A hybrid random-walk based web service recommendation enhanced by matrix factorization

Lin Jian
School of Physics and
Electronic Information Engineering,
WenZhou University
Email: neolinjian@gmail.com

Homer Simpson
Twentieth Century Fox
Springfield, USA
Email: homer@thesimpsons.com

***Abstract*—Recently, the QoS(Qaulity of Serivce) of Web Service that includes response-time, throughput and so on that needs more accuracy prediction. For many web service callers, choosing the appropriate service in right time should be more significant events. So the web serivce recommendation is right to be the choice. The collaborative filtering is major approach to predict the QoS of more web service through the observed data. But the sparse density of data need new technology to enhance the accuracy of prediction. And the matrix factorization is aslo the common approach to solve the prediction. In this paper, we propose the new hybrid approach that combined the predictions with random-walk based and matrix factorization. Comprehensive experiments on the QoS data set of real-world web service, that our approach achieve the more accuracy predictions.**

***Index Terms*—random-work, web service recommendation, matrix factorization**

I. Introduction

Overview the past few years, the collaborative filtering and matrix factorization have success in traditional fields of recommendation, such as Goods, Music, Moive and so on. The recommendation of web service was effected by the achievements. However, the scenario of web service is more complex that suffers from sparse data and incomplete related information. There are so many different web services distributing over heterogeneous network which contains several auto-systems. So the recommendation of web service should solve the problems that sparse QoS(Quality of Service) value collected from various with the intrusted infomation about location or network. In a word, more measures should be made to enhanced the limited information to achieve the more accuracy of web service recommendation. Only that, the system of web service can provide the more quality service.

Web Service QoS predicted information enhanced technology is developing fast. For example, time-aware recommendation that makes prediction by history call record, location-aware recommendation that make use of numbers of AS(auto system), IP or GPS(Global Position System). But the measures all achieve improvement in accuracy of prediction in small scale of the sparse data. Athough the information is critical to prediction, the experiments prove the factor that the more appropriate neighhorhood ranking can really boost the accuracy of prediction. So the paper that Random Walk Models can efficiently work in real-world datasets in the past year. With the transition probability matrix which based on the principle of markov random process, the undirected connected users can calculate the similarity in neighborhood selection.

In the field of web service recommendation, the random-walk model is efficient, but the accuracy of prediction needed more improvement. The matrix factorization had ever solved the sparse efficiently in similar scenario. Naturally, we will try combining the random-walk model with the matrix factorization to get the good performance. And the matrix factorization also is the best approach to reduction of dimensions, when we calculate the similarity between user and user, the time complexity will be smaller. With mroe high-efficiency model, the hybrid algorithm improve the accuracy of prediction in final.

In summary, to solve the web service recommendation and to increase the accuracy of QoS prediction, in this paper, the contributions we made as following:

- We explore the extreme rate of data mining in known probability, and study the sparse density dataset and statistic phenomenon in real-world through calculation.
- We propose the hybrid approach to combine the user-based collaborative filtering with matrix factorization.
- We conduct the experiments on real-world datasets, and achieve the best accuracy of QoS prediction.

The rest of this paper is organized as follows. Section II summarizes the related work and our thought about sparse dataset. Section III introduces our approach to combine the CF and MF algorithm. Section IV reports the experiments and analyst the result of approaches. Section V concludes the paper and discusses the future work.

II. Related Work

In this section, we will introduce the initiation of sparse density data, and explore the extreme rate of data mining in ideal environment that the sampling rate given in advance. Then the review of technology of recommendations will be displayed, includes collaborative filtering, matrix factorization, and random-walk model.

A. *Initition of the sparse density data*

In the real-world dataset environment, our recommendation system samples the whole dataset with $d\%$ density. Suppose that the matrix Q have m users, n services, and the $Q \in \mathbf{R}^{m \times n}$. q_{ij} means the QoS of user i called service j.

