f_i，f_j 間で RMC 操作が発生してから両ファイルのサイズの増加分と減少分の絶対値の和で，σ はファイルサイズの変化による影響の度合を調節するためのパラメータである．

ファイル間 RMC 関連度に加え，RMC 操作があってからの経過時間，編集回数，そしてファイルサイズの増減による関連度の減衰を考慮し，タスク間 RMC 関連度を下記の式 (7) によって算出する．

$$rmc_t(T_m \to T_n) = \sum_{(f_i, f_j) \in (T_m, T_n)} \{rmcf(f_i \to f_j) * T(f_i, f_j) * E(f_i, f_j) * S(f_i, f_j)\} \quad (7)$$

3.4 キーワード検索の実現

ファイルがもつテキスト情報に加え，タスクとタスク間関連度を用いることで，提案手法 SUGOI はキーワードを含まないファイルも検索可能となる．具体的には，あるクエリが与えられたときに．まず既存の全文検索手法を利用して，検索キーワードを含むタスクを特定してから，タスク間関連度を使ってタスクのキーワードに対する関連度（以降，タスクスコア）を K ホップ先まで伝搬させ，その都度タスクスコアの再計算を行う．そうすることにより，キーワードを含まないタスク内のファイルも検索可能となり，再現率の向上及び精度の高いランキングを実現する．以下では，SUGOI のキーワード検索における一連の処理について記述する．

STEP 1 全文検索より検索キーワード q を含むファ

イルを特定する．その際，ファイルがキーワードに対して付与されるスコアを $score_f(q, f_j)$ とし，そのスコアを利用してタスクスコアの初期値 $score_t^0(q, T_m)$ を算出する（式 (8)）．

$$score_t^0(q, T_m) = \sum_{f_j \in T_m} score_f(q, f_j) \tag{8}$$

STEP 2　タスク間関連度を用いて，キーワードに関連するタスクを検索する処理を行う．ここでは Connections や FRIDAL で用いられた関連度算出方法をベースに，タスク間関連度に基づき，全タスクに対して，タスクスコア $score_t^k(q, T_m)(1 \leq k \leq K)$ を伝搬させる処理を K 回繰り返す．$score_t^k(q, T_m)$ は式 (9) に従って算出される．

$$score_t^k(q, T_m) = score_t^{k-1}(q, T_m) + \sum_{T_n \in InLink(T_m)} score_t^{k-1}(q, T_n) * R(T_n \to T_m) \tag{9}$$

ただし，$InLink(T_m)$ は T_m に対するタスク関連度 $R(T_n \to T_m) > 0$ となる T_n の集合を表す．

STEP 3　タスクスコアを正規化し，$score_t^K(q, T_m) > TH_{score}$ を満たす全ての T_m をキーワードに対する検索結果として出力する．

4. 評価実験

4.1 実験環境

4.1.1 アクセスログの収集

SUGOI の有効性を検証するため，我々の研究グループで使用しているファイルサーバ（Windows Server 2003 SP2, NTFS）における実際のファイルアクセスログで実験を行う．ファイルアクセスをモニタリングするのに，FAccLog [12] というツールを利用している．FAccLog は OS と LAN アダプタ経由のアクセスを監視し，ファイルに対する参照，書込み，新規作成，削除，改名操作のログを検知することが可能である．FAccLog によって記録されたログにはアクセス時刻，クライアントのユーザ名，アクセスしたファイルのフルパス，ファイルに対する操作，ファイルサイズなどの情報が含まれる．RMC 操作のうち，改名と移動のログは「改名／移動前のパス ≫ 改名／移動後のパス」の形で区別されずに記録されるが，コピーログの記録はない．そこで，本論文では移動操作に関しては，ディレクトリの変更を伴う改名ログを移動とし，コピー操作に関しては，一連のファイルアクセスログの中，あるファイル f に対する新規作成ログの前に，他のディレクトリにある同名のファイル f' に対する参照ログが記録された場合，ファイル f を f' のコピーとする．

なお，本論文の実験で使用したアクセスログにおける各種操作の分布を図 4 に示す．RMC 操作ログの割合は全体の約 10%を占め，そのうちのほとんどがコピー操作であることが分かる．

4.1.2 全文検索

SUGOI の実装では，全文検索エンジンである Hyper Estraier [13] を使用している．実験では，プレーンテキスト，HTML のほかに，PDF，DOC，XLS，PPT，DOCX，XLSX，PPTX のファイルを検索対象としている．

4.2 評価方法

実験では，RMC 操作を用いたことによる効果を測定するため，評価に用いるキーワードと正解セットは，なるべく RMC 操作を行ったことのある作業が検索対象となるように，7 名の被験者によって作成された．評価では正解セットと比較して行っている．なお，実験を実施した時点で既に削除されたファイルは評価対象外とする．表 1 は実験用データセットの概要である．表 1 から分かるように，全文検索で検索が可能なファイル数（「全文」列）は全体の半分にも満たない場合が多い．なお，被験者 E, F, G に関しては，正解

