Applying User-Guided, Dynamic FCA to Navigational Searches for Earth Science Data and Documentation

Bruce R. Barkstrom

15 Josie Ln, Asheville NC 28804, USA

Abstract. This paper describes data structures and algorithms that allow disciplinary taxonomic experts to embed Formal Contexts within a graph of Archive Information Packages (AIP's). The AIP's are standardized objects that provide access to the Inventoried Objects (IO's) in an archive. For an archive containing Earth science data, IO's may be physical specimens or numerical data files. They are not just textual files that provide a corpora of keywork phrases. The graph serves as a Table of Contents for the archive's collection. A data user familiar with the discipline's taxonomy having a recognizable search target can navigate through the graph to identify and access the archive's IO's.

1 Introduction

An archive containing Earth science data may contain physical objects or numerical data files. Its Inventoried Objects (IO's) do not necessarily contain textual content. Thus, they may not provide keyword phrases chosen from their textual content. This paper's algorithms assume that a disciplinary taxonomist develops an IO classification. To do this, the taxonomist creates vertices in a network. Each nonterminal vertex points to a unique Partitioned Formal Context (PFC). Each terminal vertex points to a unique IO.

The Archive Information Packages (AIP's) described in the ISO standard known as the Open Archive Information System (OAIS) Reference Model (RM) [1] serve as prototypical data structures for the network's vertices. The nonterminal vertices are Archive Information Collections (AIC's). The terminal ones are Archive Information Units (AIU's) that point to IO's. Disciplinary taxonomists create the network as the Archive adds to its collection of IO's. Archive users familiar with that taxonomy can navigate through the network's paths to select a target IO that satisfies their search criteria. Navigation differs from a browsing approach in which a user does not have a specific kind of target IO in mind.

This taxonomic approach markedly reduces the size of the Formal Contexts a FCA algorithm uses to create Formal Concepts from the PFC's. The computed Formal Concepts guide the user's navigation from AIC to AIC until the user finds an AIU with the target IO. The next section includes brief examples of the kinds of IO's in archives of Earth science data. Then, it describes the way they enter the taxonomic network. Later sections include outlines of data structures for the network vertices and pseudo code for the key algorithms in this approach.

ISBN 978–2–7466–6566–8, Laboratory L3i, University of La Rochelle, 2013.

2 Creating a Taxonomic Classification Network

This section discusses the basis for creating a taxonomic classification network. The first subsection notes that a disciplinary taxonomist provides the a priori categories for the taxonomy. This subsection also provides examples of categories for two kinds of Earth science data. These categories depend on the disciplines providing the data. Thus, this paper treats them as axiomatic. The second subsection provides a formal discussion of the AIC and AIU data structures. The third subsection deals with a rough scaling analysis of a taxonomic network.

2.1 The Axiomatic Disciplinary Basis for a Taxonomy

The a priori basis for creating a taxonomy for an archive's collection lies in the scientific disciplines that produce the IO's. As with the OAIS RM, we assume that the IO's are atomic so they have no internal components that the archive records. In Earth science, IO's may be physical objects, such as biological or geological specimens. They may include hand-written or printed observational records or technical reports. Finally, Earth science IO's may be digital files, the bulk of which contain numerical values based on measurements from automated instruments.

A second axiomatic basis is the Formal Context (FC) formed by all the IO's in the archive. The union of the IO's forms the extent of this FC. Its intent is the union of all the IO attributes.

The NOAA Emergency Response Imagery Collection is an archive of digital images taken by an automated camera mounted on an aircraft [2]. The aircraft flew one or two days after disastrous storms. There have been about twenty storms in the last decade that were disastrous enough to lead to aircraft missions. These included Hurricane Katrina in 2005 and Hurricane Sandy in 2012. Each mission produces three kinds of IO's: high resolution gif images, low resolution gif thumbnails, and zipped files containing collections of images. The archive's curators organize the images in a zipped file in a sequence along an aircraft flight path.

The ERI project provides a web site through which emergency responders, such as federal disaster coordinators or insurance adjustors can download IO's. These users evaluate the storm damage and plan appropriate action based on the images. Altogether, the total collection probably contains about 60,000 IO's. Each IO has an ID, a storm name, a data collection date, and a text storm centroid location. The high resolution and thumbnail gif's have four latitude/longitude positions at the corners of each image. With eleven attributes for most of the files, there will be about 660,000 attributes ($11 \times 60,000$) in the Formal Context intent for this collection's IO's.

