Vannevar Bush Best Paper Candidate

Exploring Erotics in Emily Dickinson's Correspondence with Text Mining and Visual Interfaces

Catherine Plaisant*
James Rose*#
*Human-Computer Interaction Lab.
#Computer Science Dept.
University of Maryland
+1 (301) 405-2768
plaisant@cs.umd.edu

Bei Yu&
Loretta Auvil+
&GSLIS
+NCSA
University of Illinois
lauvil@ncsa.uiuc.edu

Matthew G. Kirschenbaum
Martha Nell Smith
Tanya Clement
Greg Lord
Dept. of English and MITH
University of Maryland
mgk@umd.edu

ABSTRACT
This paper describes a system to support humanities scholars in their interpretation of literary work. It presents a user interface and web architecture that integrates text mining, a graphical user interface and visualization, while attempting to remain easy to use by non specialists. Users can interactively read and rate documents found in a digital libraries collection, prepare training sets, review results of classification algorithms and explore possible indicators and explanations. Initial evaluation steps suggest that there is a rationale for "provocational" text mining in literary interpretation.

Categories and Subject Descriptors
J.5 [**Arts and Humanities**]: Literature; H.3.6 [**Library Automation**]: Large text archives; H.5.2 [**Information Interfaces and Presentation**]: User Interfaces. I.5.4 [**Pattern Recognition**]: Applications – Text Processing

General Terms: Design, Experimentation, Human Factors

Keywords: User interface, text mining, visualization, literary criticism, humanities, case studies.

1. INTRODUCTION
This paper describes a system to support humanities scholars in their interpretation of literary works. While extensive digital libraries for literature and other areas of the humanities are now available, their infrastructure has been mainly focused on providing access to large repositories of encoded documents (and occasionally images or other multimedia materials). Examples include the Women Writers Project (www.wwp.brown.edu), the Valley of the Shadow (valley.vcdh.virginia.edu), and the Dickinson Electronic Archive (www.emilydickinson.org). Those archives are seen as supplying the raw material of humanities scholarship, but beyond access, search, and retrieval, computers are not otherwise regarded as tools that contribute to the basic mission of most humanities scholars: critical interpretation. We propose to go further and allow users of digital library collections to conduct their scholarly work online using web-based processing tools that serve as instruments for provoking interpretation. Specifically, this paper presents a user interface and web architecture that integrates text mining and an interactive visual user interface, while attempting to remain easy to use by non-specialists.

Text mining or machine learning is a rapidly expanding field. Canonical applications are classification and clustering [34, 35, 37]. These applications are becoming common in industry, as well as defense and law enforcement. They are also increasingly used in the sciences - particularly bioscience - and social sciences, where researchers frequently have very large volumes of data. The humanities, however, are still only just beginning to explore the use of such tools. A new project, the Nora Project (www.noraproject.org) brings together multidisciplinary teams from five institutions and multiple domains, from the humanities to information science and computer science. We are collaborating to develop an architecture for non-specialists to employ text mining on some 5 GB of 18th and 19th century British and American literature. Besides the development of tools, the team hopes to discover what unique potential these tools might have for the literary scholar [5].

The work described in this paper is led by a team from the University of Maryland bringing expertise in literary research and user interfaces, and a team from the University of Illinois bringing text mining expertise.

2. SELECTED CASE STUDY
We selected a specific, realistic problem to guide our designs and engage scholars and students in our investigation. We used a corpus of about 300 XML-encoded letters comprising nearly all the correspondence between the poet Emily Dickinson and Susan Huntington (Gilbert) Dickinson, her sister-in-law. Because debates about what constitutes the erotic in Dickinson have been primary to study of her work for the last half century, we chose to explore patterns of erotic language in this collection. In the first step, our Dickinson expert classified by hand all the documents into two categories "hot" and "not hot." This was done in order to provide a baseline for the evaluation of the classification algorithms (see section 8). We then interviewed our expert to

JCDL'06, June 11–15, 2006, Chapel Hill, North Carolina, USA.

determine what her a-priori belief was about the features that could possibly be indicative of erotics in the documents. The expert believed that possible indicators of erotics included certain words, a rhetoric of similarity (e.g. "Each to Each,"), the presence of mutilation in the documents (e.g. erasures, scissorings or ink-overs). We then started designing and developing prototypes of the interface and of the data mining components, which were presented to the domain expert regularly for feedback. Today we have integrated all components in a web-based application and conducted a one-day evaluation of the system with a new literary research question.

3. RELATED WORK

While there are undoubtedly opportunities for all of the traditional text mining applications in large humanities repositories and digital library collections, their straightforward implementation is not our primary objective with Nora. As Jerome McGann and others have argued, computational methods, in order to make significant inroads into traditional humanities research, must concern themselves directly with matters of interpretation [19]. Our guiding assumption, therefore, has been that our work should be provocational in spirit—rather than vocational, or merely utilitarian—and that the intervention and engagement of a human subject expert is not just a necessary concession to the limits of machine learning but instead an integral part of the interpretative loop. In important respects we see this work as an extension of insights about modeling [21], deformation [19], aesthetic provocation [6], and failure [33]. It also comports with some of the earliest applications of data mining, such as when Don Swanson associated magnesium deficiency with migraine headaches, an insight provoked by patterns uncovered by data mining but only subsequently confirmed through a great deal more traditional medical testing [30].

Literary text mining is a new research area. Over the past decade, various text mining methods have been used to tackle literary research problems. Many well-studied literary text mining tasks can be modeled as classification problems, such as authorship attribution, stylistic analysis, and genre analysis. Literary text classification tasks are much broader than topic spotting. Scholars need to categorize texts by all kinds of labels, namely topics, styles, genres, authors, eras, and many other literary and historical concepts that combine these categories. Discriminant analysis [4], neural network [31], single-layer perceptron [23], logistic regression [2], cross entropy [15], and decision tree [25] methods have been explored as literary text classification approaches. These related works focus on one single classification method and require the involvement of computer science experts to drive the interface.

