

ArtFinder: A Faceted Browser for Cross-Cultural Art Discovery

Matt Thompson¹, Steve Battle³, Julian Padget¹, and Hideaki Takeda²

¹ University of Bath, UK

² National Institute of Informatics, Japan

³ Sysemia Ltd, UK

Abstract. Exploring art from a different culture without prior knowledge of the domain is difficult. Though traditionally experts are required to guide people through the unknown knowledge base, the use of linked data can help consumers to explore for themselves. In this paper, we use clustering methods to create a faceted hierarchy for the exploration and recommendation of Japanese artists to tourists visiting Japan. This opens up future work in the understanding of the links between artists from different cultures as well as in automatically categorising and browsing linked data.

Keywords: Clustering, Taxonomy, Linked Data, Faceted Search

1 Introduction

Discovering new art can be difficult. When a user knows what they are looking for, a direct text input search interface is effective enough. However, when the user wants to explore a selection of cultural artefacts which they are unfamiliar with, knowing the right keywords to use is difficult. Since it assumes prior knowledge of the domain, a user would require many trial-and-error search queries before they are able to find what they are looking for, even if they have a very clear image of what it is.

In this paper, we describe the use of faceted search as a method of exploring art and cultural artefacts from different cultures. Consider a scenario where a person visits an art gallery while on holiday in a foreign country. The person could be looking at an exhibit filled with artwork by totally unfamiliar artists. What would be the most efficient way to explore these works?

Faceted search has its roots in the work of Ranganathan [1] for use in library classification. Pollitt [2] adapts it for browsing digital library archives, and it is developed further by Hearst [3] and Yee [4]. It has been shown to be an effective method for a user to explore a large and unfamiliar dataset [5][4]. Rather than having to browse through a single hierarchy, multiple hierarchies are created that provide different ways to browse through the data.

For example, a user could browse a database of films by navigating through distinct sets of links based on era (decade/year), cast (actors/director), genre (romance/thriller) or language. These act as different ways of viewing and browsing a dataset, in a similar way to how an email client's filters allow a user to browse their mail according to

sender, date, flags or attachments. Each distinct facet or view can contain a hierarchy of subdivided categories, or a simple list. If a single hierarchy is used to browse the data instead of using facets, a film might only be discoverable if a user knows what year it was shown or what actors appeared in it. Faceted search allows the user to browse data according to domains that they have knowledge of.

Faceted search can be combined with existing search systems such as direct text input or query languages to facilitate exploration of data. For example, a user could first filter the data by entering a search term, then use faceted search to browse hierarchies that contain data related to that search term only.

Given faceted search's suitability for browsing through large datasets, how might it be used to explore a dataset that is *completely* alien to the user? What would be the best way of arranging the hierarchy of categories to make foreign categories understandable in context?

We explore this question by creating a faceted browser for artists from the Japanese LODAC ontology of artists and museum exhibits, cross-referencing data from English and Japanese DBpedia to convert the names and categories from Japanese to English. Given a choice of facets to browse by, a user should be able to pick one that they are familiar with in order to find artists that they might be interested in. We test this theory experimentally by asking a number of users to look for interesting Japanese artists, by evaluating their experience with a questionnaire and by measuring the time taken for each task. The ease of use and effectiveness of faceted search are compared with using a non-faceted, hierarchical classification to browse the same dataset.

In the next section, we describe the generation of faceted and non-faceted hierarchies from the ontologies and the design of the faceted browser interface. The section after describes an experimental evaluation of the interface and hierarchies, and then follows a discussion and evaluation of the results. The paper concludes with a review of related work and examination of possible future research.

2 Design of the system

2.1 Cross-cultural faceted browsing

The specific problem that this research addresses is the difficulty of exploring the myriad genres and subclassifications of Japanese art for a non-Japanese person. There are many types of Japanese art that have no direct English translation. One example would be “hangaka”, which approximately translates to “woodblock artist” or “printmaker”. However, this description alone is inadequate in explaining the nuances of the genre, so a non-Japanese speaker would have difficulty forming an image in their mind of the type of art it describes.