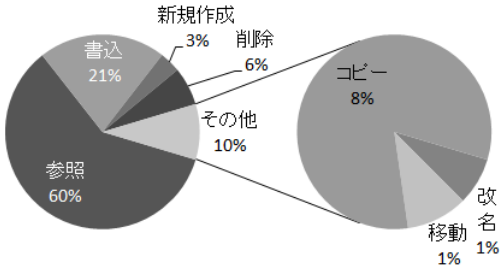

図 4　実験用アクセスログの分布
Fig. 4　Distribution of the Access Logs.

表 1　実験用データセット
Table 1　Experimental datasets.

被験者	ログ数	ファイル	全文	キーワード	正解数
A	3591	201	137	デスクトップ検索	70
B	2808	416	113	日本支部	25
C	3424	318	276	RAPoSDA	32
D	5911	764	311	XCP	84
E	8203	642	335	語学番組	244
F	5123	3422	1152	MTTDL	227
G	13102	3258	338	video slide keyword	73

表 2　FI タスクの実験結果（F 値）
Table 2　Average F-measure. (FI task mining)

TransactionTime [秒]	900	1800	3600	5400	7200
MinSuppCnt=2 [F 値]	0.543	0.513	0.507	0.470	0.506
MinSuppCnt=3 [F 値]	0.452	0.432	0.450	0.411	0.453

セットにある全文検索可能なファイルは全てインデックスの抽出を行ったが，それ以外のファイルの一部はプライバシーの理由でインデックスを抽出せず，全文検索の対象には含まれていない．それによって全文検索はより良い適合率を得られる可能性が高い．また被験者 E の実験では，システム開発作業のファイルが対象に含まれ，その結果プログラムのソースファイルが正解ファイルとなることで，正解数が多くなっている．

下記のパラメータは固定値に設定して実験を行った．ログクリーニングでは，$TH_{min} = 30$，$TH_{sec} = 5$ とし，関連ファイル検索で使用するスコアのしきい値を $TH_{score} = 0$ に設定した．また，スコア伝搬の繰返し回数 $K = 3$ とした．

4.3　タスクマイニングに関する実験

SUGOI では，キーワード検索を拡張するためにアクセスログに基づき，2 種類のタスクを抽出した後，タスク間関連度を算出している．そのため，まずタスクマイニング用パラメータ *TransactionTime*, *MinSuppCnt*, *RMCTaskTime* の値を決めるために実験を行う．なお，タスク間関連度算出用パラメータは下記の値に設定した．$\theta = 0.5$, $(\alpha_1, \alpha_2, \beta_1, \beta_2, \gamma_1, \gamma_2) = (1.0, 1.0, 1.0, 1.0, 1.0, 1.0)$, $(\tau, \epsilon, \sigma) = (0.0, 0.0, 0.0)$.

4.3.1　FI タスク抽出用パラメータの選定

FI タスクの抽出に用いる *TransactionTime*, *MinSuppCnt* の値を決めるため，FI タスクのみを検索対象とし，$TransactionTime = \{900, 1800, 3600, 5400, 7200\}$ [秒], $MinSuppCnt = \{2, 3\}$ の組合せで実験を行った．

被験者全員の F 値の平均を表 2 に示すように，*TransactionTime* が同じの場合，$MinSuppCnt = 2$ の方が $MinSuppCnt = 3$ より良い性能を示している．それは *MinSuppCnt* を上げることで，タスクに属さないファイル数が増え，タスク間関連度ではたどり着けないファイルが多くなったことが原因だと考えられる．また *MinSuppCnt* に比べ，*TransactionTime* が F 値に与える影響は少ないことが分かる．

図 5 は $MinSuppCnt = 2$ 時の被験者別実験結果である．*TransactionTime* の増大による F 値の変化は被験者ごとに異なる傾向を示した．それは *TransactionTime* を小さい値に設定した際，*MinSuppCnt* 回以上出現するタスクが多くなり，FI タスクが多く抽出される一方，同じ作業に使用されるファイル群内の要素が組合せとして複数のトランザクションに出現する可能性が小さくなり，得られる FI タスクのサイズが小さくなるためである．それによってタスク間関連度が抽出されにくくなり，再現率の低下する可能性が増大する．また，*TransactionTime* が増えることにより，ファイルの使用時間に多少の間隔が開いても同じトランザクションに入ることができ，他のファイルと関連づけやすくなるが，*TransactionTime* 内における複数回のアクセスが 1 度としかカウントされないため，*MinSuppCnt* は満たされにくくなる．このことから，本手法適応の際には利用者の作業パターンに応じて *TransactionTime* のチューニングが必要であることが分かった．なお，以降の実験では被験者 A, B, E については $TransactionTime = 3600$ を設定し，被験者 C, D, F, G については $TransactionTime = 900$ を用いて測定を行うことにした．

