iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 Issue: VI Month of publication: June 2025

DOI: https://doi.org/10.22214/ijraset.2025.72630

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue VI June 2025- Available at www.ijraset.com

Prediction System Algorithm Using QoS Values

Pallavi R. Kathar[1], Kalyani Gosavi[2], Sachin Ragde[3], Priyanka Bonde[4], Jayshree Jadhav[5]
Assistant Professor, Dept. of CSE, ICEEM, Aurangabad

Keywords: Quality of Services, Web Services, Prediction, Recommendation, Internet.

I. INTRODUCTION

1) Web Services: The use of the web services technology on the internet has increased widely as it has improved the efficiency and throughput for developers in developing applications. When a web service is created we share WSDL document of that Web Service to the consumers. The consumers require a source to search for these Web services and all the Web services after their creation the developer must publish it to a registry called UDDI, it is like the yellow pages of WS. Consumer can query the UDDI for the required WS. The SOAP protocol is used for accessing the object of the Web service
2) Recommender Systems: Recommender systems are applied to various applications now days. E-commerce sites like amazon and eBay, social networking sites like Facebook, Twitter and YouTube are some examples of the variety of applications in which recommender systems are being used. As there is huge amount of data to be searched, it is difficult to get a limited and accurate set of results when a user searches for a particular information, hence the recommender system will make use of the "likes" and "dislikes" of various users and generates recommendations based on his/her interests and the advantage of a recommender system is to reduce the user's time for searching the required information by narrowing down the choices that the recommender algorithm predicts a user might be interested in.Recommender systems employ Information Filtering technique that focuses on providing the recommendations of the items to the users that are likely to be of the user's interest.
3) Background of Recommender Systems: The recommender systems endeavor at helping users in service selection. Some QoSproperties are user independent, having matching values for different users while other QoS properties are user-dependent. To avoid the expensive and time-consuming web service invocations, collaborative quality-of-service (QoS) prediction approach for web services by taking advantages of the past web service usage experiences of service users. We first apply the concept of user-collaboration for the web service QoS information sharing. Then, based on the collected QoS data, a neighborhood-integrated approach is designed for personalized web service QoS value prediction.
4) Collaborative QoS Prediction Model: A collaborative QoS prediction framework is shown in Figure 1, where the service users are encouraged to share their individually observed past web service QoS information. In this collaborative framework, a service user will obtain the QoS prediction service from the centralized server only if he/she contributes some QoS values. Meanwhile, more web service QoS values are contributed by a service user; more user features can then be mined from those contributed data. In this way, higher QoS value prediction accuracy can therefore be achieved. It is the essence of this collaborative framework.

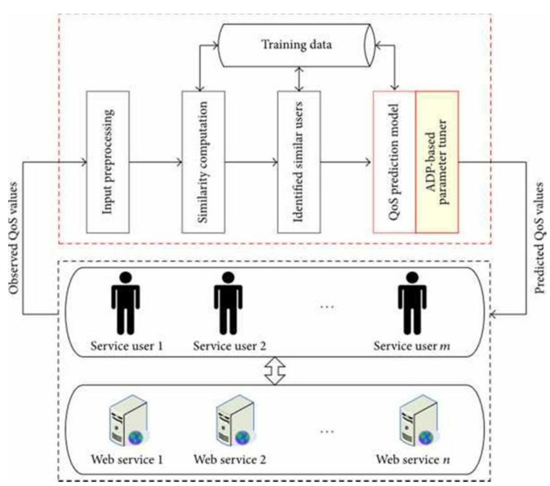


Fig 1: Proposed QoS framework

II. METHODOLOGY

The motivation for this topic comes from the idea that people often get the best recommendations from someone with similar tastes to themselves. Collaborative filtering algorithms often require (1) Users' active participation, (2) An easy way to represent users' interests to the system, and (3) Algorithms those are able to match people with similar interests.

Objective: Our purpose is related to define a prediction algorithm to realize these characteristics, allowing the unknown QoS values to be predicted accurately.

Scope: Scope of the project is to test more real-world web services and more QoS properties of web services will be measured. The procedure of the anticipated QoS values and the permutation of different QoS properties will be used. Intensification the basic Knowledge Bases to cover more measures related to privacy. Area Explicit Knowledge Bases to comprise more sub-domains and their associations. Discover application of other data attributes that are accessible by users such as: user response about quality of historical recommendations, more broad historic data, and user online activities.

- Method user studies and understands how careful users find the recommendations.
- Instrument the recommender system for dissimilar platforms (e.g. mobile) and make it existing to other researchers and consultants.

III. LITERATURE SURVEY

Collaborative Filtering helps in removing tasks like data sparsity, scalability, synonymy, gray sheep, shilling attacks, privacy protection, etc. Three main categories of CF techniques: memory-based, model based, and hybrid CF algorithms are defined in the paper. The authors have surveyed various collaborative filtering systems to explore the related challenges that exist in the respective area and have proposed some solutions. They have even provided a categorization and classification of the systems and identified some possible areas of enhancements [1].

