# CLUSTERING VERSUS FACETED CATEGORIES FOR INFORMATION EXPLORATION

By MARTI A. HEARST

Information seekers often express a desire for a user interface that organizes search results into meaningful groups, in order to help make sense of the results, and to help decide what to do next. A longitudinal study in which participants were provided with the ability to group search results found they changed their search habits in response to having the grouping mechanism available [2].

There are many open research questions about how to generate useful groupings and how to design interfaces to support exploration using grouping. Currently two methods are quite popular: *clustering* and *faceted* categorization. Here, I describe both approaches and summarize their advantages and disadvantages based on the results of usability studies.

**Clustering** refers to the grouping of items according to some measure of similarity. In document clustering, similarity is typically computed using associations and commonalities among features, where features are typically words and phrases [1]. One of the better implementations of clustering of Web results can be found at Clusty.com.[1]

The greatest advantage of clustering is that it is fully automatable and can be easily applied to any text collection. Clustering can also reveal interesting and potentially unexpected or new trends in a group of documents. A query on "New Orleans"' run on Clusty.com on Sept. 16, 2005 (shortly after the devastation wreaked by Hurricane Katrina), revealed a top-ranked cluster titled *Hurricane*, followed by the more standard groupings of *Hotels, Louisiana, University*, and *Mardi Gras*.

Clustering can be useful for clarifying and sharpening a vague query, by showing users the dominant themes of the returned results [2]. Clustering also works well for disambiguating ambiguous queries; particularly acronyms. For example, ACL can stand for Anterior Cruciate Ligament, Association for Computational Linguistics, Atlantic Coast Line Railroad, among others. Unfortunately, because clustering algorithms are imperfect, they do not neatly group all occurrences of each acronym into one cluster, nor do they allow users to issue follow-up queries that only return documents from the intended sense (for example, "ACL meeting" will return meetings for multiple senses of the term).

[1]Some of Clusty's power comes from performing metasearch and showing only the top-ranked results. This function alone can produce improved results, since it combines the power and judgment of several different search engines' rankings.

An underappreciated aspect of clusters is their utility for eliminating groups of documents from consideration. This result is supported by participant comments found in several studies [2, 3]. For example, if most documents in a set are written in one language, clustering will very quickly reveal if a subset of the documents is written in another language.

The disadvantages of clustering include their lack of predictability, their conflation of many dimensions simultaneously, the difficulty of labeling the groups (Clusty.com's top-level labels are among the best implementations), and the counterintuitiveness of cluster subhierarchies. Some algorithms [2, 8] build clusters around dominant phrases, that make for understandable labels, but whose contents do not necessarily correspond to those labels.

To illustrate these weaknesses, consider a recipe example, chosen because the relevant dimensions are familiar to most people and because exploration and browsing are natural tasks for recipe collections. A search for "chicken recipes" on Clusty.com (also on Sept. 16, 2005) turns up the following motley assortment of groups:

*Salad*
*Crockpot*
*Chicken Breast*
*Barbeque/Grilled*
*Soup Recipes*
*Healthy*
*Lowfat*
*Easy Chicken Recipes*
*Italian*

