Social Information Filtering

Tersia Gowases, Student No: 165531

June 21, 2006

1 Introduction

In today's society an individual has access to large quantities of data – there is literally information about anything and everything. The Internet and the World Wide Web have contributed to the quantity of information available. According to Google.com there are approximately 4,285,199,744 websites on the web, each catering to different topics and different target groups.

Before the rise of the Internet, most of the information that people received was filtered to a certain extent and quality of the content was known. Newspaper and magazine editors only printed articles which their readers would find interesting. Bookshop owners stocked books which they thought their customers would find interesting [5] and purchase.

The Internet has opened up new and important opportunities. One such opportunity is knowledge exchange [4], where people from different social, cultural and professional backgrounds are able to express ideas and exchange information between users with similar interest. Sometimes a user needs to find out information about a particular topic or keep up-to-date with recent developments or increase his/her contacts with other people with similar interests and / specialties. Kobsa [4], has identified two possible reasons why this is not easy to achieve. First, there is no quality control since anyone is allowed to put any information they like on the Internet. Second, there is often too much information, making it difficult to find what is interesting and relevant.

One can thus conclude that Internet users are in need of some type of information sorting tool. Several information filtering tools have been developed over the recent years. One of the most popular methods is *content-based filtering* [5]. A Content-based filtering system [1] recommends an item to a

user based on how similar the item's content is to that of other items which the user has recommended (or preferred) in the past. However some of the drawbacks of content-based filtering are [5]: the items must be in some machine parsable form therefore excluding multimedia information, they have no inbuilt methods for generating unexpected results, and items can not be filtered according to the quality of their content.

These drawbacks have led to the development of *social information filtering* [5]. Social information filtering attempts to automate the process of "word of mouth" recommendations by filtering information based on the (valued) recommendations of other people with similar interests. Social information filtering overcomes the limitations experienced by content-based filtering because items are not subject to computer parsing [5]. In addition, recommendations are based on the quality of the item and the system may recommend an item that is not similar to other items that the user may have indicated earlier.

This paper focuses on the different techniques that are used to achieve social information filtering. Section 2 deals with the different approaches of social information filtering, whereas Section 3 introduces applications that make use of social information filtering techniques. The conclusions are drawn in Section 4.

2 Social information filtering techniques

There are numerous approaches of achieving social information filtering. This section describes three approaches: user profiling, implicit information extraction and the HITS algorithm [5, 2, 3].

2.1 User profiles

User profiling is the process of storing user details. The stored information may include items such as personal details of the user (i.e. user's name, location, email address, etc.), user's preferences (what the user has stated as his/her likes or dislikes), user's interests, and items that the user has personally selected in the past.

A social information filtering system creates user profiles and from these profiles it implements social information filtering algorithms in order to assign a user to a group of people with similar interests or profiles. The system

recommends items to the user based on the ratings that other people within the same group have give to the (same) item.

Shardanand [5] introduces three social information filtering algorithms the *mean square differences algorithm*, the *Pearson r algorithm* and the *constrained Pearson r algorithm.*

The Mean squared difference (MSD) algorithm measures the amount of variation between two user profiles, user U_x and user U_y, by calculating the mean squared difference between the two profiles. A similarity exists between the users if the MSD value is smaller than a treshold value L. The mean squared difference algorithm can be defined as [5]:

$$MSD = (U_x, U_y) \leq L$$

In contrast, the Pearson r algorithm measures the similarity between users. Predictions can be made by computing the weighted average of users ratings, where the Pearson r coefficients are used as weights. Compared to the mean squared differences algorithm, the Pearson r Algorithm makes use of both negative and positive correlations (associations between users) in order to make predictions. The standard Pearson r algorithm can be defined as [5]:

$$r_{(x,y)} = \frac{\sum U_x U_y}{\sqrt{(\sum U_x^2 \times \sum U_y^2)}}$$

The Constrained Pearson r algorithm is a variation of the Pearson r algorithm which only considers the positive ratings. Due to the nature of the ratings it is possible to identify ratings above and below a median point M. For example; if a rating scale from 1 to 7 is used, then 4 would be the median value: all the values that are below 4 are negative and all those that are above 4 are positive. The constrained Pearson r equation is defined as [5]:

$$r_c = \frac{\sum (U_x - M)(U_y - M)}{\sqrt{\sum (U_x^2 - M) \times (\sum (U_y^2 - M))}}$$

To produce recommendations to a user, the constrained Pearson r algorithm first computes the correlation coefficient between the user and all other users. Then all users whose coefficient is greater than a certain threshold L are

identified. Finally, a weighted average of the ratings of the similar users is computed, where the weight is proportional to the coefficient.