It proposes an evaluation framework which combines the key aspects of web service for recommendation systems. Based on this evaluation framework, they gave a comparison and analysis of the current web service systems, and identify some challenges which require further research and development [2]. A fine-grained model was proposed to express Web service providers' privacy policies andusers' privacy preferences based on several privacy dimensions – sensitivity, purpose, retention period, visibility – while other approaches to privacy aware [3].

Runtime service adaptation has been recognized as a key solution to achieve thisgoal. To make timely and accurate adaptation decisions, effective QoS prediction is desired to obtain the QoS values of component services. However, their research has focused mostly on QoS prediction [6] of the working services that are being used by a cloud application, but little on QoS prediction of candidate services that are also important for making adaptation decisions [4]. New recommender system technologies are needed that can quickly produce high, quality recommendations [7] even for very large-scale problems.To address these issues.Explored item-based collaborative techniques. Item-based techniques first analyze the user-item matrix to identify relationships between different items, and then use these relationships to indirectly compute recommendations for users [5]. Collaborative filtering (CF) is one of the most widely-used usercentric recommendation techniques in practice. For a specific user, CF recommends items according to the preference of similar users. User similarity plays an important role in CF, including both memory-based and model-based approaches [8]. A new convolution neural network based multimodal [10] disease risk prediction (CNN-MDRP) algorithm using structured and unstructured data from hospital was discussed. The work focused on both data types in the area of medical big data analytics [9].

Decision Tree and Support Vector Machine was the machine learning classification algorithm used by the majority of researchers in their heath care predictive research and are the best algorithm in case of accuracy. Machine Learning and Artificial Intelligence have virtually endless applications in the healthcare and medical domain. Machine learning is helping to streamline administrative processes, diagnosis diseases, prognosis diseases, treatment schedule and personalize medical treatments in healthcare to map and treat diseases [11]. Genetic algorithm has been used for attribute reduction and RBF Network for classification. Classification accuracy has been enhanced by reducing the number of attributes [12]. UIQPCA as a unique way of quality prediction with the help of covering algorithm is proposed. UIQPCA clusters users and Web services based on their opinions on quality and historical quality data, respectively. Given a target Web service for a target user, the similar users and Web services are found out on the basis of clustering results [13]. Proposed a blockchain-based QoS prediction framework that can effectively resisted unreliable users to obtain more accurate prediction results [14].

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue VI June 2025- Available at www.ijraset.com

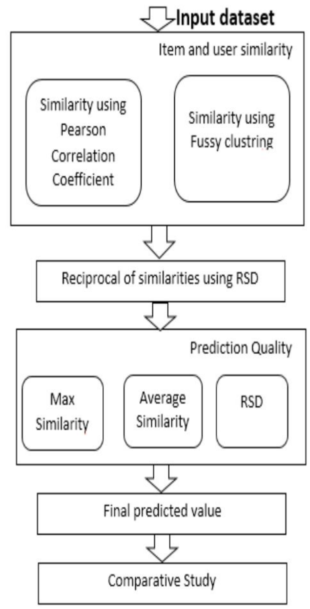


Fig 2: Recommendation System

IV. SYSTEM ARCHITECTURE

Software architecture serves as the blueprint of Highly Accurate Prediction Algorithm with Fuzzy Clustering, describing the task allotments to facilitate the design and execution. The architecture is the key transporter of system capabilities such as modifiability, presentation, and safety; no one can be recognized by lacking a combining architectural vision. HAPA with CF is consisting of following modules. The description of individual module is explained with detail below. As shown in architecture diagram in fig. 2, here we take a wsrec_dataset as an input. In existing system item and user similarity is evaluated using Pearson correlation coefficient and in our proposed we are using with that fussy clustering which gives more accurate similarity between users and items. After that we calculate reciprocal of all this similarity using reciprocal of standard deviation, and then we are going to calculate quality of predictions by calculating max Similarity, average similarity and RSD values. After calculating it for all user and item final predicted value is calculated.

1) Pearson Correlation Coefficient (PCC): It is used to measure user similarity [2] in recommendation systems. It measures the similarity between two service users based on the QoS values of Web services. PCC[3] similarity sim (a,b) of two service users ranges from -1 to 1.Two service users have similar Web service usage experiences if the PCC value is positive and a negative PCC value indicates that their experiences are opposite. The value is null when two service users have no commonly invoked web service.
2) Prediction Quality by RSD: The following measures are use to calculate prediction quality.
3) Max Similarity (MS): For pvuserand pvitem, MS represents the maximum similarity between users in KU and services in KI, denoted by ms (pv_{user}) and ms (pv_{item}), respectively.
4) Average Similarity (AS): For pvuser and pvitem, AS represents the average similarity between users in KU and services in KI, denoted by (pv_{user}) and (pv_{item}), respectively
5) Reciprocal of Standard Deviation (RSD): For pvuser and pvitem, RSD represents the reciprocal of the standard deviation of the similarities between users in KU and services in KI, denoted by rsd(pv_{user}) and rsd(pv_{item}), respectively.
6) Collaborative filtering (CF) is a technique used by some recommender systems. It has two senses, a narrow one and a more general one, In general, collaborative filtering is the process of filtering for information or patterns using techniques involving collaboration among multiple agents, viewpoints, data sources, etc. Applications of collaborative filtering typically involve very large data sets.
7) Neighborhood-based Collaborative Filtering: This type of CF approaches use the observed QoS data to compute the similarity values between users or services, and further leverage them for QoS prediction.

