RJ 10300 (A0308-014) August 20, 2003
Computer Science

IBM Research Report

Interactive Methods for Taxonomy Editing and Validation

Scott Spangler, Jeffrey Kreulen
IBM Research Division
Almaden Research Center
650 Harry Road
San Jose, CA 95120-6099

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

Interactive Methods for Taxonomy Editing and Validation

Scott Spangler
IBM Almaden Research Center
650 Harry Road, San Jose, CA 95120
408-927-2887
email: spangles@us.ibm.com

Jeffrey Kreulen
IBM Almaden Research Center
650 Harry Road, San Jose, CA 95120
408-927-2431
email: kreulen@us.ibm.com

ABSTRACT

Today's enterprise understands that improved utilization of its collective knowledge assets leads to improved business performance. The reality of the proliferation of electronic information and the pressure to produce more with fewer resources while performing increasingly complex tasks makes this a continuous challenge. To address this challenge enterprises are building knowledge repositories and structuring them in ways that are meaningful to their organization, business and processes. This structuring typically manifests itself in the form of one or more taxonomies. The taxonomies are meaningful hierarchical categorizations of documents into topics reflecting the natural relationships between the documents and their business objectives. Improving the quality of these taxonomies and reducing the overall cost required to create them is therefore an important area of research. Supervised and unsupervised text clustering are automated approaches to creating and maintaining document taxonomies. However, human expertise also has an indispensable role to play in guiding the taxonomy generation process and validating the results. Towards this end we have developed an interactive approach to taxonomy creation and validation. This approach involves helping the taxonomy editor understand and evaluate each category of a taxonomy and visualize the relationships between the categories. Multiple techniques allow the user to make changes at both the category and document level. Metrics then establish how well the resultant taxonomy can be modeled for future document classification. Our approach enables the development of multiple taxonomies so that multiple

relationships in the documents can be modeled. In this paper, we present our approach to document taxonomy creation and modification and then demonstrate the effectiveness of this approach in real time analysis and reporting of discussion forum topics during IBM's corporate wide "ValuesJam" event.

1 INTRODUCTION

Businesses have been able to systematically increase the leverage gained from enterprise data through technologies such as relational database management systems and techniques such as data warehousing. Additionally, it is conjectured that the amount of knowledge encoded in electronic text far surpasses that available in data alone. However, the ability to take advantage of this wealth of knowledge is just beginning to meet the challenge. Businesses that can take advantage of this potential will surely be at an advantage through increased efficiencies. One important step in achieving this potential has been to structure the inherently unstructured information in meaningful ways. A well-established first step in gaining understanding is to segment examples into meaningful categories [2]. This leads to the idea of taxonomies--natural hierarchical organizations of the information in alignment with the business goals, organization and processes. While there will be some commonality in some industries, these natural organizations will have significant diversity across domains and organizations.

Research to address this need for taxonomy development has concentrated largely around automated grouping techniques such as text clustering. While we believe that text clustering is an invaluable tool, indeed it is part of our solution, we assert that it is insufficient to meet the full challenge of taxonomy generation by itself. Our experience using variations of K-Means [9][20] and Expectation Maximization (EM) clustering algorithms [25] [26] have shown that they generate useful seed taxonomies, but rarely generate a satisfactory final taxonomy for a given business problem. For example, if you were to cluster a set of patents with the intent to create a technology based taxonomy you would typically find some of

the clusters to be technologies and some to be based on some other aspect or relationship found in the text such as processes. Careful feature selection is one approach to address this problem by leveraging controlled vocabularies [14]. However, we find this approach to be very labor intensive and would still yield results that would need further refinement.

Our approach to solve this problem focuses on the visualization, editing and validation of clustering results [23]. We will go into details of our approach below but further clarification on the problem and its relationship to cluster validation is warranted. The problem we are attempting to solve has been referred to in the literature [5] [8] as clustering validation. Validation methods have typically been based on one of three types of criteria: external, internal and relative. External criteria typically use a pre-specified 'ground truth' by which we can directly measure the quality of our clusters. Internal criteria are based on statistics or measures computed from a given taxonomy. Relative criteria are based on comparison with alternative taxonomies. Our approach integrates internal and external criteria, with the external criteria (a human expert) being the final determinant. Clearly it is not practical to read each document and categorize it, however, expert inspection guided by appropriate feedback is a powerful combination. We wish to stress that our innovation is not a particular clustering or visualization technique, but is rather a general strategy for applying clustering and visualization techniques interactively with human expertise to create the best possible taxonomy for a business application.

In this paper we will outline a system and methodology that leverages clustering or keyword queries as a seed taxonomy and provides the appropriate feedback to a human analyst to efficiently guide the user to refine the taxonomy toward a desired, if not previously known, quality and model-able taxonomy. In section 2 we describe how we generate an initial taxonomy. In section 3, we describe the important

capabilities for viewing and understanding a taxonomy. This gives the taxonomy analyst the necessary feedback to modify a taxonomy, which we describe in section 4. In section 5, we describe our approach to validating and ensuring that a taxonomy can be modeled for the purpose of classifying future documents. In section 6, we give a detailed illustration of how our approach was used successfully to analyze and report on IBM's corporate wide "ValuesJam" discussion forum event. Finally, in section 7 we summarize and outline areas for future research.

2 GENERATING A TAXONOMY

Because there is no one "right" taxonomy to cover all possible uses of document collection, it is important to provide multiple methods for generating the initial seed taxonomy from which the user begins to create the final document classes. Our methodology provides two main alternatives for taxonomy generation, via clustering and via keywords queries.

2.1 Taxonomy Generation via Clustering

In the cases where the user has no preconceived idea about what categories the document collection should contain, text clustering may be used to create an initial breakdown of the documents into clusters, grouping together documents having similar word content. To facilitate this process we represent the documents in a vector space model. We represent each document as a vector of weighted frequencies of the document features (words and phrases) [22]. We use the txn weighting scheme [21]. This scheme emphasizes words with high frequency in a document, and normalizes each document vector to have unit Euclidean norm. For example, if a document were the sentence, "We have no bananas, we have no bananas today," and the dictionary consisted of only two terms, "bananas" and "today", then the unnormalized document vector would be {2 1} (to indicate two bananas and one today), and the normalized version would be: $\left[2/\sqrt{5}, 1/\sqrt{5} \right]$.

The words and phrases that make up the document feature space are determined by first counting which words occur most frequently (in the most documents) in the text. A standard "stop word" list is used to eliminate words such as "and", "but", and "the" [7]. The top N words are retained in the first pass, where the value of N may vary depending on the length of the documents, the number of documents and the number of categories to be created. Typically N=2000 is sufficient for 10000 short documents of around 200 words to be divided into 30 categories. After selecting the words in the first pass, we make a second pass to count the frequency of the phrases that occur using these words. A phrase is considered to be a sequence of two words occurring in order with out intervening non-stop words. We again prune to keep only the N most frequent words and phrases. This becomes the feature space. A third pass through the data indexes the documents by their feature occurrences. The user may edit this feature space as desired to improve clustering performance. This includes adding in particular words and phrases the user deems to be important, such as named entities like "International Business Machines". Stemming is usually also incorporated to create a default synonym table that the user may also edit [10].

As our primary tool for automated classification, we used the k-means algorithm [6], [9] using a cosine similarity metric [20] to automatically partition the documents into k disjoint clusters. The algorithm is very fast and easy-to-implement. See [20] for a detailed discussion of various other text clustering algorithms. The k-means algorithm produces a set of disjoint clusters and a centroid for each cluster that represents the cluster mean. Typically k is initially set to 30, for the highest level of the taxonomy, though the user may adjust this if desired. The initial taxonomy assigns each document to only one category (cluster). Any of the initial high level categories may be subcategorized if desired, either automatically and recursively, or manually one at a time. After clustering is complete, a final "merging" step takes place. In this step, two or more clusters which are each dominated by the same keyword

(dominated means that 90% of the examples contain this keyword) are merged into a single cluster and a new centroid is calculated based on the combined example set. The reason we do this is to avoid arbitrarily separating similar examples into separate subsets before the expert user has had a chance to evaluate the class as a whole.

2.2 Cluster Naming

In order to help the user understand the meaning of each cluster, the system gives each document category a name that describes it. Cluster naming is not an exact science, but our method attempts to describe the cluster as succinctly as possible, without missing any important constituent components. The first rule of naming is that if a single term dominates a cluster, then this term is given as the cluster name. If no term dominates then the most frequent term in the cluster becomes the first word in the name and the remaining set of examples (those not containing the most frequent term) are analyzed to find the dominant term. If a dominant term for the remaining examples is found then this term is added to the name (separated by a comma) and the name is complete, otherwise the process continues for up to four word length names. Beyond four words we simply call the class "Miscellaneous".

2.3 Taxonomy generation via keyword queries

An alternative to k-means clustering is to create an initial categorization via Boolean keyword queries. This approach is most useful when the domain expert already has a strong idea of how the taxonomy should be structured. Each category is described via a set of keywords connected by "and", "or", or "not". The resulting query defines those document examples that belong to the category. The queries are then ordered and the document initially falls into the category of the query that matches its content. Any document that does not match any query is placed in a special "Miscellaneous" category. The user may reorder the queries based on the results to insure that every category starts with a significant number of matching documents. Once the initial taxonomy is created, further refinement can take place

via k-means clustering (using centroids of the existing categories as seeds) or by manual editing of the categories (see next section). The user may create different sets of queries to generate multiple taxonomies if desired [24].

3 VIEWING THE TAXONOMY

Before a user can begin editing a taxonomy, they must first understand the existing categories and their relationships. In this section, we describe our strategy to communicate the salient characteristics of a document taxonomy to the user.

Our primary representation of each category is the centroid [6]. The distance metric employed to compare documents to each other and to the category centroids is the cosine similarity metric [20]. This metric is used at the user's request to automatically partition any subset of the documents into k disjoint clusters. As will be seen, during the category editing process, we are not rigid in requiring each document to belong to the category of its nearest centroid, nor do we strictly require every document to belong to only one category.

To provide a good understanding of a given taxonomy, which is generally hierarchical in structure, we provide a series of views that cover different levels of detail. The views include a global, single level within a tree (all children of a single parent including the root), and a detailed view of each individual category. In the global "Categorization Tree" categories with subcategories are displayed as "folders" that can be expanded.

[Insert Figure 1 Here]

Leaf categories are displayed as nodes. Selecting a folder can take the user down a level to the Category Table view, which shows statistics about just the categories that are immediate subclasses of the selected

category (see Figures 1). Selecting a leaf category or selecting a row in the Category Table displays the Category (Class) View (see Figure 9). This view provides several different windows on a single category that help to explain and summarize the content of the selected category's documents. The remainder of this section describes in detail what information is communicated in the Category View.

3.1 Summaries

Since we cannot expect the user to spend the time to read through all of the individual documents in a category, summarization is an important tool to help the user understand what a category contains. Summarization techniques based on extracting text from the individual documents [11] were found to be insufficient in practice for the purpose of summarizing an entire document category, especially when the theme of that category is somewhat diverse. Instead, we employ two different techniques to summarize a category. The first is a feature bar chart. This chart has an entry for every dictionary term (feature) that occurs in any document of the category. Each entry consists of two bars, a red bar to indicate what percentage of the documents in the category contain the feature, and a blue bar that indicates how frequently the feature occurs in the background population of documents from which the category was drawn. The bars of the chart are sorted in decreasing order of the difference between blue and red. Thus the most important features of the category in question are shown at the beginning of the chart. This chart quickly summarizes for the user all the important features of a category, with their relative importance indicated by the size of the bars (see Figure 2).

[Insert Figure 2 Here]

The second technique is a dynamic decision tree representation that describes what feature combinations define the category. This tree is generated in the same manner as a binary ID3 [15], selecting at each decision point the attribute that is most helpful in splitting the whole population of documents so that the two new classes of documents created are most nearly pure category and pure non-category. Each

feature choice is made on the fly as the user expands each node, until a state or purity is reached or when no additional features will improve the purity with respect to the category. The result is essentially a set of classification rules that define the category to the desired level of detail. At any point the user may select a node of the decision tree to see all the documents at the node, all the in-category documents at the node, or all the non-category documents at the node. The nodes are also color coded: red is a node whose membership is more than (or equal to) 50% in category, blue is a node whose membership is less than 50% in category. This display (see figure 3) gives users an in depth definition of the class in terms of salient features and lets them readily select various category components for more in depth study. In figure 3 the highlighted node selects the rule: "+respect, -individual" which represents those examples that contain the term "respect" but not the term "individual". Documents that follow this rule make up 10% of the "respect for individual" category, but out of all documents that match the rule, 82.11% do not belong to the "respect for individual" category.