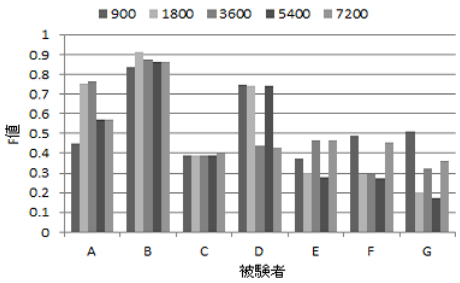

図 5　*TransactionTime* に関する被験者別実験結果 ($MinSuppCnt = 2$)
Fig. 5　Detailed Results for *TransactionTime*. ($MinSuppCnt = 2$)

4.3.2　RMC タスク抽出用パラメータの選定

RMC タスクの抽出に用いる *RMCTaskTime* の値を決めるために FI タスクに加え，$RMCTaskTime = \{60, 180, 300, 600, 1800\}$[秒] のときに抽出された RMC タスクを用いて検索を行った．

表 3 で実験結果を示しているように，*RMCTaskTime* の増加によって再現率が上昇していくものの，適合率がわずかに低下した．RMC タスクを抽出することで関係のないファイル同士が近い時間に RMC されたことにより，同一作業に関連するも

表 3　RMC タスクの実験
Table 3　Average F-measure. (RMC task mining)

RMCTaskTime [秒]	60	180	300	600	1800
適合率	0.776	0.758	0.762	0.762	0.756
再現率	0.669	0.674	0.673	0.675	0.684
F 値	0.684	0.681	0.679	0.684	0.684

表 4　ファイル間 RMC 関連度に関する実験の構成
Table 4　Setups for comparison of RMC operations.

	改名		移動		コピー	
実験構成	α_1	α_2	β_1	β_2	γ_1	γ_2
RMC なし	0	0	0	0	0	0
改名	1	1	0	0	0	0
移動	0	0	1	1	0	0
コピー	0	0	0	0	1	1
RMC 考慮	1	1	1	1	1	1

のとされてしまうことによる適合率の低下が原因である．しかし，RMC 操作は頻繁に行われていないため，その影響が小さくて F 値の変化が少ない．なお，以降の実験では $RMCTaskTime = 60$ とする．

4.4　タスク間関連度に関する実験

タスク関連度の算出において多くのパラメータを使用している．各パラメータの特性を調べるために実験を行った．

4.4.1　タスク間 RMC 関連度に関する実験

タスク間関連度の算出に用いるタスク間 RMC 関連度（式 (7)）は RMC 操作の種類と向きを考慮したファイル間 RMC 関連度を用いている．ここでは，タスク間 RMC 関連度の算出に関わるパラメータの適切な値を調べるために実験を行う．

ファイル間 RMC 関連度は式 (3) によって定義されており，ファイル間 RMC 操作の種類と向きを区別するために α_1，α_2，β_1，β_2，γ_1，γ_2 の六つのパラメータを使用している．実験では，$(\alpha_1, \alpha_2, \beta_1, \beta_2, \gamma_1, \gamma_2)$ を表 4 の構成で行った．

実験結果を図 6，被験者別の結果を表 5〜表 7 にて示している．今回実験で使用したデータセットでは RMC のうち改名の効果が見られなかった．また，移動の効果が見られたのは，被験者 F だけだった．改名，移動したファイルの元ファイルは評価対象でないことや（ファイルシステムに存在しないため），ファイルに対する改名と移動操作が少なかったことがその一因であると考える．また，被験者 B と E はコピー操作を考慮したことで再現率と F 値の改善につながった．

4.4.2　θ に関する実験

タスク間関連度を算出する際に，タスク間類似度とタスク間 RMC 関連度を使用し，パラメータ θ で重要視する指標を調節している．θ による検索結果の変化を調べるために，$\theta = \{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\}$ に設定したときの 11 点平均適合率の平均値で評価を行った．ただし，その他のパラメータは $(\alpha_1, \alpha_2, \beta_1, \beta_2, \gamma_1, \gamma_2) =$

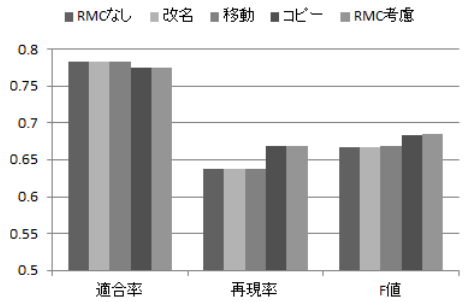

図 6　ファイル間 RMC 関連度に関する実験結果
Fig. 6　Comparison of experimental results on RMC links.