By arranging categories of artists onto a faceted hierarchy, a user is able to discover a genre of art in context, so that they can see its relationship to other styles of art. If an unknown category is listed near others that the user likes, then it is likely that they will be interested in artists that belong to it.

Another advantage of facets is that they give a user some choice as to how they can browse the data. If they do not know the meaning of one facet, they can choose to browse using another facet that they are more familiar with.

So, in order to solve this problem, we have implemented “ArtFinder”, a faceted browser for Japanese art. ArtFinder presents two novel approaches for exploring unfamiliar cultural data:

1. Cross-referencing DBpedia and LODAC to create faceted hierarchies for RDF data in an ontology (as opposed to document collections, as shown in most faceted hierarchy approaches).
2. Using a faceted search approach to allow a user to explore artists from one culture using knowledge from another.

The effectiveness of the approaches outlined in this paper are then evaluated in an experimental study where the experiences of users of the system are measured by studying their interactions with the system and with a questionnaire. The results are compared with those of users browsing with a single, non-faceted hierarchical browser that is otherwise identical.

2.2 Faceted Search and Non-hierarchical RDF Metadata

We create a hierarchical faceted browsing system from RDF data retrieved using SPARQL queries, where the metadata is not sorted into a hierarchy of any sort. In the literature discussed in section 4, the authors either extract text from documents to create a hierarchical organisation of those documents, or use hierarchical RDF metadata to make a faceted browsing interface. In contrast, here we draw RDF data from multiple sources to automatically create a hierarchy of categories, then use this to make a faceted browsing interface.

2.3 Ontologies Used

LODAC The LODAC (Linked Open Data for Academia) [6] project is an ontology developed by the National Institute of Informatics (NII) in Tokyo for the sharing of museum and art exhibit data. It is accessible on the web at <http://lod.ac>, complete with a SPARQL endpoint. The ontology is populated with exhibit data from Japanese museums and art galleries. The art and artists presented are not arranged into any kind of hierarchy of categories. They are, however, tagged with metadata and can easily be cross-queried with English and Japanese DBpedia.

DBpedia, DBpedia Japanese DBpedia [7, 8] is a Linked Open Data conversion of the contents of Wikipedia, accessible through a public SPARQL endpoint. DBpedia Japanese [9] is maintained separately for the Japanese version of Wikipedia.

Keio University Wikipedia Ontology Created by Keio University in Japan, the Japanese Wikipedia Ontology [10] is an alternative ontology based on Japanese DBpedia. Advantages of this ontology over the Japanese DBpedia ontology are that it contains more information tagged with more semantic data, with links to LODAC and DBpedia Japanese where relevant.

Overlap Querying the SPARQL endpoints revealed that there are 893 artists that have information in both LODAC and ontologies based on Japanese Wikipedia. This is the subset of artists that we use for clustering.

2.4 Technology used

SPARQL queries were constructed and issued using a combination of the Clojure and Python programming languages. These languages offer a good balance of library support and expressiveness for rapid prototyping. Python was used to send Japanese names and tags to the Google Translate API to translate from Japanese to English whenever a translation could not be found on DBpedia Japanese. We also used Python to cluster the data and create the hierarchies.

The browser interface was written in HTML5 and Javascript with the Angular.js framework. These technologies were chosen because they would allow any user with a web browser to be able to use the interfaces to explore the datasets from home, if such a system were to be implemented outside of a research context.

2.5 Extraction and translation of tags

SPARQL queries were sent to the subset of 893 artist entries that exist in both LODAC and Japanese Wikipedia (through either DBpedia Japanese or the Japanese Wikipedia Ontology). Tags based on artist genres were taken from LODAC for the ‘genre’ facet, explained in section 2.6. Queries were also sent to determine the materials used for each artefact the artist has created. This is used for the ‘media’ facet.

Each artist has a list of ‘subject’ tags in the DBpedia Japanese ontology. These tags contain a lot of information from a variety of categories, for example ‘births in 1695’, ‘artists from Paris’ and ‘woodblock artists’. The tags were retrieved and translated from Japanese to English using the Google Translate API. We then processed these tags in order to sort them into distinct facets: *era*, *location*, *genre* and *media*.

