Discovering interesting usage patterns in text collections: Integrating text mining with visualization

Anthony Don[1], Elena Zheleva[2], Machon Gregory[2], Sureyya Tarkan[2], Loretta Auvil[4], Tanya Clement[3], Ben Shneiderman[1, 2] and Catherine Plaisant[1]

[1]Human Computer Interaction Lab
[2]Computer Science Department
[3]English Department
University of Maryland, USA

[4]National Center for Supercomputing Applications, University of Illinois, USA

{don,elena,mbg,sureyya,ben,plaisant} @cs.umd.edu, lauvil@ncsa.uiuc.edu, tclement@wam.umd.edu

ABSTRACT

This paper addresses the problem of making text mining results more comprehensible to humanities scholars, journalists, intelligence analysts, and other researchers, in order to support the analysis of text collections. Our system, FeatureLens, visualizes a text collection at several levels of granularity and enables users to explore interesting text patterns. The current implementation focuses on frequent itemsets of n-grams, as they capture the repetition of exact or similar expressions in the collection. Users can find meaningful co-occurrences of text patterns by visualizing them within and across documents in the collection. This also permits users to identify the temporal evolution of usage such as increasing, decreasing or sudden appearance of text patterns. The interface could be used to explore other text features as well. Initial studies suggest that FeatureLens helped a literary scholar and 8 users generate new hypotheses and interesting insights using 2 text collections.

Keywords: D.2.14.a User interfaces, H.2.8.h Interactive data exploration and discovery, H.2.8.l Text mining

1. INTRODUCTION

Critical interpretation of literary works is difficult. With the development of digital libraries, researchers can easily search and retrieve large bodies of texts, images and multimedia materials online for their research. Those archives provide the raw material but researchers still need to rely on their notes, files and their own memories to find "interesting" facts that will support or contradict existing hypotheses. In the fields of the Humanities, computers are essentially used to access to text documents but rarely to support their interpretation and the development of new hypotheses.

Some recent works [4, 11] addressed this problem. One approach, supports the analysis of large bodies of texts by interaction techniques together with a meaningful visualization of the text annotations. For example Compus [4] supports the process of finding patterns and exceptions in a corpus of historical document by visualizing the XML tag annotations. The system supports filtering with dynamic queries on the attributes and analysis using XSLT transformations of the documents. Another approach is to use data-mining or machine learning algorithms integrated with visual interfaces so that non-specialists can derive benefit from these algorithms. This approach has been successfully applied in the literature domain in one of our prior project [11]. Literary scholars could use a Naive Bayesian classifier to determine which letters of Emily Dickinson's correspondence contained erotic content. It gave users some insights into the vocabulary used in the letters.

While the ability to search for keywords or phrases in a collection is now widespread such search only marginally supports discovery because the user has to decide on the words to look for. On the other hand, text mining results can suggest "interesting" patterns to look at, and the user can then accept or reject these patterns as interesting. Unfortunately text mining algorithms typically return large number of patterns which are difficult to interpret out of context. This paper describes FeatureLens, a system designed to fill a gap by allowing users to interpret the results of the text mining thru visual exploration of the patterns in the text. Interactivity facilitates the sorting out of unimportant information and speeds up the task of analysis of large body of text which would otherwise be overwhelming or even impossible [13].

FeatureLens[1] aims at integrating a set of text mining and visualization functionalities into a powerful tool, which provokes new insights and discoveries. It supports discovery by combining the following tasks: getting an overview of the whole text collection, sorting frequent patterns by frequency or length, searching for multi-word patterns with gaps, comparing and contrasting the characteristics of different text patterns, showing patterns in the context of the text where they appear, seeing their distributions in different levels of granularity, i.e. across collections or documents. Available text mining tools show the repetitions of single words within a text, but they miss the support for one or more of the aforementioned tasks, which limits their usefulness and efficiency.

We start by describing the literary analysis problem that motivated our work and review the related work. We then describe the interface, the text mining algorithms, and the overall system architecture. Finally we present several examples of use with 3 collections and discuss the results of our pilot user studies.

2. MOTIVATION

This work started with a literary problem brought by a doctoral student from the English department at the University of

[1] A video and an online demonstration are available from http://www.cs.umd.edu/hcil/textvis/featurelens/

Maryland. Her work deals with the study of *The Making of Americans* by Gertrude Stein. The book consists of 517,207 words, but only 5,329 unique words. In comparison, *Moby Dick* consists of only 220,254 words but 14,512 of those words are unique. The author's extensive use of repetitions (Figure 1) makes *The Making of Americans* one of the most difficult books to read and interpret. Literature scholars are developing hypotheses on the purpose of these repetitions and their interpretation.

> Everyone then sometime is a whole one to me, everyone then sometimes is a whole one in me, some of these do not for long times make a whole one to me inside me. Some of them are a whole one in me and then they go to pieces again inside me, repeating comes out of them as pieces to me, pieces of a whole one that only sometimes is a whole one in me.
>
> Paragraph 1225, *The Making of Americans*

Figure 1: Extract from The Making of Americans.

Recent critics have attempted to aid interpretation by charting the correspondence between structures of repetition and the novel's discussion of identity and representation. Yet, the use of repetition in *The Making of Americans* is far more complicated than manual practices or traditional word-analysis could indicate. The text's large size (almost 900 pages and 3183 paragraphs), its particular philosophical trajectory, and its complex patterns of repetition make it a useful case study for analyzing the interplay between the development of text mining tools and the way scholars develop their hypotheses in interpreting literary texts in general.

This collaboration between computer scientists and humanity scholars is part of the MONK project (www.monkproject.org), which brings together multidisciplinary teams from six institutions. Because this case study used a very unusual text we also tested FeatureLens with other collections: a technical book, a collection of research abstracts, and a collection of presidential addresses which we use here to describe the interface and also used in our pilot user study.

3. RELATED WORK

Visualizations have been applied successfully to retrieving, comparing, and ranking whole text documents [14, 16] and computer programs [3, 7]. Instead of ranking documents according to their content, FeatureLens ranks text patterns according to their length and frequency, and it provides a visualization of the text collection at the document level and at the paragraph level. These two levels of granularity allow the user to identify meaningful trends in the usage of text patterns across the collection. It also enables the analysis of the different contexts in which the patterns occur.

A recent interactive NY Times display [8] shows the natural representation of the text of the *State of the Union Addresses* with line, paragraph, and year categorization. It displays word frequency, location, and distribution information in a very simple manner which seemed to be readily understandable by the literary scholars we have been interacting with. It allows search but does not suggest words or patterns that might be interesting to explore. It also does not support Boolean queries.

Visualizing patterns in text is also related to visualizing repetitions in sequences. A number of techniques such as arc diagrams, repeat graphs and dot plots have been developed and applied to biological sequence analysis [2, 5, 6]. Compared to DNA, literary text has different structural and semantic properties such as division into documents, paragraphs, sentences, and parts of speech that one could use to create a more meaningful visualization. Arc diagrams have been used to visualize musical works and text, and have advantages over dot plots [15], though it has not been shown how they can be adapted to large collections of text without creating clutter. TextArc [9] is a related project, which visualizes text by placing it sequentially in an arc and allowing a user to select words interactively and to see where in the text they appear. It does not support ranking of patterns and selecting longer sequences of words. Most of the tools describe above only handle small datasets and display the collection as a fixed level of granularity.

4. FEATURELENS

Figure 2 shows the graphical user interface of FeatureLens. The *State of the Union Addresses* collection consists of eight documents, one for each of President Bush's eight annual speeches (there were two in 2001 because of 9/11). The documents are represented in the *Document Overview* panel. Each rectangular area represents one speech and its header contains the title of the document, i.e. the year of the speech in this case. Within the rectangular representation of the document, each colored line represents a paragraph in this collection. When the document is very large each line may represent a unit of text longer than a paragraph so that the overview remains compact. FeatureLens computes the default unit of text to be such that the overview fits on the screen, and users can change that value using a control panel. For simplicity we call that arbitrary small unit of text a paragraph in the rest of the paper.

The *Frequent Patterns* panel, located on the left of the screen, displays the pre-computed text patterns generated by the data mining algorithms. Currently we combine only 2 types of patterns: frequent words, and frequent itemsets of n-grams (which capture the repetition of exact or similar expressions in the collection - more details in section 5). Frequent words naturally occur at the top of the pattern list since the default ordering of the list is by frequency. This also makes it easier for users to learn the interface with simple patterns, then move on to more complex patterns later on as they chose other sorting and filtering options.

In Figure 2, the list of patterns has been sorted by decreasing frequency and the user has clicked on four of the most frequent patterns. The location of the patterns is displayed on the *Document Overview*. Each pattern has been assigned a different color reflected in the *Legend* panel. When a paragraph contains one of the selected patterns, the color saturation of the line reflects the score of the pattern in the paragraph: the more occurrences of the pattern in the paragraph, the more saturated the color. I AM HERE

The *Collection Overview* panel shows a graph of the distribution of the support for each selected pattern. The vertical axis

represents the support of the pattern per document and the horizontal axis shows the documents of the collection. When the user lets the mouse hover on a specific portion of the graph a popup shows the exact number of occurrences. In Figure 2, the distribution of the word "world" is displayed in blue, showing that the word was barely used in the first speech.

By looking for lines that contain all the colors in the *Documents Overview,* it is possible to identify the parts of the text where selected patterns occur together. Clicking on a colored line in the overview displays the text of the corresponding paragraph in the *Text View* along with five paragraphs before and after the selection, to provide context while maintaining fast response time. In Figure 2, the user as selected a paragraph that contains all 4 patterns. A blue horizontal bar indicates which paragraph is currently displayed in the *Text View*. The text of the selected paragraph has a matching light-blue background. In the right margin of the *Text View*, small colored tick marks indicate the position of the occurrences of the patterns with respect to the scrollbar to make them easier to find. The occurrences of the patterns in the text are highlighted in color as well, matching the colors used in the overview and the legend.

The *Frequent Patterns* panel on the left provide many controls to search, filter and sort the list of patterns. A search box allows users to find patterns that include particular keywords or Boolean combinations of keywords. Patterns can be filtered by minimum size (i.e. number of words) and minimum frequency within the whole collection. Patterns can be sorted by length or frequency. Above the list of patterns, a check box allows users to append patterns to the current display in order to compare and study correlations between different patterns (the default option is to show one pattern at a time). Buttons allow users to load the history of previously explored patterns, load another collection, or set options such as the size of the text to be represented as a line in the *Document Overview*.

In the next section, we describe the pattern mining process used in FeatureLens, to explain how patterns other than the trivial single word patterns are mined from the text.

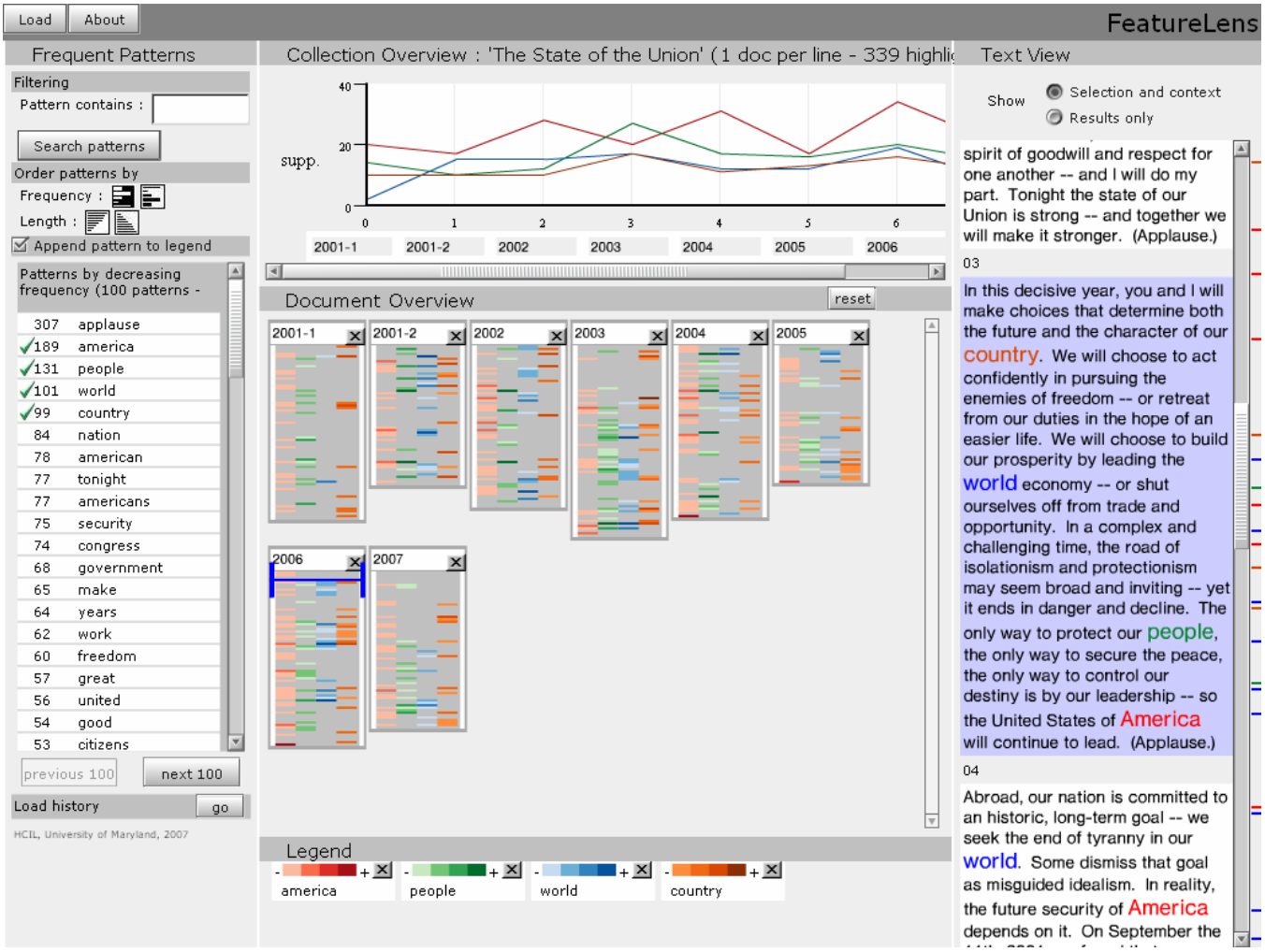

Figure 2: Main screen of FeatureLens with four of the most frequent text-patterns displayed in the overview.

5. MINING FREQUENT PATTERNS

For a given collection of texts, a hierarchical structure with two levels is assumed. Each collection contains documents which contain at least one paragraph (our chosen name for a small unit of text in this paper). This document-paragraph hierarchy can be used for a variety of text collections (see Section 7).

At this stage of the project, our focus is on the study of repetitions so we chose mining techniques that look for frequently occurring patterns, but we believe that the interface we developed can be used to interpret the results of other types of data mining techniques that generates lists of patterns.

Single words are the simplest form of patterns. For longer expressions, exact repetitions are useful because they often correspond to meaningful concepts or slogans, for example, "the No Child Left Behind Act" appears several times in President Bush's speeches. Exact repetitions, though, cannot capture language constructions that include some varying parts, such as the ones in "improve our health care system" and "improve the health of our citizens." In order to enable the user to study exact repetitions as well as repetitions with some slight variations, we resorted to the analysis of frequent closed itemsets of n-grams.

For each collection of texts, one set of frequent words and one set of frequent closed patterns of 3-grams are extracted using algorithms implemented in the Data-to-Knowledge (D2K) framework which leverages the Text-to-Knowledge (T2K) components [12].

Frequent expressions

In order to qualify a word or a longer expression as "frequent," we introduce the definitions of n-gram and the support of a pattern.

Definition 1. N-gram: a subsequence of n consecutive words from a sequence of words.

Definition 2. Support of an expression: Let $C = \{p_1, ..., p_n\}$ be a collection of n paragraphs, and let e be a text expression. The support of e in the collection C, denoted $S(e,C)$, is:

$$S(e,C) = Cardinality\left(\{p_i \mid e \subset p_i\}\right).$$

We consider an expression as "frequent" if its support is strictly greater than one. In case of large collections of texts, the threshold for the support may be increased in order to limit the number of frequent patterns.

Frequent words

D2K/T2K provides the means to perform the frequent words analysis with stemming and we know that humanists are interested in looking at both stemmed and non-stemmed versions. We also know that sometimes, the humanist is interested in keeping stop words. In our current scenario, we did not use stemming, but we did remove stop words, such as 'a, 'the,' 'of,' etc. using the predefined list provided with T2K. One set of frequent words per collection of documents was computed using a minimum support of 1.

Frequent closed itemsets of n-grams

Frequent pattern mining plays an important role in data and text mining. One relevant example is detecting associations between fields in database tables [1, 10]. We use these ideas in our text pattern analysis.

In order to provide repeated expressions that are exact repetitions as well as repetitions with slight variations, we propose to use frequent closed itemsets of n-grams, which we will refer to as *frequent patterns of n-grams* in the rest of this paper. We first reproduce the general problem definition found in [10] for clarity, we will then define how it can be applied to text pattern analysis.

Let $I=\{i_1, i_2, ..., i_n\}$ be a set of items. An itemset X is a non-empty subset of I. Duple $<tid, X>$ is called a transaction if tid is a transaction identifier and X is an itemset. An itemset X is contained in a transaction $<tid, Y>$ if $X \subset Y$.

We are interested in finding all the frequent itemsets in the database. An itemset is called frequent if its support is above a given threshold.

In our case, the set I is the set of all the possible sequences of *3* consecutive words (3-grams) in all the paragraphs of the collection of text. A transaction is a tuple $<par\_id, X>$, where *par_id* is a paragraph identifier and X is the set of 3-grams of this paragraph. One frequent itemset is a set of 3-grams that occur together in a minimum number of documents (fixed with a support threshold). Such a set of 3-grams may correspond to an exact repetition in the text or may be a repetition with variations, where only parts of a sentence are exactly repeated but where some "holes" correspond to variations.

Let us consider the set of paragraphs shown in Table 1.

par_id	paragraph
1	I will improve medical aid in our country
2	I will improve security in our country
3	I will improve education in our country

Table 1: Toy collection with three paragraphs

Let us consider I, the set of all 3-grams for these paragraphs:

I = {"I will improve", "will improve medical", "will improve security", "will improve education", "improve medical aid", "improve security in", "improve education in" , "medical aid in", "aid in our", "security in our", "education in our", "in our country"}

If we consider a support threshold of 3 paragraphs, then the frequent itemsets are:

X_1 = {"I will improve", "in our country"}

X_2 = {"I will improve"}

X_3 = {"in our country"}

X_1 is an example of a frequent itemset of 3-grams that captures a repetition with slight variations. In the context of a collection of political speeches, we hope that this pattern would invite the user to analyze not only the common parts, but also the differences, which, in this case, are meaningful, i.e. the user may declare: "the speaker is making promises."

X_2 and X_3 are also frequent itemsets but they seem redundant because X_1 carries the same information in one single itemset. We get rid of such smaller itemsets because in case of real documents the number of such sub-patterns could be dramatically high, making their analysis by the user impossible in practice.

According to the following definition, X_1 is a frequent **closed itemset** but X_2 and X_3 are not.

Definition 3. Closed itemset [10]: An itemset X is a closed itemset if there exists no itemset X' such that:

1. X' is a proper superset of X, and
2. Every transaction containing X also contains X'.

The following definition is adapted to our work from the definition of closed itemsets.

Definition 4. Pattern (a closed itemset of 3-grams): A set of 3-grams X is a pattern if there exists no set of 3-grams X' such that:

3. X' is a proper superset of X, and
4. Every paragraph containing X also contains X'.

Pattern visualization

In order to help the user interpret patterns made of sets of 3-grams, a tooltip is associated to long patterns in the list. Figure 3 shows an example of the tooltip. The list of 3-grams that compose the pattern is shown in popup window. Each 3-gram is separated by a vertical bar. The first occurrence of the pattern in its context is given as an example to help the user interpret it.

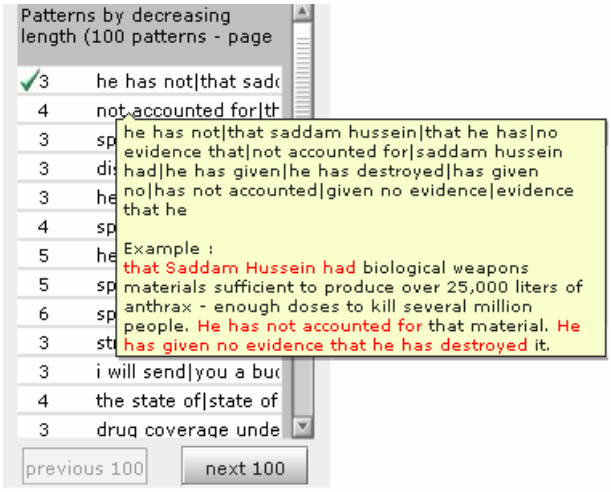

Figure 3: The list of pattern sorted by length, and the tooltip associated to the longest frequent patterns of 3-grams from *The State of the Union* collection.

When a pattern is selected, the paragraphs that contain all the 3-grams of the pattern are colored in the *Documents Overview* panel. The corresponding paragraphs can be displayed in the *Text View* to read the different contexts associated with the pattern. Some paragraphs in the *Text View* may contain only a subset of the n-grams of the pattern; these partial matches are distinguished from exact matches by using different font size. Figure 4 shows three paragraphs that contain the pattern (itemset of 3-grams) shown in Figure 3. A larger font size is used along with coloring to show where an exact match occurs (i.e. all the 3-gram of the selected pattern are contained in the paragraph). Partial matches are also highlighted but they appear with a regular font size.

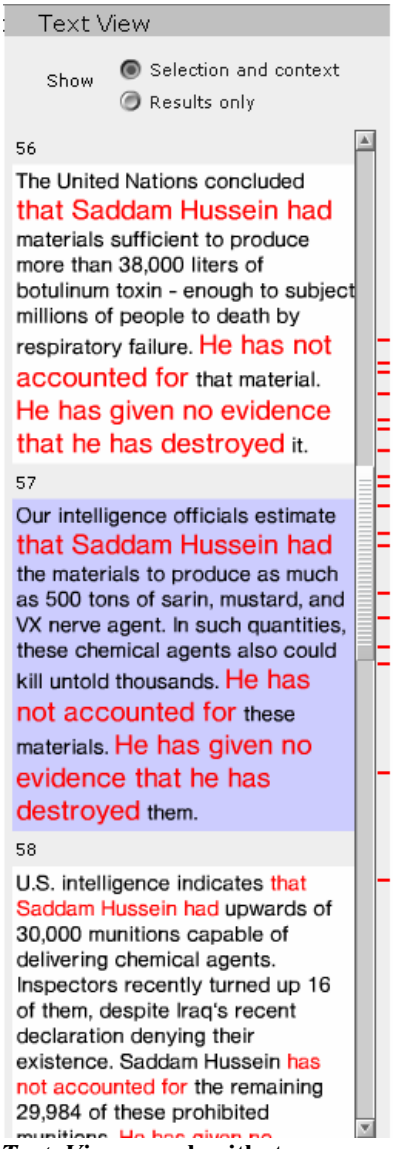

Figure 4: *Text View* panel with two paragraphs that contain an exact match (paragraphs 56 and 57) and one paragraph with only a partial match (paragraph 58).

6. ARCHITECTURE

The system was implemented with OpenLaszlo for the client interface and with Ruby and MySQL for the backend part. OpenLaszlo was chosen in order to provide a zero-install access to the user interface and to make FeatureLens accessible from any web browser. Figure 5 shows a sequence diagram of the different parts of the architecture.

The text collections are preprocessed off-line. The frequent words and frequent closed itemsets of 3-grams are computed and stored in a MySQL database together with the text from where they were extracted. OpenLaszlo's visual components are tied to XML files. The XML files may be static or returned by a Web Service over HTTP. The application heavily relies on textual data and full text queries, therefore it needs to:

1) Store the text documents in a structured way,

2) Have an efficient way to make full-text queries and format the output with text coloring,

3) Format the output documents into XML so that OpenLaszlo can use the results.

The Ferret package (a Ruby port of the Lucene tool) was used to build text indices for full-text queries within text documents and the set of frequent patterns. This is very efficient and has a lot of useful features for building text indices (stemming, stop-words filtering) or for querying the index (Boolean queries and sort filters).

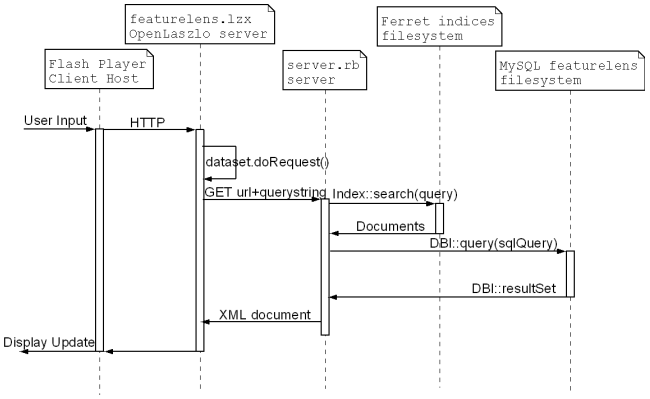

Figure 5: Architecture overview: sequence diagram of the software parts of FeatureLens.

7. APPLICATION EXAMPLES

FeatureLens can handle a collection of texts which can be represented as a two-level hierarchy. In a collection of multiple books, the two-level hierarchy can use books and chapters; for a collection made of a single book it can be chapters and sections, in a collection of publication abstracts, year and abstract, etc.

We experimented with different types of text collections. These texts included two books, *The Making of Americans* by Gertrude Stein and *Gamer Theory* by McKenzie Wark, one speech sequence, namely the *State of the Union Addresses* of the U.S. President Bush for the years 2001 through 2007, and abstracts of the research papers published by the University of Maryland Human-Computer Interaction Lab (HCIL) from 1985 through 2006. Each text collection has its own unique characteristics, and using FeatureLens led to interesting insights for each of them.

The book *The Making of Americans* includes a large number of repetitions. The text is divided into paragraphs and the paragraphs make up the nine sections of the book. Because of the large size of the book, the second level in the hierarchy was chosen to be a unit of five paragraphs instead of one to provide a more compact overview.

The second book, *Gamer Theory*, is a "networked book" created by *The Institute for the Future of the Book*. It is designed to investigate new approaches to writing in a networked environment, when readers and writer are brought together in a conversation about an evolving text. A challenge was set forth to visualize the text, and FeaturesLens participated in it[2]. The text consists of nine chapters.

To show FeatureLens' ability to handle diverse text collection types and to provide interesting but simple examples for our user testing, the *State of the Union Addresses* 2001 through 2007, and the HCIL technical report abstracts from 1984 to 2006 were preprocessed, as well. They were both separated into documents by the publication year.

8. PILOT USER EVALUATIONS

FeatureLens was evaluated by performing 2 forms of pilot studies. The tool was used by the literary scholar whose doctoral study deals with the analysis of *The Making of Americans*. In addition a pilot study using *the State of the Union Addresses* was conducted to identify usability problems and see if FeatureLens' allows users to generate interesting insights about the text collection.

The State of the Union Addresses

The study had eight participants, all of them either had advanced degrees, or were graduate students. Seven had experience in writing computer programs, and five had written software for text analysis. The evaluation consisted of 1) providing the user with background information on FeatureLens and the user study, 2) showing a short demonstration of the interface, 3) allowing the user to explore the text collection with the tool and to comment aloud on the interface and on interesting insights on the text collection. The output from the user study was a list of insights, suggestions for improvement of the tool, and a record of which parts of interface were used during the free exploration.

The exploration had two parts, and it lasted 20 minutes per user unless the user chose to continue further. In the first part, the users were asked to answer two questions:

1. How many times did "terrorist" appear in 2002? The president mentions "the American people" and "terrorist" in the same speeches, did the two terms ever appear in the same paragraph?
2. What was the longest pattern? In which year and paragraphs did it occur? What is the meaning of it?

[2] http://web.futureofthebook.org/mckenziewark/visualizations

These questions allowed the users to get acquainted with all the parts of the interface. The users were allowed to ask questions during the exploration, for example, on how to do a particular task or what some part of the interface meant. In the second "free exploration" part, the users were asked to explore the text collection freely, and to comment on their findings.

The goal of the second question was to check if frequent itemsets of 3-grams could be interpreted. The user could find the correct answer via sorting the list of frequent patterns by decreasing size and then, by reading the first element of the list, which is the following set of twelve 3-grams:

I = {"he has not", "that saddam hussein", "that he has", "no evidence that", "not accounted for", "saddam hussein had", "he has given", "he has destroyed", "has given no", "has not accounted", "given no evidence", "evidence that he"}.

By selecting this pattern from the list, the user could identify three consecutive paragraphs, in the 2003 speech (paragraphs 55, 56 and 57), that contained the twelve 3-grams. Figure 4 shows the details of paragraphs 56 and 57 with the corresponding 3-grams highlighted in red.

All the eight users found correctly and localized the longest pattern in the text collection. Moreover, all of them were able to provide a meaningful interpretation of the pattern, in this case, a repeated expression with variations. This repetition was interpreted by all users as either: an accusation, an attempt to persuade or an attempt to accuse. This example shows that large itemsets of 3-grams can lead to valuable insights, and that they "can" be interpreted with the proposed visualization. Nevertheless it is clear that understanding what a itemsets of 3-grams is is challenging. In our early prototypes which did not provide any example in context (see Figure 3) users were very confused by the long patterns. In our pilot study users were successful in understanding the particular example we selected after receiving a demonstration, but novice users stumbling on such a tool on the internet may still be overwhelmed. While the inclusion of single word patterns alongside the more complex patterns lowered the entry level complexity, video demonstrations and tutorials will be important to help users learn to take advantage of the more advanced pattern mining.

During the free exploration, the users mostly used text queries to find patterns that included specific word of interest, mainly dealing with war, economy and education. Some of the insights came from looking at a single pattern and others required looking at several. Many insights related the appearance of a specific term with another. For example, whenever the President used the phrase "lead the world," he was referring to environmental topics and not to the "war on terrorism.". Some other examples include "the President usually means the *U.S. economy* when mentioning *economy,*" "*security* and *congress* occur once together, and it is related to the *Bioshield* project". These examples illustrate the benefit of visualizing different expressions on the same overview: it helps quickly identify the collocation of different expressions and derive new insights and questions.

Four out of eight users derived some questions from the trends in the distribution of support for particular expressions. Example comments are: "there is a peak in 2005 for the expression *men and women,"* "the term *terror* has a peak in 2002 and *law* has one in in 2004," and "before 2002, there is no mention of *economy* whereas there were mentions after the Internet Bubble Crash." Most users tried to elaborate on these comments by analyzing the context of each expression in order to find an explanation of these trends.

The most common work flow consisted of typing an expression in the search field, then selecting patterns from the returned list. In this case, patterns made out of large itemsets of 3-grams were useful because they provided some contextual information about the searched expression. Surprisingly, most users did not use the sorting by pattern length or frequency during the free exploration, this might be due to the short exploration time (20 minutes) which led users to search for things they had a particular interest in opposed to looking at the default lists of patterns. Finally, the visualization of the trends in the distributions was also used successfully.

Suggestions for improvement included allowing search for 2-grams, exact n-gram search, or using a different color-code for paragraphs where all selected patterns co-occured. One interesting comment concerned the size of frequent patterns of 3-grams. When patterns are sorted by decreasing size, the size corresponds to the number of 3-grams in the pattern and not to the actual number of distinct words. Sorting by the number of distinct words in the pattern might be more appropriate.

The Making of Americans

We also collected feedback from the early stages of our planned long term case study of the use of FeatureLens with *The Making of Americans*. Our literary expert acted as a design partner during the development of the tool and was eager to test it with her data. There had been dozens of meetings over a period of six months, and the feedback on early interfaces had been incorporated in the version described here. She was particularly interested in frequent patterns with variations and looking forward to sharing her findings with other literary experts. After the prototype reached a satisfactory level of stability, she was able to use it with her data in a free exploration session that lasted two hours. The output from this pilot study was a list of comments and insights about the text.

The first task we encouraged her to tackle was to try to confirm with FeatureLens a finding she had discovered with difficulty with other tools. In this case, her chosen question was about the way the author is referring to the *bottom nature* of her characters (i.e. personality traits). She started by searching for the patterns including the word "kind" then refined the query with a Boolean search (a feature she had requested in early design sessions) "kind NOT men NOT women NOT them". After scrolling through the resulting list of itemsets of 3-grams, the user easily found and selected *the attacking kind*, *the resisting kind*, *independent dependent kind*, *dependent independent kind* and *engulfing kind of*. These character traits were found in the different sections which describe each character (something the expert already knew); in this way, she was able to confirm that particular personality descriptors could be matched to particular characters.

Afterwards, the expert started to study the way that one particular character is described. The book, *The Making of Americans* is about two families, and in section 1 and 2, the story is centered on the childhood of three members, two brothers and a sister. The

expert identified this part of the text by selecting several house-related terms, which are depicted in Figure 6.

The expert knew the story began with the children and assumed that the words *house* and *governess* would appear in sections 1 and 2. After selecting these terms, the expert noticed that these terms were also occurring together in the first part of section 4. By reading the corresponding paragraphs, she noticed that section 4, which focuses on the sister Martha Hersland, begins with a repetition of the childhood story from sections 1 and 2. The user developed the hypothesis that this reintroduction of the children was probably linked to the story of Martha's failed marriage with her husband *Redfern*. The display of all the occurrences of the word *Redfern* showed that his character was indeed mentioned, just after the repeated section concerning the children.

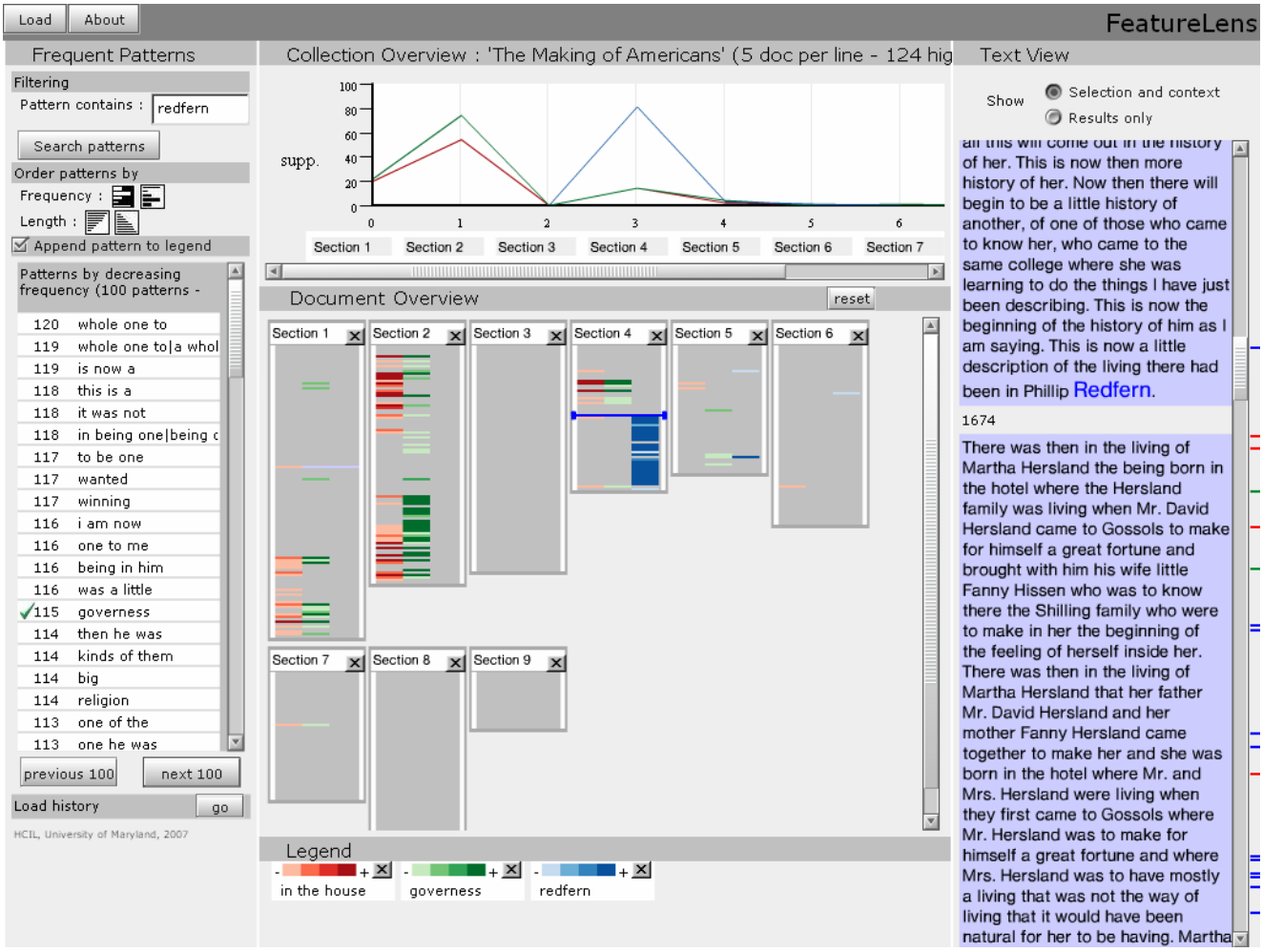

Figure 6: Screen capture of FeatureLens with house-related terms displayed.

Similarly, the expert was able to discover that the author uses the concepts of success and failure when describing marriage in section 6. The words *succeeding*, *failing* and *married* were selected together and the user noticed that the concept of marriage was mentioned in sections 1, 2 and 6, and that it was only associated with the concepts of failure and success in section 6. For the user, this fact supported the idea that the author was describing the marriage factually at the beginning of the book, and then the author introduced a judgment in section 6.

These explorations took place in rapid succession and illustrate how users combine searching, browsing, comparing, and reading in tightly connected ways. The expert was very pleased by what she had been able to accomplish in a couple of hours, providing some evidence that the proposed system could supports discoveries and lead to useful insight. As for the first group of users, the long frequent itemsets of 3-grams were not examined in the short session, but the shorter itemsets of 3-grams were used extensively in conjunction with text search. A longer case study will shed more light on the potential benefits of the tool.

9. CONCLUSION

We described FeatureLens, a system which allows the visual exploration of frequent text patterns in text collections. We applied the concepts of frequent words, frequent expressions and frequent

frequent closed itemsets of n-grams to guide the discovery process. Combined with the interactive visualization, these text mining concepts can help the user to analyze the text, and to create insights and new hypotheses. FeatureLens facilitates the use of itemsets of 3-grams by providing the context of each occurrence. The display of the frequency distribution trends can be used to derive meaningful information. The display of multiple expressions at a time allows studying correlations between patterns.

The user study with *The State of the Union* collection suggests that at first time users use text search as a mean of initial exploration to find patterns of interest, instead of looking at the longest patterns. Being able to display patterns simultaneously was important to make comparisons.

In our future work we will investigate better means of exploration of long patterns and look at more diverse kinds of texts, especially large collections of text where a two level hierarchy may not be sufficient. We will also support the filtering of patterns by their usage trend over time. Metrics can be defined to characterize frequency distributions associated with each pattern and identify that are increasing, decreasing, showing spikes or gaps, etc. Finally, we have focused here on patterns of repetitions, other features can be extracted from the text (e.g. name entities, part of speech patterns) and explored in a similar fashion.

10. ACKNOWLEDGMENTS

We would like to thank the volunteers who participated in the user studies for their time and feedback, and Celeste Paul and Matt Kirschenbaum for their feedback and suggestions . Support for this research was provided by the Andrew Mellon Foundation.

11. REFERENCES

[1] Agrawal, R., and R. Srikant, Fast algorithms for mining association rules. In *Proc. 1994 Int. Conf. Very Large Data Bases (VLDB'94)*, 487-499. 1994.

[2] Church, K.W., and Helfman, J.I., Dotplot: A Program for Exploring Self-Similarity in Millions of Lines of Text and Code, In *Proc. of the 24th Symposium on the Interface, Computing Science and Statistics V24*, 58-67. 1992.

[3] Eick, S.G. and Steffen, J.L. and Sumner Jr, E.E., Seesoft-A Tool for Visualizing Line Oriented Software Statistics, In *IEEE Transactions on Software Engineering*, Vol 18, No 11, 957-968. 1992.

[4] Fekete, J. and Dufournaud, N., Compus: visualization and analysis of structured documents for understanding social life in the 16th century. In *Proc. of the Fifth ACM Conference on Digital Libraries*, 47-55. 2000.

[5] Frank, A. C., Amiri, H., Andersson, S., Genome Deterioration: loss of repeated sequences and accumulation of junk DNA. *Genetica*, Vol. 115, No. 1, 1-12. 2002.

[6] Kurtz, S & Schleiermacher, C. REPuter: fast computation of maximal repeats in complete genomes. *Bioinformatics* 15, 426–427. 1999.

[7] G. Lommerse, F. Nossin, L. Voinea, A. Telea, The Visual Code Navigator: An Interactive Toolset for Source Code Investigation. In *Proc. IEEE InfoVis'05*, 24–31. 2005.

[8] NY Times: The State of the Union in Words. http://www.nytimes.com/ref/washington/20070123_STATEOFUNION.html

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

[10] J. Pei and J. Han and R. Mao, CLOSET: An Efficient Algorithm for Mining Frequent Closed Itemsets, *ACM SIGMOD, Workshop on Research Issues in Data Mining and Knowledge Discovery*, 21-30. 2000.

[11] Plaisant, C. and Rose, J. and Yu, B. and Auvil, L. and Kirschenbaum, M. and Smith, M. and Clement, T. and Lord, G., Exploring Erotics in Emily Dickinson's Correspondence with Text Mining and Visual Interfaces, in *Proc. of the 6th ACM/IEEE-CS Joint Conference on Digital Libraries*, 141-150. 2006.

[12] Data to Knowledge (D2K) and Text to knowledge (T2K), NCSA. http://alg.ncsa.uiuc.edu/do/tools.

[13] Thomas, J.J. and Cook, K.A. (eds.), Illuminating the Path: Research and Development Agenda for Visual Analytics, IEEE. 2005.

[14] Veerasamy, A. and Belkin, N. Evaluation of a Tool for Visualization of Information Retrieval Results, in *Proc. of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, 85-92. 1996.

[15] Wattenberg, M., Arc diagrams: visualizing structure in strings. In *proc IEEE Symposium on Information Visualization 2002*, 110- 116. 2002.

[16] Wise, J. A. and Thomas, J. J. and Pennock, K. and Lantrip, D. and Pottier, M. and Schur, A. and Crow, V., Visualizing the non-visual: spatial analysis and interaction with information from text documents, In *proc IEEE Symposium on Information Visualization 1995,* 51-58. 1995