8) Model-based Collaborative Filtering: Model-based CF approaches provide a predefined model to fit the observed QoS data, and then the trained model can be used to predict the unknown QoS values. Matrix factorization (e.g., PMF) is one of the most popular model-based CF approaches, which was first introduced to address the QoS prediction problem in.
9) User-based collaborative filtering: The neighborhood-based algorithm calculates the similarity between two users or items produces a prediction for the user by taking the weighted average of all the ratings. Similarity computation between items or users is an important part of this approach. Multiple measures, such as Pearson correlation and vector cosine based similarity are used for this.
10) Item-based collaborative filtering: Item-item collaborative filtering, or item-based, or item-to-item, is a form of collaborative filtering based on the similarity between items calculated using people's ratings of those items.

V. PERFORMANCE ANALYSIS

It is difficult to mine the peculiarities of Web service QoS values, and the prediction accuracy of previous algorithms cannot be trusted without believable and sufficient real-world Web service QoS data.

- We now discuss the computational complexity of predicting one unknown QoS value using our prediction algorithm.
- These corollaries are the theoretical foundation of our proposed algorithm HAPA. HAPA includes user-based and item-based prediction according to Corollaries and respectively.
- All similarities are generally calculated in advance since it is very time-consuming with a large dataset. Therefore similarities calculations are not included in the complexity of our prediction algorithm.
- To validate the accuracy of our algorithm, we predict only the known values, so that we can evaluate the error between the predicted values and real values.

A. Theoretical Foundation of HAPA

Pearson Correlation Coefficient (PCC) was introduced in a number of recommender systems for similarity computation, since it can be easily implemented and can achieve high accuracy. In user-based collaborative filtering for Web services, PCC is employed to define the similarity between two service users a and$u$ based on the Web service items they commonly employed using the following equation:

A popular similarity measure in user-based CF: Pearson correlation

a, b : users

〖 r〗_(a, p): rating of user a for item p

P : set of items, rated both by $a$and b

–Possible similarity values between -1 and 1

$$\boldsymbol{sim(a,b)} = \frac{\sum_{p\in P}(r_{a,p}-\bar{r}_a)(r_{b,p}-\bar{r}_b)}{\sqrt{\sum_{p\in P}(r_{a,p}-\bar{r}_a)^2}\sqrt{\sum_{p\in P}(r_{b,p}-\bar{r}_b)^2}}$$

B. Evaluation

In this section, we conduct a set of experiments based on a real-world Web service QoS dataset to evaluate our AMF approach from various aspects, including accuracy comparison, impact of parameters, efficiency analysis, and scalability analysis. All the experiments were conducted on a machine with a 3.2 GHz Intel CPU and 4 GB RAM, running Win7. A. Data Description In our experiments, we focus primarily on two QoS attributes: response time (RT) and through put (TP). Response time stands for the time duration between user sending out a request and receiving a response, while throughput denotes the data transmission rate (e.g., kbps) of a user invoking a service

C. Accuracy Comparison

In order to evaluate the prediction accuracy, we compare our AMF approach with the following approaches that have been introduced for QoS prediction. It is worth noting that although these approaches are included for comparison purpose, they cannot be directly used for runtime service adaptation in practice, due to the aforementioned limitations.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue VI June 2025- Available at www.ijraset.com

Statistics	Values
#Users	142
#Services	4,500
#Time slices	64
#Time interval	15*min*
RT range	$0 \sim 20s$
RT average	$1.33s$
TP range	$0 \sim 7000kbps$
TP average	$11.35kbps$