2.6 Determination of facets

The DBpedia Japanese ontology contains a lot of metadata about artists. However, this metadata is not very well structured, since most of the useful information is contained in a `dc:subject` relation. In order to separate the tags into appropriate facets, some filtering was required.

First, in order to facilitate the filtering of the tags, all retrieved tags were translated from Japanese to English. Then regular expressions were used to sort the tags into facets.

According to Ranganathan, quoted by Oren et al [5], the basis of good facets should be:

1. temporal (e.g. year of publication, date of birth)
2. spatial (e.g. conference location, place of birth)
3. personal (e.g. author, friend)
4. material (e.g. topic, colour)

5. energetic (e.g. activity, location)

Based on these guidelines, we determine suitable facets to be: *era* (years active), *location* (country of birth, countries active in), *medium* (materials used, e.g. oil paint or sculpture) and *genre* (artistic movement).

Era (temporal) Era tags are generated from each artist’s list of tags by applying a regular expression that checks for “century”, “year”, “birth”, or “death”, and adding any matches to the list of era tags.

Location (spatial) The location tags were drawn from the list of tags by applying a regular expression containing a list of countries and nationalities to the tag. If the tag contains any word that mentions a country or nationality, then the corresponding country is added to the list of location tags for that artist. Also, another regular expression checking for the mention of “country”, “university”, “prefecture”, “people from” or “person from” is applied, and any matching tags are added to the location tags list.

Media (material) These tags were found by sending a query to LODAC over all artworks that an artist has made in order to determine the medium used.

Genre (energetic) Tags for genre are, by a process of elimination, any tags that were not used for the creation of the location or era tags. Tags for genre were also added by querying all of an artist’s artefacts in LODAC to discover the genre of an artist’s creations.

These facets reuse the broad categories outlined by Ranganathan. As Kwasnik notes in *The Role of Classification in Knowledge Representation and Discovery* [11]:

“A good classification functions in much the same way that a theory does, connecting concepts in a useful structure. If successful, it is, like a theory, descriptive, explanatory, heuristic, fruitful, and perhaps also elegant, parsimonious, and robust.”

2.7 Hierarchy Generation

For both the non-faceted hierarchy and within facets, Sanderson and Croft’s subsumption approach [12] to hierarchy creation is used. As in the original paper, we find that relaxing the constant to 0.8 gives better results, determined by examining the resulting hierarchy informally.

$$P(x|y) \geq 0.8, P(y|x) < 1, D_i > 4$$

That is to say, x becomes the parent of y if the documents in which y occurs are a subset of the documents in which x occurs. The constant value of 0.8 is relaxed from the value of 1 by the authors to give better results.

Incorporating the ideas of Schmitz et al [13], we add a constant D_i , which is the number of documents in which each tag occurs. This acts as a threshold specifying the minimum number of artists a tag must describe before it can be included in the

hierarchy. Through testing different values by generating multiple different hierarchies, we found 4 to be a value that produced good results.

Our algorithm, based on Heymann's greedy tag tree generation approach [14], goes through all of the tags that appear in the dataset in pairs. For each pair of tags, x and y , the probabilities of each occurring are calculated and used to find whether or not subsumption should occur. If subsumption occurs, then x subsumes y as its child node in the tree.

In the case of the faceted hierarchies, this algorithm is run separately on the tags that are members of each facet. For the single hierarchy, all of the tags in the dataset (that appear more than 4 times) are used to generate a single tree of tags.