[Insert Figure 3 Here]

3.2 Visualization

In order to understand specifically how two or more categories at the same level of the taxonomy relate to each other, a visualization strategy is employed. The idea is to visually display the vector space of a bag-of-words document model [21] [22] so that the documents will appear as points in space. The result is that documents containing similar words occur near to each other in the visual display. If the vector space model were two dimensional, this would be straightforward—we could simply draw the documents as point on an X,Y scatter plot. The difficulty is that the document vector space will be of much higher dimension. In fact the dimensionality will be the size of the feature space (dictionary), which is typically thousands of terms. Therefore we need a way to reduce the dimensionality from thousands to two in such a way as to retain most of the relevant information. Our approach uses the

CViz method [4], which relies on three category centroids to define the plane of most interest and to project the documents as points on this plane (by finding the intersection with a normal line drawn from point to plane). The selection of which categories to display in addition to the selected category is based on finding the categories with the nearest centroid distance to the selected category. The documents displayed in such a plot are colored according to category membership. The centroid of the category is also displayed. The resultant plot is a valuable way to discover relationships between neighboring concepts in a taxonomy (see figure 4).

[Insert Figure 4 Here]

3.3 Sorting of Examples

When studying the examples in a category to understand the category's essence, it is important that the user not have to select the examples at random. To do so can sometimes lead to a skewed understanding of the content of a category, especially if the sample is small compared to the size of the category (which is often the case in practice). To help alleviate this problem, our software allows sorting of examples based on the "Most Typical" first or "Least Typical" first criteria. This translates in vector space terms to sorting in order of distance from category centroid (i.e. most typical is closest to centroid, least typical is furthest from centroid). The advantage of sorting in this way is two fold: reading documents in most typical order can help the user to quickly understand what the category is generally about, without having to read a large sample of the documents in the category, while reading the least typical documents can help the user to understand the total scope of the category and if there is conceptual purity.

4 EDITING THE TAXONOMY

Once the content manager understands the meaning of the classes in the taxonomy and their relationship to each other, the next step is to provide tools for rapidly changing the taxonomy to reflect the needs of the application. Keep in mind that our goal here is not to produce a "perfect" taxonomy for every possible purpose. Such a taxonomy may not even exist, or at least may require too much effort to obtain. Instead we want to focus the users efforts on creating a "natural" taxonomy that is practical for a given application. For such applications, there is no right or wrong change to make. It is important only that the change accurately reflect the expert user's point of view about the desired structure. In this situation, the user is always right. The tool's job is to allow the user to make whatever changes may be deemed desirable. In some cases such changes can be made at the category level, in other cases a more detailed modification of category membership may be required. Our tool provides capabilities at every level of a taxonomy to allow the user to make the desired modifications with a simple point and click.

4.1 Category Level

Category level changes involve modifying the taxonomy at a macro-level, without direct reference to individual documents within each category.

4.1.1 Merging

Merging two classes means creating a new category that is the union of two or more previously existing category memberships. A new centroid is created that is the average of the combined examples. The user supplies the new category with an appropriate name.

4.1.2 Deleting

Deleting a category (or categories) means removing the category and its children from the taxonomy. The user needs to recognize this may have unintended consequences, since all the examples that

formerly belonged to the deleted category must now be placed in a different category at the current level of the taxonomy. To make this decision more explicit, we introduce the graphic called "View Secondary classes" chart (see figure 10).

This chart displays what percentage of a category's documents would go to which other categories if the selected category were to be deleted. Each slice of the displayed pie chart can be selected to view the individual documents represented by the slice. Making such information explicit allows the user to make an informed decision when deleting a category, avoiding unintended consequences.

4.1.3 Clustering

At any node of the Categorization Tree the user may request subclassing via text clustering. This will apply a standard clustering algorithm, such as k-means [9] [20] to the set of documents represented by the selected category. The user will be asked to provide a value for the number of classes to create. Subclasses are derived by applying the clustering algorithm to the vector space model. Each newly derived subcategory will be given a name based on the features that have an in-class frequency most different from that of the background frequency. If desired, this same clustering approach can be applied repeatedly in a recursive fashion on each derived category until a stopping criteria is reached (either some user supplied minimum category size or sufficiently high value category cohesion). The resulting sub-categorization can then be edited if desired to reflect a more natural partitioning.

4.1.4 Dragging and Dropping

From the Categorization Tree view the user can select any category and drag and drop the category into any existing folder (a category with children). An example of when such an operation might be performed is when a very specific category is created at the root node of the tree, which would more naturally belong within an already existing, more general, category. The operation of dragging and

dropping a category to a folder has consequences to all other folders in a direct line from the root of the tree to the destination node (which gain the contents of the source node) and to all other folders in a direct line form the root to the source node (which lose the contents of the source node).

4.2 Document Level

While some changes to a taxonomy may be made at the class level, others require a finer degree of control. These are called document level changes, and consist of moving or copying selected documents from a source category to a destination category. The most difficult part of this operation from the users point of view is selecting exactly the right set of documents to move so that the source and destination categories are changed in the manner desired. To address this problem a number of document selection mechanisms are provided.

4.2.1 Selection by Keywords

One of the most natural and common ways to select a set of documents is with a keyword query. The user may enter a query for the whole document collection or for just a specific category. The query can contain keywords separated by "and" and/or "or" and also negated words. Words that co-occur with the query string are displayed for the user to help refine the query. Documents that are found using the keyword query tool can be immediately viewed and selected one at a time or as a group to move or establish a new category.

4.2.2 Selection by Sorting

Another way to select documents to move or copy is via the "Most/Least Typical" sorting technique described in section 2. For example, the documents that are least typical of a given category can be located, selected, and moved out of the category they are in. They may then be placed elsewhere or in a new category.

4.2.3 Selection by Visualization

The scatter plot visualization display described in section 2 can also be a powerful tool for selecting individual or groups of documents. Using a "floating box", groups of contiguous points (documents) can be selected and moved to the new desired class (see figure 5).

[Insert Figure 5 Here]

4.2.4 To Move, Copy, or Delete

Independent of the document selection method, the user is allowed to choose between moving, copying, or deleting the selected documents. Moving is generally preferable because single class membership generally leads to more distinct categories which are better for the classification of future documents. Still, in cases where a more ambiguous category membership better reflects the user's natural understanding of the taxonomy, the user may create a copy of the documents to be moved and to place this copy in the destination category. In such cases the individual document will actually exist in two (or more) categories at once, until the user deletes the example. Deletion is the third option. It allows the document to be removed from the taxonomy, if it is judged to be not applicable.

5 VALIDATION

Whenever a change is made to the taxonomy, it is very important for the user to validate that the change has had the desired effect on the taxonomy as a whole, and that no undesired consequences have resulted from unintentional side effects. Our software contains a number of capabilities that allow the user to inspect the results of modifications. The goal is to insure both that the categories are all meaningful, complete, and differentiable, and that the concepts represented by the document partitioning can be carried forward automatically in the future as new documents arrive.

5.1 Direct Inspection

The simplest method for validating the taxonomy is through direct inspection of the categories. The category views described in section 2 provides many unique tools for validating that the membership of a category is not more or less than what the category means. Looking over some of the "Least Typical" documents is an especially valuable way to quickly ascertain that a category does not contain any documents that do not belong.

Another visual inspection method is to look at the nearest neighbors of the category being evaluated through the Scatter Plot display. Areas of document overlap at the margins are primary candidates for further investigation and validation.

5.2 Validation Metrics

Much research has been done in the area of evaluating the results of clustering algorithms [8] [20]. While such measures are not entirely applicable to taxonomies that have been modified to incorporate domain knowledge, there are some important concepts that can be applied from this research. Our vector space model representation [21] [22], while admittedly a very coarse reflection of the documents actual content, does at least allow us to summarize a singe level of the taxonomy via some useful statistics. These include:

- Cohesion: a measure of similarity within category. This is the average cosine distance of the documents within a category to the centroid of that category.
- Distinctness: a measure of differentiation between categories. This is one minus the cosine distance of the category to the centroid of the nearest neighboring category.

These two criteria are variations to the ones proposed by [1]: Compactness and Separation. The advantage of using this approach as opposed to other statistical validation techniques is that they are more easily computed and also readily understood by the taxonomy expert. In practice, these metrics often prove useful in identifying two potential areas of concern in a taxonomy. The first potential problem is "Miscellaneous" classes. These are classes that have a diffuse population of documents that talk about many different things. Such classes may need to be split further or subcategorized. The second potential problem is when two different categories have very similar content. If two or more classes are almost indistinguishable in terms of their word content, they may be candidates for merging.

Statistical measures such as Cohesion and Distinctness provide a good rough measure of how well the word content of a category reflects its underlying meaning. For example, if a category that the user has created is not cohesive, then there is some doubt as to whether a classifier could learn to recognize a new document as belonging to that category, because the category is not well defined in terms of word content. On the other hand, if a category is not distinct, then there is at least one other category containing documents with a similar vocabulary. This means that a classifier may have difficulty distinguishing which of the two similar categories to place a candidate document in. Of course, cohesion and distinctness are rough and relative metrics, therefore there is no fixed threshold value at which we can say that a category is not cohesive enough or lacks sufficient distinctness. In general, whenever a new category is created, we suggest to the user that the cohesion and distinctness score for the new category be no worse than the average for the current level of the taxonomy.

5.3 Other Metrics

Metrics such as cohesion and distinctness provide a rough measure of how well a given document taxonomy can be modeled and used to classify new documents. A more accurate measure can be

created by applying a suite of classification algorithms to a training sample of the data and seeing how accurately such classifiers work on a corresponding unseen test sample. If one or more of the classifiers can achieve a high level of accuracy on each of the categories, this indicates that there is sufficient regularity in the document word content to accurately categorize new documents, assuming the right modeling approach is used. For each of these classifiers the labeled training set consists of two thirds of the original document set, randomly selected without replacement. The test set is then the remaining one third. The label of each document is the category it belongs to. The resulting accuracy of such classifiers is not guaranteed to be high, even for the Centroid classifier. This is due to the fact that the user may have made arbitrary or inconsistent edits to the taxonomy using the methods described above. Applying these classification metrics then helps to make clear which of these edits can really be modeled using known supervised learning techniques. If a category created from user edits cannot be modeled accurately, then we cannot expect the system to automatically maintain its meaning in the future as new documents are added to the taxonomy.

We incorporated the following classifiers into our suite of available classifiers in the toolkit.

Centroid

This is the simplest classifier. It classifies each document to the nearest centroid (mean of the category) using a cosine distance metric.

Decision Tree

The decision tree algorithm is an implementation of the well-known ID3 algorithm [19]. Some classification algorithms benefit from additional reduction in the feature space [16][6]. In these algorithms we use a method to select terms based on their mutual information with the categories [16], and selecting all terms where the mutual information is above some threshold.

Naïve Bayes

We have incorporated two variations of Naïve Bayes classifier into our suite. The first is based upon numeric features (multinomial), the second on binary features (multivariate). Both use the well known Bayes decision rule and make the Naïve Bayes assumption [15][16][17] and differ only in how the probability of the document given the class, $P(d \mid C_k)$, is calculated.

In the multinomial model [16], classification is based upon the number of occurrences of each word in the document:

$$P(d \mid C_k) = \prod_{w_i \in d} P(w_i \mid C_k)$$

Where the individual word probabilities are calculated form the training data using Laplace smoothing [16]:

$$P(w_i \mid C_k) = \frac{n_{k,i} + 1}{n_k + \mid V \mid}$$

Where n_k is the total number of word positions in documents assigned to class C_k in the training data, $n_{k,i}$ is the number of positions in these documents where w_i occurs, and V is the set of all unique words.

The the multivariate model [16], calculates probabilities based on the whether words appear in documents, ignoring their frequency of occurrence:

$$P(d \mid C_k) = \prod_{w_i \in V} [B_i P(w_i \mid C_k) + (1 - B_i)(1 - P(w_i \mid C_k)]$$

Where B_i is 1 if w_i occurs in d and 0 otherwise, and the individual word probabilities are calculated as:

$$P(w_i \mid C_k) = \frac{1 + \sum_{d \in D} B_i P(C_k \mid d)}{2 + \sum_{d \in D} P(C_k \mid d)}$$

where $P(C_k \mid d)$ is 1 if d is in class C_k and 0 otherwise.

Rule Based

The rule induction classifier [12] is based on a decision tree system that takes advantage of the sparcity of text data feature space, and a rule simplification method that converts a decision tree into a logically equivalent rule set. The system also uses a modified entropy function that both favors splits enhancing the purity of partitions and, in contrast to the gini or standard entropy metrics, is close to the classification error curve, which has been found to improve text classification accuracy.

Statistical

The statistical classifier is a version of regularized linear classifier that has similar behavior as a support vector machine, but also provides a probability estimate for each class. It also employs the sparse regularization condition described in [29] to produce a sparse weight vector. The numerical algorithm is described in [29].

For each category, a Precision/Recall score is provided in the Category Table view that indicates how well that category can be modeled with that approach (see figure 6). Categories that cannot be modeled with any approach should be re-examined to see if they can be modified to make them more model-able.

[Insert Figure 6 Here]

In cases where no single classifier works adequately well for all categories, so that a "mixture" of classifiers is needed, it is possible to combine the results of several different classification algorithms into a single classifier [13][28]. This "mixture of experts" approach is partially based on the intuition that multiple generative processes may be involved in the creation of a taxonomy.

6 TEXT ANALYSIS VS. STRUCTURED INFORMATION

Up to this point, we have only considered the document text in our analysis of data. Of course in most cases text information has a corresponding structured component. Such structured information usually includes the creation/modification date of the document, the document author or assignee, geographic location, and so forth. Any complete analysis of the text information will need to take this structured information into account.

