博物館関連語検索のための木構造を反映した資料群の構成法

画像電子学会 画像ミュージアム研究会 博物館・美術館 DTD-SG

山田 篤† 安達 文夫‡ 小町 祐史§

Atsushi YAMADA† Fumio ADACHI‡ Yushi KOMACHI§

†京都高度技術研究所 †ASTEM RI/Kyoto
‡国立歴史民俗博物館 ‡National Museum of Japanese History
§大阪工業大学 §Osaka Institute of Technology
E-mail: †yamada@astem.or.jp, ‡adachi@rekihaku.ac.jp, §komachi@y-adagio.com

1. はじめに

博物館・美術館情報の電子化が進み，ネットワークを通じて個々の館の収蔵品に関する情報の提供サービス，検索サービスなどが開始されている。博物館情報の利用者にとっては，どの館にアクセスするかを意識せず，各館の差異を意識せずにシームレスに横断検索できることが望ましい[1]。

本稿では，利用者の検索要求から関連する語をたどって収蔵品を見つけ出す際に，階層化して管理されている収蔵品の木構造を利用してグループ化した資料群を構成し，関連語を計算する方法について述べる。

2. 横断検索のためのフレームワーク

収蔵品の横断検索のためには，個々の館が持つ情報を統合する仕組みが必要となる[2]。画一的な共通フォーマットを用いずに，多様性を許容する情報構造として，次の 3 レベルに階層化される情報共有のフレームワークが提案されている[1]。

(1) 情報記述構造レベル
(2) 情報記述内容レベル
(3) 情報ナビゲーションレベル

このうち，(1)については，様々な取り組みがなされており，記述スキームの共有や標準化といった試みもある[3][4]。

これに対して，内容レベルの情報共有については，様々な収蔵品を扱うという事情から困難な問題として残っている。対象物が基本的に物であることから，図書のような全文検索といった手法も適用できない。オントロジ[6]や概念辞書[7]を用いた手法も検討されているが，収蔵品に関して様々なメタデータを付与することは博物館にとっても大きな負担となる。

このため，個々の博物館がなるべく簡単に用意できる情報をもとにして，内容レベルの情報共有を行い，横断検索を可能にすることが望まれる。

3. 資料群と資料名称を用いた関連語の計算

収蔵品に関して，博物館になるべく負担をかけずに収集可能なメタデータとして，資料群と資料名称を考える。

博物館では，様々な理由によって収蔵品をグループ分けして管理していることがある。このように

グループ化された資料の集まりを資料群と呼ぶことにする。資料群はさらに多階層に階層化されて管理されていることもある。[8]では資料群を分類と見なしているが，厳密な分類ではなくとも，グループ化され同じ資料群に属する収蔵品の間には何らかの関連があると考えられる。現状で，資料群の構成方法について，統一された基準は見あたらないが，大量の資料群を集めてくることによって，互いに関連する可能性の高い収蔵品を見いだす手がかりが得られるのではないかと考えている。

一方，個々の収蔵品には名称（資料名称）が付与されている。これと先の資料群を組み合わせると，資料群によってグループ化された資料名称の集まりができる。

[9]では資料群と資料名称との関係を，文書検索と同様の手法を用いて取り扱うことを提案している。また，[10]では，この考え方に基づき，実データに対して，資料名称の形態素解析を行い，資料名称中の主要語を抽出して，関連語の計算を行っている。さらに[11]では，関連語の順位を用いた到達容易性の評価方法について，[12]では，資料群の分割の仕方を変えることによる到達容易性の変化を，[13]では到達容易性の比較による資料群の分割の評価について述べており，関連語計算の際に，適切に資料群を設定することが課題となっている。

4. 階層構造を考慮した資料群の構成

博物館において収蔵品が階層化されて管理されている場合に，階層の最上位でまとめた資料群を作成し，これをもとに関連語を計算すると，望ましくない関連が発生することがわかっている。逆に，階層ごとに細かく資料群を分割していくと過分割となり，関連語としてふさわしい語までが排除されてしまう。適切な分割位置を人手で与えた場合に望ましい結果が得られることもあるが，このような分割位置の決定は容易ではない。そこで，階層構造を反映させた資料群を機械的に構成することができれば，望ましい結果に近づけることができるのではないかと考えた。

そこで，[13]と同じく，国立歴史民俗博物館の収蔵品データから，考古資料 21,935 点の資料名称を対象として，資料群の構成方法の違いにより，到達容易性がどのように変化するか，実験を行った。

どのような結果が望ましいかについては，「石器」，「石斧」，「土器」，「土偶」のそれぞれの語の関連語群を専門家に提示し，それぞれが関連語としてふさわしいかどうかを○×で判定してもらった。×が付与された語は，専門家にとっては出発語との関連がないと判定された語であるため，到達容易性が低くなることが望ましいものであるとして評価を行う。