表 5　ファイル間 RMC 関連度に関する被験者別実験結果（適合率）
Table 5　Detailed results on RMC links. (Precision)

被験者	RMC なし	改名	移動	コピー	RMC 考慮
A	0.938	0.938	0.938	0.938	0.938
B	0.920	0.920	0.920	0.923	0.923
C	0.818	0.818	0.818	0.818	0.818
D	0.397	0.397	0.397	0.397	0.397
E	0.685	0.685	0.685	0.646	0.646
F	0.856	0.856	0.857	0.856	0.857
G	0.875	0.875	0.875	0.854	0.854

表 6　ファイル間 RMC 関連度に関する被験者別実験結果（再現率）
Table 6　Detailed results on RMC links. (Recall)

被験者	RMC なし	改名	移動	コピー	RMC 考慮
A	0.643	0.643	0.643	0.643	0.643
B	0.920	0.920	0.920	0.960	0.960
C	0.844	0.844	0.844	0.844	0.844
D	0.821	0.821	0.821	0.821	0.821
E	0.418	0.418	0.418	0.590	0.590
F	0.339	0.339	0.344	0.339	0.344
G	0.479	0.479	0.479	0.479	0.479

表 7　ファイル間 RMC 関連度に関する被験者別実験結果（F 値）
Table 7　Detailed results on RMC links. (F-measure)

被験者	RMC なし	改名	移動	コピー	RMC 考慮
A	0.763	0.763	0.763	0.763	0.763
B	0.920	0.920	0.920	0.941	0.941
C	0.831	0.831	0.831	0.831	0.831
D	0.535	0.535	0.535	0.535	0.535
E	0.519	0.519	0.519	0.617	0.617
F	0.486	0.486	0.491	0.486	0.491
G	0.619	0.619	0.619	0.614	0.614

表 8　θ による 11 点平均適合率の推移
Table 8　11-point average precision.

θ	0.0	0.2	0.4	0.6	0.8	1.0
11 点平均適合率の平均値	0.430	**0.482**	**0.482**	0.476	0.476	0.475

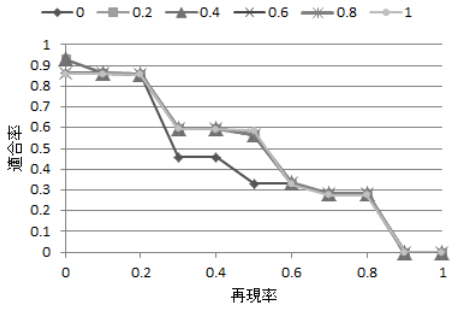

図 7　θ に関する実験結果
Fig. 7　Experimental results for θ.

表 9　被験者別 11 点平均適合率の推移（被験者 A）
Table 9　11-point average precision. (Tester: A)

再現率	$\theta = 0.0$	$\theta = 0.2$	$\theta = 0.4$	$\theta = 0.6$	$\theta = 0.8$	$\theta = 1.0$
0	1.000	1.000	1.000	1.000	1.000	1.000
0.1	1.000	1.000	1.000	1.000	1.000	1.000
0.2	0.944	0.975	0.975	0.975	0.975	0.975
0.3	0.000	0.975	0.975	0.975	0.975	0.975
0.4	0.000	0.975	0.975	0.975	0.975	0.975
0.5	0.000	0.975	0.975	0.975	0.975	0.975
0.6	0.000	0.000	0.000	0.000	0.000	0.000
0.7	0.000	0.000	0.000	0.000	0.000	0.000
0.8	0.000	0.000	0.000	0.000	0.000	0.000
0.9	0.000	0.000	0.000	0.000	0.000	0.000
1	0.000	0.000	0.000	0.000	0.000	0.000

表 10　被験者別 11 点平均適合率の推移（被験者 B）
Table 10　11-point average precision. (Tester: B)

再現率	$\theta = 0.0$	$\theta = 0.2$	$\theta = 0.4$	$\theta = 0.6$	$\theta = 0.8$	$\theta = 1.0$
0	1.000	1.000	1.000	1.000	1.000	1.000
0.1	1.000	1.000	1.000	1.000	1.000	1.000
0.2	1.000	1.000	1.000	1.000	1.000	1.000
0.3	1.000	1.000	1.000	1.000	1.000	1.000
0.4	1.000	1.000	1.000	1.000	1.000	1.000
0.5	1.000	1.000	1.000	1.000	1.000	1.000
0.6	1.000	1.000	1.000	1.000	1.000	1.000
0.7	1.000	1.000	1.000	1.000	1.000	1.000
0.8	1.000	1.000	1.000	1.000	1.000	1.000
0.9	0.000	0.000	0.000	0.000	0.000	0.000
1	0.000	0.000	0.000	0.000	0.000	0.000

