

作業内容別に分類された仮想的なフォルダに フォルダ名を付与する手法の検討

野村 優文¹ 乃村 能成²

概要：計算機内のファイル操作履歴を利用して使用時期や作業内容ごとに再分類し、1つの仮想的なフォルダとして提示するシステムが提案されている。ここで、仮想フォルダにその内容をユーザが想起しやすい名前を付与する必要がある。本稿では、仮想フォルダに含まれるファイルの拡張子を用いてフォルダの特徴を抽出し、そこから仮想フォルダの名前を自動的に付与することを試みる。ここで使用するファイルの拡張子について、仮想フォルダの中に含まれるファイルの静的な情報からファイルの拡張子を抽出する方法と、一定の期間の間にアクセスされたファイルのみを候補として抽出する方法がある。このうち、どちらのファイルの拡張子を使用した方がフォルダ名の付与の精度が良いか検討するため、評価を行った。この結果、フォルダ名の付与の際、一定の期間の間にアクセスされたファイルのみを候補として抽出したファイルの拡張子を使用した方が精度が良いことがわかった。

キーワード：ワーキングディレクトリ、仮想フォルダ、フォルダ名推定、ファイル拡張子

1. はじめに

計算機を利用して作業を行う際は、以前の作業内容の確認やファイルの再利用を目的に過去のファイルを参照することが多い。また、文書作成やプログラミングなど、計算機内にはユーザが作業を行ったフォルダ（以下、ワーキングディレクトリ）が存在する。このワーキングディレクトリは年度や作業、および目的等さまざまな観点で、さらに上位のフォルダに整理されている。また、過去のファイルを参照する際は、使用時期ごとや作業内容ごとといった様々な観点で見たいという要求がある。そこで、計算機内のワーキングディレクトリを推定し、使用時期や作業内容ごとに再分類した仮想フォルダを生成して、ユーザに提示するシステム（以下、仮想フォルダ生成システム）が提案されている [1]。このシステムでは、使用時期や作業内容といった特定の観点でワーキングディレクトリを仮想フォルダと呼ばれるフォルダに保存する。仮想フォルダ生成システムは仮想フォルダを生成し、ユーザに提示することで、フォルダの再参照を支援できる。

ここで、仮想フォルダには、どのようなワーキングディレクトリが保存されているかをユーザが想起しやすくするため、適切なフォルダ名を付与する必要がある。たとえば、

TeX を用いて原稿を執筆する際、図や表を作成するための書式がわからないといった場合がある。この場合、ユーザは過去に TeX を用いて作成した原稿を参照することで、これらの書式を参考にして原稿に反映する方法が考えられる。そこで、仮想フォルダには「2019 年に TeX で書いた原稿」といったフォルダ名が付与されていると、どのようなファイルが保存されているかわかりやすい。あるいは、Python のプログラム書式を過去の Python プログラムから思い出したいような場合も有効である。このような目的を達成するフォルダ名を付与するために、仮想フォルダに含まれるファイルの拡張子を用いてフォルダ名の一部にその情報を付与することが有効であると考えた。そこで本稿では、作業内容ごとに分類した仮想フォルダ配下にあるファイルの拡張子を用いてフォルダ名を代表するファイル拡張子を抽出する手法を提案する。

以降では、2 章にワーキングディレクトリを分類し、仮想フォルダを生成する方法について述べる。次に、3 章ではフォルダ名を代表するファイル拡張子の抽出手法について述べる。最後に、4 章で提案手法がどの程度の精度でフォルダ名の付与にふさわしい拡張子を抽出できるかについて、評価を述べる。

¹ 岡山大学大学院自然科学研究科

² 岡山大学学術研究院自然科学学域

2. ワーキングディレクトリの分類

2.1 作業内容による分類

ワーキングディレクトリを作業内容ごとに分類するにはクラスタリングすることが考えられる。ワーキングディレクトリで行われた作業を表現する特徴として、ワーキングディレクトリ内のファイル種別やファイル更新の時系列が挙げられる。これらの特徴に着目した、フォルダのクラスタリング手法 [2] について、以降で詳細を説明する。