具体的には，専門家によって「土器」の関連語としてはふさわしくないと判定された「石鏃」と，ふさわしいと判定された「高杯」を選び，それぞれについて「土器」からの到達容易性を計算した。

階層構造を資料群の構成に反映させる方法として，まず階層構造の部分木毎に資料群を構成して実験を行った。このとき，階層構造の上位でまとめた資料群には望ましくない関連が含まれることがあるため，その深さに応じて下位の部分木ほど優遇するようにした。

到達容易性は，[11]と同様に，ある語を入力した場合に，その関連語を関連度の高いものから順に提示することを考え，先頭の語からはじめて，一つ下位の語をみる確率を p とし，n 番目の語に到達する確率を p^{n-1}，ある語の関連語を見に行く確率を q（ただし，p+q<1）として，それぞれの資料群設定における到達容易性の計算を行った。具体的には p=0.89，q=pˆ20 として，関連語は第 60 位まで，別の関連語を辿る回数は 4 回までに制限した。

こうして，「土器」から「石鏃」に至る到達可能な全経路の計算を行い，全経路の確率の和を求めたところ，

分割 1（コレクションをそのまま資料群とした場合）：0.168589930619463
分割 2（階層構造で資料群を分割した場合）：0.047174548107095
分割 3（人手で分割位置を設定した場合）：0.0744339146642573
分割 4（部分木毎に資料群を構成した場合）：0.0593648767031672

となり，望ましい分割は順に，分割 2 ＞ 分割 4 ＞ 分割 3 ＞ 分割 1 となった。

次に，「土器」から「高坏」についても同様の計算を行うと，

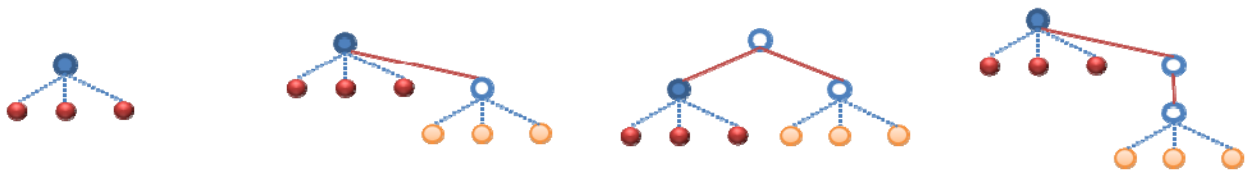
分割 1（コレクションをそのまま資料群とした場合）：0.00049785187787727
分割 2（階層構造で資料群を分割した場合）：0.00142262644849791
分割 3（人手で分割位置を設定した場合）：0.00203064928687728
分割 4（部分木毎に資料群を構成した場合）：0.00425238737485268

となり，望ましい分割は順に，分割 4 ＞ 分割 3 ＞ 分割 2 ＞ 分割 1 となった。

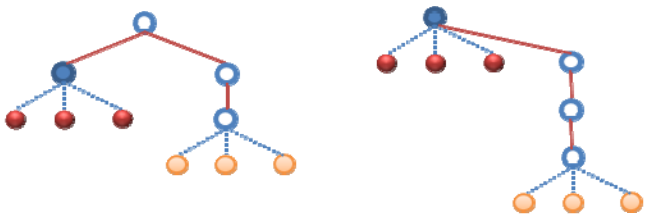
これらの結果から，資料群の構成時に階層構造を考慮することについて一定の効果はあると考えられるが，この構成方法では下位の部分木ほど一律に優遇しているため，浅い階層よりも深い階層が重視されることになる。これは階層の深さが一定であれば，それほど問題とはならないが，深い木と浅い木が混在する状況では，浅い木での関連が不利となるため，さらに木構造を反映した構成法を次に考える。

5. 木構造を反映した資料群の構成方法

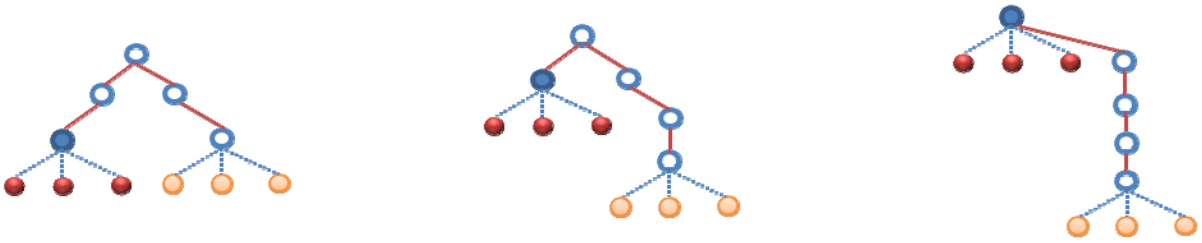
木構造を反映した資料群の構成法として，木の中間ノード間の枝の数に着目した方法を考える。同一の中間ノードを親に持つ終端ノードの集まりをパス長 0 の集合とする。次に，親となる中間ノード間の枝の数が 1 の終端ノードの集まりをパス長 1 の集合とする。以下，同様に親となる中間ノード間の枝の数に従い，パス長 2 の集合，パス長 3 の集合といったように定める。これらの関係を図示すると，図 1 のようになる。



