VOL. 10, NO. 6, APRIL 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

A SURVEY ON COLLABORATING TECHNIQUES AND QOS BASED RECOMMENDATION SYSTEM

N. Kannammal[1], S.Vijayan[1] and R. Sathishkumar[2]
[1]Surya Engineering College, Erode, Tamilnadu, India
[2]Hindusthan College of Engineering and Technology, Coimbatore, Tamilnadu, India
E-Mail: n.kannammal@yahoo.in

ABSTRACT

The immense growth in internet technologies leads to increase in size of the web service repository. Normally, all the service requestor expects very qualitative resultant web service for their request. These requestors may not have previous knowledge about their requesting domain. So it is difficult for them to filter out the relevant web service from huge pool of data. Moreover, the resulting of irrelevant services for the user request will affect the user satisfaction. Recommender system is being widely used to recommend products or items to consumer. This system can also be used to recommend a service or a list of service to service requestor. Collaborative filtering technique (CF) is one the efficient recommending system that recommends the service based on the past users experiences or ratings on that service. The past users are the nearest neighbors to the requestors. Traditional CF does the user-based and item-based similarity computation between the users and items for recommendation. They do not take into account nonfunctional components (QOS parameters) of the service which greatly have impact on performance. This paper is a review about CF technique and need of QOS parameter for the recommendation system to improve the performance.

Keywords: recommendation system, QOS parameter, collaborative technique.

INTRODUCTION

Web services are modular, self describing applications that are described, published by service providers and invoked by service requestors. They are important in business application. Web services can be dynamically deployed and invoked. Web mining enables Web based businesses to provide the best access routes to services or other advertisements. When a company advertises for services provided by other companies, the usage mining data allows for the most effective access paths to these portalsare personalized based on the characteristics (interests, social category, context ...) of an individual.

As there are many web services with the same functionality it is difficult to find appropriate web services based on functional component. Users must be further assisted in selecting the relevant WS for their needs. Sometimes a web service which does not satisfy the user query may be a important service. But due to a mere keyword search it may not be in the resultant set. So explicit user preference is needed to specify

The proliferation of web service in WWW demands very effective selection methods. These selection methods are effectively used to recommend optimal service suitable for their request [1]. Recommender system is a information filtering technique. It plays a vital role for web mining from a pool of services. The explosive growth of e-commerce has led to the development of recommender system [2].

Recommendation system will identify the user like or dislike towards a particular item. The system will predict a item or recommend a set of items that user may prefer [3]. The suggestions given by the system increase its application in ecommerce and business. Several techniques are available that direct the users towards a solution as the searching space is very large.

Recommendation system are of several technique namely Collaborative, Content-based, Demographic, Utility-based and Knowledge-based [4], [5], [6].
They are given in the following table:

Table-1. Recommendation techniques.

Types of recommendation technique	Process
1. Collaborative	Identify similar users to active users and predict or recommend from their ratings
2. Content-based	Users rating for the items are the input. Using the features of the items it generate a classifier that fits the user rating and used it on item
3. Demographic	Based on demographic information of users similar users are identified and recommended using their ratings,
4.Utility-based	A utility function over the item that describes the user preference is given. Using the features of the item and on applying utility function item is ranked.
5. Knowledge-based	Features of the item and user preference is taken as Input and on matching both recommendation is given

Among the above recommendation technique, collaborative technique is most widely used. It works based on the historical information on users and the items. It is successfully used in marketing of a product to improve the sales of a product. This paper provides an insight into various collaborative techniques and impact of recommendation based on QOS properties of services.

Collaborative techniques are one of the recommender techniques that recommend services to the current user by automatically extracting preferences from similar users to current users [7]. Not all the users of a particular item are similar users. Those users who have same experiences on same set of service invocation are similar users [1].

The basic assumption is that if x and y are two users *experiencing* similarly rating will also have same behavior to other items [8]. CFtechniques use a database of preferences for items by users to predict additional topics or products a new user [9]

In a typical *CF* scenario, let m be list of users {u1, u2…um} and n be a list of items (services) {s1, s2… ….sn} and each user have rated and some may not have rated. The ratings can either be explicit indications, and so forth, on a 1-5 scale [10]. The user-item matrix is given below

Table-2. A simple example of ratings matrix.

Users /Items (services)	S1	S2	S3	S4
U1	4	?	5	5
U2	4	2	1	?
U3	3	2	4	?
U4	4	4	?	?
U5	2	1	3	5

The numbers are ratings of user on item (services) and some of the values are missing value (?) which leads to sparsely db.

CF technique is categorized into

(a) Memorybased (b) model based (c) hybrid approaches

Memory based CF will generate predictions from sample of user-item database with ratings. Model based CF will use a model of datamining and machine learning algorithm to predict the suggestions. There are some limitations in memory based algorithms. It always depends on the ratings provided on common items and so it is unreliable when database is sparse. To overcome this, the model based algorithm design a model which will learn to predict [11].

Another important class of recommendation system is content based system which will consider the textual information and find out the regularities and patterns. From the content, only uses the user-item ratings data to make predictions and recommendations, while content-based recommender systems rely on the features of users and items for predictions [12].

But both the CF technique and Content-base technique have limitations that CF do not include the features explicitly whereas content-based do not necessarily incorporate the information of preference in similarity across individuals [13].

Hybrid *CF* techniques which is combination of (a) CF techniques, (b) CF techniques and other recommendation techniques (c) multilevel combination of both the techniques are designed to overcome the shortcomings of CF and content-based systems [14] .

CHALLENGES OF COLLABORATIVE TECHNIQUES

All the recommendation technique work dynamically on time to time changing environment.