2.2 フォルダの作業を表現する特徴

フォルダをクラスタリングするためには、フォルダで行われた作業を表現する特徴が必要となる。フォルダの作業を表現する特徴として、フォルダ内のファイル種別、フォルダの階層構造の形状、ファイルの名称、およびファイルアクセス履歴等が挙げられる。本稿では、これらの特徴の中から以下の2つを用いる。

(特徴 1) ファイル種別

(特徴 2) ファイル更新の時系列

(特徴 1) のファイル種別は、ファイルの拡張子により識別する。フォルダ内に存在するファイル種別は、そのフォルダでユーザが行った作業を表していると考えられる。たとえば、「tex」や「pdf」といった拡張子を持つファイルが存在するフォルダは、「 $\text{\TeX}$ による文書作成」を行ったフォルダである可能性が高い。また、(特徴 2) について、同様の作業のフォルダでは、ファイル更新の時系列に類似性が存在すると考える。たとえば、「 $\text{\TeX}$ による文書作成」という作業では、ユーザはソースファイルを編集し、それをコンパイルすることで PDF の文書を生成する。このため、「tex」ファイルが更新された後「pdf」ファイルが更新されるという更新順序が頻出する可能性が高い。

次項にて、(特徴 1) と (特徴 2) を表現する特徴ベクトルの作成方法を述べる。

2.3 特徴ベクトルの作成

ファイル種別とファイル更新の時系列を用いて、フォルダの作業を表現する特徴ベクトルを作成する方法を図 1 および表 1 に示し、手順を以下に説明する。

(手順 1) 対象のフォルダのファイルアクセス履歴を抽出
ワーキングディレクトリ推定手法において作業ごとに分割されたファイルアクセス履歴を使用する。図 1 にアクセス履歴の例を示す。

(手順 2) 隣接するファイルアクセス履歴の集計

(手順 1) により得られた履歴に対して、隣接する更新履歴のファイル拡張子の組を集計する。図 1 の集計結果は、表 1 となる。そして、履歴 1 と履歴 2 について集計した結果を合計し、特徴ベクトルを作成する。

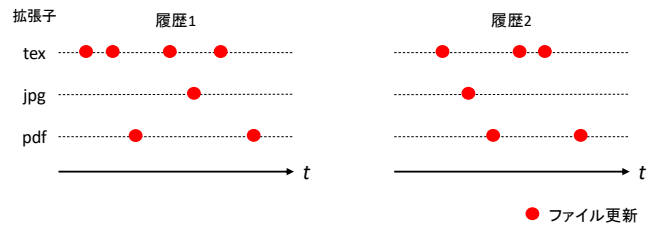


図 1 ワーキングディレクトリ推定手法で作業ごとに分割されたファイル更新履歴

表 1 隣接する更新履歴のファイル拡張子の組を集計した結果

	tex-tex	tex-pdf	pdf-tex	tex-jpg	jpg-tex	jpg-pdf
履歴 1	1	2	1	1	1	0
履歴 2	1	1	1	1	0	1
合計	2	3	2	2	1	1

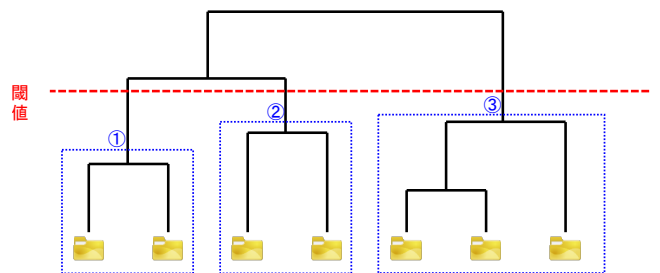


図 2 階層的クラスタリングによって得られるデンドログラム

また、フォルダのファイルアクセス履歴数による差を軽減するため、 L^2 ノルム (長さ) が 1 になるように正規化する。本稿では、これにより得られた特徴ベクトルを用いて階層的クラスタリングにより、フォルダを作業ごとに分類する。