Visual user interfaces have been used to browse large collections of text, mostly by allowing users to explore the metadata of the collection [13, 27, 16]. Some visualization techniques have been used in conjunction with data mining [8, 28] but not with text mining. Text visualization has used colored bars to represent attributes as their value varies over the length of a document in a collection. Compus identifies structural elements of XML files by their color [10]. SeeSoft displays a variety of per-line attributes of source code, such as authorship, age, or execution frequency [7]. TileBars shows a visual preview the incidence of search terms within each search result document [11]. Liu uses color to represent the computationally-extracted emotional content of a document [17]. Other text visualizations focus on keywords as a means of reducing the data. One approach maps keywords to basis vectors, which are used to create a 3-D blob for each document; blobs with similar features should have similar content [26]. ThemeView (formerly ThemeScape) identifies topics within a corpus and produces a relief map; similar themes are adjacent and dominant themes are higher [36]. TextArc displays the entire text of one book arranged in an arc, with all the words occurring in the book placed inside. Frequently occurring words appear larger, and when a word is selected its occurrences in the text are highlighted [22].

4. USERS NEEDS

Informal observations lead us to believe that there are two main categories of potential users among literary scholars. A small group of enthusiastic computer users who are avid adopters of new technology, and a much broader base of scholars who tend, to varying degrees, to be uninterested in computational tools and methods, and unlikely to start using online analysis tools unless they offer simple, comprehensible user interfaces that have been shown to provide benefits to their peers. In general, our users consider provocation and argumentation a very valuable outcome of their work, in contrast with other domains that focus on decision-making. Based on interviews of members of the large and diverse Nora team and a set of general humanities research methods [32], we prepared a list of user needs we could address.

First we define some terms that had different connotations among our team: we call *documents* the units of text in a collection; e.g. a letter in the Emily Dickinson collection, but it could be a paragraph, a section or a chapter of a book in another collection. We call *document level metadata* the information such as date, author, or condition which is encoded in the XML tags, or calculated - e.g. amount of punctuation - but has no specific location in the text. In contrast, text-level metadata (e.g. line breaks, action verbs or metaphors) has location information and can be associated with a specific string of text.

Among the user needs, we mark with an asterisk those that have been addressed so far:

- Situate: what is available in Nora? What can I do?
- *Characterize: what are the characteristics of the documents in a collection (i.e. review the document level metadata). Can I restrict to a subset documents? or combine compatible collections?
- *Read: Reading, especially writerly reading, is an (some would say "*the*") essential activity for literary scholars.
- *Classify: What documents have a certain characteristic of interest (e.g. documents that are "hot", or that use metaphors, or are representative of sentimentality) and which ones do not.
- *Find indicators: Classification is not the most interesting aspect for a literary scholar. What is important is to discuss what makes a document fall in one category or another. So finding out what indicators or features seem to be characteristic of a particular class of documents might help understand the differences between classes. For example, what characterizes documents rated as hot? Indicators could be linked to document level metadata (is there some certain document time periods, or document conditions – e.g. cuts and scratches – that seem characteristic of hot documents? Is there some text level

information (presence of certain words, writing style, etc.) that is found more often in those documents?

- *Understand results of automatic analysis: one of the hopes of computer analysis is to automate the previous two tasks, but users will need to understand the results and eventually build trust in their validity.
- Interpret: annotate, compare, refer, illustrate, represent...
- Compare analysis with those done by other scholars and engage in discussion with colleagues
- Archive, publish results

We have started to address some of those needs and developed an exploratory prototype tool to allow users to perform automatic classification of documents based on a training set of documents classified manually, and to interactively explore the results (classification and indicators).

5. DESCRIPTION OF THE INTERFACE

To render the system description more lively, we will follow a usage scenario based on the study of the erotic in Emily Dickinson letters. The basic task of the user is to classify "hot" and "not-hot" documents, then try to understand what could characterize them.

5.1 Rating documents - Preparing training set

As users start, they are presented with the list of all the documents in the collection (Figure 1). Letters have no titles, so the first line(s) of the documents are used. For our study, a document would be rated as "erotic" that is flirtatious, has sexual connotations, is seductive, enticing and aims to pull the addressee in by imbuing the emotional and the intellectual with attention to the physical, even physical arousal. The text thus intends to be sexual or sexualized in some way(s). For simplicity, we use the short word "hot".

Clicking on a document title loads the document, which can be read and rated. To choose a rating, users click on one of the five colored checkboxes at the bottom of the screen. The range of the possible ratings goes from bright red for "hot", to black for "not hot". Once a document has been manually rated in such a way, a matching colored circle appears next to the title of the document. The ratings can be saved and retrieved at a later time.

5.2 Automatic classification

After manually rating a representative set of examples (e.g. 15 hot and 15 not-hot documents) it can be used as a training set by an automatic data mining classifier. Using a menu item, the classification request is submitted, and after about a minute the results of the classification are returned. Each document that had not been rated manually now has a colored square next to it representing the likelihood of being a hot document (Figure 2). Bright colors mean that the algorithm calculations resulted in a high probability for the document to be hot. Black squares signify that the document is likely to be non-hot. A numerical value indicates the probability ratio of being in one class versus the other. Users can see where the likely hot documents are by spotting bright colored squares while scrolling the list. For instance, in our example they see brighter squares for "I fit for them" (Figure 2) or "To take away our Sue" (Figure 3) and can click on the title to display and read the letter. They can acknowledge or reject the suggested classification by selecting the rating they think is appropriate. Users can sort the list of documents by the predicted hot-ness in order to first review the documents at the extreme ends of the scale. Like with all data mining techniques, the quality of the classification is highly dependent on the quality of the training set, therefore users are encouraged to re-run the prediction after having validated additional ratings. Similarly balanced training set are recommended. Dialog boxes can suggest a re-run after 8 to 10 new ratings have been added, and encourage a similar number of hot and not hot documents in the training set. The dialog boxes include explanations and a "don't show this again" check box for people who are already familiar with those recommendations..

5.3 Reviewing predicted word indicators

Anytime after the first run of the classification users can review the list of words suggested as possible indicators. The list returned by the text mining is shown on the left side of the screen (Figure 2 and 3). At the top on the list are the words suggested as most likely indicators of hot documents and at the bottom words that are most indicative of not-hot documents. Currently a fixed number of word indicators (i.e. 100) is shown to users. When a user clicks on a document all the words of the documents that are listed as indicators are highlighted in the text (in purple are the "hot" word indicators, and black the "not-hot" words.) To find documents that include a particular hot word, users can click on a word in the list and a red mark is shown next to title of documents that include the word.