VOL. 10, NO. 6, APRIL 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

Efficient recommendation techniques are to overcome the following challenges of CF techniques [15].

Data sparsity: The user or item rating matrix is the input for CF technique. As it is filled with ratings, many entries are null for which the user has not rated. This led the database to sparse which reduce the prediction performance.

Cold start problem- when a new user has entered to db there won't be any history created (also called new user problem or new item problem).

Scalability: As the number of existing users and items grow tremendously the db will expand and shrink from which the ratings are taken to predict.

Synonymy: Usually there are items with the same characteristics but with different names or entries. This item though they are similar and some time very near to active user they are not considered due to lack of meaning

Gray sheep: There are users whose opinion will not be similar with anyone. So it is difficult for them to give recommendation as well as to take account their ratings.

Shilling attacks: As recommendation system is mainly used for business improvement of a product, the users are going to be their competitors. There is possibility of giving negative ratings for the opponent's item. Similarly, strongly rating their own product. This may lead to false prediction performance result.

Other challenges: Noisy data and explain ability are other challenges.

Fast and accuracy of the recommendation system is decided based on dealing this challenge.

COLLABORATIVE TECHNIQUES

a) Memory-based collaborative filtering techniques

Memory-based *CF* algorithms use the entire or a sample of the user-item database to generate a prediction. Every user ispart of a group of people with similar interests. By identifying the so-called neighbors of a new user (or active user), a prediction of preferences on new items for him or her can be produced.

Neighborhood-based *CF* algorithm

This algorithm calculates the similarity measure between two similar users. The following are the steps involved [16].

Step-1: Calculate the similarity *wi*, *j*, which reflects distance, correlation, or weight, between two users or two items, *i* and*j*.

Step-2: Produce a prediction using weighted average of all the ratings (or simple weighted average) of the user or item on a certain item or user.

Top-N recommendations

Step-1: Find k most similar users or items by computing similarities between them
Step-2: Aggregate the nearest neighbor.
Step-3: Sort them and recommend top- N in the list

1. Similarity computation

Pearson correlation will calculate linear relation between two variables [17] [18] [19]. Two types are user-based and item-based computation.

User-based: The correlation between two users u and v is

$$w_{u,v} = \frac{\sum_{i\in I}\left(r_{u,i}-\bar{r}_u\right)\left(r_{v,i}-\bar{r}_v\right)}{\sqrt{\sum_{i\in I}\left(r_{u,i}-\bar{r}_u\right)^2}\sqrt{\sum_{i\in I}\left(r_{v,i}-\bar{r}_v\right)^2}}$$

where

i∈*I* that summations are over the items that both the users *u* and *v* have rated
*ru*is the average rating of the co-rated items of the *u*th user.

Item-based: The set of users$u \in U$ who rated both items *i* and *j*, then the *Pearson Correlation* [19] will be

$$w_{i,j} = \frac{\sum_{u\in U}\left(r_{u,i}-\bar{r}_i\right)\left(r_{u,j}-\bar{r}_j\right)}{\sqrt{\sum_{u\in U}\left(r_{u,i}-\bar{r}_i\right)^2}\sqrt{\sum_{u\in U}\left(r_{u,j}-\bar{r}_j\right)^2}}$$

Types of Pearson similarity

Constrained Pearson: correlation: use midpoint instead of mean rate.

Spearman rank correlation: ratings are ranks

Kendall's correlation: relative ranking is used [20] [21].

For user-based approach, the similarity is found between two users who have rated or experienced same items. They are co-rated users. For item-based approach, similarity between the items for which a user has rated is computed. They are corated items [22].

Vector Cosine-Based Similarity. This computes the similarity between two documents. The word frequencies are stored as vectors. The cosine angle for the frequency vectors is computed. Formally, if *R* is the$m \times n$ user-item matrix, then the similarity between two items, *i* and *j*, is defined as the cosine of the *n* dimensional vectors corresponding to the *i*th and *j*th column of matrix *R*.

Vector cosine similarity between items *i* and *j* is given by

$$w_{i,j} = \cos\left(\vec{i}, \vec{j}\right) = \frac{\vec{i} \bullet \vec{j}}{\|\vec{i}\| * \|\vec{j}\|}$$

As the, vector formed from word frequency and user rating may be in different scale, vector cosine formulae is adjusted by subtracting the corresponding user average from each co-rated pair [23].

Other similarities: Another similarity measure is conditional probability-based similarity [24].

PREDICTION AND RECOMMENDATION COMPUTATION

Predicting and recommendation is ultimate result of the algorithm. In basic neighborhood -based collaborating algorithm, from nearest neighbor the aggregate of the similarity with the active user is calculated from which prediction is obtained [25].

Weighted sum of others' ratings: The weighted average of all the ratings on particular item I for the active user a will give the prediction by following formula:

$$P_{a,i} = \overline{r}_a + \frac{\sum_{u \in U} \left(r_{u,i} - \overline{r}_u\right) \cdot w_{a,u}}{\sum_{u \in U} |w_{a,u}|}$$

Where *r a* and *ru* are the average ratings for the user *a* and
user *u* on all other rated items, and *wa*, *u* is the weight between the user *a* and user *u*. The summations are over all the users $u \in U$ who have rated the item *i*

Simple weighted average: For item-based prediction, we can use the *simple weighted average* to predict the rating, *Pu*,*i*, for user *u* on item *i*

$$P_{u,i} = \frac{\sum_{n \in N} r_{u,n} w_{i,n}}{\sum_{n \in N} |w_{i,n}|}$$

Where
the summations are over all other rated items $n \in N$for user *u*, *wi*, *n* is the weight between items *i* and *n*, *ru*, *n* is theratng for user *u* on item *n*.