2.4 階層的クラスタリングによる分類

階層的クラスタリングでは、クラスタの併合の様子をデンドログラムと呼ばれる樹形図で表せる。図 2 に、7つのフォルダの階層的クラスタリングにより得られるデンドログラムの例を示す。また、デンドログラムの高さはクラスタ間距離を表しており、これを閾値で分割することでクラスタ数を決定することができる。図 2 では、閾値によりデンドログラムを分割することで3つのクラスタを得ている。この閾値を変化させることで、クラスタ数を調整できる。

本論文では、2.3 項で述べた方法により作成した特徴ベクトル V_A と V_B の距離をコサイン類似度を用いて、式 (1) により定義する。

$$\text{dist}(V_A, V_B) = 1 - \frac{V_A \cdot V_B}{\|V_A\|_2 \|V_B\|_2} \quad (1)$$

また、階層的クラスタリングにおいて、クラスタ間距離を群平均法により算出する。群平均法とは、2つのクラスタ同士で、すべての組合せのサンプル間距離の平均をクラスタ間の距離とする方法である。

上記のように仮想フォルダはワーキングディレクトリを分類することによって生成される。ここで、計算機ユーザ

にとって、生成した仮想フォルダにどのような作業が行われたワーキングディレクトリが保存されているかを確認できる必要がある。このため、仮想フォルダに対し、フォルダ名を付与する必要がある。そこで、3章でフォルダ名を代表するファイル拡張子の抽出手法について述べる。

3. フォルダ名を代表するファイル拡張子の抽出手法

3.1 概要

仮想フォルダ名を代表するファイル拡張子を考える際には、いくつかの考慮すべき観点がある。1つ目は、配下にあるワーキングディレクトリとの関係である。仮想フォルダには複数のワーキングディレクトリが保存されている。この例を図3に示す。図3の例では、仮想フォルダには6個のワーキングディレクトリが保存されている。この図のうち、「.py」はPythonのプログラムを保存するための拡張子を示しており、「.rb」はRubyのプログラムを保存するための拡張子を示している。また、1個のワーキングディレクトリには「.py」のファイル拡張子が20個存在し、5個のワーキングディレクトリには「.rb」のファイル拡張子が計9個存在する。この例において仮想フォルダに付与するフォルダ名には、Pythonのファイルが保存されていることを示すべきか、Rubyのファイルが保存されていることを示すべきかについて検討する必要がある。前者の場合、仮想フォルダに保存されているファイルの中で、一番多いファイル拡張子をもとに仮想フォルダにフォルダ名を付与することで、Pythonのファイルが多数保存されていることを示せる。後者の場合、1つのワーキングディレクトリを1つの作業と捉え、仮想フォルダの中で一番多いRubyプログラミングの作業をもとに仮想フォルダにフォルダ名を付与することになる。よって、このように仮想フォルダの数とワーキングディレクトリの変化することで、フォルダ名の付与におけるファイル拡張子の使用手法について、以下の2種類が考えられる。

- (1) 仮想フォルダ配下の全ファイル中で一番多い拡張子を使用する手法 (以下、全体手法)

図3における「.py」がこれにあたる。

- (2) ワーキングディレクトリごとに候補を取り、候補に多数あがった拡張子を使用する手法 (以下、個別手法)

図3における「.rb」がこれにあたる。

2つ目の観点は、時系列を考慮した場合、配下の全てのファイルを平等に扱ってよいかという観点である。再び図3を例に取り、図中の全てのファイルを拡張子の抽出に用いるべきかを考える。図中の○で囲まれたファイルは、一定期間内にユーザがアクセスをしたファイルを示している。これによると、仮想フォルダ中にPythonのプログラ