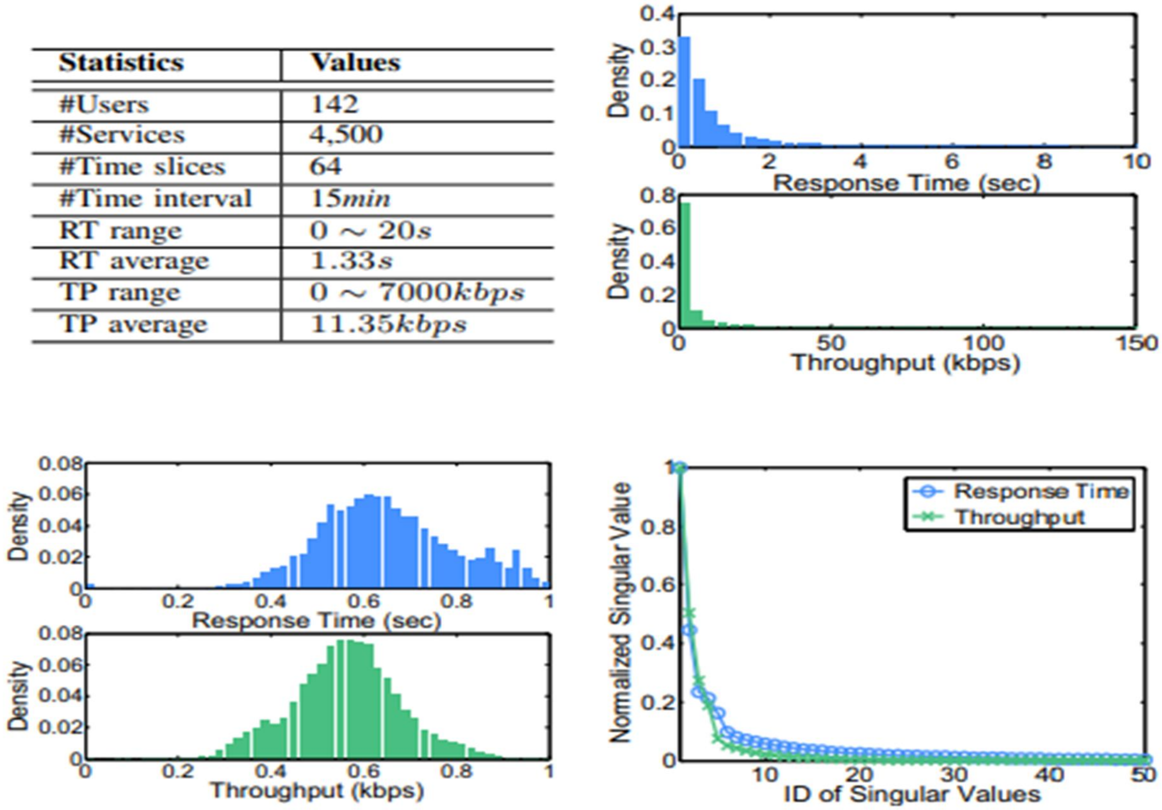


Fig 3: Analysis

QoS	Approach	Density = 10%			Density = 20%			Density = 30%			Density = 40%			Density = 50%		
		MAE	MRE	NPRE	MAE	MRE	NPRE	MAE	MRE	NPRE	MAE	MRE	NPRE	MAE	MRE	NPRE
RT	UPCC	1.224	0.769	7.842	1.076	0.611	5.893	1.006	0.557	4.943	0.967	0.529	4.547	0.940	0.511	4.332
	IPCC	1.273	0.776	6.650	1.218	0.779	6.354	1.144	0.736	5.768	1.070	0.680	5.192	1.020	0.647	4.826
	UIPCC	1.215	0.764	7.489	1.076	0.610	5.889	1.005	0.558	4.977	0.962	0.530	4.571	0.932	0.524	4.383
	PMF	1.104	0.593	3.017	1.030	0.596	3.414	0.982	0.581	3.390	**0.948**	0.564	3.294	0.928	0.546	3.198
	AMF	**1.076**	**0.478**	**1.765**	**1.007**	**0.386**	**1.080**	**0.974**	**0.356**	**0.968**	0.950	**0.344**	**0.929**	**0.921**	**0.334**	**0.914**
	Improve.(%)	2.5%	19.4%	41.5%	2.2%	35.2%	68.4%	0.8%	38.7%	71.5%	-0.2%	39.0%	71.8%	0.8%	38.8%	71.4%
TP	UPCC	9.019	2.179	26.176	8.237	1.900	25.091	7.691	1.697	23.830	7.382	1.624	24.134	7.131	1.537	23.850
	IPCC	8.744	0.833	12.816	8.434	0.832	12.750	7.960	0.789	11.830	7.452	0.729	10.438	7.127	0.699	9.748
	UIPCC	8.596	1.534	17.982	8.048	1.842	21.250	7.501	1.694	20.499	7.074	1.511	18.804	6.764	1.390	17.667
	PMF	6.894	0.567	2.899	6.474	0.525	2.929	6.235	0.488	2.847	5.960	0.459	2.764	5.668	0.436	2.657
	AMF	**6.303**	**0.513**	**2.148**	**5.920**	**0.414**	**1.424**	**5.742**	**0.385**	**1.170**	**5.694**	**0.368**	**1.042**	**5.621**	**0.356**	**0.983**
	Improve.(%)	8.6%	9.5%	25.9%	8.6%	21.1%	51.4%	7.9%	21.1%	58.9%	4.5%	19.8%	62.3%	0.8%	18.6%	63.0%

Fig 4: Analysis Data

- UPCC: This is a user-based collaborative filtering approach that employs the similarity between users to predict the QoS values.
- IPCC: This is an item-based collaborative filtering approach that employs the similarity between services to predict the QoS values.
- UIPCC: This is a hybrid approach, by combing both UPCC and IPCC approaches to make full use of the similarity between users and the similarity between services for QoS prediction.
- PMF: This is a widely-used implementation of matrix factorization model.