6.1 Time Analysis

Most text document data sets will contain at least one time stamp associated with each document. This information provides an opportunity to discover how the document collection has evolved over time. Using the time stamp for each document (usually the creation date) we can form an additional metric called "Recency". Recency is a measure of what percentage of the text documents were authored in the most recent time period. For example, we can define the most recent time period to be the last 10% of the document collection if the documents are sorted in chronological order. The Recency score for each category then becomes the percentage of that categories documents that fall in this the last 10%. The usefulness of the Recency metric is that it helps to discover categories that may contain new concepts as data is added to the taxonomy. Categories with a high level of Recency (greater than 10%) can be studied further by calling up a Trend Chart. Such a chart shows the category occurrence vs. Time when compared to the overall document occurrence vs. Time. Individual points in the chart can be selected to reveal the documents that correspond to any given time period.

6.2 Emerging Categories

Frequently an initial clustering formed via text clustering and edited using our techniques will be used to classify new data as it emerges. This "automatic categorization" is a powerful means of quickly assimilating new documents into a collection. The process for adding new documents to a taxonomy is

straightforward. We first use the original dictionary and classification model from the old document set to classify the new documents. Then we remember the classification labels for both the old and new documents and generate a new dictionary (of size N, where N is typically 2000 terms) on the combined set of new and old documents (see section 2.1). A new set of centroids can then be created from the complete set of categorized documents. Unfortunately, this process does not take into account any new emerging concepts that may require additional categories. To discover such categories we use a technique called "Recent Trends Analysis". Using our earlier definition of Recent (last 10% of the document collection), we analyze all the dictionary terms (i.e. the terms generated using the method described in section 2.1 from data in the overall taxonomy of both "new" and "old" documents) to determine which, if any, occur with an unusually high frequency in the Recent set. Unusually high is determined using a Chi-squared test, which determines the independence of two discrete random variables [18] and selecting those terms that occur with probability less than 0.01. The resulting term list is displayed to the user for further investigation via Trend Charts and Example Displays which can then be used to create new categories of documents.

6.3 Other Structured Information

In addition to dates, other kinds of structured information may be analyzed along with text. For such information a Contingency style table is displayed, showing the occurrence of the text clusters along one side of the table and the occurrence of the structured field in question along the other side. The cells of the table then display the co-occurrence counts (raw and percentage) for every combination of text cluster and structure field. The cells can be colored to indicate the relative likelihood of a particular combination occurring by random chance (see figure 7). Individual cells or combinations may be selected by the user for Trend Charts or Example displays which can then be used to create new categories of documents.

[Insert Figure 7 Here]

7 USAGE SCENARIO: INTERACTIVE TEXT MINING ON DISCUSSION FORUM DATA

One application of interactive text mining techniques is in the area of discussion forum analysis. In this paper, we describe in detail one such analysis that was done for IBM ValuesJam, an internal company wide, on-line discussion of IBM's corporate values. ValuesJam was a 72-hour global brainstorming event on the IBM internal website, held July 29 - August 1, 2003. IBMers described their experiences and contributed ideas via four asynchronous discussion forums. The purpose of real-time interactive text mining of the jam was to generate forum "topics" thus allowing participants to learn which themes are emerging in each forum -- and in the Jam overall -- in 12-hour intervals. Total posts for this event were in excess of 8000 over the course of the event, with one of the largest forum containing in excess of 3000 posts.

Analyzing discussion forum data to produce topic areas of interest presents several challenges which an interactive text mining approach is well suited to address. First the forum analyzer must produce categories that reflect meaningful groups of posts, and these groups must not contain a significant number of extraneous or misclassified examples. Second, each cluster of posts must be given a concise yet meaningful name. Third, when presenting a cluster of posts, a set of representative examples are needed to further explain the meaning of the post, and direct the user to the appropriate point in the discussion. Finally, the clusters need to evolve with the discussion, adding new clusters over time as appropriate to incorporate the new topics that will inevitably arise, without losing the old clusters and thus the overall continuity of the discussion topic list. Clearly a completely automated solution is impractical given these requirements, and a manual approach requiring a set of human editors to read 8000 posts in 72 hours and classify them is prohibitively expensive (and mind numbing). Interactive text mining is thus an ideal candidate for this application, and indeed our approach had been employed

successfully on two previous IBM "Jam" events. At the time of this writing the ValuesJam event was the largest discussion forum to which our approach had been applied and the first time that the results of our approach would be presented "live" to Jam participants.

7.1 Initial Taxonomy Generation

The first taxonomy generated for discussions in the largest forum of ValuesJam was created on 1308 posts representing 20 hours of discussion (see figure 8).

[Insert Figure 8 Here]

We begin by sorting the categories created via text clustering by their cohesion scores. This gives us a useful order in which to tackle the problem of quickly understanding the taxonomy, category by category, and making any necessary adjustments. We view each category in detail making any necessary adjustments and giving the category a new name if appropriate (e.g. the category name "stock price" was given to replace the name "stock" given by the system). Occasionally we find clusters that are formed based on words that are not relevant to the content of the post, such as for the "question,term" cluster in figure 9.

[Insert Figure 9 Here]

By viewing the Secondary Classes we can determine where the examples will go when the centroid for this class is removed (see figure 10).

[Insert Figure 10 Here]

Seeing that they will distribute themselves evenly throughout the taxonomy, we can feel safe in deleting the centroid without ill effect.

The Miscellaneous class requires special attention. Frequently individual dictionary terms can be used to extract a common set of examples from a Miscellaneous category and create a useful separate

category. An example here is a category centered around the word "trust". Clicking on the red "trust" bar in the bar graph in figure 11 will cause all those examples in Miscellaneous that contain the word "trust" to be selected. These can then be further edited and a new category called "trust" can be created in the taxonomy.

[Insert Figure 11 Here]

Finally the complete initial categorization emerges (figure 12).

[Insert Figure 12 Here]

Using our methodology and software text analysis tools, this entire process required only about a half hour of concentrated effort. Now we can use this information to generate reports to the ValuesJam audience. First we add a metric to our table to measure the "Recency" of the different categories (see figure 13).

[Insert Figure 13 Here]

Sorting by this metric reveals the key themes that have surfaced in most recently. Then we sort by size to discover the "cumulative themes". The resulting web page report is shown in Figure 14.

[Insert Figure 14 Here]

Selecting any of the above links will take the user to a display of 10 of the "most typical" comments for that theme. This process was then repeated for each of the remaining forums and for the Jam as a whole. The entire reporting operation took about 3-4 hours.

7.2 Emerging Themes

As the Jam progressed, new topics naturally emerged. To identify these, the Recent Trends analysis described earlier was especially valuable. A good example of this came late in the Jam when a breaking news story had an impact on the discussion [27].

[Insert Figure 15 Here]

In figure 15 the result of running a Recent Trend Analysis is shown. We observe the word "pension" occurred 51 times overall, and 11 times in the last 10% of the data. This was deemed by the software to be a low probability event (P=0.0056). A view of the trend line for this keyword shows the spike (see figure 16).

[Insert Figure 16 Here]

Pension mentions had been decreasing as a percentage of the total posts, but on the last day there was a sharp increase. Looking at the text for these examples quickly revealed the cause, and thus a new category was created centered around this word and the examples that used the word in a context related to the news event.

7.3 Success of Interactive Text Mining during ValuesJam

Our Interactive Text Mining approach showed itself quite capable of analyzing a discussion forum among thousands of users in real time using only a single human analyst with an IBM laptop PC. A survey of 1248 respondents done after ValuesJam indicated that 42% of all Jam participants used the theme pages generated by eClassifier to enter the Jam. The survey further shows that those who used this feature found it to be both important and satisfactory for the most part (72% important, 61% satisfactory). Only 10% of those who used this feature were dissatisfied with it.

8 CONCLUSIONS

In summary, we have described in detail a system with a unique combination of capabilities for the generation of practical quality taxonomies. We have shown that the combination of automated text mining with interactive human expert guidance integrated in an interactive platform provides a practical

way to create natural taxonomies in a document collection. Further user studies are required to validate that the methodology we have designed can work for a broader user population, and to compare its effectiveness to other possible approaches.

We believe there are many other practical aspects of taxonomy generation and utilization that are not well covered in the literature. One such issue is the generation of multiple taxonomies over a single collection of documents. This will enable applications to leverage multiple attributes and relationships of a collection of documents. Another area we believe to be a promising area for future work, will be the integration of taxonomies and text with structured and semi-structured data. [3]

9 ACKNOWLEDGEMENTS

The authors gratefully acknowledge Dharmendra Modha, Justin Lessler, and Ray Strong for their contributions to the original design of our interactive text mining approach. Thanks to Mike Wing, James Newswanger, and Kristine Lawas for their suggestions and facilitation in applying our technology to Values Jam.

10 REFERENCES

[1] Berry, J. and Linoff, G., (1996) Data Mining Techniques for Marketing, Sales, and Customer Support. John Willey & Sons, Inc.

[2] Brachman, R. and Anand T. (1996). The Process of Knowledge Discovery in Databases. In Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, P., and Uthurusamy, R., editors, Advances in Knowledge Discovery and Data Mining, Chapter 2, pages 37-58. AAAI/MIT press.

[3] Cody, W., Kreulen, J., Spangler, S., Krishna, V. (2002). The Integration of Business Intelligence and Knowledge Management. IBM Systems Journal, Vol. 41, No. 4, pp 697-713.

[4] Dhillon, I., Modha, D., and Spangler, S. (2002). Visualizing class structure of multidimensional data with applications. Journal of Computational Statistics & Data Analysis (special issue on Matrix Computations & Statistics) 4:1. November 2002. pp 59-90.

[5] Dom, B. (2001) "An Information-Theoretic External Cluster-Validity Measure", IBM Research Report RJ 10219, 10/5/2001

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

[7] Fox, C. (1992). Lexical Analysis and Stoplists. In Frakes, W. B. and Baeza-Yates, R., editors, Information Retrieval: Data Structures and Algorithms, pages 102-130. Prentice Hall, Englewood Cliffs, New Jersey.

[8] Halkidi, M., Batistakis, Y. Vazirgiannis, M. (2001) On Clustering Validation Techniques. Journal of Intelligent Information Systems 17(2-3): 107-145.

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

[10] Honrado, A., Leon, R., O'Donnel, R., Sinclair, D., A Word Stemming Algorithm for the Spanish Language, Seventh International Symposium on String Processing Information Retrieval, (SPIRE 2000).

[11] Jing, H., Barzilay, R., McKeown, K., and Elhadad, M. (1998) Summarization evaluation methods experiments and analysis. In AAAI Intelligent Text Summarization Workshop (Stanford, CA, Mar. 1998), pp. 60--68.

[12] Johnson, D. E., Oles, F. J., Zhang, T., and Goetz, T., 2002. A decision-tree-based symbolic rule induction system for text categorization. IBM Systems Journal 41:3, pp. 428-437.

[13] Kittler, J.,Hatef, M., Duin, R., and Matas, J., (1998) "On Combining Classifiers", IEEE Trans. on Pattern Analysis and Machine Intelligence, 20, pp. 226-239.

[14] Kornai, A., Stone, L., Automatic translation to controlled medical vocabularies. To appear in A. Abraham and L. Jain (editors): Innovations in Intelligent Systems and Applications Physica (Springer) Verlag, Germany (http://www.kornai.com/Papers/kornai_stone.pdf)

[15] Manning, Christopher D., Schütze, Hinrich (2000). Foundations of Statistical Natural Language Processing. The MIT Press.

[16] McCallum, Andrew, Nigam, Kamal. A Comparison of Event Models for Naïve Bayes Text Classification, AAAI-98.

[17] Mitchell, Tom M. (1997). Machine Learning. McGraw-Hill.

[18] Press, W. et. al. Numerical Recipes in C. 2nd Edition. New York: Cambridge University Press, 1992, 620-623.

[19] Quinlan, J.R. (1986) Induction of Decision Trees. *Machine Learning* 1 (1):81-106.

[20] 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.

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

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

[23] Spangler, S. and Kreulen, J. (2002). Interactive Methods for Taxonomy Editing and Validation. Proceedings of the Conference on Information and Knowledge Mining (CIKM 2002).

[24] Spangler, S., Kreulen, J., Lessler, J. (2003). Generating and Browsing Multiple Taxonomies over a Document Collection. Journal of Management Information Systems. Vol. 19:4, Spring 2003 pp 191-212.

[25] Vaithyanathan S. and Dom B. (2000) Hierarchical Unsupervised Learning. The Seventeenth International Conference on Machine Learning (ICML-2000).

[26] Vaithyanathan S. and Dom B (1999). Model Selection in Unsupervised Learning With Applications To Document Clustering. The Sixteenth International Conference on Machine Learning (ICML-99) Proceedings Published by Morgan Kaufman.

[27] Walsh, M. W., Judge Says IBM Pension Shift Illegally Harmed Older Workers, New York Times, August 1, 2003.

http://query.nytimes.com/gst/abstract.html?res=F00B14F73E5A0C728CDDA10894DB404482

[28] Wolpert, D.H. (1992), Stacked Generalization, Neural Networks, Vol. 5, pp. 241-259, Pergamon Press.

