Digital Library Curriculum Development
Module 7-c: Recommender Systems

(Draft, 12/09/09)

1. **Module title: Recommender Systems**

2. **Module Scope:**
   This module addresses the concepts underlying Recommender Systems, along with the different design approaches, recommenders in current use and challenges.

3. **Learning Objectives**

   The student will be able to:
   a. Articulate how and why recommender systems are used
   b. Demonstrate an understanding of different approaches to modeling recommender systems
   c. Apply this knowledge to the design of recommender systems.
   d. Explain the challenges involved in design of recommender systems.

4. **5S Characteristics of the Module**

   - Stream: Recommendations can be provided for search of various digital objects such as text, images, audio and video.
   - Structure: Recommender Systems aim to structure content present in the digital libraries as a vector of items for each user so as to meet the requirements tailored to an individual and to provide only the items that relate to the user. Recommendations can be structured as hyperlinks to related information.
   - Scenarios: Relates to the process where the user queries his need and related information on the search query is provided to the user based upon his current query term and his profile. Explicit feedback is a scenario where the user provides input of his preferences to the recommender. Implicit feedback is a scenario where the system gathers information about user in an implicit manner through user behavior.
   - Space: Recommender systems estimate the conditional probability of relevant items in a probabilistic space. The similarity between user’s prior search items and the universe of alternatives could also be computed in a vector space. User’s prior search history and a profile describing other characteristics are stored in a physical space.
   - Society : Recommendation seekers, computer programmers, librarians, and recommendation providers

5. Level of effort required

a. Prior Reading : 1 hour
b. In Class Time : 4 hours
c. Out of class exercises : 3 hours

6. Relationships with Other Modules

a. Recommender System should display relevant information that meets the needs of the user and hence this module should be taught after Module 6-a Information Needs/Relevance.

b. Recommender System is a type of Personalization and hence should be taught after 7-g Personalization.

7. Prerequisite Knowledge Required

None

8. Introductory Remedial Instruction

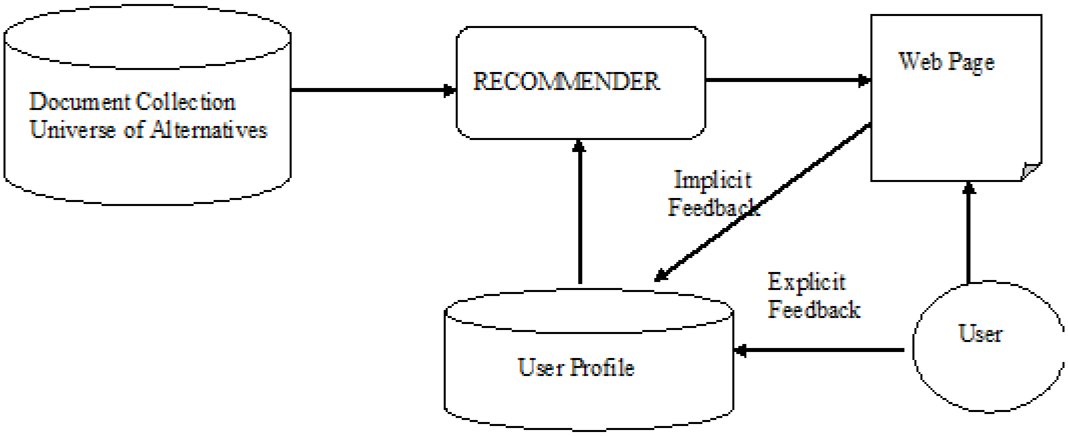
None

9. Body of Knowledge

Topic 1 : Recommender Systems:

Recommender Systems are part of Information Retrieval Systems that attempt to present information that would be of interest to the user. A recommender system would compare the user's profile with that of his/her previous history or the similar profiles of other users with similar interests or background and provide top k references that are relevant to the current item of interest. Recommender systems attempt to reduce information overload and retain customers by selecting a subset of items from a universal set based on user preferences. A preference may reflect an individual mental state concerning a subset of items from the universe of alternatives. Individuals form preferences based on their experience with the relevant items, such as music, games, food etc. The recommendation serves as a filter onto the whole, often inaccessible, universe.

RECOMMENDER SYSTEM



A typical scenario for modern recommendation systems is an online portal with which a user interacts. Typically, a system presents a summary list of items to a user, and the user selects among the items to receive more details on an item or to interact with the item in some way. E-commerce sites like amazon.com present a page with a list of individual products and then allow the user to see more details about a selected product and to purchase the product. A web server typically has a database of items and dynamically constructs web pages with a list of items. Because there are often many more items available in a database than would easily fit on a web page, it is necessary to select a subset of items to display to the user or to determine an order in which to display the items, especially those which might be of interest to the user and also possibly result in a purchase or customer retention/satisfaction for the e-commerce portal.