This coordinated display of word indicators, document titles, predictions, ratings and full text allows users to easily switch between the tasks of reading, rating, reviewing suggested classifications and trying to understand what makes a document hot or not.

We provide simple layman explanations to the users when they encounter the result screen for the 1st time. A window pops up with a set of Frequently Asked Questions and their answers. The questions are 1) What are those words? 2) What are those numbers? 3) What do the purple squares mean? and 4) What do the results seem wrong sometimes? A check box allows users to specify that they do not need to see this information when new predictions results are displayed (but it can still accessed via a FAQ menu item). An animated video demonstration of the interface is also provided.

5.4 Algorithms used

The automatic classification is performed using a multinomial naive Bayes (NB) algorithm [20] executed by a data mining tool called D2K (see the Architecture section of the paper). Given a set of training examples with binary labels (e.g. "hot" or "not hot"), the algorithm outputs 1) the learned classifier with prediction results sorted by the ratio of the class posterior probabilities; 2) the feature set ranked by the ratio of the class conditional probabilities. Currently the classifier deals with binary classification: "hot" and "not-hot". In the future we will extend the classifier to handle multiple classes. More classification algorithms, including regression methods, will also be explored to deal with scaled ratings. Therefore, we decided to allow the manual ratings to remain on a 1 to 5 scale to allow the more subtle user ratings to be recorded but those ratings are currently automatically grouped into 2 classes when used as part of the

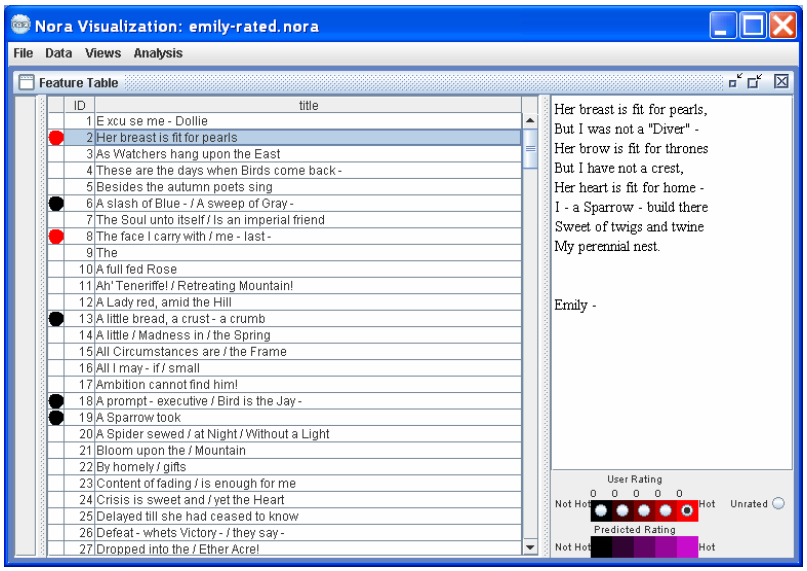


Figure 1: Users select a document in the collection (here "Her breast is fit for pearls") and read. They can then rank the document on a 5-point scale ranging from not-hot (black) to hot (red), which helps build the training set.

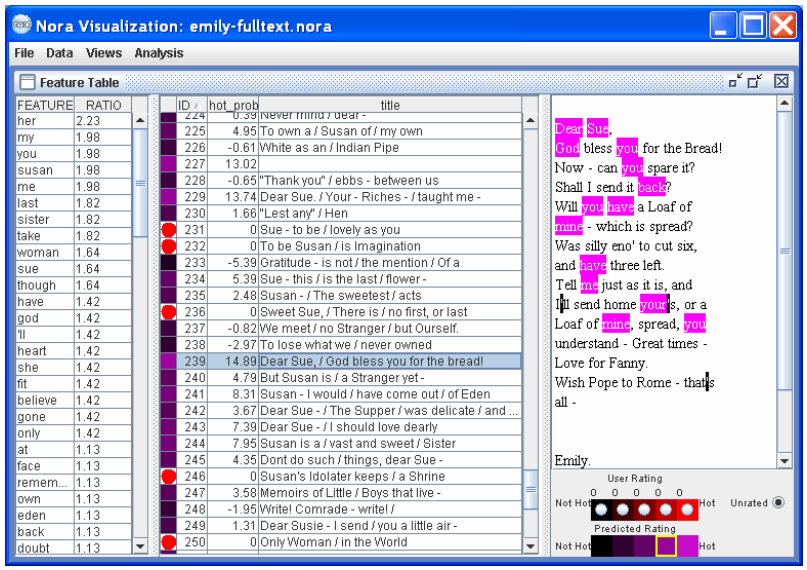


Figure 2: After requesting that the remaining documents be automatically classified, purple color squares are placed next to each document. Again bright colors mean "hot" and black means "not hot". On the most left pane, a list of words is provided, with the words found to be more representative of the hot documents of the training set listed at the top.

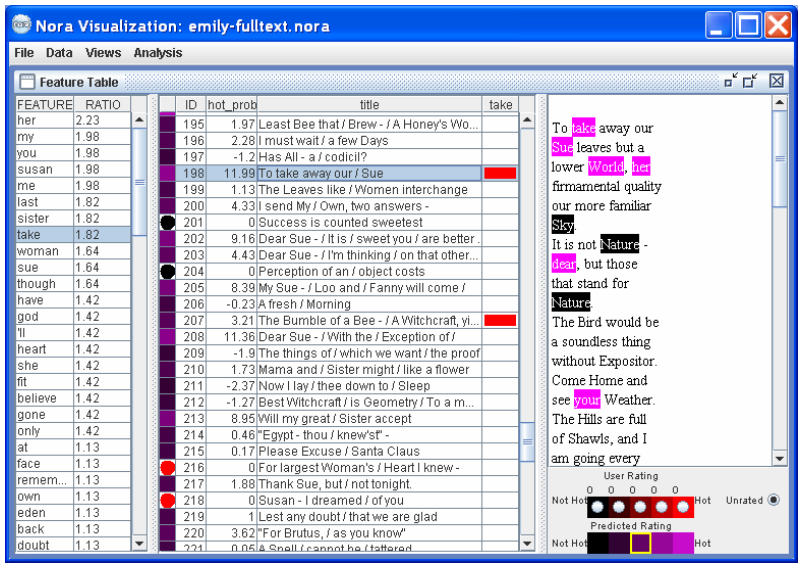


Figure 3: Users can select a word in the list to see what documents contain that word. Here "take" was selected, showing that "To take away our Sue" is one of the documents that contains the word "take". Selecting the document reveals the location of all the "hot" and "not hot" words in the document itself.

training set (the top 2 values for "hot" and the two low values for "not-hot").