In our ideal sampling method that we suppose the system samples the data onto normal distribution. So the every user samples the QoS of service with number $n \times d\%$. Athough the sampling method is ideal, the client user can implement the measure in the real environment. For example, with parameters $m = 300, n = 5000, d\% = 5\%$, 300 users get $300 \times 5000 \times 5\%$ about 300×250 QoS data from the dataset, and it can vividly reflected the sparse level from the fig1. And we can approximately calculate the expectation of number on the common invoked service number. Firstly, we suppose the $user_i$ samples the 250 service in whole, the $user_j (i \neq j)$ repeat the sampling process about 299 times. The model will subordinate to the binomial distribution of $X \sim b(299, 0.05)$, and we can get the common invoke service number about $user_i$ by $E = np = 14.95, D = np(1-p) = 14.20$. So the real factor is the low sampling rate is hard to recover the data of whole, and the location-aware information is less affected the result of user-based collaborative filltering prediction.

B. The extreme rate of data mining

The extreme data mining rate maybe propose to solve our confusion that how the accuracy we can achieve by improve the recommended model. Maybe, we can use the statistic approaches to get the expectation about the extreme prediction. Finally, with the expectation we can improve our alogirthm to achieve more accuracy efficiently.

And the expectation of number about common invoked serivce is also the major latent factors in matrix factorization process. And Section IV will displayed the relationship bewteen the best latent facters parameter and the matrix factorization accuracy.

C. User-based Collaborative Filtering

The CF(Collaborative Filtering)-based algorithm have been widely used. The CF mines a user's common invoked services, which is identitied by response-time or throughput, by calculation of similarity (the Euclidean distance) between the $user_i$ and $user_j$.

$$dst(i,j) = \frac{1}{N_{ij}} \sqrt{\sum_{k \in S_{ij}} (q_{ij} - q_{ij})^2} \tag{1}$$

where $S_{i,j}$ means the set includes the commonly invoked serivce by $user_i$ and $user_j$, and the N_{ij} means the number of the common invoked service. By the Equcation (1), we get the distance between two users, there is a defect that the two users have no common invoked serivce, the distance will be zero, we identity the smaller value means the more similarity between two users, so the condition should be excluded in algorithm.

$$sim(i,j) = \frac{1}{1 + \frac{1}{N_{ij}} \sqrt{\sum_{k \in S_{ij}} (q_{ij} - q_{ij})^2}} \tag{2}$$

where the number 1 in the denominator is a way of Laplacian smooth to avoid the denominator being 0. It can conclude that the pair of users with the smaller distance is calcuated the value more near to number 1. In the reversed condition, the value will be number 0.

With the similarity calculated by front step. We can construct the similarity matrix $Sim \in \mathbf{R}^{m \times m}$, the s_{ij} means the similarity between $user_i$ and $user_j$. Then, we ranking the neighbors by cells of matrix, and choose the top K users to predict the QoS value with the Equcation (3).

$$q'_{ik} = \frac{\sum_{j \in TopK_i} Sim_{ij} \times (Q_{jk} - \overline{Q_j})}{\sum_{j \in TopK_i} Sim_{ij}} + \overline{Q_i} \tag{3}$$

where $\overline{Q_j}$ means the average value of $user_j$, the Equcation (3) also considers the different user has different baseline of QoS prediction.

D. Matrix Factorization

The MF(Matrix Factorization) has also been chosen for its accuracy. By factorizing the matrix $Q \in \mathbf{R}^{m \times n}$ into user and service latent matrix $U \in \mathbf{R}^{m \times k}$, $S \in \mathbf{R}^{n \times k}$. The decomposing process like the Equcation (4)(5)(6).

$$\begin{aligned} &\arg\min_{U,S} \sum_{i=1}^{m} \sum_{j=1}^{n} (Q_{ij} - U_i \cdot S_j^T)^2 \\ &+ \lambda_U \cdot \sum_{i=1}^{m} \|U_i\|_F^2 + \lambda_S \cdot \sum_{i=1}^{n} \|S_i\|_F^2 \end{aligned} \tag{4}$$

$$\frac{dloss}{dU_i} = \sum_{j=1}^{n} (Q_{ij} - U_i \cdot S_j^T) \cdot S_j + \lambda_U \cdot \|U_i\| \tag{5}$$

$$\frac{dloss}{dS_j} = \sum_{i=1}^{m} (Q_{ij} - U_i \cdot S_j^T) \cdot U_i + \lambda_S \cdot \|S_j\| \tag{6}$$

where the Equation (4) is used to minimize the loss of equcation, and the $\| \cdot \|_F$ denotes the Frobenius norm to penalize the norms of U and S. The Equation (5) (6), we can use the gradient descent algorithm with several iterations, and find appropriate matrix U and S at last. Finally, the QoS value will be predicted by the inner product of $U_i \cdot S_j$.