Recommendation Systems are referred to as Community Filtering in some texts with Item Based Community Filtering relating to the Content Based Approach and User Based Community Filtering mapping to the Collaborative Filtering Approach which would discussed in the following sections. Similarly User based Active Filtering and User Based Passive Filtering refer to the explicit and implicit collection of user data.

Topic 2 – Types of Recommendation Techniques:

Topic 2-a) - Content Based Recommenders(CB):

Content Based Recommender Systems are those that provide recommendations for an item based upon the user's current query item/information need itself and also the user's profile, if it exists. User characteristics are gathered over time and profiled automatically based upon a user's prior feedback and choices. Hence such a system not only retrieves information related to the current item, it also tries to ensure that the retrieved recommendations match the user's preferences.

The content-based approach to recommendation has its roots in the information retrieval (IR) classification community, and employs many of the same techniques. The recommender problem could be stated as extending the text categorization problem using a classifier such as Naïve Bayes. The training set consists of the items that the user found interesting. These items form training instances that all have an attribute. This attribute specifies the class of the item based on either the rating of the user or on implicit evidence.

Given:
A document space X

A fixed set of classes C = $\{c_{1,}\ c_{12,}\ c_{3,\ldots\ldots\ldots\ldots}\ c_j\}$
The set of classes could be binary with only two classes: relevant or nonrelevant.

A training set D of labeled documents with each labeled document belonging to a class in C. User's documents and their corresponding classification based on prior history and explicit feedback are part of the training set.

Using a learning method or learning algorithm, we then wish to learn a classifier γ that maps documents to classes:
$\gamma : X \rightarrow C$

Given: a description $d \in X$ of a document
Determine: $\gamma(d) \in C$, that is, the classes that are most appropriate for d, ordered on basis of probabilistic scores.

Feature selection could then be employed to select only a subset of relevant data in the training set for classification. Thus a Content-Based recommendation system selects items based on the correlation between the content of the items , the user's preferences and other training data.

A pure Content-Based system has disadvantages. Generally, only a very shallow analysis of certain kinds of content can be supplied. Even for text documents the representations capture only certain aspects of the content, and there are many others that would influence a user's experience. For Web pages, for instance, IR techniques often completely ignore aesthetic qualities, all multimedia information (embedded text in images), and network factors such as loading time.

Topic 2-b) - Collaborative Filtering

Collaborative Filtering is a social environment based method of recommendation used to propose items which like-minded users favor (and the active user has not yet seen). These recommendations match a user’s needs based upon information gathered over time from other people having interests matching that of the current user. This approach gives recommendations based on correlation between users. Collaborative Filtering is the pivot of modern day recommender systems. Collaborative Filtering is effective since people’s tastes are typically not orthogonal. A Collaborative Filtering scheme aims to make suggestions to users based upon his/her previous likings and also the preferences of like-minded users i.e. users falling into similar categories/groups/communities as the current user.

The Collaborative Filtering problem could be stated as below:

Let U be the set of users

U = (U_1, U_2……………… ,U_m)

and I be the set of items

I = (I_1, I_2……………… ,I_n)

User U_i could provide a rating for an item I_{ij} either through explicit feedback from that user or rated implicitly through the user’s prior search history, purchase history, browsed pages etc.

Now the aim of a Collaborative Filtering scheme would be to find the list of likely items I_{aj} for the current user denoted by $U_{a..}$The likeliness could be based on prediction which is a numerical value, $P_{a,j}$, expressing the predicted likeliness of item for an active user U_a.

Methods of Collaborative Filtering:

Most of the collaborative filtering algorithms can be divided into two main categories - Memory-based (user-based) and Model-based (item-based) approaches.

i). Memory-based Approach:

Memory-based algorithm uses the entire user-item database for generating a prediction. The most widely used algorithm, k-Nearest Neighbor (kNN) algorithm is as follows:

1) Identify the ratings of the user and represent it as a sparse vector.
2) Define the similarity measure between two sparse vectors. The commonly used measures are (i) the Pearson correlation coefficient which is used in statistics to

measure the degree of correlation between two variables and (ii) the Cosine similarity measure which is used in information retrieval to compare between two documents.

3) Find various users that have rated the item in question and are most similar to the active user. This constitutes the user's neighborhood. (The active/target user is the one to whom a recommendation has to be produced based on prediction).
4) Compute prediction of the active user's rating for the item in question by calculating the weighted average of the ratings given to that item by other users from the neighborhood.