To visually indicate the results of the classification algorithm we chose to use double encoding by color and shape. We use different shapes for manual ratings (circles) and predictions (squares) which makes clear what the origin of the rating is, and shows how much progress has been made in the classification task. Simply using shapes and brightness would be sufficient to encode all the information but double encoding the origin of the rating by color (red for manual ratings, purple for predictions) reinforces the effect while guaranteeing that users with color blindness can still use the interface. We chose to use black for the non-hot extreme of the scale for both manual ratings and predictions with the assumption that users interest was primarily in the "hot" side and that having two separate colors for "non-hot" would add unneeded complexity.

6. Finding correlations

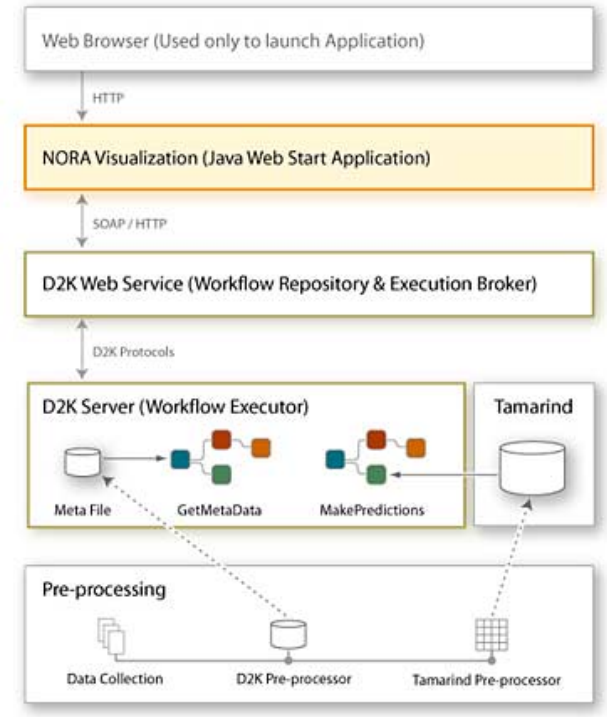
When the document classification is adequate (i.e. either when all ratings were confirmed manually, or when users have gained confidence in the predicted ratings) users can explore the correlations between the ratings and the set of available metadata at a document level. Our metadata is extracted from the XML tags of the original documents (such as date, geographical origin, destination, addressee, condition of the letter) or pre-calculated (e.g. rate of use of punctuation, amount of repetitions, etc.) Many other potential indicators could be proposed and preprocessing algorithms made available. At this stage we only have a small number of them but the interface architecture would allow any number of new attributes to be added.

We allow users to explore possible correlations by displaying customizable scatterplots (Figure 4) and a dynamic query style of interface (Ahlberg, 1994). For example, our Dickinson expert had wondered whether the use of dashes had something to do with the general hotness of the document. A special attribute representing the percentage of lines ending with dashes was computed separately and added as an attribute for each document. A scatterplot was then created using this attribute (Figure 5). The document with the largest relative number of dashes was indeed marked as hot, but overall there didn't seem to be a correlation between the use of dashes and hotness. On the other hand, the display suggests that the use of dashes was more prevalent during a certain time period, as we can see a cluster of larger dots. Users can explore various combinations of attributes to find characteristics of the collection and explore possible correlations.

Even though scatterplot displays are common visual representations of data and the reading of scatterplots is taught in elementary school, it is also known that many people find them difficult to specify and interpret [24]. Our prototype of the scatterplot view was rapidly built using the InfoVis Toolkit [9], and in fact it was built first, before the table view (Figure 1-3). Nevertheless, because it was more difficult to use - especially in an unpolished state - we then focused our effort on developing a much simpler table view, and its evaluation. Our next step will be to improve and simplify the scatterplot view to make it more usable. One possible direction is to use a multi-layer approach [29] where users first start at level 1 with the simplest scatterplot interface with only a few controls and progressively move to higher levels with more controls.

7. SYSTEM ARCHITECTURE

A key aspect of our work has been to test the feasibility of this fairly complex distributed process of building a prototype application. This prototype includes work on 3 major components developed at 3 different institutions: Nora Visual Interface, D2K and T2K, and Tamarind.



The Nora Visual Interface provides the front-end graphical user interface to users. It is a Java Web Start application, which can be launched from the Web browser. The Nora Visual Interface was developed at the University of Maryland. It uses the InfoVis Toolkit [9], provides access to different views of the document collection, and controls the execution of the D2K and T2K components.

D2K (Data to Knowledge: http://alg.ncsa.uiuc.edu/do/tools/d2k) is a rapid, flexible data mining and machine learning system that integrates analytical data mining methods for prediction, discovery, and deviation detection, with data and information visualization tools. It offers a visual programming environment that allows users to connect programming components together to build data mining applications and supplies a core set of modules, application templates, and a standard API for software component development. For Nora, the users of D2K are the developers, but in the future this infrastructure will allow expert end users to modify the D2K modules themselves). The T2K (Text to Knowledge) components provides text mining and analysis capabilities that have been specially designed to operate in and capitalize upon the complexity of rich natural language domains of very large stores of text and multimedia documents.