2.8 Browser Interface

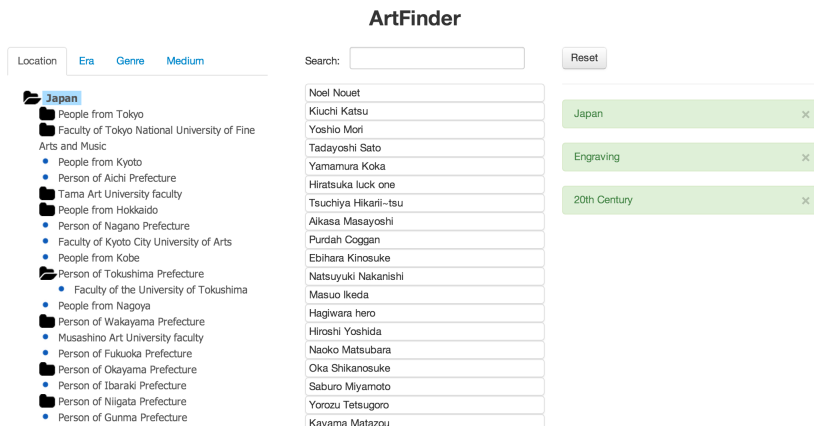
The user interface (figure 1) was written in Javascript using the Angular.js framework and is designed to run in a web browser. The facets appear as a set of tabs at the top of the navigation bar, which is displayed at the left of the screen. When a facet's tab is selected, the hierarchy of tags for that facet is displayed. The user is then able to expand and collapse nodes in the hierarchy to explore its structure and drill down from general to specific categories.

All artists corresponding to the currently selected categories in each facet are displayed as a list in the centre of the screen. When the name of a tag in the hierarchy is selected, it is added to the list of filters, displayed on the right. The list of artists in the centre is then changed to only show those that have all of the selected tags. Tags can be removed at any time.

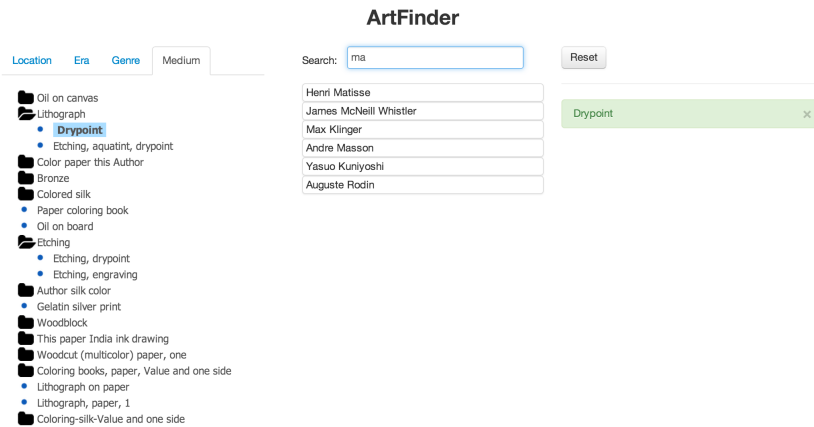
Users may filter this list further by typing in a search term in a text box at the top. This filters the list down to any artist that contains a tag matching the search term in any of their category facets.

Two different interfaces are evaluated in the study in section 3: one with the faceted interface for the tag browser on the left of the screen (figures 1a and 1b) and another with a single hierarchy where only one tag can be selected at a time (figure 1c). The textual search input can be used for both interfaces.

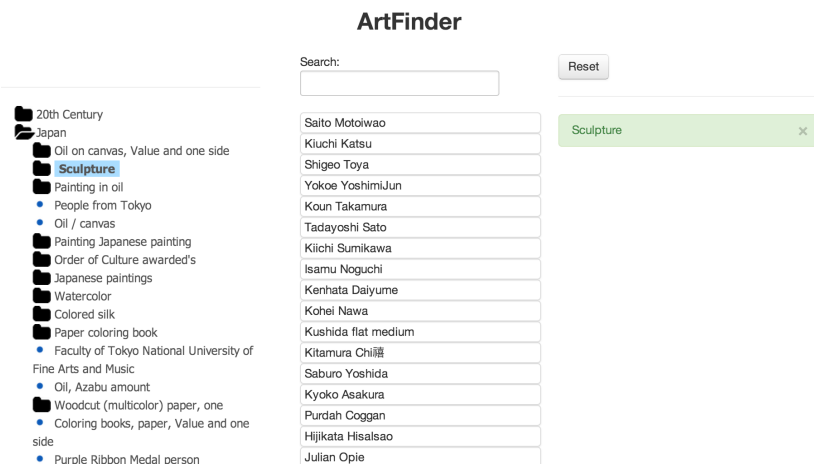
A live version of the interface is available online [15], along with its source code[16].