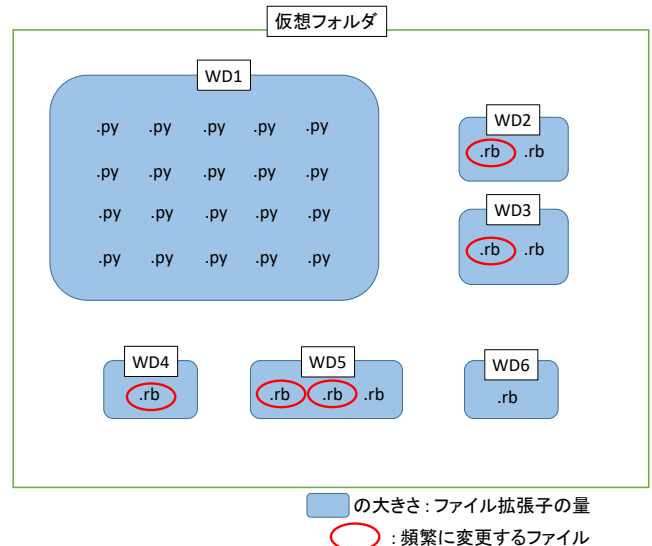


図3 ワーキングディレクトリごとに保存されているファイル拡張子の例

ムは多いものの、頻繁に変更しているのはRubyのプログラムである。つまり、仮想フォルダのフォルダ名はユーザーにとってPythonのファイルが多いという観点が重要か、Rubyのプログラムを頻繁に書いているという観点が重要かによって、結果が変わるといえる。よって、フォルダ名の付与に使用するファイル拡張子を時系列を考慮するかどうかという点で、2種類の使用手法が考えられる。

- (1) 時系列を考慮して拡張子を使用する手法 (以下、動的手法)

図3の場合、「.rb」がこれにあたる。

- (2) 時系列を考慮せずに拡張子を使用する手法 (以下、静的手法)

図3の場合、「.py」がこれにあたる。

上記2つの観点の組み合わせから4種類のファイル拡張子の抽出手法が考えられる。3.2節では、これらのファイル拡張子の抽出手法について述べる。

3.2 ファイル拡張子の抽出手法

3.1節で示した観点より、表2に示す4種類のファイル拡張子の抽出手法が考えられる。具体的には、(全体+動的)であれば、3.1節に示す全体手法と動的手法を用いて拡張子を抽出する手法であることを示している。ここで、フォルダ名を付与する際、どの手法を用いるかを決定する必要がある。そこで4章では、実際にこれらの抽出手法から得た拡張子を用いて仮想フォルダにフォルダ名を付与し、精度を比較する。

表 2 4 種類のファイル拡張子の抽出手法

		拡張子の使用手法	
		全体手法	個別手法
時系列を考慮 するか	動的手法	(全体+動的)	(個別+動的)
	静的手法	(全体+静的)	(個別+静的)

4. ファイル拡張子の抽出手法の精度の評価

4.1 概要

3 章ではフォルダ名を代表するファイル拡張子の抽出手法について示した。ここで、フォルダ名を付与する際に、表 2 の抽出手法のうち、どの手法を用いるかを決定する必要がある。そこで実際に、これらの手法で得た拡張子を用いて仮想フォルダにフォルダ名を付与し、それがどの程度ユーザとの感覚に近いのか、精度を比較する。比較手法はまず、フォルダのクラスタリング手法におけるクラスタ間距離の閾値を変化させ、仮想フォルダの数を変化させる。次に、数を変化させた仮想フォルダに対し、3.2 節の 4 種類のファイル拡張子の抽出手法を用いて、それぞれの手法で拡張子を抽出する。この次に、表 3 の拡張子とフォルダ名の対応表を使用し、抽出した拡張子を表 3 の拡張子と対応するフォルダ名を仮想フォルダに付与する。たとえば、図 3 の仮想フォルダから (個別+動的) を用いてファイル拡張子を抽出した場合、「.rb」が抽出される。この拡張子を表 3 と対応させて得られるフォルダ名は「Ruby」となり、これを仮想フォルダに付与する。最後に、それぞれの手法から得たフォルダ名の付与結果が正解データとどの程度一致しているかという観点で精度を計測し、比較する。

