iAGENT: A Novel and Intelligent Assistant to Personalised Search

Sompa Malakar
College of Engineering and Technology
Techno Campus,Ghatikia
Bhubaneswar,Orissa,India

ABSTRACT
This paper presents a novel approach to personalised search. By constantly learning and updating user profiles to the current needs of the user, relevant results are returned. The paper puts forward the concept of "iAGENT" which is an intelligent agent that assists a user to get relevant documents by modifying the query given by the user in accordance with the web pages previously visited. iAGENT has a novel method of maintaining two kinds of profile for the user. The first profile keeps track of the pages visited by the user and the second keeps track of the pages that were not visited by the user. The second profile is an added feature of iAGENT that keeps a backup of the websites that were irrelevant at some point of time, but may be required in the future. When a request to a page from the second profile is made, the overhead of following the entire procedure of querying the search engine is avoided. Besides, both the profiles are constantly updated to keep track of the user's current interest. Experimental results show that the iAGENT is efficient enough to personalise the search results to an appreciable degree.

Categories and Subject Descriptors
H.3.3 **[Information Search and Retrieval**]: Information Search and Retrieval – *Query formulation, Search process, Information filtering.*

General Terms
Algorithms, Design, Theory.

Keywords
Personalization, Intelligent agents, User behaviour.

1. INTRODUCTION
The World Wide Web is growing at a very fast pace, so is its amount of information and resources contained in it. This has indeed resulted in numerous options for any particular topic in question.

Every time one tries to find any particular information with the help of a search engine, millions of results are returned within fractions of seconds, out of which a few of them are relevant to the user. The results which are relevant depend on his current field of interest. It is really a tedious job to find correct personalised information by browsing through hyperlinks or querying the search engine or following any of the existing categories under popular websites. Today's commercial search engines strategies do not really work for individual user's requirement and it is really a time consuming process to search for information through these traditional methods. A study of user behaviour reveals that the users usually expect results bang on point and are too lazy to navigate through different web pages looking for potential information. Also, it reveals that 90% of the users are unwilling to specify their intentions clearly or give a feedback about a web page (thumbs up or thumbs down) due to the time constraint. All these necessitates for an alternative and easy way of information retrieval based on individual's interest. Hence, in order to solve the challenge of retrieving personalized information intelligent agents are being developed. An Intelligent agent is a program that does the tough job of information gathering automatically without any human intervention. The past decade has seen a tremendous tilt towards agent oriented technology. The success of such agents can be a breakthrough in the technological world and renders great scope ahead. In particular, the iAGENT is a program running on a client terminal that assists the user to get the relevant information. It acts like a stand-alone proxy, exactly like the WebMate [2], that monitors all the HTTP transactions taking places between the end user and the WWW. To give a brief overview of the entire approach, firstly in this paper, a brief study of user behaviour is made to find out the key points to be considered. Then we describe the procedure to build an initial user profile, update the naive profile and propose an algorithm for query modification in an entirely new way.

2. USER BEHAVIOUR
A study of user behaviour reveals several critical points to be considered when building a search agent. It is not possible for a user to specify his/her intentions in greater detail to any agent or search engine due to the time constraint. It is not possible for a user to exactly represent his or her intention in possibly few words. When a search engine returns results for a particular query, the user is only interested in the top results that appear on the first ,second and third pages, and do not navigate any further though the search engine returns millions of hits within seconds. Also users are too lazy to give a feedback on whether they liked a particular page or not. This necessitates the need for an automatic learning technique that can learn and predict what a user is looking for. Besides, the web page that may be of interest to the user may not be useful at some later point of time. This can be achieved by learning, maintain and updating the user profiles as and when required. The general agent architecture is shown in figure 1. The agent stands as a proxy between the search engine and the user machine.