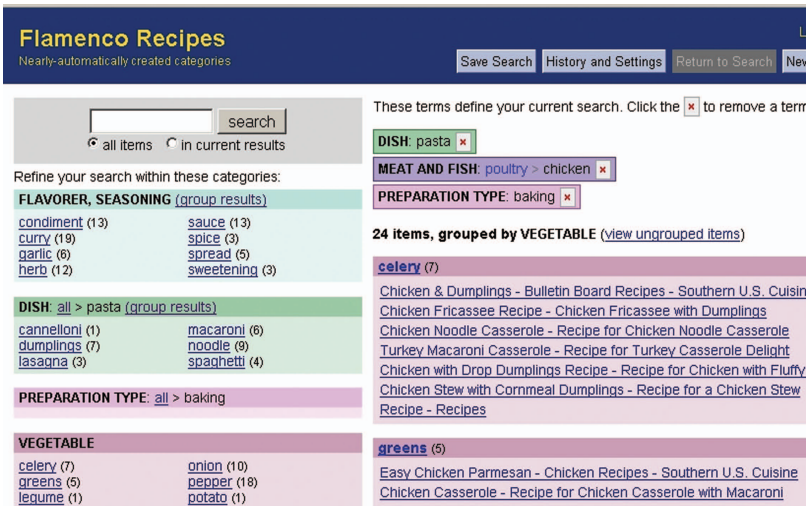
This list is incomplete and inconsistent. Why *Crockpot* and *Barbeque/Grilled*, but not *Baked* and *Fried*? Why *Chicken Breast* but not *Leg* and *Wing*? Why *Salad* and *Soup* but not *Main course*? Why *Italian* recipes but not *Indian*, *Thai*, or *French*? Furthermore, drilling down into the hierarchies rarely reveals intuitive results. The 29 documents listed under *Salad* are organized by the labels:

*Complete selection of Trusted Chicken Recipes*
*Cakes*
*Better Homes and Gardens*
*Collection*
*Share*
*Boneless Chicken Breast*
*Pasta Salad*

and so on. Only *Pasta Salad* really belongs here as a label; it does not make sense for *Boneless Chicken Breasts* to appear in this cluster rather than in the *Chicken Breasts* cluster, and clearly *Cake* belongs in a *Dessert* category alongside *Salad* and *Soups.*

These kinds of errors are quite typical for clustering output. Usability results show that users do not like disorderly groupings like these, preferring understandable hierarchies in which categories are presented at uniform levels of granularity [4, 5].

**Hierarchical Faceted Categories.** A category system is a set of meaningful labels organized in such a way as to reflect the concepts relevant to a domain. They are usually created manually, although assignment of documents to categories can be automated to a certain degree of accuracy. Good category systems have the characteristics of being coherent and (relatively) complete and thus pose an advantage over the unpredictable results of clustering; the studies that compare the two find that participants prefer categories [4, 5].




**Navigating a recipes collection using hierarchical faceted metadata (partial results).**

A question arises as to what kind of category structure is most effective for exploration and browsing of information collections. There is increasing recognition that strictly hierarchical organization of categories is impoverished for these uses.

An alternative representation, intermediate in complexity and very rich in flexibility, has become influential over the last few years. This representation is known as hierarchical faceted categories (HFC) [7]. The main idea is quite simple. Rather than creating one large category hierarchy, build a set of category hierarchies each of which corresponds to a different facet (dimension or feature type) relevant to the collection to be navigated. In the case of chicken (and other) recipes, these category hierarchies can include Dish Type (*Main, Soup, Salad, Side, Dessert*), Ingredi-

ent Type (*Meat, Vegetables, Grains, Spices*), Cooking Method (*Bake, Fry, Grill, Easy*), Cuisine Type (*Italian, Indian, French*), and so on. Each facet has a hierarchy of terms associated with it.

After the facet hierarchies are designed, each item in the collection can be assigned many labels from the hierarchies. Thus a recipe for "Chicken Noodle Casserole" might be assigned:

*Dish Type > Pasta*
*Preparation Type > Baking*
*Meat > Poultry > Chicken*
*Vegetables > Celery*
*Vegetables > Carrot*

and so on. Our research group has been investigating how to build an intuitive interface for exploration and discovery within information collections using HFC; we call the resulting interface framework Flamenco [7] (flamenco.berkeley.edu).

This kind of interface allows flexible ways to access the contents of the underlying collection. For example, from the *Meat* facet, a user can choose to select the *Poultry* subcategory, and from this select in turn the *Chicken* subcategory. The user can choose any other facet, perhaps Dish and Courses, and from this select the *Pasta* category, and then group the resulting recipes by *Vegetables,* or *Preparation Type*, or any other facet (see the accompanying figure). Navigating within the hierarchy naturally builds up a complex query that is a conjunction of disjunctions over subhierarchies.