Once a neighborhood of users is formed, these systems use different algorithms to combine the preferences of neighbors to produce a prediction or top-N recommendation for the active user. The variants of this algorithm are (i) k-Nearest Neighbor using the Pearson correlation (kNN Pearson), (ii) kNN using Cosine similarity measure (kNN Cosine) and (iii) the popularity predictor (Popularity).

ii) Model-based Approach:

Model-based collaborative filtering computes the item recommendation using similar approach as its method-based equivalent. But these algorithms take a probabilistic approach and envision the process as computing the expected value of a user prediction, provided his/her ratings on other items. The model is built using different machine learning algorithms such as Bayesian network, clustering, and rule-based approaches.

The Bayesian network model formulates a probabilistic model for the problem. The clustering model interprets this filtering as a classification problem and groups similar users in same class. It estimates the probability that a particular user is in a particular class and hence computes the conditional probability of ratings. The rule-based approach applies association rule discovery algorithms to find association between co-purchased items and then generates item recommendation based on the strength of the association between items. Support Vector Machine is an example of a model-based filtering approach.

iii) Community-Based Collaborative Filtering for Classification

Collaborative filtering can also be interpreted as a form of classification, where classes are formed based on different rating values. Virtually any supervised learning algorithm can be applied to perform classification (i.e. prediction). To predict a rating, we need to classify the item into one of the classes representing rating values. The model based Support Vector Machines (SVMs) are a popular classification technique backing in statistical theory. The standard kNN algorithm (using Pearson and Cosine as the similarity measures) with SVM classifier and SVM regression can also be used for computing classification. If the rating is to be predicted on a continuous scale, regression approach can be used instead of classification.

Challenges in Collaborative Approaches:

Sparse User Feedback:

Sparse user feedback is the single greatest bottleneck in a collaborative filtering scheme. Thus, the number of users maybe small relative to the volume of information in the system. Because there is a very large or rapidly changing database), then there is a danger of the coverage of ratings becoming very sparse, thinning the collection of recommendable items.

Unusual Preferences:

Another problem concerns a user whose tastes are unusual compared to the rest of the population there will not be any other users who are particularly similar, leading to poor recommendations.

Over-specialization:

Overspecialization, sometimes referred to as the ‘banana’ problem, also arises since frequently purchased items, such as bananas in a grocery market basket, will always be recommended. Conversely, some products are seldom bought more than a few times in a lifetime (e.g., locks) and thus suffer from a low number of evaluations.

Cold Start:

This problem concerns the issue that the system cannot draw any inferences for users or items about which it has not yet gathered sufficient information. Hence any new item would suffer from a lack of data and hence would have no positive evaluations. Similarly, when a recommender system starts, it has data from few users.

Topic 2-c) Hybrid Systems

Disadvantages of the Collaborative Filtering and Content Based approaches can be solved by combining the two into a hybrid method. Many hybrid approaches use two recommendation algorithms and combine their results in some manner, such as combining the results by their relevance, mixing the output of the two algorithms, switching from CB into CF once the cold-start phase is over, or using the output of one algorithm as input to the second algorithm.

Hybrid recommendation systems help overcome problems mentioned in the above approaches and can produce outputs which outperform single component systems by combining these multiple techniques. The most common hybridizing methodology is combining different techniques of different types, for example, mixing content based and community filtering approaches. It is also possible to mix different techniques of the same type, like naïve Bayes based Content Based Recommendation plus kNN based Collaborative Filtering

Types of Hybrid Recommender Systems:

Burke (2002) introduced a taxonomy for the hybrid recommendation systems. He classified them into seven categories, weighted, switching, mixed, feature combination, feature augmentation, cascade, and meta-level.

- Weighted hybrid: This hybrid combines scores from each component using linear formula. Therefore, components must be able to produce recommendation scores which can be linearly combinable.

- Switching hybrid: This kind of hybrid is used for selecting one recommender among candidates. This selection is made according to the situation it is experiencing. The criterion for the selection like confidence value or external criteria, should exist, and the components might have different performance with different situations.

- Mixed hybrid: This is a hybrid which is based on the merging and presentation of multiple ranked lists into one. Each component of this hybrid should be able to produce recommendation lists with ranks and the core algorithm of mixed hybrid recommendation merges them into a single ranked list. The issue here is how the new rank scores should be produced. One simple example is adding each rank score like CF_rank (3) + CN_rank (2) → Mixed_rank (5).

- Feature combination hybrid: There exists two very different recommendation components for this hybrid, contributing and actual recommender. The actual recommender works with data modified by the contributing one. The contributing one injects features of one source to the source of the other component. In a feature combination hybrid, collaborative data is treated as a feature and a content-based approach is used on this data.

