An Extensive Survey on Information Retrieval and Information Recommendation Algorithms Implemented in User Personalization

INTRODUCTION

In this modern world, e-commerce websites play a major role in reaching a product to the customer. With hundreds of e-commerce stores coming up every day, the competition among the variety of stores were also increased. As a trend of current scenario each of the e-commerce websites works on targeting their customers to reach out a particular product. For to target the customers the e-commerce sites have to work on user personalization, which creates a database holding the customer’s interest values. User personalization in turn supports effective information retrieval and information recommendation.

Information Retrieval consists of mainly four stages such as identifying the exact subject of search, locating the subject, locating documents and locating the information in the document. It is also called as information management which means establishing and controlling the information.

Information recommendation is an art of personalization. It means comparing the similarities between the user profile and the unseen document set. The main aim is to provide the user what they need without asking explicitly. Based on the constructed profile vector the system is able to provide the information what the user actually wants.

In this paper, the survey is made on personalization in which many works have done by different researchers. This survey is mainly based on the algorithm used in the information retrieval and information recommendation and how the algorithm works and the comparison between those algorithms are made and the advantages and disadvantages are analysed.

2. Information Retrieval Algorithms:

2.1 Boolean Algorithm:

Boolean Algorithm is a traditional model used in information retrieval. Boolean algorithm is based on Boolean expressions which are formed from user queries. In some systems those are provided as expressions directly and in some systems users enter as natural languages which are converted to Boolean expressions. They are evaluated to form results. Boolean expressions are built from queries which are translated into set of documents. Let \( U \) represent the names of all documents stored. Let \( D_1 \) and \( D_2 \) represent the names of those documents that contain patterns \( P_1 \) and \( P_2 \), respectively. The following list defines how to evaluate Boolean expression operators in terms of the sets:

1. \( U \cap D_1 \) is the set of all documents not containing \( P_1 \) (not).
2. \( D_1 \cap D_2 \) is the set of all documents containing both \( P_1 \) and \( P_2 \) (and).
3. \( D_1 \cup D_2 \) is the set of all documents containing either \( P_1 \) or \( P_2 \) (or).

Keywords:

© 2015 AENSI Publisher All rights reserved.

Corresponding Author: Mr. N. Yuvaraj, Prof / CSE, KPR Institute of Engineering and Technology, Coimbatore
E-mail: yuvaitksr@gmail.com
4. $D_1 \cup D_2 - D_1 \cap D_2$ is the set of all documents containing either $P_1$ or $P_2$ but not both (xor).

Each document is assigned a n-bit binary number which are stored in BBC structure. The search will proceed from left to right processing each term to shrink and result in final answer.

2.2 Ranking Algorithm:

2.2.1 AdaRank Algorithm:

A novel algorithm which can optimize a loss function based performance measures of IR is referred to as ‘AdaRank’ Algorithm. It takes a training set as input and the performance measure function and the number of iterations as parameters. It runs many rounds and at each round it creates a weak ranker. At last it gives the output combining the weak rankers. Each round, it maintains a distribution of weights over the queries in the training data. Initially, AdaRank sets equal weights to the queries. It increases weights for queries which are not ranked. Weak ranker is constructed based on training data with weight distribution . The goodness of a weak ranker is measured by the performance measure.

2.2.2 Page Rank Algorithm:

PageRank algorithm is developed by Brin and Page. It is used by google search engine. this algorithm depends upon link structure of the web pages. The concept is based on important links towards it then page linked to that is considered as important. The Page Rank considers the back link in deciding the rank score. If back links addition is large then the page is given high rank. It computes relevance of web pages than counting the number of pages linking to it. If a back link comes from an important page, that is given a higher weighting than those backlinks comes from non-important pages. In a simple way, link from one page to another page may be considered as a vote. Based on the vote ranking is done.

2.2.3 Weighted Page Rank:

Weighted PageRank Algorithm is proposed by Wenpu Xing and Ali Ghorbani. WPR is the modification of the original PageRank algorithm. It decides the rank score by considering the importance of both the inlinks and outlinks of the pages. High value of rank is provided for more popular pages and does not equally divide the rank of a page among it’s outlink pages. Every out-link page is given a rank value based on observing its number of in-links and out-links. The performance of WPR is to be tested by using different websites and future work include to calculate the rank score by utilizing reference page list and increasing the number of human user to classify the web pages.

