

FutureLens: Software for Text Visualization and Tracking

Gregory L. Shutt*

Andrey A. Pureskiy*

Michael W. Berry*

January 18, 2009

Abstract

Visualization can be a very powerful tool for text/data mining. It can facilitate knowledge discovery as well as a *big picture* overview of overwhelmingly large amounts of data. Built as a successor to the *FeatureLens* utility developed at the University of Maryland Human-Computer Interaction Laboratory (HCIL), *FutureLens* is a Java-based (portable) software environment that can organize a large set of SGML-tagged documents for both spatial and temporal tracking of words, phrases, and entities. We demonstrate its use as a back-end scenario/plot discovery tool using outputs (tensor groups) generated from non-negative tensor factorization models applied to the IEEE VAST (Visual Analytics Science and Technology) 2007 Contest dataset.

1 Background

Visual analytics is a relatively new discipline rooted in the support of analytical reasoning by highly interactive visual interfaces. Software for this field is generally designed to synthesize information and to derive insight from large, dynamic, and heterogeneous data. Effective tools will facilitate data mining of both expected and unexpected associations. The discipline of visual analytics is very interdisciplinary and extends well beyond traditional scientific and information visualization to include statistics, mathematics, knowledge representation, information retrieval, management and discovery technologies, cognitive and perceptual sciences, and decision sciences.

In this paper, we illustrate the use of a Java-based software environment, *FutureLens*, that has been used to support the extraction and tracking of scenarios and plots from news articles defining the VAST 2007 Contest [6]. Using groups of related persons, locations, and organizations and context-specific words and phrases identified (through time) by nonnegative tensor factorization (NTF) models [3], *FutureLens* was instrumental in

extracting the underlying (fictitious) terrorist activities created by Whiting et al. for the VAST 2007 Contest. Rather than focusing on the specific mathematical models (NTF) used to identify term-entity relationships for the contest, in Section 2.1 we briefly describe the scenario mining process and expectations that warrant the design of visual analytic software like *FutureLens*. An early prototype of *FutureLens* is discussed in Section 3, followed by an illustration of some of the important features of *FutureLens* in Section 4. An example of scenario discovery with the VAST 2007 Contest dataset is provided in Section 5, and a brief discussion of future enhancements to *FutureLens* is given in Section 6.

2 Scenario Discovery

The intent of the IEEE VAST 2007 Contest [6] was to promote the development of benchmark datasets and metrics for visual analytics as well as to establish a forum for evaluating different solution strategies. In providing news stories, blog entries, background information, and limited multimedia materials (small maps and data tables), the contest organizers challenged the participants to investigate a major law enforcement/counter-terrorism scenario, form a hypotheses, and collect supporting evidence. Tasks that each team/entry was expected to address included *i)* identify entities (e.g., people, places and activities) from text and multimedia information; *ii)* develop interactive tools to visualize/analyze this information; *iii)* answer specific (contest-provided) questions based on the analysis; and *iv)* produce a video that demonstrates how those answers were derived. *FutureLens* was primarily used for the second task to visualize and track the entity groups generated by the nonnegative tensor factorization models discussed in [2, 3].

2.1 Scenario. The primary (terrorism-based) scenario depicted in the VAST 2007 Contest involved wildlife law enforcement incident occurring in the fall of 2004. Endangered species issues and ecoterrorism activities played key roles in the underlying terrorist scenario/plot. The data used to describe the details of the plot included text, images, and some statistics. Al-

*Department of Electrical Engineering and Computer Science, 203 Claxton Complex, 1122 Volunteer Boulevard, University of Tennessee, Knoxville, TN 37996-3450; [shutt,pureski,berry]@eecs.utk.edu