As we mentioned before, the available QoS data matrix is sparse in practice, because each user typically only uses a small number of candidate services out of all of them. To simulate the sparse situation, we randomly remove entries from the data matrix at each time slice so that each user only keeps a few available historical values. In this way, we vary the matrix density from 10% to 50% at a step increase of 10%. Matrix density = 10%, for example, indicates that each user invokes 10% of the services, and each service is invoked by 10% of the users. For AMF approach, the preserved data entries are randomized as a QoS data stream for training. Then the removed entries are used as the testing data to evaluate the prediction accuracy. In the sequel, for simplicity, we set $\lambda_u = \lambda_s = \lambda$ for AMF. Specifically, in this experiment, we set $d = 10$, $\lambda = 0.001$, $\beta = 0.3$, $\eta = 0.8$, $\alpha = -0.007$ for RT, and $\alpha = -0.05$ for TP. Note that the parameters of the other approaches are also optimized accordingly to achieve their optimal accuracy.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue VI June 2025- Available at www.ijraset.com

At each time slice, each approach is performed 20 times for each matrix density. Then the results on average prediction accuracy over the first time slice are reported. Table provides the comparison results over three metrics, but we focus more on relative error metrics, i.e., MRE and NPRE. As we can observe, our AMF approach significantly outperforms the other approaches over MRE and NPRE, while still achieving comparable (or best) results on MAE. Concretely, for response time (RT) data, AMF achieves 19.4%~39.0% improvement on MRE and 41.5%~71.8% improvement on NPRE at different matrix densities. Similarly, for throughput (TP) data, AMF has 9.5%~21.1% MRE improvement and 25.9%~63.0% NPRE improvement. Note that all improvements are computed as the percentage of how much AMF outperforms the other most competitive approach. We also find that although UIPCC achieves higher accuracy over MAE than UPCC and IPCC and PMF achieves better performance compared with the first three approaches, all these approaches have large errors over MRE and NPRE. Thus, only minimizing the absolute error may lead to large relative error, which is not suitable for QoS prediction problem. To further analyze the benefit of our AMF approach, we plot the distributions of prediction errors in Fig. We can observe that AMF achieves denser distribution around the center 0, while UIPCC and PMF have flat error distributions, which indicates the better performance of AMF.

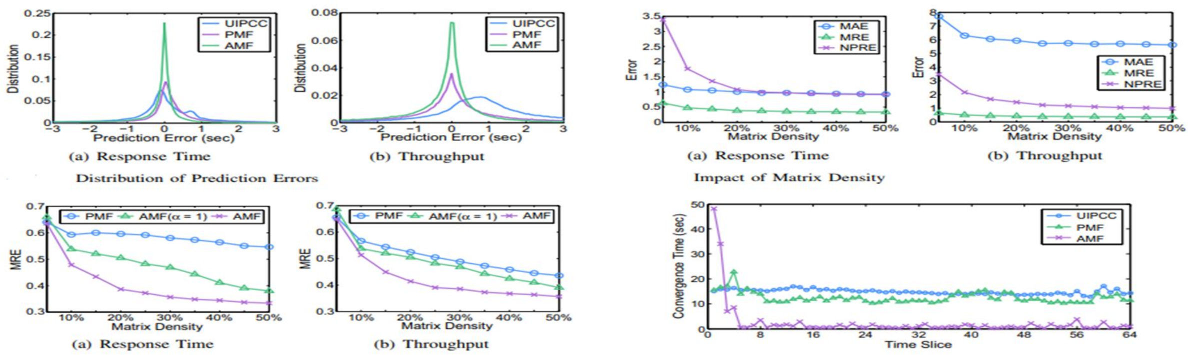


Fig 5: Response Time and Throughput

1) Impact of Data Transformation

The effect of data transformation on data distributions has been illustrated in Fig. To further evaluate the impact of data transformation on prediction accuracy, we compare the prediction accuracy among three approaches, including PMF, AMF($\alpha = 1$), and AMF. In AMF($\alpha = 1$), $\alpha = 1$ indicates that the data transformation is relaxed to a linear normalization procedure, since the effect of the function boxcox(x) is masked. In contrast, AMF is our approach with a well tuned α (e.g., $\alpha = -0.007$ for response time and $\alpha = -0.05$ for throughput). In this experiment, we also vary the matrix density and then compute the corresponding MRE values. The results are illustrated in Fig. We can observe that the data transformation method has a significant impact on improving prediction accuracy over MRE. Especially, the PMF approach aggressively minimizes the absolute error, resulting in large MRE as shown in Fig. Besides, AMF improves a lot in MRE compared with AMF($\alpha = 1$) due to the effect of BoxCox transformation on QoS data distributions.