Top-N recommendations: Top-N recommendation is to recommend a set of N top-ranked items that will be of interest to a certain user. Top-N recommendation techniques analyze the user-item matrix to discover relations between different users or items and use them to compute the recommendations.

Userbased top-*N* recommendation

The algorithm result in N recommendations

a) Use Pearson or vector similarity identify k neighbor to active user.
b) Each user is treated as a vector in the *m*-dimensional item space
c) Similarities between the active user and other users are computed between the vectors.
d) After the *k* most similar users have been discovered, their corresponding rows in the user-item matrix *R* are aggregated to identify a set of items, *C*, purchased by the group together with their frequency.
e) With the set *C*, user-based *CF* techniques then recommend the top-*N* most frequent items in *C* that the active user has not purchased.

User-based top-*N* recommendation algorithms have limitations related to scalability and real-time performance [24].

Item based top-*N* recommendation

This algorithm overcomes the limitation of user based algorithm.

1) Compute the k most similar items for each item by similarities.
2) Form a candidate set C (take the union of k and remove already purchased items in set U.
3) Find similarity between C and U.
4) Sort the result set C in decreasing order of similarity.
5) Top-N list is recommended.

The limitation is it produces suboptimal result due to joint distribution of different individual item in the set.

This is overcome by item-based top-*N* recommendation algorithms that use all combinations of items up to a particular size when determining the item sets to be recommended to a user [12].

Extensions to memory-based algorithms

Default voting

The similarity value mainly depends on similar users rating. If the numbers of similar users are less in number then decision is based on very few ratings. The result is not reliable. It is solved by

(a). Assuming some *default voting* values for the missing ratings can improve the *CF* prediction performance [17].

(b). Reducing the weight of users that have fewer than 50 items in common [26].

(c). Uses the average of the clique [27].

(d). Use neutral or negative preference for the unobserved ratings [17].

Inverse user frequency

Universally liked items are not as useful in capturing similarity as less common items. The inverse frequency can be defined as $f_j = \log(n/n_j)$, where *nj* is the

VOL. 10, NO. 6, APRIL 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

number of users who have rated item j and n is the total number of users[28].

Case amplification

Case amplification refers to atransform applied to the weights used in the basic collaborative filtering prediction. The transform emphasizes high weights and punishes low weights. *Case amplification* reduces noise in the data. It tends to favour high weights as small values raised to a power become negligible [29].

Imputation-boosted *CF* algorithms

When the user-item matrix is very sparse, prediction is not accurate. I*mputation-boosted collaborative filtering (IBCF)*, which first uses an imputation technique to fill in the missing data and then find similarity. Mean imputation, linear regression imputation, and predictive mean matching imputation, Bayesian multiple imputation, and machine learning classifiers (including na¨ıve Bayes, SVM, neural network, decision tree, lazy Bayesian rules) as imputers for *IBCF* are effective [30] [31] [32].

Weighted majority prediction

This makes its prediction using the rows with observed datain the same column, weighted by the believed similarity between the rows, with binary rating values [33].

b) Model-based collaborative filtering techniques

Designing of data mining or machine learning algorithm which will study on complex patterns based on training data and make predictions.

Model-based *CF* algorithms are

i. Bayesian models
ii. Clustering models
iii. Dependency networks solve the shortcomings of memory-based *CF* algorithm [34]

A. Bayesian belief net CF algorithms

A Bayesian belief net (*BN*) is a directed, acyclic graph (*DAG*) with a triplet _N,A,Θ, where each node $n \in N$ represents a random variable, each directed arc $a \in A$ between nodes is a probabilistic association between variables, and Θ is a conditional probability table quantifying how much a node depends on its parents .

Bayesian belief nets (*BNs*) are often used for classification tasks [35].

Simple Bayesian CF algorithm

The simple Bayesian *CF* algorithm uses a na¨ıve Bayes (*NB*) strategy to make predictions for *CF* tasks. Assuming the features are independent given the class, the probability of a certain class given all of the features can be computed, and then the class with the highest probability will be classified as the predicted class (the subscript o in the following equation indicates observed values[36].

$$\text{class} = \underset{j \in \text{classSet}}{\arg\max}\, p\left(\text{class}_j\right)\prod_{o} P\left(X_o = x_o \mid \text{class}_j\right)$$

The *Laplace Estimator* is used to smooth the probability calculation and avoid a conditional probability of 0.

$$P(X_i = x_i \mid Y = y) = \frac{\#(X_i = x_i, Y = y) + 1}{\#(Y = y) + |X_i|}$$

where |*Xi*| is the size of the class set *{Xi}* [37].

NB-ELR and TAN-ELR CF algorithms

Because of the limitations of the simple Bayesian algorithm for *CF* tasks, advanced *BNs CF* algorithms, with their ability to deal with incomplete data, can be used instead. Extended logistic regression (*ELR*) is a gradient-ascent algorithm, which is a discriminative parameter-learning algorithm that maximizes *log conditional ikelihood.*

TAN-ELR and *NB-ELR* (tree augmented na¨ıve Bayes andna¨ıve Bayes optimized by *ELR*, resp.) have been proven to have high classification accuracy for both complete and incomplete data [38].

Other Bayesian CF algorithms

Bayesian belief nets with decision trees at each node:

This model has a decision tree at each node of the *BNs*, where a node corresponds to each item in the domain and the states of each node correspond to the possible ratings for each item.

Baseline Bayesian model uses a Bayesian belief net with no arcs (baseline model) for collaborative filtering and recommends items on their overall popularity. However, the performance is suboptimal [39].