- Cascade hybrid: The cascade hybrid involves a staged process. In this technique, one recommendation technique is employed first to produce a coarse ranking of candidates and a second technique refines the recommendation from among the candidate set. The secondary recommender is just a tie breaker and does refinements.

- Feature augmentation hybrid: This is similar to the cascade hybrid but different in that the contributor generates new features. The resulting information (ranking or classification) from the first technique is used by the second as an added feature. It is more flexible and adds smaller dimensions than the feature combination method.

- Meta-level hybrid – In the meta-level approach, two recommendation techniques are combined by using the model produced by one as input for the other

A simple example can illustrate the benefits of such hybrid systems. For example, suppose one user has rated the cricket page from cricinfo.com favorably, while another has rated the cricket page from cricnext.com favorably. Pure collaborative filtering would find no match between the two users. However, content analysis can show that the two items are in fact quite similar, thus indicating a match between the users. Such systems analyze the content of items that users rate favorably to build content-based profiles of user interest. They then apply collaborative filtering techniques to identify other users with similar interests.

New items (which are not rated) would be assigned a rating automatically, based on the ratings assigned by the community to other similar items. Item similarity would be determined according to the items' content-based characteristics.

Topic 3: User Profiles:

A profile of the user's interests is used by most recommendation systems. This profile may consist of a number of different types of information reflecting user preferences and behavior. The most important tradeoff to consider in user modeling is to minimize user side effort while maximizing the expressiveness of the representation.
Explicit approaches to user modeling allow the user to retain control over the amount of personal information supplied to the system, but require dedicated investment of time and effort. Implicit approaches, on the other hand, minimize effort, collect huge amounts of data, and make the social element to recommender systems important. Yet they may be inaccurate if users don't logon, or if logs are unclear, and may raise ethical issues.

Explicit Learning:

Explicit Learning requires direct intervention of the user in providing feedback.
Examples of explicit data collection include the following:

- Asking a user to rate an item on a sliding scale.
- Ranking a collection of items qualitatively.
- Presenting the user with binary choices for selection.
- Requesting a list of preferences from the user.

A new user preference elicitation strategy needs to ensure that the user does not:
a) Abandon a lengthy signup process, and
b) Lose interest in re-turning to the site due to the low quality of initial recommendations.

Opinions could be asked of items based upon measures like their popularity or entropy.

Implicit Data Collection:

In implicit feedback the user's interests are inferred by observing the user's actions, which is more convenient for the user but more difficult to implement.

The idea is to gather user preference in a manner transparent to the user by observation, to serve as a substitute for explicit ratings. Most of the techniques for implicitly gathering and exploiting user information are based on methods and algorithms from machine learning and data mining, which attempt to discover relevant patterns or trends from large and diverse data sources. These techniques are largely based on heuristics

For example the construction of the user's profile may be automated by integrating information from other user activities, such as browsing histories. If, for example, a user has been reading information about a particular movie director from a media portal, then the associated recommender system would automatically propose that director's movies releases when the user visits the online video portal.

Examples of implicit data collection include the following:

- Observing the items that a user views in an online store.
- Analyzing item/user viewing times
- Keeping a record of the items that a user purchases online.
- Obtaining a list of items that a user has listened to or watched on his/her computer.
- Analyzing the user's social network and discovering similar likes and dislikes

Recommender Systems also strike a balance between:

1) Long-term modeling:

Long time modeling of a person's evolving interests, preferences, knowledge, goals, and social networks that will be required to help people manage their personal digital libraries during a lifetime of use.

2) Short term user modeling:

Short term user modeling which reflects the current trends in general and also those of the user.

Topic 4: Ubiquitous Computing and Recommender Systems:

Ubiquitous computing is a model of human-computer interaction in which information processing is thoroughly integrated into everyday objects and activities. In ubiquitous computing, access to information sources such as digital libraries is not only possible in many locations, but is dynamically adaptive with regard to a person's location as well as the artifacts and interaction devices and other people in those locations. For example, when visiting a museum, one might automatically be presented with information selected from the museums digital library as one moves through the galleries. Such systems must have adaptation to the shifting context of use, in order to personalize and contextualize information access regarding libraries.

Applications such as tourist and restaurant guides, navigation aids and shopping systems make recommendations based on user activities and behavior patterns. A car navigation system could recommend routes based on the driver's travel patterns such as routes that are more serene, suggest routes that might have food chains matching the user's taste etc.

Websites could make use of location specific information of the user and provide recommendations based upon local choices. Online collaboration tools like Google Wave could provide the names or e-mail contacts of people or groups having expertise in a particular area.