[29] Zhang, T. (2002), On the Dual Formulation of Regularized Linear Systems, Machine Learning, Vol. 46, pp 91-129.

FIGURES

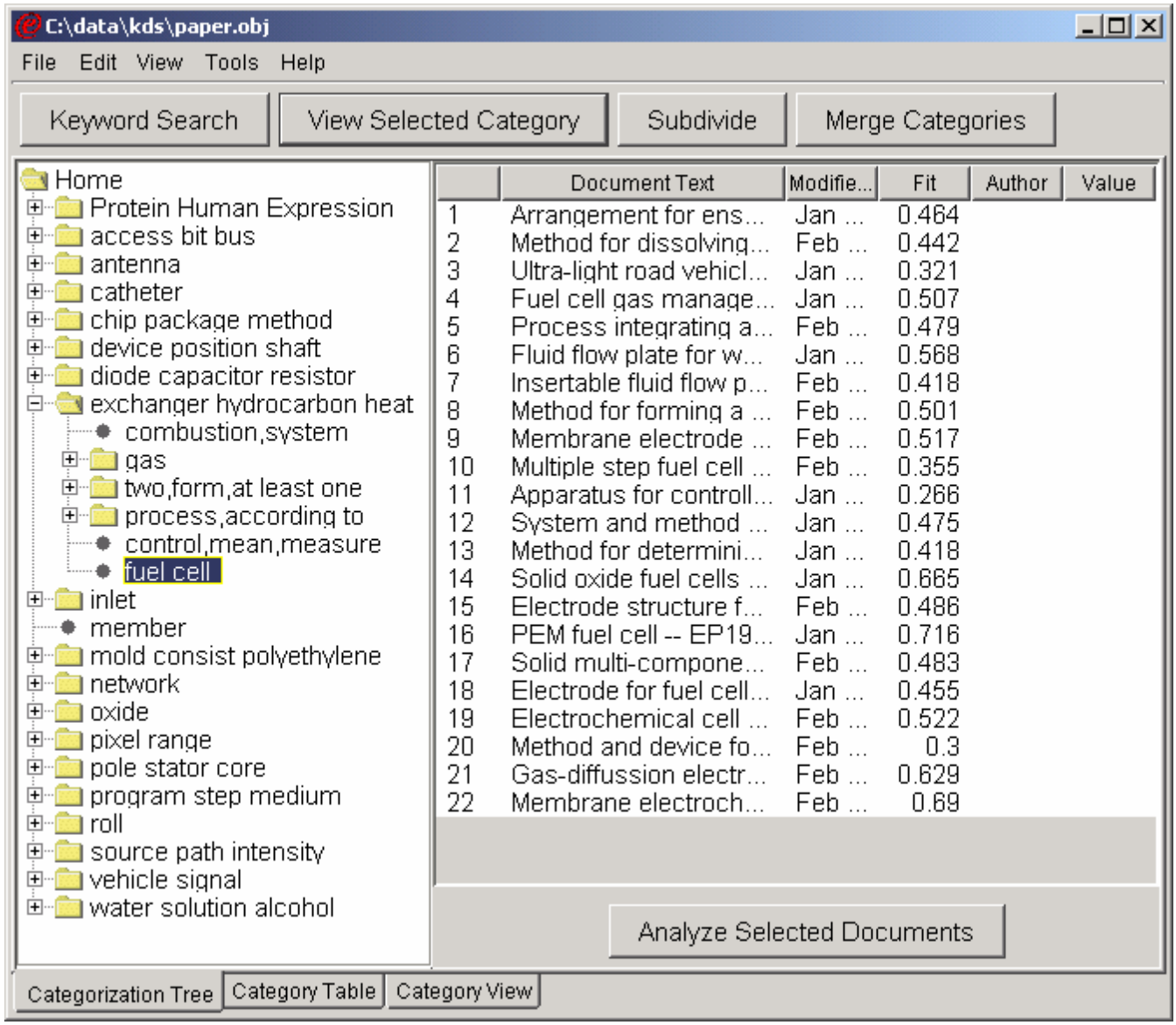


Figure 1: Categorization Tree

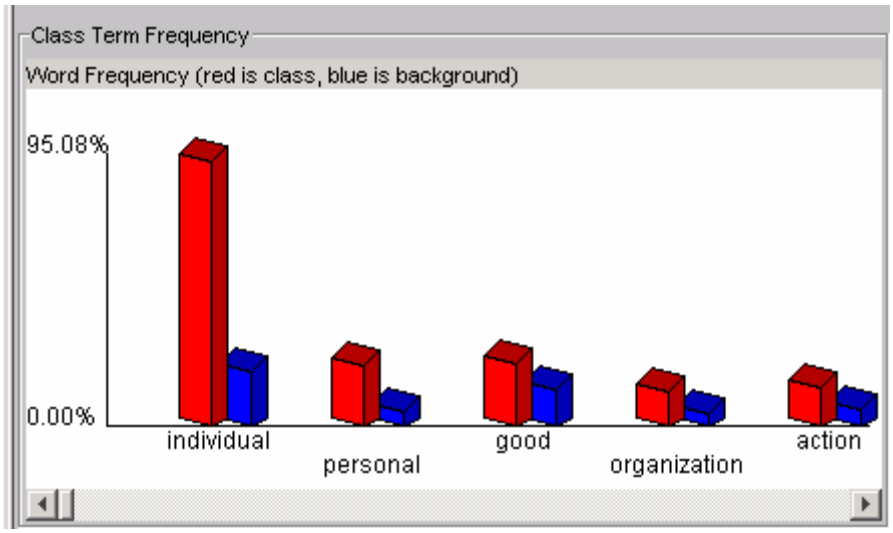


Figure 2: Term Frequency Bar Chart

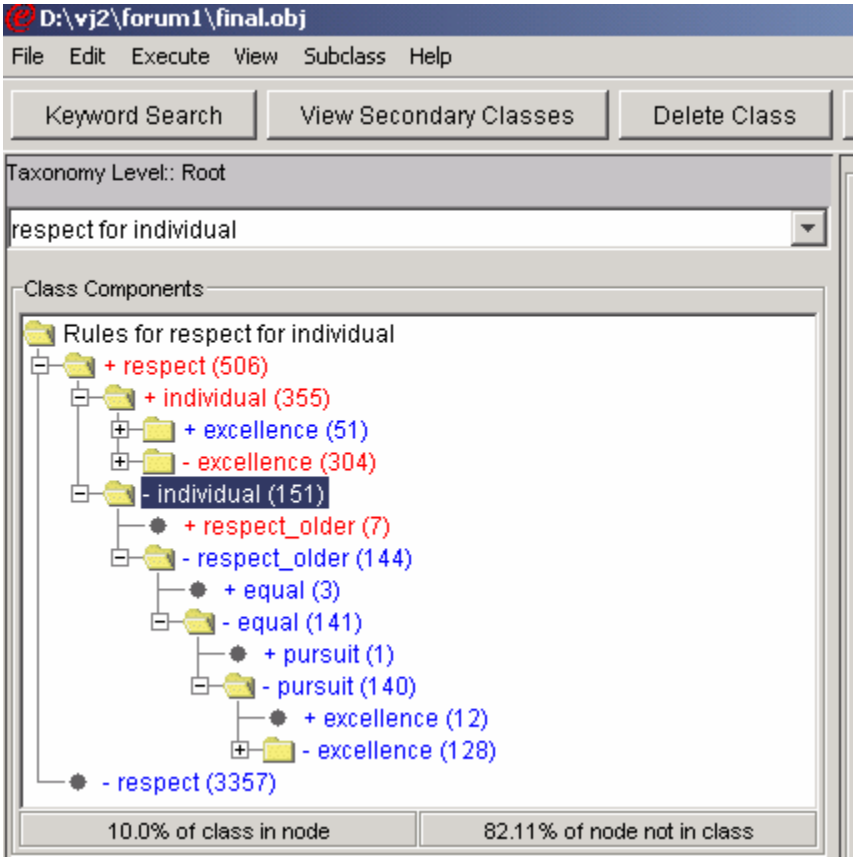


Figure 3: Dynamic Decision Tree View of Category

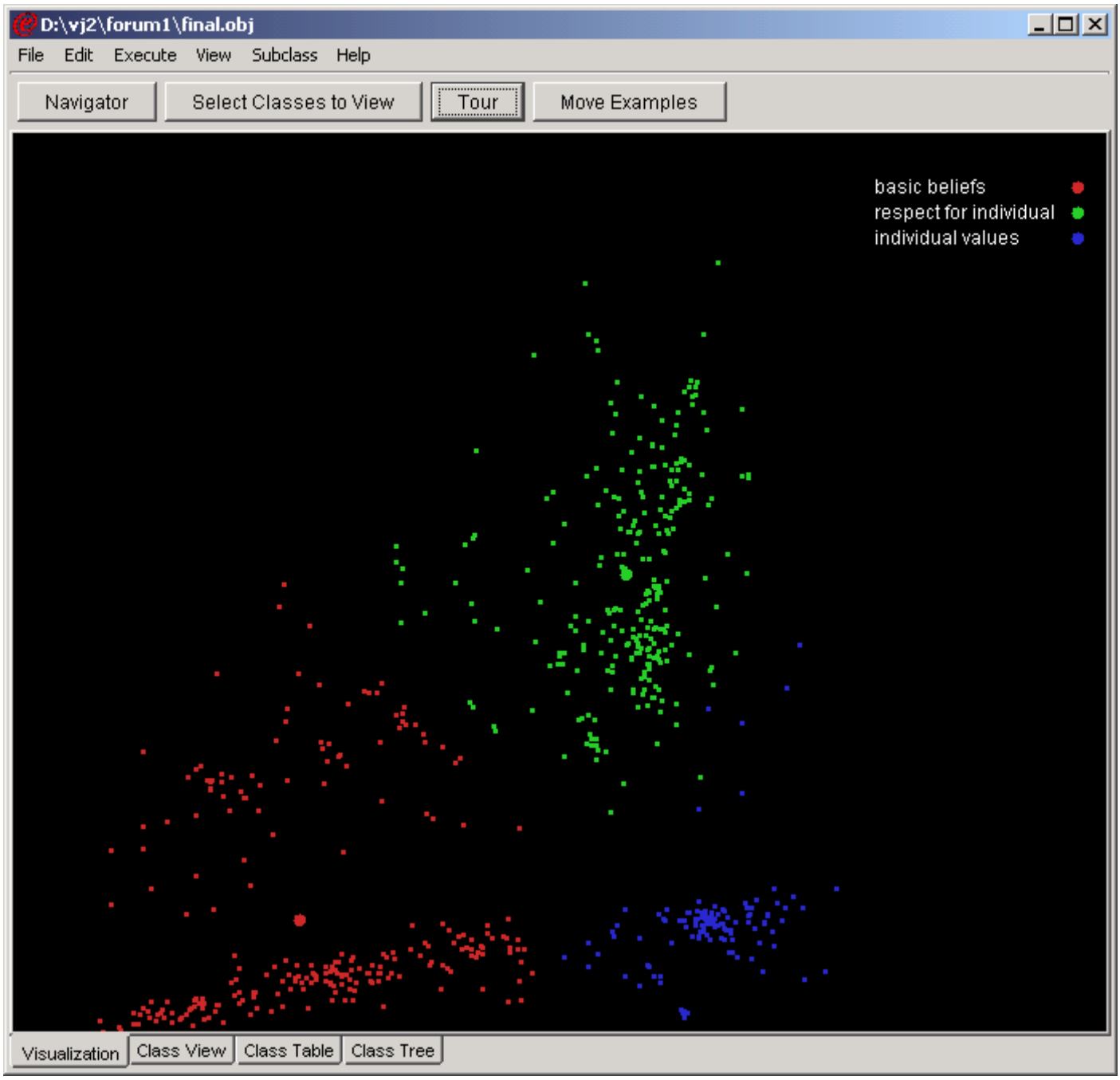


Figure 4: Class Visualization

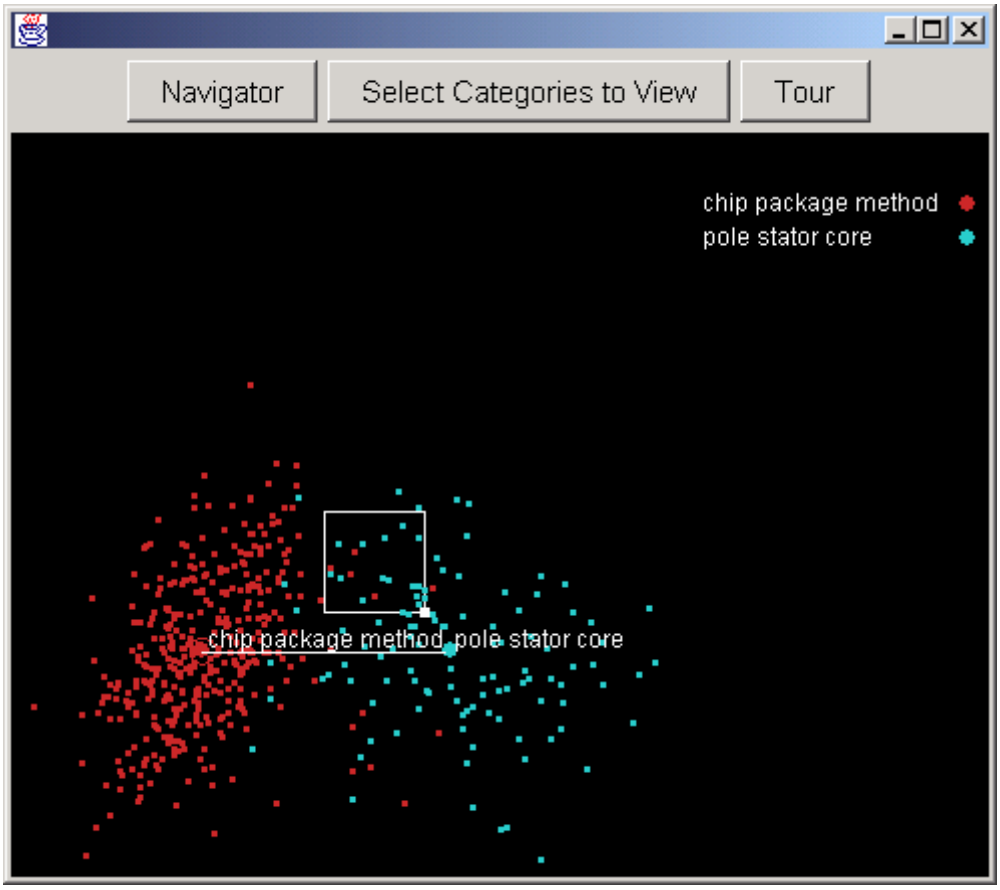
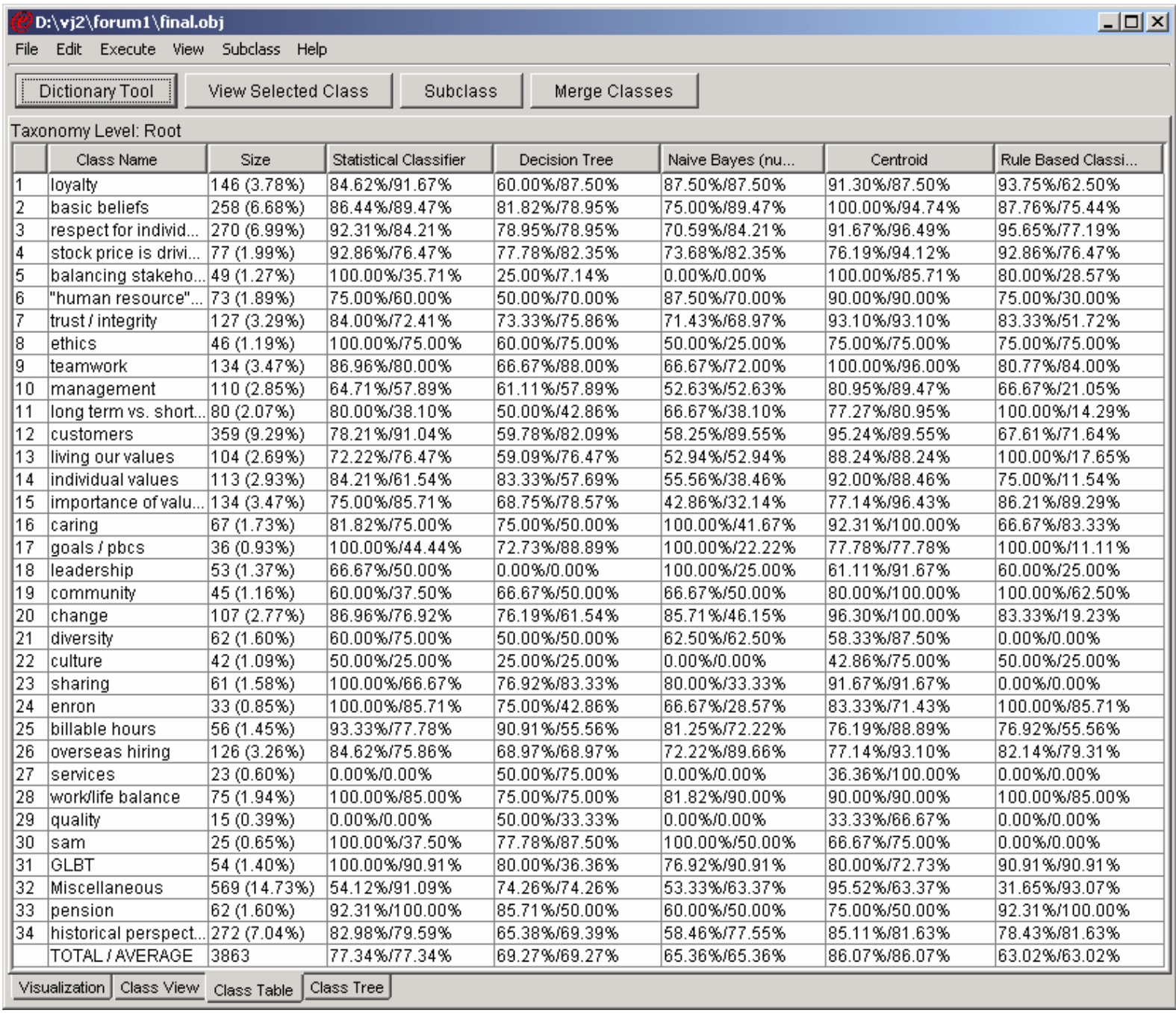


Figure 5: Floating box for moving documents



	Class Name	Size	Statistical Classifier	Decision Tree	Naive Bayes (nu...	Centroid	Rule Based Classi...
1	loyalty	146 (3.78%)	84.62%/91.67%	60.00%/87.50%	87.50%/87.50%	91.30%/87.50%	93.75%/62.50%
2	basic beliefs	258 (6.68%)	86.44%/89.47%	81.82%/78.95%	75.00%/89.47%	100.00%/94.74%	87.76%/75.44%
3	respect for individ...	270 (6.99%)	92.31%/84.21%	78.95%/78.95%	70.59%/84.21%	91.67%/96.49%	95.65%/77.19%
4	stock price is drivi...	77 (1.99%)	92.86%/76.47%	77.78%/82.35%	73.68%/82.35%	76.19%/94.12%	92.86%/76.47%
5	balancing stakeho...	49 (1.27%)	100.00%/35.71%	25.00%/7.14%	0.00%/0.00%	100.00%/85.71%	80.00%/28.57%
6	"human resource"...	73 (1.89%)	75.00%/60.00%	50.00%/70.00%	87.50%/70.00%	90.00%/90.00%	75.00%/30.00%
7	trust / integrity	127 (3.29%)	84.00%/72.41%	73.33%/75.86%	71.43%/68.97%	93.10%/93.10%	83.33%/51.72%
8	ethics	46 (1.19%)	100.00%/75.00%	60.00%/75.00%	50.00%/25.00%	75.00%/75.00%	75.00%/75.00%
9	teamwork	134 (3.47%)	86.96%/80.00%	66.67%/88.00%	66.67%/72.00%	100.00%/96.00%	80.77%/84.00%
10	management	110 (2.85%)	64.71%/57.89%	61.11%/57.89%	52.63%/52.63%	80.95%/89.47%	66.67%/21.05%
11	long term vs. short...	80 (2.07%)	80.00%/38.10%	50.00%/42.86%	66.67%/38.10%	77.27%/80.95%	100.00%/14.29%
12	customers	359 (9.29%)	78.21%/91.04%	59.78%/82.09%	58.25%/89.55%	95.24%/89.55%	67.61%/71.64%
13	living our values	104 (2.69%)	72.22%/76.47%	59.09%/76.47%	52.94%/52.94%	88.24%/88.24%	100.00%/17.65%
14	individual values	113 (2.93%)	84.21%/61.54%	83.33%/57.69%	55.56%/38.46%	92.00%/88.46%	75.00%/11.54%