B. Clustering CF algorithms

A cluster is a collection of data objects that are similar to one another within the same cluster and are dissimilar to the objects in other clusters. The measurement of the similarity between objects is determined using metrics such as *Minkowski* distance and *Pearson correlation*. For two data objects, $X = (x1, x2, \ldots, xn)$ and $Y = (y1, y2, \ldots, yn)$, the popular *Minkowski* distance is defined as

$$d(X, Y) = \sqrt[q]{\sum_{i=1}^{n} |x_i - y_i|^q}$$

VOL. 10, NO. 6, APRIL 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

where n is the dimension number of the object and xi, yi are the values of the ith dimension of object X and Y respectively and q is a positive integer. When $q = 1$, d is *Manhattan* distance; when $q = 2$, d is *Euclidian* distance [40].

Clustering methods can be classified into three categories:

a) partitioning methods
b) density-based methods
c) hierarchical methods

Partitioning method

A commonly-used partitioning method is k-means, which has two main advantages, relative efficiency and easy implementation.

Density-based clustering methods

Typically search for dense clusters of objects separated by sparse regions that represent noise DBSCAN and OPTICS are well-known density-based clustering methods.

Hierarchical clustering methods

Example is BIRCH creates a hierarchical decomposition of the set of data objects using some criterion. Clustering models have better scalability than typical collaborative filtering methods because they make predictions within much smaller clusters rather than the entire customer base [41]-[45].

Regression-based CF algorithms

A regression method uses an approximation of the ratings to make predictions based on a regression model. Let X = (X1, X2, . . . , Xn) be a random variable representing a user's preferences on different items. The linear regression model can be expressed as

$$Y = \Lambda X + N$$

where Λ is a n × k matrix. N = (N1, . . . ,Nn) is a random variable representing noise in user choices, Y is an n × m matrix with Yi j is the rating of user i on item j andX is ak×m matrix with each column as an estimate of the value of the random variable X.

MDP-based CF algorithms

An MDP is a model for sequential stochastic decision problems, which is often used in applications where an agent is influencing its surrounding environment through actions. An MDP can be defined as a four-tuple: _S,A, R, Pr, where S is a set of states, A is a set of actions, R is a real-valued reward function for each state/action pair, and Pr is the transition probability between every pair of states given each action.

Latent semantic CF models

A Latent semantic CF use latent class variable that discover user communities and prototypical interest profiles. Conceptionally, it decomposes user preferences using overlapping user communities. The main advantages of this technique over standard memory-based methods areits higher accuracy and scalability [46].

Other latent models are aspect model which models individual ratings as a convex combination of rating factors.
A multinomial model that assumes there is only one type of user.

A multinomial mixture model assumes that there are multiple types of users underlying all profiles, and that the rating variables are independent with each other and with the user's identity given the user's type.

A user rating profile (URP) model combines the intuitive appeal of the multinomial mixture model and aspect model.

Other model-based CF techniques

Association rule based CF use traditional association rule mining algorithm to find rules for developing top-N recommender systems. They find the top-N items by simply choosing all the rules that meet the thresholds for support and confidence values, sorting items according to the confidence of the rules so that items predicted by the rules that have a higher confidence value are ranked higher, and finally selecting the first N highest ranked items as the recommended set [47].

C. Hybrid collaborative filtering techniques

Hybrid recommender systems combine two or more recommendation techniques to gain better performance with fewer of the drawbacks of any individual one. Hybrid CF systems combine CF with other recommendation techniques (typically with content-based systems) to make predictions or recommendations. Table-3 shows some of the combination methods that havebeen employed. While the space remains to be fully explored, research has provided some insight into the question of which hybrid to employ in particular situations.

VOL. 10, NO. 6, APRIL 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

Table-3. Hybridization methods.

Hybrid method	Description	Existing work	Combination of recomm. tech.
Weighted	The scores (or votes) of several recommendation techniques are combined together to produce a single recommendation	1.P-Tango[48] 2. (Pazzani 1999) [49] 3. Towle and Quinn 2000 [50]	1.CF/CN 2. CF/DM 3 CF/KB
Switching	The system switches between recommendation techniques depending on the current situation.	1. (R. Burke 2002) [51] 2. (Tran and Cohen, 2000) [52]	1. CF/CN 2. CF/KB
Mixed	Recommendations from several different recommenders are presented at the same time	1 PTV, Prof Builder [53]	CF/CN
Feature combin -ation	Features from different recommendation data sources are thrown together into a single recommendation algorithm.	1. Basu, C., Hirsh, H. and Cohen W.: 1998, [54] 2. Condliff, M. K., Lewis, D.D. Madigan, D. and Posse, C.1999 [55]	1. CF/CN 2. CN/DM
Cascade	One recommender refines the recommendations given by another	1. Fab [56] 2. EntreeC [57]	1. CF/CN 2. KB/CF
Feature augment -tation	Output from one technique is used as an input feature to another	1. Libra [58] 2. Group Lens (1999) [59]	1. CF/CN 2. KB/CF
Meta-level	The model learned by one recommender is used as input to Another	1. Fab, (Condliff, et al. 1999), Labo Ur [56]	1. CN/CF

The hybridization strategy must be a function of the characteristics of the recommenders being combined. More research is needed to establish the tradeoffs between these hybridization options

Evaluation metrics for recommendation system

The quality of the recommendation system is measured through by evaluating the system by metrics [60]. The choosing of metric depends on the approach applied in the system. The metrics are classified into

- Predictive accuracy metrics
  - Mean Absolute Error(MAE)and Normalized MAE
- Classification accuracy metrics
  - Precision
  - Recall
  - F1-measure
  - ROC sensitivity curve
- Rank accuracy metrics
  - Pearson's product-moment correlation
  - Kendall's Tau
  - Mean Average Precision
  - Half-life utility
- Normalized distance-based performance metric

We only discussed the commonly used Predictive accuracy and classification accuracy metrics.