An interface using HFC simultaneously shows previews of where to go next, and how to return to previous states in the exploration, while seamlessly integrating free text search within the category structure. The approach reduces mental work by promoting recognition over recall and suggesting logical but perhaps unexpected alternatives at every turn, while at the same time avoiding empty results sets. This organizing structure for results and for subsequent queries can act as scaffolding for exploration and discovery.

We have conducted a series of usability studies that find that, for browsing tasks especially, HFC-enabled interfaces are overwhelmingly preferred over the standard keyword-and-results listing interfaces used in Web search engines [7]. Study participants find the design easy to understand, flexible, and less likely to result in dead ends.

One drawback of HFC interfaces (as opposed to clusters) is that the categories of interest must be known in advance, and so important trends in the data may not be shown. But by far the greatest drawback is the fact that in most cases the category hierarchies are built by hand and automated assignment of categories to items is only partly successful.

Our group has recently made some progress in the problem of nearly automatic creation of hierarchical faceted categories [6]. A portion of the output of the system applied to the text of a recipe collection is shown in the figure. The algorithm, which makes use of the WordNet hierarchy, draws out detailed categories for ingredients, dishes, and (unexpectedly) cooking equipment and people, but misses facets such as cuisine. We call the algorithm nearly automated, since the results require some editing by hand. There is much room for improvement, and we see automatic creation of faceted hierarchies as an important area for research.

## Impact and the Future

To date, both HFC and clustering are boutique search interfaces; they are applied and used primarily in domain-specific collections. There are many movements afoot to promote larger scale use of metadata more generally. Hierarchical faceted metadata is already common in many e-commerce interfaces; for example, eBay and Shopping.com are experimenting with different variations of the idea, and Endeca.com provides a custom solution. It is probably possible to automatically impose a faceted structure onto grassroots created tag collections such as those seen at Flickr. However, it is an open question whether these will eventually be widely and regularly used on the open-domain Web. C

## References


1. Cutting, D.R., Pedersen, J.O., Karger, D., and Tukey, J.W Scatter/Gather: A cluster-based approach to browsing large document collections. In *Proceedings of the 15th Annual International ACM/SIGIR Conference* (Copenhagen, Denmark, 1992), 318–329.
2. Kaki, M. FindexSearch result categories help users when document rankings fail. In *Proceedings of ACM SIGCHI* (Portland, OR, Apr. 2005).
3. Kleiboemer, A.J., Lazear, M.B., and Pedersen, J.O. Tailoring a retrieval system *for naïve users. In Proceedings of the 5th Annual Symposium on Document Analysis and Information Retrieval* (Las Vegas, NV, 1996)
4. Pratt, W., Hearst, M., and Fagan, L. A knowledge-based approach to organizing retrieved documents. In *Proceedings of 16th Annual Conference on Artificial Intelligence* (Orlando, FL, 1999).
5. Rodden, K., Basalaj, W., Sinclair, D., and Wood, K.R. Does organization by similarity assist image browsing? In *Proceedings of ACM SIGCHI 2001*, 190–197.
6. Stoica, E. and Hearst, M. Nearly automated metadata hierarchy creation. In *HLT-NAACL '04, Companion Volume,* 2004.
7. Yee, K.P., Swearingen, K., Li, K., and Hearst, M. Faceted metadata for image search and browsing. In *Proceedings of CHI 2003* (Fort Lauderdale, FL, Apr. 2003).
8. Zamir, O. and Etzioni, O. Grouper: A dynamic clustering interface to Web search results. In *Proceedings of WWW8* (1999).



**Marti A. Hearst** (hearst@sims.berkeley.edu) is an associate professor in the School of Information Management and Systems (SIMS) at the University of California, Berkeley.



This research was funded in part by NSF IIS-9984741. Contributing to this work was Ping Lee, Kevin Li, Emilia Stoica, Ame Elliot, Rashmi Sinha, and Kirsten Swearingen.