15	importance of valu...	134 (3.47%)	75.00%/85.71%	68.75%/78.57%	42.86%/32.14%	77.14%/96.43%	86.21%/89.29%
16	caring	67 (1.73%)	81.82%/75.00%	75.00%/50.00%	100.00%/41.67%	92.31%/100.00%	66.67%/83.33%
17	goals / pbcs	36 (0.93%)	100.00%/44.44%	72.73%/88.89%	100.00%/22.22%	77.78%/77.78%	100.00%/11.11%
18	leadership	53 (1.37%)	66.67%/50.00%	0.00%/0.00%	100.00%/25.00%	61.11%/91.67%	60.00%/25.00%
19	community	45 (1.16%)	60.00%/37.50%	66.67%/50.00%	66.67%/50.00%	80.00%/100.00%	100.00%/62.50%
20	change	107 (2.77%)	86.96%/76.92%	76.19%/61.54%	85.71%/46.15%	96.30%/100.00%	83.33%/19.23%
21	diversity	62 (1.60%)	60.00%/75.00%	50.00%/50.00%	62.50%/62.50%	58.33%/87.50%	0.00%/0.00%
22	culture	42 (1.09%)	50.00%/25.00%	25.00%/25.00%	0.00%/0.00%	42.86%/75.00%	50.00%/25.00%
23	sharing	61 (1.58%)	100.00%/66.67%	76.92%/83.33%	80.00%/33.33%	91.67%/91.67%	0.00%/0.00%
24	enron	33 (0.85%)	100.00%/85.71%	75.00%/42.86%	66.67%/28.57%	83.33%/71.43%	100.00%/85.71%
25	billable hours	56 (1.45%)	93.33%/77.78%	90.91%/55.56%	81.25%/72.22%	76.19%/88.89%	76.92%/55.56%
26	overseas hiring	126 (3.26%)	84.62%/75.86%	68.97%/68.97%	72.22%/89.66%	77.14%/93.10%	82.14%/79.31%
27	services	23 (0.60%)	0.00%/0.00%	50.00%/75.00%	0.00%/0.00%	36.36%/100.00%	0.00%/0.00%
28	work/life balance	75 (1.94%)	100.00%/85.00%	75.00%/75.00%	81.82%/90.00%	90.00%/90.00%	100.00%/85.00%
29	quality	15 (0.39%)	0.00%/0.00%	50.00%/33.33%	0.00%/0.00%	33.33%/66.67%	0.00%/0.00%
30	sam	25 (0.65%)	100.00%/37.50%	77.78%/87.50%	100.00%/50.00%	66.67%/75.00%	0.00%/0.00%
31	GLBT	54 (1.40%)	100.00%/90.91%	80.00%/36.36%	76.92%/90.91%	80.00%/72.73%	90.91%/90.91%
32	Miscellaneous	569 (14.73%)	54.12%/91.09%	74.26%/74.26%	53.33%/63.37%	95.52%/63.37%	31.65%/93.07%
33	pension	62 (1.60%)	92.31%/100.00%	85.71%/50.00%	60.00%/50.00%	75.00%/50.00%	92.31%/100.00%
34	historical perspect...	272 (7.04%)	82.98%/79.59%	65.38%/69.39%	58.46%/77.55%	85.11%/81.63%	78.43%/81.63%
	TOTAL / AVERAGE	3863	77.34%/77.34%	69.27%/69.27%	65.36%/65.36%	86.07%/86.07%	63.02%/63.02%