(a) Faceted browser with tags selected across facets



(b) Faceted browser with tags selected and text search filter applied



(c) Non-faceted browser with tags selected

Fig. 1. The ArtFinder browser interface

3 Usability Study

To evaluate the effectiveness of using faceted search to explore information about artists from another culture, we conduct a preliminary study.

The goals of the study are:

1. To determine whether using faceted search allows users to find artists more quickly in our dataset, when compared to using a single hierarchy.
2. To see how much faceted search facilitates exploration and discovery of new artists.

Time is the criterion for the first goal, as it is measured by timing how long participants take to complete a set of artist-finding tasks. The second goal is determined by user preference, and so is measured by setting the participants an exploratory activity to do and evaluating their experience with a questionnaire.

3.1 Participants

Using opportunity sampling, we select sixteen users to evaluate the effectiveness of the two browsing approaches. Users are from the postgraduate office at the University of Bath department of Computer Science, as well as from offices at the Bristol and Bath Science Park in Bristol.

3.2 Tasks

The users are given tasks to complete in the browsing interfaces. Eight participants are shown the faceted browsing interface first, then the single hierarchy browser. The other eight are shown the single hierarchy browser first. This accounts for the fact that each user becomes familiar with the browsing interface after the first set of tasks, and so is faster during the second set.

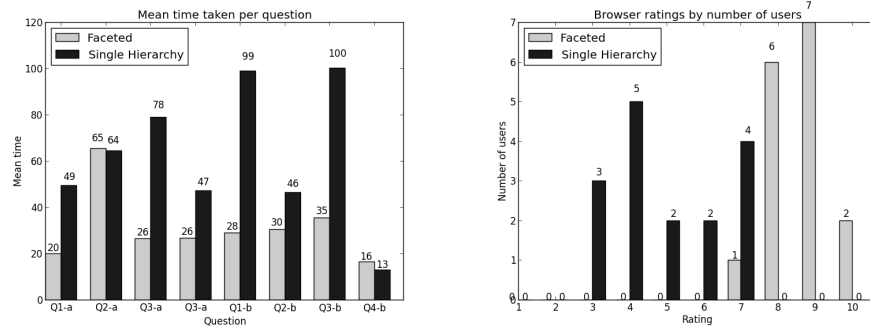
Participants are given five tasks to complete for each browsing interface. These tasks are divided into two sets, with one set for each interface:

Set A

1. What is the name of the only Dadaist from Tokyo?
2. Which person from Aichi prefecture died in 1999?
3. Name one 19th century German oil painter.
4. Name an artist from Los Angeles called 'John'.
5. Find an artist you like (filter using search). Click on them to see their tags. Using these tags, find similar Japanese artists.

Set B

1. Name a watercolour artist from Kyoto.
2. Which person from Nagano prefecture died in 1997?
3. Name one 18th century French engraver.
4. Name an artist from the United States called 'Joseph'



(a) Mean time taken for each interface, by ques- (b) Ratings for each interface, by number of users
tion

Fig. 2. Usability study results

5. Find an artist you like (filter using search). Click on them to see their tags. Using these tags, find similar Japanese artists.

Four users are given the faceted browser with set A, then the non-faceted browser with set B. Another four are given the faceted browser with set B, then non-faceted with set A. Another four have A with non-faceted, then B with faceted. The final four have B with non-faceted, then A with faceted.

The time taken to complete the first four tasks for set A and set B are recorded. The fifth task in each set is more of an exploratory exercise to allow the user to form an opinion of each interface's effectiveness in exploring the data.

At the end of the browser trials, participants are given a short questionnaire to complete in order to evaluate their experience of each interface. The questionnaire asks:

1. Which browser did you prefer? (Faceted or Single Hierarchy)
2. Which browser allowed you to find the answers more quickly? (Faceted or Single Hierarchy)
3. Which browser was better for finding artists you liked? (Faceted or Single Hierarchy)
4. How would you rate the usefulness of the faceted browser? (On a scale of 1 to 10)
5. How would you rate the usefulness of the single hierarchy browser? (On a scale of 1 to 10)
6. Do you have any comments on either browser interface?

