Newscape - An Interactive, Personalized, Online Newspaper

Krishna Bharat, Tomonari Kamba, Mike Albers

1.0 Introduction

Newscape is an online newspaper on HotJava. It is personalized because the layout and selection of articles is driven by previous relevance feedback from the user about articles he/she has read. In addition to *explicit* input from the user about the relevance of an article, we also record *implicit* feedback through observation of the user's interaction. The time spent in reading an article, the portions of the article that were "visited", and information about whether it was ever maximized, saved to the "scrapbook" etc. are all contributing factors.

Since the interface is implemented in Java, much of the presentation, layout and feedback can be managed within the browser itself. In addition, HotJava provides us the opportunity to embed interactive services such as "live" weather maps, and dynamic features such as animation, audio/video channels, whiteboards and billboards within the newspaper.

The architecture consists of two parts:

- **The Server End** (the news provider's end). Here articles are collected under some human supervision. The 'SMART' indexing system converts them into document vectors suitable for querying. Currently document vectors are computed based on content based filtering of articles using the TFIDF (term frequency times inverse document frequency) metric. We may add community based filtering at a later date.

  The server site also maintains a user-profile for each user. As interaction proceeds, the user's browser provides relevance feedback to the server site, where it gets assimilated. Whenever new articles come in (and also when the user requests it), the user's profile is compared with document vectors to compute the importance (weight) of each document to the user. At the same time, cumulative weights for the community are also computed. The layout is driven by both sets of weights.

  It is worth mentioning that there is no live process at the server end save the HTTP daemon, which spawns Perl scripts when necessary. Indexing is done in batch mode whenever articles are added.

- **The Client End** (the HotJava browser). The code needed to drive the presentation and interaction is downloaded when the user accesses the newspaper's web page. Subsequently, the code runs locally and may periodically contact the server site to fetch documents or provide feedback. The layout is computed based on both the user's importance vector as well as the community importance vector. The rest of this report is a "paper" design of our electronic newspaper, intended as a blueprint for our implementation.

2.0 Interaction and Layout

A major goal is to make the layout natural and keep the "look" as similar to regular newspapers as possible. This means a tiled embedding of articles in a multi-column format, with continuations and sections. At the same time we would like to include innovative interaction techniques that will support browsing in ways a hard-copy newspaper cannot. Whenever it makes sense we will deviate from the norm, and replace hard-copy browsing strategies with new ones based on accepted human computer interaction paradigms such as focus in context. For instance, rather than having to move to a new page to find the continuation of an article we will provide four browsing techniques:

(i) Scroll/Automatic scroll which slides the article vertically (if it is single column) or horizontally (if it is multi-column)

(ii) Peek, which displays the article over the entire screen as long as the mouse is held down over the peek button in the top left corner.

(iii) Maximize/Revert, which is like peek but requires double clicking on the body of the article to switch states.

(iv) Resize, where the user resizes the article and permanently affects layout.

(v) Document Lens Mode, where the column under the pointer is dynamically magnified.

Note that all these browsing strategies provide feedback about the relevance of the article to various degrees. The need for explicit feedback is almost eliminated.

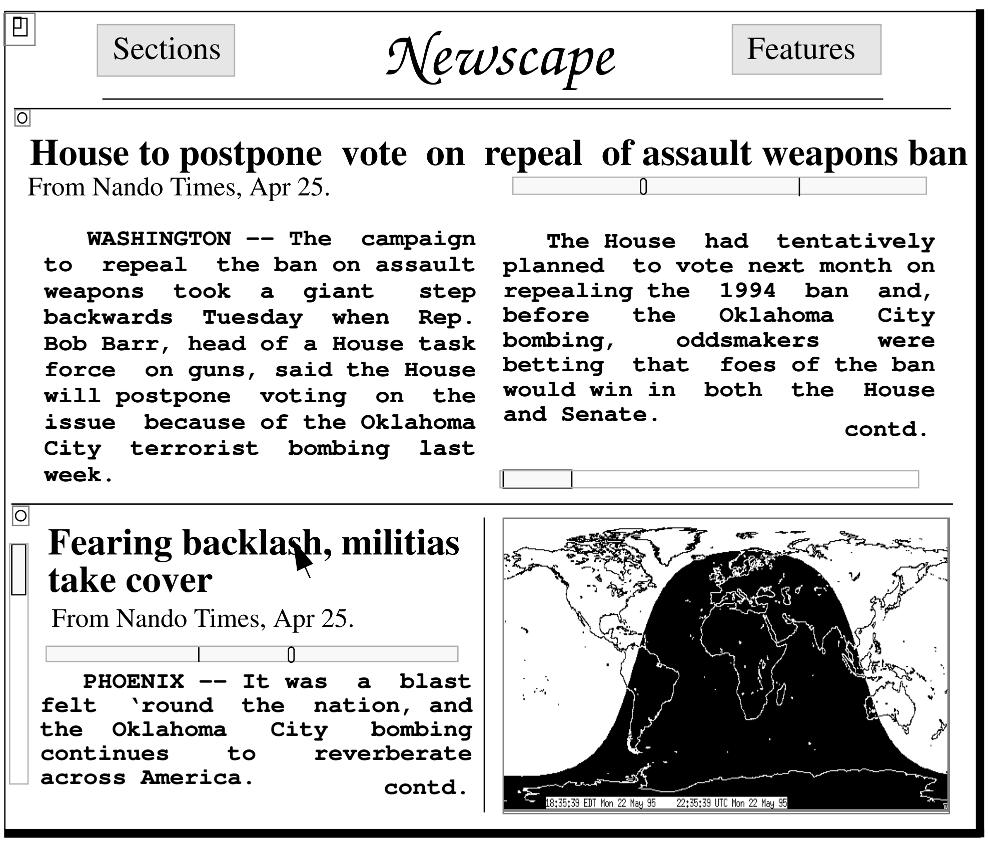

Figure 1. Partial View of *Newscape* (there will be more rows/columns typically)

The amount of real estate allocated to each article, and nearness to the top are proportional to the user's predicted interest in the article. Images are scaled to fit in the available space. To avoid cluttering the page with widgets, we have tried to integrate some gestures into the user interface. In the above the interface, the gestures "tick" (mouse button down, stroke from bottom-left to top-right, mouse button up) and "slash" (mouse button down, stroke from top-left to bottom-right, mouse button up), executed over the *title* of the article, will cause it to be saved to the scrapbook and deleted from the layout respectively. When an article is deleted, the layout needs to be recomputed.

The horizontal (or vertical) drag operation over the *body* of the article will cause the text to scroll as if it were being dragged. This can also be accomplished by dragging the thumb of the scrollbar. The drags are in different directions in the two cases. A gesture that cuts across the scrollbar will start an automatic scroll to that point in the scrollbar. To help the user distinguish between gestures over the title and gestures over the body, the cursor is change to a "hand" over the body. Clicking on the article causes the scrolling to stop. Clicking on a point inside the scrollbar allows random access to any point inside the article.

All these actions provide feedback besides assisting in browsing. At present there is no way to incorporate feedback at the level of individual paragraphs/words into the maintenance of the user profile. Nonetheless we collect feedback at this level of granularity to provide for future improvements. We can use the time spent in browsing the article and the interesting actions performed therein as an implicit measure of the user's interest. In addition there is an explicit feedback bar. This has a grayscale (not color which can be distracting) gradient to show interest ranging from "Not Relevant" to "Very Relevant" through "No Comment". The predicted community interest is shown as a vertical line on the scale. The user's predicted interest is shown by a bullet on the scale. As the user's implicit interest is recorded the bullet is moved correspondingly. It can be moved by the user as well to provide explicit feedback. Resize is done by moving the dividing bars. These can only be moved by discrete amounts. This causes a discretionary constraint to be added to the article, to be used during future layout operations.

We could have the font temporarily change size or become bold in response to the movement of the pointer, to assist the user in reading. This helps elicit feedback about the user's focus of interest. Since we cannot monitor the user's gaze vector, we can use this technique to "reward" users for telling us where their focus of interest is.

3.0 Layout Control

Layout is a function of a number of parameters: score that each article receives based on the user's profile (the user score), the average (or possibly, mode) score received by each article over the community of users (the community score), and also the size and composition (e.g. number of pictures) of each article. Since there are many ways in which these parameters may be combined to decide the final layout, we shall give the users a set of controls which they can manipulate to dynamically change the significance of various layout parameters.

Specifically the following sliders are provided:

- **User Score vs Community Score.** This decides the ratio in which the scores are combined in deciding the layout.
- **Sensitivity Factor.** This decides how significant the scores are in deciding layout.
- **Density of Articles per Page.**

We hope to let users arrive at a comfortable setting for these controls, over the course of the experiment, and probably learn something in the process. The ability to dynamically change layout settings will let them get multiple views of the newspaper, showing them what the community is interested in, as well as what their previous feedback would indicate as a viable layout.

Some of the points to keep in mind during layout:

- Articles should be laid out in their entirety whenever possible.
- The title should be placed in a single line preferrably. If that is not possible it may be left justified if it is a multiple column layout or centered if it is a single column.
- A simple layout strategy is OK if there is sufficient variation. Rectangular tesselations of the page are sufficient, provided the layout is varied over pages.
- If part of the article is hidden in a multi-column article, the dimensions should be adjusted so that the last column is completely filled.
- Pictures may be scaled down to fit in the space available. Multi-column layout is preferrable when pictures are present. It is better to keep the picture in the visible portion of the article. Partially visibility is undesirable.
- Articles in the main section (politics, headline news) need to be justified. Articles in other sections need (arts, sports etc.) can be ragged.

(to be contd)

4.0 Dynamic Components

A new feature of *Newscape* is the integration of dynamic components into the framework of a newspaper. A billboard that shows text and images and is constantly changing is an example of a dynamic component. Other examples are shared whiteboards and live maps (e.g a weather map). Some dynamic components may allow a fair degree of interaction. For example users could write to the shared whiteboard or manipulate the view point/zoom of a weather map. Typically the component will bring over the code it needs, and do all the input/output handling and most of the processing locally. Occasionally there may be communication to the server site. The integration of dynamic components into a document on HotJava is fairly easy. However in the context of a newspaper, the screen real-estate allocated to each dynamic component and its position within the newspaper is determined during layout. For this purpose users are allowed to specify their rating of each dynamic component in the system via a set-up form (not shown).