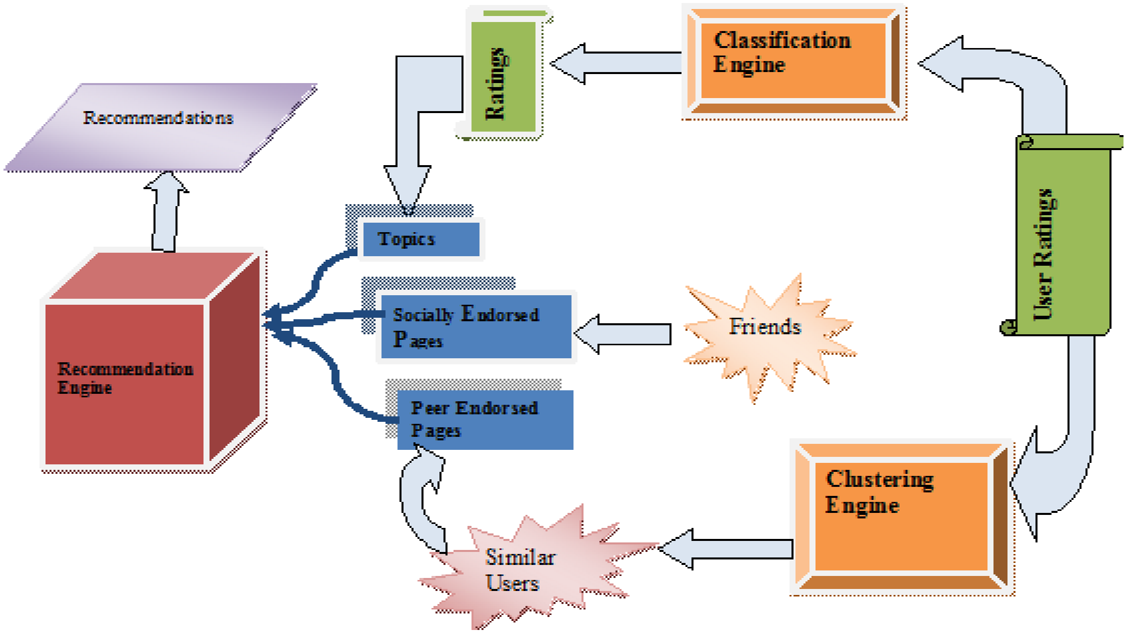
Thus Recommendation systems can help people cope with information overload in ubiquitous computing environments and provide seamless preferences to the user based upon the current context and need.

Topic 5: Recommender Systems in current use:

Amazon:

Amazon.com uses recommendations as a targeted marketing tool in many email campaigns and on most of its Web sites' pages, including the Amazon.com homepage. Clicking on the "Your Recommendations" link leads customers to an area where they can filter their recommendations by product line and subject area, rate the recommended products, rate their previous purchases, and see why items are recommended. Rather than matching the user to similar customers, item-to-item collaborative filtering matches each of the user's purchased and rated items to similar items, then combines those similar items into a recommendation list. To determine the most-similar match for a given item, the algorithm builds a similar-items table by finding items that customers tend to purchase together. The key to item-to-item collaborative filtering's scalability and performance is that it creates the expensive similar-items table offline. The algorithm's online component - looking up similar items for the user's purchases and ratings scales independently of the catalog size or the total number of customers.

Stumble Upon:



Stumble Upon - Architecture

StumbleUpon is an online recommendation portal which integrates peer-to-peer and social networking principles with one-click blogging to create an emergent content referral system. It automates the collection, distribution and review of web content within an intuitive social framework, providing users with a browsing experience similar to surfing channels on television. This architecture scales to millions of users.

StumbleUpon combines collaborative human opinions with machine learning of personal preferences to create virtual communities of like-minded users. Rating websites update a personal profile (weblog) and generates peer networks of users linked by common interest. These social networks coordinate the distribution of web content in a way such that users are rendered pages explicitly recommended by friends and peers. This social content discovery approach automates the "word-of-mouth" referral of peer-approved websites and simplifies web navigation.

Topic 6: Challenges in Recommender Systems:

Interface design:

The recommender systems should have an interface design which provides a good user experience. The User interface should be modeled in such a way the user does not get tired providing explicit feedback and also the recommendation list should be displayed in an uncluttered manner which not only captures the attention of the user but also provides it in a non intrusive fashion.

Amount of Data:

One of the issues facing recommender systems is that they need a lot of data to effectively make recommendations. The industry leaders in recommendations like Google, Amazon, Netflix, etc are those with a lot of consumer user data: A good recommender system initially needs item data (from a catalog or other form), then it captures and analyzes user data (behavioral events), and then the appropriate algorithm is carried out. The more item and user data a recommender system has to work with, the stronger the chances of getting good recommendations.