3. BUILDING A USER PROFILE

3.1 Initial Stage

When the search agent is introduced with a new user's query (i.e there is no prior information in the user's profile, we term it as "**naive**" profile), the agent simply forwards the same query to the search engine without any modification. As the agent is only supplied with a native profile of the user, it does not have any previous information of the user's field of interest. After the search engine returns results for the query given, the agent segregates them into one or more categories or the overhead of categorising the search results can be avoided by sending the query to the existing search engines that already have this feature of categorising the results. Popular search engines of today like Yahoo! (http://www.yahoo.com), ODP (http://www.dmoz.org) and Google (http://www.google.com) return both categories as well as documents. Other search engines like Northern Light (http://www.northernlight.com) and WiseNut (http://www.wisenut.com) cluster their results into categories. The most recent search engine launched by Microsoft Corp. named Bing (http://www.bing.com) provides different categorises for different query along with the top rated search results. When the iAGENT is provided with such categorised search results, it only returns the categories to the user for the first time. When the user chooses to explore a topic from among the given categories, the user's field of interest becomes clear for the current query and the naive profile is updated. Now the user profile has some initial information about the user's intent i.e. what the user is actually looking for. We sent a query to the search engine Bing and the results are shown in a screen shot in figure 2. The user did not specify whether he/she is looking for the airline services, tourism, visa, or the universities in Singapore, however when the user clicks on one of the category , the iAGENT returns to it ,the user's field of interest for the current query becomes clear. Then the iAGENT returns only the top 50 results under that category to the user. The agent then runs an algorithm on the result set and the naive profile to build an initial user profile. Let for a query, Q, a total of C_i number of categories is returned to the iAGENT by the search engine. For any i, the category C_i has D_i number of documents under it. Let **tf *(w, d)*** be the term frequency i.e. the number of times a particular word ***w*** appears in a document ***d***. The ***DF(w)*** is the document frequency i.e. the number of documents in which the word ***w*** appears at least once. The inverse document frequency ***IDF(w)*** is then calculated as

$$IDF(w) = \frac{log|D|}{DF(w)}$$

In the above equation, |D| is the total number of documents. The weight $W^{(i)}$ of an element in a vector is then calculated as

$$W^{(i)} = tf(w_i,d) \text{ x } IDF(w_i)$$

Each document D_i is represented as a vector in the vector space. Documents with similar vector have similar content in them. Each document vector is represented in the space in different dimensions by the words and their weight. We consider the Euclidian distance between two vectors d_x and d_y given by $dx.dy/(|d_x|d_y||)$. This ratio defines the cosine angle between the vectors, with values between 0 and 1. This ratio is calculated to normalize the length of documents since long documents tend to have large term frequencies. The agent preferably builds two profiles for a single user. Initially, it starts with the naive profile and gathers sufficient information for it. The second profile is gradually built during the process of learning. The second profile is a supporting profile acts like a cache that gives additional efficiency to the agent. It keeps track of the pages that were not explored by the user much or explored but were not relevant to the user during that point of time. It may so happen that the user may be interested in the contents of one of these pages at some later point of time. The following algorithm is applied to the result set obtained initially under a category C_i to build an initial user profile.

*Step1.*Extract the top 50 results from the result set returned to the agent for the category C_i.

*Step2.*Do the following to the documents, whenever the user clicks one of the links in the result set.

a. Start a timer and keep track of the time. If the timer exceeds a threshold time t_{max} or if the user bookmarks the web page

i) Parse the HTML document

ii) Delete all the stop word (is, an, the, in, or, a, that, then, etc. (the agent is provided with a predefined set of stop words))

iii) Determine the base form of a word by stemming the plural noun to its singular form and inflexed verb to its original form.

iv) Extract the title<TITLE>, headers<H1, H2, and H3> as they are given more weights.

a. Extract the **tf-IDF** vector for the document and let it be represented as V_i.

b. Update the naive profile with the vector V_i.

1. For those web pages whose timer value $t < t_{max}$ and the documents that were not explored by the user (remaining documents out of 50) do

i) Parse the HTML documents, deleting all the stop words, determine the base form of the word and extract the headers and title for each page.

ii) Extract the tf-IDF vector for each of the pages and update them to the second profile P2.

With the probability that the web pages (whose $t < t_{max}$) which were once visited by the user may of some interest in the long run, the second profile is maintained. The advantage is when the user looks for information contained in such pages; the profile P2 gives a faster retrieval of the relevant information without having to again to follow the long procedure of querying the search engine that returns results with poor relevancy.