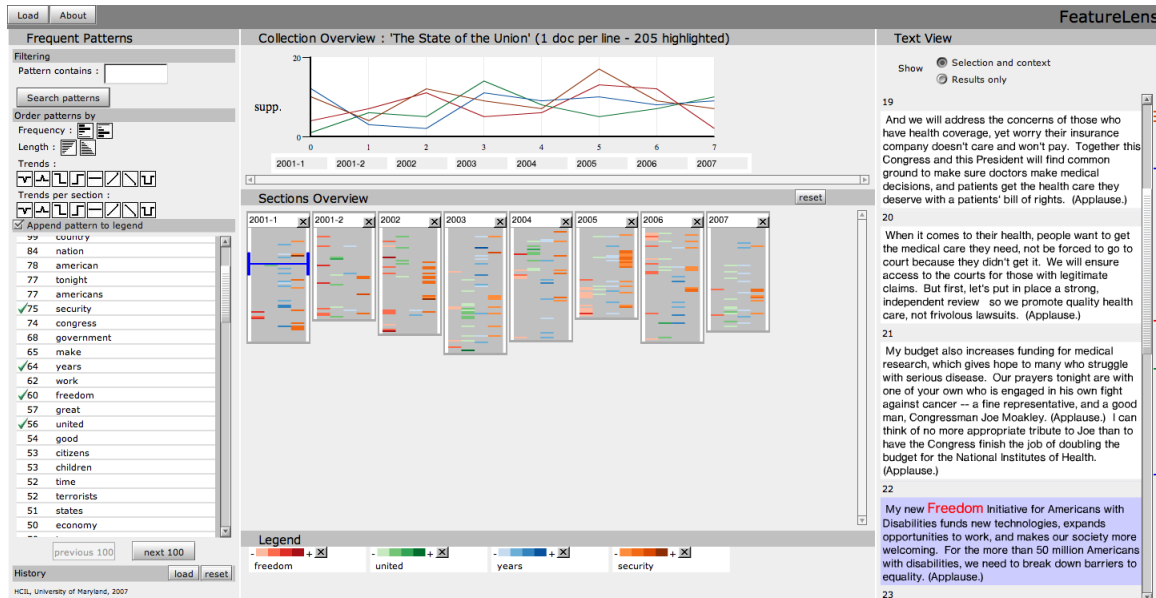


Figure 1: FeatureLens prototype (written in Ruby) developed at the University of Maryland Human-Computer Interaction Laboratory.

though activities of certain animal rights groups, such as the People for the Ethical Treatments of Animals (PETA) and Earth Liberation Front (ELF), were involved with the plot, the contest organizers did not consider them to be the primary (interesting) parties for investigation.

2.2 Evaluating Solutions. Although entries (or answers) submitted to the VAST 2007 Contest were judged according to the correctness of the answers to the questions and the evidence provided, a more subjective assessment of the quality of the displays, interactions and support for the analytical process was also provided. We are particularly interested in this second category of assessment as the field of text mining, in general, could greatly benefit from the design of more intuitive visualizations that expose or verify potential scenarios of human activity.

Following the traditional cues of journalistic reporting, visual analytics (as reflected by the VAST 2007 contest) seeks to answer the questions (who, what, where, and when) for an alleged activity using the the most relevant documents or other materials from the dataset as evidence. Contest participants were required to describe the plot(s) and subplots(s) and how people, motivations, activities and locations relate to the plot, that is, their relationships, and any uncertainties or information gaps that exist. For example, some of the questions each entry was required to answer include:

- **(Who)** Who are the players engaging in questionable activities in the plot(s)? When appropriate, specify the organization they are associated with.
- **(When/What)** What events occurred during this time frame that are most relevant to the plot(s)?
- **(Where)** What locations are most relevant to the plot(s)?

3 Earlier Prototype

Many of the concepts and ideas of this project stem from FeatureLens, a University of Maryland (Human-Computer Interaction Laboratory) text and pattern visualization program [1]. FeatureLens allows the user to explore frequently occurring terms or patterns in a collection of documents. Connections between these frequent terms and the dates at which they appear in the set of documents can quickly be visualized and investigated. A screen shot of FeatureLens prototype is shown in Figure 1.