However, the matrix factorization is independent process, the latent matrix $U \in \mathbf{R}^{m \times k}$ can be used as dimension reduction matrix of origin matrix $Q \in \mathbf{R}^{m \times n}$, the dimension reduces from n to k. To some degree, the condition that user is with sparse records will be alleviated, and the the data in large scale will be dealt efficiently in short time.

E. Random-Walk model

The random-walk that based on PageRank alike measures (a Markov process) to get more appropriate neighbors ranking with the transition matrix.

In the last two Subsection, we get the matrix $Sim \in \mathbf{R}^{m \times m}$, the Sim_{ij} means the similarity between $user_i$ and $user_j$ by one step. As we all known, if $user_i$ and $user_j$ are undirected with one step, the $user_i$ is similar to $user_k$ with similarity

Sim_{ik}, and the $user_j$ is similar to $user_k$ with similarity Sim_{jk}, then we can achieve the similarity between $user_j$ is similar to $user_k$ by two step.

So we build the graph $G(V_U, Sim)$ and use the Markov chain to model the state transition of random walk. Let $U_0 \in V_U$ and the $Sim_{0,k}$ means the similarity between $user_0$ and the others. There are two choices for $user_0$:

- the $user_0$ with α probability by one step to link to itself.
- the $user_0$ with $1-\alpha$ probability by one step to link to others.

The process goes by following equation.

$$P(X_t = j|X_0 = i) = \begin{cases} \alpha^t \cdot M^{t-1} \cdot Sim_i^T & i \neq j, \\ \alpha^t \cdot M^{t-1} \cdot Sim_i^T + 1 - \alpha & i = j. \end{cases} \tag{7}$$

The t steps random walking is

$$P_t = (1-\alpha)P_0 + \alpha M^T P_{t-1} \tag{8}$$

In summary, we have the transition matrix M. Along with the step t being infinite, the probability will converge to be stable, which is decided by the steday state distribution of the Markov chain. When the probability is stable, then the P_t will equal to P_{t-1}, then the Equation 8 can be futher transformed into the following shape by linear algebra calucation.

$$P^* = (1-\alpha)(I - \alpha M^T)^{-1} P_0 \tag{9}$$

With the Equation 9, we can transform the matrix Sim more with more appropriate similarity value.

III. Hybrid appoach with RW and MF

The matrix Q will be decomposed into U and S which the latent dimension k at first. And the similarity matrix Sim will be calculated by $\frac{1}{k} \cdot U \cdot U^T$. Then the probability matrix M will by calucate by following Equation:

$$M_{i,j} = \frac{Sim_{ij}}{\sum_{k \in Adj_i} Sim_{ik}} \tag{10}$$

And the Equation (9) will get the P^* through the matrix M.

$$\varphi_i = \frac{1}{N(j)} \cdot \sum_j \frac{Sim_{ij}}{P^*_{ij}} \tag{11}$$

The parameter φ_i can easily calucated by Equation.

$$Sim^*_{ij} = \frac{\varphi_i \times P^*_{ij} + \varphi_j \times P^*_{ji}}{2} \tag{12}$$

With the new similarity matrix Sim^* value, the top K nearest neighbors will be selected.

$$\begin{aligned} q^*_{ij} = \lambda \cdot (\frac{\sum_{j \in TopK_i} Sim_{ij} \times (Q_{jk} - \overline{Q_j})}{\sum_{j \in TopK_i} Sim_{ij}} + \overline{Q_i}) \\ +(1-\lambda) \cdot \sum_k U_{ik} \cdot S_{kj} \end{aligned} \tag{13}$$

The final QoS prediction will be calucated by Equation (13). With the parameter λ, the predictions can adjust to different scenarios. The CF algorithm in sparse data will be more low than the real QoS value, and the MF algorithm will be more low high that the real QoS value with the regularizations. The overfitting or underfitting and changed by combine algorithm. The details of algorithm is in Algorithm 1. And the code of algorithm could been found in WebSite [1].

Algorithm 1 the RWEMF

Require: Q , $\overline{U}$,max_iter,min_thr,λ_{mf},λ_{un},
Ensure: Q^*
initialize U_i, S_j with random value
// decompose the matrix Q to U and S
for $t = 0$ to max_iter **do**
  $U_i = U_i - (Q_{ij} - U_i \cdot S_j) - \lambda_{mf} U_i$
  $S_i = S_j - (Q_{ij} - U_i \cdot S_j) - \lambda_{mf} S_i$
  $loss = \sum (Q_{ij} - U_i \cdot S_j)$
  if $loss < min\_thr$ **then**
    break
  end if
end for
// calucate the similarity matrix
for $i = 0$ to m **do**
  for $j = 0$ to m **do**
    $Sim_{ij} = \frac{1}{1 + \frac{1}{N_{ij}} \sum (U_i - U_j)^2}$
  end for
end for
// calucate the transtion matrix
for $i = 0$ to m **do**
  for $j = 0$ to m **do**
    $M_{ij} = \frac{Sim_{ij}}{Sim_i}$
  end for
end for
// fix the similarity matrix with transtion matrix
for $i = 0$ to m **do**
  for $j = 0$ to m **do**
    $\varphi_i = \frac{1}{N(j)} \cdot \sum \frac{Sim_{ij}}{P^*_{ij}}$
    $Sim^*_{ij} = \frac{\varphi_i \times P^*_{ij} + \varphi_j \times P^*_{ji}}{2}$
  end for
end for
// get the prediction by similarity matrix
for $i = 0$ to m **do**
  for $j = 0$ to m **do**
    $cf = \frac{\sum_{j \in TopK_i} Sim_{ij} \times (Q_{jk} - \overline{Q_j})}{\sum_{j \in TopK_i} Sim_{ij}} + \overline{Q_i}$
    $mf = U_i \cdot S_j$
    $Q^*_{ij} = \lambda_{un} \cdot cf + (1 - \lambda_{un}) \cdot mf$
  end for
end for
return Q^*

IV. Experiment and Evaluation

A. Dataset and Description

The dataset is from WS-DREAM [2]. The whole dataset includes two attribute sub-datasets: response time(RT) and

[1] github.com/neoinmatrix/neosci/tree/master/rwemf
[2] github.com/wsdream/wsdream-dataset

throughput(TP). The statistics of dataset are shown in Table I. The dataset reflects the real-world condition that we have few clients to observe the QoS value and there are so many service on the Internet.

TABLE I
STATISTICS OF DATASET

QoS	numU	numS	min	max	mean	std
RT(sec)	339	5825	0.001	19.999	0.9086	1.9727
TP(kbps)	339	5825	0.004	1000	47.5617	110.7971

The information about the location of users and services can get in Table II. The row of "user_as" means there are 339 users in the dataset. And the 339 users are distributing in 136 areas. Every area has at least 1 user and no more than 31 users. And the average of users on one area about 2.4745 with 2.8338 standard deviation. Notice that the postfix "_as" and "_ct" means area is as(auto system) and ct(country) respectively. From the statistic information about data, the fact that users or services distribute in different area are extremely unbalanced. The location dataset provides inefficiency information, that is why the location information is hardly enhanced the accuracy of our experiments.

TABLE II
STATISTICS OF USERINFO AND SERVICEINFO

infoattr	num	size	min	max	mean	std
user_as	339	136	1	31	2.4745	2.8338
user_ct	339	30	1	161	10.9355	28.3673
service_as	5825	992	1	1212	5.8661	40.6092
service_ct	5825	73	1	2395	78.7162	285.9846

B. Evaluation Metric and Parameter

The MAE(Mean Absolute Error) and NMAE(Normalized Mean Absolute Error) may be the common measurable metrics. MAE is defined as

$$MAE = \frac{1}{N} \sum_{i,j} |q_{ij} - \hat{q_{ij}}| \tag{14}$$

The NMAE is computed as the MAE normalized by the mean of all values, which is defined as

$$NMAE = \frac{MAE}{\frac{1}{N} \sum_{i,j} |q_{ij}|} \tag{15}$$

The MAE reflects the absolute error of the predictions. The NMAE reflects the relative error of the predictions. We can compare the ability of predictions from different dataset with NMAE relatively.

