QoS Based Ranking for Composite Web Services

F.Ezhil Mary Arasi[1], Aditya Anand [2], Subodh Kumar[3]

[1]SRM University, De[partment of Computer Applications,
Kattankulathur, Chennai, India

[2]SRM University, De[partment of Computer Applications,
Kattankulathur, Chennai, India

[3]SRM University, De[partment of Computer Applications,
Kattankulathur, Chennai, India

Abstract***: Web service security is a challenging area in SOA environment. Quality of Service represents an important issue which is often considered when selecting and composing services. In this paper, we explore that the quality of the page delivery should play an important role in the page ranking process, especially for users with a slow Internet connection or mobile users. We define several important quality attributes and explain how we rank the web page based on these attributes. Simulation experiments demonstrate that proposed framework with QoS for service selection and ranking can satisfy service consumers non-functional information requirements and achieve better web service selection effectiveness.***

.

Keywords: QOS, Composite Web Services, Meta search Engine, Web Search**.**

ISSN: 2278 – 7798

1. Introduction

The quality of a web page can have a few meanings, such as the quality of its content (e.g. authority), or the quality of its delivery. Many of today's search engines use the content quality to rank a web page, whereas few of them use the delivery quality in the ranking process. An example of the delivery quality is the response time. If the network is fast and the hosting web site is fast, response time is not a concern at all. But if the hosting web site is very slow or the Internet user has a very slow connection, then it could be an issue. For instance, if two pages have similar content, page *A* has a smaller response time than page *B*, definitely users would prefer *A* than *B* if the difference is obvious.

In this paper, we propose a Quality of Service (QoS) based ranking algorithm for the web search. We define two types of QoS attributes, one is performance related, and the other is personal preference related. The basic idea is that if two pages are both relevant to a query with the same degree of relevancy, the page with a higher performance will be ranked higher than the other page.

In the experiment, in order to evaluate our proposed algorithm, we implemented it in a meta search engine. We built the meta search engine based on three most popular search engines – Google, Yahoo and MSN, and then combined them with QoS based ranking. The experiment results show the obvious quality improvement in the top 10 results. We also compared several combination methods, and found that the QoS-based re-ranking algorithm achieves the largest increase on the QoS values of the top 10 results.

The rest of the paper is organized as follows. Section 2 reviews the related works. Section 3 defines the search-related quality attributes. Section 4 describes how to combine QoS attributes in the ranking procedure. Section 5 explains the experiment design and the results. We conclude the paper in section 6.

2. Related works

There are two types of meta search algorithms. One is called the rank aggregation algorithm [1] [8] and the other is the score combination algorithm [3] [5].

Borda-fuse [1] and Condorcet-fuse [8] are two commonly used rank aggregation algorithms. Both of them were proposed to solve the problem of voting. In Borda-fuse, points are given to each page based on its rank. It is a very simple procedure, but it has been proven to be both efficient and effective. In Condorcet-fuse, the winner should win (or ties in) every possible pair-wise majority contest. By going through all pages in the result set and assigning them points based on pair-wise comparisons, a ranked order is obtained. For both methods, a weight can be assigned to each search engine based on its average precision.

The most commonly used score combination model is the linear combination model. Fox and Shaw [5] tested five methods and found that CombSUM performs the best. Bartell *et al* [3] added scalars derived from numerical optimization techniques in the linear model. There are also some algorithms proposed [2] to estimate the score based on the rank, such as by a linear mapping.

QoS has been an active research topic in several areas. Our review focuses on the web service area because how they use QoS in web services is similar to how we can use QoS for web search. In [9], Ran described a list of QoS attributes and how they can be used in web service discovery. We define our search-related QoS attributes based on this list.

There are a limited number of papers studying how to use the QoS information in IR systems, and they mainly use it in digital libraries or other distributed information systems. Kapidakis *et al* [7] provided the resource allocation and the distributed searching based on QoS attributes, such as the server/network load and reliability. They focused on how to provision, monitor and manage QoS in such an environment.

3. QoS attributes related to page ranking

There are many QoS attributes that have been proposed in the communication area or web service area, but not all of them are applicable to the web search. Based on our observation on the average user behavior in the web searching process, we identify a list of QoS attributes that we believe are quite relevant to the web search and categorize them into three types – content-related, performance-related, and preference-related [4]. Very few search engines consider the second type and the performance related third type, which indeed is the focus of this paper. We choose several most important and representative quality attributes, including reliability, response time (type 2), file size, the number of embedded multimedia objects, and the data freshness (type 3). Our system framework would be flexible and extensible enough so as to easily include new QoS attributes and adjust the granularity of the attributes. Below we give a definition for each of them and explain how we measure and calculate them.

ISSN: 2278 – 7798

Reliability – it measures how reliable a web site is and whether it could deliver a page without error. It is defined as the percentage of successful delivery among all the requests to this page. In order to measure it, we make a number of requests to the same web page during different time of a day and different days of a week. Each time, if the HTTP response code is 2** [6], it is considered as a successful delivery, or otherwise, it means a failure.