$(1.0, 1.0, 1.0, 1.0, 1.0, 1.0)$，$(\tau, \epsilon, \sigma) = (0.0, 0.0, 0.0)$ に設定した．

全体の実験結果を図 7 に，また被験者別の実験結果を表 9〜表 15 に示している．図 7 から分かるように，$\theta = 0.0$ 以外の場合におけるランキング結果が優れているが，θ による影響が少なかった．それは提案手法で

表 11　被験者別 11 点平均適合率の推移（被験者 C）
Table 11　11-point average precision. (Tester: C)

再現率	$\theta = 0.0$	$\theta = 0.2$	$\theta = 0.4$	$\theta = 0.6$	$\theta = 0.8$	$\theta = 1.0$
0	0.963	0.963	0.963	0.963	0.963	0.929
0.1	0.963	0.963	0.963	0.963	0.963	0.929
0.2	0.963	0.963	0.963	0.963	0.963	0.929
0.3	0.963	0.963	0.963	0.963	0.963	0.929
0.4	0.963	0.963	0.963	0.963	0.963	0.929
0.5	0.963	0.963	0.963	0.963	0.963	0.929
0.6	0.963	0.963	0.963	0.963	0.963	0.929
0.7	0.963	0.963	0.963	0.963	0.963	0.929
0.8	0.963	0.963	0.963	0.963	0.963	0.929
0.9	0.000	0.000	0.000	0.000	0.000	0.000
1	0.000	0.000	0.000	0.000	0.000	0.000

表 12　被験者別 11 点平均適合率の推移（被験者 D）
Table 12　11-point average precision. (Tester: D)

再現率	$\theta = 0.0$	$\theta = 0.2$	$\theta = 0.4$	$\theta = 0.6$	$\theta = 0.8$	$\theta = 1.0$
0	0.535	1.000	1.000	0.539	0.539	0.545
0.1	0.535	0.539	0.539	0.539	0.539	0.545
0.2	0.535	0.539	0.539	0.539	0.539	0.545
0.3	0.535	0.539	0.539	0.539	0.539	0.545
0.4	0.535	0.539	0.539	0.539	0.539	0.545
0.5	0.355	0.357	0.361	0.364	0.364	0.545
0.6	0.355	0.357	0.361	0.364	0.364	0.364
0.7	0.000	0.000	0.000	0.000	0.000	0.000
0.8	0.000	0.000	0.000	0.000	0.000	0.000
0.9	0.000	0.000	0.000	0.000	0.000	0.000
1	0.000	0.000	0.000	0.000	0.000	0.000

表 13　被験者別 11 点平均適合率の推移（被験者 E）
Table 13　11-point average precision. (Tester: E)

再現率	$\theta = 0.0$	$\theta = 0.2$	$\theta = 0.4$	$\theta = 0.6$	$\theta = 0.8$	$\theta = 1.0$
0	0.710	0.704	0.708	0.708	0.705	0.706
0.1	0.710	0.704	0.708	0.708	0.705	0.706
0.2	0.710	0.704	0.708	0.708	0.705	0.706
0.3	0.710	0.704	0.708	0.708	0.705	0.706
0.4	0.710	0.695	0.695	0.695	0.695	0.695
0.5	0.000	0.660	0.660	0.660	0.660	0.660
0.6	0.000	0.000	0.000	0.000	0.000	0.000
0.7	0.000	0.000	0.000	0.000	0.000	0.000
0.8	0.000	0.000	0.000	0.000	0.000	0.000
0.9	0.000	0.000	0.000	0.000	0.000	0.000
1	0.000	0.000	0.000	0.000	0.000	0.000

は他のタスクと密な関連にあるタスクが上位にランキングされやすいため，タスク間関連度が多少変動してもその性質により多くのスコアが付くためだと思われる．また，一部の被験者（被験者 D）によっては，タスク間 RMC 関連度を考慮しない場合（$\theta = 1.0$）でも良い結果が出ているケースが見られた．作業とは関係なくコピー操作が行われた場合に，タスク間 RMC 関連度を用いたことで適合率が低下することが分かった．

今回の実験では，$\theta = \{0.2, 0.4\}$ で最も高い 11 点平均適合率の平均値を得られた（表 8）．$\theta = 0.0$ にお

表 14　被験者別 11 点平均適合率の推移（被験者 F）
Table 14　11-point average precision. (Tester: F)