Figure 6: Precision/Recall for different classifiers

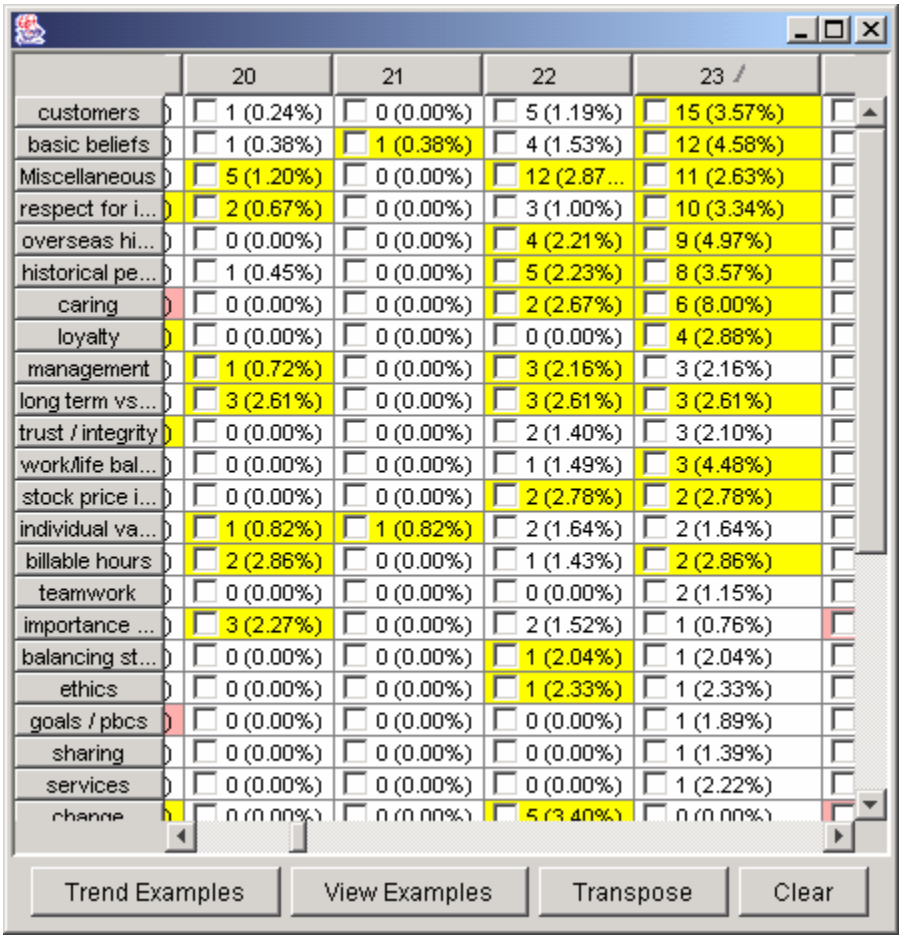
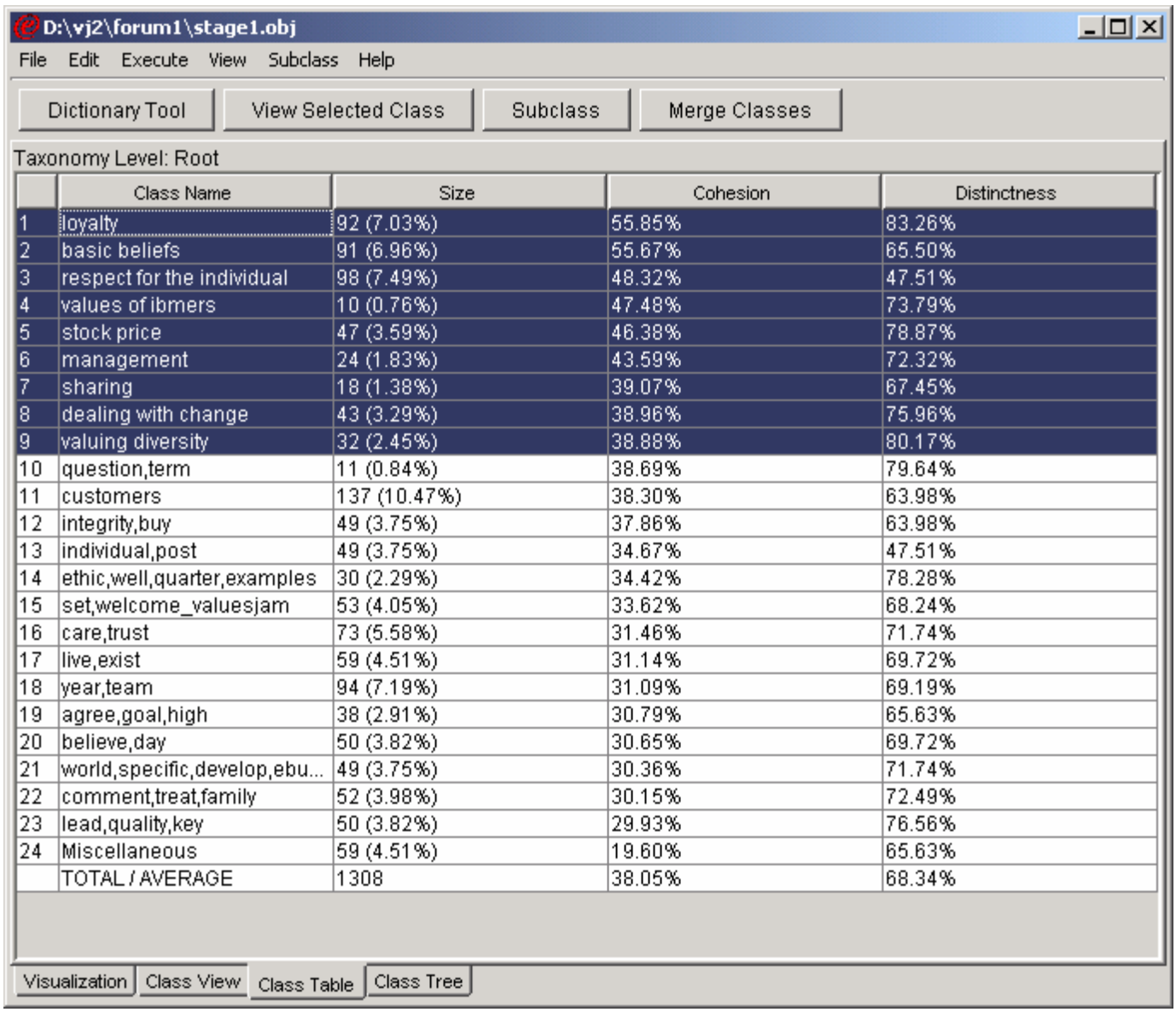


Figure 7: Comparing clusters to structured information



	Class Name	Size	Cohesion	Distinctness
1	loyalty	92 (7.03%)	55.85%	83.26%
2	basic beliefs	91 (6.96%)	55.67%	65.50%
3	respect for the individual	98 (7.49%)	48.32%	47.51%
4	values of ibmers	10 (0.76%)	47.48%	73.79%
5	stock price	47 (3.59%)	46.38%	78.87%
6	management	24 (1.83%)	43.59%	72.32%
7	sharing	18 (1.38%)	39.07%	67.45%
8	dealing with change	43 (3.29%)	38.96%	75.96%
9	valuing diversity	32 (2.45%)	38.88%	80.17%
10	question,term	11 (0.84%)	38.69%	79.64%
11	customers	137 (10.47%)	38.30%	63.98%
12	integrity,buy	49 (3.75%)	37.86%	63.98%
13	individual,post	49 (3.75%)	34.67%	47.51%
14	ethic,well,quarter,examples	30 (2.29%)	34.42%	78.28%
15	set,welcome_valuesjam	53 (4.05%)	33.62%	68.24%
16	care,trust	73 (5.58%)	31.46%	71.74%
17	live,exist	59 (4.51%)	31.14%	69.72%
18	year,team	94 (7.19%)	31.09%	69.19%
19	agree,goal,high	38 (2.91%)	30.79%	65.63%
20	believe,day	50 (3.82%)	30.65%	69.72%
21	world,specific,develop,ebu...	49 (3.75%)	30.36%	71.74%
22	comment,treat,family	52 (3.98%)	30.15%	72.49%
23	lead,quality,key	50 (3.82%)	29.93%	76.56%
24	Miscellaneous	59 (4.51%)	19.60%	65.63%
	TOTAL / AVERAGE	1308	38.05%	68.34%