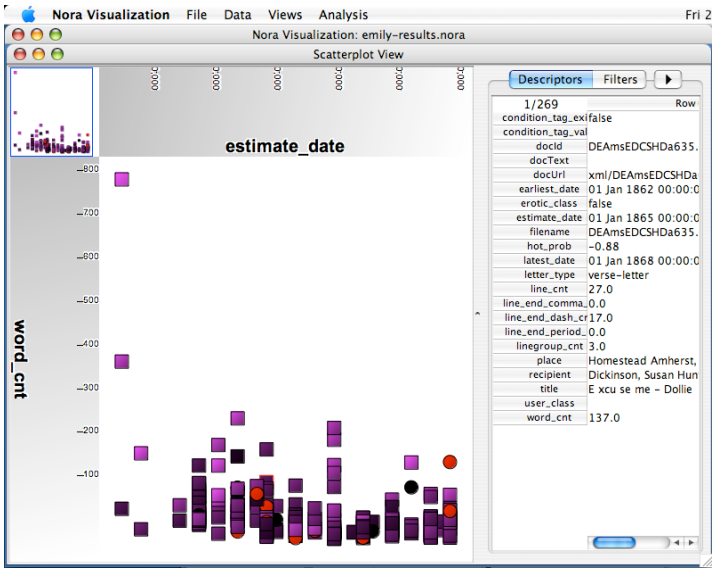


Figure 4: The scatterplot view of the collection (each shown running on a MacIntosh). Each dot is a document. Time (i.e. the median of the estimated date range) is mapped on the X axis, and the length of the document is mapped on the Y axis. Color represents hotness. Here deeper reds indicate hot documents while other documents are paler We can see that the longer documents were written earlier on, The display also suggests that there is no correlation between time and hotness, and no particular time periods where significantly more hot documents were written. Zooming is possible to inspect particular clusters.

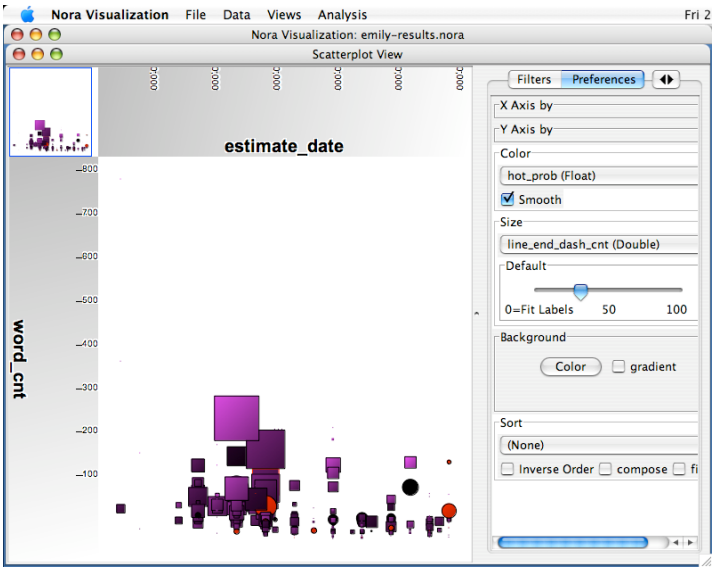


Figure 5: The control panel on the right side allows users to map document attributes to visual attributes. In this scatterplot the size of the dot was set to be proportional to the ratio of dashes in the document (i.e. documents with lots of "-" appear bigger). The document that has the largest relative number of dashes is indeed predicted as hot, but overall there doesn't seem to be a correlation between the use of dashes and hotness.

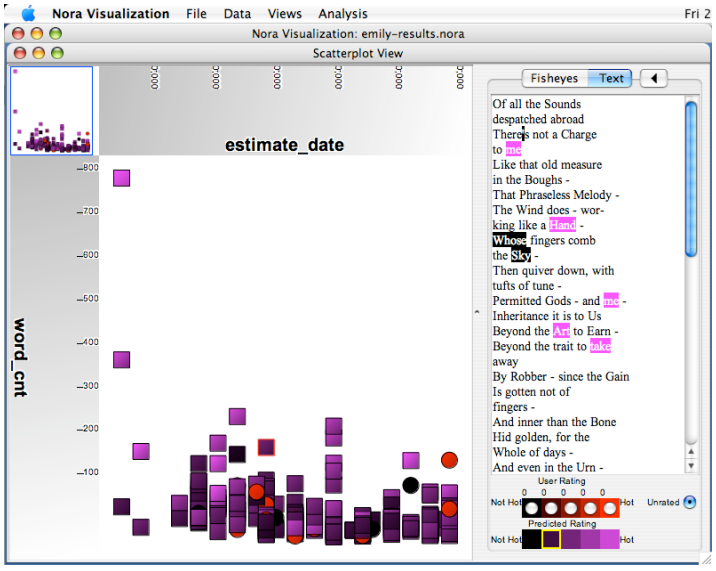


Figure 6: The scatterplot view can also be used as an alternate support for classification tasks.

D2K leverages the D2K Web Service implementation to execute pre-constructed T2K workflows. The D2K Web Service (WS) provides a WS-I Basic Profile 1.1 compliant programming interface for executing D2K Workflows. D2K workflows are XML files that define data mining applications composed of D2K and T2K components (Java classes) that have been connected together to form a directed graph. The D2K Server, like other D2K-driven applications uses the D2K Infrastructure as the itinerary execution engine. Each D2K Web Service endpoint contains a library of registered workflows available for execution by clients. For each workflow, a pool of resources required for execution (Java classes, property files, etc.) is also stored. Associated with each workflow definition is a list of D2K Servers that are eligible to process it. When service clients submit job requests, the D2K Web Service automatically handles the brokering of execution to an appropriate, and available, D2K Server. In return, the D2K Server requests from the D2K Web Service the resources it will need to process the job. While a job is processing, the D2K Web Service monitors its progress and persists any results that are produced. All communications between the D2K Web Service and D2K Servers occur over transmission control protocol (TCP) socket connections using D2K specific protocols. One D2K workflow using T2K components processed the document collection and creates the meta-data. Two other workflows have been provided as web services for the Nora Visual Interface. The first, GetMetaData workflow returns the pre-computed meta-data to the visualization. The second, MakePredictions builds the classification model by using the hand tagged documents and word counts from the Tamarind datastore, returns predictions for the entire document collection by applying the computed model, and returns calculated metrics. The D2K web service endpoints are hosted at the National Center for Supercomputing Applications at the University of Illinois.