パス長 0　パス長 1　パス長 2（a）　パス長 2（b）



パス長 3（a）　パス長 3（b）



パス長 4 (a)	パス長 4 (b)	パス長 4 (c)

図 1　木構造を反映した資料群の構成

今，中間ノードを N，終端ノードを T，カレントノードの子供の終端ノードを CT とし，makeGroup を，その引数のノードで資料群を構成する関数とすると，それぞれのパス長の資料群は以下のようにして構成することができる。

パス長 0：makeGroup(CT)
パス長 1：for-each child::N makeGroup(CT | child::T)
パス長 2 (a)：for-each following-sibling::N makeGroup(CT | child::T)
パス長 2 (b)：for-each child::N/child::N makeGroup(CT | child::T)
パス長 3 (a)：for-each (preceding-sibling::N | following-sibling::N)/child::N
makeGroup(CT | child::T)
パス長 3 (b)：for-each child::N/child::N/child::N makeGroup(CT | child::T)
パス長 4 (a)：for-each parent::N/following-sibling::N/child::N makeGroup(CT | child::T)
パス長 4 (b)：for-each (preceding-sibling::N | following-sibling::N)/child::N/child::N
makeGroup(CT | child::T)
パス長 4 (c)：for-each child::N/child::N/child::N/child::N makeGroup(CT | child::T)

ルートノードから順に，中間ノードに対してこれらの処理を再帰的に適用することでそれぞれのパス長の資料群を得ることができる。関連語の計算においては，より近い関係の資料群を優遇するために，パス長 n で打ち切る場合，(n-パス長)の重み付けを行う。

このようにして資料群を機械的に設定し，考古資料について，先と同様の計算を行ったところ，「土器」から「石鏃」に至る到達容易性は，

分割 5（階層構造：パス長 0 のみ）：0.0439894373190579
分割 6（階層構造：パス長 1 まで）：0.0704652440989046
分割 7（階層構造：パス長 2 まで）：0.0230753126453718
分割 8（階層構造：パス長 3 まで）：0.0133875812892122
分割 9（階層構造：パス長 4 まで）：0.00999321824507865

となり，望ましい分割は順に，分割 9＞分割 8＞分割 7＞分割 5＞分割 2>分割 4>分割 3>分割 1＞分割 6 となった。

次に，「土器」から「高坏」についても同様の計算を行うと，

分割 5（階層構造：パス長 0 のみ）：0.00154847291340535
分割 6（階層構造：パス長 1 まで）：8.73784760843763e-005
分割 7（階層構造：パス長 2 まで）：0.000377237442896436
分割 8（階層構造：パス長 3 まで）：0.00180140362183845
分割 9（階層構造：パス長 4 まで）：0.000785995570023153

となり，望ましい分割は順に，分割 4>分割 3>分割 8＞分割 5＞分割 2>分割 9＞分割 1＞分割 7＞分割 6 となった。

次に，「石器」，「石斧」，「土器」，「土偶」のそれぞれについて，関連語の中で専門家によって唯一×がつけられていないものを探し，そこへの到達容易性を計算してみた。たとえば，「のみ」は「石斧」以外の語の関連語としてはふさわしくないとされている。そこで，「石器」，「石斧」，「土器」，「土偶」のそれぞれの語から「のみ」への到達容易性を計算したところ，

分割 5（階層構造：パス長 0 のみ）：土偶 ＞ 土器 ＞ 石器 ＞ 石斧
分割 6（階層構造：パス長 1 まで）：土偶 ＞ 石器 ＞ 土器 ＞ 石斧
分割 7（階層構造：パス長 2 まで）：土偶 ＞ 土器 ＞ 石器 ＞ 石斧
分割 8（階層構造：パス長 3 まで）：土偶 ＞ 石器 ＞ 土器 ＞ 石斧
分割 9（階層構造：パス長 4 まで）：土偶 ＞ 石器 ＞ 石斧 ＞ 土器

となった。パス長 4 までを取り入れても「石斧」を出発語とした場合の到達容易性がそれほど上がっていないが，これは，元の資料群の階層化において，「のみ」と「石斧」を関連づけるような区分がなされていなかったためであると考えられる。このように，対象の機能に着目するといった全く別の観点からの関連性を取り入れるためには，そのような観点からの階層化の事例が必要となる。