In the taxonomic network, the AIC's that lie in the first level below the root refer to images from specific storms. Thus, these collections include images from storms such as Hurricanes Katrina, Ike, or Sandy. In the level below that, each storm AIC has two collections. One is an AIC for a collection of images in coarse geographic bins. Those images include the high-resolution gif images and the thumbnails. The second AIC is a collection of zipped files along the flight paths.

The AIC's that are children of each geographic bin AIC contain AIC's with one high resolution image and one low-resolution image. The AIC's below the one with the flight path collection contain AIU's based on an individual flight path. The network replicates this layered AIC structure across the entire collection of IO's. Even so, each PFC is unique.

For biological specie classification, the categories for the AIC's are Kingdoms, Subkingdoms, Classes, Orders, Families, Genera, Species, and Races. The individual specimens in an archive's biological specimen collection are the IO's in the AIU's.

2.2 A Formal Definition of AIC's and AIU's

A disciplinary taxonomist creates AIP's and AIU's. This is a familiar process in biology. [3] [p. viii] notes that such a "grouping ... though based on natural characters and relationships is not governed by any rule drawn from nature for determining just what [...attributes] shall be sufficient to constitute a Specie, a Genus or a Family. These groups are, therefore, necessarily more or less arbitrary and depend upon the judgement of scientific experts."

The AIC's and AIU's are vertices in a graph. The pseudo code in the Taxonomy Network Data Structure provides a pointer-based formalization of the relationship between the AIC's and the AIU's.

Taxonomy Network Data Structure

```
type p_PFC is access PFC_Type; -- Pointer to a PFC
type PFC_Type(N_O : in positive; N_A : in positive)
    is array(1 .. N_O, 1 .. N_A) of Boolean;
type Type_Of_AIP is (Collection, IO);
type p_AIP is access AIP; -- Pointer to an AIP
type AIP(AIP_Type : Type_Of_AIP := Collection) is record
    AIP_Identifier : Bounded_String;
    Parent_AIP : p_AIP;
    Next_Sibling_AIP : p_AIP;
    case AIP_Type is
        when Collection =>
            PFC : p_PFC;
            First_Child_AIP : p_AIP;
        when IO =>
            Object_Identifier : Bounded_String;
    end case;
end record;
```

If all of the PFC's in the graph were diagonal, each object in the PFC would have only one unique attribute. The taxonomic graph would collapse to a tree. There would be a unique edge that linked each AIC to each of its children. When the parent PFC is not diagonal, the formal concepts formed from the PFC create a lattice that links parents to their children. For a taxonomic network, the lattice must provide a unique path from a parent to each of its children.

2.3 Preliminary Scaling Analysis of Taxonomic Classification

The number of children and the number of attributes for the PFC of a particular AIP can vary widely across the network. This variability makes it difficult to formulate simple scaling relationships for the work of computing formal concepts from the PFC's. For example, the root PFC in the ERI example has about twenty storm objects. Each storm has three attributes: storm name, date, and centroid location name. Thus, for this PFC, $N_O = 20$ and $N_A = 60$. At the next level, the PFC's have $N_O = 2$ and $N_A = 2$. The children of each PFC have only a coarse geographic bin collection and a flight path collection. At the level below this, the coarse bin PFC's may have twenty to fifty objects. The flight path PFC's have five to twenty objects. Clearly, the partitioning produces much smaller binary matrices than the unpartitioned Formal Context.

A rough approximation for scaling assumes that the graph reduces to a multiway tree. The number of children for each AIC is M. The number of levels is L. The total number of AIC nodes in the tree, N, is

$$N = 1 + M + M^2 + \ldots + M^L \tag{1}$$

If the number of attributes in each PFC is $N_A = aM$, then the size of each PFC is (M, aM). Both M and a are usually small compared with the dimensions of the Formal Context for the archive's total collection. This approximation suggests that the computational work from the network partitioning can reduce that burden by several orders of magnitude.

The archive does not have to calculate the Formal Concepts for any of the PFC's when it establishes the network. That calculation can occur when the user's navigational search reaches an AIC. Under this simplifying assumption, a user with a successful navigation search will only select L formal contexts. The archive only needs to calculate Formal Concepts for each of the selected PFC's. User attribute pruning to remove attributes the user regards as irrelevant will further reduce the computational load.

Disciplinary users familiar with a taxonomy form a much smaller community than the public using commercial search engines. For example, only about 1% to 2% of students entering U.S. colleges want to enter scientific or mathematical curricula. The reduction in computational burden and in user search requests should make the taxonomic partitioning approach computationally acceptable.