本章ではまず、4.2 節でフォルダ名の付与手法における比較の観点について述べる。次に、4.3 節で比較手法について述べる。最後に、4.4 節で比較結果について述べる。

4.2 比較の観点

仮想フォルダの数を変化させると、これに伴い、それぞれの仮想フォルダに含まれるワーキングディレクトリの数も変化する。これにより、仮想フォルダの数に応じてフォルダ名の付与結果が異なると考える。ここで、仮想フォルダの数が少ない場合と多い場合に考えられる影響について以下に示す。

(1) 仮想フォルダの数が少ない場合

仮想フォルダの数が少ない場合、仮想フォルダ 1 つあたりに含まれるワーキングディレクトリの数は多くなる。この場合、ファイル拡張子の使用手法が 3.1 節に示す全体手法と個別手法で、フォルダ名の付与結果に差が生まれると考える。この内容を図 3 を用いて説明

表 3 拡張子とフォルダ名の対応表

拡張子	フォルダ名
tex	LaTeX
Syntexbuzy	LaTeX
aux	LaTeX
xbb	LaTeX
sh	Shell
txt	Text
c	C
rb	Ruby
py	Python
rs	Rust
cgi	CGI
png	PNG
jpeg	JPEG
jpg	JPEG
wasm	Wasm
js	JavaScript
xlsx	Excel
pptx	PowerPoint
mp4	MP4
zoom	Zoom

する。図 3 の仮想フォルダは Ruby を用いた作業が多いため、正解データは「Ruby」としている。図 3 の仮想フォルダは個別手法を用いて拡張子を抽出した場合、仮想フォルダのフォルダ名には「Ruby」と付与されて正解データと一致する。しかし、全体手法を用いて拡張子を抽出した場合、「Python」と付与されて正解データと一致しない。これは、あるワーキングディレクトリがほかのワーキングディレクトリと比較して、特定の拡張子を大幅にもつ場合、この拡張子を使用してフォルダ名が付与されるためである。図 3 の仮想フォルダに対して全体手法を用いて拡張子を抽出した場合では、20 個の「.py」の拡張子と 9 個の「.rb」の拡張子が抽出される。この場合は「.rb」よりも「.py」の拡張子の方が多いため、「Python」が仮想フォルダのフォルダ名として付与される。このように、全体手法を用いて拡張子を抽出した場合、特定の拡張子を大幅にもつワーキングディレクトリに付与結果が大きく影響されるため、正解データと付与結果が異なると考える。

(2) 仮想フォルダの数が多い場合

仮想フォルダの数が多い場合、仮想フォルダ 1 つあたりに含まれるワーキングディレクトリの数は少なくなる。この場合、ファイル拡張子の使用手法が 3.1 節に示す動的手法と静的手法で、フォルダ名の付与

表 4 rails new コマンドの実行環境

項目	バージョン内容
OS	macOS Big Sur, version 11.5.2
Ruby	3.0.1p64
Rails	6.1.4.6

結果に差が生まれると考える。まず、正解データはユーザが行った作業に基づいて付与している。たとえば、Ruby on Rails を用いて作業を行う際は、主に Ruby のプログラムを変更しながら作業を行う。このため、仮想フォルダのフォルダ名には作業内容に基づいて、「Ruby」を正解データとして与えている。また、拡張子が「.rb」のファイルを頻繁に変更するため、動的手法を用いて抽出した拡張子をフォルダ名の付与に使用した場合、仮想フォルダのフォルダ名には「Ruby」と付与される。しかし、はじめて作業を行う際、rails new コマンドを用いてアプリケーションを生成すると、ライブラリとして、拡張子が「.js」である JavaScript のファイルが多く生成される。表 4 に示す環境で rails new コマンドを実行したとき、拡張子が「.rb」であるファイルが 30 個生成されたのに対し、拡張子が「.js」のファイルは 11,117 個生成された。このため、静的手法を用いて抽出した拡張子をフォルダ名の付与に使用した場合、拡張子が「.js」のファイルの方が多いため、仮想フォルダのフォルダ名には「Javascript」と付与される。このように、静的手法ではユーザが更新するファイルの拡張子とライブラリファイルの拡張子が異なる場合、フォルダ名の付与結果が正解データと異なると思われる。