3.3 Results

A bar chart of the mean times taken by participants is shown in figure 2a. The mean time taken to complete a task using the faceted browser was 31.3 seconds. Compared with the mean of 61.1 seconds for the single hierarchy browser, this is a 51% reduction in time required for tasks.

Fourteen out of sixteen (87.5%) preferred the faceted browser, with fifteen out of sixteen (93.75%) stating that the faceted browser both allowed them to find the answers more quickly, and that it was also better for finding the artists that they liked.

The faceted browser received a mean usefulness rating of 8.6, while the single hierarchy browser only had a mean score of 4.9. Figure 2b shows the individual ratings of each participant.

The two people that preferred the single hierarchy browser stated that they saw it as more of a ‘challenge’ or ‘game’, and therefore enjoyed trying to work out how to work around the limitations of the interface.

The categories in the hierarchy are sorted by popularity, with the categories describing the greatest number of artists at the top. Some users commented that an alphabetical ordering of categories would have been more intuitive. This is discussed further in section 5.

4 Related Work

4.1 Automated Hierarchy Generation

Automatic clustering of documents into a hierarchy for search purposes is a well-established research topic. Willett [17] presents an excellent review of such methods. Generally these methods work on a fixed corpus of documents, generating a single tree-structured hierarchy that is based on the words that occur within them.

Cutting et al’s *Scatter/Gather* [18] is an early approach to hierarchy generation. While previous clustering methods would group documents according to a single shared attribute (*monothetic clustering*), *Scatter/Gather* uses *polythetic clustering*, where a document is only in a cluster if it contains enough of the terms that define the cluster, where the terms are taken from the frequencies of words that appear in the document. So while *monothetic* clusters could be described with one word (for example, *technology*), *polythetic* clusters would be described with many tags (*battery california technology mile* [18]). *Scatter/Gather* consists of two steps: (i) in the *scatter* step, documents are organised into groups and short summary ‘labels’ are presented, then (ii) when the user selects groups, they are *gathered* together to form a sub-collection. This sub-collection is then scattered again, and the process repeats.

Later approaches make use of Sanderson and Croft’s subsumption method [12] to create hierarchies (see the explanation in section 2.7). Many methods for automatic hierarchy generation such as Dakka et al [19] and Schmitz [13] based their approaches on Sanderson and Croft’s subsumption idea.

Dakka et al [19] present many techniques for the automatic discovery of facets for the generation of faceted hierarchies. Their technique uses a machine learning classifier. However, this does not generalise well (sic), so they expand each keyword using their hypernyms drawn from WordNet [20]. Hypernyms are more general versions of a word, for example ‘feline’ would be the hypernym of ‘cat’ or ‘lion’, and ‘animal’ would be the hypernym of ‘feline’. This can be thought of as an ‘is-a’ relationship (‘cat’ is-a ‘feline’). Expanding the keywords using hypernyms allows for a more general, fuzzy interpretation of each word that can help encapsulate related words and synonyms. The

authors show that this helps their classifier to generalise better. Thus, a keyword such as ‘cat’ leads to a list of hypernyms such as *cat*, *feline*, *carnivore*, *mammal*, *animal*, *living being*, *object*, *entity*. This allows their classifier to generalise, since words that the classifier does not know can be looked up via WordNet and categorised. Using WordNet to look up words and their hypernyms can introduce issues with sense ambiguity, where a word can have multiple meanings. For example, ‘bass’ could mean a musical instrument, a type of fish, or musical tones in a certain frequency range. They overcome this by adding associated keywords and their hypernyms to each facet or category.

Dakka et al’s implementation also uses a merit-based ranking to determine the sorting of the top-level facets to display to the user, in which the time it would take for a user to find a category in the hierarchy is estimated. This is done by modelling the user by taking a random walk along the hierarchy. At each node in the hierarchy, the time taken to read the category name is added to the cost score, along with time to correct mistakes (there is a small probability of browsing the wrong subtree). The time required to reach the desired object in the hierarchy from the root node can be estimated from this model, and the categories are sorted according to this measure.