2.2 Implicit information extraction

User profiling makes use of an explicit rating approach. That is, the user has to search for and rate the items which are in turn recommended to other users. One of the biggest drawbacks of this method is the additional tasks (rating items) which are assigned to users. *Implicit rating* or *information extraction* [4] makes use of systems that automatically filter or rate items. These systems derive their filtering conditions from (past and current) user actions or evaluations. The interaction between the user and the system is used to establish the filtering conditions.

Kobsa [4] identifies implicit ratings as user ratings (the same as in user profiles) but, no extra effort is required on the part of the user. He goes on to states that in order to generate implicit ratings, it is necessary to observe users' behavior. Different kinds of information can be extracted or obtained by analyzing the results of the user's browsing or the actual documents themselves. Examples of such information are [4]:

1. Document read time: A positive correlation exists between the time spent reading a document and the reader's assessment of its quality.
2. Documents that the user has bookmarked: Bookmarked items tend to be evidence of strong, rather than marginal interest, so bookmarks set a relatively high threshold for recommendations.
3. Keywords, either as provided by the author or extracted automatically.
4. Text/image ratio, text/image/hyperlink ratio and number of hyperlinks in the current document.
5. Language of the document, including identification of its source language (English, etc.) and stylistic quality.

Implicit rating has traditionally been a non-Internet rating technique which, according to Kobsa [4], was used in traditional publishing services such as:

- Newspapers, magazines and books, which are rated by their editors or publishers, selecting information that they think their readers will want.

- Consumer organizations and trade magazines which evaluate and rate products.
- Published reviews of books, music, theater, films, etc.
- Peer review method of selecting submissions to scientific journals.

2.3 HITS

Hypertext induced topic selection (HITS) or "hubs and authorities algorithm" [3] is a method for extracting information from link structures environments such as the World Wide Web. The basic principle that the algorithm follows is that the importance of a webpage depends on the search query being performed.

Each query has a set of *hubs* and *authorities*. Authorities can be defined as pages which are relevant to the initial query (the most central pages), whereas hubs are pages that have links to multiple relevant authoritative pages [3]. Hubs and authorities have an "mutually reinforcing relationship"; a hub is considered to be "good" if it points to several authorities, in turn an authorities is said to be "good" if it is pointed to by several hubs.

The HITS algorithm is implemented as a three-phase algorithm. In the first phase we have to construct a subgraph of the World Wide Web. The second phase is concerned with extracting the relevant hubs and authorities from the graph structure. The third phase deals with extracting the most relevant hubs and authorities from the available ones.

The first phase of the algorithm is to construct a directed graph $G = (V, E)$, where V is a collection of hyperlinked pages and $(p, q) \in E$ represent a directed edge between p and q. The out-degree, p is the number of nodes a page has links to and q the in-degree represents the number of nodes that have links to a page (from the graph we can create subgraphs) [3].

The aim is to obtain a small set S of the most authoritative pages for a given query string from a text-based search. The first step is to find a set of all pages that contain the query string. Next we search for Γ, which is a collection of the highest ranked pages for the query. The Γ-pages represent the root set which is a "small" set of authorities from which the most relevant are selected to form the base set S

Alg. 1 Subgraph construction

Input:
$R :=$ search engine results
Output:
$S :=$ subgraph of Internet pages
$d :=$ number of central pages

```
1    begin
2        S = R
3        for all p ∈ R
4        begin
5            Γ+(p) = q— p points to q
6            Γ−(p) = p— q points to p
7            ϵ =S ∩ Γ+(p)
8            D = select d pages from Γ−(p)
9            S = ϵ ∪ D
10       end
11   Return S
12   end
```

Once we have the subgraph (Alg. 1), the next step is to extract the authorities based entirely on the analysis of the link structures. This is achieved by ordering pages according to their in-degree. Therefore, pages with higher in-degrees (greater number of pages pointing to them) will be regarded as the best authorities. Similarly, pages with a high out-degree (pages pointing to a greater number of authorities) [3].