3 Disciplinary Specialist Knowledge of a Taxonomy

A disciplinary specialist, such as a research scientist or resource manager, spends a substantial amount of time learning a discipline's taxonomy for classifying objects. Such a specialist is likely to have a specific target for a search. Thus, such a user seems highly likely to use the discipline's taxonomy classification.

A biologist who has a new, unidentified plant would usually use a biological taxonomy to see if there were any previously identified specimen's in an archive's collection that matched her new one. Even if the biologist moved rapidly through the upper levels of the collection, she would eventually get to a level where she

would need to match the attributes of her sample against the standard attributes that identify the particular specie within a probable GENUS.

The search behavior of a disciplinary specialist differs markedly from that of an individual with an unclear target for his or her search. A person *browsing* needs a recommender site rather than a navigational one [4].

4 User-Guided Navigational Searches

The taxonomic classification approach expects that a user can identify a useful IO based on the taxonomy network traversal. If the user obtains his or her goal, then the search terminates successfully. If the user recognizes that the search is not likely to reach the target, he or she can back up to a higher level and make different selections. Thus, we expect the user search to be iterative. In addition, the user can get tired of searching and terminate the interaction. The following pseudo code outlines the search algorithm.

User-Guided Navigational Search Algorithm

```
1   Level := 0; AIP := Root; Done := False;
2   while not Done loop
3       User prunes AIP attributes;
4       Compute all Formal Concepts for the pruned Formal Contex;
5       Construct Nearest Superset Navigation (NSN) graph;
6       User navigates through the graph to a Candidate AIP
          at Level + 1; AIP := Candidate AIP; Level := Level + 1;
7       if Candidate AIP is target AIU then
8           Done := True; --SUCCESS
9       elsif User judges the search unlikely to reach target then
10          AIP := Previous AIP; Level := Level - 1; --BACK UP
11      elsif User is tired of search then
12          Done := True; --ABANDON SEARCH
13      end if;
14  end loop;
```

The user-guided navigational search algorithm outlined in the pseudo code starts at the root AIP. The loop moves down the taxonomic AIP graph from the root at Level 0 to deeper levels where the user can find the desired target IO. The first step for the user is to prune the selected AIP attributes to create a pruned formal context. In pruning, the user removes attributes regarded as irrelevant to finding the target IO. Pruning the static Formal Context removes the irrelevant columns and thereby produces a smaller Formal Context. The pruning also removes any rows that contain no objects after removal of the attribute columns. The pseudo code for the NSN subalgorithm provides the logic for lines 5 and 6 of this listing.

Nearest Superset Navigation (NSN) Subalgorithm

```
1   Order Pruned Context objects into layers based on increasing number
      of attributes;
2   Construct a directed graph with objects identified as vertices
      and edges that connect vertices to their immediate successors
      in the layer with the smallest increase in number of attributes;
3   Construct a web site in which each page contains information on
      a single vertex and has links to the immediate successors;
4   Search Successful := False; Done := False;
5   Select page with the vertex that has no attributes and all objects;
6   while not Done loop
7       User selects pages from the links on the current page;
8       if selected page has only one IO then
9           if IO is User Target then
10              Search Successful := True;
11              Done := True;
12          else
13              Done := True;
14          end if;
15      end if;
16  end loop;
```

5 Conclusion

This paper shows how a taxonomic classification with FCA can fit within the OAIS RM's standard structure for information packages in an Archive of Earth science data. The scaling analysis of the computational load of user-guided navigation and dynamic FCA needs refinement. Even so, it suggests that this approach is affordable. [2] shows that a navigational approach similar to one derived from this algorithm offers an alternative to the conventional metadata query approach for selecting IO's from an archive of Earth science data.

6 Acknowledgements

The author gratefully acknowledges helpful discussions of this paper's contents with Dr. Mike Folk, Ms. Beth Huffer, Dr. Nancy Hoebelheinrich, and Mr. Gustaf Barkstrom.

References

1. CCSDS: *Reference Model for an Open Archival Information System (OAIS): Recommended Practice; CCSDS 650.0-M-2* (2012) CCSDS Secretariat, Washington.
2. NOAA: Emergency Response Imagery (2013) URL: `http://storms.ngs.noaa.gov/eri_page/index.html`
3. Britton, N. and Brown, A.: *An Illustrated Flora of the Northern United Sates and Canada: in Three Volumes.* Dover Publications, Mineola, NY. (1970)
4. Agarwal, D., Chen, B-C., Elango, P., and Ramakrishnan, R.: Content Recommendation on Web Portals. *Comm. ACM*, **5**6 (6) (2013) 92-101