RC 10211 (A0106-001) June 6, 2001
Computer Science

IBM Research Report

MindMap: Utilizing Multiple Taxonomies and Visualization to Understand a Document Collection

W. Scott Spangler, Jeffrey T. Kreulen, Justin T. Lessler
IBM Research Division
Almaden Research Center
650 Harry Road
San Jose, CA 95120

Research Division
Almaden - Austin - Beijing - Delhi - Haifa - India - T. J. Watson - Tokyo - Zurich

MindMap: Utilizing Multiple Taxonomies and Visualization to Understand a Document Collection

Scott Spangler
IBM Almaden Research
spangles@us.ibm.com

Jeffrey T. Kreulen
IBM Almaden Research
kreulen@us.ibm.com

Justin Lessler
IBM Almaden Research
lessler@us.ibm.com

Abstract

We present a novel system and methodology for browsing and exploring topics and concepts within a document collection. The process begins with the generation of multiple taxonomies from the document collections, each having a unique theme. These taxonomies then become an integral tool in the exploration of the document collection.

It is assumed that the user of our system may have only a vague notion of exactly what they are attempting to understand, and would like to explore related topics and concepts rather than simply being given a set of documents. For this purpose, we have developed the MindMap interface to the document collection. Starting from an initial keyword query, the MindMap interface helps the user to explore the concept space by first presenting the user with related terms and high level topics in a radial graph. After refining the query by selecting any related terms, one of the related high level concepts can be selected for further investigation. The MindMap uses a novel binary tree interface to explore the composition of a concept based on the presence or absence of terms.

From the binary tree a concept can be further explored and visualized. Individual documents are presented as spatial coordinates where distance between points relates to document similarity. As the user browses this spatial representation, text is presented from the document that is most relevant to the user's initial query. Individual points can be selected to pull up the relevant paragraphs from the document with the keywords highlighted. Finally, selected documents are displayed and the user is allowed to further interact and investigate.

1. Introduction

The need for individuals and corporations to make informed decisions based on awareness in the face of the ever-increasing amount of information makes the need for tools and techniques to explore and understand this information paramount. Currently, the most popular methods are a combination of Boolean keyword searching and the use of a single taxonomy, which are best represented by the Internet search engines Yahoo! and Google. These techniques still fall short in the face of the many ambiguities and complex relationships that are contained in the documents.

To address these deficiencies, we present a novel system and methodology for browsing and exploring topics and concepts within a document collection. Our system leverages multiple taxonomies, related terms, visualization and user interaction to navigate and explore a document collection and the concepts that it contains. We call our system MindMap, because we have modeled our interface on techniques used for brainstorming.

The techniques developed have been found to be particularly useful when exploring a complex topic that is not yet fully understood by the user. The techniques bring to light related concepts and terms that help round out the understanding, while still allowing the user to get to specific documents and delve into the detail needed for in depth understanding.

In section 2, we describe techniques used in the generation of multiple thematic taxonomies for a document collection. In section 3, we describe the radial graph interface used to visualize and explore the multiple taxonomies. In section 4, we describe the binary tree interface that is used to partition a concept based on discriminatory terms. In section 5, we describe our technique for document and category visualization. Section 6 describes a user scenario to show how the system can be used to provide useful insights into corporate strategic relationships. In section 7 we discuss

the scalability of our approach. Finally, section 8 provides a summary and thoughts on future work.

2. Multiple Taxonomy Generation

Before a document collection can be explored using the MindMap interface, the collection must be categorized into multiple taxonomies. Each taxonomy is designed around a specific theme. The purpose of each taxonomy is to group related documents together in order to present a user with sets of documents that share common characteristics. Taxonomies generated using characteristics that bring to light interesting relationships are always more enlightening and tend to be domain and application specific. Some interesting example characteristics that we have come across are industry, geography, technology, document source, process stage, and document creation time, but there are many more. The need for multiple taxonomies arises because in many cases there exists no single taxonomy that captures all interesting relationships between documents and users will approach an investigation with different prior knowledge and goals.

For example, assume we have a set of documents containing a brief description of the Fortune 500 companies with one document per company. Each document might describe what the company produces, where the headquarters are located, what technologies the company has leadership in, and what business partnerships each company has. In other words, several taxonomies could be created over a single set of documents, each with a different theme. An example is shown in Figure 1.

Geography
North America
Europe
Africa
Asia
…

Technology
Chemical Engineering
Computer Networking
Alternative Fuels
Software
…

Industry
Energy
Transportation
Computers
Services
…

Figure 1: Example Thematic Taxonomies

Each of these taxonomies provides a unique way of defining what it means to be "similar". In the "geography" context to be similar means to be located in the same geographic region, while in the industry context it means to be competitors in the market. Each of these taxonomies is valuable in some information retrieval context.

Many approaches may be used to generate multiple taxonomies over a document collection. Text clustering over a feature space of term occurrence within documents is one way to generate a generic content-based taxonomy. After eliminating common stop words and (high- and low-frequency) non-content-bearing words, we represented the text data set as a vector space model. That is, we represented each document as a vector of certain weighted frequencies of the remaining words [11]. Using the **txn** weighting scheme [10]. This scheme emphasizes words with high frequency in a document, and normalizes each document vector to have unit Euclidean norm. We have found the k-means algorithm [3] to be an effective tool for generating a high level taxonomy over a collection of short documents. Different clustering approaches may be employed to generate different taxonomies (see [5], [9]).