他にも「石器」からのみ×がつけられていなかった「石皿」，「土偶」からのみ×がつけられていなかった「冠」，「土器」からのみ×がつけられていなかった「杯」についても同様の計算を行った結果を以下に示す。

「石器」→「石皿」

分割 5（階層構造：パス長 0 のみ）：石斧 ＞ 石器 ＞ 土器 ＞ 土偶
分割 6（階層構造：パス長 1 まで）：石斧 ＞ 石器 ＞ 土器 ＞ 土偶
分割 7（階層構造：パス長 2 まで）：石斧 ＞ 石器 ＞ 土偶 ＞ 土器
分割 8（階層構造：パス長 3 まで）：土偶 ＞ 石器 ＞ 石斧 ＞ 土器
分割 9（階層構造：パス長 4 まで）：土偶 ＞ 石器 ＞ 石斧 ＞ 土器

「土偶」→「冠」

分割 5（階層構造：パス長 0 のみ）：石器 ＞ 土偶 ＞ 石斧 ＞ 土器
分割 6（階層構造：パス長 1 まで）：土偶 ＞ 石斧 ＞ 石器 ＞ 土器
分割 7（階層構造：パス長 2 まで）：石器 ＞ 土偶 ＞ 石斧 ＞ 土器
分割 8（階層構造：パス長 3 まで）：石器 ＞ 土偶 ＞ 石斧 ＞ 土器
分割 9（階層構造：パス長 4 まで）：石器 ＞ 土偶 ＞ 石斧 ＞ 土器

「土器」→「杯」

分割 5（階層構造：パス長 0 のみ）：土器 > 石器 > 土偶 > 石斧
分割 6（階層構造：パス長 1 まで）：土器 > 土偶 > 石器 > 石斧
分割 7（階層構造：パス長 2 まで）：土器 > 石器 > 土偶 > 石斧
分割 8（階層構造：パス長 3 まで）：土器 > 土偶 > 石器 > 石斧
分割 9（階層構造：パス長 4 まで）：土器 > 土偶 > 石器 > 石斧

実際には，関連する語といっても，コンテキストの違いでどのように関連するかは異なるため，すべての観点をうまくカバーするような階層化がなされていない限り，対象によって，より強い共起関係をもつものが上位に現れている。

6. おわりに

本稿では，博物館，美術館の収蔵品の横断検索において，資料名称と資料群に基づき計算された関連語を用いてある語から別の語へ辿っていく場合に，収蔵品の木構造をもとに資料群を構成する方法について述べた。あくまでも機械的に設定してどこまでできるかということで，計算結果は対象とする収蔵品の階層化のされ方に強く依存するものになっている。また，階層化の際に，あまり考慮されていない関係は取り出すことができないといった限界もあるが，専門家の知見と比較してもそれほど違わない結果が得ることができている。この考え方を発展させて，一般的な語彙から出発して，未知の専門用語で記述された収蔵品にたどり着く方法について検討を進めたい。

文　献

[1] 山田篤，他：博物館情報の知的横断検索のためのフレームワーク，画電年次大会，2002-06.
[2] 山本泰則，中川隆：博物館資料情報共有の試み，画電年次大会，2004-06.
[3] 文化財情報システムフォーラム（http://www.tnm.go.jp/bnca/）.
[4] The International Committee for Documentation of the International Council of Museums (ICOM-CIDOC) (http://www.cidoc.icom.org/).
[5] 山田篤，他：博物館情報の分類マッピングを用いた横断検索，画電年次大会，2004-06.
[6] 山田篤，他：博物館情報の横断検索におけるオントロジ利用の試み，画像ミュージアム研究会，2005-03.
[7] 山田篤，他：博物館横断検索に向けた概念辞書の枠組みの検討，画像ミュージアム研究会，2007-03.
[8] 山田篤，他：部分的分類知識の統合による博物館情報の横断検索の提案，画像ミュージアム研究会，2008-02.
[9] 山田篤，他：博物館資料群中の語の共起関係を用いた関連語抽出，画像ミュージアム研究会，2009-03.
[10] 山田篤，他：博物館資料群中の語の共起関係を用いた関連語抽出における主要語選定の効果，画電年次大会，2009-05.
[11] 山田篤，他：博物館情報探索における関連語の順位を考慮した到達容易性の評価，画電年次大会，2010-06.
[12] 山田篤，他：博物館情報検索における到達容易性向上のための資料群分割の効果，画像ミュージアム研究会，2011-02.
[13] 山田篤，他：博物館情報探索における資料群分割の到達容易性による評価，画電年次大会，2011-06.