再現率	$\theta = 0.0$	$\theta = 0.2$	$\theta = 0.4$	$\theta = 0.6$	$\theta = 0.8$	$\theta = 1.0$
0	1.000	1.000	1.000	0.981	0.981	0.981
0.1	1.000	1.000	0.981	0.981	0.981	0.981
0.2	0.981	0.981	0.981	0.981	0.981	0.981
0.3	0.000	0.000	0.000	0.000	0.000	0.000
0.4	0.000	0.000	0.000	0.000	0.000	0.000
0.5	0.000	0.000	0.000	0.000	0.000	0.000
0.6	0.000	0.000	0.000	0.000	0.000	0.000
0.7	0.000	0.000	0.000	0.000	0.000	0.000
0.8	0.000	0.000	0.000	0.000	0.000	0.000
0.9	0.000	0.000	0.000	0.000	0.000	0.000
1	0.000	0.000	0.000	0.000	0.000	0.000

表 15　被験者別 11 点平均適合率の推移（被験者 G）
Table 15　11-point average precision. (Tester: G)

再現率	$\theta = 0.0$	$\theta = 0.2$	$\theta = 0.4$	$\theta = 0.6$	$\theta = 0.8$	$\theta = 1.0$
0	0.850	0.850	0.850	0.850	0.850	0.850
0.1	0.850	0.850	0.850	0.850	0.850	0.850
0.2	0.850	0.850	0.850	0.850	0.850	0.850
0.3	0.000	0.000	0.000	0.000	0.000	0.000
0.4	0.000	0.000	0.000	0.000	0.000	0.000
0.5	0.000	0.000	0.000	0.000	0.000	0.000
0.6	0.000	0.000	0.000	0.000	0.000	0.000
0.7	0.000	0.000	0.000	0.000	0.000	0.000
0.8	0.000	0.000	0.000	0.000	0.000	0.000
0.9	0.000	0.000	0.000	0.000	0.000	0.000
1	0.000	0.000	0.000	0.000	0.000	0.000

ける性能が低かった原因として RMC 操作が少なかったため，タスク間 RMC 関連度だけ使用しては十分にタスク間を関連付けることが困難であることが挙げられる．

4.5　キーワード検索に関する実験

本論文ではタスクマイニングとタスク関連度の算出において RMC 操作を考慮している．タスクマイニングでは，頻出ファイル集合のほかに近い時間に RMC されたファイルをタスクとして抽出する．また，タスク間関連度の算出では，タスク間類似度のほかに RMC 操作を考慮したタスク間 RMC 関連度を用いている．RMC 操作を考慮したことによる結果の改善を確認するため，全文検索のみの場合のほかに，検索に用いるタスクの種類とタスク関連度を組み合わせた表 16 で示す四つの構成での検索結果を用いて比較を行った．ただし，タスク間 RMC 関連度を用いる場合のパラメータは $\theta = 0.5$ に設定し，タスク間類似度のみ用いる場合では $\theta = 1.0$ とした．

実験結果は図 8，被験者別実験結果は表 17〜表 19 になっている．図 8 では提案手法の全ての構成において全文検索のみ使用した場合に比べ，再現率及び F 値の向上が確認された．その理由として，ファイルアクセスログを考慮することで，キーワードを含まないファイルが，キーワードを含むファイルと同じタスクまたはその関連するタスクに入っていたことが挙げられる．

FI タスクのみ使用した SUGOI 構成 1 と構成 2 では，適合率の改善も見られた．FI タスクは頻繁にア

表 16　実 験 構 成
Table 16　Setup of SUGOI.

実験構成	タスクマイニング	タスク間関連度
SUGOI 構成 1	FI タスクのみ使用	類似度のみ使用
SUGOI 構成 2	FI タスクのみ使用	類似度＋ RMC
SUGOI 構成 3	FI タスク＋ RMC タスク	類似度のみ使用
SUGOI 構成 4	FI タスク＋ RMC タスク	類似度＋ RMC

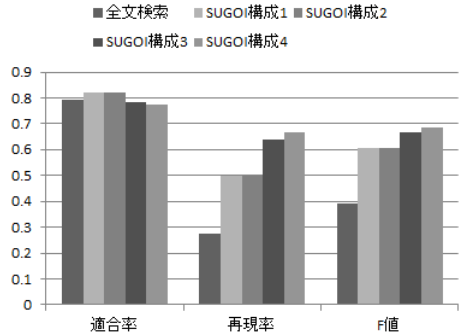

図 8　RMC 操作を考慮した効果の検証実験
Fig. 8　Experimental results of SUGOI.