Stoica et al’s Castanet [21] algorithm also uses WordNet hypernyms along with Sanderson and Croft’s subsumption method to automatically generate hierarchical faceted metadata from textual descriptions of items. Their system uses WordNet *domains* as a cross-categorisation mechanism to generate facets. Domains in WordNet assign general category labels to groups of synonyms. Castanet counts how many times a domain can be used to describe terms that appear in the textual descriptions. A list of the most commonly occurring domains is built, and a person manually selects the best-suited domains for use as facets.

However, these approaches all focus on creating hierarchies from (text) documents, in contrast to the Linked Open Data sources that are the subject of our interest. RDF-based approaches focusing purely on hierarchical facet generation are rare in the literature, even though RDF naturally supports the creation of hierarchical metadata via, for example, the `rdfs:subClass` relation.

4.2 Faceted Search

Originally devised by Ranganathan for library classification [1], Vickery ([22], quoted in [23]) defines faceted classification as “the sorting of terms in a given field of knowledge into homogeneous, mutually exclusive facets, each derived from the parent universe by a single characteristic of division”.

The classification technique involves the creation of several distinct facets (such as ‘year’, ‘cast’ and ‘genre’ in the case of films, as described in section 1). A list of items or a hierarchy of related subcategories are grouped under each facet.

An early example of a faceted search interface is the FLAMENCO browser created by Hearst et al [3], implementing an interface which supports both direct search and faceted browsing. A navigation bar shows a list of facets to the left of the screen, along with the number of elements sorted into that facet. When a user clicks on a facet, the interface displays a ‘matrix view’ of all the photos of buildings in that category from an architectural database of 40,000 images. From this view, the user can select a photo, then add a subcategory from that photo’s list of tags. This category is added to the search

query with the originally selected facet at the top of the page. At any time, a user can input a direct text search to add that to the query.

The approach creates the facets based on the hierarchical metadata in the image database's ontology. This means that since the facets already exist in the ontology, there is no need to create them using any special approach. The FLAMENCO browser is more concerned with the presentation of a faceted browsing experience rather than the generation of facets and hierarchies.

Mäkelä et al's 'Veturi' interface [24] develops the FLAMENCO browsing paradigm further, improving navigation of the hierarchy by allowing the user to drill down into subcategories. The faceted approach has also been extended by allowing queries and browsing based on sets of facets or tags [25]

Personalising the facets to suit the preferences of different users of a system is also explored in the literature. Faceted hierarchies have been created by creating computational models of users [26] or facets based on what users think are most intuitive [27].

5 Conclusions and Future Work

The preliminary study of the faceted search interface showed overwhelming evidence of its effectiveness for exploring foreign cultural data. The vast majority of users preferred using faceted search with textual search over just textual search with a single hierarchy. Furthermore, most participants found the faceted search not only better for completing the tasks set in the study, but also for free exploration of the data.

During the course of the study, one or two areas for improvement were suggested by the participants. For example, some users had difficulty with distinguishing the difference between the 'genre' and 'medium' facets. They would try to find watercolour artists in the genre facet, for example. A further study could be designed to see whether or not it would be worth combining these into a single facet.

Another problem is that some tags were not intuitive for the use of textual search. For example, users would often be confused when typing 'French' into the search box would yield no results. The reason is that the artists were tagged with only a 'France' tag, but no 'French' tag. A search for 'minimalist', for example, would be similarly fruitless given that artists belonging to that movement are tagged with 'minimalism' only. Fuzzy search, or perhaps a more robust tagging system, could be implemented to solve this problem.

Most users relied heavily on the textual search filter, especially in the case of the single hierarchy browser. As the search only worked for one tag at a time, some users suggested implementing multiple keyword search as an improvement. Though this would have improved the interface, it could arguably reduce the number of serendipitous discoveries that could be made by exploring a faceted (or otherwise) hierarchy.

Though the study proved the effectiveness of faceted search in exploring cultural data, future work is needed to determine ways of choosing optimal facets for this kind of browser. An automated way of doing this would be ideal, though more studies could also be done to test the differences in effectiveness between manually chosen facets.