Predictive accuracy metrics

Mean Absolute Error (MAE) and Normalized MAE:

If the recommendation is on prediction based on user's rating on item MAE is used. MAE is average of difference between the predicted rating and actual user rating.

$$Mean\ Absolute\ Error = \frac{\Sigma_{(u,i)\in T} \mid \hat{r}_{ui} - r_{ui} \mid}{N}$$

Where

r_{ui} = actual rating of user u for item i,

$\hat{r}_{ui}$ = predicted rating

T = number of test data(user-item pair)

N = total ratings of overall users

When the ratings are in different numerical scale NMAE is used which normalizes MAE for error correction.

$$normalized\ MAE = \frac{MAE}{r_{\max} - r_{\min}}$$

Where

$r_{\max} - r_{\min}$ is difference between maximum and minimum rating values

Classifying metrics

These metrics classify, produced recommendations into groups as indicated in Table-4.

VOL. 10, NO. 6, APRIL 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

Table-4. Classification metrics.

	Recommended	Not recommended	Total
Used	True positive (TP)	False negative (FN)	Total used
Not used	FalsePositive (FP)	True Negative (TN)	Total not used
Total	Total recommended	Total not recommended	Total T

Once categories are defined, metrics will be calculated according to the following formulae:
Precision:

$$Precision = \frac{TP}{TP+FP}$$

$$Recall\ (True\ Positive\ Rate) = \frac{TP}{TP+FN}$$

F-measure is introduced as a measure of the harmonic mean of precision and recall

$$F\text{-}Measure = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

ROC sensitivity curve

ROC emphasizes the items that are not preferred but are recommended. An *ROC (Receiver Operating Characteristic)* curve is a two-dimensional depiction of classifier performance, on which *TPR* (true positive rate) is plottedon the *Y*-axis and *FPR* (false positive rate) is plotted onthe *X*-axis. *ROC sensitivity* is given by the *Area under the ROC Curve* (*AUC*).

$$AUC = \frac{S_0 - n_0(n_0+1)/2}{n_0 n_1}$$

Where n_0 and $n_1$are number of negative and positive examples and s_0 is summation of the rank of the positive example.

IMPROVING THE RECOMMENDATION PERFORMANCE BASED ON QOS PARAMETER

QoS is a set of performance and domain-dependent attributes that has a substantial impact on WS requesters' expectations. Thus, it can be used for distinguishing between many functionally equivalent WSs that are available nowadays. [61].

A web service is formally described in a standardized language (WSDL). The service description may include the names and types of input and output parameters, preconditions and effects, as well as Quality of Service (QoS) attributes, such as price, execution time, availability, and reputation. As web services and service providers proliferate, therewill be a large number of candidate, and likely competing, services for fulfilling a desired task. Hence, effective service discovery mechanisms are required for identifying and retrieving the most appropriate services. [62].

QoS based Web Service Selection plays an essential role because consumers want to use the services that meets their requirements. Currently web service selection is based on reputation [63].

Recommendation is calculated based on the ratings provided by the user for each web service. As the ratings alone cannot describe the quality of web service, other QoS parameters should also be considered. The seven QoS parameters considered in this paper are execution time, response time, throughput, scalability, reputation, accessibility and availability [64]

Among different QoS properties of W services, some properties are user independent and have identical values for different users (e.g., price, popularity, availability, etc.). The values of the user independent

QoS properties are usually offered by service providers or by third-party registries (e.g., UDDI). On the other hand, some QoS properties are user dependent and have different values for different users (e.g., response time,invocation failure rate, etc.). Obtaining values of the user dependent QoS properties is a challenging task, since realworld Web service evaluation in the client side is usually required for measuring performance of the user dependent QoS properties of Web services.

Client-side Web service evaluation requires real-world Web service invocations and encounters the following drawbacks: most service users are not experts on Web service evaluation. However, without sufficient client-side evaluation, accurate values of the user-dependent QoS properties cannot be obtained. Optimal Web service selection and recommendation are thus difficult to achieve.

To attack this critical challenge, we propose a collaborative filtering based approach for making personalized QoS value prediction for the service users.

Generally, service QoS information is derived in three ways: delivered by services providers, evaluated based on user feedback and predicted based on monitoring information [65].

In collaborative filtering methods, predicting QoS for a user is done by referring to information from similar users. The user environment and input have more influence on personalized QoS prediction.

QOS parameters

The following are the nonfunctional parameters of services provided by the service providers.

Accessibility

The degree of response to service request.

Accuracy

The probability of giving accurate result to the request.

VOL. 10, NO. 6, APRIL 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

Availability

The ratio between the time the user can access the service to the time taken for requesting the service.

Latency

The subtraction of response time from request time of the web service invocation gives latency value.

Reliability

Ratio between the time to request web service and all of the time the request in specific time.

Response time

The time measured from the time since the users send request to respond.

Scalability

Refers to the ability to consistently serve the request when there is increase or decrease in volume of requests.

Throughput

Maximum number of services can process for a unit time.

Related work

At present the research for QOS prediction in recommendation system is limited due to lack of QOS data set in real world. Some related work which to justify the enhancement of recommendation techniques based on QOS consideration that solve the challenges of CF is given below:

A. (Data sparsity, scalabilityvgray sheep problem)

Szu-Yin Lin, -"A trustworthy QoS-based collaborative filtering approach for web service discovery "is probabilistic approach that predict QOS values from past use r experience. It is based on Bayesian inference model and matrix formulation method [66].