3.2 Learning and Updating User Profiles

The need of a user fluctuates from time to time. A user usually looks for information in varied fields. A relevant web page may not be relevant at all at some later point of time. Hence, it is highly recommended to keep the profile of a user updated with his or her current interests. IAGENT continually keeps learning and updating the user profile (both P1 and P2) to meet the requirements of the user. The following algorithm helps the agent to learn from the user profile and update it with the currently required information:

Step 1. Calculate the cosine similarity for every two ***tf-IDF*** vectors in both the profile sets P1 and P2.Also calculate the cosine similarity of the new vector V_i along with each of the vectors ,first from profile P1 and then from the profile P2. Assume the profile set

P1={V_{11},V_{12},V_{13},V_{14},V_{15}...V_{1n}} and

P2= {V_{21},V_{22},V_{23},V_{24},V_{25},.....V_{2n}}

$$Sim(V_j,V_k) = \frac{Vj.Vk}{|Vj||Vk|} \text{ (j, k=1, 2, 3...n)}$$

Step 2. If the cosine similarity of the vector V_i and the vector V_{1i} of profile P1 is closed to 1, then P_1UV_i and combine the two vectors with the greatest cosine similarity

$$V_l = V_l + V_m \qquad (l, m) = arg \max_{(xy)} Sim(Vx, Vy)$$

Else P_2UV_i.

1. If a new vector is added to profile P2, combine the two vectors with the greatest cosine similarity

$$V_l = V_l + V_m$$

$$(l, m) = arg \max_{(xy)} Sim(Vx, Vy)$$

And add the vector V_l to the profile set P1.

2. Sort the weights in the new vector V_i (or V_l) in decreasing order and keep the highest M elements.

The above algorithm runs when the user bookmarks a new web page or explores a new link from the result set. The new vector V_i always has a higher weight because it is extracted from a relevant web page, from the result set returned as a result of query modification by the iAGENT. Thus there is no need of any relevancy feedback from the user; the agent automatically finds the relevancy of a page by learning from the profile and modifying the query.

3.3 Query Modification

Query Modification is an age old technique being used to enhance the precision of the search systems. Many previous attempts have been made in this area. Mitra et al. [6] describes an automatic approach to discover extra query terms that can improve search precision. Generally, a user query is modified by adding extra words that can help increase the relevancy rate remarkably i.e. by expansion of the given query. The query can be expanded using three ways: manual query expansion, semi-manual query expansion and automatic query expansion. The iAGENT provides an automatic query modification technique that increases the relevancy rate of the results obtained to an appreciable degree. Query modification is faster approach to retrieve relevant information from the web. In the process of query modification, the query given by the user is different from the internal query sent to the search engine by the agent. The iAGENT modifies the user query based on its learning from the user profile P1. A separate thread runs to store the words of the web pages explored by the user ($t>t_{max}$) in its database, obtained by parsing the HTML documents (deleting the stop words and determining the base form of different words) where the words in the title, headers and top of the document are given more weight age than the other words. Also the words that are in proximity with the user query are given a higher weight age. For each web page, we select at most 10 words that represent the content of the page and store them as a separate set. Every query made to the search engine returns documents containing the query word. In a particular document, it is usually a case that the words that occur in close proximity with the query word are of much importance and specific to the word. These words together with the query word represent a better query to retrieve that document to the intended user. For example, a user looking for information on heart diseases writes an ambiguous query word "heart", which can be inferred as either related to medical terminologies which the user is actually looking for or can be related to emotions, love, valentine, etc. Single word query are ambiguous and do not specify a user's intent appropriately. This can lead to a lot of spurious hits that are of poor relevancy. If the query given is modified by appending words that are extracted from the documents previously learned from the user profiles, then the relevancy of the results can be drastically improved. An added point to the above made example is that the words in a document that occur near to the query word in that document can specify the document content more appropriately. If a document contains the word "heart", the word following it may be "disease", "attack", and "anatomy", "surgery", giving a clear idea that the document is related to treatment of heart diseases or other medical purposes. However, if the words "emotion", "wishes", "love", "valentine" occurs in the same sentence as the word "heart", it gives an entirely different notion. Hence, the distance of a word from the query word in a document is an important point to be considered. As the study of user behaviour reveals that the users are reluctant to explicitly specify their intentions and also it is not possible to exactly represent a user's intent in words, even if the user is quite clear about what information he or she is actually looking for, automatic query modification can serve as an antidote to poor relevancy of results. An attempt to search refinement by query expansion has been made by Liren Chen and Katia Sycara et al. [2] using the Trigger Pair model [16, 17]. Let S be the set of words representing a document D termed as the "*summary set*" of that particular document. The summary set S of a document can comprise a maximum of ten words. The elements of the set S are those words that occur in the same sentence as the query word, the words that occur in the < title>, <header1>, <header2>, <header3> of the html document, the words from the text portion that are in block or bold or italicized, along with the words that occur frequently (not including the stop words).The words are sorted in decreasing order of their weights. The highest weight age is given to the words occurring in the same sentence as the query word. The next highest weight age is given to the words in the <title>, <header1>, <header2>, <header3> and then to the words in block letters, bold or italicized. The least weight age is given to the words that occur frequently in the document. The algorithm proposed for query modification is given below:

Step1. Search for the documents containing the query word in the database of profile P1.