2) Impact of Matrix Density

To present a comprehensive evaluation on the impact of the matrix density, we vary the matrix density from 5% to 50% at a step increase of 5%. Besides, we set the other parameters. Fig. illustrates the evaluation results. We can observe that as the matrix density increases, better prediction accuracy can be achieved. In particular, the error decreases dramatically with the increase of matrix density, when the QoS matrix is excessively sparse (e.g., matrix density = 5%). It shows that the model can fall into the over fitting problem due to data sparsity. With more data collected, the over fitting problem can be alleviated, thus further improving QoS prediction accuracy.

3) Efficiency Analysis

To evaluate the efficiency of our approach, we compare the convergence time of AMF with two other approaches, UIPCC and PMF. Despite the long convergence time for the first time slice, our AMF approach becomes quite fast in the following time slices because AMF incrementally updates the model by online learning using sequentially observed data samples. In contrast, UIPCC and PMF are more computationally expensive, since they need to re-train the whole model at each time slice, which incurs high computational overhead compared to our online algorithm. Thus, they are more appropriate for one-time training as used in traditional recommender system.

4) Scalability Analysis

To analyze the scalability of our AMF model on new users and services, we evaluate the prediction accuracy on these new users and services, as well as the robustness of the prediction results. For this purpose, we simulate the new users and services from our dataset. Specifically, we randomly select 80% of users and services from our dataset at time slice 1 as existing users and services, and then train the AMF model using their data. After the model converges, we add the remaining 20% of users andservices into the model at time t = 400s. Ideally, by using our algorithm 1, AMF can scale well to the new users and services, and perform robustly by keeping updating the feature vectors of existing users and services with small weights, and the feature vectors of new users and services with large weights. Fig. presents the results, where we can see that the MRE for the new users and services rapidly decreases after their joining. However, the MRE for existing users and services still keep stable, which indicates the robustness of our model under the churning of users and services. Therefore, our AMF approach shows good scalability on new users and services.

5) Concluding Remarks

This is the first work to study the problem of QoS prediction on candidate services for service adaptation. Towards this end, we propose adaptive matrix factorization (AMF) to address the online

QoS prediction problem that is fundamental for runtime service adaptation. AMF formulates the QoS prediction problem as a collaborative filtering problem inspired from recommender systems, and extends the traditional matrix factorization model with techniques of data transformation, online learning, and adaptive weights, in order to address the unique challenges faced in runtime service adaptation. Comprehensive experiments based on a real-world QoS dataset have been conducted to evaluate our AMF approach, which demonstrates its good performance in achieving accuracy, efficiency, and scalability.

VI. CONCLUSION

Collaborative filtering (CF) is one of the most successful recommender techniques. Broadly, there are memory-based CF techniques such as the neighborhood-based CF algorithm; model-based CF techniques such as Bayesian belief nets CF algorithms, clustering CF algorithms, and MDP-based CF algorithms; and hybrid CF techniques such as the content boosted CF algorithm and Personality diagnosis. As a representative memory-based CF technique, neighborhood-based CF computes similarity between users or items, and then use the weighted sum of ratings or simple weighted average to make predictions based on the similarity values. Pearson correlation and vector cosine similarity are commonly used similarity calculations, which are usually conducted between co-rated items by a certain user or both users that have co-rated a certain item. To make top N recommendations, neighborhood-based methods can be used according to the similarity values. Memory-based CF algorithms are easy to implement and have good performances for dense datasets. Shortcomings of memory-based CF algorithms include their dependence on user ratings, decreased performance when data are sparse, new users and items problems, and limited scalability for large datasets, and so forth. Memory-based CF on imputed rating data and on dimensionality-reduced rating data will produce more accurate predictions than on the original sparse rating data. Model-based CF techniques need to train algorithmic models, such as Bayesian belief nets, clustering techniques, or MDP-based ones to make predictions for CF tasks. Advanced Bayesian belief nets CF algorithms with the ability to deal with missing data are found to have better performance than simple Bayesian CF models and Pearson correlation-based algorithms. Clustering CF algorithms make recommendations within small clusters rather than the whole dataset, and achieve better scalability. An MDP-based CF algorithm incorporates the users' action of taking the recommendation or not into the model, and the optimal solution to the MDP is to maximize the function of its reward stream. The MDP-based CF algorithm brings profits to the customized system deploying it. There are downsides of model-based CF techniques, for example, they may not be practical when the data are extremely sparse, the solutions using dimensionality reduction or transformation of multiclass data into binary ones may decrease their recommendation performance, the model-building expense may be high, and there is a tradeoff between prediction performance and scalability for many algorithms.