上記の 2 点を考慮して、仮想フォルダの数を変化させて、比較を行う。4.3 節に比較手法を示す。

4.3 比較手法

比較手法について、以下の手順で比較を行う。

- (1) フォルダのクラスタリング手法におけるクラスタ間距離の閾値を変化させ、仮想フォルダの数を変化させる。本評価で使用するファイルアクセス履歴の期間と数、およびワーキングディレクトリの数について、表 5 に示す。表 5 より、ワーキングディレクトリの数は 80 個である。そこで、本評価では、仮想フォルダの数を以下のように変化させてフォルダ名の付与を行う。
 - (a) 仮想フォルダの数を 1 から 9 まで 1 ずつ変化させる
この場合は 4.2 節に示す、仮想フォルダの数が少ない場合について比較するために行う。
 - (b) 仮想フォルダの数を 10 から 80 まで 10 ずつ変化させる
この場合は 4.2 節に示す、仮想フォルダの数が多いため

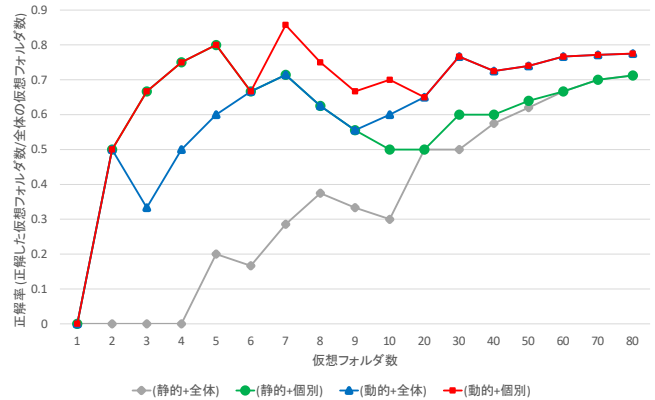


図 4 4 種類の手法で拡張子を抽出し、仮想フォルダにフォルダ名を与えた場合の正解率

合について比較するために行う。

- (2) (1) でフォルダ数を指定した仮想フォルダに対し、3.2 節の 4 種類のファイル拡張子の抽出手法を用いて、それぞれの手法で拡張子を抽出する。
- (3) (2) で抽出した拡張子と対応するフォルダ名を表 3 から参照し、仮想フォルダに付与する。
- (4) (3) の付与結果と正解データを比較するために、正解データを仮想フォルダに対して与える。この正解データは仮想フォルダにフォルダ名を付与する際に、ユーザ自身が正解とする名前を表 3 に示す対応表から選択する。また、正解データは仮想フォルダ 1 つに対し、1 つ与える。
- (5) 付与した結果について、式 (2) を用いて正解率を求める。

$$\text{正解率} = \frac{\text{正解した仮想フォルダ数}}{\text{仮想フォルダ数}} \quad (2)$$

4.4 比較結果

比較結果について図 4 に示す。この結果より、拡張子を(個別+動的)の手法で抽出した拡張子を用いてフォルダ名を付与した結果に関して、どのように仮想フォルダの数を変化させた場合でも、正解率が高い結果を示した。このため、この付与手法が作業内容別に分類した仮想フォルダにフォルダ名を付与する際、最もふさわしいと考える。

5. 関連研究

フォルダ構造を動的に変更するシステムとして、文献 [3][4][5] では、タグ情報を用いて、フォルダ構造を動的に変更するシステムが提案されている。このタグ情報は登録日時やファイルの特徴を表すキーワード群などで構成され、ファイル 1 つにつき 1 つ付与される。文献 [3][4][5] ではタグ情報を用いてファイル単位でフォルダ構造を分類し、フォルダ構造を動的に変更する。これに対し、本稿で