While FeatureLens may sound suitable for the given task, it is not without its shortcomings. For one, its design is rather complex as it requires a MySQL database server, an HTTP server, and an Adobe Flash enabled web browser to function properly. As such, it is not a trivial task to set up an instance of FeatureLens from scratch and may take an inexperienced user a significant amount of time to get started. Datasets must be parsed and stored in the database, an operation

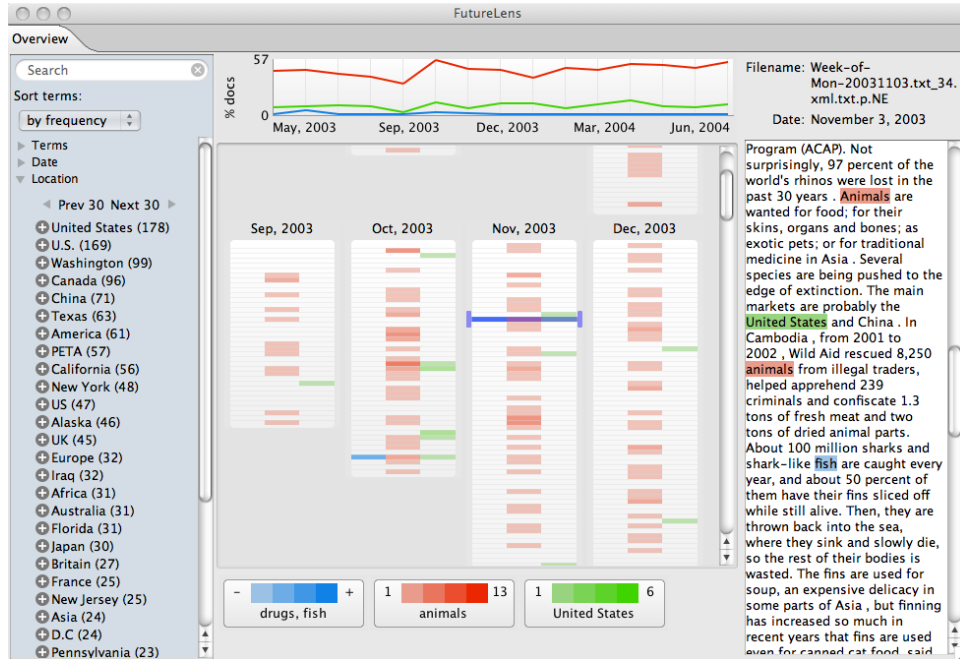


Figure 2: FutureLens prototype (written in Java) developed at the University of Tennessee for visualization of NTF-generated outputs.

that an end user cannot perform so examining arbitrary datasets is out of the question. In implementing the architecture of FeatureLens, the designers chose to use a variety of languages—Ruby for the back end, XML to communicate between the front end and back end, and OpenLaszlo for the interface. Because of this variety in languages adapting and modifying FeatureLens would prove quite difficult. Responsiveness of the interface also tends to degrade to the point that it impacts usability when given even the simplest of tasks. Clearly a better solution was needed.

4 Features of FutureLens

FutureLens is a text visualization tool that implements much of the functionality of FeatureLens while adding a few necessary missing features. It is written in the Java programming language using the Standard Widget Toolkit so it is not only cross platform but uses native widgets where possible to maintain a look and feel consistent with the users' platform. For end users not familiar with the program, FutureLens has a built in demo feature that demonstrates its basic functionality. An example of FutureLens running under Mac OS X is shown in Figure 2.

All the basic functionality of FutureLens can be seen in this example. The boxes along the bottom show the terms that are currently being investigated. The in-

tensity of the color hints at the concentration of the term throughout the documents. A graph of the percentage of documents containing the term versus time is shown at the top, while the raw text of the selected document is shown to the right with the selected terms highlighted in the appropriate color. Multiple terms can be combined into extended patterns easily by dragging and dropping. Terms may be combined into either collections or phrases. A collection is created when the user drags and drops terms onto each other. Term adjacency does not affect search results for a collection. If the users holds down the Copy key (this key varies depending on the operating system, for example, on Mac OS X this is the Alt key), a phrase rather than a collection will be created. In this case, term adjacency will be considered when the software performs searching. While this presents an excellent overview of the data, it is also possible to load the output (groups of terms and/or entities) derived from a data clustering method. An example of this is shown in Figure 3.

Here a file containing pertinent terms output from a nonnegative tensor factorization (NTF) tool has been loaded as a separate view into FutureLens. The view is nearly identical to the overview. However, the list of terms has been limited to only what was contained in the input file. This allows the user to quickly view the different clusters of entities through time [3].