C. Result Accuracy and Comparison

There are several classical recommendation algorithms in the experiments as the comparisons. The result of response time and throughput are displayed in Table III and Table IV respectively.

The comparisons including

- UPCC is the user-based collaborative filtering algorithm that calculate the similarity between users with Pearson correlation coefficient. In this case of small number of users, the algorithm is fast with short running time.
- IPCC is the user-based collaborative filtering algorithm that calculate the similarity between users with Pearson correlation coefficient. In this case of large number of services, the algorithm is slow with long running time.
- UIPCC is the hybrid method linearly combines the results of UPCC and IPCC, but the accuracy is more precise than that two. With the running time of total two algorithm, the algorithm is aslo slow.
- PMF is the matrix factorization algorithm with the model of probability. In this case with sparse data, the process is fast to convergence stable state. So the maximum iteration and convergenced threshold are signifant to keep the running time within accepcteble range.
- UL_RWE is user-based random walk model enhanced by the matrix factorization. The dimension-reducted matrix U with k dimensions latent elements, the algorithm is more fast and achieve more accuracy.
- RWEMF is our approach which are more efficient in the experiment. In the base of UL_RWE, the approach successfully combined the matrix factorization prediction. The running time is close to matrix factorization to achieve more accuracy.

TABLE III
THE MAE AND NMAE OF RESPONSE TIME PREDICTION

model	DS	5%	10%	15%	20%
UPCC	MAE	0.6159	0.5371	0.4966	0.4737
	NMAE	0.6794	0.5917	0.5471	0.5219
IPCC	MAE	0.6805	0.6572	0.5670	0.4955
	NMAE	0.7507	0.7240	0.6246	0.5459
UIPCC	MAE	0.6045	0.5336	0.4879	0.4601
	NMAE	0.6668	0.5879	0.5374	0.5068
PMF	MAE	0.5704	0.4894	0.4584	0.4390
	NMAE	0.6292	0.5391	0.5050	0.4837
UL-RWE	MAE	0.5255	0.4735	0.4462	0.4291
	NMAE	0.5797	0.5216	0.4916	0.4727
XEMF	MAE	0.5518	0.4891	0.4756	0.4877
	NMAE	0.6087	0.5388	0.5239	0.5373
RWEMF	MAE	0.5068	0.4560	0.4344	0.4251
	NMAE	0.5591	0.5023	0.4786	0.4683

TABLE IV
THE MAE AND NMAE OF THROUGHPUT PREDICTION

model	DS	5%	10%	15%	20%
UPCC	MAE	26.8039	22.2826	20.0274	18.689
	NMAE	0.5643	0.4688	0.4212	0.3931
IPCC	MAE	29.5539	29.4531	30.1322	27.5450
	NMAE	0.6222	0.6196	0.6338	0.5794
UIPCC	MAE	26.0401	21.9952	20.0911	18.6256
	NMAE	0.5483	0.4627	0.4226	0.3918
PMF	MAE	22.5499	17.9761	16.5358	15.0594
	NMAE	0.4748	0.3782	0.3478	0.3168
UL-RWE	MAE	19.4043	15.6509	14.3058	13.5797
	NMAE	0.4085	0.3293	0.3009	0.2857
XEMF	MAE	21.0512	17.2567	15.9693	15.5798
	NMAE	0.4432	0.3630	0.3359	0.3277
RWEMF	MAE	18.5121	15.1752	13.9855	13.3388
	NMAE	0.3898	0.3193	0.2942	0.2806

Form the experimental results are shown in Table III IV, we have some observations.

- The matrix factorization algorithm(PMF) achieved more accuracy than the user-based or item-based without enhanced algorithm(UPCC,IPCC,UIPCC).
- The algorithm (HL-RWE) enhanced by random-walk model is achieve more accuracy than the similarity calculated based collaborative filtering algorithm(UPCC,IPCC,UIPCC). So the precision similarity calcualtion and the appropiate and ranking neignhbors seleceted are the efficient approaches to improve the accuracy.
- The RWEMF algorithm is more efficient than other algorithms and achieves the best accuracy. The sparse density of 5% is more appropiate for the algorithm to have accuracy that the dense density of 20%.
- In the different sub-dataset, the algorithms achieve different performance. The response-time dataset with value range (0.001-19.999) and standard deviation 1.9727 is with fluctuation about 9.86%. The throughput dataset with value range (0.004-1000) and standard deviation 110.797 is with fluctuation about 11.08%. The RWEMF achieves $\frac{0.6794-0.5591}{0.6794} = 0.1771$ in rt dataset and $\frac{0.5643-0.3898}{0.5643} = 0.3092$ in tp dataset. So the sparse density and the fluctuation in the dataset is the important elements to the RWEMF algorithm.

D. Analysis and Deduction

The signifcant parameters in RWEMF are top K, latent dimensions, the rate of mf union.

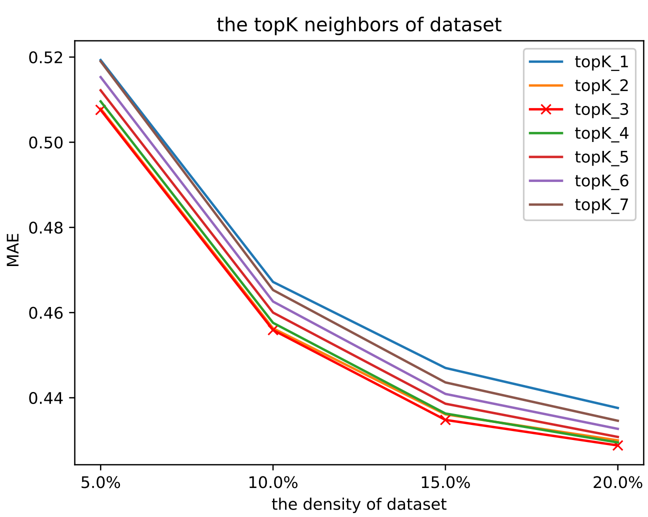

Fig. 1. the MAE of different topk of RWEMF on the response-time dataset

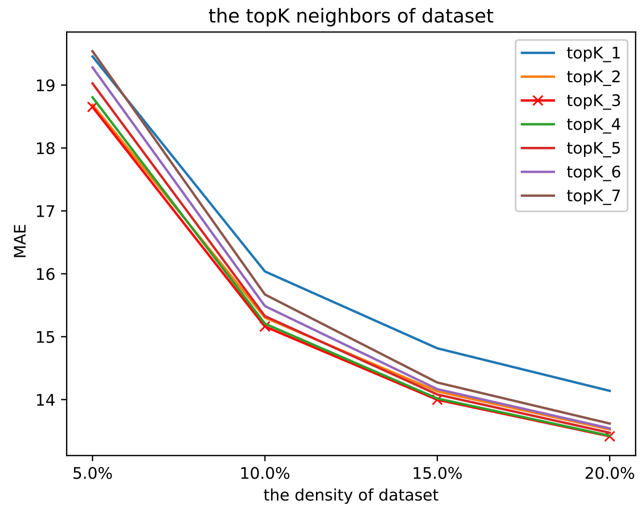

Fig. 2. the MAE of different topk of RWEMF on the throughput dataset

From the Figure 1 2, the nearby neighbors selected obviously effects the mae accuracy. when topk=3, the RWEMF achieves the best accuracy in both response-time and throughput dataset.

From the Figure 3 4, the rumf parameter is the rate united the mf. In the experiments, we choose the 5% and 20% density. And every rate of density has three lines including the rwe line ,mf line and rwemf line. It is clearly to know, the MAE of mf is largest in three, the MAE of rwe is smaller than mf's, with the 0.7 of rumf, the rwemf reaches the best accuracy of MAE. The phenomenon in the throughput dataset is similar. But the response-time dataset with small value is more sensitive to the rate, and it reaches the best accuracy in short range.