B. (Noisy data)

Chengying Mao, Jifu Chen, - QoS Prediction for Web Services Based on Similarity-Aware Slope One Collaborative Filteringattempts to predict missing QOS value and recommend service by combining Pearson similarity and Slope One method . In the paper, we adopt the Pearson similarity between two services as the weight of their deviation. Meanwhile, some strategies like weight adjustment and SPC based smoothing are also utilized for reducing prediction error. In order to evaluate the validity of our algorithm (i.e., similarity-aware Slope One algorithm, SASO) [67].

C. (data smoothing, data sparsity, scalability)

Jian Wu, Liang Chen, Yipeng Feng, Zibin Zheng, *r*Predicting Quality of Service for Selection by Neighborhood-Based Collaborative Filtering proposes neighbourhood based filtering approach to predict missing QOS value, adusted cosine vector similarity to remove different scale of value and similarity fusion approach to solve data sparsity [68].

D Shao *et al.* propose a user-based personalized QoS value prediction for Web services. [69].

CONCLUSIONS

As the number of web service increases, it is difficult for the user to decide the apt service. Also, as most of the users may have limited knowledge in all the domains and about web service description, they need a recommendation or suggestions to choose a service suitable for their request.

Traditional searching method follows keyword searching which mere a word is matching. It results in both the relevant and irrelevant service list. Again the list will have many similar services for a single purpose itself. This is because more than one service provider can publish service for same function. Here, the user needs to select an optimal service among similar service. Similar services can be differentiated based on non functional components called QOS properties.

Already, many existing systems are there which use either functional parameter or non-functional parameter to answer a request. Collaborative technique is one of the widely used techniques that recommend service based on past users experiences towards a service already they have used. This past users is those who will have same interest as active users. Their experiences can be collected as ratings or preferences given as grades e.t.c. After finding out the similar users, the algorithm will predict suitable service by comparing the past users rating and new active user's request.

As non-functional parameters are very important (proved from many existing research) for web service selection, considering QOS properties in similarity computation of CF algorithm will have a positive impact on prediction performance. When QOS is considered, the filtering of suitable web services from much similar web service collection can be finely tuned with tradeoff between QOS parameters. This recommendation will increase user satisfaction. This paper will give an insight on Collaborating filtering technique and proof for the above said proposal.

REFERENCES

[1] Zibin Zheng, Student Member, IEEE, HaoMa, MichaelR. Lyu, Fellow, IEEE, andIrwining, Senior Member, IEEE," QoS- Aware Web service Recommendation by Collaborative Filtering", 2011.

[2] RESNICK, PANDVARIAN, H.R. 1997. Recommender systems. Commun. ACM 40, 3, 56-58.

[3] M. Deshpande and G. Karypis, "Item-Based Top-N Recommendation," ACM Trans. Information System. vol. 22, no. 1, pp. 143-177, 20.

[4] M. J. Pazzani, “A framework for collaborative, content-based and demographic filtering”, Artificial Intelligence Review. vol.13, no. 5-6, pp. 393-408, 1999.

[5] B. Krulwich, “Lifestyle finder: intelligent user profiling using large-scale demographic data,” Artificial Intelligence Magazine. vol. 18, no. 2, pp. 37-45, 1997.

[6] R. Burke, “Hybrid recommender systems: survey and experiments,” User Modelling and User-Adapted Interaction, vol. 12,no. 4, pp. 331-370, 2002.

[7] J.L. Herlocker, J.A. Konstan, A. Borchers, and J. Riedl “An Algorithmic Framework for Performing Collaborative Filtering,” Proc. 22nd Int’l ACM SIGIR Conf. Research and Development in Information Retrieval SIGIR’99), pp. 230-237, 1999.

[8] K. Goldberg, T. Roeder, D. Gupta, and C. Perkins, “Eigentaste: a constant time collaborative filtering algorithm,” Information Retrieval, vol. 4, no. 2, pp.133-151, 2001.

[9] Savoy, J., Picard, J.: Retrieval effectiveness on the web. Information Processing and Management 37(4) (2001) 543-569.

[10] B. N. Miller, J. A. Konstan, and J. Riedl “PocketLens: toward a personal recommender system,” ACM Transactionson Information Systems. vol. 22, no. 3, pp. 437-476, 2004.

[11] J. Breese, D. Heckerman, and C. Kadie, “Empiricalnalysis of predictive algorithms for collaborative filtering,” in Proceedingsof the 14th Conference on Uncertainty in Artificial Intelligence (UAI ’98), 1998.

[12] L. Si and R. Jin, “Flexible mixture model for Collaborative filtering,” in Proceedings of the 20th International Conference on Machine Learning (ICML ’03), vol. 2, pp. 704-711, Washington, DC, USA, August 2003.

[13] A. Ansari, S. Essegaier, and R. Kohli, “Internet recommendation systems,” Journal of Marketing Research, vol. 37, no. 3,pp. 363-375, 2000.

[14] Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, “Grouplens: an open architecture for collaborative filtering of netnews,” in Proceedings of the ACM Conference on Computer Supported Cooperative Work, pp. 175-186, New York, NY, USA, 1994.

[15] Sun, H., Z. Zheng, J. Chen and M.R. Lyu, 2011. NRCF: A novel collaborative filtering method for service recommendation. Proceeding of the IEEE International Conference on Web Services (ICWS’11), pp: 702-703.

[16] Z., Ma, H.: WSRec: A Collaborative Filtering Based Web Service Recommender System. Proc IEEE Intl conference on Web Services, 437-444 2009).