Response time – it measures how soon a user can see the web page after a request is submitted to the web server. It is defined as the time difference between the time when a user sends a request and the time when the user receives the response. Since the network speed is not a constant number and the web server is not in the same state all the time because of the change of the incoming traffic and the internal status of the server itself, the response time may vary for the same page when it is accessed at different times. In order to more accurately measure the response time, we submit the request to the same page several times, and get the average value.

File size – it measures the size of the web page excluding all the embedded objects. It can be obtained from the HTTP header or by downloading the page.

Media richness – it measures the number of embedded multimedia objects within a web page.

Data freshness – it measures how fresh the web page is. A recently updated page is fresher than a page that was created years ago. It is defined as the time difference between the last modified date and the page access time. If the last modified date is not specified for a page, a predefined value will be used. For the freshness, we also send the same request several times to calculate the average value, and in this way, we could more accurately measure a frequently updated or newly updated page.

For mobile users or users with slow Internet connections, the first four QoS attributes are especially important. The first two directly affect the satisfaction level of their Internet experiences, and the latter two determine the cost of their Internet access. For this particular type of users, we assume that they prefer a smaller file size and a smaller number of multimedia objects. The freshness attribute is more for general Internet users.

4. Combining QoS attributes in ranking

After we define the search-related QoS attributes, we need to decide how we can integrate them into the ranking procedure. In this study, we build a meta search engine, and then use QoS attributes to re-rank the search results.

There are three ways to re-rank the original search results. The first one is to simply re-rank the search results from the meta search engine based on all the QoS attributes. The second way is to rank the pages based on QoS attributes and then do the rank aggregation with the original rank from the meta search engine. And the third way is to calculate the QoS values for all the pages and then do the score combination with the meta search engine result.

4.1. QoS based re-ranking

The first step is to build the meta search engine. We choose the three most popular search engines – Google, Yahoo and MSN Live Search as the underlying search engines. Since only the rank information is available from each of these search engines, we choose Condorcet-fuse as the rank aggregation method for the meta search engine.

In the second step, for each page in top *n* positions, we calculate all five QoS attributes based on the previous definitions, and normalize the values to (0, 1) range using a simple normalization method. The reliability value is in (0, 1) range already, and thus we don't do the normalization for this attribute. For the response time, because the page with a lower response time should be ranked higher, the normalization formula is as below:

$$NQoS_{ij} \square \frac{Max_j(QoS_{ij}) - QoS_{ij}}{Max_j(QoS_{ij}) - Min_j(QoS_{ij})}$$

Where QoS_{ij} is the value of the *j*th QoS attribute for page *i*, $NQoS_{ij}$ is the normalized value of QoS_{ij}, *Min* and *Max* are operations to get the minimum and maximum values. In the current system, *j* is a value from 1 to 5 because we have only 5 QoS attributes, and it could be expanded later.

For the other three attributes, because they are preference related, the page with a lower value could be ranked higher or lower depending on user's preference. If we take media richness as an example, when a user prefers a lower value on media richness, the normalization formula is the same as the previous one, when a user prefers a higher value, we would use a different formula as shown below:

$$NQoS_{ij} \square \frac{QoS_{ij} - Min_j(QoS_{ij})}{Max_j(QoS_{ij}) - Min_j(QoS_{ij})}$$

In the current experiment, we assume a lower value

ISSN: 2278 – 7798

is preferred on these three attributes. After the normalization step, we do the linear combination of the five normalized values, and each page would have an overall QoS value.

$$QoS_i = \sum_j w_j \cdot NQoS_{ij}$$

Where QoS_i is the overall QoS value of page i, and w_j is the weight of the jth QoS attribute. The value of this weight is between 0 to 1, 0 means that a user does not care about this QoS attribute, 1 means the QoS attribute is important to the user, any value in between could indicate the relative importance of the QoS attribute.

In this method, when a user submits a query to the meta search engine, it calculates the QoS values for the top n ranked pages, and then these n pages will be re-ranked based on their overall QoS values.

4.2. Rank aggregation with QoS attributes

Similar to the previous method, we also get the top n pages from the meta search engine and calculate the normalized five QoS values for each of them. Next step is different. We rank these n pages based on each QoS attribute, and afterwards we get five rankings. With the original ranking from the meta search engine, altogether we have six rankings. Based on users preferences, each QoS-based ranking has a weight, and we use the weighted Condorcet-fuse [8] to aggregate these six rankings.

4.3. Score combination with QoS attributes

In this method, we choose the linear combination. We don't have the scores returned from the meta search engine, and thus, we convert the rank to the score using the linear mapping [2]. Then we could combine it with the other five QoS values,

$$S_i = \alpha \cdot OS_i \; \beta \cdot \sum_j w_j \cdot NQoS_{ij}$$

Where S_i is the score of page i, OS_i is the converted score of page i from the original ranking, and α and β are parameters to measure the weight of each part of the score. Currently, we set α and β as 1, which means two parts of the score are equally important.