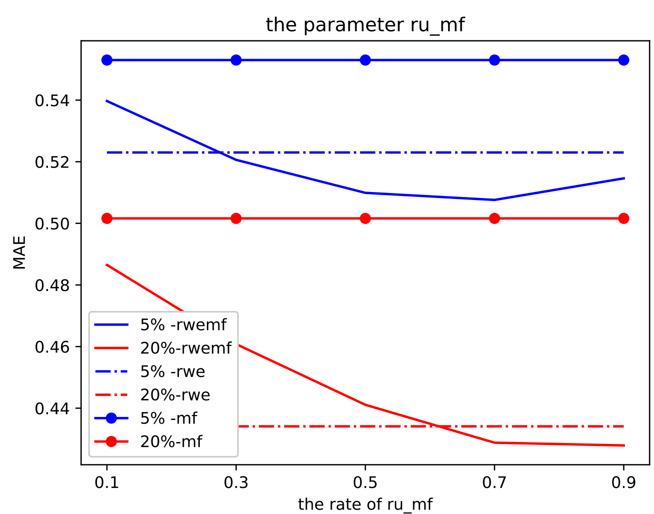

Fig. 3. the MAE of different rate union mf on the response-time dataset

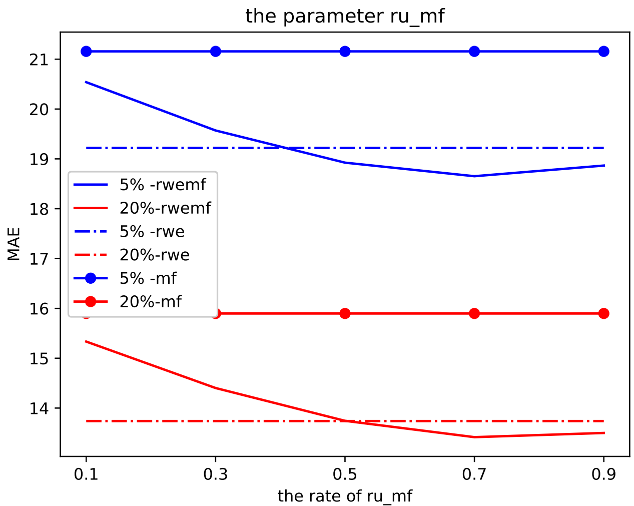

Fig. 4. the MAE of different rate union mf on the throughput dataset

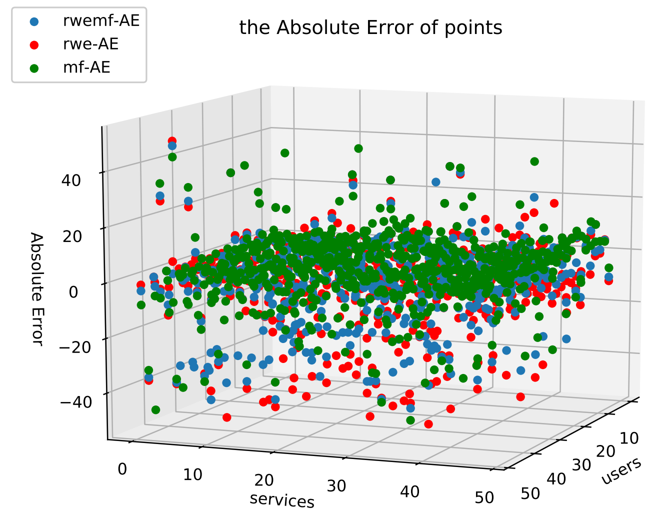

Fig. 6. the Absolute Error on the throughput dataset

Every point in Figure 5 6 means the prediction of RWEMF,RWE,MF three algorithm minus the real QoS value of dataset, and the points in view are sampling randomly that on behalf of the whole predicitons. It is easy to see the AE(Absolute Error) of RWEMF is locating in the middle between RWE's and MF's. Sometimes, the RWE can get the accuracy, but it also processes with the big variance. And the ME can not get the accuracy, but it aslo runs steadily with the small variance. And the predictions of algoithms are sensitive to value of dataset. The absolute error in throughput dataset fluctuated in large range compared to response-time dataset's.