Step2.If such documents are found, extract the summary set of those documents and find out the weights of the each element of each summary set. (The weight is found by calculating the distance of the word from the query word, lower the distance, higher is the weight of the word.)

Step3. Extract the word with the highest weight from each of the summary sets and from among them append the word with the highest weight to the query word to get the modified query.

If two or more words have the same weight, then the word belonging to the document having a higher vector length is chosen for query modification.

4. EXPERIMENTS

Two experiments were conducted to test for the accuracy of the results returned by the iAGENT. The learning ability of the iAGENT helps to identify the web pages that are relevant to the user. The agent is trained with a prior set of documents on various fields. The training set also included some web pages that contained information on heart diseases, anatomy of human heart and other medical treatments. The iAGENT has already built two user profiles for that particular user. Suppose the user looking for information on the treatment of heart diseases gives an ambiguous query "heart" to the search engine. In this case, the search engine used was Google. Without running the iAGENT software on the client machine, the results returned where recorded. We took into consideration the first eight results out of a total of 50 hits.

i) *HEART- Official site includes news, tour dates, journal, photos, biography, discography, video clips*. www.heart-music.com/

**ii) The Human Heart: An Online Exploration from the Franklin Institute... Interactive tour of the heart. From Franklin Institute in Philadelphia. www.fi.edu/learn/heart*

iii) Heart-Wikipedia, the free encyclopaedia en.wikipedia.org/wiki/heart

iv) Roxette- Listen to your heart VIDEO-5min www.youtube.com/watch

***v) Heart risk advice for people with psoriasis guardian.co.uk*

vi) 800th Implant of World's only Approved Total Artificial Heart... MarketWatch

***vii) American Heart Association-Learn more about American Heart Associations effort to reduce health caused by cardiovascular diseases www.americanheart.org*

viii) The Heart-KidsHealth –the right side of your heart receives blood from the body and pumps it to the lungs. www.kidshealth.org/kid/htbw/heart.html

It is clear from the above recorded results that only three out of eight hits were relevant to the user. Next recorded results are the results returned by the iAgent after internally modifying the query.

***i) American Heart Association-Learn more about American Heart Association's effort to reduce death caused by cardiovascular diseases. www.americanheart.org*

***ii) WebMD Heart Disease Health Center- Information about heart diseases.. Learn about heart disease symptoms, risk factors and prevention, as well as information on heart attack, heart failure... www.webmd.com*

***iii) Heart Disease<<FAQs<<womenshealth.gov www.womenshealth.gov/faq/heart-disease ...*

***iv) CDC- DHDSP-Heart Disease Home- Heart disease is the leading cause of death in the US and is a major cause of disability... www.cdc.gov/heartdisease*

***v) The Heart Disease and Cardiology Home page-The starting place for exploring information of heart disease and its treatment heartdisease.about.com*

***vi) Coronary Heart Disease- Symptoms, diagnosis, treatment of...Sept 3, 2008.. Coronary Heart disease (CHD) is narrowing of the small...Coronary artery disease; Arteriosclerotic heart.. health.nytimes.com/../overview.html*

***vii) Heart Disease- MayoClinic.com- Comprehensive overview covers symptoms, causes, treatment and prevention of heart disease.*

***viii) Cholesterol Genetics and heart disease Institute- Home... Services- About us- Contact us- Home www.heartdisease.org*

All the eight links returned by the iAGENT were relevant to the user. We conducted the second experiment where the user with its initial profile gives a query "heart" and looks for information related to its emotional meaning. The agent uses its intelligence to see that the same query is given for a second time. Hence, it returns categories to the user (like heart disease, cards for your sweetheart, Valentine's Day Heart, Love Poems, Quotes and SMS, Heart attack). When the user clicks on one of the categories, the agent returns the relevant results accordingly and updates the user profile with the new interest field of the user. The following are the eight out of the 50 links that were returned to the user when he clicked on the category "love poems, quotes, SMS"