表 17　RMC 操作を考慮した効果の検証実験の被験者別結果（適合率）
Table 17　Detailed results of SUGOI. (Precision)

被験者	全文	構成 1	構成 2	構成 3	構成 4
A	0.905	0.938	0.938	0.938	0.938
B	1.000	0.913	0.913	0.920	0.923
C	0.750	0.643	0.643	0.818	0.818
D	0.743	0.838	0.838	0.397	0.397
E	0.472	0.654	0.654	0.685	0.646
F	0.769	0.856	0.856	0.856	0.857
G	0.923	0.897	0.897	0.875	0.854

表 18　RMC 操作を考慮した効果の検証実験の被験者別結果（再現率）
Table 18　Detailed results of SUGOI. (Recall)

被験者	全文	構成 1	構成 2	構成 3	構成 4
A	0.271	0.643	0.643	0.643	0.643
B	0.560	0.840	0.840	0.920	0.960
C	0.281	0.281	0.281	0.844	0.844
D	0.310	0.679	0.679	0.821	0.821
E	0.070	0.365	0.365	0.418	0.590
F	0.088	0.339	0.339	0.339	0.344
G	0.329	0.356	0.356	0.479	0.479

表 19 RMC 操作を考慮した効果の検証実験の被験者別結果（F 値）
Table 19 Detailed results of SUGOI. (F-measure)

被験者	全文	構成 1	構成 2	構成 3	構成 4
A	0.418	0.763	0.763	0.763	0.763
B	0.718	0.875	0.875	0.920	0.941
C	0.409	0.391	0.391	0.831	0.831
D	0.437	0.750	0.750	0.535	0.535
E	0.121	0.468	0.468	0.519	0.617
F	0.158	0.486	0.486	0.486	0.491
G	0.485	0.510	0.510	0.619	0.614

クセスされたファイルの組合せからなっており，そのサイズが小さく，平均は 3.0 未満になっている．そのため，無関連のファイルが非常に少ない．それに対して，RMC タスクでは近い時間に RMC されたファイル群からなっているため，人手によるバックアップ操作やファイル整理などの偶然の共起によって無関係のファイルが混入しやすい．ただし，今回実験に使用したデータセットでは作業に無関係な RMC 操作が少なかったため，全文検索のみの場合に比べ，適合率がわずか 0.02（構成 4）低下した．

適合率の変化がわずかであったことに対して，再現率の改善幅が 0.3 以上と大きく，F 値の上昇につながり，RMC 操作を考慮してタスクを抽出する効果が見られた（SUGOI 構成 3，構成 4）．また，タスク間 RMC 関連度を用いた SUGOI 構成 2 と構成 4 を比較すると，RMC タスクを利用している場合のみ，タスク間 RMC 関連度の効果が見られたことが分かる

タスクマイニング及びタスク間関連度の算出で RMC 操作を考慮した SUGOI 構成 4 では，再現率と F 値が最も高い値となった．その理由は RMC 操作を考慮したことにより，キーワードを含むファイルがなくて類似度だけでは見つからなかったタスクにもスコアが配分されたと考えられ，提案手法の有効性を示した．

4.6 実験のまとめ

4. において被験者実験により，提案手法 SUGOI の有効性を確認し，パラメータの特性を調べ，以下の結論が得られた．

- タスクマイニングに関する実験では，$MinSuppCnt = 2$ における F 値が $MinSuppCnt = 3$ のときより高く，トランザクションにおける共起回数が 2 回以上あれば，関連するファイル同士である可能性が高いことが分かった．

また，RMC タスク抽出用の $RMCTaskTime$ を増やすことで適合率が低下し，再現率の増加傾向が見られたが，今回使用したデータセットにおいて RMC 操作が頻繁に行われていなかったためその影響は小さい．

- タスク間 RMC 関連度に関する実験では，改名，移動，コピーの 3 操作のうち，コピー操作が最も有効であることが分かった．元のファイルに変更を加えることなく，他の作業で使用するという作業パターンが多いことが推測できる．

式 (1) において，タスク間類似度とタスク間 RMC 関連度を考慮する度合を調節するためのパラメータ θ に関する実験では，$\theta = \{0.2, 0.4\}$ におけるランキングの精度が最も良かった結果となった．

- RMC 操作を考慮した効果を検証するための実験では，RMC タスク及びタスク間 RMC 関連度を利用した構成が既存の全文検索エンジンに比べ，F 値が 0.292 上昇し，RMC 操作を考慮した提案手法 SUGOI の有効性を示した．

5. むすび

ファイルシステム上のデータ量が急速に増加しており，その多くは非構造化ファイルで，全文検索だけでは検索できない．本論文では，このようなファイルも検索可能にするため，タスクとタスク間関連度の概念を取り入れた検索手法 SUGOI を提案した．