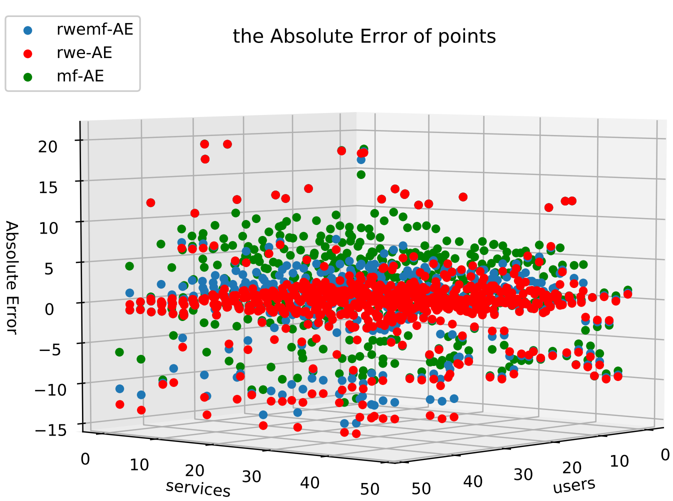

Fig. 5. the Absolute Error on the response-time dataset

V. Conclusion

We propose the RWEMF a hybrid approach to best accuracy in QoS real-world web service dataset. Firstly, We explore the sparse density dataset through statistic calucation. Clearly, the similar calucation and the nearby neighbors selection are significant. And the combination of random-walk user-based collaborative filtering and matrix factorization algorithm is described in the papers. The experiments of RWEMF prove our algorithm is most efficient and the best parameters chosen made RWEMF achieved the best accuracy in this QoS dataset.

In the future, with the best accuracy in this dataset, the RWEMF can be extended by more efficient model. The short running time and exquisite mind can help the algorithm using in real-world web service recommendation easily. The parameters for the hybrid model need more exploration and more study to keep the algorithm more efficient.Athough the adherent users's and service's infomation impove the accuracy finitely, there are more latent information value should be mined in the dataset. The MAE in 5% on response-time dataset is 0.5068 now, athough the value is relative, sometime it could be metrics to measure the ability of algorithm in sparse dataset. Futhre, the MAE could be lower that 0.5000 by the new hybrid model.

Acknowledgment

The authors would like to thank... [1] [2] [3] [4] [5] [6] [7].

References

[1] W. Lo, J. Yin, S. Deng, Y. Li, and Z. Wu, "An extended matrix factorization approach for QoS prediction in service selection," pp. 162–169. [Online]. Available: http://ieeexplore.ieee.org/document/6274140/

[2] D. D. Lee and H. Sebastian Seung, "Algorithms for non-negative matrix factorization," pp. 535–541. [Online]. Available: http://dl.acm.org/citation.cfm?id=3008829

[3] X. Chen, Z. Zheng, Q. Yu, and M. R. Lyu, "Web service recommendation via exploiting location and QoS information," vol. 25, no. 7, pp. 1913–1924.

[4] J. Liu, M. Tang, Z. Zheng, X. . Liu, and S. Lyu, "Location-aware and personalized collaborative filtering for web service recommendation," vol. 9, no. 5, pp. 686–699.
[5] Y. Yin, F. Yu, Y. Xu, L. Yu, and J. Mu, "Network location-aware service recommendation with random walk in cyber-physical systems," vol. 17, no. 9, p. 2059, random-walk. [Online]. Available: http://www.mdpi.com/1424-8220/17/9/2059
[6] Y. Zhang, Z. Zheng, and M. R. Lyu, "Exploring latent features for memory-based QoS prediction in cloud computing," in *2011 IEEE 30th International Symposium on Reliable Distributed Systems*, pp. 1–10, NMF.
[7] Li Kuang, Yingjie Xia, and Yuxin Mao, "Personalized services recommendation based on ..."