An iterative algorithm (Alg. 2) is used to exploit the mutually reinforcing relationship that exists between the hubs and the authorities. The iterative algorithm maintains and update numerical weights. The I-operation is used to update the authority weights and the O-operation is used to update hub weights. Typically, I-operations first sum all the hub values in the pages pointing to p and the O-operation sums all other values in the pages pointed to by p . For every page p one can define a non-negative authoritative weight $x^{(p)}$ and a non-negative hub weight $y^{(p)}$. The weights are normalized such that their squares to sum up to 1. The pages with large x and y values are regarded as the best authorities and hubs.

Alg. 2 Iterative algorithm (G,k).

Input:
G:= subgraph of Internet pages
k:= number of iterations
Output:
$x^{(p)}$:= authority weights after k iterations
$y^{(p)}$:= hub weights after k iterations

```
1    begin
2        x:=[1,1,...,1]
3        y:=[1,1,...,1]
4        for i=1 to k do
5        begin
6            x^(p)=I(x^(p) - 1, y^(p) - 1)
7            y^(p)=O(x^(p), y^(p) - 1)
8            normalize x^(p) and y^(p)
9        end
10       Return(x^(p),y^(p))
11   end
```

The final phase is to identify the most relevant hubs and authorities (Alg. 3). This is achieved by identifying the top c hubs and top c authorities. Where c represents the desired number of hubs and authorities. It is important to note that Alg 3. is the main algorithm which calls the other two algorithms (Alg. 1 and Alg. 2) and ouputs the results.

Alg. 3 Filter(G,k,c)

Input: $(k,c) \in N$ *// k: number of iterations*

```
1    begin
2        (x_k, y_k) := Integrate(G,k)
3        authorities = largest c values for a_k
4        hub = largest c values for h_y
5    end
```

3 Applications (that use social information filtering techniques)

Social information filtering techniques are used in a wide range of applications, ranging from music and movie selection programs to news and adaptive learning applications. In this section we introduce real life applications of the various social information filtering techniques in Section 2.

3.1 Ringo

Ringo [5] is a personal music recommendation system. Ringo recommends a song/artist to a user based upon the prior recommendations of other Ringo users with a similar taste in music. It achieves this by allowing the user to indicate his/her listening preferences by rating 125 songs according to a scale from one ("pass the earplugs") to seven ("one of my few favorites can't live without it").

Ratings constitute the user's personal profile. The profile changes over time as the user makes more ratings. Ringo makes use of the algorithms stated in Section 2.1 to compare user profiles. User profiles are compared in order to identify users with similar interests.

The artist/songs (that are to be rated) are divided into two groups: The first group is a list of all artist/songs which are currently popular. This allows the system to establish a commonality between different users. The second group consists of randomly selected songs from an open database. Apart from rating an artist/song the system also allows users to ask for a prediction on a specific artist/song and post anonymous comments about their ratings.

3.2 SELECT

SELECT [4] is a collaborative information filtering tool that is mainly focused on searching for information from the Internet and Usenet news. SELECT provides Internet users with reliable and interesting information in a quick and easy manner. SELECT therefore spares users from reading unnecessary information and reduces the information overload.

SELECT is targeted at two types of users: users that make use of the Internet in search for specific information and those who use the Internet to keep

up-to-date with what is happening in particular areas.

SELECT achieves its objectives by making use of two techniques: The first technique involves recommendations that are derived from an individual user's past choices. The second technique uses social information filtering in order to identify users with similar tastes and interests and make recommendations to the user based on the recommendations of other users with similar interests.

Both techniques make use of user ratings which may be either implicit or explicit. Explicit ratings are concerned with users giving a value from a particular scale e.g. rating an Internet document on a scale from one ("I hate it") to five ("I love it"). Implicit ratings deal with observing the user in order to obtain rating information e.g. obtaining information about the amount of time the user spends reading a document or what type of documents the user has bookmarked.

3.3 INSPIRE

INSPIRE [2] is a prototype of an adaptive educational hyperlink system (AEHS) [2]. AEHSs try to increase the functionality of multimedia systems by tailoring them to individual learner needs. In addition, AEHSs attempt to minimize the "distortion" and "cognitive overload" which learners have to deal with by helping learners find the most relevant content and guiding them through the lecture material in hyperspace.

INSPIRE includes a learner model, which consists of all the information that the system has gathered about the particular learner. It constitutes the learner's "current state", his/her current position in hypermedia. This position is constantly updated as the learner interacts with the system. It stores items such as the learner's learning goals, different concepts that the learner has studied, assessment of test performances, the amount of time the learner has spent studying and the learner's preferred learning style.

At the beginning of the learning process, the INSPIRE system restricts the domain knowledge (the number of links) a learner has access to, thus allowing novice users to focus on the task at hand without becoming distracted. The number of links and the depth (detail) of the content increases as the learner becomes more familiar with the system.

An AEHS's adaptivity may be of either a *content-level* or *link-level* nature.

Content-level adaptivity is concerned with the dynamic generation of content (learning material) based on the learner – the learner's skills, knowledge, personal traits, learning abilities, and cognitive capacity are identified. Link level adaptivity, on the other hand, assumes a static content and alters the appearance or prominence of the links in the hyperspace.

INSPIRE makes use of both types of adaptivity to achieve navigation and content personalization implemented through the following technologies [2]:

1. Curriculum sequencing: which allows the gradual presentation of the outcome concepts for the learning goal that the learner studies, by making use of information about the learner's progress.

2. Adaptive navigation support: that helps learners navigate in the lesson contents by annotating (interpreting) the links according to learners' progress.

3. Adaptive presentation: which supports various alternative forms of presentation of the educational material pages according to the learning style of the learner.

INSPIRE's lesson structure is organized in a three hierarchical levels of knowledge abstraction' namely learning goals, concepts and educational material. A learning goal is the knowledge which the learner hopes to acquire at the end of the learning session. A learning goal corresponds to a topic of the domain knowledge which is selected by the learner. Concepts are interconnected subsets of the domain knowledge and represent 'assigned qualitative characterizations' such as concept outcomes and are presented as hyperlinks. The educational material that presents the concepts of a learning goal consists of various types of knowledge modules each constituting multiple external representations of the concepts, such as theory. Education material can be further subdivided into three levels (remember, use and find) each corresponding to a learner's performance. Combinations of various types of knowledge modules (theory presentations, examples, exercises, activities using computer simulation, etc.) constitute the pages of the educational material that correspond to each of the three levels of performance for each particular concept.

4 Conclusions

Social information filtering is a process whereby items are recommended to users based on the recommendations of other users with similar interests. Various techniques can be applied in order to achieve social information filtering: rating the items explicitly on a value scale (user profiling), implicitly extracting information form user interactions with the system, and identifying authorities and hubs in order to rank an items relevance.

Given all the various techniques that have been developed, there is no one filtering approach that is best for all users [6]. The best way to proceed in this domain is either to combine the different systems to form one universal information filtering technique or to allow the multiple filtering methods to co-exist and to provide an 'overarching system' that coordinates their outputs such that only the best recommendations (from whatever source or method) are presented to the user [6].

References

[1] M. Balabanovi; and Y. Shoham. Fab: content-based, collaborative recommendation. *Communications of the ACM*, 40(3):66–72, 1997.

[2] C.M. Chen, H.M. Lee, and Y.H. Chen. Personalized e-learning system using item response theory. *Computers & Education*, 44:237–255, 2005.

[3] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.

[4] A. Kobsa and C. Stephanidis. User-tailored information environments. In *Proceedings of the 5th ERCIM Workshop on User Interfaces for All*, pages 23–37. ERCIM (The European Research Consortium for Informatics and Mathematics), 1999.

[5] U. Shardanand and P. Maes. Social information filtering: algorithms for automating "word of mouth". In *Proceedings of the SIGCHI conference on Human factors in computing systems, (CHI '95)*, pages 210–217, New York, NY, USA, 1995. ACM Press.

[6] Y. Z. Wei, L. Moreau, and N. R. Jennings. Recommender systems: a market-based design. In *Proceedings of the second international joint con-*

ference on Autonomous agents and multiagent systems, (AAMAS '03), pages 600–607, New York, NY, USA, 2003. ACM Press.