2.3 HITS Algorithm:

The HITS algorithm is proposed by Kleinberg in 1988. It identifies two different forms of Web pages called hubs and authorities. Authorities are pages having important contents and hubs are pages that act as resource lists. Always, good hub page for a subject points to many authoritative pages on that content, and vice versa. The page may be a good hub and a good authority at the same time which leads to the definition of an iterative algorithm called HITS (Hyperlink Induced Topic Selection). HITS algorithm is ranking the web page by using inlinks and outlinks of the web pages. The web pages are named as authority and hubs based on many and various hyper links respectively. HITS is, technically, a link based algorithm. It concentrates on the structure of the web only, neglecting their textual contents. It has some problems such as wrong decision in ranking pages based on query, Topic Drift occurs when the hub has multiple topics of equivalent weights, less efficiency, irrelevant links, Mutually effective relationship between hosts. Because of these above problems, the HITS algorithm is not preferred to be used in Google search engine. So PageRank algorithm is used in Google search engine because of its preference and efficiency.

2.4 Distance Rank Algorithm:

A distance rank algorithm is proposed by Ali Mohammad Zarheh Bidoki and Nasser Yazdani. This algorithm, concentrates on the distance between the pages to compute rank of web pages in search engine. It follows, Shortest logistic method that is the page will small distance will be given the higher rank. The Advantage is that, it can find pages quickly which have high quality compared to other algorithms. The limitation for this algorithm is that the crawler should perform a large calculation to calculate the distance vector, if new page is inserted between the two pages. This Distance Rank algorithm adopts the PageRank properties i.e. the rank of each page is computed as the weighted sum of ranks of all incoming pages to that particular page. Then, a page has a high rank value if it has more incoming links on a page.

2.5 EigenRumor Algorithm:

The EigenRumor algorithm is proposed by Ko Fujimura that ranks each blog entry on basis of weighting the hub and authority scores of the bloggers based on eigenvector calculations. This assigns high score for the blog entered by good blogger and not based on the blogger’s prior work. Page rank and HITS provides rank value to the blogs but some issues arise, if these two algorithms are applied directly to the blogs then the number of links to a blog entry is generally very small, Some time is needed to develop a number of in-links. The rank scores of blog entries as decided by the page rank algorithm is often very low so it cannot allow blog
entries to be provided by rank score according to their importance. These are resolved by EigenRumor algorithm that has similarities to PageRank and HITS in that all are based on eigenvector calculation of the adjacency matrix of the links. This model is constructed from agent-to-object links and not page-to-page links. Agent is the blogger and object refers to blog entity. Using the EigenRumor algorithm, the hub and authority scores are calculated as attributes of agents (bloggers) and the inducement of a blog entity that does not yet have any in-link entered by the blogger can be computed.

2.6 Vector Space Model:

The vector space model overcomes the drawback in Boolean model which projects a framework in which partial matching is possible. The degree of similarity is calculated based on index terms in queries and documents. Decreasing order of degree of similarity gives the ranked documents with partial match. For the vector model, the weight associated with a pair is positive and non-binary and the index terms in the query are also weighted. The vector model proposes to evaluate the degree of similarity of the document with regard to the query has the correlation between the vectors. This correlation can be measured by the cosine of the angle between these two vectors. The factor provides normalization and does not affect the ranking because it is the same for all documents. A document might be retrieved even if it matches the query only partially. In this model, the frequency of a term inside a document referred as the factor that provides measure of how well that term describes the document contents. The success of vector space model lies in its partial matching strategy and similarity measure.

3. Information Recommendation Algorithms:

3.1 Content-Boosted Collaborative Filtering:

Content-boosted collaborative filtering is done by creating a pseudo user-ratings vector for each user in the database. The pseudo user-ratings vector consists of item ratings by user and ratings predicted by content based predictor. The pseudo user-ratings vectors of all users put together give the dense pseudo ratings matrix. Based on the dense matrix, similarity between the active user and another user is computed using the Pearson correlation. The original votes are substituted by pseudo user ratings.

Harmonic Mean Weighting is used for computing the accuracy of pseudo user-ratings vector that depends on the number of ratings provided. Based on the user ratings the content-based predictions are done. If the rating is high then the content-based prediction is good and pseudo user-ratings are accurate and if ratings are less then pseudo user-ratings are not accurate. Inaccuracies in pseudo user-ratings yield misleadingly high correlation between active user and other users. To incorporate confidence we use Harmonic mean weighting factor. The highest weight is provided for correlation between pseudo user-ratings of 50 users rated items. If one of the pseudo user-rating vectors is based on less than 50 user-rated items, the correlation will be devalued. By using the 10-fold cross-validation we evaluate the performance of content-based factor which is 50 given above. Performance test is made by generating a learning curve on testing the system after training all data sets. If the predictor is given more and more training then the prediction performance increases and beyond the point of threshold is the point of diminishing returns.