References

1. Ranganathan, S.R.: Prolegomena to library classification. (1967)
2. Pollitt, A.S.: The key role of classification and indexing in view-based searching. Technical report, Citeseer (1998)
3. Hearst, M., Elliott, A., English, J., Sinha, R., Swearingen, K., Yee, K.P.: Finding the flow in web site search. *Communications of the ACM* **45**(9) (2002) 42–49
4. Yee, K.P., Swearingen, K., Li, K., Hearst, M.: Faceted metadata for image search and browsing. In: Proceedings of the SIGCHI conference on Human factors in computing systems, ACM (2003) 401–408
5. Oren, E., Delbru, R., Decker, S.: Extending faceted navigation for rdf data. In: The Semantic Web-ISWC 2006. Springer (2006) 559–572
6. Matsumura, F., Kato, F., Kamura, T., Ohmukai, I., Takeda, H.: Generating lod from web: A case study on building integrated museum collection data. In: The Second Joint International Semantic Technology Conference, JIST 2012 Nara, Japan, December 2012 Poster and Demonstration Proceedings. 23
7. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: Dbpedia: A nucleus for a web of open data. In: The semantic web. Springer (2007) 722–735
8. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: Dbpedia. <http://www.dbpedia.org/About> (2014) [Online; accessed 13-January-2014].
9. NII: Dbpedia japanese. <http://ja.dbpedia.org> (2014) [Online; accessed 13-January-2014].
10. Keio: Japanese wikipedia ontology. www.wikipediaontology.org (2014) [Online; accessed 13-January-2014].
11. Kwasnik, B.H.: The role of classification in knowledge representation and discovery. *Library trends* **48**(1) (2000) 22–47
12. Sanderson, M., Croft, B.: Deriving concept hierarchies from text. In: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, ACM (1999) 206–213
13. Schmitz, P.: Inducing ontology from flickr tags. In: Collaborative Web Tagging Workshop at WWW2006, Edinburgh, Scotland. Volume 50. (2006)
14. Heymann, P., Garcia-Molina, H.: Collaborative creation of communal hierarchical taxonomies in social tagging systems. (2006)
15. Thompson, M.: Artfinder web site. <http://cblop.github.io/artfinder> (2014) [Online; accessed 4-March-2014].
16. Thompson, M.: Artfinder source code. <https://github.com/cblop/artfinder> (2014) [Online; accessed 4-March-2014].
17. Willett, P.: Recent trends in hierarchic document clustering: a critical review. *Information Processing & Management* **24**(5) (1988) 577–597
18. Cutting, D.R., Karger, D.R., Pedersen, J.O., Tukey, J.W.: Scatter/gather: A cluster-based approach to browsing large document collections. In: Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval, ACM (1992) 318–329
19. Dakka, W., Ipeirotis, P.G., Wood, K.R.: Automatic construction of multifaceted browsing interfaces. In: Proceedings of the 14th ACM international conference on Information and knowledge management, ACM (2005) 768–775
20. Fellbaum, C.: WordNet. Springer (2010)
21. Stoica, E., Hearst, M.A., Richardson, M.: Automating creation of hierarchical faceted metadata structures. In: HLT-NAACL. (2007) 244–251

22. Vickery, B.C., y Oficinas, A.d.B.E.: Faceted classification: a guide to construction and use of special schemes. Volume 3. Aslib London (1960)
23. Star, S.L.: Grounded classification: Grounded theory and faceted classification. *Library Trends* **47**(2) (1998) 218–32
24. Mäkelä, E., Hyvönen, E., Sidoroff, T.: View-based user interfaces for information retrieval on the semantic web. In: *Proceedings of the ISWC-2005 Workshop End User Semantic Web Interaction*. Volume 7. (2005)
25. Huynh, D.F., Karger, D.: Parallax and companion: Set-based browsing for the data web. In: *WWW Conference*. ACM, Citeseer (2009)
26. Koren, J., Zhang, Y., Liu, X.: Personalized interactive faceted search. In: *Proceedings of the 17th international conference on World Wide Web*, ACM (2008) 477–486
27. Suominen, O., Viljanen, K., Hyvänen, E.: User-centric faceted search for semantic portals. In: *The Semantic Web: Research and Applications*. Springer (2007) 356–370