Tamarind, developed by the University of Georgia, is a large-scale XML pre-processor for scholarly text analysis. It creates a database that contains every discrete token, the type classification for each token, its character length, part of speech, main orthographic features, the filename in which it appears, and an exact, unique, XPath expression indicating its location in the document. Tamarind uses Postgres as its datastore environment. Tamarind was used to preprocess and provide the datastore for the Emily Dickinson collection.

Once the Nora Visual Interface is launched and running on the user's computer, a D2K web service (GetMetaData) is engaged to download the stored Meta File for the document collection. Once the hand tagging of documents has been done, the automatic classification is performed using the D2K web service (MakePredictions).

The current prototype is somewhat customized for the Emily Dickinson collection, but our design is general and our next step will be to investigate other collections.

8. ALGORITHM VALIDATION

For the Dickinson eroticism problem, classification and feature-category correlation analysis are two closely related tasks. As Craig pointed out [4], high classification accuracy confirms the validity of correlation analysis, and the correlation description helps to understand the mechanism underlying a successful classification.

Naive Bayes is a widely used text classification algorithm [38, 14]. We picked three Reuters-21578 categories ("acq", "grain", and "trade") to test our naive Bayes implementation (Table 1). The results are similar to the results reported in [14].

Table 1: Binary classification results

	F1	Breakeven
Acq	0.93	0.92
Grain	0.70	0.74
Trade	0.45	0.59

To evaluate the naive Bayes classification results in literary classification we used the collection of 269 Dickinson's letters rated by hand by our expert. 102 letters were rated as erotic, and 167 as not. Each letter is represented as a vector of word features. The total number of features is 4058. Add-one smoothing is used to avoid zero probabilities.

A 10-fold cross validation showed an overall accuracy of 65% (Table 2), slightly better than the standard majority-class baseline (62%), which outputs the most common class (here "non-erotic") for all documents. But scholars are much more interested in the erotic poems than the non-erotic ones so the majority class baseline does not help identify erotic poems at all, so looking at the naive Bayes algorithm scores of 0.51 precision and 0.55 recall (F1 score 0.53) for the erotic category is more useful. Given those encouraging classification results, it seems worth exploring further the feature-category correlations also provided by the naive Bayes algorithm.

Table 2: Results of 10-fold cross validation

	Accuracy	Precision	Recall	F1
10-fold	0.65	0.51	0.55	0.53
Leave-one-out	0.64	0.53	0.57	0.55

9. USER EVALUATION

The prototype has been used by some of our project partners, in particular our Dickinson expert. We first include a long quote, then report on one-day case study during which we used a different research question about spirituality, and finally summarize users' suggestions for improvements.

9.1 Example of use

The following quotation was written by an English Department professor - our Dickinson expert - to her colleagues, after reviewing the results of one of our early text mining experiment, and before the user interface had been connected to the data mining system. It shows how this literary scholar could use the results of the data mining to shed new light on the texts despite the fact that she had already studied the texts extensively:

"The textual critic Harold Love has observed of 'undiscovered public knowledge' (consciously employing the aforementioned Don Swanson's phrase) that too often knowledge, or its elements, lies (all puns intended) 'like scattered pieces of a puzzle but remains unknown because its logically related parts are diffused, relationships and correlations suppressed' [18]. The word 'mine' as a new indicator identified by D2K is exemplary in this regard. Besides possessiveness, 'mine' connotes delving deep, plumbing, penetrating--all things we

associate with the erotic at one point or another. So 'mine' should have already been identified as a 'likely hot' word, but has not been, oddly enough, in the extensive critical literature on Dickinson's desires. 'Vinnie' (Dickinson's sister Lavinia) was also labeled by the data mining classifier as one of the top five 'hot' words. At first, this word appeared to be a mistake, a choice based on proximity to words that are actually erotic. Many of Dickinson's effusive expressions to Susan were penned in her early years (written when a twenty-something) when her letters were long, clearly prose, and full of the daily details of life in the Dickinson household. While extensive writing has been done on the blending of the erotic with the domestic, of the familial with the erotic, and so forth, the determination that 'Vinnie' in and of itself was just as erotic as words like 'mine' or 'write' was illuminating. The result was a reminder of *how* or *why* some words are considered erotic: by their relationship to other words. While a scholar may un-self-consciously divide epistolary subjects within the same letter, sometimes within a sentence or two of one another, into completely separate categories, the data mining classifier will not. Remembering Dickinson's 'A pen has so many inflections and a voice but one,' "the data mining has made us, in the words of our subject expert, "plumb much more deeply into little four- and five-letter words, the function of which I thought I was already sure, and has also enabled me to expand and deepen some critical connections I've been making for the last 20 years."

9.2 Case study on Spirituality

A day was set aside to work on a completely new question with the same expert: "Was Dickinson a Christian believer?" The question was chosen because many scholars have debated this question (but not our expert so it was a new problem, and one that she only has a mild interest in). We first used a "think aloud" protocol during a 2.5 hour session. The user worked alone but the observer asked a few questions. After a pause, the user continued working, alone, for another few hours.

The user started by searching for a document she knew. She rated 5 documents, then started to read documents without rating them, because could not decide how to rate them, then decided to change the question. She had been rating documents suggesting that Dickinson "seemed to believe" as red, and "seemed to doubt" as black. The new question became: "How much was spirituality, afterlife etc. on her mind? (i.e. all things we associate with religion)? Was she as interested in questions of belief as many of her readers believe her to have been?". She corrected the rating to reflect the new question: "Shows spirituality" as red, "Does NOT show spirituality" as black. Titles (which consist of the first one or two lines of the letter) were useful to choose documents to review.

After rating 17 documents in each category, a first prediction was run and the top ranked suggestions reviewed. Out of the 10 top ranked many seemed to be on the wrong side, but this led to some insight nevertheless. In this first run, the word "pray" came as a predictor for "non spirituality" but prompted the user to reflect that Dickinson used religious metaphor without really meaning it as religious but in order to draw on metaphors familiar to her audience. Some predictor words came out not surprising to her (e.g. 'Father and Son'). Others were useless ('then,' 'often,' 'go') and prompted us to reflect that some words should be eliminated automatically, and others may require the user to tell the system to ignore them. Puzzling but "interesting," was the word "little" which prompted some tentative hypothesis that human or self-deprecation might often accompany Dickinson's employment of religious metaphor. The user commented that this will have to be more thoroughly explored in order for any hard and fast conclusions to be drawn.

