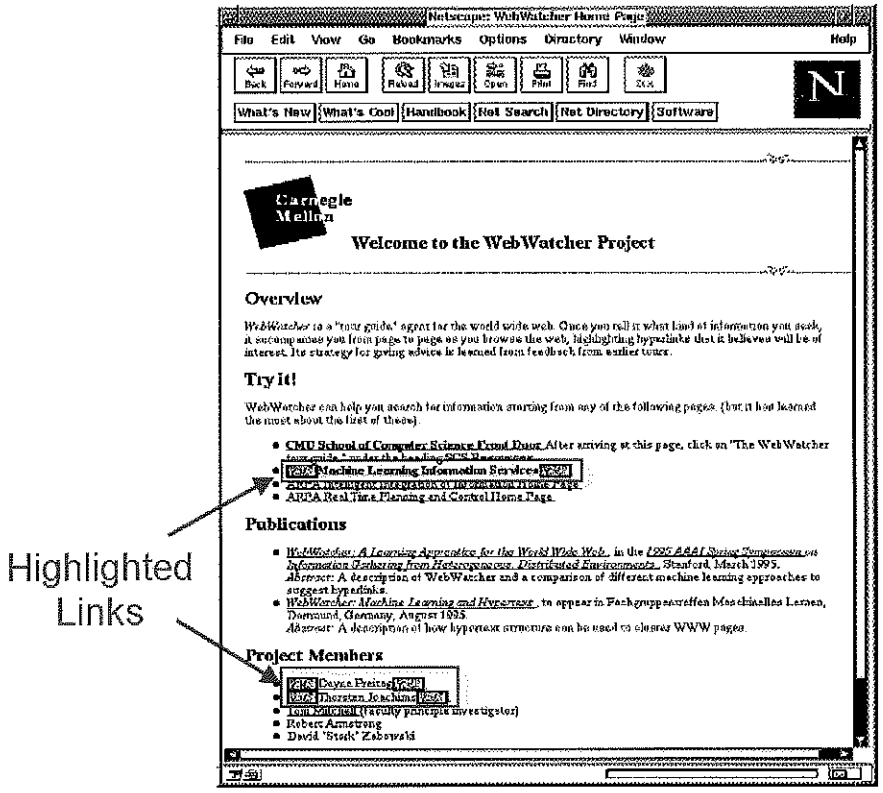

Figure 3: Example of HTML page presented to the user by PWW.

Figure 11: Mladenic at 9, Fig. 3 (notations in red added)

102. Personal Web Watcher treats a web page as a "bag of words," meaning that "all words from the document are taken and no ordering of words or any structure of text is used." (Mladenic at 3-4.) For example, Figure 1 of Mladenic discloses counting each keyword within the document, i.e. computing a frequency vector:

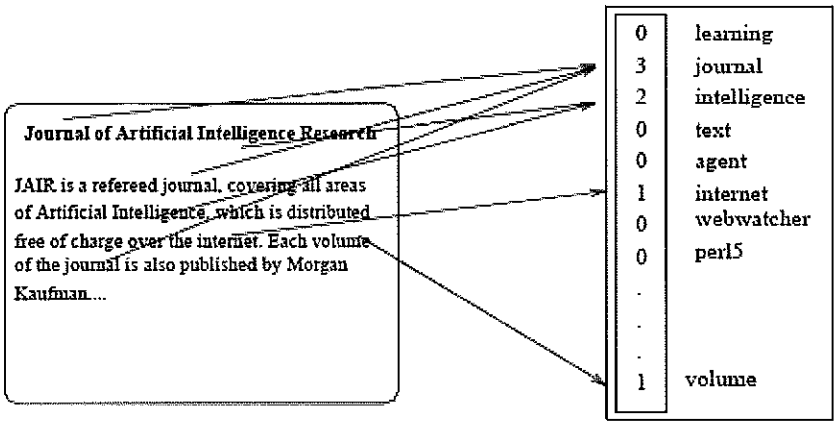

Figure 1: Bag-of-words representation using frequency vector.

Figure 12: Mladenic at 5, Fig. 1

Mladenic details various prior art means of representing documents before settling on using frequency of words. (*Id.* at 4.) Of note, using frequency vectors is an embodiment of the asserted patents. (11:1-4.)

103. Personal WebWatcher deems hyperlinks visited by the users to be positive examples and hyperlinks not visited by the user to be negative examples: "[Personal WebWatcher] fetch[es] visited documents and documents one step behind the hyperlinks of visited documents and store[s] them as positive or negative examples of user interests, depending on whether the user visited the document or not." (Mladenic at 8). In other words, "all the hyperlinks clicked by the user are considered positive examples and all other hyperlinks that were shown to the user (but remain un-clicked) are negative examples." (Mladenic2 at 2.) "The model of user interests is designed to predict if some document is a positive or negative example of user interests." (Mladenic at 10). This is accomplished by a learner module, which uses "a Naïve (Simple) Bayesian classifier on frequency vectors to generate a model of user interests, that is used for advising hyperlinks." (Mladenic at 7); *see also* Mladenic at 12, table 2:

UserId and data source	probability of interestingness	number of examples	data entropy
usr150101			
Doc	0.094	1 333	0.449
HL	0.104	2 528	0.480
usr150202			
Doc	0.107	3 415	0.492
HL	0.053	1 798	0.301
usr150211			
Doc	0.089	2 038	0.436
HL	0.044	2 221	0.259
usr150502			
Doc	0.100	1 272	0.468
HL	0.100	2 498	0.468

Table 2: Data characteristics for document (Doc) and hyperlink (HL) data for each of the four HomeNet users.

Figure 13: Mladenic at 12

6. Autonomy's Agentware

104. Autonomy's Agentware system is described in a March 10, 1997 press release issued by Autonomy regarding its Agentware software ("Autonomy PR"), the 1999 *Autonomy Technology Whitepaper*, AUT00068-88 ("Autonomy WP"), and the 1996 Autonomy Agentware User Guide ("Autonomy UG"). I understand that the papers are accordingly prior art publications with respect to the asserted patents, which claim priority to a December 1999 provisional application. Furthermore, Autonomy Agentware was in use by the 1996 publication date of its User Guide. I understand that Agentware is a prior use with respect to the asserted patents, as it was in use more than a year before December 1999.

105. Autonomy's Agentware is a piece of software that "enables automatic personalization of Web content as well as flexible management of unstructured information." (Autonomy WP at 11.) Autonomy's Agentware employs "agents," or small programs that "locate information based on concepts and context, thereby selecting the most relevant information according to the individual's preferences." (Autonomy PR at 1). These personalized agents can

learn about a user's interests in a variety of ways, including by "simply observing a user's actions." Accordingly, agents "do not require [the user] to fill out lengthy questionnaires or rate his likes and dislikes." (Autonomy WP at 3; *see also id.* at 2.)

106. Agents can "analyze a text and identify the key concepts within the document because [they] understand[] how the frequency and relationships of terms correlate with meaning." (Autonomy WP at 1). Once an agent has been trained, "it can compare [a pattern] to the documents it finds on the Internet. Autonomy does not use keyword searches, but actually identifies the concepts involved in the text and compares them. It then assigns a relevance to the document depending on how closely it matches the patterns established by the training." (Autonomy UG at 4). "By maintaining a set of agents that correspond to a user's interests, on-line publishers and service providers can then offer a range of personalized services." (Autonomy WP at 2.)

7. A Personal Evolvable Advisor for WWW Knowledge-Based Systems (Montebello)

107. The Personal Evolvable Advisor system or "PEA" is described in "A Personal Evolvable Advisor for WWW Knowledge-Based Systems" by M. Montebello et al., which was published in March 1998. I understand that Montebello is accordingly a prior art publication with respect to the asserted patents, which claim priority to a December 1999 provisional application. Furthermore, PEA was in use by at least Montebello's publication date. I further understand that PEA is thus a prior use with respect to the asserted patents, as it was in use more than a year before December 1999.

108. PEA is essentially a "meta-agent" that sits on top of existing search engines as well as existing search retrieval systems such as Mladenic. (Montebello at 2.) PEA retrieves user documents, specifically the documents that the user has in his bookmarks. (*Id.* at 3.) PEA uses the term frequency/inverse document frequency measure to generate a user profile based on

those documents. (*Id.*) As a user employs search engines or other document retrieval systems, PEA filtering incoming documents according to the profile and presents personalized information to the user. (*Id.*, Figs. 1, 2.)

8. Experience with Rule Induction and k-Nearest Neighbor Methods for Interface Agents that Learn (Payne)

109. Payne, published in 1997, describes an interface agent architecture that learns users' profiles from observations. I understand that Payne is accordingly a prior art publication with respect to the asserted patents, which claim priority to a December 1999 provisional application. "In essence, the system is an apprentice which autonomously observes and analyzes user behavior in dealing with mail." (Payne at 3.) Payne generates feature sets from the monitored mail messages, which in turn are used to create a user profile. (*Id.* at 2.) The feature extraction module is also used to make predictions on new articles. As before, features are extracted from the mail (*id.*), then compared with the user profile to generate a prediction of the user's interest. (*Id.* at 4.) The agent then informs the user if any incoming articles were judged to be interesting. (*Id.*)

9. Learning Mechanisms for Information Filtering Agents (Payne2)

110. Payne2, published in 1995, describes both an email and a browser system. I understand that Payne2 is accordingly a prior art publication with respect to the asserted patents, which claim priority to a December 1999 provisional application. Regarding the mail system, "[a] version of Xmail, a graphical user interface for mail, was modified to record observations of the user, i.e. to record what actions the user performed on his/her mail messages. Keywords, or features, were extracted from these observations, and used to generate a user-profile." (Payne2 at 4.) "A feature extraction mechanism must be used to identify terms, such as keywords in news articles or mail messages, and map these to attributes for learning." (*Id.* at 3.) "Features were then extracted from incoming mail messages and tested by the classification engine. A

classification was generated, which determined in which mailbox the message should be placed." (*Id. at 4.*)

111. Regarding the browser, "LAW observed the user as he/she browsed the World-Wide Web, using a modified version of the web browser, *Chimera*. These observations were then used to induce the user-profile. (Payne2 at 5.) The profile was used to identify interesting web pages, for example by highlighting links pointing to interesting pages. (*Id.*)

10. U.S. Patent No. 7,631,032 (Refuah)

112. U.S. Patent No. 7,631,032 to Refuah was issued on December 8, 2009 based on a PCT application filed Jan. 28, 1999. I understand that Refuah is accordingly a prior art patent with respect to the asserted patents, which claim priority to a December 1999 provisional application. Refuah discloses a personalization system that uses "personalities" and/or "moods" (dynamic personalities) associated with the user to generate appropriate content: "In one example, a harried access to the Internet (not waiting for images to download, short dwell times) will result in the identification/definition of a rushed mood. Thereafter, search engines may steer the user away from sites which require long download times." (Refuah at 3:3-11.) Personality may be updated automatically: "the mood is updated based on the one or more of the identification of sites visited by a user, the number of site visited, the dwell time at each site, the order in which sites are visited, the contents of the sites, services purchased, information downloaded, actions performed at the sites and/or a predefined or adaptive time-line based function." (*Id.* at 5:34-50.) These tracked variables are compared to a global standard or to a previously acquired baseline for that user to determine the mood. (*Id.*)

113. "[D]epending on a persona, several characteristics of a site may be defined, which may be used in filtering out and/or prioritizing such a site." (Refuah at 7:53 – 8:6.) "[A]n atmosphere of a site may be automatically evaluated by analyzing the content of a site, in

addition to or instead of utilizing a client's reaction to the site or statistics of accessing the site." (*Id.* at 21:6-30.) Sites may also personalize advertisements based on the personality or mood of the incoming user. (*Id.* at 3:56 – 4:4.)

11. U.S. Patent No. 6,567,797 (Schuetze)

114. U.S. Patent No. 6,567,797 to Schuetze et al. was issued on May 20, 2003 and traces priority to a provisional application filed on Jan. 26, 1999. I understand that Schuetze is accordingly a prior art patent with respect to the asserted patents, which claim priority to a December 1999 provisional application. Schuetze discloses "a system and method capable of providing document recommendations to a user based on various users' information browsing and retrieval histories." (Schuetze at 1:29-33.) It scans user logs purchasing information, software usage, and time spent using documents to gain information about the user's preferences. (*Id.* at 11:12-14, 18:11-17.) Feature vectors are derived from the corresponding documents and stored in a database. (*Id.* at 10:14-18, 10:32-39.) Schuetze then computes the "weighted average of the text content of pages that each user has accessed" and stores that as a user vector. (*Id.* at 27:44-64.) Incoming documents are similarly analyzed and compared with the user vector. (*Id.* at 10:58-64.)