Self Weighting is a prediction for the active user, computed by sum of the mean centered votes of the best-n neighbours of that user. To increase the confidence we place in the pure-content predictions for the active user. Self Weighting factor is used for final prediction. Producing Predictions Combining the above two weighting schemes, the final CBCF prediction for the active user and item is produced.

3.2 Item-based Collaborative Filtering Algorithm:

Item-based recommendation algorithms are used for producing predictions to users. It is same as the user-based collaborative filtering algorithm. The item-based approach counts on set of items target user has rated and it compares that with the target item to compute the similar items. Once the similarity is found then prediction is computed using weighted average of the target users rating on similar items. It is describes by similarity computation and prediction generation.

Item Similarity Computation is a challenging stem to find the similarity between items and to compute the match. The similarity is computed based on finding the co-rated items. There are different ways to compute similarities; they are cosine-based similarity, correlation-based similarity and adjusted-cosine similarity.

1. Cosine-based Similarity is measured by computing the cosine of angle between two vectors. It is denoted as, \( \text{sim}(i, j) = \cos(\mathbf{i}, \mathbf{j}) \) where \( \mathbf{i} \) and \( \mathbf{j} \) are the vectors.

2. Correlation-based Similarity means computing the similarity based on the pearson correlation between two items. To make the correlation computation accurate we must first isolate the co-rated cases.

3. Adjusted Cosine Similarity is fundamental difference between the similarity computation in user-based CF and item-based CF. The user-based CF is computed along the rows of the matrix and item-based CF is computed based on the columns of the matrix. The drawback in the cosine measure that is, difference in rating scale is not measured that has overcome by the Adjusted cosine similarity by subtracting ranking of items similar to prediction.
3.2 Prediction Computation:

Prediction Computation is an important step in a collaborative filtering system that is to generate the result based on the prediction, once the similarities are measured prediction is done based on the user ratings. There are two techniques given as follows:

1. Weighted Sum: The prediction is done by computing the sum of ratings given by the user for particular item to similar items. The weighted sum is scaled by sum of similarities to make sure the prediction is in the predefined range.

2. Regression: It is similar to weighted sum but it uses approximation rates instead of direct ratings based on regression model.

3.3 Entropy Based Collaborative Filtering Algorithm (EBCFA):

Entropy Based Collaborative Filtering Recommender System, every user in the system other than active user is a prospective mentor to the active user. When rating for a target item is estimated, it searches the ratings matrix to find all the users rated for the item if nobody has rated then it terminates resulting rating prediction for active user for the target item is not possible since nobody has ranked. If it has one or two ratings then all the users rated will be preliminarily qualified mentors. For each preliminarily qualified mentor in the set, number of items rated by active user and predictability index towards active user for target item are examined. All those users whose number of commonly rated items Pls at the appropriate level are above the respective threshold levels form the set of fully qualified mentors for the active user. If there is no fully qualified mentor then the algorithm terminates saying no mentors were found for the active user. If there is at least one fully qualified mentor then the subjective probabilities assigned by each of the fully qualified mentors to each of the rating levels are weighted by their respective predictability indices and aggregated to obtain the overall probability of each of the rating levels for the active user. On an n point rating scale, we get n overall probability values for the active user corresponding to each of the n rating labels; the sum of which will be at most equal to unity . The final estimation of the rating for the active user is then done by multiplying each rating label with the corresponding probability and then by summing up all these products.

3.4 Neighbourhood-Aware Matrix Factorization:

Neighbourhood-aware matrix factorization is a part of regularized matrix factorization (RMF) in a neighbourhood-based model. This algorithm does three prediction such as user- neighbourhood model, item- neighbourhood model and both. The resultant is the prediction of three algorithms.

User-neighbourhood model is determined by computing the similarity of two users by the Pearson correlation based on list of items rated by both users. There are some problems in pearson correlation, the main problem is when the ratings are small for user pairs then correlation between these ratings is unreliable. In some datasets, there will be availability of more user, so the reliable correlation can be computed easily. Size of correlation matrix is another problem in computing the correlation between the users. For the Netflix dataset where the number of users is around 480,000, the whole matrix needs about one TByte of memory. This problem is solved by storing for each user only the correlations with the other users with highest correlation. Rating prediction is computed based on the weighted sum of ratings of best correlating users given by predicted rating of RMF model. Squashing function helps in computing the weighting coefficient from pearson correlation. Item-neighborhood model, the same principle is applied on the item side. Correlations are rated between the items and the highest correlation is stored and the weighting coefficient is computed.