Additional taxonomies may be generated by starting from a keyword description of each category in the taxonomy. These keywords are then included a priori as terms in the vector space dictionary for that taxonomy (thus dictionaries can and often do vary with each taxonomy). An initial classification of the documents is then generated by selecting for each document the category with which it shares the most keywords. Documents containing no keywords can be placed in a "Miscellaneous" category. After this initial classification by keywords is completed, nearest centroid methods may be employed to classify some or all of the examples in the Miscellaneous class.

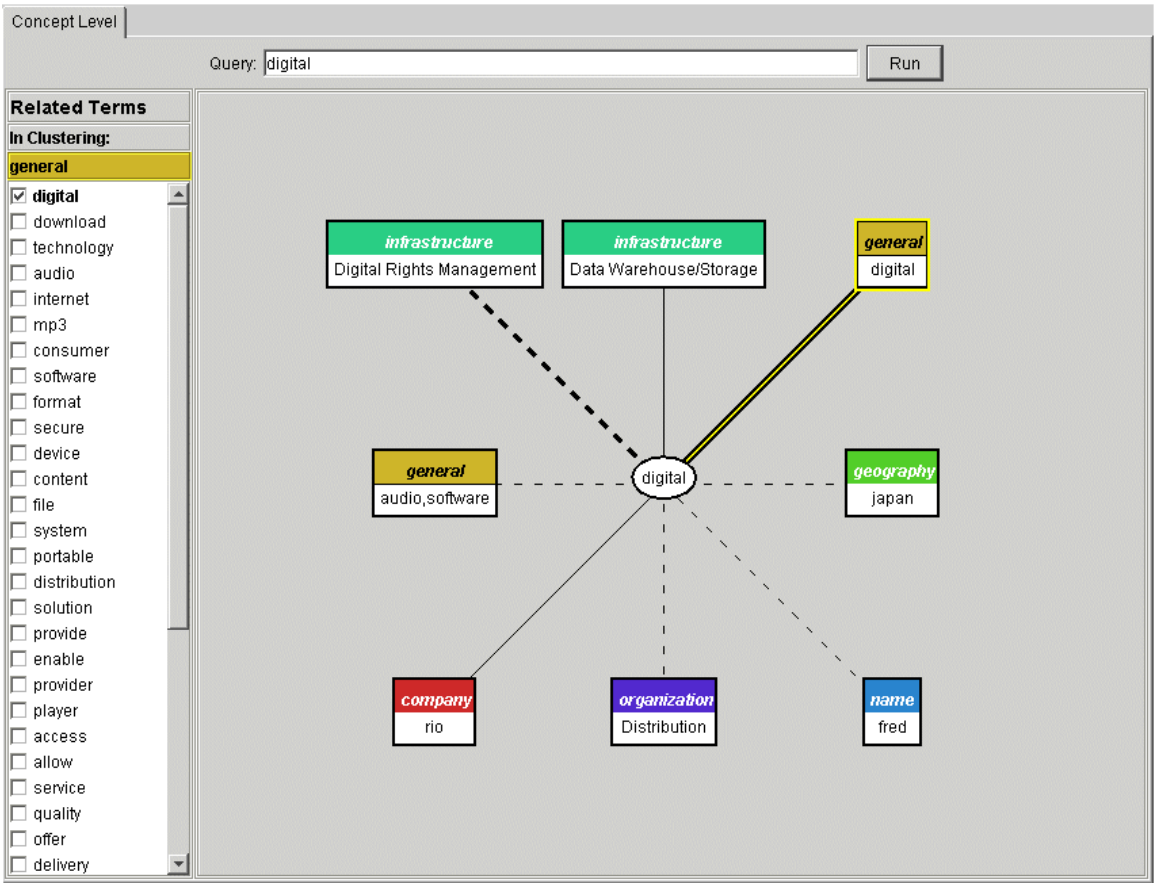

Figure 2: The Radial Graph

Ultimately, once all taxonomies have been generated, using whatever methods are appropriate for the domain, we save this information in a data structure. The data structure records the system or user given class name of each class, the membership of each example, and a centroid (mean) in the term occurrence vector space corresponding to each class. In addition the term occurrence matrix is saved, so that we have a record of what terms occur in what documents.

3. The Radial Graph

Now that several taxonomies have been created, the challenge becomes allowing users to quickly find the category or categories in the various taxonomies that are most relevant to the topic they wish to investigate. To do this we present the user with a radial graph representing eight classes, selected from among all of the taxonomies, that best match an entered query.

These classes are selected by first converting the query into the vector space model representation. The query is then compared with every class in each of the taxonomies. First those taxonomies whose dictionaries contain the greatest number of the keywords in the query are selected. From among these the eight classes whose centroids are closest to the query in the vector space model are displayed. The radial graph of these classes presented to the user has a node representing the query at its center, and the classes color-coded by taxonomy surrounding the query. The edges of the graph vary in thickness and stroke, indicating how closely associated they are to the query. The user can now select one of these classes to further explore in the Binary Tree view, described below, or further refine the query by selecting from a list of related terms presented on the left of the graph. [1].

The list of related terms is calculated by counting the co-occurrence of every word in the dictionary with the query string and using the Chi-squared test for independence of two discrete random variables to find the forty most related terms (those with the lowest probability) [7]. Since the dictionaries (vector space features) may vary between taxonomies, we allow the user to select the most relevant taxonomy and use the corresponding dictionary to be for the related term calculation. Selecting any checkbox immediately adds the selected term to the query string. This may then

cause the radial graph to change since the vector used to compare to the centroids has changed. Therefore the radial graph is updated both in terms of which nodes displayed and line thickness.

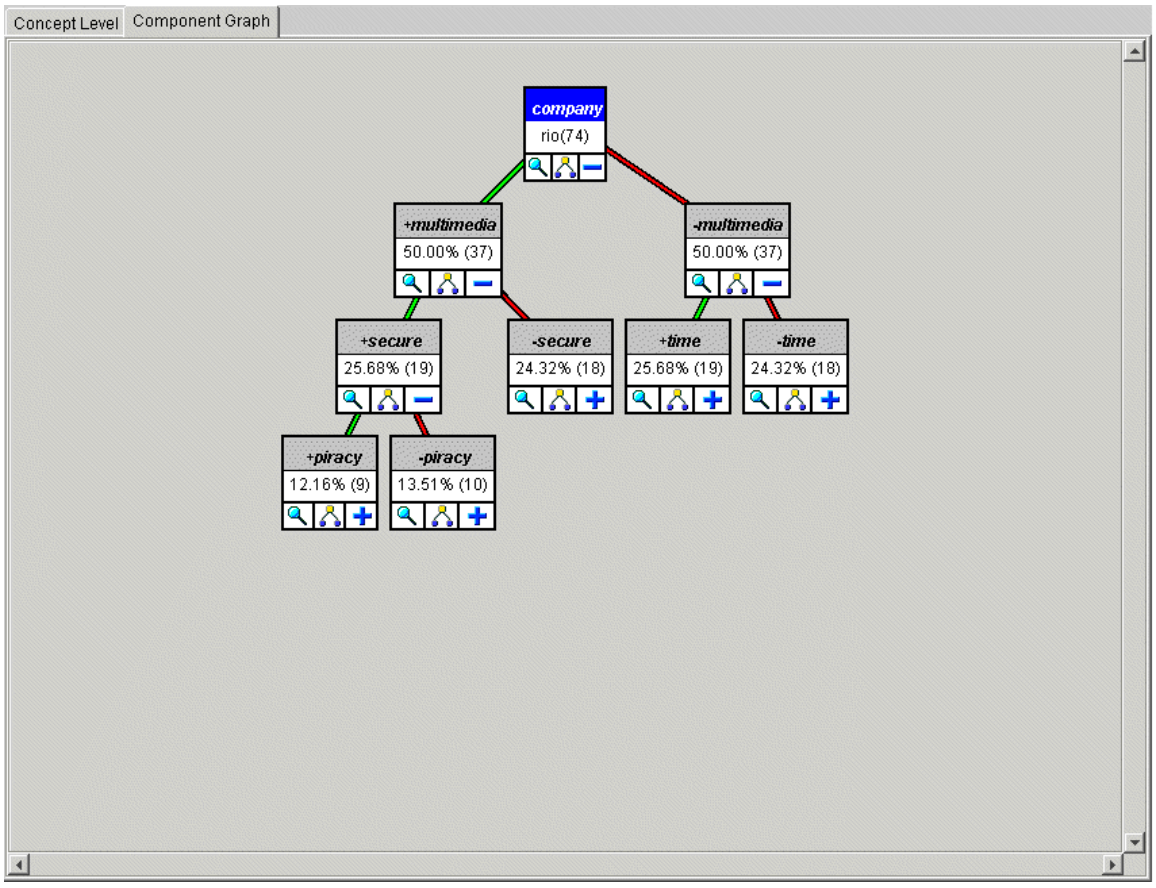

Figure 3: The Binary Tree

The radial graph provides a simple and intuitive way for users to find out which categories are relevant to the topic that they are exploring. Perhaps more importantly, it provides a way for users to get a sense of the relationships between taxonomies, sometimes finding surprising relationships that will aid them in gaining knowledge from the documents. The list of related terms allow users to refine their query in a sensible manner, and suggests themes to explore that may not have been otherwise known or considered. An example is shown in Figure 2.

4. The Binary Tree

Once the user selects a class from the radial graph a binary tree is presented that can be used to further refine the query. The root node of the tree represents the entire collection of documents matching the user's query in the selected class. Each branching of the tree divides the documents based on whether or not they contain some word (i.e. kd-tree). [4] The tree is initially expanded three levels, with each branching based on the word that most evenly divides the documents represented by the parent node. An example is shown in Figure 3.

Each node displays the number and percentage of documents represented by the node, and the word whose presence or absence characterizes it. A user can select to change the word to split on at any level in the tree, as well as contract previous expansions. When the user decides to split a node, a list of the five dictionary words that most evenly split the documents are presented, but a user can also select to choose from all of the words in the dictionary. If the user does not explicitly select a word to split on, previously unexpanded nodes are split on the word that most evenly divides the documents. When the user has found a subset of documents to examine more closely, the documents can be viewed in the category visualization screen described in Section 4.

The advantage of this approach to query refinement is that it allows the user to narrow the investigation to a manageable subset of documents without having to think of all the right words up front. The user can also gain insight into the structure of the class, quickly discovering important subdivisions.

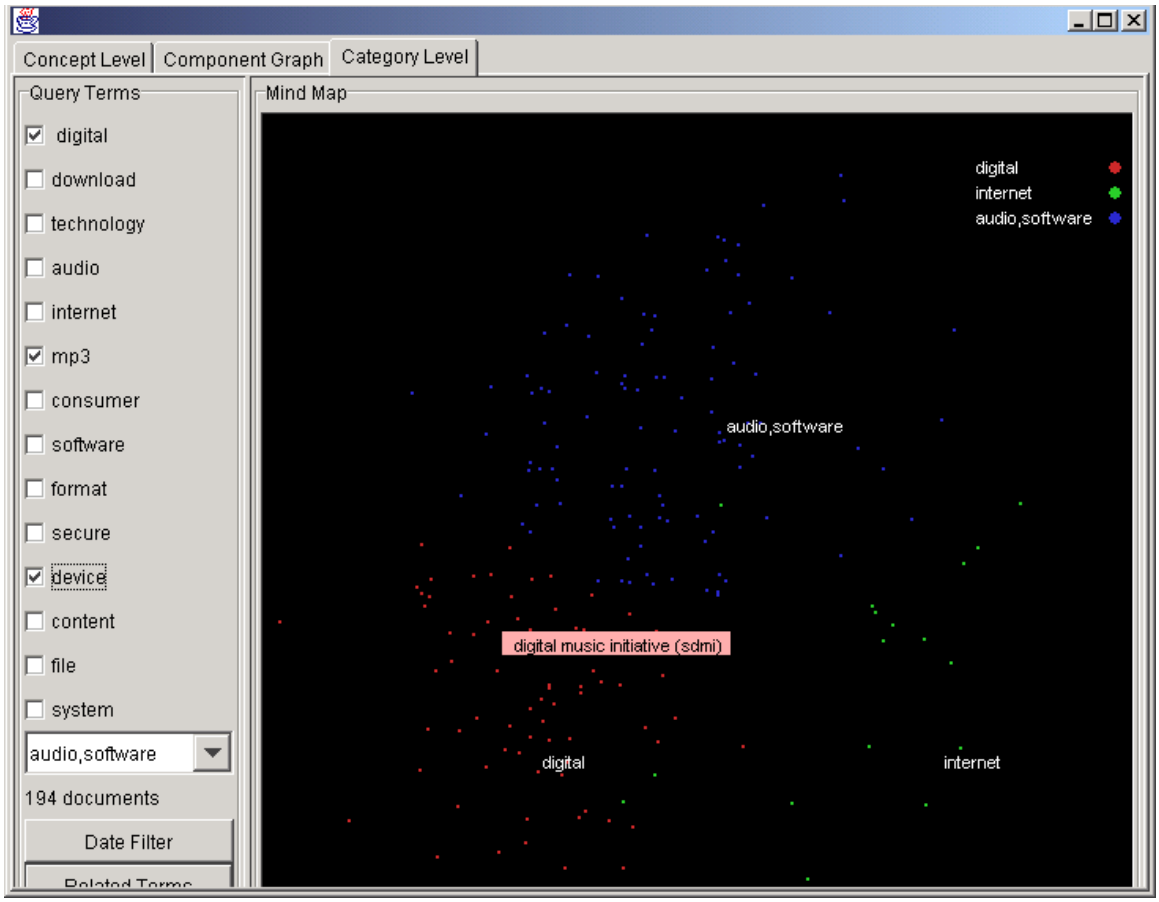

Figure 4: Document Visualization

5. Category Visualization

After the user selects a node from the MindMap binary tree, the next step is to present to the user those examples that match the query. One simple way to do this would be to merely list the examples in order of increasing distance from the centroid of the class. The disadvantage of this approach is that it essentially reduces our understanding of each example to a single number. More detailed statistics about each document are available through the word occurrence matrix calculated earlier. We can use this information to represent each document as a point in a high-dimensional geometric space, the position being related to the words the document contains. [8]

The high dimensionality (one dimension for each term) of this space makes it difficult to visualize in two dimensions. To get around this problem we find an "interesting" plane in the high dimensional space and project onto this plane by finding the intersection between the plane and a normal line drawn to each point. It turns out that the most interesting plane from the perspective of distinguishing categories of examples is a plane drawn through the centroid of three categories. [2] It is important to establish a coordinate system via *points of interest* on the display, in order to make the visualization meaningful. [6] We do this by labeling the centroid of each cluster at the position of its projected spatial coordinate.

In MindMap, we display the category chosen by the user along with two other categories that are nearest neighbors to the chosen category determined using the cosine distance between centroids. The centroids of these three categories define the plane of the visualization, and all points in the three categories that match the query are displayed in the visualization. An example of such a plot is shown in Figure 4.

The points are colored based on class membership. Each point may be investigated further by placing the mouse over it to see a short document excerpt or clicking on the point to view the document text.

The advantage of this visualization is that documents that are near to each other in space should also share many terms. This allows the user to quickly locate documents related to a document of interest. In addition, by showing classes related to the class the user has selected, we may find documents just outside the

users area of interest that are related, and perhaps relevant to, the users query. One weakness of the current visualization approach is that if focuses on only one taxonomy at a time. A possible improvement would be to show similar displays of some related taxonomies in additional panels.

Tests of the MindMap prototype with typical users of revealed the necessity to simplify the visualization plots as much as possible. Hence we only display three classes at a time. Further simplifications may be necessary depending on the sophistication of the user community (e.g. displaying only two classes at a time and using the origin as the third centroid, or even hiding all points not in the selected user class).

6. User Scenario

The following pictures represent a typical usage of MindMap on a data set of music articles downloaded from the web. We assume our user is a music company executive who is interested in determining what other music companies are strategically aligned with Napster. The investigation begins by entering the keyword "Napster" as an initial query. MindMap displays the diagram shown in Figure 5.

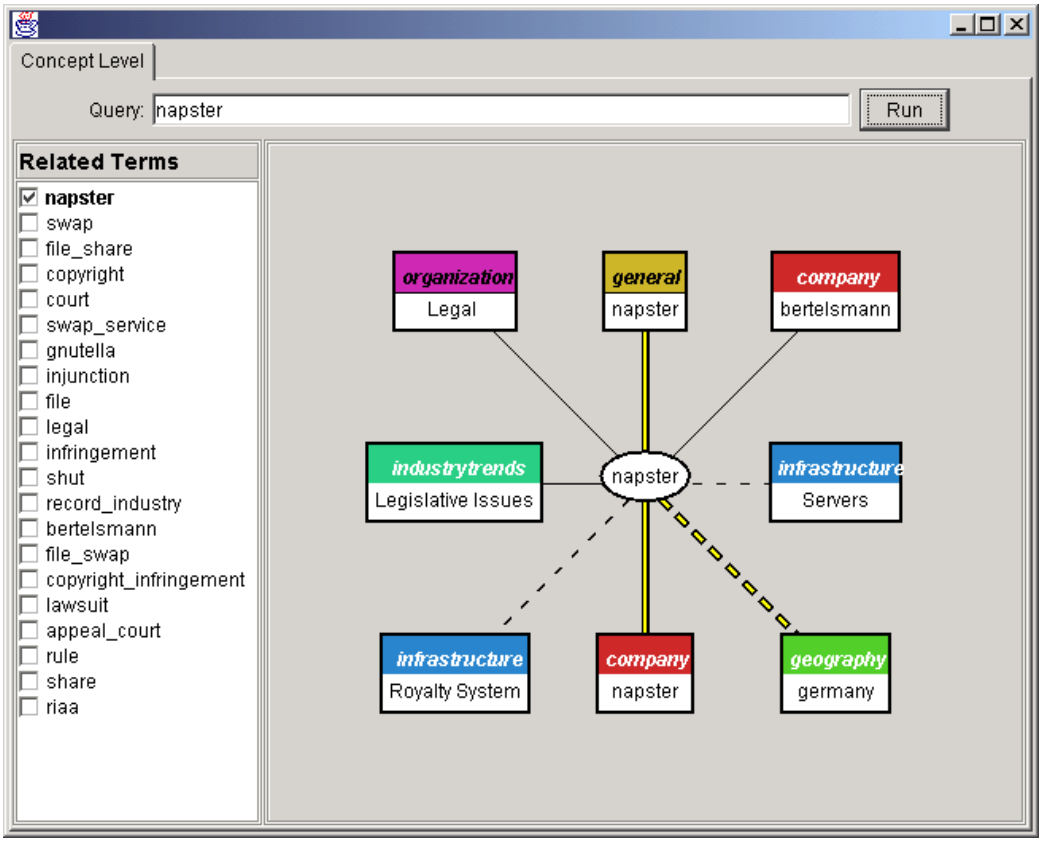

Figure 5: Radial Graph

Notice that in addition to the company "napster" the company "bertelsmann" comes up as a related concept. The related words that are presented help the user to refine the query to a specific area of interest. Notice that in addition to single words, commonly co-occurring phrases are also displayed in the related terms list. Next, our hypothetical user selects the "bertlesmann" class because this is a company the user knows has a strategic relationship with Napster. Keep in mind that the selected class represents those documents which are focussed on or talk about the Bertelsmann company, not just those that may contain the work "Bertelsmann". Selecting this node brings up a further breakdown of the bertelsmann class by system selected dictionary terms. The result is shown in Figure 6.

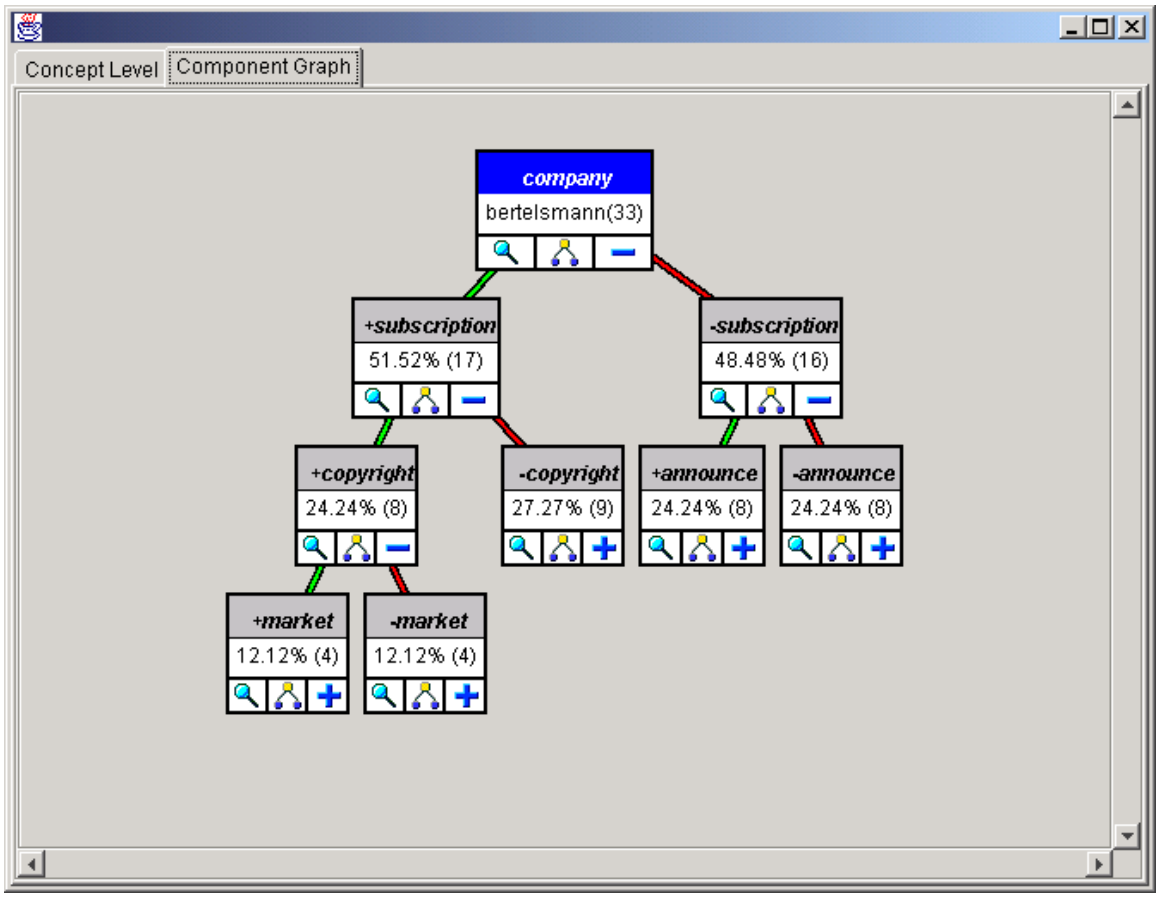

Figure 6: Napster Binary Tree

Let us say that the user is interested in the subscription issue, but not in copyrights, so that therefore the user selects the "-copyright" node underneath the "+subscription" node of the tree (the node containing 9 examples). This brings up the class visualization screen shown in Figure 7. Note that more than 9 document icons are displayed because matching documents in two neighboring classes are displayed as well.

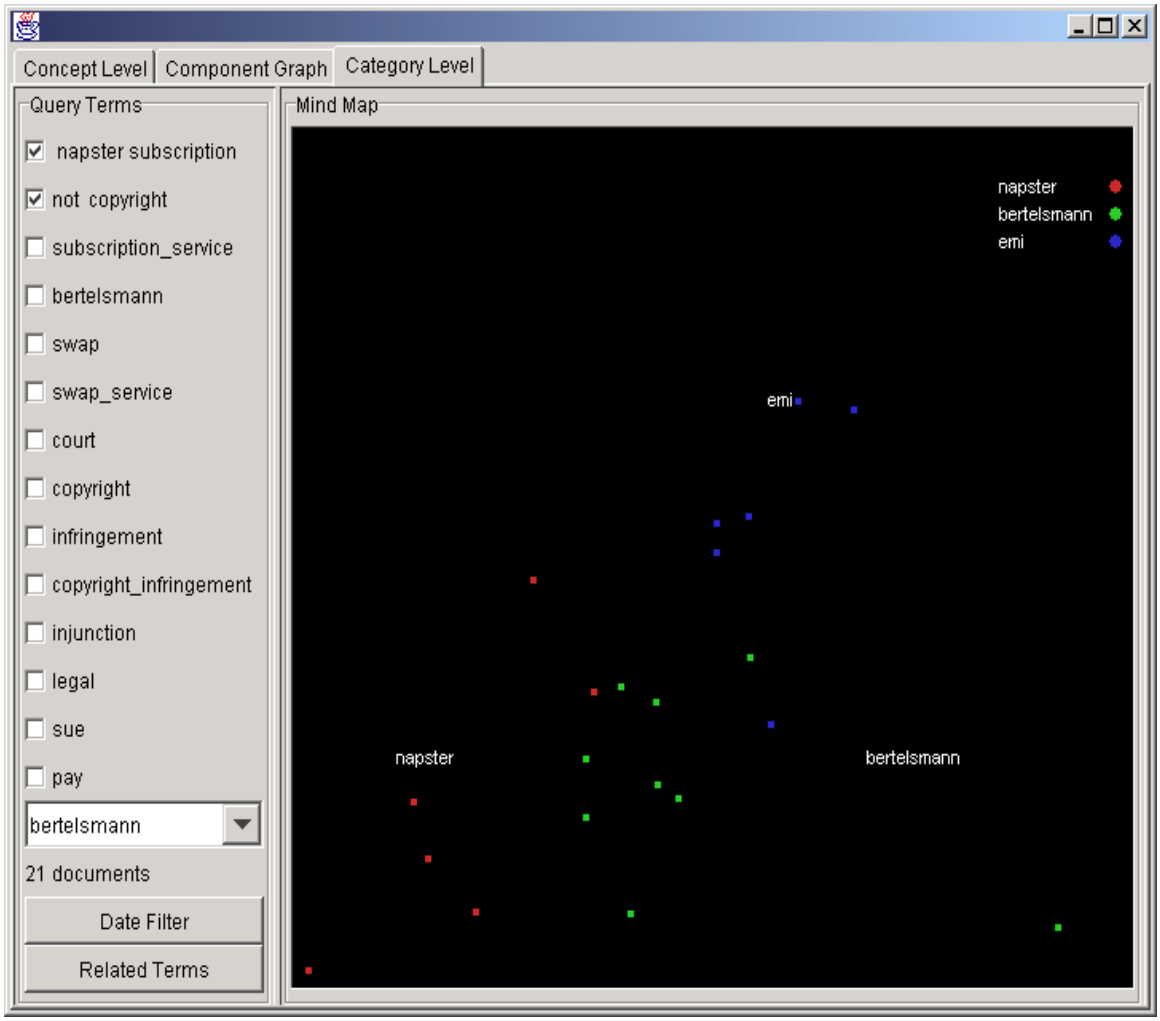

Figure 7: Napster Document Visualization

The two classes that are nearest to Bertelsmann are displayed along with it and only those documents containing the text "napster" and "subscription" and not containing "copyright" are displayed. In Figure 7 we see that Napster and EMI were selected by the system because of their dictionary vector space proximity. Though the Bertelsmann relationship with Napster is well known to our user, the relationship between Napster, Bertelsmann, and EMI is less so. Thus MindMap has revealed a high-level relationship at the macro level that can be communicated to the user independent of any specific document. To find out more detailed information, the user selects an EMI document near the 'Bertelsmann' concept, resulting in the display shown in Figure 8.

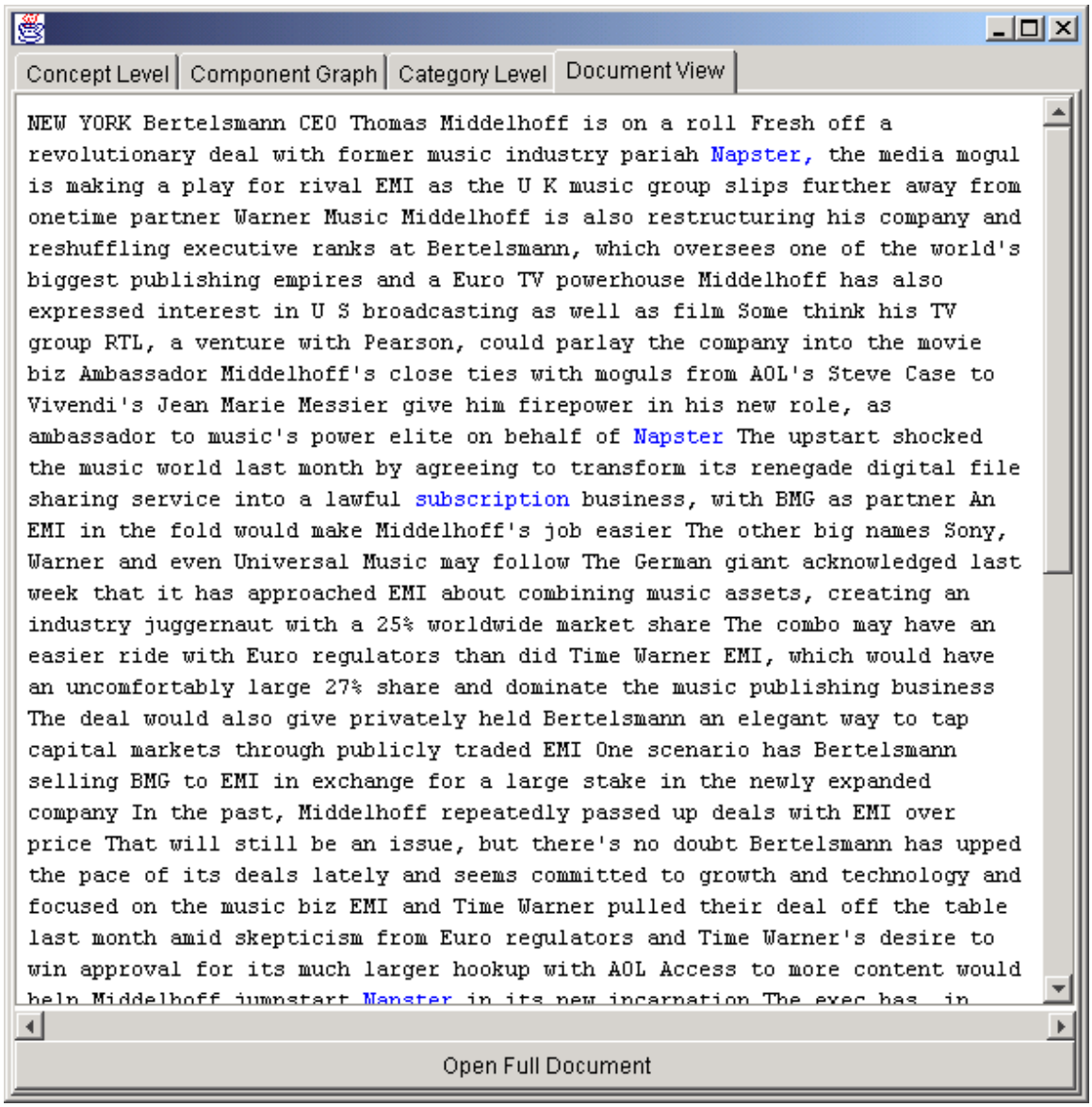

Figure 8: Napster Related Document

Thus the user has rapidly found a document discussing strategic relationships in the music industry related to Napster and subscription services. Formulating a query that would discover this same document without pulling down a thousand other uninteresting documents would not be nearly as easy for most users. Clearly in some instances the MindMap approach allows the user to find a specific type of document much more readily than standard keyword queries alone can.

7. Scalability

The example document sets that we have tested using this approach have ranged in size from 5000 to 30,000 documents. The total size of the largest text corpus processed is 100Mb. The text corpus is represented in memory as a sparse term occurrence matrix. This matrix is stored on disk for each classification until needed, and then loaded into memory on demand. This allows us to work with a very large number of different classifications, in a relatively small memory footprint. Also in memory we keep a representation of each classification containing that classifications dictionary (usually around 2000 terms) along with the centroid vector for each cluster in each classification. The centroid is a dense vector containing a floating point number for each dictionary term. Most of our examples use less than 10 classifications, each containing between 10 and 20 clusters.

If the dictionary size and number of classifications remain constant, tests show this approach should scale

up to about 1-2 GB of text information on current PC hardware (256Mb of RAM and 500Mhz clock speed). The need for RAM increases linearly with an increase in dictionary size or with an increase in the total number of clusters. The scalability issues would probably preclude out MindMap implementation from being used in conjunction with a general search engine generating queries over the entire World Wide Web. We feel this approach is more suitable for much small to medium size document collections, such as abstracts for corporate strategic document repositories.

8. Future Directions

In summary, we have described a system and methodology for the exploration of topics and concepts contained in a document collection. We have leveraged multiple taxonomies, related terms, visualization and user interaction to allow for a comprehensive and flexible investigation of the content.

We believe there is much more that can be done to enable more understanding and exploration within document collections. User studies need to take place to determine the efficacy of each of the methods described here. There is a potential for many types of analytics that provide the user with deeper insights into concepts and relationships. We believe that techniques for automatically finding relationships across multiple taxonomies will be very promising. Additionally, we believe that techniques borrowed from the information extraction research community will provide additional insights and enable deeper analytics.

9. References

[1] Cooper, J. and Byrd, R. Lexical navigation: visually prompted query expansion and refinement. *Proceedings of the 2nd ACM International Conference on Digital Libraries,* July 1997.

[2] Dhillon, I., Modha, D., and Spangler, S. (1998). Visualizing Class Structures of Multi-Dimensional Data. *Proceedings of 30th Conference on Interface, Computer Science and Statistics.* May 1998.

[3] Duda, R. O. and Hart, P. E. (1973). *Pattern Classification and Scene Analysis.* Wiley.

[4] Volker Gaede and Oliver Günther, Multidimensional Access Methods, ACM Computing Surveys, 30(2):170-231, June 1998.

[5] Hartigan, J. A. (1975) *Clustering Algorithms.* Wiley.

[6] Olsen, K.A., Korfhage, R.R., et al. Visualization of a Document Collection: The VIBE System. Information Processing &Management 29(1) (1993), 69-81.

[7] Press, W. et. al. *Numerical Recipes in C. 2nd Edition*. Cambridge University Press. (1992), 620-623.

[8] Raghaven, V., and Wong, S. A Critical Analysis of Vector Space Model for Information Retrieval, *Journal of the American Society for Information Science*, 37(5), 279-287.

[9] Rasmussen, E. (1992). Clustering algorithms. In Frakes, W. B. and Baeza-Yates, R., editors, *Information Retrieval: Data Structures and Algorithms,* pages 419-442. Prentice Hall, Englewood Cliffs, New Jersey.

[10] Salton, G. And Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing & Management,* 4(5):512:523.

[11] Salton, G. and McGill, M. J. (1983). *Introduction to Modern Retrieval.* McGraw-Hill Book Company.