***i) Heart Quotes- Quotes About the Heart, Romantic Heart...send to someone special www.links2love.com/love_heart_quote*

***ii) a beautiful revolution: blog: the girl I love with all my heart www.abeautifulrevolution.com/blog/t*

***iii) Heart | What About Love lyrics www.lyricsfreak.com/h/heart/what+about+love*

***iv) Moment of love, Heart Love- Every person in the world has a heart. The Moment of Love reminds us of our common humanity and invites us to... www.momentoflove.org*

***v) Amzon.com: The Heart of Love: How to go beyond fantasy to find true relationship fulfilment: John. F. Demartini: Books www.amazon.com/.../1409874*

***vi) How does the heart know love? Brings new and meaningful personal insights from his emotional and spiritual transformation in.. www.howdoestheheart.com*

***vii) Romance at Heart Magazine: Free online romance reads and books www.romanceatheart.com/fos.html*

***viii) From the Heart Romance Writers: FTHRW is an Online Chapter of Romance Writers of America. www.fthrw.com/index.php*

4. RELATED WORK

Personalising web search is a challenging task, though numerous attempts have been made to meet this need of the hour. Syskill & Webert et al.[1] is a software agent that identifies interesting web sites for the user. Letizia et al.[4] is a software agent that monitors the user behaviour when he or she is browsing the WWW, and tries to infer the user's interest based on the browsing behaviour. This removes the necessity of user's overhead of explicitly providing a feedback whether the page returned was useful or not. The WebWatcher et al. [3] is a tour guide to the WWW. The system is designed to help user retrieve information from the web sites previously visited by some user. It learns from the experience of multiple users. It watches a user traversing the WWW and it helps the user when similar goals occur in the future. WebMate et al.[2] is another agent that helps in effective browsing and searching of personalized information. It also compiles and sends the users a personal newspaper by automatically spiding news sources. Some approaches have been made wherein the users are

explicitly asked to describe their general interests. Google Personal asks users to build a profile of them by selecting their categories of interests. This profile can be used to personalize search results by mapping the web pages to the same categories.

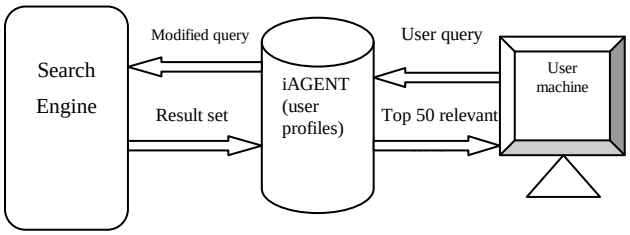

Figure 1: iAGENT Architecture

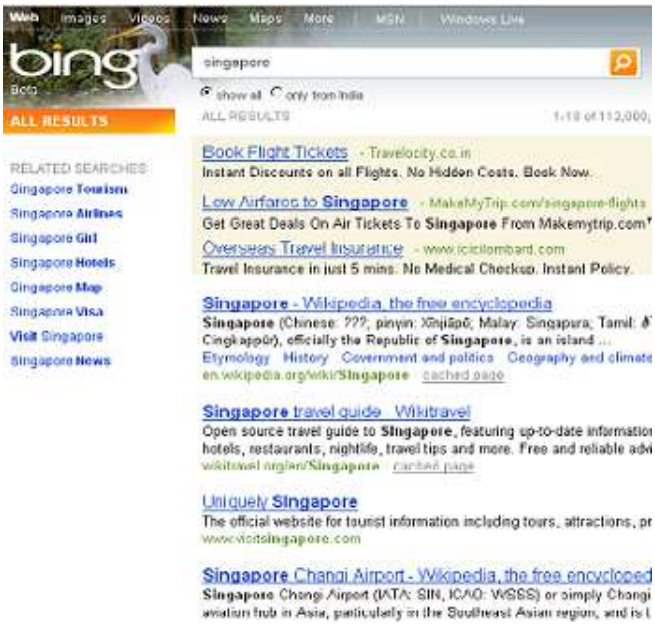

Figure 2: Screenshot showing results returned by Bing

5. CONCLUSION AND FUTURE WORK

With the advent of agent oriented technology, newer and more efficient strategies for personalising search are being developed. iAGENT presents a novel approach to personalise the search results and improve the relevancy rate. It's a software running on the client side to filter out irrelevant results by learning and updating the user profiles to its current interest. iAGENT maintains a second profile apart from the regular one. The advantage of having a second profile is that it acts like a cache and keeps track of the web pages that were not explored by the user. This profile bolsters the agent by increasing the retrieval rate of any such pages in future.