Combining the information means prediction is made based on the both user neighborhood model and the item neighborhood model. Predictive accuracy depends mostly on the number of ratings from training data. The weighted sum of both is used for the prediction. Training schedule can be summarized as, correlations between users and correlations between items are computed. Based on the shrunken correlation and computed correlation best correlating user/item is computed. Once this is done, the predictions of the neighborhood models can be computed very efficiently. Once the training is done, evaluation of individuals is very fast and optimization step can be done efficiently.

5. Conclusion:

This paper is mainly concludes about the algorithms used in information retrieval and information recommendation. These are the algorithms generally used and researched upon. Many approaches are being used as a process for improving the efficiency of personalization. The aim of this paper is to discuss about the algorithms in detail. This clearly explains about the algorithms, comparisons and the limitations of the algorithms. This paper can be used by the researchers to get a clear idea about various information retrieval and information recommendation algorithms while implementing and enhancing user personalization. E-commerce site developers can have an analysis over the different set of algorithms and its advantages and disadvantages to make a decision over the algorithm to be implemented. Right choice of algorithms during e-commerce site development can make the task of targeting the customer much simpler.
### 4. Comparison:

<table>
<thead>
<tr>
<th>NO</th>
<th>ALGORITHM</th>
<th>ADVANTAGES</th>
<th>DISADVANTAGES</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Boolean Algorithm</td>
<td>It is simple and easy to implement</td>
<td>There is no concept of ranking, gives only the exact match and cannot afford to give the correct logical operations</td>
</tr>
<tr>
<td>2</td>
<td>AdaRank Algorithm</td>
<td>AdaRank can incorporate any performance measure. It employs a more reasonable framework for performing the ranking task.</td>
<td>Ranking the most relevant documents on the tops of document lists is crucial for document retrieval.</td>
</tr>
<tr>
<td>3</td>
<td>Page Rank Algorithm</td>
<td>For small set of pages it is easy to calculate. It provides efficient output.</td>
<td>Time consumption for large number of pages</td>
</tr>
<tr>
<td>4</td>
<td>Weighted Page Rank</td>
<td>It totally depends on the link structure. It is target oriented. User cannot intentionally increase the rank.</td>
<td>The periodic crawling of web-servers is difficult. So specialized crawlers need to be designed for fetching information.</td>
</tr>
<tr>
<td>5</td>
<td>HITS Algorithm</td>
<td>HITS is sensitive to user query. Important pages are obtained on basis of calculated authority and hubs value.</td>
<td>Inefficiency. Irrelevant links. Mutually effective relationship between hosts</td>
</tr>
<tr>
<td>6</td>
<td>Distance Rank Algorithm</td>
<td>It is less sensitive, finds pages with high quality and more quickly with the use of distance.</td>
<td>The crawler should perform a large calculation if new page is inserted between the two pages</td>
</tr>
<tr>
<td>7</td>
<td>EigenRumor Algorithm</td>
<td>Page rank is slow in ranking the blogs. It resolves these issues by ranking the blogs.</td>
<td>More detail analysis on the durability of spamming is also an important.</td>
</tr>
<tr>
<td>8</td>
<td>Vector Space Model</td>
<td>Simple model based on linear algebra. Allows computing a continuous degree of similarity between queries and documents. Allows ranking documents according to their possible relevance. Allows partial matching.</td>
<td>Mutual independence of index terms has said to be disadvantage of vector space model but practically, consideration of term dependencies is not fruitful.</td>
</tr>
</tbody>
</table>

### INFORMATION RECOMMENDATION

<table>
<thead>
<tr>
<th>ALGORITHM</th>
<th>ADVANTAGES</th>
<th>DISADVANTAGES</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>Content-Boosted Collaborative Filtering</td>
<td>Collaborative Filtering performs better than a pure content based predictor, collaborative filtering, and a naive hybrid of the two.</td>
</tr>
<tr>
<td>10</td>
<td>Item-based Collaborative Filtering Algorithm</td>
<td>Item-based similarity is more static and allows us to precompute the item neighborhood. It has certain performance benefits.</td>
</tr>
<tr>
<td>11</td>
<td>Entropy Based Collaborative Filtering Algorithm (EBCFA)</td>
<td>Better quality, some tradeoff in terms of response time and memory requirement may be justified.</td>
</tr>
<tr>
<td>12</td>
<td>Neighbourhood-Aware Matrix Factorization</td>
<td>Memory usage scales linearly with the number of users or items as compared to a quadratic scaling of most other algorithms.</td>
</tr>
</tbody>
</table>

### REFERENCES


Tao Tan, Leting Tan, “Study on Personalization Recommendation System Based on Recruitment Information” – Elsevier Tao Tan et al. nTida nL eetti nagl. /T Parno /c Pedroic eEdnaig iEnenegriinnege r08n g (2 2091 1(2) 01020)– 078000 – 784.