12. ProfBuilder and Collecting User Access Patterns for Building User Profiles and Collaborative Filtering (Wasfi)

115. Wasfi (published in January 1999) describes ProfBuilder, a World Wide Web recommender system that "inhabits a Web site and is assigned the goal of being online responsive to the information needs of the site's users." (Wasfi at 58.) I understand that Wasfi is accordingly a prior art publication with respect to the asserted patents, which claim priority to a December 1999 provisional application. Furthermore, ProfBuilder was in use by July 1998, the date on which papers were required to be submitted to the conference, and is thus a prior use with respect to the asserted patents, as it was in use more than a year before December 1999.

116. ProfBuilder "learns user interests and adapts automatically to their changes without user intervention." "ProfBuilder keeps track of each individual user and provides that person online assistance. The assistance includes two lists of recommendations based on two different filtering paradigms: content-based and collaborative." (*Id.* at 60.) "By making content-based filtering, [ProfBuilder] can deal with pages unseen by others." (*Id.*) Pages deemed relevant to the user are represented as keyword vectors and combined to form a user profile vector.[12] (*Id.*)

117. Filtering occurs similarly: pages are translated into their vector space representations using the well-known TFIDF algorithm, and compared to the user profile. (Wasfi at 61.) Matching documents are then presented to the user, along with a number of "balls" that correspond to the relevancy of the document. (*Id.*) ProfBuilder can make recommendations based on collaborative filtering mechanisms as well. (*Id.*)

[12] Keyword vectors are "commonly used in information retrieval (IR) literature." (Wasfi at 58.)

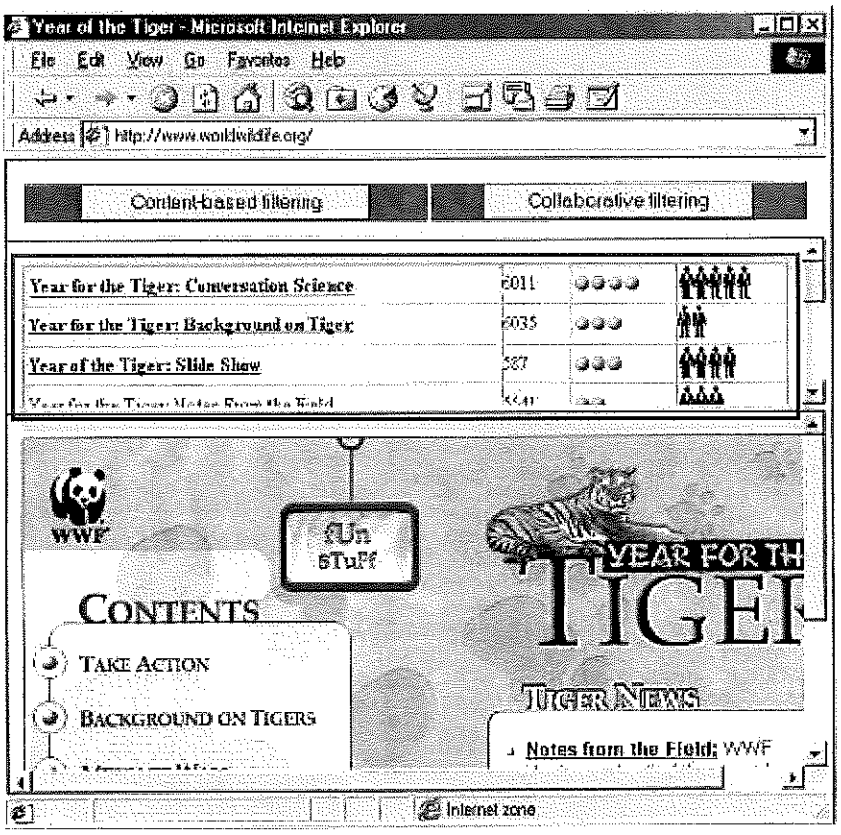

Figure 2 ProfBuilder's interface.

Figure 14: Wasfi at 62, Fig. 2

13. Learning User Profiles for Personalized Information Dissemination (Tan)

118. Tan, published in 1998, describes a personalization system termed PIN which "retrieves on-line news articles from World Wide Web (WWW) and provides customized news to registered users." (Tan at Introduction.) I understand that Tan is accordingly a prior art publication with respect to the asserted patents, which claim priority to a December 1999 provisional application. Tan discloses "incorporat[ing] implicit feedback mechanism" wherein "the order that a user browses through an electronic newspaper and the time that he/she spends on a news article (relative to the length of the article) give indications of his/her interests." (*Id.* at 4.3.2.) Tan "represents a user's interest profile by a set of recognition categories, each associating a set of conjunctive features to a relevance factor." (*Id.* at 3.3.) The text within an

incoming document is matched against a keyword list contained within the user profile. (*Id.* at 4.2.2.) Based on how well the article matches, "PIN's interactive news browser presents personalized news categories and ranked lists of news titles in decreasing order of relevance." (*Id.* at 4.)

14. An Evolutionary Approach to Intelligent Information Filtering (Hofferer)

119. Hofferer, published in 1994, discloses CIFS (Cognitive Information Filtering System), which "distills e-mails from the input stream depending on the user's interests and evaluation judgment which are used to rank e-mail information." (Hofferer at 1.) I understand that Hofferer is accordingly a prior art publication with respect to the asserted patents, which claim priority to a December 1999 provisional application. CIFS transparently monitors user interactions with email, including deleting, forwarding, storing, replying, and printing. (*Id.* at 4.1.) Actions such as storing, forwarding, or replying are deemed positive reactions, while actions such as deleting are deemed negative reactions. (*Id.* at 4.2.) Hofferer then creates email agents that represent the content of the user's emails. (*Id.* at 4.1) Incoming emails are "converted to a word index. . . . [t]he index entries are in the first step matched against the user profile based on the result of the indexation process," then further analyzed "by use of a simple parsing algorithm to detect syntactic constructs (e.g. noun phrase, verb phrase) or special patterns." (*Id.* at 3.)

15. Krakatoa Chronicle, The Krakatoa Chronicle – An Interactive, Personalized Newspaper on the Web (1995) (Kamba); and Personalized, Interactive News on the Web (1997) (Bharat)

120. The Kamba and Bharat articles—published in 1995 and 1997, respectively—describe the Krakatoa Chronicle, "an experimental system which implements an interactive, personalized newspaper on the WWW. (Kamba at 1.) I understand that the articles are accordingly prior art publications with respect to the asserted patents, which claim priority to a

December 1999 provisional application. Furthermore, the Krakatoa Chronicle was in use by 1995. I understand that it is thus a prior use with respect to the asserted patents, as it was in use more than a year before December 1999.

121. "When the user scrolls, peeks at, maximizes, resizes, or saves an article to a scrapbook, the Krakatoa Chronicle increments the user's interest in the article by a corresponding amount, and subsequently changes the personal profile." (*Id.* at 8.) "Each user profile has almost the same format as a document vector. The weight of each keyword represents the system's reckoning of the user's interest in the keyword." (*Id.* at 4.) Prospective news articles are "collected from several news sites daily, and changed into plain text for content-analysis and re-formatting. . . . Then, a document vector is computed for each article" through the well-known TFIDF metric. (*Id.* at 3-4.) The document vector is then compared with the user profile vector, which computes the similarity or "weight" of the article. (*Id.* at 4.) These similarity measurements are then used to create a personalized newspaper for the user. (*Id.* at 5.)

16. Letizia, Letizia: An Agent That Assists Web Browsing (1995) ("Letizia2"), and Autonomous Interface Agents (1997) ("Letizia 3")

122. I understand that the articles, which were published in 1995 and 1997, are prior art publications with respect to the asserted patents, which claim priority to a December 1999 provisional application. Furthermore, Letizia itself was in use by 1995. I understand that Letizia is thus a prior use with respect to the asserted patents, as it was in use more than a year before December 1999.

123. Letizia is an "autonomous interface agent for browsing the Web." (Letizia3 at 1.) "The agent tracks the user's browsing behavior – following links, initiating searches, requests for help – and tries to anticipate what items may be of interest to the user." (Letizia2 at 1.) "Letizia records the URLs chosen by the user and reads the pages to compile a profile of the user's interests." (Letizia3 at 4.) Letizia computes the content of documents using the familiar TFIDF

metric (*id.* at 7), and "uses Netscape's own interface to present its results, using an independent window in which the agent browses pages thought likely to interest the user." (*Id.* at 4.)

VI. THE ASSERTED CLAIMS ARE INVALID AS ANTICIPATED

124. Exhibit 3 of this expert report contain element-by-element claim charts of each of the asserted claims in this case with references to the prior art, and are fully incorporated in their entirety into this report. The charts also list additional references that would render each claim obvious should a finder-of-fact determine that the corresponding element is not present in the prior art reference. Further narrative discussion of these references is below.

A. The Mladenic Paper Anticipates Claims 1, 11, 32 and 34 of the '040 Patent and Claims 1, 5-7, 21, and 22 of the '276 Patent; Personal WebWatcher Anticipates Claims 1, 32 and 34 of the '040 Patent.[13]

1. Background on Mladenic and Personal WebWatcher

125. In the mid-1990's, Dunja Mladenic at Carnegie-Mellon University (CMU) and Jozef Stefan Institute in Slovenia developed a system named Personal WebWatcher (PWW). PWW built on the WebWatcher project that was also at CMU. Indeed, both PWW and WebWatcher were part of a larger project known as the "CMU Text Learning Group." As noted above, PWW is described in the paper entitled *Personal WebWatcher: design and implementation*, by Dunja Mladenic, Technical Report IJS-DP-7472, Department of Intelligent Systems, J. Sefan Institute, Slovenia (1996) ("Mladenic").

126. A key difference between the WebWatcher and PWW projects is that while WebWatcher builds a user model based on data obtained from all users, PWW builds different user models for each user based on training data obtained only from that user. Moreover, while

[13] While Mladenic discloses receiving and processing a query as required by many of the claims, it discloses that element in the context of the predecessor WebWatcher system rather than the Personal WebWatcher system.

WebWatcher is designed to work on a single website, PWW is designed to work on the Internet as a whole.

127. Another distinction is that while WebWatcher requires a user to enter a set of keywords to indicate their interests, PWW requires no explicit action on the part of the user. Rather, PWW simply follows a user as he or she browses the Internet. Thus the user model is built using implicit training data, such as the user's choice of hyperlinks via clicks.

128. PWW also simplifies WebWatcher in some respects. In particular, while WebWatcher evaluates a user's interest in a document via a bag-of-words model of the words in the document itself, PWW focuses on the hyperlink pointing to a document as a representation of that document. In particular, PWW uses the anchor text around the hyperlink as well as other text on the ***referring page***, and builds a bag-of-words user model based on this text. This should be viewed as an engineering simplification that is made for reasons of efficiency. Retrieving a document and analyzing the document at query time would have been judged particularly slow in 1995. To build a system that works effectively at Internet scale it makes sense to evaluate a document based on the hyperlink representation. It is important to note, however, that Mladenic's 1996 paper also discussed the use of a user model based on the document content. This is not surprising given that many earlier projects involving user models, including WebWatcher, were based on analyzing document content.

129. Finally PWW is more closely tied to probabilistic ideas than WebWatcher. In particular, PWW makes use of naïve Bayes models as user models, and also makes use of Shannon mutual information for feature selection. (*See, e.g.,* Mladenic at 5-7). This reflects a general gravitation toward probabilistic ideas in the machine learning field during the 1990s.

2. Mladenic and Personal WebWatcher Anticipate Claim 1 of the '040 Patent

130. Mladenic discloses a computer implemented method for providing automatic, personalized services to a user. Mladenic discloses that "Personal WebWatcher is a system that observes users of the WWW and suggests pages they might be interested in." (Mladenic at 2). Personal WebWatcher is also personalized to a particular user: "Personal WebWatcher (PWW) is structured to specialize for a particular user, modeling her/his interests." (Mladenic at 3). *See also* Mladenic at 8, figure 2:

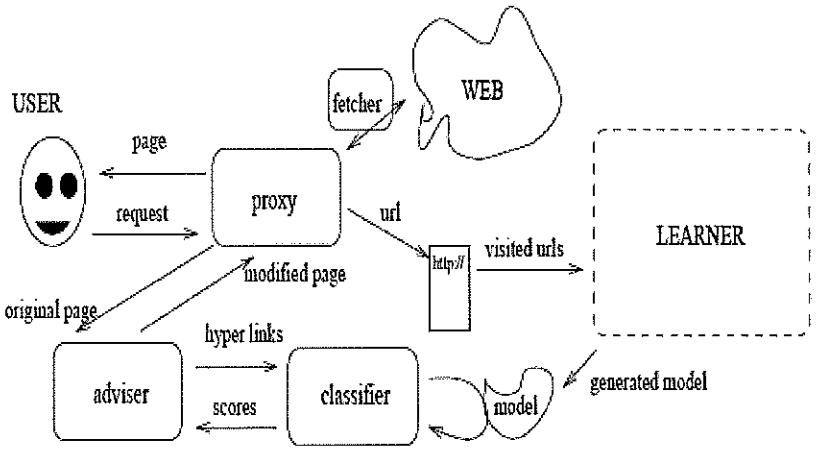

Figure 2: Structure of Personal WebWatcher. The learning part is described separately

Figure 15: Mladenic at 8, Fig. 2.