[17] J.S. Breese, D. Heckerman, and C. Kadie, “Empirical Analysis of Predictive Algorithms for Collaborative Filtering,” Proc. 14th Ann. Conf. Uncertainty in Artificial Intelligence (UAI ’98), pp. 43-52, 1998.

[18] R. Jin, J.Y. Chai, and L. Si, “An Automatic Weighting Scheme for Collaborative Filtering,” Proc. 27th Int’l ACM SIGIR Conf. Research and Development in formation Retrieval (SIGIR’04), pp. 337-344, 2004.

[19] G. Xue, C. Lin, Q. Yang, W. Xi, H. Zeng, Y. Yu, and Z. Chen, “Scalable Collaborative Filtering Using Cluster- Based Smoothing,” Proc. 28th Int’l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR ’05), pp. 114-121, 2005.

[20] G. Linden, B. Smith, and J. York, “Amazon.com recommendations: Item-to-Item Collaborative Filtering,” IEEE Internet Computing. vol. 7, no. 1, pp. 76-80, Jan./Feb. 2003.

[21] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, “Item-Based Collaborative Filtering Recommendation Algorithms,” 2001.

[22] M. R. McLaughlin and J. L. Herlocker, “A Collaborative filtering algorithm and evaluation metric that accurately model the user experience,” in Proceedings of 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR ’04), pp. 329-336, Sheffield, UK, 2004.

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

[24] G. Karypis, Evaluation of item-based top-N Recommendation algorithms,” in Proceedings of the International Conference on Information and KnowledgeManagement (CIKM ’01), pp. 247-254, Atlanta, Ga, USA, November 2001.

[25] Ranieri Baraglia C. Lucchese S. Orlando, M. Serrano, F. Silvestri “A privacy preserving Recommender system”, SAC’06ACM 2006.

[26] Vander Meer, D., K. Dutta, A. Datta, K. Ramamritham, S. B. Navanthe. 2000. Enabling scalable online personalization on the web. Proc. of

the 2nd ACM Conf. on Electronic Commerce, Minneapolis, MN, 185-196.

[27] S. H. S. Chee, J. Han, and K. Wang, "RecTree: anefficientcollaborative filtering method," in Proceedings of the 3rd International Conference on Data Warehousing and Knowledge Discovery. pp. 141-151, 2001.

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

[29] D. Lemire, "Scale and translation invariant collaborative filtering systems," Information Retrieval. vol. 8, no. 1, pp. 129-150, 2005.

[30] X. Su, T. M. Khoshgoftaar, X. Zhu, and RGreiner," Imputation-boosted collaborative filtering using machine learning classifiers," in Proceedings of the 23rd Annual ACM Symposium on Applied Computing (SAC '08), p. 949-950.

[31] Cear´a Fortaleza, Brazil, R. J. A. Little, "Missing-data adjustments in large surveys," Journal of Business and Economic Statistics. vol. 6, no. 3, pp. 287-296, 1988.

[32] D. B. Rubin, Multiple Imputations for Nonresponse in Surveys, John Wiley and Sons, New York, NY, USA 1987.

[33] S. A. Goldman and M. K. Warmuth, "Learning Binary relations using weighted majority voting," Machine Learning. vol. 20, no. 3, pp. 245-271, 1995 March 2008.

[34] C. Basu, H. Hirsh, and W. Cohen, "Recommendation As classification: using social and content-based informationin recommendation," in Proceedings of the 15th National Conference on Artificial Intelligence (AAAI '98), pp. 714-720, Madison, Wis, USA, July 1998

[35] J. Pearl, Probabilistic Reasoning in Intelligent J. Pearl, Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, Morgan Kaufmann, San Francisco, Calif, USA, 1988 Systems: Networks of Plausible Inference, Morgan Kaufmann, San Francisco, Calif, USA, 1988

[36] Miyahara and M. J. Pazzani, "Improvement of Collaborative filtering with the simple Bayesian classifier," Information Processing Society of Japan. vol. 43, no. 11, 2002.

[37] B. Shen, X. Su, R. Greiner, P. Musilek, and C. Cheng, "Discriminative parameter learning of general Bayesian network classifiers," in Proceedings of the 15th IEEE International Conference on Tools with Artificial Intelligence,pp. 296-305,Sacramento, Calif, USA, November 2003.

[38] D. Geiger, and M. Goldszmidt, Bayesian network classifiers," Machine Learning. vol. 29, no. 2-3, pp.131-163, 1997.

[39] D. Heckerman, D. M. Chickering, C. Meek, R. Rounthwaite,and C. Kadie, "Dependency networks for inference, collaborative filtering, and data visualization," Journal of Machine Learning. vol. 1, no. 1, pp. 49-75, 2001.

[40] Paolo Buono, Maria Francesca Costabile, Stefano Guida, Antonio Piccinno, Giuseppe Tesoro, "Integrating User Data and Collaborative Filtering in a zeb Recommendation System" in proceedings of 18th International conference on user modelling. vol. 2266, pp: 315-321, 2002.

[41] X. Su, M. Kubat, M. A. Tapia, and C. Hu, "Query Size estimation using clustering techniques," in Proceedings of the 17th International Conference on Tools with Artificial Intelligence (ICTAI '05), pp. 185-189, Hong Kong, November 2005.

[42] B. MacQueen, "Some methods for classification and analysis of multivariate observations," inProceedings of the 5th Symposium on Math, Statistics, and Probability, pp. 281–297, Berkeley, Calif, USA, 1967.

[43] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in Proceedings of the International Conference on Knowledge Discovery and Data Mining (KDD '96), 1996.