Unpredictable Items:

There are some items that people either love or hate in equally strong terms. There are movies that the puritans rubbish but the commoner's love. These types of items are difficult to make recommendations on, because the user reaction to them tends to be diverse and unpredictable. Music especially has lot of cases like this where the uses likes both soft rock (MLTR) or heavy metal bands (Metallica).

Dynamically Changing data:

Recommender systems mostly do a long term profiling of users and hence biased towards the old and have difficulty showing new. The past behavior of users would not always be a good tool because the trends are always changing. Hence a simple algorithmic approach would find it difficult to keep up with current trends in fast changing domains such as fashion.

Scalability:

A recommender system would need to make millions of recommendations to millions of users across the globe especially in the case of collaborative filtering algorithms which might need to compute the K nearest neighbors at runtime and hence recommender systems should be scalable across various sizes and types of data and users.

Changing User Preferences:

While a user may have a particular intention when browsing a portal like amazon.com, the next day the user might have a different intention. A classic example is that one day the user might be searching books, but the next day the same user could be searching for house hold appliances. Hence a recommender systems should not take all decisions based on prior content and also should be able to make an intelligent choice based on current context.

Shilling Attacks:

An underhanded and cheap way to increase recommendation frequency is to manipulate or trick the system into doing so. This can be done by having a group of users (human or agent) use the recommender system and provide specially crafted opinions" that cause it to make the desired recommendation more often. For example, it has been shown that a number of book reviews published on Amazon.com are actually written by the author of the book being reviewed. A consumer trying to decide which book to purchase could be misled by such reviews into believing that the book is better than it really is. This is known as shilling attack and recommender systems should protect against these attacks.

Privacy Issues:

Recommender system users who rate items across disjoint domains face a privacy risk. An identity compromise could happen because of the following:

i) Using explanations of recommendations to deduce connections
ii) Combining connection information with other available data to deduce people's personal details.

A recommender system must protect the individual's right to privacy and protect him/her against malicious identity hackers.

10. Resources:

Required Reading:

For Students:

- Smeaton, Alan F. and Callan, Jamie (2005) *Personalisation and recommender systems in digital libraries.* International Journal on Digital Libraries, 57 (4). pp. 299-308. ISSN 1432-1300. DOI : http://doi.acm.org/10.1145/948716.948720
- San-Yih Hwang, Shi-Min Chuang (2004), "*Combining article content and Web usage for literature recommendation in digital libraries*", Online Information Review, Vol.28, No. 4.

  DOI: www.emeraldinsight.com/10.1108/14684520410553750

- Micheline Beaulieu, Pia Borlund, Peter Brusilovsky, Matthew Chalmers,Cliord Lynch, John Riedl, Barry Smyth, Umberto Straccia, and Elaine Toms (May 2003). *Personalisation and recommender systems in digital libraries*, Technical Report Joint DELOS-NSF Working Group.

  http://www.citeulike.org/user/claudioferreira/article/532560

- Belkin, N. J. and Croft, W. B. 1992. Information filtering and information retrieval: two sides of the same coin?. *Commun. ACM*35, 12 (Dec. 1992), 29-38. DOI: http://doi.acm.org/10.1145/138859.138861

- Janet Webster, Seikyung Jung, Jon Herlocker. November 2004. Collaborative Filtering: A New Approach to Searching Digital Libraries. Royaume-Uni. 177-191. DOI: http://www.citeulike.org/group/1732/article/198207

- Badrul Sarwar, George Karypis, Joseph Konstan, and John Reidl. 2001. Item-based collaborative filtering recommendation algorithms. ACM, (April 2001), 285 – 295. DOI: http://doi.acm.org/10.1145/371920.372071.

For Faculty:

- Saverio Perugini, Marcos A. Gonzalves, Edward A. Fox (September 2004). *Recommender Systems Research: A Connection-Centric Survey*, J. Intell. Inf. Syst., Vol. 23, No. 2. pp. 107-143.
  DOI: http://dx.doi.org/10.1023/B:JIIS.0000039532.05533.99

- Greg Linden, Brent Smith, and Jeremy York (January 2003). *Amazon.com recommendations: Item-to-item collaborative filtering*. IEEE Internet Computing, 7(1):76–80.
  DOI: http://doi.ieeecomputersociety.org/10.1109/MIC.2003.1167344

- L. Terveen and W. Hill. Beyond recommender systems: *Helping people help each other*. In *HCI* in the New Millennium, J. Carroll, Ed. Addison Wesley, 2001. DOI: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.26.2437

- Balabanovic, M. and Shoham, Y. (1997). *Fab: Content-Based, Collaborative Recommendation. Communications of the ACM*, 40(3), 66–72.
  DOI: http://doi.acm.org/10.1145/245108.245124