Most hybrid CF techniques combine CF methods with content-based techniques or other recommender systems to alleviate shortcomings of either system and to improve prediction and recommendation performance. Besides improved performance, hybrid CF techniques rely on external content information that is usually not available, and they generally have increased complexity. It is always desirable to design a CF approach that is easy to implement, takes few resources, produces accurate predictions and recommendations, and overcomes all kinds of challenges presented by real-world CF applications, such as data sparsity, scalability, synonymy, privacy protection, and so forth. Although there is no cure-all solution available yet, people are working out solutions for each of the problems.

To alleviate the sparsity problem of CF tasks, missing-data algorithms such as TAN-ELR, imputation techniques such as Bayesian multiple imputation and dimensionality reduction techniques such as SVD and matrix factorization can be used. Clustering CF algorithms and other approaches such as an incremental SVD CF algorithm are found promising in dealing with the scalability problem. Latent semantic indexing (LSI) is helpful to handle the synonymy problem. And sparse factor analysis is found helpful to protect user privacy. Advances in Artificial Intelligence Besides addressing the above challenges, future CF techniques should also be able to make accurate predictions in the presence of shilling attacks and noisy data, and be effectively applied in fast-growing mobile applications as well. There are many evaluation metrics for CF techniques. The most commonly used metric for prediction accuracy include mean absolute error (MAE), recall and precision, and ROC sensitivity. Because artificial data are usually not reliable due to the characteristics of CF tasks, real-world datasets from live experiments are more desirable for CF research.

We have presented a novel approach to assist users and Web service providers in the composition and selection of composite services that are more privacy preserving. With respect to other proposals for privacy-preserving Web service composition, our approach supports the specification of fine-grained privacy policies and preferences based on different privacy dimensions, i.e. purpose, visibility, retention period and sensitivity. In addition, our approach ranks the generated composite Web services with respect to their privacy level, which quantifies the risk of unauthorized disclosure of user information based on sensitivity, visibility and retention period. As future work, we are planning to conduct an extensive evaluation of our Java-based prototype. First, we will evaluate its performance with respect to the number of candidate Web services, the complexity of the privacy policies of the orchestrator and component services, and to the (re)delegation depth.

Then, we will conduct a controlled experiment with master students in computer science to evaluate participants' perceived ease of use, perceived usefulness, and intention to use according to the Technology Acceptance Model (TAM).

REFERENCES

[1] Xiaoyuan Su and Taghi M. Khoshgoftaar (2009) "A Survey of Collaborative Filtering Techniques", Hindawi Publishing Corporation Advances in Artificial Intelligence Volume 2009, Article ID 421425, doi:10.1155/2009/421425

[2] Z. Zheng, H. Ma, M. R. Lyu and I. King (2013) "Collaborative Web Service QoS Prediction via Neighborhood Integrated Matrix Factorization" in *IEEE Transactions on Services Computing*, vol. 6, no. 3, pp. 289-299.

[3] Elisa Costante, Federica Paci and Nicola Zannone 2013) "Privacy-Aware Web Service Composition and Ranking", International Journal of Web Services Research · July 2013

[4] Jieming Zhu, Pinjia He, ZibinZheng and Michael R. Lyu (2014) "Towards Online, Accurate, and Scalable QoS Prediction for Runtime Service Adaptation", IEEE 34th International Conference on Distributed Computing Systems 1063-6927/14

[5] BadrulSarwar, George Karypis, Joseph Konstan, and John Riedl (2001) "Item-Based Collaborative Filtering Recommendation Algorithms", Hong Kong, ACM 1-58113-348-0/01/0005.

[6] XiongLuo, HaoLuo and Xiaohui Chang (2015) "Online Optimization of Collaborative Web Service QoS Prediction Based on Approximate Dynamic Programming", Hindawi Publishing Corporation International Journal of Distributed Sensor Networks Vol. 2015, Article ID 452492, http://dx.doi.org/10.1155/2015/452492

[7] Priyanka Sebastian and B. Janani (2014) "A Survey on Web Service Recommender Systems", International Journal of Advanced Research in Computer, Volume 5, No. 1, ISSN No. 0976-5697

[8] YitongMeng, Weiwen Liu and Benben Liao (2019) "PMD: A New User Distance for Recommender Systems", CSIR, arXiv:1909.04239v2

[9] Min Chen, YixueHao, Kai Hwang, Lu Wang and Lin Wang (2017) "Disease Prediction by Machine Learning Over Big Data from Healthcare Communities", IEEE- Healthcare Big Data, 10.1109/ACCESS.2017.2694446

[10] Amirah Mohamed Shahiria, WahidahHusaina and Nuraini Abdul Rashida (2015) "The Third Information Systems International Conference A Review on Predicting Student's Performance using Data Mining Techniques", Published by Elsevier B.V., Procedia Computer Science 72, 414 – 422 1877-0509