(a) Transparently monitoring user interactions with data while the user is engaged in normal use of a computer

131. As disclosed in Mladenic, "Personal WebWatcher is a system that observes users of the WWW and suggests pages they might be interested in. It learns user interests from the pages requested by the user." (Mladenic at 2.) PWW "'watches over the user's shoulder' [...], but it avoids involving the user in its learning process (it doesn't ask the user for any keywords or opinions about pages)." (*Id.* at 3). Rather, it "solely records the addresses of pages requested by the user and highlights hyperlinks that it believes will be of interest." Accordingly, Mladenic and PWW use *implicit* user feedback rather than explicit feedback such as ratings, thereby

meeting the requirement that the user be transparently monitored during normal use of the computer. (*See, e.g.*, 2:23-28.)

(b) Updating user-specific data-files comprising the monitored user interactions and a set of documents associated with a user.

132. Mladenic also discloses updating user-specific data files that comprise both the monitored user interactions and a set of documents associated with a user. More specifically, Personal WebWatcher includes both a "Learner" module for learning a new user model from scratch, and an "Updater" module for updating an existing module. (Mladenic at 8.) "Both versions fetch visited documents and documents one step behind the hyperlinks of visited documents and store them as positive or negative examples of user interests, depending whether the user visited the document or not." (*Id.*) "Hyperlinks whose documents were visited by the user are considered to be positive examples, and all other to be negative examples of the user interests. The idea is that all hyperlinks were presented to the user and the user chose to visit some of them that meet her/his interests." (*Id.*)

133. Accordingly, Mladenic and PWW disclose updating a set of documents associated with the user, as the "Updater" modules fetches documents visited by the user as well as documents one step behind those visited documents. Mladenic and PWW also disclose updating monitored user interactions with the documents—specifically, whether or not the user visited the document.

(c) Estimating parameters of a learning machine

134. Mladenic discloses estimating parameters of a learning machine, wherein the parameters define a User Model specific to the user and wherein the parameters are estimated in part from the user specific data files. Mladenic's work belonged to a long tradition of user modeling, a tradition which included the predecessor WebWatcher, and her work distinguished itself from WebWatcher via its focus on the utilization of personal data in forming a user model

specific to that person. "Unlike WebWatcher, Personal WebWatcher (PWW) is structured to specialize for a particular user, modeling his/her interests." (Mladenic at 3).

135. More specifically, PWW is described as using a "bag-of-words" representation using frequencies of the words. In other words, "all the words from the document are taken and no ordering of words or any structure of text is used" (*id.* at 3), as disclosed in Figure 12 above. Documents are accordingly represented by a count of the various words within the document—one of the embodiments disclosed in the specification: "One embodiment of the informative measure is a word probability distribution P(w|u) representing the interest of a user u in a word or phrase w, as measured by the word's frequency in user documents." (11:1-4.) *See also* Figure 7 above.

136. Mladenic further discloses typical IR techniques such as eliminating stop words and stemming to reduce the number of words present within the document representation. (Mladenic at 5.) It further details the "well-established" Term Frequency / Inverse Document Frequency document representation (TFIDF), which calculates vector components using the product of the Term Frequency—the number of times the word occurred in the document—and the Inverse Document Frequency, which is based on the fraction of documents in which the word occurred once. (*Id.* at 6.) Weighting document vectors using TFIDF is another disclosed embodiment of the patents:

> A preferred embodiment uses the TFIDF measure, described in Ricardo Baeza-Yates and Berthier Ribeiro-Neto, Modern Information Retrieval, Addison Wesley, 1999, in which TF stands for term frequency, and IDF stands for inverse document frequency. Mathematically, if $f_{u,w}$ denotes the frequency of the word w in user u documents, and D_w denotes the number of documents containing the word w, then the importance of a word w to a user u is proportional to the product $f_{u,w} \cdot D/D_w$.

(11:12-20.)

137. Mladenic and PWW further disclose combining the document representations of the user's documents to estimate the parameters of a learning machine:

> The model of user interests is generated "off-line," usually during the night and thus its generation is not so critical in time as its usage for prediction. One of the simplest idea for learning is to use hyperlinks that occurred on the documents presented to the user as training examples and learn to predict if a new hyperlink is positive or negative example of the user interests....But during the learning phase we can afford using more time than when adding advice, so why not retrieving document behind hyperlinks, instead of using the extended hyperlink representation? In that case, we can learn the model of user interests directly from documents whose interestingness we are trying to predict.

(Mladenic at 10-11.)

138. Mladenic and PWW accordingly disclose extracting keywords from each user document, assigning a weight to that keyword for that document, and combining the resultant document representations into a user profile. This is the same method the asserted patents use to create the "Informative Word/Phrase List" portion of the User Model. (6:13-14, 10:52 – 11:31, Fig. 4A.)

(d) <u>Analyzing a document</u>

139. Mladenic discloses analyzing a document to identify properties of the document. As detailed above, Mladenic discloses some of the standard document analysis techniques and also discloses the implementation of two variants of these techniques with PWW. The basic document representation was a bag-of-words representation of anchor text:

> We decided to use the bag-of-words representation using frequency of word and observe success of given advice (whether user selected the advised hyperlink). In case of poor system performance, some additional information from HTML-structure could be added, for example, frequency of word in headlines of a given document.

(Mladenic at 4). Mladenic also disclosed methodology that makes use of the words in the document pointed to by the hyperlink. (*Id.*).

140. During the application phase, PWW is "actually predicting interestingness of document based on the hyperlink pointing to it, and not document itself" because "retrieving documents behind the requested hyperlinks is usually time consuming." (Mladenic at 10.) Links to a document are often viewed as a property of the document itself, and indeed the asserted patents take this view:

> The probability is estimated by analyzing properties of the document and applying them to the learning machine. ***Documents of multiple distinct media types of analyzed, and identified properties include***: the probability that the document is of interest to users who are interested in particular topics, a topic classifier probability distribution, a product model probability distribution, product feature values extracted from the document, the document author, the document age, ***a list of documents linked to the document***, the document language, number of users who have accessed the document, number of users who have saved the document in a favorite document list, and a list of users previously interested in the document.

('040 Patent, 4:35-47.) This requirement is further an element of unasserted claim 7 of the '040 patent, which depends from claim 1:

> 7. The method of claim 1 ***wherein the identified properties of the document d comprise a user u-independent property selected from the group consisting of***:
>
> a) a probability P(t,d) that the document d is of interest to users interested in a topic t;
> b) a topic classifier discrete probability distribution P(t|d);
> c) a product model discrete probability distribution P(p|d);
> d) product feature values extracted from the document d;
> e) an author of the document d;
> f) an age of the document d;
> ***g) a list of documents linked to the document d;***
> h) a language of the document d;
> i) a number of users who have accessed the document d;
> j) a number of users who have saved the document d in a favorite document list; and
> k) a list of users previously interested in the document d.

141. Accordingly, analyzing the links *to* the document during the application phase, as PWW does, meets the limitation of "analyzing a document to identify properties of the document."

142. Mladenic (but not PWW) further discloses analyzing the document itself, rather than links to the document. In describing the WebWatcher system, Mladenic states that "Some later versions of the WebWatcher system change slightly the way of constructing text for learning, e.g., adding words in the document retrieved behind hyperlink." (Mladenic at 4.) In addition, the WebWatcher system had the ability to add *new* links to the presented webpage, an ability PWW did not share: "WebWatcher highlights related hyperlinks on the current page and/or adds new hyperlinks to the current page." (*Id.* at 2.) Since the links did not previously exist, WebWatcher could not have relied on anchor text within the current webpage to make those recommendations, indicating that it must have analyzed the content of the suggested page. While neither feature was present within PWW, both were disclosed in the Mladenic paper itself.

(e) Estimating a probability that an unseen document is of interest to the user

143. Mladenic discloses estimating the probability that an unseen document is of interest to the user. Her work is explicitly based on the naïve Bayes model, although she also investigates other approaches to classification of a document: "We decided to test different learning algorithms on PWW data (see Section 5), since it is not clear which algorithm is the most appropriate. The current version of PWW uses a Naive (Simple) Bayesian classifier on frequency vectors to generate a model of user interests, that is used for advising hyperlinks." (Mladenic at 7.)

144. Mladenic discusses two kinds of learned models based on the naïve Bayes formulation. The first yields the probability of a link being interesting (or not) given the link itself. (Mladenic at 10). The second estimates the probability of a link being interesting (or not) given the document pointed to by the link. (Mladenic at 11.) A Naive Bayes classifier returns a posterior probability, obtained from the prior probability of a document being of interest to the user and the likelihood of the link (in the first model) or the document (in the second model):

UserId and data source	probability of interestingness	number of examples	data entropy
usr150101			
Doc	0.094	1 333	0.449
HL	0.104	2 528	0.480
usr150202			
Doc	0.107	3 415	0.492
HL	0.053	1 798	0.301
usr150211			
Doc	0.089	2 038	0.436
HL	0.011	2 221	0.259
usr150502			
Doc	0.100	1 272	0.468
HL	0.100	2 498	0.468

Table 2: Data characteristics for document (Doc) and hyperlink (HL) data for each of the four HomeNet users.

Figure 16: Mladenic at 12.

145. Furthermore, Mladenic's probability estimate functions on unseen documents. As detailed above, Mladenic analyzes the *content* of the document using a "bag of words" implementation. This content analysis is independent of whether or not the user viewed the document. In fact, Mladenic explicitly discloses analyzing documents that the user did *not* view: "Both versions [of the document analyzer used in PWW] fetch visited documents and documents one step behind the hyperlinks of visited documents and store them as positive ***or negative*** examples of user interests, depending whether the user visited the document ***or not.***" (Mladenic at 8 (emphasis added)). Moreover, a content-based recommendation system that does not exclude unseen documents has the inherent capability to recommend unseen documents.

146. Accordingly, Mladenic and PWW disclose estimating the probability that an unseen document is of interest to the user.

(f) Providing automatic, personalized information services to the user.

147. Mladenic discloses using the estimated probability to provide automatic, personalized information services to the user:

> A limited number of hyperlinks that are scored above some threshold are recommended to the user, indicating their scores with graphical symbols placed around each advised hyperlinks. For example, in Figure 3 three hyperlinks are suggested by PWW: "Machine Learning Information Services" and two project members (Dayne Freitag, Thorsten Joachims).

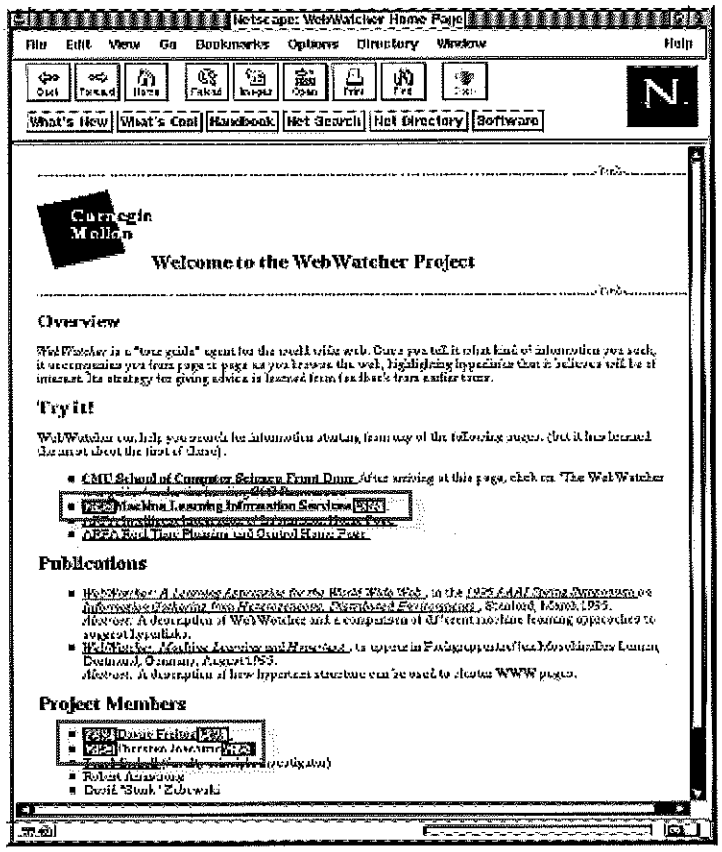

Figure 3: Example of HTML-page presented to the user by PWW.

Figure 17: Mladenic at 9, Fig. 3 (red notations added).

(Mladenic at 7-8.) Of note, specifying or highlighting links to documents that may be of interest to the user is one of the claimed embodiments of the patents:

> The personal links application is illustrated in FIG. 20. In this application, the ***hyperlinks in a document being viewed by the user are graphically altered, e.g.,***

> ***in their color, to indicate the degree of interest of the linked documents to the use.*** As a user views a document (step 210), the document is parsed and analyzed (212) to locate hyperlinks to other documents. The linked documents are located in step 214 (but not shown to the user), and evaluated with the User Model (214) to estimate the user's interest in each of the linked documents. In step 216, the graphical representation of the linked documents is altered in accordance with the score computed with the User Model. ***For example, the links may be color coded, with red links being most interesting and blue links being least interesting, changed in size, with large links being most interesting, or changed in transparency, with uninteresting links being faded.*** If the user follows one of the interesting links (218), then the process is repeated for the newly viewed document (210).

(29:41-60; *see also* Fig. 20.) Accordingly, Mladenic and PWW disclose this claim element.

3. Mladenic Anticipates Claim 11 of the '040 Patent

148. Claim 11 requires "The method of claim 1 further comprising estimating a posterior probability P(u|d,q) that the document d is of interest to the user u, given a query q submitted by the user." Mladenic discloses accepting a query from the user and using that query to determine whether the document is of interest to the user:

> WebWatcher can be described as an agent that assists users in locating information on the WWW. It learns by observing a user on her/his way through the WWW and suggests interesting hyperlinks whenever it is confident enough. The idea is that the user provides a few keywords describing a search goal and WebWatcher highlights related hyperlinks on the current page and/or adds new hyperlinks to the current page.

(Mladenic at 2.) While this feature is not present within the PWW system, it ***is*** present in the predecessor WebWatcher system described by the Mladenic reference itself.

4. Mladenic and Personal WebWatcher Anticipate Claim 32 of the '040 Patent

149. Claim 32 of the '040 patent requires "A program storage device accessible by a central computer, tangibly embodying a program of instructions executable by the central computer to perform method steps for providing automatic, personalized information services to a user u," followed by the same method steps required in Claim 1 of the '040 patent. As detailed above, Mladenic discloses all method steps required in claim 1 and thus it discloses all method

steps required in claim 32. Mladenic further discloses "a program of instructions executable by the central computer" to perform those method steps. *See, e.g.,* Mladenic at 7 (disclosing that Personal WebWatcher is implemented in the Perl and C++ programming languages). PWW is also stored on a central computer, specifically a proxy server. (*Id.* at 7-8.) Accordingly, Mladenic and PWW meet all limitations of claim 32 of the '040 Patent.

5. Mladenic and Personal WebWatcher Anticipate Claim 34 of the '040 Patent

150. Claim 34 requires "The program storage device of claim 32 wherein analyzing the document d provides for the analysis of documents having multiple distinct media types." Mladenic discloses processing HTML documents: "The model of user interests is designed to predict if some document is positive or negative example of user interests. It is used to advice hyperlinks on ***the HTML document requested by the user***." (Mladenic at 10.) HTML documents contain multiple media types, including text, images, and video. By processing HTML documents, Mladenic "provides for the analysis of documents having multiple distinct media types." (*See, e.g.,* 9:14-17.)

6. Mladenic Anticipates Claim 1 of the '276 Patent

151. Claim 1 of the '276 patent is substantially similar to claim 1 of the '040, as described in Section IV.E above. It contains three additional limitations: a) requiring that the monitoring occur during normal use of a browser rather than normal use of a computer, b) requiring that the method receive a search query from the user, and c) requiring that the documents analyzed be documents retrieved in response to the search query.

(a) Normal use of a browser program

152. Mladenic discloses monitoring the user during normal use of a browser program. "Personal WebWatcher is a system that observes users of the WWW and suggests pages they might be interested in. It learns user interests from the pages requested by the user." (Mladenic

at 2.) PWW "'watches over the user's shoulder' [..], but it avoids involving the user in its learning process (it doesn't ask the user for any keywords or opinions about pages)." (*Id.* at 3). Rather, it "solely records the addresses of pages requested by the user and highlights hyperlinks that it believes will be of interest." "Personal WebWatcher consists of two main parts: a ***proxy server that interacts with the user via Web browser*** and a learner that provides the user-model to the server." (*Id.* at 7.) "Proxy waits in an infinite loop ***for a page request from the browser***." (*Id.*; *see also* Fig. 2.) Accordingly, Mladenic and PWW disclose this limitation.

(b) Receiving a search query from the user

153. Mladenic discloses receiving a search query from the user. As detailed in Section VI.A.3, above, Mladenic discloses accepting a query from the user and using that query to determine whether the document is of interest to the user:

> WebWatcher can be described as an agent that assists users in locating information on the WWW. It learns by observing a user on her/his way through the WWW and suggests interesting hyperlinks whenever it is confident enough. The idea is that the user provides a few keywords describing a search goal and WebWatcher highlights related hyperlinks on the current page and/or adds new hyperlinks to the current page.

(Mladenic at 2.) While this feature is not present within the PWW system, it ***is*** present in the predecessor WebWatcher system described by the Mladenic reference itself. Accordingly, Mladenic discloses receiving a search query from the user.

(c) Retrieving a plurality of documents based on the search query

154. Mladenic also discloses retrieving a plurality of documents based on the search query. As detailed above, the predecessor WebWatcher system can also "add[] new hyperlinks to the current page" based on the few keywords entered by the user. Accordingly, those documents are retrieved and analyzed based on the search query entered by the user.

7. Mladenic Anticipates Claim 5 of the '276 Patent

155. Claim 5 requires "The method of claim 1, further comprising analyzing the monitored data to determine documents not of interest to the user, and wherein estimating parameters of a user-specific learning machine further comprises estimating parameters of a user-specific learning machine based at least in part on the documents not of interest to the user." Mladenic and PWW disclose analyzing both documents that the user visited and documents that the user did ***not*** visit. (Mladenic at 8.) "Hyperlinks whose documents were visited by the user are considered to be positive examples, and all the other to be negative examples of the user interest." (*Id.*) Accordingly, Mladenic discloses this limitation.

8. Mladenic Anticipates Claim 6 of the '276 Patent

156. Claim 6 requires "The method of claim 1, wherein monitoring user interactions with data for a document comprises monitoring at least one type of data selected from the group consisting of information about the document, ***whether the user viewed the document***, information about the user's interaction with the document, context information, the user's degree of interest in the document, time spent by the user viewing the document, whether the user followed at least one link contained in the document, and a number of links in the document followed by the user." Mladenic and PWW disclose at least monitoring the user interactions for whether the user viewed the document: "Hyperlinks whose documents were visited by the user are considered to be positive examples, and all the other to be negative examples of the user interest. The idea is that all hyperlinks were presented to the user and the user chose to visit some of them that meet her/his needs." (Mladenic at 8.) Accordingly, Mladenic discloses this limitation.

9. Mladenic Anticipates Claim 7 of the '276 Patent

157. Claim 7 requires "7. The method of claim 1, wherein said plurality of retrieved documents correspond to a respective plurality of products." I understand that PUM contends that this limitation is met by Google's AdWords system, which displays advertisements that ***may*** correspond to products. (Plaintiff's Fourth Supplemental Infringement Contentions, Attachment B at 22-23.) For example, a Google AdWords search for "cancer donations" returned a number of advertisements that do not correspond to products:

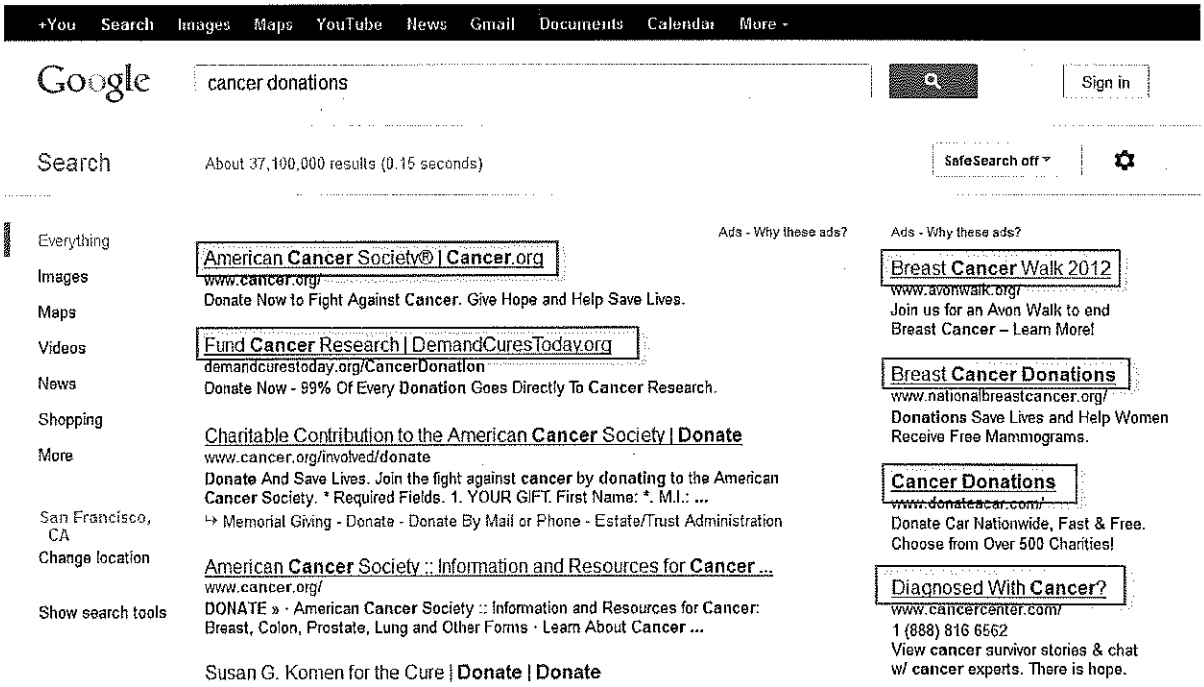

Figure 18: Google Search results page

158. Thus, PUM apparently interprets this claim element as requiring only that the documents ***may*** correspond to products.[14] Applying PUM's interpretation, Mladenic meets this limitation. As any document on the Internet may correspond to a product, and as Mladenic and PWW make no attempt to distinguish pages that correspond to products from pages that do not correspond to products, any given set of retrieved documents in Mladenic/PWW may also

[14] I take no position on whether an advertisement can be a "document," as PUM asserts.

correspond to products. Accordingly, Mladenic discloses this claim element under PUM's infringement theories.

10. Mladenic Anticipates Claim 21 of the '276 Patent

159. Claim 21 requires "The method of claim 1, wherein using the estimated probabilities for the respective plurality of retrieved documents to present at least a portion of the retrieved documents to the user comprises presenting to the user at least said portion of the retrieved documents based on the estimated probability that the retrieved document is of interest to the user and the relevance of the retrieved document to the search query." Mladenic discloses presenting document based both on the query and on the computation of whether the document is of interest to the user:

> WebWatcher can be described as an agent that assists users in locating information on the WWW. It learns by observing a user on her/his way through the WWW and suggests interesting hyperlinks ***whenever it is confident enough***. The idea is that ***the user provides a few keywords*** describing a search goal and WebWatcher highlights related hyperlinks on the current page ***and/or adds new hyperlinks to the current page***.

(Mladenic at 2.) Accordingly, Mladenic suggests pages based both on the user's keywords (query) and on its computed confidence (probability) that the pages will be of interest to the user.

11. Mladenic Anticipates Claim 22 of the '276 Patent

160. Claim 22 requires "The method of claim 1, wherein identifying properties of the retrieved document comprises identifying properties selected from the properties consisting of ***a topic associated with the retrieved document***, at least one product feature extracted from the retrieved document, an author of the retrieved document, an age of the retrieved document, ***a list of documents linked to the retrieved document***, a number of users who have accessed the retrieved document, and a number of users who have saved the retrieved document in a favorite document list." Mladenic and PWW disclose at least identifying a list of documents linked to the retrieved document. *See, e.g.,* Mladenic at 2 ("[Personal WebWatcher] can also suggest

pages related to the current page using information stored in the structure of hypertext without considering the text itself"); Mladenic at 10 ("Since the prediction should be performed while the user is waiting for an HTML-document, we are actually predicting interestingness of [sic] document based on the hyperlink pointing to it, and not the document itself... [b]ut during the learning phase we can afford using more time than when adding advice, so why not retrieving [sic] documents behind hyperlinks, instead of using the extended hyperlink representation?"). Mladenic further discloses analyzing a topic associated with the document: "In case of poor system performance, some additional information from HTML-structure could be added, for example, ***frequency of word in headlines of a given document***." (Mladenic at 4.) Accordingly, Mladenic discloses this limitation.

B. The Autonomy Agentware System Anticipates Claims 1, 11, 32 and 34 of the '040 Patent and Claims 1, 3, 7, 21, and 22 of the '276 Patent.

1. Background on Autonomy

161. Autonomy's technology is a comprehensive solution for adaptive, dynamic personalized delivery of information. Autonomy uses machine learning technology such as Bayesian statistical inference and neural networks to analyze the digital content by extracting concepts and categorizing documents. Autonomy also uses this technology to model the users by learning their preferences and by identifying the most relevant information for each user. The system learns the users' preferences and information needs dynamically and over time, further adjusting as users change their behavior.

2. Autonomy Agentware Anticipates Claim 1 of the '040 Patent

162. Autonomy discloses a computer-implemented method for providing automatic, personalized services to a user. As described in the press release, Agentware 1.1 is "the first-ever product that delivers personalized information from the Internet by learning the preferences and needs of the user." (Autonomy PR at 1.) Further, "Agentware 1.1 requires a 486DX or

microprocessor running Windows 95 or 3.11." (*Id.* at 2.) The Agentware User Guide similarly lists hardware requirements as well as a process for installing the program onto the user's computer.

(a) Transparently monitoring user interactions with data while the user is engaged in normal use of a computer

163. As the press release states, Agentware "automatically learns user preferences to deliver personalized information on demand." (Autonomy PR at 1.) "The Agent keeps learning about your user's interests and will adapt accordingly as your interests change." (*Id.*) As the white paper explains:

> Many of Autonomy's products rely on user interest or employee expertise profiles created implicitly, simply by observing a user's actions.
>
> ....
>
> Online publishers and Intranet developers can maintain sets of agents representing an individual's interests to offer a number of personalized services. Because these services are based on a user's actual interests, they do not require him to fill out lengthy questionnaires or rate his likes and dislikes. There profiles can be kept completely anonymous and do not require the user to provide any private demographic information.
>
> ...
>
> As an individual reads additional articles online, publishes material on the corporate intranet or submits documents to the knowledge management system, the Autonomy system updates his agents by recalculating interests levels in the different ideas. Concepts that once occurred frequently but no longer are important are replaced over time. In this way, the Autonomy system keeps pace with an individual's changing interests. This is in contrast to an explicit preference setting, which users must remember to adjust as their interests evolve.

(Autonomy WP at 3.)

164. Autonomy thus mentions several motivations for transparent monitoring. It is convenient for a user, as it does not "require him to fill out lengthy questionnaires or rate his likes and dislikes." Since it "keeps pace with an individual's changing interests" rather than requiring the users to "remember to adjust as their interests evolve," it is better able to track

users' interests over time. Finally, by tracking the user's actions, Autonomy is able to build a profile based on actual user interests as opposed to claimed interests.

(b) Updating user-specific data-files comprising the monitored user interactions and a set of documents associated with a user.

165. Autonomy Agentware discloses updating user-specific data files comprising the user's interactions with a set of documents associated with the user. Users may "train" their autonomy agents using previously recorded documents. (Autonomy UG at 8.) After the agent has retrieved a set of documents, those documents may be used to retrain the agent to better locate desired information. (*Id.* at 11, 19.) As the white paper states:

> To determine an individual's topical interests or expertise, Autonomy's software applies the pattern-matching technology to extract key ideas from the articles a user reads online. As the user is served documents, Autonomy software automatically creates a set of *Concept Agents*. By weighting the frequency with which certain topics occur, the server can then encode a set of interests into Autonomy *Concept Agents*.

(Autonomy WP at 3.)

(c) Estimating parameters of a learning machine

166. Autonomy Agentware also discloses estimating parameters of a learning machine, wherein the parameters define a User Model specific to the user and wherein the parameters are estimated in part from the user specific data files. Autonomy uses neural networks as its specific instantiation of a parameterized learning machine. Autonomy uses the term "user profiling" rather than "user modeling," but the two terms are conceptually identical; moreover, the same technologies are often used to address what some researchers call profiling and others call modeling.[15] The core idea is a system that presents a different set of objects to each user, where this differentiation is based on user-specific data. This is precisely what Autonomy does. Indeed, the user profiling/modeling is a key component of their technology.

[15] Other terms often used are "personalization," "recommendation systems," and "user parameterization."

167. In particular, Autonomy models the users with several actions: 1) it analyses all the articles that the user reads (to extract concepts that are of interest), 2) it tracks changes in the user's needs and interests by continuously tracking the reading activities and updating the models, 3) it tracks web activities (browsing, searches etc.), 4) it includes, when available, any known demographic information and 5) it tracks transactions and product and brand preferences.

168. For example, Autonomy Agents "can use the best documents [retrieved from a search] to expand [the] Agent's training. (Autonomy UG at 11.) "The major difference from your initial training is that a webpage holds a lot more text, and so gives a more rounded view of the subject." (*Id.*) "Autonomy employs advanced pattern matching technology to extract a document's digital essence and determine the characteristics that give the text meaning. (Autonomy WP at 1.) "Once Autonomy's technology has identified and encoded the unique 'signature' of the key concepts, Concept Agents are created to seek out similar ideas in websites, news feeds, email archives and other documents." (*Id.*) These concept agents store both the relationships and weights associated with the concepts within the documents. (*Id.* at 2.)

(d) <u>Analyzing a document</u>

169. The analysis of documents is central to Autonomy's technologies. One of Autonomy's main tasks is to analyze and manage unstructured digital information, e.g., documents such as "word processing and HTML-based files, email messages and electronic news feeds." (Autonomy WP at 1.) As stated in the User Guide,

> Once Autonomy has identified this "pattern" in the training text, it can compare it to the documents it finds on the Internet. Autonomy does not use keyword searches, but actually identifies the concepts involved in the text and compares them. It then assigns a relevance to the document depending on how closely it matches the pattern established by the training.

(Autonomy UG at 4.) The white paper further explains that

> Autonomy's architecture combines innovative high-performance pattern-matching algorithms with sophisticated contextual analysis and concept extraction

> to automate the categorization and cross-referencing of information, improve the efficiency of information retrieval and enable the dynamic personalization of digital content.
>
> Autonomy's strength lies in its high-performance pattern matching algorithms. These algorithms are informed by Claude Shannon's principles of information theory, Bayesian probabilities, and the latest research in neural networks. This technique enables Autonomy's system to identify patterns in text and look for similar patterns in other sources, quickly and automatically. Most importantly, the technology can analyze a text and identify the key concepts within the document because it understands how the frequency and relationships of terms correlate with meaning.

(Autonomy WP at 1.)

170. The goals of such analysis are multifold:

1) to "analyze a text and identify the key concepts within the documents";

2) to return a representation for each of these concepts, *i.e.* a compact way of describing them;

3) to find related documents, *i.e.* given a document, Autonomy finds "references to documents in another source of text with the highest degree of relevance" (Autonomy WP at 1);

4) to find the users' interests by analyzing the articles the user reads and by extracting the main concepts, as described above;

5) to automatically sort documents into predefined categories;

171. These goals are the core of Autonomy's technology and all involve and indeed require analyzing a document to identify properties of that document.

(e) <u>Estimating a probability that an unseen document is of interest to the user</u>

172. Autonomy Agentware discloses estimating the probability that an unseen document is of interest to the user. As indicated in the quote above, Autonomy makes use of Bayesian theory—which has probability computations at its core—to calculate the probabilities of many variables and their relationships. In a Bayesian system, the variables of interest are defined (documents, topics, users…), the data is observed (e.g., user A reads document D…) and finally the posterior probabilities are calculated.

173. In the case of Autonomy, some of the probabilities of interests include:

- the probability of a topic being present in a document, P(topic| document)
- the probability of a topic being relevant to a user, P(topic| user)
- the probability of a product being relevant to a user P(product | user)
- the probability of a document being relevant to a user P(document | user)

174. As the white paper states,

> The theoretical underpinnings for Autonomy's approach can be traced to Thomas Bayes, an 18th century English clerk whose works on mathematical probability were not published until after his death. Bayes' work centered on calculating the probabilistic relationship between multiple variables and determining the extent to which one variable impacts another. Although no one knows for certain what Bayes' original goal was, Bayes' Theory has become a central tenet of modern statistical probability modeling. By applying contemporary computational power to the concepts pioneered by Bayes, it is now feasible to calculate the relationships between many variables quickly and efficiently, allowing software to manipulate concepts.

(Autonomy WP at 4.) Of note, the patents state that "[i]t is an additional object of the present invention to provide a method based on Bayesian statistics that updates the user profile" (4:8-10) and that "[t]he underlying mathematical framework of the modeling and training algorithms discussed [in the patent] is based on Bayesian statistics." (8:35-37.)

175. Furthermore, because Autonomy relies on textual and conceptual analysis of the document in question, it is able to estimate the probability of unseen documents as well as seen documents. Unlike collaborative filtering techniques, Autonomy does not require that a document be viewed by a user in the past. Indeed, the white paper points out this very drawback to collaborative filtering systems. (Autonomy WP at 6, describing the "day one" problem.) At least at some point, Autonomy likely analyzed a document that the current user had not previously seen. Under Plaintiff's own infringement allegations, the *likelihood* that an unseen document was presented is sufficient to meet this limitation. (Plaintiff's April 2012 Infringement Contentions, Tab A, p. 18: "Due to the extremely large number candidate search results in

Google's indexes, and due to the frequent changes or updates made to the candidate search results, it is likely that candidate search results have not been previously seen by the user.") (emphasis added). Moreover, a content-based recommendation system that does not exclude unseen documents has the inherent capability to recommend unseen documents.

176. Hence, Autonomy anticipates the claim element of estimating a probability to determine whether a document is of interest to the user, in that it uses the Bayesian inference to estimate not only this very specific probability but many others. Bayesian theory allows for any "query" on the variables and their relationships, once the variables have been defined and some values observed. Accordingly, Autonomy discloses estimating the probability that an unseen document is of interest to the user.

(f) Providing automatic, personalized information services to the user.

177. Autonomy Agentware uses the estimated probability to provide personalized information services to the user. As described in the press release, Agentware 1.1 is "the first-ever product that delivers personalized information from the Internet by learning the preferences and needs of the user." (Autonomy PR at 1.) "Autonomy can be used to search for information on the World Wide Web, prepare a specially edited newspaper, answer questions on a subject and find other agents with similar interests." (Autonomy UG at 4.) Accordingly, Autonomy meets this limitation.

3. Autonomy Agentware Anticipates Claim 11 of the '040 Patent

178. Claim 11 requires "The method of claim 1 further comprising estimating a posterior probability P(u|d,q) that the document d is of interest to the user u, given a query q submitted by the user." Autonomy discloses accepting a query from the user and using that query to determine whether the document is of interest to the user:

> When you train an Agent you give it a few sentences of text describing the subject you want it to look for. The neural network is able to identify the key concepts in the text, and then use its knowledge of language to decide their relative importance.

(Autonomy UG at 4.) *See also* Autonomy WP at 3 (explaining that agents can be trained on user documents and/or user queries.)

4. Autonomy Agentware Anticipates Claim 32 of the '040 Patent

179. Claim 32 of the '040 patent requires "A program storage device accessible by a central computer, tangibly embodying a program of instructions executable by the central computer to perform method steps for providing automatic, personalized information services to a user u," followed by the same method steps required in Claim 1 of the '040 patent. As detailed above, Autonomy discloses all method steps required in claim 1 and thus it discloses all method steps required in claim 32. Autonomy further discloses "a program of instructions executable by the central computer" to perform those method steps. *See, e.g.,* Autonomy UG at 3 (describing how to install the Agentware software on a user's computer). Autonomy software may also reside on a central computer such as a server. (Autonomy WP at 7-11.) Accordingly, Autonomy meets all limitations of claim 32 of the '040 Patent.

5. Autonomy Agentware Anticipates Claim 34 of the '040 Patent

180. Claim 34 requires "The program storage device of claim 32 wherein analyzing the document d provides for the analysis of documents having multiple distinct media types." Autonomy discloses processing HTML documents: "The model of user interests is designed to predict if some document is positive or negative example of user interests. It is used to advice hyperlinks on ***the HTML document requested by the user***." HTML documents contain multiple media types, including text, images, and video. By processing HTML documents, Autonomy "provides for the analysis of documents having multiple distinct media types." (*See, e.g.,* 9:14-17.) Further, Autonomy explicitly allows for the examination of images as well as text.

(Autonomy UG at 17 (allowing Agentware to retrieve images from a web page), 27 (judging the relevance of an image).)

181. The Autonomy white paper further indicates that the Commerce Server "can handle multimedia formats, including video and audio," that the Knowledge Server "automatically presents a unified view of disparate data sources across the enterprise – including email messages, word processing files, PowerPoint presentations, Excel spreadsheets, PDF files, Lotus Notes archives, intranet file servers, SQL/ODBC databases, live chat/IRC, newsfeed, and the expertise profiles of other employees. (Visualizer module)," and that the Knowledge Update "[m]onitors hundreds of specified Internet and intranet sites, news feeds and internal repositories containing Lotus Notes, HTML, word processing files, PDF files, and many others." (Autonomy WP at 7-8, 10.)

6. Autonomy Agentware Anticipates Claim 1 of the '276 Patent

182. Claim 1 of the '276 patent is substantially similar to claim 1 of the '040, as described in Section IV.E above. It contains three additional limitations: a) requiring that the user be normally using a browser in particular rather than a computer in general, b) requiring that the method receive a search query from the user, and c) requiring that the documents analyzed be documents retrieved in response to the search query.

(a) Normal use of a browser program

183. Autonomy discloses monitoring the user during normal use of a browser program:

> When you want to read the paper that your agents have compiled for you, just click on "Read Paper" or select Read Paper from the Paper menu.... Your paper is displayed on your web browser. It will have its name, date and the front page of each section. Each section contains a selection of the information retrieved. Click on the Index to obtain a list of the articles retrieved and a summary of each, clicking on the article to read it.

(Autonomy UG at 13.) *See also* Autonomy WP at 3 (monitoring the user's interactions with online articles), 9 (describing the User Profiling feature which monitors the activities of all web

site visitors), 10 (describing the Customer Profiling feature which does the same thing with product pages).

(b) Receiving a search query from the user

184. Autonomy Agentware discloses receiving a search query from the user. As detailed in Section VI.B.3, above, Autonomy discloses accepting a query from the user and using that query to determine whether the document is of interest to the user:

> When you train an Agent you give it a few sentences of text describing the subject you want it to look for. The neural network is able to identify the key concepts in the text, and then use its knowledge of language to decide their relative importance.

(Autonomy UG at 4.) *See also* Autonomy WP at 3 (explaining that agents can be trained on user documents and/or user queries).

(c) Retrieving a plurality of documents based on the search query

185. Autonomy Agentware also discloses retrieving a plurality of documents based on the search query. *See* above (disclosing retrieving documents based on the user's query).

7. Autonomy Agentware Anticipates Claim 3 of the '276 Patent

186. Claim 3 requires "The method of claim 1, wherein transparently monitoring user interactions with data comprises monitoring user interactions with data during multiple different modes of user interaction with network data." Autonomy discloses managing unstructured digital information including "word processing and HTML-based files, email messages, and electronic news feeds." (Autonomy WP at 1.) Further, the Knowledge Server deals with information for "email messages, word processing files, PowerPoint presentations, Excel Spreadsheets, PDF files, Lotus Notes archives, intranet file servers, SQL/ODBC databases, live chat/IRC, [and] newsfeeds." (*Id.* at 7.) As "Autonomy's products rely on user interest or employee expertise profiles created implicitly, simply by observing a user's actions" (*id.* at 3),

the Knowledge Server contains information regarding the user's interactions with these various files and programs.

8. Autonomy Agentware Anticipates Claim 7 of the '276 Patent

187. Claim 7 requires "The method of claim 1, wherein said plurality of retrieved documents correspond to a respective plurality of products." Autonomy discloses services explicitly geared toward e-commerce products. (Autonomy WP at 10-11.)

9. Autonomy Agentware Anticipates Claim 21 of the '276 Patent

188. Claim 21 requires "The method of claim 1, wherein using the estimated probabilities for the respective plurality of retrieved documents to present at least a portion of the retrieved documents to the user comprises presenting to the user at least said portion of the retrieved documents based on the estimated probability that the retrieved document is of interest to the user and the relevance of the retrieved document to the search query." Autonomy discloses presenting documents based both on the query and on the computation of whether the document is of interest to the user. *See* Section VI.B.3 above; *see also* Autonomy UG at 7, 11 (disclosing building agents based on both the user's query and the user's documents), Autonomy WP at 3 (disclosing building agents based on both the user's query and the user's documents).

10. Autonomy Agentware Anticipates Claim 22 of the '276 Patent

189. Claim 22 requires "The method of claim 1, wherein identifying properties of the retrieved document comprises identifying properties selected from the properties consisting of ***a topic associated with the retrieved document***, at least one product feature extracted from the retrieved document, an author of the retrieved document, an age of the retrieved document, a list of documents linked to the retrieved document, a number of users who have accessed the retrieved document, and a number of users who have saved the retrieved document in a favorite document list." Autonomy discloses retrieving a topic or concept associated with the document.

See, e.g., Autonomy UG at 4 ("Autonomy does not user keyword searchers, but actually identifies the concepts involved in the text and compares them"), Autonomy WP at 1 (describing "concept extraction").

C. The Montebello Paper and PEA Anticipate Claims 1, 11, 32 and 34 of the '040 Patent and Claims 1, 6, 7, 21, and 22 of the '276 Patent under PUM's Infringement Theories.[16]

1. Background on Montebello/PEA

190. The PEA system described in Montebello is a flexible and modular system that uses machine learning technologies to learn user models from user-specific search data. The system predicts the relevance of documents for the user according to the user model and suggests new documents that are likely to be interesting to a user. It takes personalization ideas from prior art systems, including Mladenic, and applies them to the results retrieved from a search engine. (Montebello at 1.) "Conceptually, the PEA is similar to a meta-search engine, but with the major difference that it employs user profiling to specifically target documents for individual users. In this way duplication and redundancy of information is significantly reduced, while the real needs and interests of the users are fully addressed in a more focussed [*sic*] retrieval." (*Id.*)

[16] E.g., under PUM's theory that any score is the "probability" required by the claims. See below.

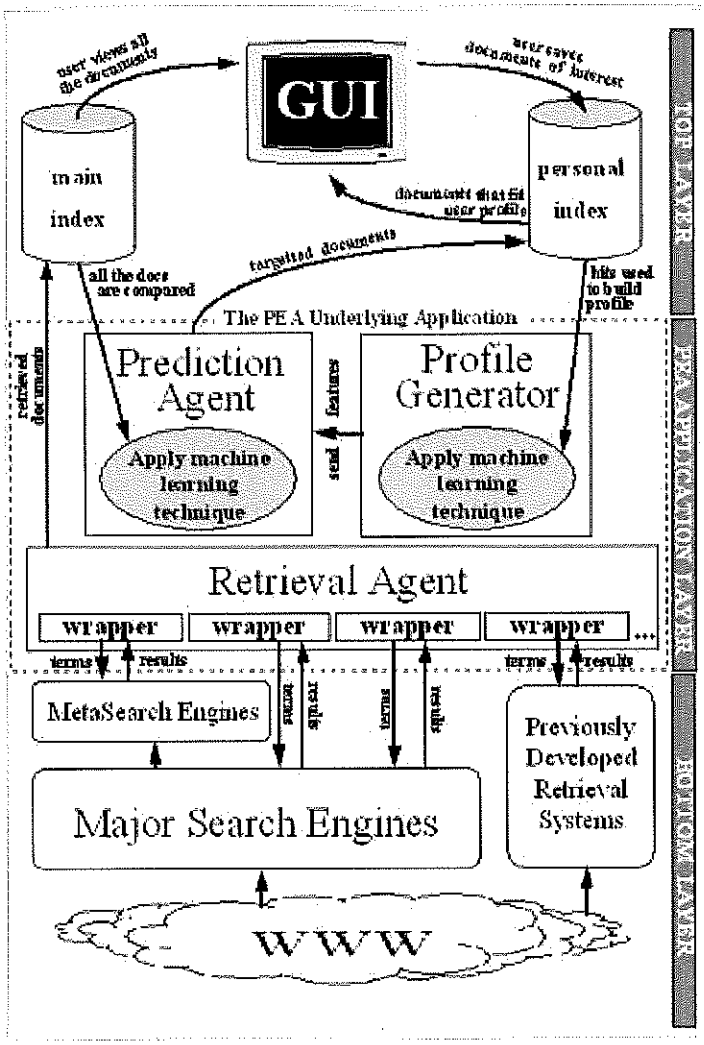

Figure 19: Montebello at 2, Fig. 1

2. Montebello and PEA Anticipate Claim 1 of the '040 Patent under PUM's Infringement Theories.

191. Montebello discloses a computer-implemented method for providing automatic, personalized services to a user. The paper discloses Personal Evolvable Advisor (PEA), "a system that reuses the information generated from search engines together with previously developed systems, and adapts it, by generating user profiles, to better meet the needs and interests of the users by improving recall and precision measure." (Montebello at 1.) At a high level, PEA accepts documents from search engines and other retrieval systems, filters the documents according to the user's profile, and presents the filtered documents to the user. (*Id.*)

"[I]t employs user profiling to specifically target documents for individual users." "[T]he real needs and interests of the users are fully addressed in a more focused retrieval," using a "prediction agent to identify if the particular document is relevant to a particular user or not." (*Id.* at 1, 3.)

(a) Transparently monitoring user interactions with data while the user is engaged in normal use of a computer

192. Montebello/PEA transparently monitors user interactions during normal use of a computer: "The underlying application layer has the difficult task of performing all the work required, transparently from the user." (Montebello at 3.) Specifically, PEA monitors the user's bookmarks: "We assume that normally, when searching or even browsing, a user bookmarks a page of interest and proceeds with the activity he/she was performing. Taking this activity into perspective, all that is required is to take into consideration what the user bookmarks, and utilise this information to generate the profile. [...] [This method] is more reliable than asking users to assign ratings, as it is less demanding on the user's time." (*Id.*) Of note, the claimed patents explicitly list tracking bookmarks as a means of transparently monitoring the user's interactions: "Through his or her actions, the user creates positive and negative patterns. Positive examples are documents of interest to a user: search results that are visited following a search query, ***documents saved in the user favorites or bookmarks file***, web sites that the user visits independently of search queries, etc.... Conceptually, positive and negative examples can be viewed as additions to and subtractions from the user data and resources." (22:12-26.)

(b) Updating user-specific data-files comprising the monitored user interactions and a set of documents associated with a user.

193. Montebello/PEA discloses updating user-specific data files: "The task that the profile generator sets out to achieve is to analyse each users' personal index and generate a profile. If users have different interests stored in their personal index, then a separate profile is

required and generated for each interest." (Montebello at 3.) As detailed above, PEA transparently monitors the user's bookmarks, which contain monitored user interactions—i.e., the action of bookmarking a page. "[A]ll that is required is to take into consideration what the user bookmarks, and utilise this information to generate the profile." (*Id.*)

194. Montebello/PEA also updates "a set of documents associated with a user" under PUM's reading of the claims. As detailed in its infringement allegations, PUM interprets this claim element to require "updating ***information*** about the documents that the user has clicked on" rather than updating the document itself. (Plaintiff's April 2012 Infringement Contentions, Tab A, p. 6.) PEA similarly updates information about documents: "If the document is valid, then an initial paragraph from the document is extracted and saved locally in the main database index together with the reference search term, its reference within the index, the URL, and the document title." (Montebello at 3.)

(c) Estimating parameters of a learning machine

195. Montebello/PEA estimates parameters of a learning machine based on the user-specific data files. "The task that the profile generator sets out to achieve is to analyse each users' personal index and generate a profile." (Montebello at 3.) The index consists of several personal data items, including information on the web pages accessed in the past (url, initial paragraphs, titles) and bookmark activities. Moreover, users can store personal interests in the index: "If users have different interests stored in their personal index, then a separate profile is required and generated for each interest." (*Id.*)

196. As the article explains, "[n]o novel machine learning technique has been developed for the profile generator. It uses specific techniques previously employed by other similar systems," referring to the systems described by Edwards, Payne, and Green that were detailed above. (*Id.*) Among other options, profile generation uses the popular "term

frequency/inverse document frequency machine learning technique," the same technique disclosed as one of the embodiments in the patent. (*Id.*; 11:12-20.)

197. Further, the User Model generated is explicitly user-specific, and is further subdivided into individual topics that might interest a specific user:

> Issues regarding how many profiles to generate for a user - one specific profile per user, a general profile for a group of users, different profiles for different users or different profiles for the same users - have been tackled differently. ***Some profile generators develop the `specific user profile'***, especially those systems which have been produced to cater for specific items like emails or newsgroups, while others specialise in a `specific topic profile'.... We take this argument one step further, and argue that what one user finds interesting in a specific topic, differs from what another user describes as interesting about the same topic. Therefore, ***different profiles need to be generated for every different interest a user has*** if the predicted results are to be focused accurately.

(Montebello at 3-4.) Hence, Montebello/PEA discloses user modeling based on personal data, as the profile is the direct product of the user modeling task. It is widely known that user modeling is the technique utilized to generate a user's profile. Accordingly, Montebello/PEA meets this claim element.

(d) Analyzing a document

198. Montebello/PEA analyzes incoming documents to identify properties of the document:

> The retrieval agent, is responsible for aggregating all the hits returned by the external systems. It collates the results, by removing duplicates and ensuring integrity, and stores the formatted and pre-ranked results as a single list in a local database, known as the main index.... A scan through the WWW page will quickly identify the URL links and list them. Some of the links are useless to the user, so the retrieval agent initially removes adverts, duplicates, and site specific links. It then analyses the vetted URLs and accesses the document on-line. This will identify whether the link is still accessible, has moved or been removed completely. If the document is valid, then an initial paragraph from the document is extracted and saved locally in the main database index together with the reference search term, its reference within the index, the URL, and the document title.

(Montebello at 3.) The retrieved documents are then compared to the profile to determine whether the user is likely to be interested in the document:

> The user interest profile generated by the profile generator will be used by the prediction agent in combination with the extracted features from documents in order to predict and suggest new interesting documents to a user. ***Documents that have been retrieved and stored within the main index by the retrieval agent will have their features extracted and compared to the profile of each individual user*** generated by the profile generator.

(Montebello at 4.)

199. As explained in the profile generation section, PEA "utilizes the term frequency/inverse document frequency machine learning technique." (Montebello at 3.) Term frequency/inverse document frequency (also known as TF/IDF) is a well known, widely used technique in the information retrieval and text mining fields to characterize the statistical content of documents. In particular, it yields a statistical measure for evaluating how important a word is for a given document in a collection. All the documents in the collection must be analyzed to calculate this statistical measure. Again, using TF/IDF is one of the embodiments of the claimed invention. (11:12-20.) Note that in PEA "[t]he machine learning techniques employed to generate the user profile is also applied to extract features from documents." (Montebello at 4.)

200. Accordingly, Montebello/PEA discloses analyzing a document to identify properties of the document.

(e) <u>Estimating a probability that an unseen document is of interest to the user</u>

201. Montebello/PEA estimates a probability that the document is of interest to the user under PUM's interpretations of the claim language. The Court has construed "probability" to mean "numerical degree of belief or likelihood." (Order at 2.) PUM has interpreted the term and the Court's construction to allow for ***any*** numerical score or ranking of a document. (Plaintiff's April 2012 Infringement Contentions, Tab A, pp. 17-19.) Accordingly, under PUM's

infringement arguments, PEA need only generate match scores for documents based on the user profile in order to meet this claim element.

202. Montebello discloses generating match scores for each document based on the user profile:

> The user interest profile generated by the profile generator will be used by the prediction agent in combination with the extracted features from documents in order to predict and suggest new interesting documents to a user. Documents that have been retrieved and stored within the main index by the retrieval agent will have their features extracted and compared to the profile of each individual user generated by the profile generator. This is performed on every item a user has shown interest in, and if any of the documents from the main index happen to fit the user's interests or needs, then they will be eventually suggested to the user the next time the user logs in (Figure 2).

(Montebello at 4.) Furthermore, both profile generation and document analysis use the TF/IDF machine learning technique. (*Id.* at 3: "This profile generation utilizes the term frequency/inverse document frequency machine learning technique"; *id.* at 4: "The machine learning techniques employed to generate the user profile is also applied to extract features from documents.") As I discuss above, TF/IDF yields a statistical measure for evaluating how important a word is for a given document in a collection. The scores for each significant word in the document or profile are aggregated together into vectors or lists. Both the profile and the document are represented as such vectors, and to compare the profile with the document, the corresponding vectors must be mathematically combined, *e.g.*, through a cosine or dot product computation. The resultant score represents the match between the document and the profile, which is a "numerical degree of belief or likelihood" under PUM's interpretation of the claim element.

203. Furthermore, because PEA relies on textual and conceptual analysis of the document in question, it is able to estimate the probability of unseen documents as well as seen documents. At least at some point, PEA likely analyzed a document that the current user had not

previously seen. Under Plaintiff's own infringement allegations, the *likelihood* that an unseen document was presented is sufficient to meet this limitation. (Plaintiff's April 2012 Infringement Contentions, Tab A, p. 18: "Due to the extremely large number candidate search results in Google's indexes, and due to the frequent changes or updates made to the candidate search results, it is likely that candidate search results have not been previously seen by the user.") (emphasis added). Moreover, a content-based recommendation system that does not exclude unseen documents has the inherent capability to recommend unseen documents.

(f) Providing automatic, personalized information services to the user.

204. Montebello/PEA uses the score to provide personalized information services to the user. As the Abstract states, the paper discloses "a system that reuses the information generated from search engines together with previously developed systems, and adapts it, by generating user profiles, to better meet the needs and interests of the users by improving recall and precision measures." PEA uses the user profile to suggest documents that it believes the user will find interesting:

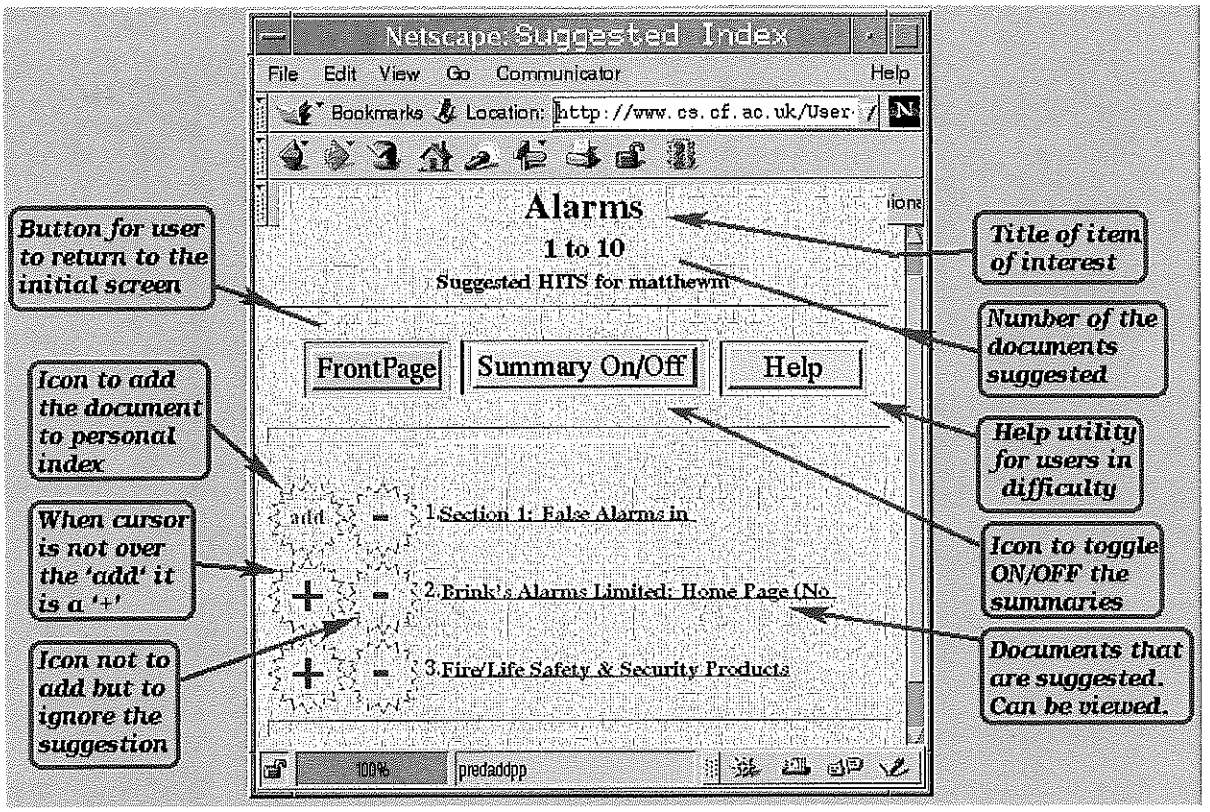

Figure 20: Montebello at 4, Fig. 2

205. Accordingly, Montebello/PEA discloses providing automatic, personalized services to the user.

3. Montebello and PEA Anticipate Claim 11 of the '040 Patent under PUM's Infringement Theories.

206. Claim 11 requires "The method of claim 1 further comprising estimating a posterior probability P(u|d,q) that the document d is of interest to the user u, given a query q submitted by the user." Montebello discloses a user profiling system that operates on top of various search engines: "The Personal Evolvable Advisor (PEA), presented in this position paper, is a system we have developed to reuse information generated by search engines and utilise previously developed retrieval systems. Conceptually, the PEA is similar to a meta-search engine." (Montebello at 1.) "Query terms are used to locate documents and retrieve results from the external systems." (*Id.* at 3.) "Every query term is employed by the wrapper which will

command the associated system to locate documents from its local index and return related results." (*Id.*) Furthermore, "users will be able to suggest any other terms to add to the main search list." (*Id.* at 5.)

4. Montebello and PEA Anticipate Claim 32 of the '040 Patent under PUM's Infringement Theories.

207. Claim 32 of the '040 patent requires "A program storage device accessible by a central computer, tangibly embodying a program of instructions executable by the central computer to perform method steps for providing automatic, personalized information services to a user u," followed by the same method steps required in claim 1 of the '040 patent. As detailed above, Montebello discloses all method steps required in claim 1 and thus it discloses all method steps required in claim 32. Montebello further discloses "a program of instructions executable by the central computer" to perform those method steps. *See, e.g.,* Montebello at 3 (disclosing that the Java programming language was used to write the retrieval agent). Finally, the system disclosed by Montebello is conceptually a personalized meta-search engine, which aggregates the results from multiple distributed searches in a central computer. Accordingly, Montebello and PEA meet all limitations of claim 32 of the '040 Patent.

5. Montebello and PEA Anticipate Claim 34 of the '040 Patent under PUM's Infringement Theories.

208. Claim 34 requires "The program storage device of claim 32 wherein analyzing the document d provides for the analysis of documents having multiple distinct media types." Montebello discloses processing HTML documents. *See, e.g.,* Montebello at 3 (disclosing that the user documents are in HTML), 4 (disclosing computing prediction scores for web pages, which are HTML documents). HTML documents contain multiple media types, including text,

images, and video. By processing HTML documents, Montebello/PEA "provides for the analysis of documents having multiple distinct media types." (*See, e.g.,* 9:14-17.)[17]

6. Montebello and PEA Anticipate Claim 1 of the '276 Patent under PUM's Infringement Theories.

209. Claim 1 of the '276 patent is substantially similar to claim 1 of the '040, as described in Section IV.E above. It contains three additional limitations: a) requiring that the user be normally using a browser in particular rather than a computer in general, b) requiring that the method receive a search query from the user, and c) requiring that the documents analyzed be documents retrieved in response to the search query.

(a) Normal use of a browser program

210. Montebello/PEA discloses monitoring the user during normal use of a browser program. Specifically, Montebello/PEA monitors the user's normal use of the "bookmark" feature within his web browser: "We assume that normally, when searching or even browsing, a user bookmarks a page of interest and proceeds with the activity he/she was performing. Taking this activity into perspective, all that is required is to take into consideration what the user bookmarks, and utilise this information to generate the profile." (Montebello at 3.)