- Baudisch, P. (1999). *Joining Collaborative and Content-based Filtering.* In Proceedings of the ACMCHI Workshopon Interacting with Recommender Systems. Pittsburgh, PA: ACM Press.
  DOI: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.16.2812

- Lynch, C. (2001). *Personalization and Recommender Systems in the Larger Context: New Directions and Research Questions* (Keynote Speech). In Proceedings of the Joint DELOS-NSFWorkshop on Personalisation and Recommender Systems in Digital Libraries (pp. 84–88). Dublin, Ireland.
  DOI: http://www.citeulike.org/group/884/article/634842

- Michael J. Pazzani, Daniel Billsus (May 2007), Content-Based Recommendation Systems, pp. 325-341, The Adaptive Web, Peter Brusilovsky, Alfred Kobsa, Wolfgang Nejdl (Ed.), Lecture Notes in Computer Science, Springer-Verlag, Berlin, Germany, Lecture Notes in Computer Science, Vol. 4321, 978-3-540-72078-2.

DOI: http://dx.doi.org/10.1007/978-3-540-72079-9_10

- Miha Grcar, Blaz Fortuna, Dunja Mladenic, Marko Grobelnik. 2005. kNN Versus SVM in the Collaborative Filtering Framework. WebKDD. 1-2. http://www.springerlink.com/content/p871073356835258/

11. Concept map (To be completed)

12. Exercises/Learning activities

a) Usage and Analysis of a recommendation system in use: (20 minutes)

Amazon is considered a leader in online shopping and particularly recommendations. Develop a class activity which would allow the students to have a hands-on use of a recommender system and gather insights into its working.

Steps to be followed:

i. Visit www.amazon.com and create an account.
ii. Choose a category of items for which you provide ratings. Note how the explicit feedback works in Amazon,
iii. Search for an item in an item category for which you have provided feedback and then search for an item in a category for which you have not provided feedback. How are the recommendations based? What recommendation methods discussed above does Amazon follow.
iv. Is there implicit feedback? If yes, how do you think does the implicit feedback work in Amazon.
v. View the daily sample of recommendations and check if they match your interests.
vi. Prepare a report for the above questions with suitable reasoning.

b) Delicious – Tag Based Recommendations (20 minutes)

Delicious is a Social Bookmarking service, which saves bookmarks online, shares them with other people, and see what other people are bookmarking. It shows the most popular bookmarks being saved across many areas of interest.

In addition, its search and tagging tools helps keep track of a person's entire bookmark collection and finds tasty new bookmarks from similar people.

Do the following:

i. Visit delicious.com. Register and learn about tags and bookmarks.
ii. Explain how delicious uses tags to provide recommendations.
iii. Architecture of Participation is a Web 2.0 concept in which a community of users contributes to the content or to the design and development process. How is the architecture of participation used efficiently in delicious? Discuss about this being an essential part in future web-based recommenders.
iv. Is popularity a good measure of recommendation? Discuss the merits and de-merits
v. Prepare a report on the above.

c) Comparison of Recommendation Engines for multimedia: (25 minutes)

Music and Movie recommenders are gaining popularity over the internet.

Compare and contrast the following recommender systems which provide movie/music recommendations. Comparisons could also include the types of features, the recommendation techniques adopted and user experience.

Tabulate the differences and similarities.

http://www.liveplasma.com/

http://www.pandora.com/ (The music genome project)

http://www.music-map.com/

http://www.whattorent.com/newuser.php

d) Analysis of a constant time adaptability algorithm: Eigentaste 5.0 (25 minutes)

Visit http://eigentaste.berkeley.edu/user/index.php which contains the jester application for recommending personalized jokes.

i) What recommendation technique do you think does this application use based upon your interaction with the application? Why?

The recommendation engine uses an algorithm known as Eigen Taste. The paper could be downloaded at

http://www.ieor.berkeley.edu/~goldberg/pubs/Eigentaste-Info-Retrieval-Journal.pdf. Read the paper and answer the questions below:

ii) What standard techniques does this algorithm follow?

iii) What are the novel approaches adopted by this algorithm?

e) Discussion on Ubiquitous Recommender Systems: (30 minutes)

Note: This discussion would be more appropriate for an audience in the CS or Human Computer Interaction (HCI) domains.

Ubiquitous recommender systems seem to be the future of recommender systems. A simple example of an ubiquitous recommender would be a Global Positioning System (GPS) device which not only give directions to a person driving a car but also provides a list of the user's favorite food chains along the route or alternate routes which would contain these chains.