For some documents the predictor made her think twice, and change her mind. "I had thought of rating (this document) as having to do with religion because it talked about immortality, but the predictor said that was a low probability so on a second reading I realized that 'immortality' was used to connote 'fame,' an afterlife in others' memory rather than an actual afterlife of an individual.

After 4.5 hours, 75 documents had been rated, and the prediction had been run 4 times. Each document had been read 2-5 times, and about 10 minutes was spent on each (Note that the documents were short and familiar to the user; similar new documents would probably take 30 minutes per document). Because few word indicators were useful in this exercise only 5 of them were checked carefully to see where they appeared in the collection (but about 75 documents were read to accomplish this task). By the end of the exercise the user reflected that the prediction seemed to be more accurate: "The predictor and I always seem to be within a couple of scores. There are some cases, such as #229, where we are at opposite ends of the spectrum, but I'm surprised at how close we tend to be."

The final insight gathered in this short exercise was that the system was identifying as religious documents that have a lot of religious, idolatrous language in them about her friend Susan. Susan serves as a sort of religion for Emily, and Dickinson uses religious metaphors to express desire and love.

One observation was that "the software is marking things as religious that I am also marking, but for reasons that seem totally unrelated to the words it is highlighting." One possible explanation is that only the top 25 words at each end of the spectrum were reported in the interface at that time, and because many words were too general and not useful they may have hidden more useful words. Showing more words and allowing users to remove useless words from the list may be helpful. Using a continuous color scale would also better represent the importance of each word highlighted in the document.

9.3 Other Feedback and Suggestions

Many of the suggestions received from Nora team members were related to general usability problems (e.g. better labeling, better use of color) or to users needs we know about (see section 3) but that we had chosen not to address yet (e.g. extending to other collections, comparing with work of another writer). We list here specific suggestions that might benefit developers of other similar applications for other domains,

- Show counts of classified documents. This is useful to show progress and create balanced training sets which data-mining algorithms prefer.

- Allow users to assign extra weights for certain words (or ignoring some words).

- Add standard string-search to allow users to search for documents that used particular words the user believe may

become good predictors, so that they can classify those documents.

- Show overview of where the indicator words appear in the list of documents without requiring scrolling. This could be done with Value Bars [3] which encoded information about location in long documents using color bars displayed on the scrollbar itself.

Users who had more knowledge or interest in the evaluation of the classifier itself asked to be able to:

- Keep the history, and allow users to compare predictions results between runs (to see what changed from one run to the next.)

- Show classification prediction results on the already rated document to reflect problems with the classifier and see when the classifier disagrees with the manual ratings.

The entire system is now functional is available on the web from the Nora website. We will refine our interface and then seek feedback from other literary scholars. Future directions include extracting other possible indicators beside single words. Ultimately, success for this project will be measured by how successful literary scholars are at publishing papers in literary journals that report on new discoveries made using our system.

10. CONCLUSIONS

Marti Hearst makes a sobering remark in [12]: "The fundamental limitations of text mining are first, that we will not be able to write programs that fully interpret text for a very long time, and second, that the information one needs is often not recorded in textual form." Fortunately for the literary scholars, their work IS text. On the other hand the odd characteristics of many documents of interest (e.g. medieval English, poetry or Emily Dickinson's prose) makes them less likely to benefit from recent language analysis techniques developed for more active domains such as intelligence analysis or biology. Nevertheless our early experience suggests that literary scholars will find results of simple data mining tools provocative and inspiring, and that they will be able to generate new insights in the process. A blogger confirms this point by saying "I do agree that Nora could act as an aid to come up with new interpretive ideas."

Our early evaluation also suggests that our basic user interface and web architecture is usable by non-specialists. This is in departure from the majority of text mining tools which require non specialists to be assisted by computer experts to accomplish their task. Finally, we believe that this user interface design can be easily applied to other document classification applications that require experts to define training sets and carefully review and edit results.

11. ACKNOWLEDGMENTS

We thank all the members of the Nora team, in particular John Unsworth who leads this project, Steve Ramsay who provides the Tamarind system, and David Clutter, Greg Pape, and Andrew Shirk from NCSA who helped setup the web services for Nora. We are also grateful to Jean-Daniel Fekete who helped us with the InfoVis Toolkit. Partial support for this work was provided by the Andrew Mellon Foundation and the University of Maryland Libraries.

12. REFERENCES

[1] Ahlberg, C. and Shneiderman, B.,Visual information seeking: Tight coupling of dynamic query filters with starfield displays, *Proc. CHI '94 Conference: Human Factors in Computing Systems*, ACM, New York (1994), 313–321 and color plates.

[2] Can, F. and Patton, J.M., Change of Writing Style with Time. *Computers and the Humanities* 38(1), (2004) 61-82

[3] R. Chimera. Value Bars: An information visualization and navigation tool for multi-attribute listings. *Proc. of the ACM Conference on Human Factors in Computing Systems, CHI'92*, (1992) 293-294

[4] Craig, H., Authorial attribution and computational stylistics: if you can tell authors apart, have you learned anything about them? *Literary and Linguistic Computing* 14(1), (1999) 103-113.

[5] Downie, J.S., Unsworth, J., Yu, B., Tcheng, D., Rockwell, G., Ramsay, S., A Revolutionary Approach to Humanities Computing? Tools Development and the D2K Data-Mining Framework. Panel at the *Joint International Conference of the Association for Computers and the Humanities and the Association for Literary and Linguistic Computing*, University of Victoria (2005).

[6] Drucker, J. and Nowviskie, B., Speculative Computing: Aesthetic Provocations in Humanities Computing. In S. Shcreibman, R. Siemens, and J. Unsworth (eds.), *The Blackwell Companion to Digital Humanities*, Oxford: Blackwell Publishing Ltd., (2004) 431-447.