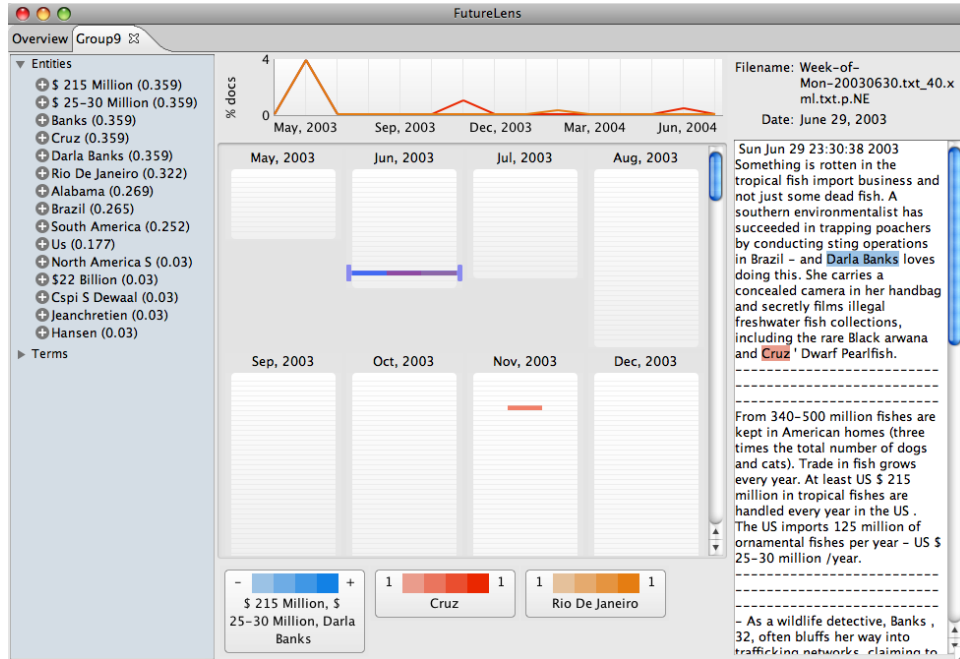


Figure 3: FutureLens tracking the co-occurrences of grouped terms and entities (persons, locations, and organizations).

5 Scenario Discovery Example

Figures 4 through 8 demonstrate how FutureLens may be used together with nonnegative tensor factorization (NTF) to quickly reconstruct a bioterrorism-related plotline that was buried within the VAST 2007 text corpus. In Figure 4, one of the NTF output groups has been loaded into FutureLens. The user is aware that he or she should be searching for some sort of an interesting and nefarious scenario. The selected terms (*Monkeypox*, *Exotic*, *Pets*, *Chinchilla*) constitute a good starting point. However, the user will not find all news articles with the occurrence of the relatively common words *Pets* and *Exotic* relevant. Thus, the two terms are combined into the phrase *Exotic Pets*, as shown in Figure 5. Figure 6 demonstrates how FutureLens allows the user to easily identify a key news story within the large dataset. The article shown in this figure contains a great amount of relevant information regarding an outbreak of a potentially deadly virus, monkeypox, in the Los Angeles area. The article implies that the outbreak may not have been accidental, and connects it to an animal rights activist and chinchilla breeder named Cesar Gil. In order to fully reconstruct the plotline, the user selects the names *Cesar Gil* and *Gil* from the Entities list, as shown in Figure 7. However, this results in too many instances of *Gil* being found, and most of them are probably irrelevant. Exploiting the link

between Gil and chinchilla breeding, the user combines the terms *Chinchilla* and *Gil* into a collection. This helps the user to quickly identify a relevant article that contains an advertisement for Gil's chinchilla breeding business (Figure 8). Not all of the articles that are relevant to this plotline have been shown in the figures, however, FutureLens enables the user to quickly and easily identify them all. FutureLens also helps the user to focus on the relevant parts of the article.

6 Future Work

While FutureLens provides numerous features for plot and scenario discovery, there is still room for improvement. It works well for evidence generation but it has no automation for any type of scenario discovery. Methods that locate interesting features in the dataset could be added to create a single analysis tool. As it stands now, the output of data mining models such as that created by the nonnegative tensor factorization (see [3]) must be manually be entered into the software environment. Eliminating this human interaction would greatly increase the efficiency of scenario discovery. An obvious extension for dynamic (time-varying) datasets is certainly needed. The portability and intuitive word/phrase tracking capability of FutureLens, however, make this public-domain software environment a solid contribution to the text mining community.