我々のアプローチは，アクセスログ中のファイル操作の発生頻度だけでなく，ファイル操作の内容まで考慮し，同一作業に関連するファイル群をタスクとして抽出し，更にタスク間関連度の情報を利用したファイル検索を実現している．タスクの抽出においては，アクセス共起を考慮した FI タスクと，ファイルの改名・移動・コピー（RMC）操作を利用した RMC タスクの 2 種類を抽出している．タスク間関連度の算出においては，RMC 操作を考慮したタスク間関連度の算出式を提案した．被験者実験では，既存の全文検索に，タスクとタスク間関連度の情報を組み合わせたことで，実際に RMC 操作に起因する関連ファイルを探し出すことができ，適合率，再現率，F 値が大きく改善されたことを確認した．

今後の課題として，長期間にわたるアクセスログを用いて評価実験を行い，ユーザのアクセスパターンに合ったパラメータの自動設定方法を検討していきたい．また，タスク間関連度の算出式及びそれを利用したスコアリング手法の改善や，タスクマイニングとタスク間関連度の情報を生かした検索結果の適切な提示方法などを考えていきたい．

謝辞　本研究の一部は，日本学術振興会科学研究費補助金基盤研究 (A)(#22240005) 及び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．

文　　献

[1] Google, “Google デスクトップ”. http://desktop.google.com
[2] Microsoft Corporation, “Windows デスクトップサーチ”. http://www.microsoft.com/japan/windows/desktopsearch/default.mspx
[3] Apple Inc., “Spotlight”. http://www.apple.com/jp/macosx/what-is-macosx/spotlight.html
[4] Beagle Team, “Beagle”. http://beagle-project.org/
[5] C.A.N. Soules and G.R. Ganger, “Connections: using context to enhance file search,” SIGOPS Oper. Syst. Rev., vol.39, no.5, pp.119–132, 2005.
[6] 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,” 第 19 回データ工学ワークショップ (DEWS2008), pp.E10–6, 2008.
[7] T. Watanabe, T. Kobayashi, and H. Yokota, “A method for searching keyword-lacking files based on interfile relationships,” OTM ’08, pp.14–15, Springer-Verlag, Berlin, Heidelberg, 2008.
[8] Y. Wu, K. Otagiri, Y. Watanabe, and H. Yokota, “A file search method based on intertask relationships derived from access frequency and rmc operations on files,” Proc. 22nd International Conference on Database and Expert Systems Applications - Volume Part I, pp.364–378, DEXA’11, Springer-Verlag, 2011.
[9] J. Chen, H. Guo, W. Wu, and W. Wang, “iMecho: An associative memory based desktop search system,” CIKM ’09: Proc. 18th ACM Conference on Information and Knowledge Management, pp.731–740, ACM, 2009.
[10] 小田切健一，渡辺陽介，横田治夫，“頻出ファイル集合のアクセス時間を考慮した仮想ディレクトリ生成手法,” 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 (DEIM2010)，pp.F9–2, 2010.
[11] M.J. Zaki, S. Parthasarathy, M. Ogihara, and W. Li, “New algorithms for fast discovery of association rules,” KDD-97 Proc., pp.283–286, 1997.
[12] だいこくネット，“FAccLog”. http://www2s.biglobe.ne.jp/~masa-nak/
[13] M. Hirabayashi, “Hyper estraier”. http://fallabs.com/hyperestraier/

（平成 24 年 7 月 5 日受付，10 月 29 日再受付）

呉　　怡

平 21 東工大・工学開発卒．平 23 同大大学院情報理工学研究科計算工学専攻修士課程了．同年（株）エヌ・ティ・ティ・データ入社．情報処理学会，日本データベース学会各会員．

渡辺　陽介

平 18 筑波大学大学院システム情報工学研究科コンピュータサイエンス専攻了．博士（工学）．平 18 科学技術振興機構戦略的創造研究推進事業 (CREST) 研究員．平 20 東京工業大学学術国際情報センター助教．主な研究分野は情報統合，データストリーム処理，デスクトップ検索技術など．ACM，情報処理学会，日本データベース学会各会員．

横田　治夫　（正員：フェロー）

昭 55 東工大・工卒．昭 57 同大大学院・情工・修士課程了．同年富士通（株）．同年 6 月（財）新世代コンピュータ技術開発機構研究所 (ICOT)．昭 61（株）富士通研究所．平 4 北陸先端大・情報・助教授．平 10 東工大・大学院情理工・助教授．平 13 東工大・学術国際情報センター・教授，平 22 東工大・大学院情理工・教授，現在に至る．工博．主として分散インデキシング，データ工学向けアーキテクチャ，高機能ストレージシステム，ディペンダブルシステム等に関する研究に従事．元信学会データ工学研究専門委員長．元 ACM SIGMOD 日本支部長．情報処理学会英文論文誌 (JIP) 編集委員長，VLDB Journal 編集委員，日本データベース学会理事・副会長．情報処理学会フェロー．IEEE シニア会員，人工知能学会，ACM 各会員．