(b) Receiving a search query from the user

211. Montebello/PEA discloses receiving a search query from the user. As detailed above, Montebello discloses a user profiling system that operates on top of various search engines: "The Personal Evolvable Advisor (PEA), presented in this position paper, is a system we have developed to reuse information generated by search engines and utilise previously developed retrieval systems. Conceptually, the PEA is similar to a meta-search engine." (Montebello at 1.) "Query terms are used to locate documents and retrieve results from the external systems." (*Id.* at 3.) "Every query term is employed by the wrapper which will

[17] *See* FN **Error! Bookmark not defined.**, *supra.*

command the associated system to locate documents from its local index and return related results." (*Id.*) Furthermore, "users will be able to suggest any other terms to add to the main search list." (*Id.* at 5.)

(c) Retrieving a plurality of documents based on the search query

212. Montebello/PEA also discloses retrieving a plurality of documents based on the search query. *See* above (disclosing retrieving documents based on the search query).

7. Montebello and PEA Anticipate Claim 6 of the '276 Patent under PUM's Infringement Theories.

213. Claim 6 requires "The method of claim 1, wherein monitoring user interactions with data for a document comprises monitoring at least one type of data selected from the group consisting of information about the document, whether the user viewed the document, ***information about the user's interaction with the document***, context information, the user's degree of interest in the document, time spent by the user viewing the document, whether the user followed at least one link contained in the document, and a number of links in the document followed by the user." Montebello/PEA discloses at least information about the user's interaction with the document, specifically that the user bookmarked that document: "We assume that normally, when searching or even browsing, a user bookmarks a page of interest and proceeds with the activity he/she was performing. Taking this activity into perspective, all that is required is to take into consideration what the user bookmarks, and utilise this information to generate the profile." (Montebello at 3.)

8. Montebello and PEA Anticipate Claim 7 of the '276 Patent under PUM's Infringement Theories.

214. Claim 7 requires "The method of claim 1, wherein said plurality of retrieved documents correspond to a respective plurality of products." As detailed in section VI.A.9 above, PUM interprets this limitation as requiring that the documents ***may*** correspond to

products. As any document on the internet may correspond to a product, and as Montebello/PEA does not make any attempt to distinguish pages that correspond to products from pages that do not correspond to products, any given set of documents analyzed by Montebello/PEA may also correspond to products. Accordingly, Montebello/PEA discloses this claim element under PUMs own infringement theories.

9. Montebello and PEA Anticipate Claim 21 of the '276 Patent under PUM's Infringement Theories.

215. Claim 21 requires "The method of claim 1, wherein using the estimated probabilities for the respective plurality of retrieved documents to present at least a portion of the retrieved documents to the user comprises presenting to the user at least said portion of the retrieved documents based on the estimated probability that the retrieved document is of interest to the user and the relevance of the retrieved document to the search query." Montebello/PEA discloses presenting a document based both on the query and on the computation of whether the document is of interest to the user. *See* Section VI.C.3 above; *see also* Montebello at 3: "Query terms are used to locate documents and retrieve results from the external systems."

10. Montebello and PEA Anticipate Claim 22 of the '276 Patent under PUM's Infringement Theories.

216. Claim 22 requires "The method of claim 1, wherein identifying properties of the retrieved document comprises identifying properties selected from the properties consisting of ***a topic associated with the retrieved document***, at least one product feature extracted from the retrieved document, an author of the retrieved document, an age of the retrieved document, a list of documents linked to the retrieved document, a number of users who have accessed the retrieved document, and a number of users who have saved the retrieved document in a favorite document list." Montebello/PEA discloses extracting and saving the document title: "If the document is valid, then an initial paragraph from the document is extracted and saved locally in

the main database index together with the reference search term, its reference within the index, the URL, and the document title." (Montebello at 3.)

D. The Wasfi Paper and ProfBuilder Anticipate Claims 1, 22, 32, and 34 of the '040 Patent

1. Background on Wasfi/ProfBuilder

217. Wasfi describes the ProfBuilder system, "a transparent, adaptive, autonomous agent which works as a recommender system." (Wasfi at 60.) "ProfBuilder inhabits a Web site and is assigned the goal of finding relevant local pages for the site's users." (*Id.*) "ProfBuilder keeps track of each individual user and provides that person online assistance.... Content-based filtering is based on the correlation between the content of the pages and the user's preferences." (*Id.*) ProfBuilder operates as a separate HTML "frame" at the top of the user's browser. (*Id.* at 61.) The frame contains a list of recommended web pages along with the strength of the recommendations:

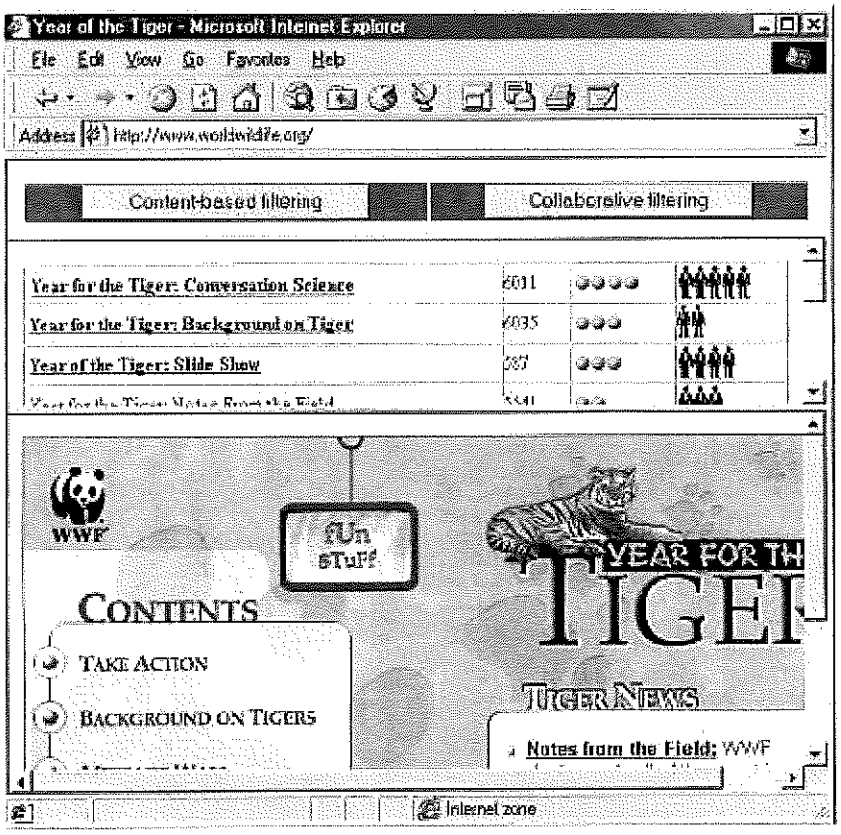

Figure 2 ProfBuilder's interface.

Figure 21: Wasfi at 62, Fig. 2

2. Wasfi and ProfBuilder Anticipate Claim 1 of the '040 Patent

218. Wasfi discloses "an autonomous agent, named ProfBuilder, that works as an online recommender system for a Web site. ProfBuilder uses the usage information as a base for content-based and collaborative filtering." (Wasfi at 57.)

(a) Transparently monitoring user interactions with data while the user is engaged in normal use of a computer

219. Wasfi/ProfBuilder discloses transparently monitoring the user's interactions with the Internet. Specifically, Wasfi discloses three basic means of learning a user's interests, including "Indirect or transparent learning technique: The system learns user preferences transparently without any extra effort from the user." (Wasfi at 57.) Wasfi then reviews current methods for transparent monitoring, including "saving a reference to a page," before "propos[ing] a new mechanism that learns user interests and adapts automatically to their

changes without user intervention." (*Id.* at 57-58.) Furthermore, ProfBuilder is described as "a ***transparent***, adaptive, autonomous agent which works as a recommender system." (Wasfi at 60.) ProfBuilder "is transparent as it extracts the preferences without user intervention." (*Id.*)

(b) Updating user-specific data-files comprising the monitored user interactions and a set of documents associated with a user.

220. Wasfi/ProfBuilder discloses updating user-specific data files. Specifically, Wasfi notes the user's current web page, designated as s_i. (Wasfi at 58.) Since s_i is the user's *current* web page, it represents part of the set of documents associated with a user. Wasfi also determines the user's interaction with that page, designated by t_{ij}, "which is a nonnegative number between zero and one, indicates the relevance or importance of page s_i to user u_j." (*Id.*) As users traverse to other pages, each current page is extracted, its importance to the user is determined, and the page and the importance are used to update the user profile. *See* below. This learning mechanism is employed in the ProfBuilder application. (*Id.* at 60: "ProfBuilder's learning mechanism is implemented based on the algorithm described in the preceding section.") Accordingly, Wasfi/ProfBuilder meets both aspects of the "user-specific data files" limitation.

(c) Estimating parameters of a learning machine

221. Wasfi/ProfBuilder estimates parameters of a learning machine based on the user-specific data files. In describing the learning mechanism, Wasfi observes that "[a]n appropriate representation for profiles and pages is based on vector-space representation, commonly used in information retrieval (IR) literature." (Wasfi at 58):

> [P]ages and queries (profiles) are represented as vectors in some hyper-space....
>
> [T]he method for profile reformulation [i.e., estimating new parameters] in response to the changes of user's interest is based on vector adjustment. Since profiles and pages are both vectors, the profile should move closer to the vectors representing pages which are relevant and away from the vector representing pages which are non-relevant....

> Consider that page s_i is the current page of user u_j. Let us assume that variable t_{ij}, which is a nonnegative number between zero and one, indicates the relevance or importance of page s_i to user u_j. A reformulation of vector **Q_j representing the user profile** is obtained by taking Q_j and adding the vector elements D_i representing page s_i after it is changed in proportion to t_{ij},
>
> $Q_j = Q_j + t_{ij} * D_i$
>
> *i.e.* the weight of each word in D_i is modified proportional to t_{ij}. The resulting effect is that, for those words already present in the profile, the word-weights are modified in proportion to $t_{ij} * D_i$. Words which are not in the profile are added to it.

(Wasfi at 58.) Representing the user profile as a keyword vector and modifying the values of that profile based on the keywords in the document is one of the embodiments of the claimed invention. (10:52 – 12:27; *see also* 1:55-60 (conceding this technique is in the prior art).) This learning algorithm is used within the ProfBuilder system. (Wasfi at 60.)

222. Further, "ProfBuilder keeps track of ***each individual user*** and provides that person online assistance.... Content-based filtering is based on the correlation between the content of the pages and the user's preferences." (Wasfi at 60.) Wasfi tracks users by their IP addresses. (*Id.*) To distinguish two users who might share the same computer, "a timeout mechanism is used to delete user's session information after a predetermined amount of idle time." (*Id.*) This means that "a connection after the specified period having the same IP is identified as a new user." (*Id.*)

(d) Analyzing a document

223. Wasfi/ProfBuilder discloses analyzing a document:

> The filtering process consists of ***translating pages to their vector space representation***, finding pages that are similar to the profile, and selecting the top-scoring pages for presentation to a user.
>
> The vector representation is obtained by a text analysis of the HTML pages. This is done by ***extracting keywords from page titles, all level of headings, and anchor hypertexts***.... Stop words [5] are filtered out an word stemming [6] is then performed to improve IR performance. ***The keywords are weighted based on the***

> ***well-test algorithm TDIDF*** [*sic*] [16]. The weight of a keyword in one page is the product of its keyword frequency and the inverse of its document frequency.

(Wasfi at 61.) Note that analyzing documents through term frequency / inverse document frequency is one of the preferred embodiments of the patents. (11:12-20.)

(e) <u>Estimating a probability that an unseen document is of interest to the user</u>

224. Wasfi/ProfBuilder estimates a probability that the document is of interest to the user under PUM's interpretations of the claim language. The Court has construed "probability" to mean "numerical degree of belief or likelihood." (Order at 2.) PUM has interpreted the term and the Court's construction to allow for ***any*** numerical score or ranking of a document. (Plaintiff's April 2012 Infringement Contentions, Tab A, pp. 17-19.) Accordingly, under PUM's infringement arguments, Wasfi need only generate match scores for documents based on the user profile in order to meet this claim element. I also note, however, that Wasfi does explicitly disclose the estimation of posterior probabilities, defining the interestingness of a page to the user as the logarithm of a posterior probability of the user's accessing that page given his or her browsing history, and updating the user profile using this interestingness value as a step size.

225. Wasfi discloses generating a similarity score by taking the scalar product of the user profile vector and the document vector:

> The similarity metric between the vector Di representing page s_i and the vector Q_j representing the interests of user u_j is calculated by taking a scalar product of the two vector[s],
>
> $$Similarity(D_i, Q_j) = \sum_k w_{ik} * w_{jk}$$

(Wasfi at 61.)

226. Further, "ProfBuilder keeps track of ***each individual user*** and provides that person online assistance.... Content-based filtering is based on the correlation between the