Involve the class in a discussion based activity on Ubiquitous recommenders. Divide the class into groups. Each group could discuss the different ways in which ubiquitous recommenders could be used with wide ranging applications and also about possible problems (eg. privacy) that could plague these systems.

The groups initially gather information among themselves and afterwards a representative from each group puts forward the views of the group before the class.

A moderator and time keeper are chosen to record, track and monitor these brainstorming sessions.

The various points are ranked by the moderator and the outcome of the discussion pasted on the class wiki page.

f) User based Filtering (15 minutes)

Everyone's a Critic (EaC) is a film community website which is based on collaborative filtering algorithm in order to obtain the film recommendations from people who share similar tastes in film.

Develop a class activity which would allow the students to have a hands-on use of recommender systems based on user-based filtering algorithm and gather insights into its working.

Steps to be followed:

i) Create an account in http://www.everyonesacritic.net/
ii) Rate the movies based on these three methods – category, movie title, browsing movies.
iii) Select "Find critics like me" to view the list of critics who have rated the movies similar to the user.
iv) Note the type of recommendation technique employed by this recommender system: content based or collaborative filtering. Why is this approach user-based? And note if the feedback implicit or explicit.
v) Note if the list of movies in the selected critic's list contains movies not rated by the user matching the user's interest.
vi) Note what happens when the user tries to search for similar critics before rating any movies. How is this related to challenges of collaborative filtering. Suggest some solution for this problem.

g) Item based Filtering (15 minutes)

Pandora is a recommender system that uses item-based collaboration filtering approach. The idea is to categorize music based on its attributes. It calculates the similarity of item like a piece of music in terms of its genetic makeup.

Develop a class activity which would allow the students to have a hands-on use of recommender systems based on item-based filtering algorithm and perceive clear understanding on its working.

Steps to be followed:

Create an account in http://www.pandora.com/

Create a new station by typing a favorite artist or song.

- Note the type of collaborative filtering employed by this recommender system: user-based or item-based (as discussed in section 8). And note if the feedback implicit or explicit.

- Look for the songs recommended by the system. Note if there is similarity among the playlist generated based on the attributes of the song.

h) Compare and Contrast Methodologies (20 minutes)

Compare and contrast the approaches used in exercises g and h. Provide a case-study on any two recommender systems (like amazon.com, ebay.com, netflix.com etc.) other than EaC and Pandora.

In class, break the students into groups of 3 or 4. Have them discuss the differences and challenges in the methodologies used in these systems and provide suggestions to improve the technique. Each group prepares a Word document or PowerPoint slides (5 minutes) and presents it to the class (10 minutes).

13. Evaluation of learning achievement

In their answers to the learning exercises, students demonstrate an understanding of

- Need for recommender systems
- Different approaches to modeling recommender systems
- Advantages and Disadvantages of the approaches
- Alternate design strategies
- Current and future use of recommender systems.

14. Glossary

- CB Recommender – Content Based Recommender
- CF - Collaborative Filtering
- DL – Digital Library
- WWW – World Wide Web
- GPS – Global Positioning Service.
- Scalability: The ability and ease with which a service may increase in size; where size may be defined as number of users, throughput of questions and responses, etc.
- Probabilistic model: a document retrieval model based on a probabilistic interpretation of document relevance (to a given user query).
- Vector space model is an algebraic model for objects as vectors of identifiers, such as, for example, index terms.

15. Additional Useful Links

- Marko A. Rodriguez, David W. Allen, Joshua Shinavier, and Gary Ebersole. *A recommender system to support the scholarly communication process*. May 2009. DOI: http://www.citeulike.org/group/4576/article/4505139

- Ramakrishnan, N., Keller, B.J., Mirza, B.J., Grama, A.Y., and Karypis, G. (2001). Privacy Risks in Recommender Systems. IEEE Internet Computing, 5(6), 54–62. DOI: http://doi.ieeecomputersociety.org/10.1109/4236.96883

- Stumble Upon Free web-browser extension which acts as an intelligent browsing tool   available at:
www.stumbleupon.com

- Real-time Collaborative Filtering at
http://www.timelydevelopment.com/demos/RealtimeCollaborativeFiltering.aspx

- Synthese Recommender Home at
http://lab.cisti-icist.nrc-cnrc.gc.ca/synthese/synthesemain

- Jester's Eigentaste algorithm at
http://eigentaste.berkeley.edu/user/index.php

- Pandora music recommender at
http://www.pandora.com/

- E-commerce Recommendation System at
http://www.bridgewell.com/ec%20portal.html

16. Contributors

Authors:

Ashwin Palani

Dr Edward Fox

Seungwon Yang

Venkatasubramaniam Ganesan

Reviewers:

Nagarajan

Ashwin K
Seungwon Yang
John Ewers