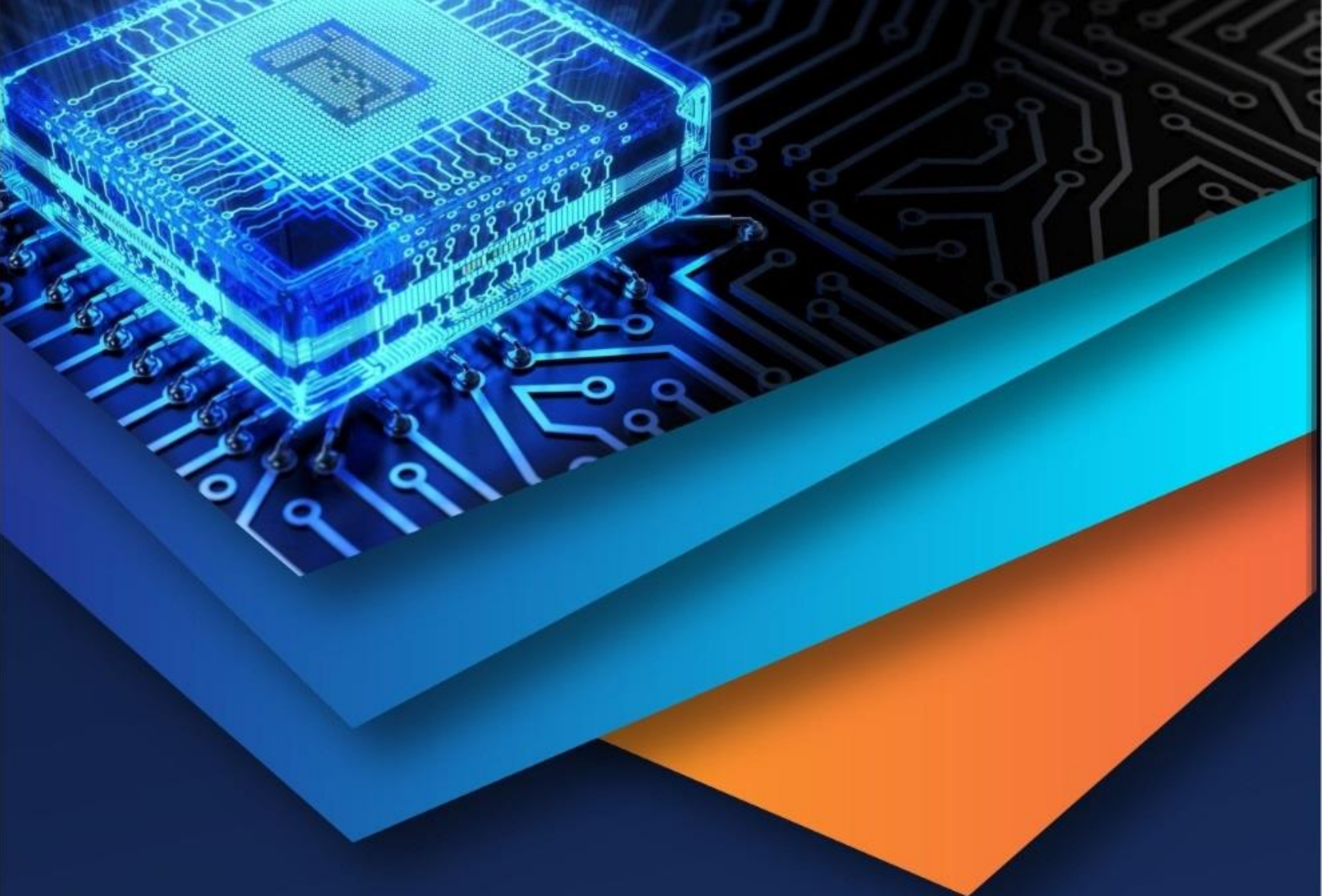
[11] Ms. ManjiriMahadevMastoli, Dr. Urmila R. Pol and Rahul D. Patil (2019) "Machine Learning Classification Algorithms for Predictive Analysis in Healthcare", International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 06 Issue: 12, p-ISSN: 2395-0072 Page 1225

[12] A Durga Devi (2015) "Enhanced Prediction of Heart Disease by Genetic Algorithm and RBF Network", International Journal of Advanced Information in Engineering Technology (IJAIET) Vol.2, No.2, 29-36

[13] Ying Jin, Guangming Cui and Yiwen Zhang (2020) "Quality Prediction of Web Services Based on a Covering Algorithm", Hindawi Complexity Volume 2020, Article ID 8572161, https://doi.org/10.1155/2020/8572161

[14] WeihongCai, Xin Du and JianlongXu (2019) "A Personalized QoS Prediction Method for Web Services via Blockchain-Based Matrix Factorization", *Sensors* 2019, *19*(12), 2749

[15] ChaimaAbid, MarouaneKessentini and Hanzhang Wang (2020) "Early prediction of quality of service using interface-level metrics, code-level metrics, and anti patterns", Information and Software Technology, Vol. 126, 106313



10.22214/IJRASET

INDEX COPERNICUS 45.98

IMPACT FACTOR: 7.129

TOGETHER WE REACH THE GOAL

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY