A Multiple-Perspective, Interactive Approach for Web Information Extraction and Exploration

Naureen Moon, Ya-Wen Hsu, Rahul Singh
Dept. of Computer Science, San Francisco State University, San Francisco, CA
numoon@sfsu.edu, logoin@sfsu.edu, rsingh@cs.sfsu.edu

Abstract

While increasing amounts of complex information are becoming available on the web, there is, beyond keyword-based search and listing of results, a paucity of user interface (UI) paradigms and implementations that support interaction, exploration, and assimilation of information. This paper describes our design of a novel framework to address this deficiency. The proposed framework supports both direct search behavior as well as more exploratory search strategies through multiple-perspective visualization and interaction with search results. The approach is developed around the twin themes of supporting data context and facilitating effective interactions between users and data. The system supports data context through determination of semantic correlations between web pages and extraction of the spatio-temporal data contained therein. A multiple-perspective environment is then used to display semantic and spatio-temporal relationships as well as to provide intuitive views of the data, specifically through web page thumbnail, map, and timeline modules. The environment supports direct interactions with the data through a reflective interface by which user selections in any one panel highlight the corresponding information in other panels. In this environment, visual cues and explicit facilities to model space and time aid in recognition, querying, and exploration of information as well as in representation and reasoning with complex relationships (such as spatio-temporal, causal, evolutionary) in the data. Experimental studies of a quantitative and qualitative nature demonstrate the efficacy of the system in facilitating both information extraction and discovery.

1. Introduction

The World-Wide-Web has evolved to become one of the primary repositories of information available to us. The universal interface to this repository is through web-search technology, the development of which has, till now, occurred under the predominant emphasis of retrieving the most relevant results for a query and presenting the results in a manner so as to improve retrieval efficiency. Today, the state-of-the-art attempts to satisfy these goals through extensive indexing of web sites, key word-based analysis of page content, analysis of hyperlink and usage patterns using concepts such as link popularity, and finally, a listing of the retrieved results based on page ranking. The success of current systems in retrieving results for well defined information-lookup type queries is impressive and underscores the popularity of search technologies.

These same methodologies for determining page relevance may, however, lead to results that are biased or unrepresentative of the available information. [9] provides examples of this phenomenon using two specific sample queries: “flowers” and “apple.” In the first case, the vast majority of top results are online florists. This is representative of a general trend for searches that involve products available for sale online, whereby results related to shopping are ranked highly. The “apple” query shows further shortcomings of using page-ranking alone to determine relevance. The results in this case are heavily biased towards “Apple Computer” due to the disproportionate online presence of the technologically apt. This reflects a general trend for polysemous queries, where different meanings of a term are unrepresented at worst, or displayed interspersed with each other at best. The latter case may be observed for the query “eclipse” which shows results for solar and lunar eclipses scattered among results for the Eclipse software development environment.

Moreover, the emergence of the web as a ubiquitous resource has impacted the nature of queries. [17] posits that the notion of an “information need” is insufficient to describe modern web search behavior, which is better described using a trichotomy of navigational, informational, and resource. The first of these refers to the situation in which a user is looking for a specific website. The second is significantly broader, encompassing such things as: (1) directed queries to learn something specific about a topic; (2) undirected queries to simply acquire more information about a topic; and (3) search for suggestions for further research directions. The last category, resource, includes goals such as downloading, entertainment, or interaction.

It is evident that most current search technology is inadequate to facilitate the majority of these goals. To use a simple example, users with a navigational goal are seeking a specific website. According to [22], 58% of web page accesses are to pages previously seen. In addition, humans process images much more rapidly than text [26]. Why then are search results so heavily biased

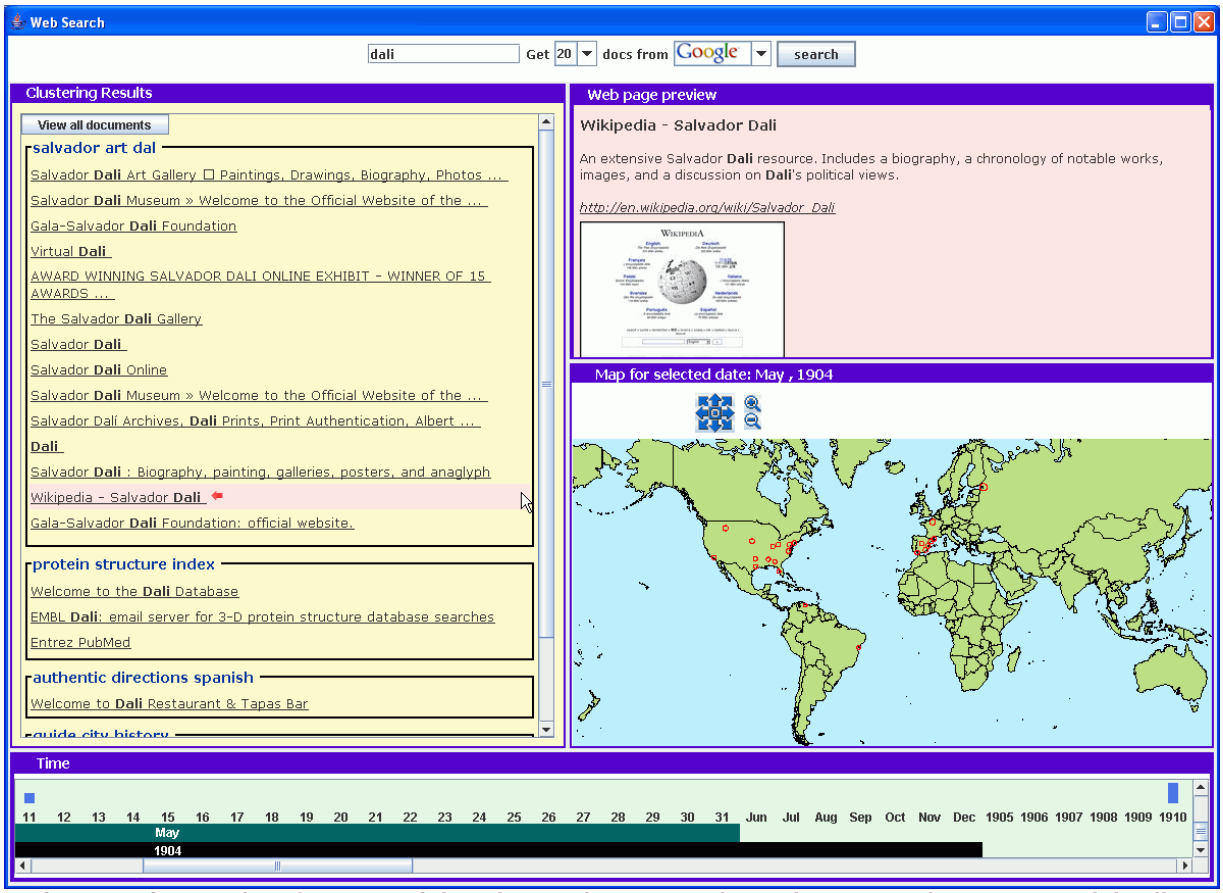


Figure 1. The user interface, containing clustered query results, web page previews, map, and timeline

towards textual rather than visual cues? Furthermore, while current systems are generally quite good at information extraction (directed queries), they fail to assist users in exploring the information available on a topic (undirected queries) or interacting with it.

We therefore propose a conceptually different way to approach web-search that allows users to view and interact with the information space relevant to the query from multiple perspectives including semantic similarity of content, temporal characteristics of the information, spatial characteristics of the information, as well as perceptual (appearance-based) characteristics of the relevant web-pages. Any subsequent interactions with the data can be carried out in through one (or a combination) of these "views". Furthermore, the proposed approach emphasizes an integrated presentation and interaction with results across each view. This holistic perspective provides a powerful mechanism for discerning patterns and relationships in the data as well as for supporting exploratory interactions with it.

Our approach involves solving the following core challenges:

Supporting data context by determining semantically correlated web pages and identifying other semantically meaningful perspectives: We use Latent Semantic Analysis (LSA) and term frequency-inverse document frequency (TFIDF) weighting on the textual content to identify pages that are semantically-related. A two-stage clustering method is then used to group related hits. Dynamically recognizing these relationships allows us to support data context, present an overview of the information space, and manage the presentation of large number of hits by grouping those that are related.

In addition to the above semantic clustering, we utilize the spatio-temporal characteristics of information related to a query as generic views of the data. The content of each web page is analyzed to identify any available geographical and temporal information. A significant percentage of the test questions in the NIST TREC 2004 Question Answering track [24] relate to location or time data ("when," "where," "in what year," etc.). This underscores the value of location and time as contextual clues for both retrieval and assimilation of information. Moreover, discerning and displaying such spatio-temporal characteristics help in understanding relationships that underlie the data. Our choice of space and time as additional perspectives on the data is also motivated by experiments indicating that spatio-temporal interfaces support highly intuitive and rapid exploration and access of the retrieved data [20].

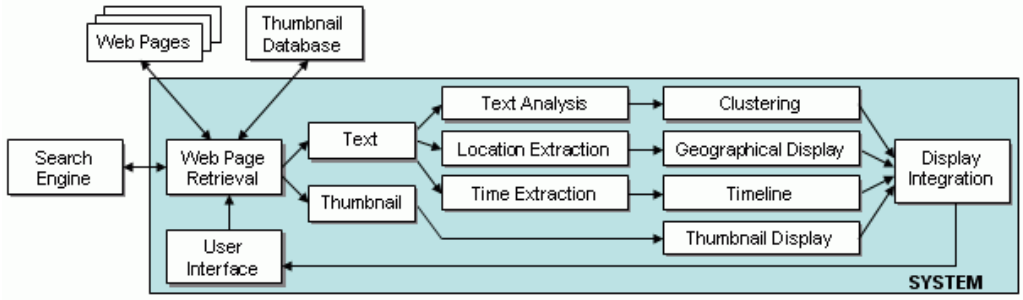


Figure 2.The system architecture

Supporting efficient and effective interactions between users and the retrieved information: We propose an interface that employs direct manipulation techniques [8] to facilitate natural user-system interactions (Figure 1). One component of the interface consists of previews of retrieved web pages in the form of thumbnails. Both the temporal and spatial displays allow users to examine information at varying resolutions of time and space. Furthermore, in the interface various views of the data (such as those across time, space, and semantically related groups) are tightly linked to each other, so that interactions in terms of any one of them are instantaneously reflected in all the views (for example, selecting a region of space leads all links related to it to be highlighted in all of the other views). This is essential for maintaining context.

We begin with an overview of prior research that has targeted the problem of information display and extraction in web search (Section 2). Our approach is presented in Section 3, followed by descriptions of the Text Analysis and Clustering modules in Sections 4 and 5. Sections 6 and 7 describe extraction and display of spatio-temporal information. We then present results and evaluations of our system in Section 8. The paper is concluded with a synopsis of our work in Section 9.

2. Previous work

The traditional approach to document correlation has been manual classification using a taxonomy, or hierarchy of descriptors. With regard to web pages, this approach, although semantically very meaningful, is fundamentally limited because: (1) developing taxonomies and assigning them to web pages is very costly [25] and (2) this approach cannot keep pace with the rapidly increasing number of web pages. The Open Directory Project overcomes these limitations to some degree through taxonomy-based classification using a global community of volunteer editors. The value of ODP is underscored by its use in Google Directory and AOL Search.

Due to these limitations, various researchers in both industry and academia have utilized linguistic techniques to develop alternate forms of presentation for web search results through automated clustering. [2] and [5] provide surveys of recent approaches to the problem of grouping search results. These include a number of engines that perform on-the-fly clustering, the best known of which is Vivisimo [25]. Other efforts include [2], which proposes hierarchical clustering of results using web page summaries, or "snippets." A conceptually different approach to classifying web pages is suggested in [10] where a URL-based classification strategy is employed. The commercial product Grokker provides hierarchical clustering with the added feature of an interactive visual display of the results [4].

An alternate approach to organization of results uses automatic classification into predefined categories. One such effort is the search engine Northern Light, which attempts to organize results by clustering them into categories predefined in library sciences [13]. Clever Search [11] filters search results using WordNet [27] senses. WordNet-based approaches are fundamentally limited, however, for a number of reasons. Most salient among them is that many query strings (particularly those containing multiple words and/or proper names) are not included in the WordNet lexicon. Furthermore, when the query string is found, the desired sense might not be (for example, the two included senses of the word "apple" relate to the fruit, while "Apple Computer" is excluded).

Certain technical overlaps aside, our solution philosophy differs fundamentally from the above approaches. All of the above approaches propose summarizing hits from web search. In contrast, our goal is to support efficient and effective interactions between users and retrieved information in holistic ways. Additionally, our key goals also include supporting data context, providing information overview, and providing cues for proactive querying. Finally, in terms of quantitative criteria, such as access complexity, the reflective multimodal presentation-interaction environment proposed by us performs significantly better than the aforementioned approaches.

3. Overview of the system

The system consists of several interactive modules (Figure 2). First, the Web Page Retrieval module obtains the query from the user interface and submits it to a search engine. Each of the web pages in the results list is

accessed for its textual content and its thumbnail (when available) is retrieved from [1]. The text is then processed by each of three modules: Text Analysis, Location Extraction, and Time Extraction. The first module preprocesses the text, extracts key terms, and builds a term frequency matrix, which is subsequently adjusted and weighted as described below. The resulting feature vectors are used to calculate document correlation values. These are passed to the Clustering module, which first uses the correlation values to construct clusters and then merges similar clusters using web page categorization information from the Open Directory Project (ODP) [14]. The clustered web pages are displayed with appropriate labels in one panel of the user interface.

The Location Extraction module extracts geographical information from the text of each web page through cross-referencing with a comprehensive gazetteer of world locations [28]. The Time Extraction module extracts temporal information by searching the text for common date formats. The data obtained from each of these modules is subsequently displayed on the Map and Timeline components of the user interface. The aforementioned thumbnails are displayed on another panel of the interface.

A major feature of the user interface is the reflectivity of the different views, so that selecting information in one highlights its corresponding characteristics in the others. For example, selecting a cluster of documents displays only the thumbnails associated with the web pages it contains. Similarly, only the spatio-temporal data for these web pages is shown on the map and timeline. This encourages multiple-perspective user interaction with the data and enables information assimilation using both visual and textual cues. Additionally, it allows users to explore the information for relationships and trends.

4. Determining correlated web pages

4.1 Preprocessing

The textual content as well as meta-data (empirically determined to yield better results than the text alone) of each web page is first preprocessed to remove html tags, punctuation, capitalization, non-English words, and numbers. The text is then stop-word filtered to remove terms with low semantic content (e.g., “our” and “this”). In the next step, the Porter stemming algorithm is executed on the terms, by which they are truncated to their pseudoroots (explained below) [16]. This allows morphological variants (e.g., “acts” and “actor”) to be viewed as the same term for frequency counting. The term lists for each document are then used to construct a term-frequency matrix, where each row denotes a term and each column a document. In the final stage of preprocessing, terms appearing in all documents or in only one are removed due to being of no value in discerning relationships between documents.

4.2 Term frequency adjustment

We experiment with a number of term frequency adjustment schemes in order to maximize both the correlation between similar documents and the distance between dissimilar documents (with the ultimate goal of obtaining semantically-meaningful clusters). We use Latent Semantic Analysis (LSA) and Term Frequency-Inverse Document Frequency (TFIDF) weighting as the base cases and then combine the techniques to improve the results. In order to compare different system configurations/parameters, we begin by manually clustering search results for a given query to obtain ground-truth clusters. For each document we then calculate its average correlation with documents in the same ground-truth cluster and its average correlation with documents outside the cluster. We define the best optimal configuration as that which maximizes the in-cluster and out-of-cluster difference.

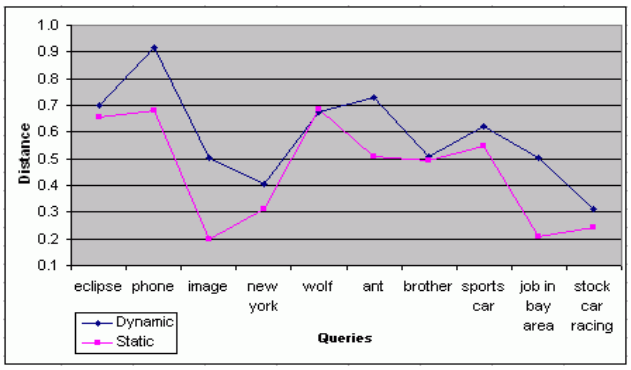
4.2.1 Latent Semantic Analysis. LSA is a well-known statistical method for inferring word meanings by the contexts in which they are appear [12]. It is especially useful for overcoming problems of synonymy and polysemy. The technique uses singular value decomposition (SVD) to decompose the term-frequency matrix *{X}* as follows:

$$\{X\} = \{W\}\{S\}\{P\}^T \quad (1)$$

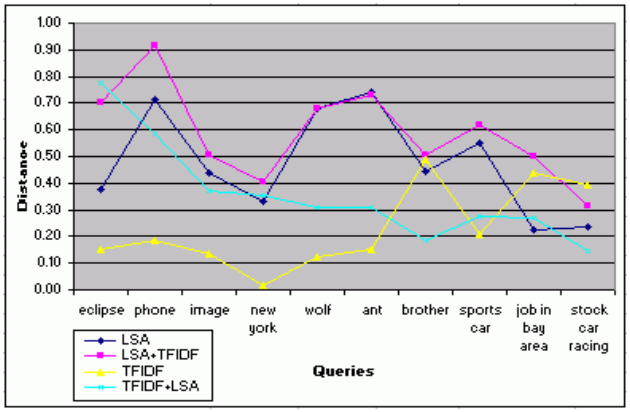
where *{S}* is a diagonal matrix of scaling values. The dimensionality of the solution is then reduced by truncating *{S}* and thereby retaining only the largest eigenvectors. This step serves to reduce the amount of noise in the data and bring forth latent correlations. The term-frequency matrix is then reconstructed. An additional consideration is that the final result is very sensitive to the number of eigenvectors. We empirically determined that calculating the difference between contiguous eigenvalues and truncating the *{S}* matrix at the location of the largest drop yields consistently good correlation values. Due to this result, we use this method to determine the number of eigenvectors to retain in all subsequent experiments.

4.2.2 Term Frequency-Inverse Document Frequency. TFIDF is a term weighting scheme that down-weights common terms and emphasizes uncommon terms [18]. The technique is motivated by the observation that terms uncommon in a given domain give better discernment between documents in that domain. The TFIDF value of a term is given by:

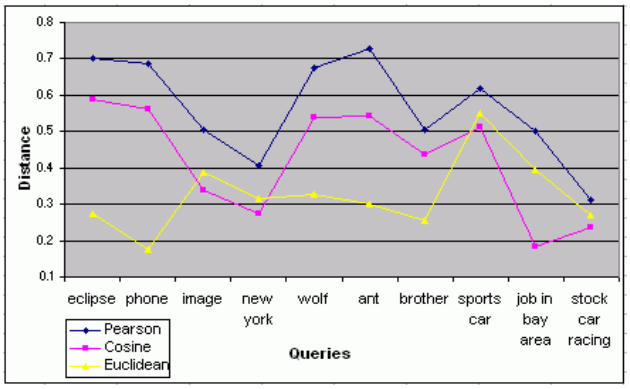
$$TFIDF(i, j) = TF(i, j) \times \log_{10} \frac{N}{DF(i)} \quad (2)$$



A



B



C

Figure 3. Comparison of (A) background sets, (B) LSA, TFIDF, and combinations, and (C) similarity measures

where *TF(i,j)* is the frequency of term *i* in document *j* and the second term is known as the inverse document frequency (IDF) of *i*, calculated using a (generally large) background set of documents from the domain of interest. *N* is the total number of documents in the background set, and *DF(i)* is the number of background set documents in which the term *i* occurs.

The feature vector for each document may optionally be normalized prior to weighting. [18] suggests the use of normalization in cases where there is large variability of vector lengths to eliminate bias towards larger documents. Despite the fact that we use only 100 words from each document after stop-word filtering (meta-data plus terms from the beginning of the document, empirically determined to give correlation values as good as greater numbers of words), this variability does exist because many web pages contain very little text. We therefore use a simple normalization that divides each term frequency by the total number of words in the document.

4.2.3 TFIDF on LSA. We propose to combine TFIDF and LSA to reap the benefits of each technique and thereby achieve better correlation values than either utilized alone. Specifically, the intuition lies in using LSA to map the data to an eigenspace and thereby expose correlations and then to apply TFIDF to augment the prominence of important terms. For purposes of comparison, we also combine the techniques in the reverse order (as done in [6] which shows better precision-recall results for LSA applied after TFIDF as opposed to LSA used on plain term frequencies).

Using a test set of 20 web pages, we first experiment with different choices of background sets for TFIDF applied on top of LSA. Specifically, a static background set of 1656 web pages is compared with a small, dynamically-generated background set consisting of the first 50 search results. The latter effectively uses a more limited definition of the problem domain, defining it as those web pages retrieved as a consequence of the query. This is motivated by the heterogeneity of information on the web and the observation that terminology that is common in one field may be very esoteric in another. For example, while the term "compound" is somewhat uncommon in normal English usage, the opposite would be the case in chemistry and finance. As a result, it would be less valuable for discerning relationships between documents in these domains. As expected, using the differences between average in-cluster and out-of-cluster values shows the dynamic background set to indeed yield better and more consistent results (Figure 3(A)) for which reason it is utilized in subsequent experiments.

Using the same measure, comparing results obtained from LSA, TFIDF, and their two combinations shows consistently better values for TFIDF applied after LSA (Figure 3(B)). We therefore consider this the optimal configuration.

4.3 Determining document similarity

We compare 3 similarity measures: Pearson, cosine, and Euclidean. For two vectors y_a and y_b, the [0,1]-normalized Pearson correlation is given by [21]:

$$s^{(P)}(y_a, y_b) = \frac{1}{2}\left(\frac{(y_a - \bar{y}_a)^T (y_b - \bar{y}_b)}{\|y_a - \bar{y}_a\|_2 \cdot \|y_b - \bar{y}_b\|_2} + 1\right) \qquad (3)$$

where $\bar{y}$ is the average feature value of vector y and 1 denotes perfect similarity.

The cosine similarity uses the angle between vectors to determine their similarity. This insensitivity to vector length makes it a popular measure for calculating document similarities. Since LSA induces negative term frequency values, we similarly normalize the cosine measure:

$$s^{(C)}(y_a, y_b) = \frac{1}{2}\left(\frac{y_a^T y_b}{\|y_a\| \cdot \|y_b\|} + 1\right) \quad (4)$$

where 1 again denotes perfect similarity and 0 total dissimilarity.

In contrast, the Euclidean metric measures the distance between vectors. Consequently, it is unbounded on the positive side and perfect similarity is given by a distance of 0. To compensate for these characteristics and thereby cast the Euclidean metric into a form suitable for comparison with Pearson and cosine, we utilize a [0,1]-normalized form, which may be computed as [21]:

$$s^{(E)}(y_a, y_b) = \frac{1}{1 + \|y_a - y_b\|_2} \quad (5)$$

Again using the in-cluster and out-of-cluster difference, the results show that Pearson performs considerably better than the cosine and Euclidean measures (Figure 3(C)). For this reason, the Pearson correlation is used to generate clustering results.

5. Presentation of correlated pages

The presentation of correlated pages is done by grouping them in labeled clusters obtained from the Clustering module, which groups the documents in two stages and labels the resultant clusters. The first stage is driven by the correlation data and uses a variant of the *K*-means algorithm to generate clusters. In the second stage, web page category information from ODP is used to merge similar clusters. The display of search results as semantically-meaningful clusters is particularly conducive to understanding the information space when query terms are polysemous (as in the aforementioned "apple" query) or when the results show thematic associations (such as florists and botany for the "flowers" query).

5.1 Data-driven clustering

We first employ a variant of the *K*-means clustering algorithm to perform clustering of the documents based on correlation values. We adapt the basic *K*-means because, while these algorithms are efficient, they are severely limited by the need to specify *K* (the number of clusters) in advance. Since the appropriate number of clusters is a function of the search results, it is impossible to use a fixed value of *K*. We therefore use a data-driven approach to determine *K* as follows. As an additional modification, we employ the *K*-medoids algorithm [7], which is a variation of the *K*-means algorithm that uses actual data points as centers of clusters. An implementation advantage of the *K*-medoids algorithm lies in that it does not require re-computation of the distance between objects and cluster medoids every time the medoids change. This is due to the constraint that a medoid also be an object in the cluster [7].

The clustering algorithm takes as its input the table of correlation values and a threshold value *T* which is used to determine when two documents can be considered to be similar. For a given cluster, if any two objects in the cluster have similarity value lower than *T*, the cluster becomes a candidate for "splitting". Analogously, for two clusters, if all documents have pair-wise similarity values higher than *T*, the clusters become candidates for "merging." A cluster is defined to be "stable" if for any two documents in the cluster, the similarity value is higher than *T* and the similarity value of a document in this cluster with any document in another cluster is less than *T*. The algorithm is initialized with the value of $K = 1$ and *K* is incremented or decremented when a split or merge occurs. The point at which all clusters become stable automatically determines *K* and acts as a stopping criterion.

Let the symbol *S(i,j)* represent the similarity between documents *i* and *j*. Also let *FC* represent the final cluster set. The clustering algorithm is initialized with $K = 1$ and $FC = \{\}$. As the initial step, the entire data set is checked for stability. If the conditions for stability do not hold (that is, there is more than 1 cluster), the value of *K* is incremented and the following steps are executed:

- *Initialization of cluster centroids:* The cluster centroids are seeded by selecting the *K* points of highest density. The density-based initialization ensures that the clustering is focused on key-regions, rather than around sparse outliers.
- *Medoid identification:* Define each cluster medoid as the document for which the sum of similarity values to all other documents in the cluster is maximum:

$$m_c = \arg\max_i \sum_j S(d_i, d_j) \quad (6)$$

- *Voronoi Tessellation*: The documents are partitioned into Voronoi regions.
- Check the clusters for splitting and merging. Update the cluster set *FC* appropriately. Iterate the above steps till all clusters become stable.
- Output the final cluster set *FC*.

5.2 Category-driven clustering

ODP is a large, manually-constructed index of web sites constructed and maintained by a global network of volunteer editors who categorize web pages according to a defined taxonomy. For example, the web site for the

New York Stock Exchange is categorized as "Top/Business/Investing/Stocks and Bonds/Exchanges." This information allows us to group semantically similar documents together despite differences in data that may lead to low correlation values. We obtain category information (when available) for all search results and use this information as follows:

- Truncate the last level of the hierarchy for categories with depth greater than 7 levels.
- Merge clusters containing documents with the same categorization.
- Merge clusters containing documents with categorizations of ancestor/descendant nature (such as "Top/Business/Investing/Stocks and Bonds" and "Top/Business/Investing/Stocks and Bonds/Exchanges/Organizations") as long as the ancestor has depth greater than 3 levels.

It is noteworthy, however, that there is an absence of ODP categorization for about half of the search results we used to test the system. Other factors that limit the influence of this second stage of clustering include: (1) a lack of category matches across clusters and (2) documents with the same categories already appearing in the same cluster. While we initially over-cluster in the first stage by using a somewhat high threshold of 0.93, if any of these cases are found (no cluster merges due to the aforementioned reasons), we cluster again after dropping the threshold to 0.90 (because data-driven clustering must suffice in these cases). We empirically determined that this clustering scheme yields slightly better results than initially clustering at the lower threshold value.

5.3 Cluster labeling

Clusters are labeled using the 3 terms that appear in the most documents contained in the cluster. When multiple words satisfy the criterion, those with the highest TFIDF values are selected. However, the terms at this point have already undergone stemming in the preprocessing phase and so exist not in their original forms but as pseudoroots. This refers to root forms of words which may not be words themselves. For example, "rac" is the pseudoroot of "race" and "racing." Since pseudoroots cannot be used for labeling, we maintain two term lists when constructing the term-frequency matrix. The first contains the pseudoroots while the second contains the shortest word found that is truncated to the pseudoroot at the same index. This index allows the pseudoroots chosen as cluster labels to be mapped to real words.

6. Location extraction, display and interaction

The Location Extraction module cross-references the web page text with lists of world locations constructed using a comprehensive gazetteer [28] to extract names of countries, states/provinces (for the United States, Canada, United Kingdom, and Australia), and cities.

A major factor that degrades precision of location extraction systems is name ambiguity, which consists of two types: locations with the same name and locations that are also words. We deal with this issue by looking only for cities whose enclosing country or state is found in the document text. We also reduce the number of ambiguous cases by only including cities with populations of 10,000 or greater. An exception is made for major world cities, which are included without mention of their countries or states. The second type of ambiguity is also dealt with by treating only capitalized terms as locations.

The following steps are executed to extract the geographical associations of each document:

- The country is determined from the country code top-level domain (when available from the URL).
- Major world region names (such as the "Middle East" or "Southeast Asia") are extracted.
- An index of countries is used to extract countries mentioned in the document.
- An index of states is referenced to extract state and province names.
- Cities with populations over 10,000 in the found regions, countries, and states are extracted.
- Major world city names are extracted.

All locations extracted are saved as a hierarchy of continent, country, state/province, and city. All fields above the extracted name are populated using the information from the gazetteer.

The OpenMap Java toolkit [15] is used to implement display and interaction with geographical data. The gazetteer is used to map cities to their latitudinal and longitudinal coordinates for display and selection through OpenMap. Extracted cities are thereby indicated on the map, where circle size varies logarithmically with the number of documents containing that location [23]. The user may select areas of interest by clicking on the location or dragging a rectangle across the area. The list of extracted locations is parsed to determine which (if any) lie fully or partially within the selected area (or within 0.5 degrees of point selections). The user is then presented with the list of cities, states, and countries thus obtained, and may select one or all of them. The documents affiliated with the selected location are highlighted. Conversely, selecting a cluster of search results shows only the locations associated with the documents in the cluster on the map.

7. Time extraction, display, and interaction

The textual content of each web page is also parsed to extract any available temporal information. Specifically, this is done by pattern matching of the following date

formats (as well as slight variations) using regular expressions:

- dd/mm/yy | dd/mm/yyyy | mm/dd/yy | mm/dd/yyyy | mm/yyyy | mm/yy | mm/dd (when there is no ambiguity)
- dd MonthName yyyy | MonthName dd yyyy | MonthName yyyy | MonthName
- ‘yy | numbers preceding “BC”/“AD”/“BCE”/“CE” | numbers following the word “year”
- 4-digit numbers between 1700 and 2099

Dates without a year are assigned the one (if any) that appears within 13 characters after them. Else, they are removed from consideration.

Since the above list incorporates the majority of common date formats, recall of time extraction is quite high. Moreover, since it is very unusual to see the above patterns (except for the last) used for other than dates, precision of this component is very high as well.

The extracted data is displayed on a timeline, which supports different levels of granularity. This allows users to switch between year, month, and day views of the temporal information. If a user selects a date, the documents affiliated with that date are indicated. Similarly, selecting a cluster of results refreshes the timeline, showing only the dates related to the documents contained in the cluster.

8. Results

8.1 Display of multiple-perspective, reflective views

The sample query “Dali” is executed on the system with results as shown (Figure 1). Two major clusters are seen, the first referring to the painter “Salvador Dali” and the second showing the DALI protein database. These are labeled fairly accurately as follows:

- “salvador,” “art,” “dal”
- “protein,” “structure,” “index”

Selecting these clusters displays the corresponding set of thumbnails shown (Figure 4).

The remaining three clusters are singletons, containing web pages about “Dali restaurant,” “Dali travel guide,” and “Dali loudspeakers.” While the overall clustering is very accurate, the labeling is considerably worse in this case:

- “authentic,” “directions,” “spanish”
- “guide,” “city,” “history”
- “based,” “countries,” “sq”

This result may be understood as follows. Since all terms in singleton clusters occur in exactly one document, those with the highest TFIDF are selected as labels. Since uncommon terms have high IDF values, however, these are favored in label selection, which may yield inaccurate labeling of singleton clusters.

Figure 4. Thumbnails for (A) Salvador Dali cluster and (B) DALI protein database cluster

In the user interface snapshot shown, Salvador Dali’s birthday, May 11, 1904, has been selected on the timeline. This serves to highlight the document containing that date, the thumbnail of which is displayed. Locations contained in the document are highlighted on the map and contain Catalonia, Spain, Dali’s place of birth.

8.2 Analysis of clustering

The purposes of clustering documents include:

- Reduction of the cognitive load for the user by decreasing the number of links to peruse
- Providing an understanding of the information space by bringing related links/motifs together

As such, we evaluated the system vis-à-vis these objectives. To assess performance with respect to the first, we counted the number of clusters as a rough measure of the reduction in data size. Success in attaining the second objective was determined by appraising correctness of clusters.

The test set for the first two analyses consisted of 10 queries on a spectrum of real-world topics, while that of the third experiment used a 3-query subset of the same. The number of results was limited to 20 documents for ease of analysis, however, the system is capable of handling much greater volumes of data. Results were generated for TFIDF on LSA both with and without category-based clustering to quantify what improvement (if any) is obtained through the addition of a second clustering stage.

Table 1. Clustering evaluation for (A) LSA+TFIDF and (B) LSA+TFIDF with category information

Queries	No. of Clusters	% Data Reduction	No. of Docs Correctly Clustered	% of Docs Correctly Clustered
eclipse	9	0.55	19	0.95
phone	10	0.50	19	0.95
image	10	0.50	15	0.75
new york	12	0.40	17	0.85
wolf	9	0.55	12	0.60
ant	9	0.55	20	1.00
brother	9	0.55	14	0.70
stock car racing	11	0.45	17	0.85
sports car	5	0.75	17	0.85
amazon rainforest	13	0.35	16	0.80
Average	9.7	0.515	16.6	0.83

A

Queries	No. of Clusters	% Data Reduction	No. of Docs Correctly Clustered	% of Docs Correctly Clustered
eclipse	7	0.65	19	0.95
phone	6	0.70	19	0.95
image	7	0.65	15	0.75
new york	7	0.65	16	0.80
wolf	6	0.70	13	0.65
ant	5	0.75	20	1.00
brother	9	0.55	14	0.70
stock car racing	5	0.75	17	0.85
sports car	4	0.80	17	0.85
amazon rainforest	11	0.45	16	0.80
Average	6.7	0.665	16.6	0.83

B

The results for the first experiment (Table 1) show an average data size reduction of 67% with categories and 52% without, which was calculated as the percent difference between the number of clusters and the number of documents they contain. This level of decrease in data volume is clearly advantageous for the user, provided that the clustering of the documents is accurate.

In order to quantify accuracy of clustering, we manually analyze the coherence of the clusters obtained by submitting each of the 10 queries to the system. Documents that are irrelevant to the cluster to which they are assigned are tallied over all of the clusters for a given query to determine overall percent correctness of clustering, found on average to be 83% for both cases (Table 1). Since this approach clearly favors small clusters, it is critical to view these results in light of those from the previous experiment, which favors low thresholds or large clusters. It is noteworthy, however, that addition of categories improves data reduction (as expected) without degrading correctness. This indicates that, for these examples, the category-based merges are appropriate.

8.3 Evaluation of location and time extraction

Location and time extraction was evaluated by comparison with manual information extraction. The first 20 documents retrieved for each of 3 queries ("eclipse," "computer," and "cell phone") were used as the test set. All unambiguous location and date information in the documents was manually tagged and compared with the automatically-extracted information. Both modules show mid-90% values for recall, indicating that the vast majority of spatio-temporal information is extracted. Temporal and geographical data extraction has precision of 99% and 88%, respectively, showing that the extracted information is largely correct. The comparatively low value for location is due primarily to the aforementioned naming ambiguity.

8.4 User evaluations

Quantitative and qualitative user studies were used to evaluate the system. For the former evaluation, fifteen users each ran a set of 5 queries on our system as well as on Google, Vivisimo, and Grokker. The first 3 queries and their information goals were predefined with two designed to involve spatio-temporal information. The first query asked users to find one location where an eclipse predicted to occur in 2008 will be visible, the second involved obtaining nutritional information for peanuts, and the third asked for the year in which American pioneer Johnny Appleseed traveled through certain states. The last two queries were open in order to test the general applicability of our approach. Efficiency of information access was quantified by automatically recording the time and number of mouse clicks it took users to reach their information goal. Both measures are significant because the first uses speed to quantify efficiency while the second uses ease of access.

Our system yields better average values than the others. Particularly striking, however, are the standard deviations (Table 2). In fact, the most salient feature of the results seems to be the huge variability between users. The average seconds per click for each user varied from about 4 to 12, a threefold difference. In particular, two users were found to use thrice as many seconds and clicks, affecting the average and standard deviation significantly. Even they, however, were able to reach their information goal in less time using our application, which displays considerably lower values for the standard deviation in all cases but one (where it is comparable to Google). This result implies that our system is especially effective for users who do not access desired information quickly. Overall, our system performs considerably better than Vivisimo and Grokker and as well as Google. It is noteworthy that most users are novice users of our system but experienced users of Google.

Table 2. Average time and clicks

		Google	Vivisimo	Grokkker	Our System
Time (sec)	Average	79	81	129	65
	Std Dev	66	80	141	47
No. of clicks	Average	9.6	11	23	8.9
	Std Dev	9.7	13.7	28.4	8.4

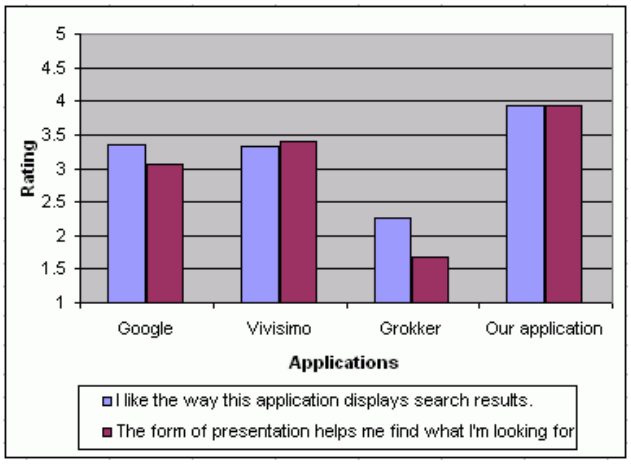


Figure 5. User ratings of 4 tested applications

Qualitative evaluations showed our interface and presentation of results receiving the highest score on average (Figure 5). In addition, the majority of users rated the components of our interface as useable and useful.

9. Conclusion

We have described a novel paradigm for user-information interaction in the increasingly ubiquitous domain of web search. The proposed system supports information extraction and exploration through a multiple-perspective framework, addressing two tasks:

- Determination and effective presentation of semantic relationships between web pages
- Design of a framework that supports direct interaction with the data

The first is achieved by: (1) computing similarities between documents and presenting related documents as clusters and (2) extracting and displaying spatio-temporal data. These views support data context and enable more rapid understanding of the information space. The latter is implemented through reflectivity of different views in the environment. Efficiency of access is facilitated by integration of web page thumbnails, which provide important cues for information preview and recognition.

The results show effective clustering of data with location and time extraction modules also functioning well, correctly extracting the vast majority of spatio-temporal i. Furthermore, user studies indicate the strengths of the approach for supporting more effective information extraction and assimilation than current state-of-the-art approaches. Finally, research in interaction paradigms for multimedia [19,20] suggests the effectiveness of these interaction metaphors.

In this environment of growing complexity and heterogeneity of web information, we believe that multiple-perspective, interactive approaches such as ours will become increasingly important for effective retrieval and assimilation of web-based information.

10. References

[1] Alexa, http://www.alexa.com.

[2] Ferragina, P. and Gulli, A., "A Personalized Search Engine Based on Web-Snippet Hierarchical Clustering," WWW, 2005.

[3] Google, http://www.google.com.

[4] Grokker, http://www.grokker.com.

[5] Hicks, M., "Search Startups Target Clustering," eWeeks.com, October 4, 2004.

[6] Hofmann, T., "Probabilistic Latent Semantic Indexing," ACM SIGIR, 1999.

[7] Hoon, D. M., Imoto, S., and Miyano, S., "The C Clustering Library," September 2004. http://biopython.org/docs/cluster/cluster.pdf.

[8] Hutchins, E. L., Hollan, J. D., Norman, D. A., "Direct Manipulation Interfaces," User Centered System Design. Lawrence Erlbaum Associates, 1986.

[9] Johnson, S., "Digging for Googleholes," Slate.com, July 16, 2003.

[10] Kan, M. Y., "Web Page Categorization without the Web Page," Proceedings, WWW, pp. 262-263, 2004.

[11] Kruse, P. M. et al, "Clever Search: A WordNet Based Wrapper for Internet Search Engines," Proceedings of 2nd GermaNet Workshop, 2005.

[12] Landauer, T. K., Foltz, P. W., and Laham, D., "An Introduction to Latent Semantic Analysis," Discourse Processes, Vol. 25, pp. 259-284, 1998.

[13] Notess, G., "Review of Northern Light," 2002, http://www.searchengineshowdown.com/features/nlight/review.html.

[14] Open Directory Project, http://www.dmoz.org.

[15] OpenMap, http://openmap.bbn.com.

[16] Porter Stemming Algorithm, http://www.tartarus.org/~martin/PorterStemmer.

[17] Rose, D. and Levinson, D., "Understanding User Goals in Web Search," WWW, 2004.

[18] Salton, G., and Buckley, C., "Term Weighting Approaches in Automatic Text Retrieval," Technical Report: TR87-881, 1987.

[19] Santini S., Gupta, A., and Jain, R., "Emergent Semantics through Interaction in Image Databases," IEEE Trans. on Knowledge and Data Engineering, Vol. 13, No. 3, 2001.

[20] Singh, R., Knickmeyer, R. L., and Gupta, P., "Designing Experiential Environments for Management of Personal Multimedia," ACM Multimedia, 2004.

[21] Strehl, A., "Relationship-Based Clustering and Cluster Ensembles for High-Dimensional Data Mining," PhD Dissertation, Dept. of Electrical and Computer Engineering, University of Texas at Austin, 2002.

[22] Tauscher, L. and Greenberg, S., "How People Revisit Web Pages: Empirical Findings and Implications for the Design of History Systems," International Journal of Human Computer Studies, Vol. 47, No. 1, 97-138, 1997.

[23] Toyama, K. et al, "Geographic Location Tags on Digital Images," ACM Multimedia, 2003.

[24] TREC 2004 QA Data, http://trec.nist.gov/data/qa/t2004_qadata.html.

[25] Vivisimo, http://www.vivisimo.com.

[26] Woodruff, A. et al, "Using Thumbnails to Search the Web," ACM SIGCHI, 2001.

[27] WordNet, http://wordnet.princeton.edu.

[28] World Gazetteer, http://www.world-gazetteer.com.