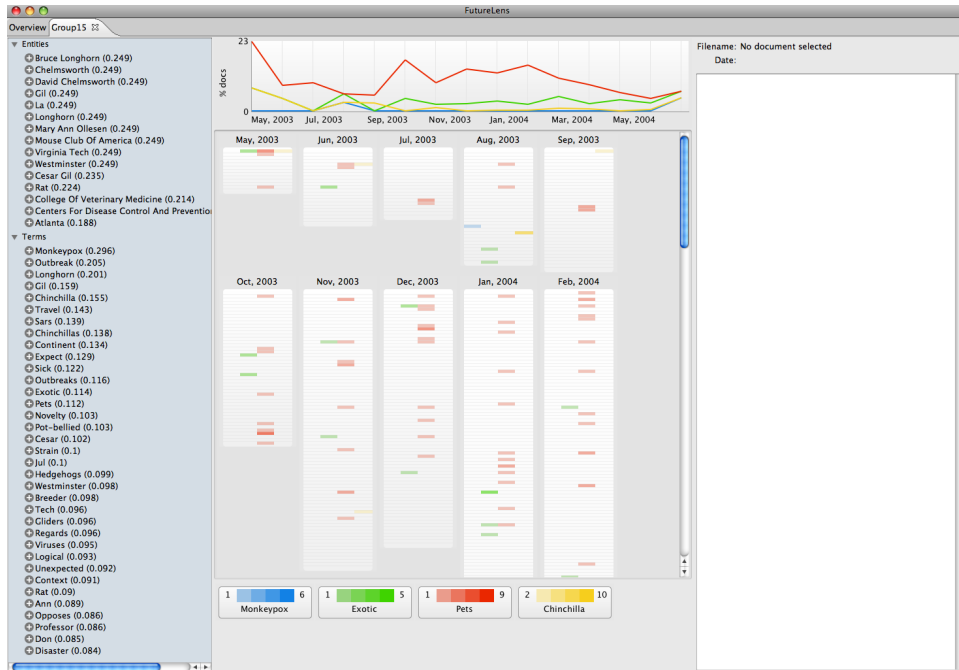


Figure 4: FutureLens with the bioterrorism NTF output group loaded.

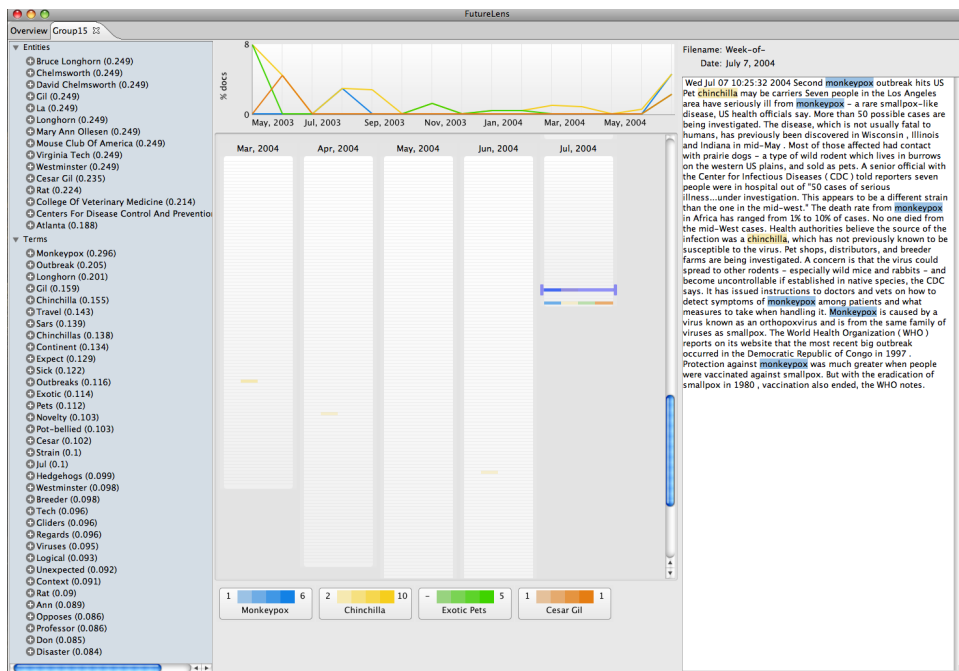


Figure 5: Demonstration of phrase creation in FutureLens.

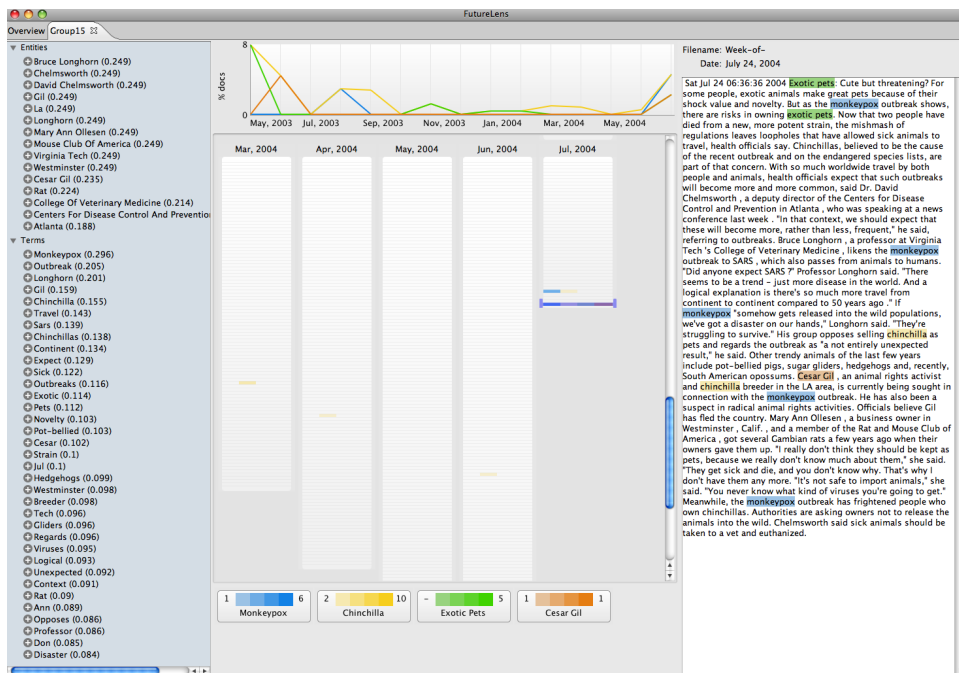


Figure 6: Key news story identification using FutureLens.

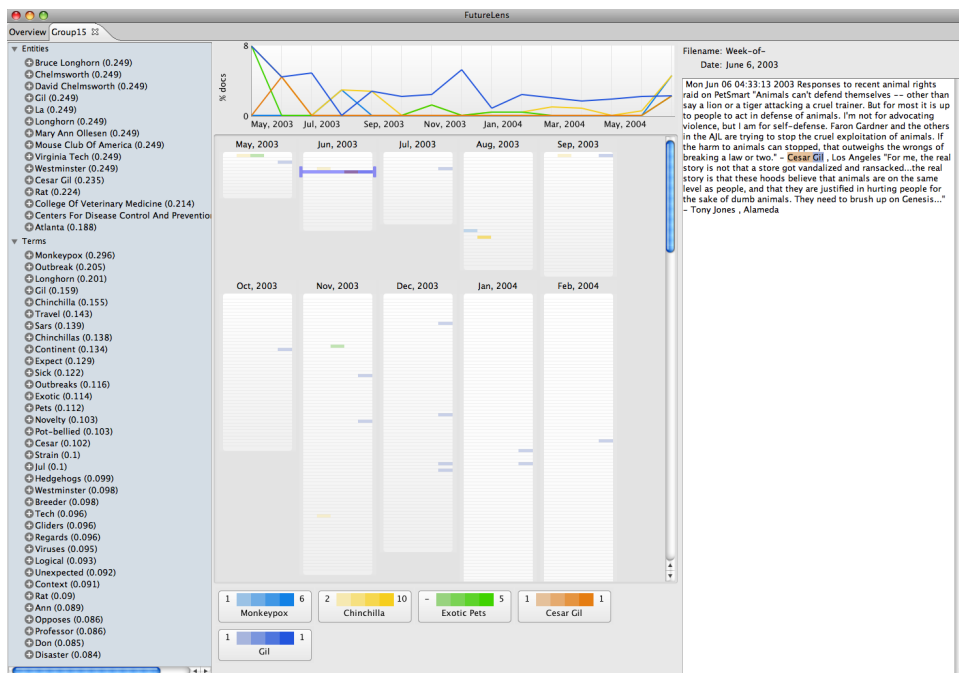


Figure 7: Entity of interest search using FutureLens.

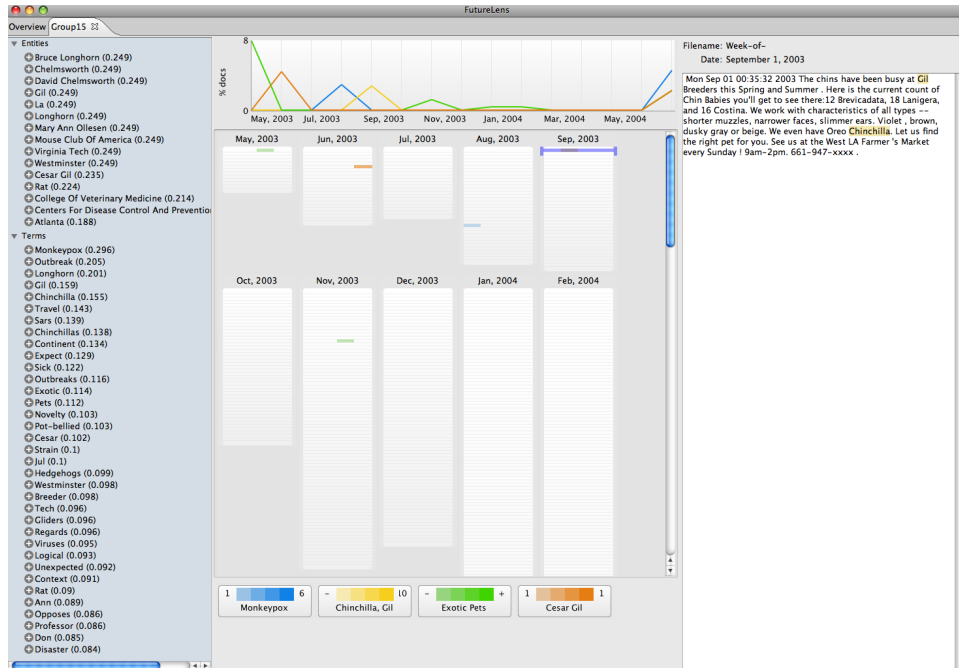


Figure 8: Term collection creation in FutureLens.

References

- [1] Exploring and visualizing frequent patterns in text collections with FeatureLens. <http://www.cs.umd.edu/hcil/textvis/featurelens>. Visited November 2008.
- [2] B. W. Bader, M. W. Berry, and M. Brown. Discussion tracking in Enron email using PARAFAC. In M. Berry and M. Castellanos, editors, *Survey of Text Mining II: Clustering, Classification, and Retrieval*, pages 147–163. Springer-Verlag, London, 2008.
- [3] B. W. Bader, A. A. Pureskiy, and M. W. Berry. Scenario discovery using nonnegative tensor factorization. In J. Ruiz-Shulcloper and W. G. Kropatsch, editors, *Progress in Pattern Recognition, Image Analysis and Applications*, pages 791–805. Springer-Verlag, Berlin, 2008.
- [4] A. Don, E. Zhelev, M. Gregory, S. Tarkan, L. Auvil, T. Clement, B. Shneiderman, and C. Plaisant. Discovering interesting usage patterns in text collections: integrating text mining with visualization. *HCIL Technical report 2007-08*, May 2007.
- [5] A. Kumar. The MONK Project Wiki. <https://apps.lis.uiuc.edu/wiki/display/MONK/The+MONK+Project+Wiki>. Last edited August 2008.
- [6] J. Scholtz, C. Plaisant, and G. Grinstein. IEEE VAST 2007 Constest. <http://www.cs.umd.edu/hcil/VASTcontest07>, 2007.