[44] M. Ankerst, M. M. Breunig, H.-P.Kriegel, and J. Sander, "OPTICS: ordering points to identify the clustering structure," in Proceedings of ACM SIGMOD Conference, pp. 49-60, June 1999.

[45] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: anefficient data clustering method for very large databases," in Proceedings of the ACM SIGMOD International Conference on Management of Data, vol. 25, pp. 103-114, Montreal, Canada, June 1996.

[46] T. Hofmann, "Unsupervised learning yprobabilistic Latent semantic analysis," Machine Learning. vol. 42, no. 1-2, pp. 177-196, 2001.

[47] C. W. K. Leung, S. C. F. Chan, and F. L. Chung, "A collaborative filtering framework based onfuzzy association rules and multi-level similarity," Knowledge and Information Systems. vol. 10, no. 3, pp. 357-381, 2006.

[48] M. Claypool, A. Gokhale, T. Miranda, et al., Combining content-based and collaborativein an online newspaper," in Proceedings of the SIGIR Workshop on Recommender Systems: Algorithms and Evaluation, Berkeley, Calif, USA, 1999.

[49] Pazzani, M. J. 1999, 'A Framework for Collaborative, Content-Based and Demographic Filtering'. Artificial Intelligence Review. 13(5/6), 393-408.

[50] Towle, B. and Quinn, C.: 2000, 'Knowledge Based Recommender Systems Using Explicit User Models'. In Knowledge-Based Electronic Markets, Papers from the AAAI Workshop, AAAI Technical Report.

[51] R. Burke, "Hybrid recommender systems: survey and experiments," User Modelling and User-Adapted Interaction, vol. 12, 2002.

[52] Tran, T. and Cohen, R.: 2000, 'Hybrid Recommender Systems for Electronic Commerce'. In Knowledge-Based Electronic Markets, Papers from the AAAI Workshop, AAAI Technical Report WS-00-04. pp. 78-83. Menlo Park.

[53] B. Smyth and P. Cotter, "A personalized TV listings service for the digital TV age," in Proceedings of the 19th International Conference on Knowledge-Based Systems and Applied Artificial Intelligence (ES '00), vol. 13, pp. 53-59, Cambridge, UK, December 2000.

[54] Basu, C., Hirsh, H. and Cohen W. 1998. Recommendation as Classification: Using Social and Content-Based Information in Recommendation'. In: Proceedings of the 15th National Conference on Artificial Intelligence, Madison, WI. pp. 714-720.

[55] Condliff, M. K., Lewis, D. D., Madigan, D. and Posse, C. 1999. 'Bayesian Mixed- Effects Models for Recommender System'. SIGIR '99 Workshop on Recommender Systems: Algorithms and Evaluation. Berkeley, CA.

[56] M. Balabanovi´c and Y. Shoham, "Content-based collaborative recommendation," Communications of the ACM, vol. 40, 1997.

[57] Kolodner, J. 1993, 'Case-Based Reasoning'. San Mateo, CA: Morgan Kaufmann.

[58] Mooney, R. J. and Roy, L. 1999, 'Content-Based Book Recommending Using Learning for Text Categorization'. SIGIR '99 Workshop on Recommender Systems: Algoritms and Evaluation. Berkeley, CA

[59] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P. and Riedl, J. 1994, 'GroupLens: An Open Architecture for Collaborative Filtering of Netnews'. In: Proceedings of the Conference on Computer Supported Cooperative Work.

[60] Jonathan L. Herlocker Oregon State University and Joseph A Konstan, Loren G. Teveen, and John T. Riedl University of Minnesota, "Evaluating Collaborative Filtering Recommender Systems,"ACM Transactions on Information Systems. Vol. 22, No. 1, J anuary 2004.

[61] Requirements for QoS-Based Web Service Description and Discovery Kyriakos Kritikos and Dimitris Plexousakis, Member, IEEE.

[62] Ranking and Clustering Web Services Using Multicriteria Dominance Relationships, Dimitrios Skoutas, Dimitris Sacharidis, Alkis Simitsis, and Timos Sellis, Fellow, IEEE.

[63] European Journal of Scientific Research, QOS based Efficient Web Service Selection ISSN 1450-216X Vol. 66 No. 3 (2011), pp. 428-440© Euro Journals Publishing, Inc. 2011.

[64] Majitha, S., Shaikhali, A., Rana, O. and Walker, D., 2004, Reputation based semantic service Discovery", In Proc. Of the 13th IEEE Intl Workshops on Enabling Technologies Infrastructures for Collaborative Enterprises (WETICE), Modena, Italy.

[65] V. Deora, J. Shao, W. Gray, and N. Fiddian, "A Quality of Service Management Framework Based on User Expectations," Proc. First Int'l Conf. Service-Oriented Computing (ICSOC '03), pp. 104-114, 2003.

[66] Szu-Yin Lin, -A trustworthy QoS-based collaborative filtering approach for web service discovery‖, Journal of Systems and Software. Vol. 93, July 2014, pp. 217-228.

[67] Chengying Mao, Jifu Chen, - QoS Prediction for Web Services Based on Similarity-Aware Slope One Collaborative Filtering‖, Journal of Informatica. 37 (2013), pp. 139-148.

[68] Wu, Liang Chen, Yipeng Feng, Zibin Zheng, Predicting Quality of Service for Selection by Neighborhood-Based Collaborative Filtering IEEE 2012.

[69] L. Shao, J. Zhang, Y. Wei, J. Zhao, B. Xie, and H. ei "Personalized QoS Prediction for Web Services via Collaborative Filtering," Proc. Fifth Int'l Conf. Web Services (ICWS '07).