表 5 本評価で使したファイルアクセス履歴の期間と数、およびワーキングディレクトリの数

項目	内容
ファイルアクセス履歴の期間	2020 年 11 月 24 日～2021 年 8 月 24 日
ファイルアクセス履歴の数	198,188 件
ワーキングディレクトリの数	80 個

はファイルアクセス履歴を用いてワーキングディレクトリというフォルダ単位でフォルダ構造を動的に変更している。

また、ファイルブラウザを拡張することでファイル検索を支援する手法として、ユーザが次にアクセスする可能性が高いフォルダを予測して、ファイルブラウザ上で強調表示する手法が提案されている [6][7]。文献 [6][7] では、強調表示されたフォルダをクリックして階層移動することで目的のファイルを探す。一方、本稿では、仮想フォルダにより計算機内のワーキングディレクトリを再分類して提示している。このため、ユーザが次にアクセスする可能性が高いフォルダを予測できれば、予測結果を提示する仮想フォルダを生成することで、文献 [6][7] の手法よりも階層移動の回数が少なくて済むと考えられる。

6. おわりに

仮想フォルダ生成システムでは、ワーキングディレクトリの推定と分類を行って、仮想フォルダを生成する。この際、作業内容ごとに分類した仮想フォルダには、どのようなワーキングディレクトリが保存されているかを判別するため、フォルダ名を付与する必要がある。そこで本稿では、作業内容ごとに分類した仮想フォルダ配下にあるファイルの拡張子を用いてフォルダ名を代表するファイル拡張子を抽出する手法を提案した。また、この手法に基づいて、フォルダ名の付与手法を 4 種類提案した。これらの付与手法について正解率の比較を行い、評価を行った。この結果より、ワーキングディレクトリのファイルアクセス履歴の拡張子からそれぞれフォルダ名の候補を取り、時系列を考慮して候補に多数あがった拡張子をフォルダ名の付与に使用することが適切であることを示した。

参考文献

- [1] 西 良太, 乃村能成: 作業を代表するフォルダの推定と分類による仮想フォルダ生成システム, 情報処理学会論文誌, Vol. 62, No. 2, pp. 527-537 (2021).
- [2] 西 良太, 乃村能成: ファイル更新の時系列に着目したフォルダのクラスタリング手法, 情報処理学会研究報告, Vol. 2018-DPS-176, No. 4, pp. 1-8 (2018).
- [3] 淳也阿部, 大志出石, 裕一杉上, 幸雄 堀, 慈郎今井: タグ情報に基づくファイル管理システム, 情報処理学会研究報告. DBS, データベースシステム研究会報告, Vol. 141, pp. 97-102 (オンライン), 入手先 (<https://ci.nii.ac.jp/naid/110006205060/>) (2007).
- [4] 淳也阿部, 大志出石, 幸雄 堀, 慈郎今井: D-010 タグ情報に基づくファイル管理システムと実用性の評価 (D 分野:データベース), 情報科学技術フォーラム一般講演

- 論文集, Vol. 6, No. 2, pp. 23-24 (オンライン), 入手先 (<https://ci.nii.ac.jp/naid/110007688401/>) (2007).
- [5] 浜崎達郎, 高間康史: フォルダ・タグ管理モデルの連携によるファイル管理手法に関する提案, 日本知能情報ファジィ学会ファジィ システム シンポジウム 講演論文集, Vol. 26, pp. 281-281 (オンライン), DOI: 10.14864/fss.26.0.281.0 (2010).
- [6] Fitchett, S., Cockburn, A. and Gutwin, C.: Improving Navigation-based File Retrieval, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, New York, NY, USA, ACM, pp. 2329-2338 (online), DOI: 10.1145/2470654.2481323 (2013).
- [7] Fitchett, S., Cockburn, A. and Gutwin, C.: Finder Highlights: Field Evaluation and Design of an Augmented File Browser, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, New York, NY, USA, ACM, pp. 3685-3694 (online), DOI: 10.1145/2556288.2557014 (2014).