In the present scenario, the challenge to our strategy of personalization is to filter out the irrelevant content of a web page such as advertisements and to solve the problem of keyword spam. A keyword spammer only needs to repeat the same word several times to increase the weight of that word. iAGENT uses words for the representation of a user profile. There is a necessity for understanding the relationship between the words and other higher concepts. More efficient machine learning algorithms are required to identify new web pages to increase the relevancy rate. Future researches will be to represent a profile using phrases, bigrams and other higher concepts of relationship between words.

6. REFERENCES

[1] Pazzani, M., Maramatsu, J., Billsus. D., 1996, Syskill and Webert: Identifying interesting web sites. In AAAI conference, Portland 1996

[2] Liren Chen, Katia Sycara, 1998. WebMate: A personal agent for browsing and searching. International Conference on Autonomous Agents, 1998

[3] Throsten Joachims, Dayne Freitag, Tom Mitchell.1997, WebWatcher: A Tour guide for World Wide Web. Proceedings of IJCA197, August, 1997

[4] Leiberman,H.,1995, Letizia: An agent that assists web browsing. In International Joint Conference of Artificial Intelligence, Montreal, August, 1995

[5] Gerard Salton, Chris Buckley.1988, Improving Retrieval Performance by Relevance Feedback, Cornell University.88-898

[6] Mandar Mitra, Amit Singhal, and Chris Buckley. Improving automatic query expansion. In ACM SIGIR 98, Melbourne Australia, 1998. ACM

[7] Liu, F., Yu, C. and Meng, W. (2002). Personalised Web Search by mapping user queries to categories. In Proceedings of CKIM, 2002, 558-565

[8] Morita, M. and Shinoda, Y. (1994). Information filtering based on user behaviour analysis and best match text retrieval. In Proceedings of SIGIR, 1994. 272-281

[9] Pitkow, J., Schutze, H., Cass, T., Cooley, R., Turnbull, D., Edmonds, A., Adar, E and Brevel, T. (2002). Personalised Search. Communications of the ACM, 45(9): 50-55

[10] Sugiyama, K., Hatano, K. and Yoshikawa, M. (2004). Adaptive Web Search based on user profile constructed without any effort from the user. In Proceedings of WWW 2004, 675-684

[11] Gauch, S., Chafee, J. and Pretschner, A. (2004). Ontology based Personalised search and browsing. Web Intelligence and Agent Systems, 1(3-4): 219-234

[12] Katia Sycara, Anandeep Pannu, Mike Williamson, Dajun Zeng, Keih Decker. 1996, Distributed Intelligent Agents. Published in IEEE Expert, Intelligent System and their applications, Dec, 1996.

[13] Marko Balabanovic, Yoav Shaham, 1995. Learning Information Retrieval Agents: Experiments with Automated Web Browsing. Proceedings of AAAI Spring Symposium Series on Information Gathering from Heterogeneous, Distributed Environments: 13-18

[14] Peter, W., Foltz, Susan, T., Dumais. 1992, Personalised Information Delivery: An Analysis of Information filtering Methods. Published in Communications of the ACM. 35(12), 51-60, 1992

[15] Salton, G., and McGill, M.,J., 1983, Introduction to Modern Information Retrieval. McGraw-Hill, New York, 1983.

[16] Ronald Rosenfeld, 1994, Adaptive Statistical Language Modelling: A Maximum Entropy Approach, Carnegie Mellon University, Ph. D. Thesis

[17] Susan Gauch, Robert, P., Futrelle. Experiments in Automatic Word Class and Word Sense Identification for Information Retrieval. Proceedings of the Third Annual Symposium on Document Analysis and Information Retrieval.

[18] Jamie Teevan, Susan, T., Dumais, Eric Horvitz.2005, Personalising Search via Automated Analysis of Interests and Activities. SIGIR'05