[7] Eick, S.G., Steffen, J.L., Sumner, Jr., E.E. SeeSoft: a tool for visualizing line-oriented software statistics. *IEEE Transactions on Software Engineering*, 18, 11 (1992) 957-968.

[8] Fayyad, U., Grinstein, G, and Wierse A., *Information Visualization in Data Mining and Knowledge Discovery*, Morgan Kaufmann Publ., San Francisco, CA (2001)

[9] Fekete, J-D., The Infovis Toolkit. *Proceedings of the 10th IEEE Symposium on Information Visualization*, IEEE Press (2004) 167-174

[10] Fekete, J-D., Dufournaud, N., Compus: visualization and analysis of structured documents for understanding social life in the 16th century. *Proceedings of DL'00: the fifth ACM Conference on Digital Libraries* (2000)

[11] Hearst, M., TileBars: Visualization of Term Distribution Information in Full Text Information Access. *Proceedings of the Conference on Human Factors in Computing Systems, CHI 1995,* (1995) 59-66

[12] Hearst, M., Untangling Text Data Mining. *Proceedings of ACL'99: the 37th Annual Meeting of the Association for Computational Linguistics* (1999).

[13] Heath, L., Hix, D., Nowell, L., Wake, W., Averboch, G., Labow, E, Guyer, S., Brueni, D., France, R., Dalal, K., Fox, E., Envision: a user-centered database of computer science literature, *Communications of the ACM*, v.38 n.4, (1995) 52-53

[14] Joachims, T., Text categorization with Support Vector Machines: learning with many features. *Proceedings of the*

10th European Conference on Machine Learning, Springer, (1998) 137-142.

[15] Juola, P., Baayen, H. and Harald, R., A Controlled-Corpus Experiment in Authorship Identification by Cross-Entropy. *Literary and Linguistic Computing*, vol. 20, iss. Supplement 1, (2005) 59-67

[16] Lee, B., Czerwinski, M., Robertson, G.G and Bederson, B.B.,Understanding Research Trends in Conferences Using PaperLens. *Extended Abstracts of the SIGCHI Conference on Human Factors in Computing Systems,* (2005) 1969-1972

[17] Liu, H., Selker, T., Lieberman, H., Visualizing the Affective Structure of a Text Document. *Adjunct Proceedings of the Conference on Human Factors in Computing Systems, CHI 2003* (2003) 740-741

[18] Love, H., *Scribal Publication in Seventeenth-Century England.* Oxford: Clarendon Press (1993)

[19] McGann, J.. *Radiant Textuality: Literature After the World Wide Web*. New York: Palgrave (2001)

[20] McCallum, A., Nigam, K., A Comparison of Event Models for Naive Bayes Text Classification, *AAAI-98 Workshop on "Learning for Text Categorization".* AAAI Press (1998)

[21] McCarty, W., Modeling: A Study in Words and Meanings. In S. Shcreibman, R. Siemens, and J. Unsworth (eds.) *The Blackwell Companion to Digital Humanities.* Oxford: Blackwell Publishing (2004) 254-270)

[22] Paley, W.B., TextArc: Showing Word Frequency and Distribution in Text. Poster presented at *IEEE Symposium on Information Visualization 2002,* (2002)

[23] Pasquale, J.D. and Meunier, J. G., Categorisation Techniques in Computer-Assisted Reading and Analysis of Texts ({CARAT}) in the Humanities. *Computers and the Humanities* 37(1), (2003) 111-118.

[24] Plaisant, C., Kang, H., and Shneiderman, B., Helping users get started with visual interfaces: multi-layered interfaces, integrated initial guidance and video demonstrations, *Proceedings of Human–Computer Interaction International 2003*, (2003) 790-794.

[25] Ramsay, S., In Praise of Pattern. In *"The Face of Text" - 3rd Conference of the Canadian Symposium on Text Analysis (CaSTA)*. (2004)

[26] Rohner, R., Sibert, J., Ebert, D., The Shape of Shakespeare: Visualizing Text using Implicit Surfaces. *Proceedings of the 1998 IEEE Symposium on Information Visualization.* (1998) 121-129

[27] Shneiderman, B., Feldman, D., Rose, A., and Grau, X.F., Visualizing digital library search results with categorical and hierarchical axes, *Proceedings of the fifth ACM conf. on Digital libraries*, ACM Press, (2000) 57-66.

[28] Shneiderman, B., Inventing Discovery Tools: Combining Information Visualization with Data Mining *Information Visualization* 1, 1, (2002) 5-12

[29] Shneiderman, B., Promoting Universal Usability with Multi-Layer Interface Design, *ACM Conference on Universal Usability,* ACM Press, (2003) 1-8

[30] Swanson and Smalheiser, Assessing a gap in the biomedical literature: Magnesium deficiency and neurologic disease. *Neuroscience Research Communications*, 15 (1994) 1-9.

[31] Tweedie, F., Singh, S. and Holmes, D., Neural Network Applications in Stylometry, *The Federalist Papers* 30(1), (1996) 1-10

[32] Unsworth, J., The Importance of Failure. *The Journal of Electronic Publishing* 3.2 (1997) http://www.press.umich.edu/jep/03-02/unsworth.html.

[33] Unsworth, J., Scholarly Primitives: What methods do humanities researchers have in common, and how might our tools reflect this? Presented at the *Humanities Computing: formal methods, experimental practice symposium*, King's College, London (2000) http://www3.isrl.uiuc.edu/~unsworth/Kings.5-00/primitives.html

[34] Weiss, S., et al., *Text Mining: Predictive Methods for Analyzing Unstructured Information*. New York: Springer (2005).

[35] Widdows, D., Geometry and Meaning. Stanford: CLSI Publications (2004)

[36] Wise, J. A., Thomas, J., J., Pennock, K., Lantrip, D., Pottier, M., Schur, A., and Crow, V., Visualizing the non-visual: Spatial analysis and interaction with information from text documents, *Proc. IEEE Information Visualization '95,* IEEE Press (1995), 51–58.

[37] Witten, I. and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. San Diego: Academic Press (2000).

[38] Yang, Y. and Liu, X., A re-evaluation of text categorization methods. *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, (1999) 42-49.