Figure 8: Class Table View

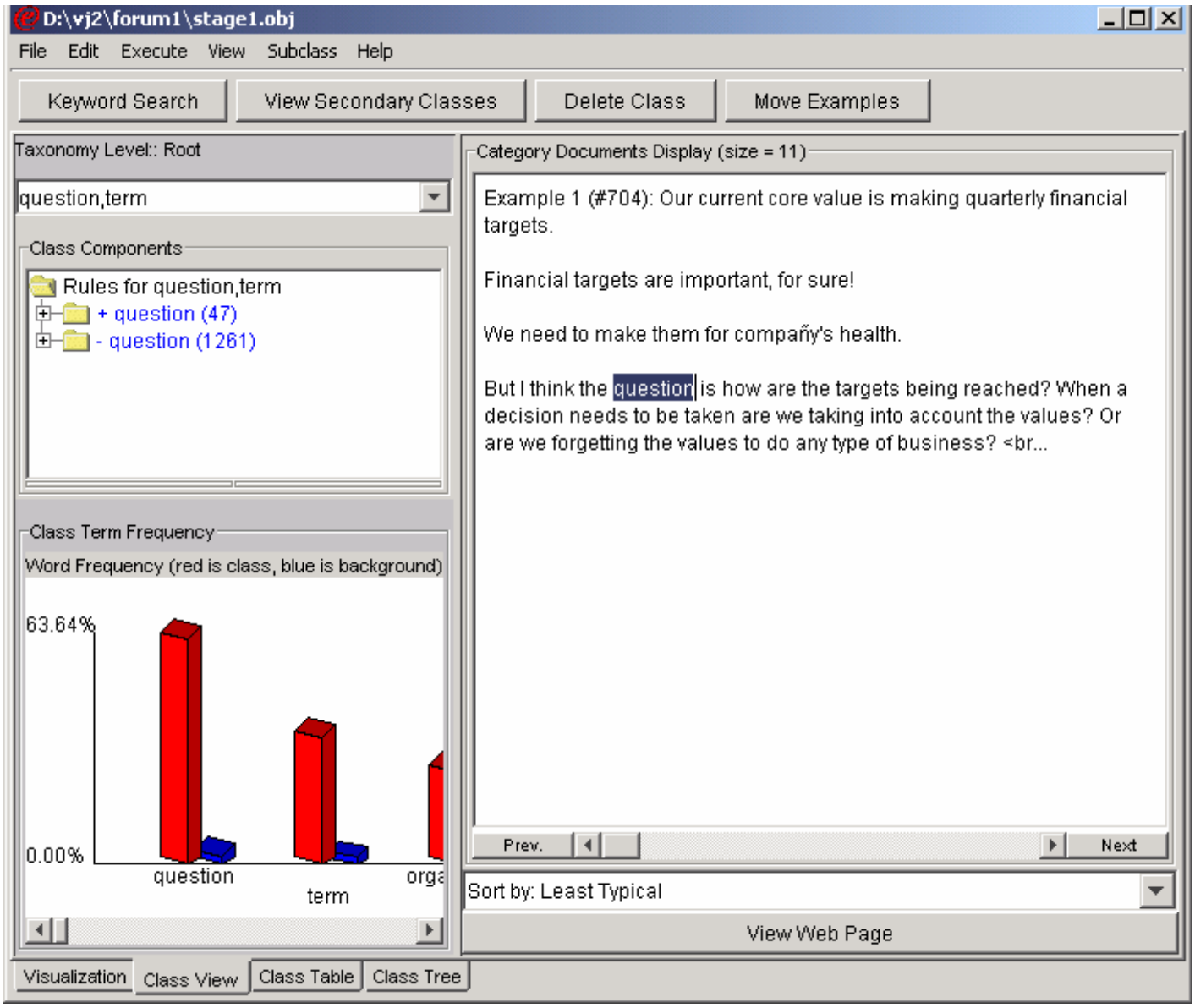


Figure 9: Class View

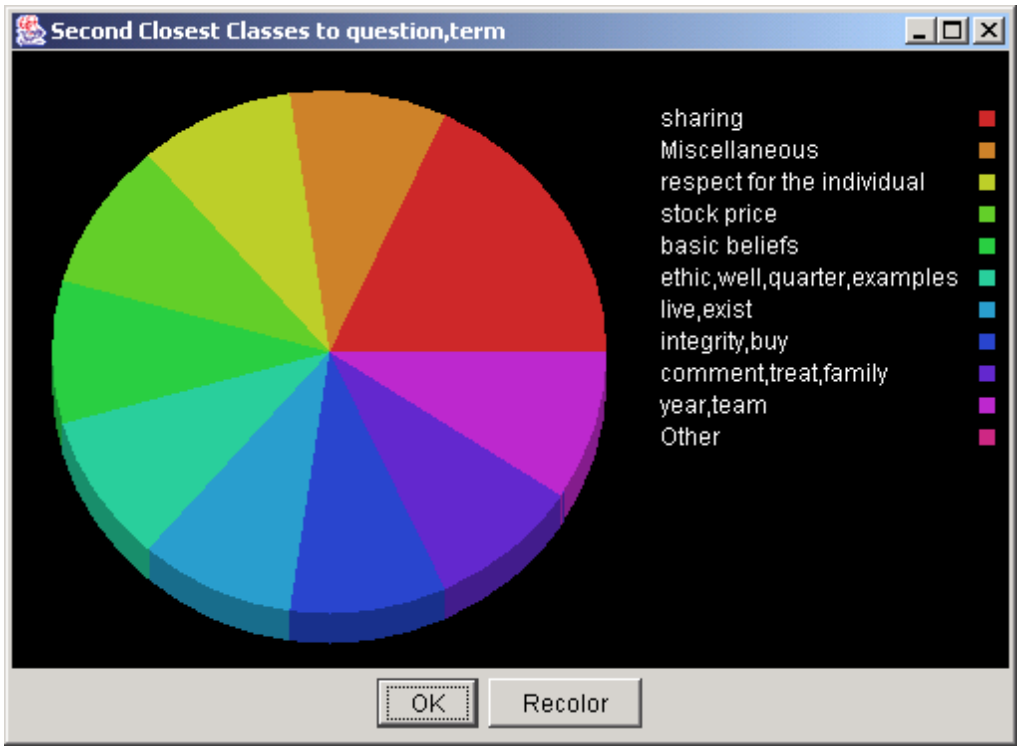


Figure 10: Secondary Classes

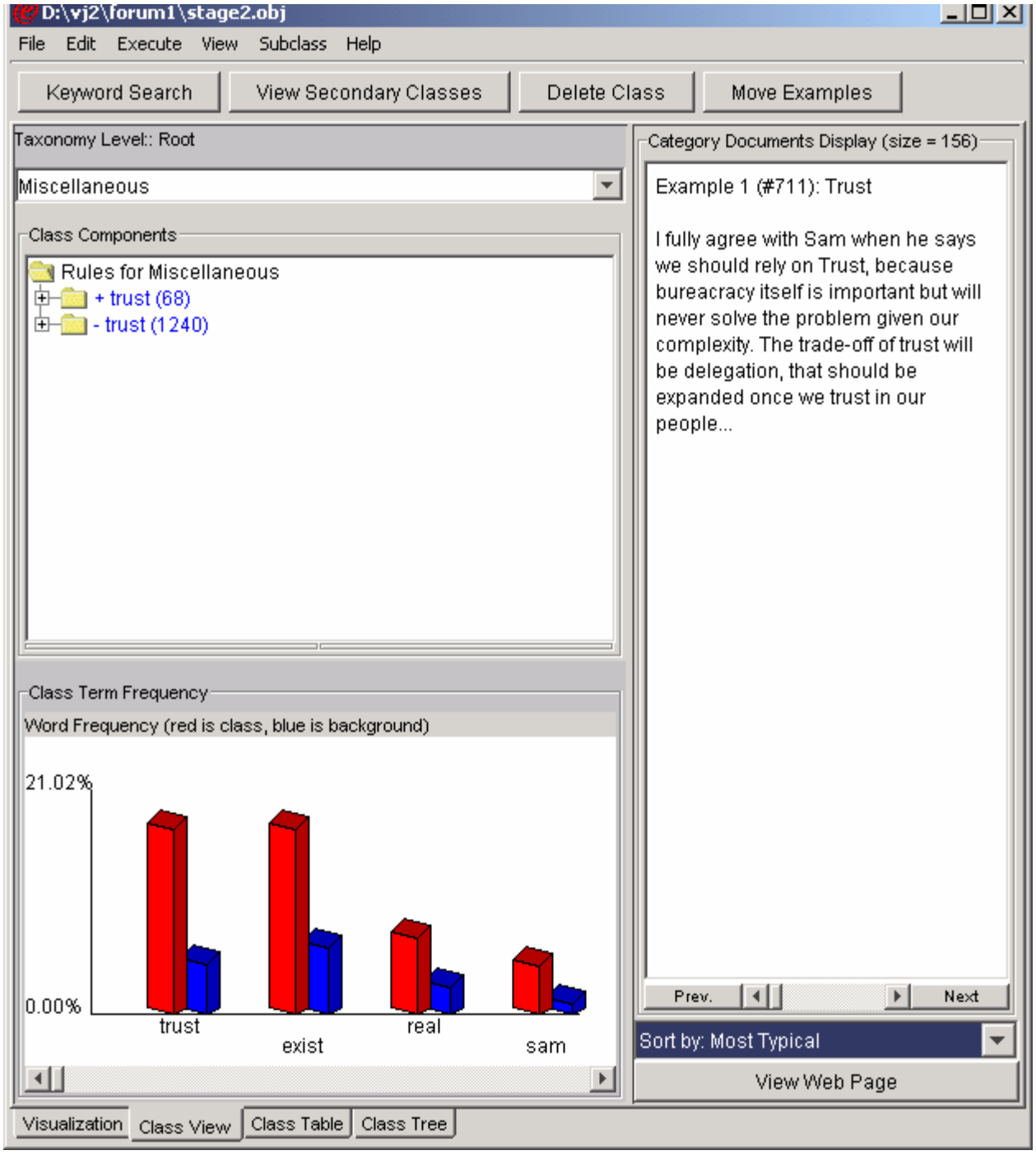
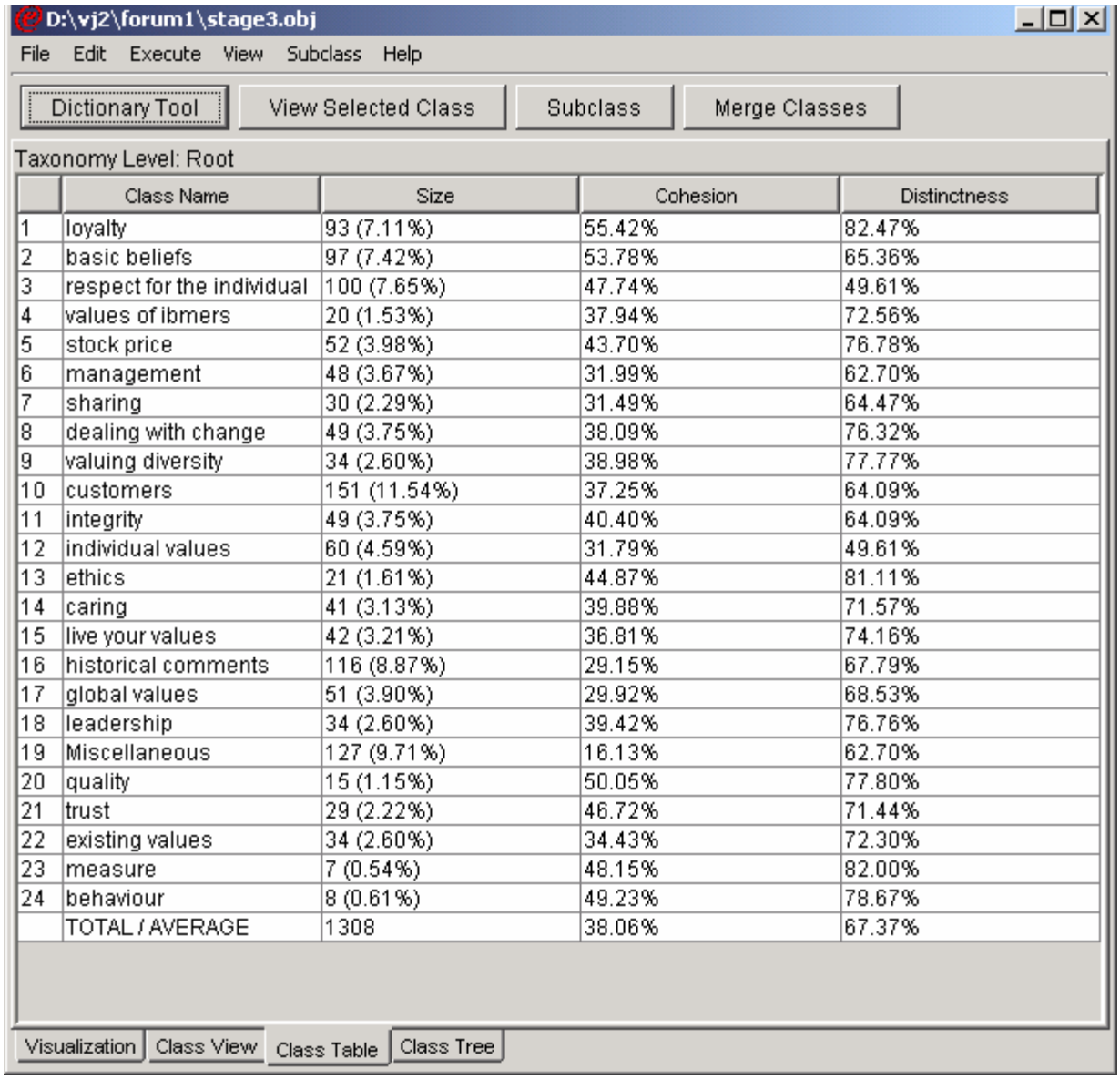
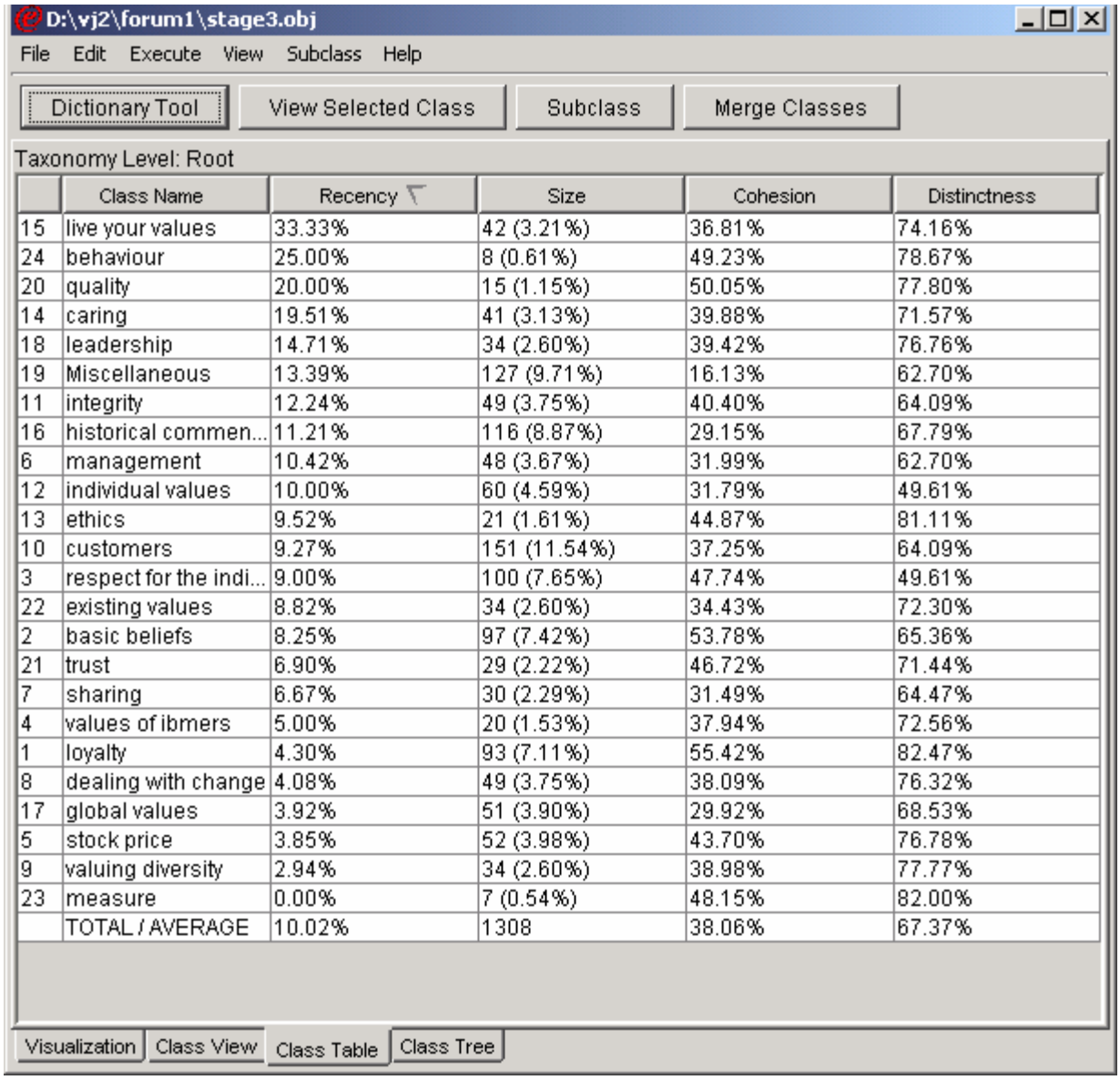


Figure 11: Miscellaneous class



	Class Name	Size	Cohesion	Distinctness
1	loyalty	93 (7.11%)	55.42%	82.47%
2	basic beliefs	97 (7.42%)	53.78%	65.36%
3	respect for the individual	100 (7.65%)	47.74%	49.61%
4	values of ibmers	20 (1.53%)	37.94%	72.56%
5	stock price	52 (3.98%)	43.70%	76.78%
6	management	48 (3.67%)	31.99%	62.70%
7	sharing	30 (2.29%)	31.49%	64.47%
8	dealing with change	49 (3.75%)	38.09%	76.32%
9	valuing diversity	34 (2.60%)	38.98%	77.77%
10	customers	151 (11.54%)	37.25%	64.09%
11	integrity	49 (3.75%)	40.40%	64.09%
12	individual values	60 (4.59%)	31.79%	49.61%
13	ethics	21 (1.61%)	44.87%	81.11%
14	caring	41 (3.13%)	39.88%	71.57%
15	live your values	42 (3.21%)	36.81%	74.16%
16	historical comments	116 (8.87%)	29.15%	67.79%
17	global values	51 (3.90%)	29.92%	68.53%
18	leadership	34 (2.60%)	39.42%	76.76%
19	Miscellaneous	127 (9.71%)	16.13%	62.70%
20	quality	15 (1.15%)	50.05%	77.80%
21	trust	29 (2.22%)	46.72%	71.44%
22	existing values	34 (2.60%)	34.43%	72.30%
23	measure	7 (0.54%)	48.15%	82.00%
24	behaviour	8 (0.61%)	49.23%	78.67%
	TOTAL / AVERAGE	1308	38.06%	67.37%

Figure 12: Completed Categorization



	Class Name	Recency	Size	Cohesion	Distinctness
15	live your values	33.33%	42 (3.21%)	36.81%	74.16%
24	behaviour	25.00%	8 (0.61%)	49.23%	78.67%
20	quality	20.00%	15 (1.15%)	50.05%	77.80%
14	caring	19.51%	41 (3.13%)	39.88%	71.57%
18	leadership	14.71%	34 (2.60%)	39.42%	76.76%
19	Miscellaneous	13.39%	127 (9.71%)	16.13%	62.70%
11	integrity	12.24%	49 (3.75%)	40.40%	64.09%
16	historical commen...	11.21%	116 (8.87%)	29.15%	67.79%
6	management	10.42%	48 (3.67%)	31.99%	62.70%
12	individual values	10.00%	60 (4.59%)	31.79%	49.61%
13	ethics	9.52%	21 (1.61%)	44.87%	81.11%
10	customers	9.27%	151 (11.54%)	37.25%	64.09%
3	respect for the indi...	9.00%	100 (7.65%)	47.74%	49.61%
22	existing values	8.82%	34 (2.60%)	34.43%	72.30%
2	basic beliefs	8.25%	97 (7.42%)	53.78%	65.36%
21	trust	6.90%	29 (2.22%)	46.72%	71.44%
7	sharing	6.67%	30 (2.29%)	31.49%	64.47%
4	values of ibmers	5.00%	20 (1.53%)	37.94%	72.56%
1	loyalty	4.30%	93 (7.11%)	55.42%	82.47%
8	dealing with change	4.08%	49 (3.75%)	38.09%	76.32%
17	global values	3.92%	51 (3.90%)	29.92%	68.53%
5	stock price	3.85%	52 (3.98%)	43.70%	76.78%
9	valuing diversity	2.94%	34 (2.60%)	38.98%	77.77%
23	measure	0.00%	7 (0.54%)	48.15%	82.00%
	TOTAL / AVERAGE	10.02%	1308	38.06%	67.37%

Figure 13: Recency Metric

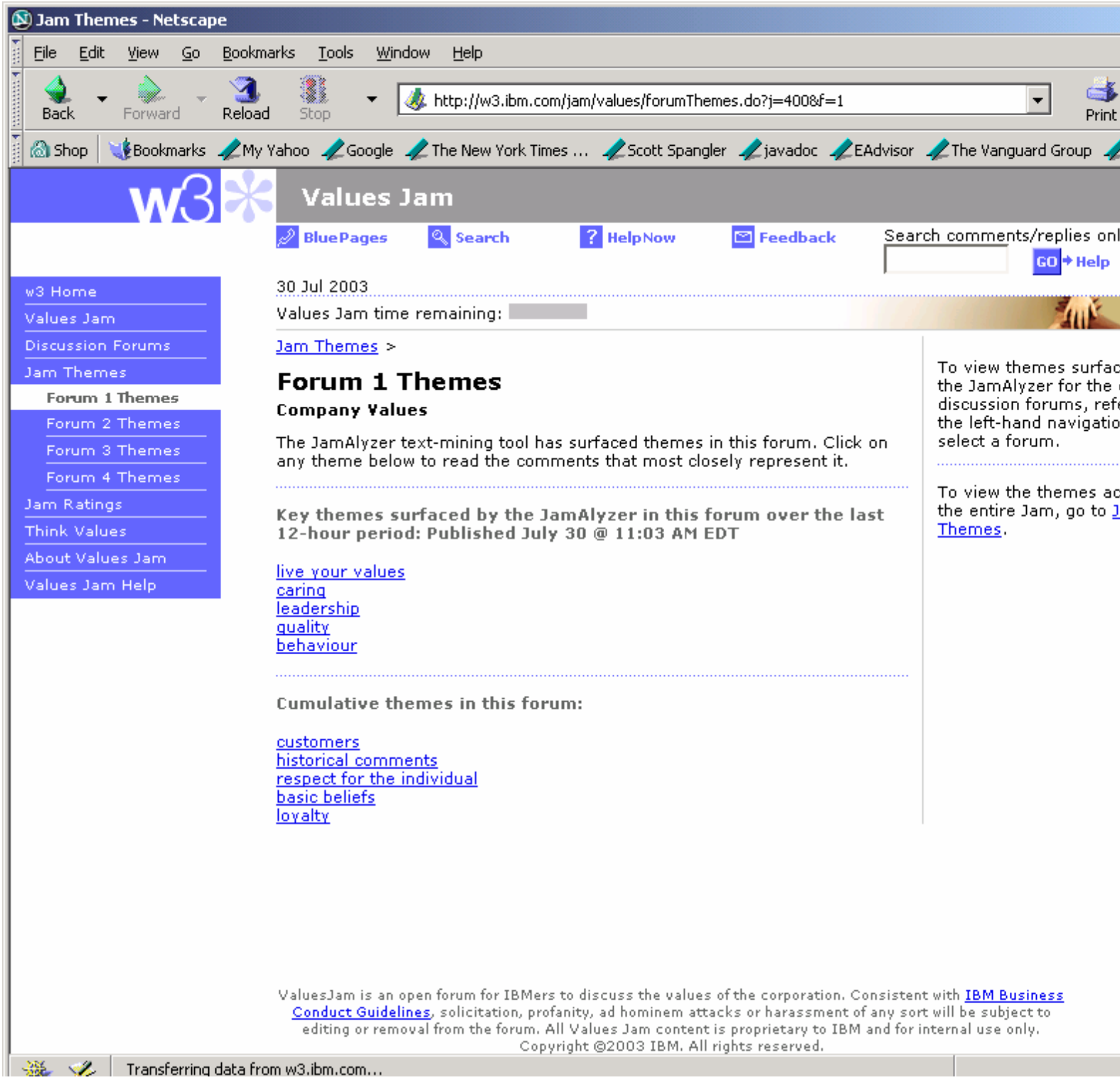
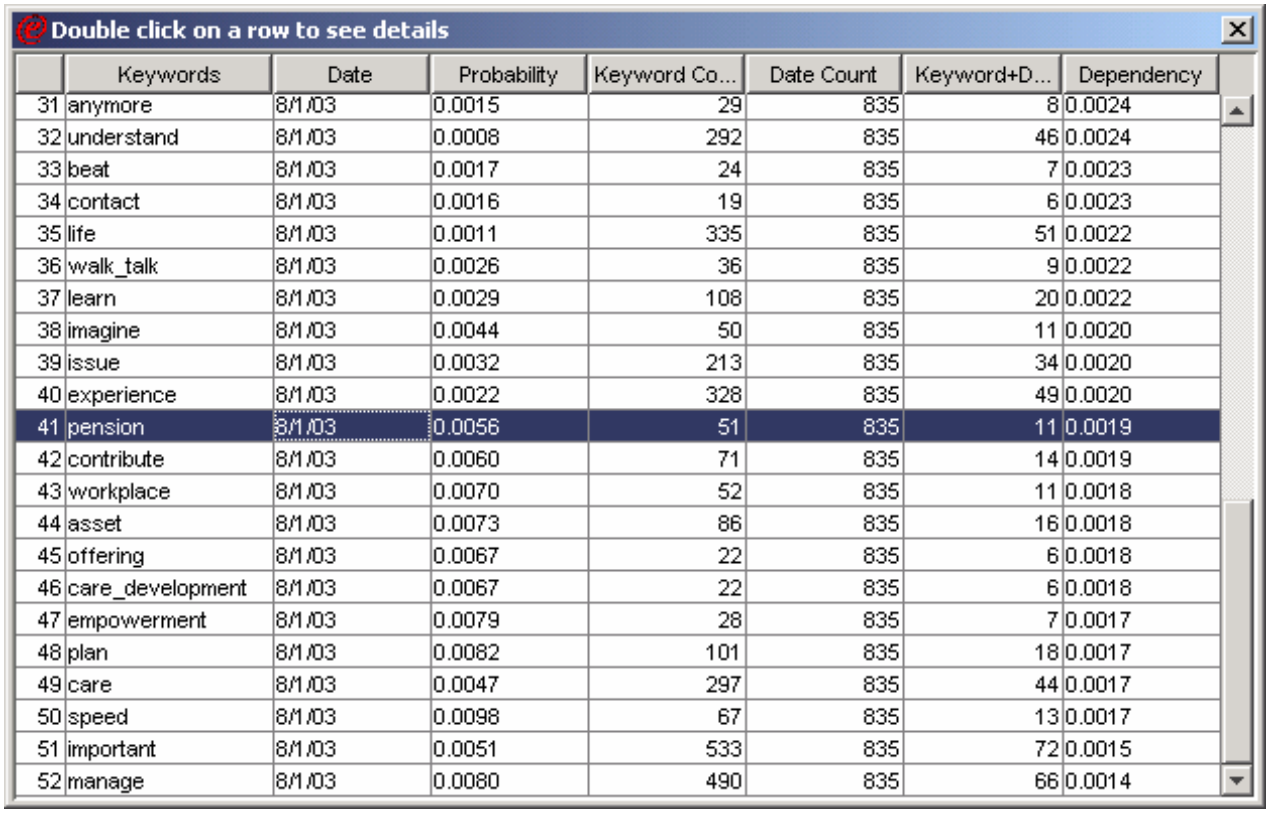


Figure 14: Jam Themes



	Keywords	Date	Probability	Keyword Co...	Date Count	Keyword+D...	Dependency
31	anymore	8/1/03	0.0015	29	835	8	0.0024
32	understand	8/1/03	0.0008	292	835	46	0.0024
33	beat	8/1/03	0.0017	24	835	7	0.0023
34	contact	8/1/03	0.0016	19	835	6	0.0023
35	life	8/1/03	0.0011	335	835	51	0.0022
36	walk_talk	8/1/03	0.0026	36	835	9	0.0022
37	learn	8/1/03	0.0029	108	835	20	0.0022
38	imagine	8/1/03	0.0044	50	835	11	0.0020
39	issue	8/1/03	0.0032	213	835	34	0.0020
40	experience	8/1/03	0.0022	328	835	49	0.0020
41	pension	8/1/03	0.0056	51	835	11	0.0019
42	contribute	8/1/03	0.0060	71	835	14	0.0019
43	workplace	8/1/03	0.0070	52	835	11	0.0018
44	asset	8/1/03	0.0073	86	835	16	0.0018
45	offering	8/1/03	0.0067	22	835	6	0.0018
46	care_development	8/1/03	0.0067	22	835	6	0.0018
47	empowerment	8/1/03	0.0079	28	835	7	0.0017
48	plan	8/1/03	0.0082	101	835	18	0.0017
49	care	8/1/03	0.0047	297	835	44	0.0017
50	speed	8/1/03	0.0098	67	835	13	0.0017
51	important	8/1/03	0.0051	533	835	72	0.0015
52	manage	8/1/03	0.0080	490	835	66	0.0014

Figure 15: Recent Trends

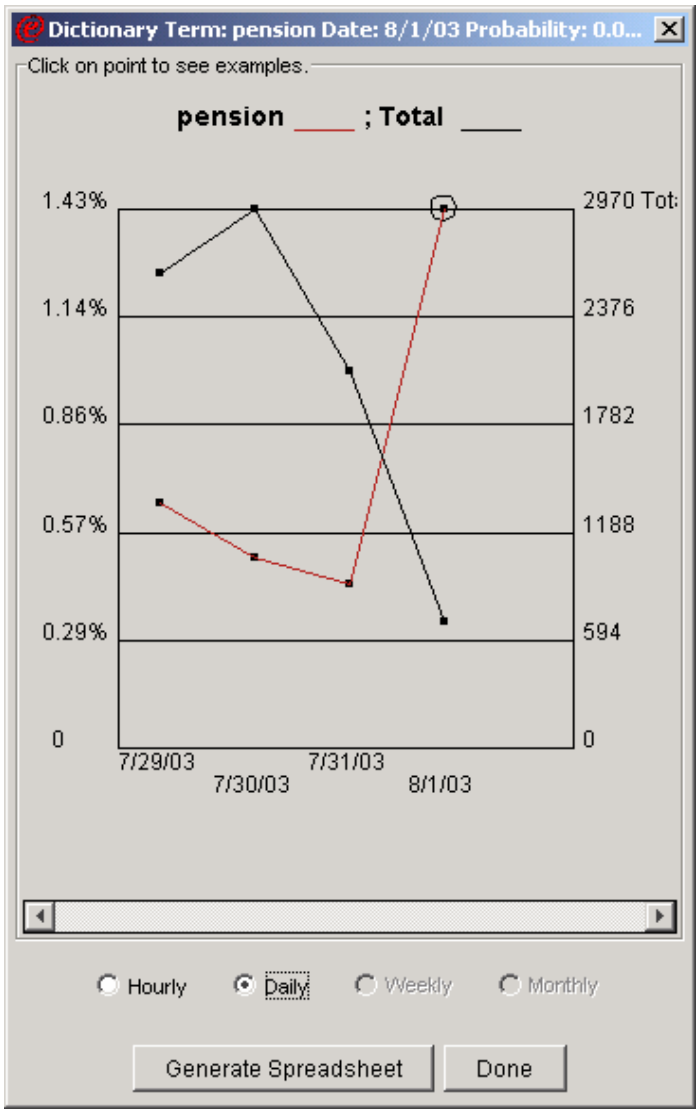


Figure 16: Time Graph