5. Experiment design and result analyses

We chose 36 queries as our query set [4]. We assume that the top 20 results are the most important pages for users and they would be viewed in most situations. Keeping this in mind, we only downloaded the top 20 pages from the meta search engine and calculated the QoS values for each of them. We believe that by re-ranking these 20 pages, we wouldn't sacrifice the relevancy of the results too much and in the mean time we could improve QoS of the results.

In order to calculate the QoS value more accurately, we downloaded each page ten times and then we used the average value for each QoS attribute. After that, we tested the proposed three combination methods. The weight of the QoS attribute could be either 0 or 1 as we explained before. If we consider all the possible combinations of five weights, there are altogether 31 of them, without considering the all zero case. Therefore, for each query, each method, there are 31 groups of results.

Since the main purpose of our proposed algorithms is to improve QoS of the search results, we use the top 10 QoS improvement to measure the effectiveness of the algorithm. It is defined as the improvement on the average QoS value of the top 10 results,

$$\Delta QoS = \frac{Avg_{i \in new-top10}(QoS_i) - Avg_{i \in old-top10}(QoS_i)}{Avg_{i \in old-top10}(QoS_i)}$$

In the following figures, T1 means the QoS based re-ranking, T2 means the linear score combination, and T3 means the Condorcet-fuse rank aggregation.

Figure 1 shows the comparison of T1, T2 and T3. Besides the improvement on five individual QoS attributes, the overall QoS improvement is alsodisplayed.

We could see that overall speaking, T1 is the best performing method regarding the improvement of overall QoS in the top 10 search results, and it is also consistently the best for all 5 attributes. T2 is the second best one, which is 18% worse than T1 and 28% better than T3. We could also see that there is a good improvement for all attributes except the reliability. The reason for its low improvement is that the original reliability is already very high, usually above 90%.

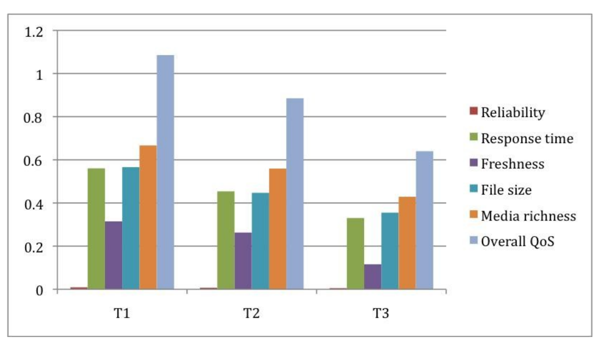

Figure 1. The comparison of 3 combination methods on average QoS

As we mentioned before, one of our target user groups is the mobile user and the user with slow Internet connections. For them, response time, file size and media richness are three most important QoS attributes. In Figure 2, we show the comparison when only three attributes are considered. If we only look at the overall QoS, T1 is still the best performing method, and T2 is slightly worse than T3. The main reason that T3 is better than T2 in this figure is that the freshness attribute is not included here. When the pages don't have the last modified date information, a predefined value is assigned, which makes the calculation of the freshness value not accurate. In this case, it seems that the rank aggregation method performs better than the score combination method, and thus T3 performs better than T2.

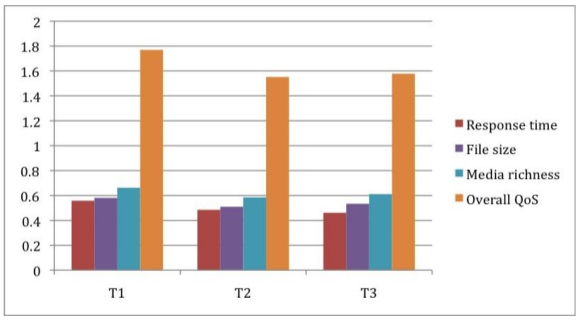

Figure 2. The comparison of 3 combination methods considering 3 QoS attributes only

6. Conclusions

In this paper, we defined five QoS attributes that are mainly performance related. By including them in the ranking procedure to re-rank the top positioned pages, users could have chance to see pages with a high delivery quality earlier than pages with a low quality, and thus it could improve users' searching experiences. The experiment showed the promising results. And by comparing three algorithms of combining QoS attributes, the QoS based re-ranking algorithm achieved the best result.

There are several directions of works we would like to work in the future. For instance, we would like to test our algorithm on a wireless device such as a cell phone or PDA, and on Internet connections with different speed. We would also like to implement a user interface so that users can choose their own preferences on QoS attributes, and a user study on their satisfaction level can be done afterwards.

7. References

[1] J. A. Aslam, and M. Montague, Models for Metasearch, in *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval,* 2001, pp. 276-284.

[2] T. Avrahami, L. Yau, S. Luo, and J. Callan. The Fedlemur Project: Federated Search in the Real World, *Journal of the American Society for Information Science and Technology*, 57(3):347-358, 2006.

[3] B. T. Bartell, G. W. Cottrell, and R. K. Below, Automatic Combination of Multiple Ranked Retrieval Systems, in *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval,* 1994, pp. 173-181.

[4] X. Chen, QoS-based Ranking for Web Search, Master Thesis, Ryerson University, 2008.

[5] E. A. Fox and J. A. Shaw, Combination of multiple searches, in *Proceedings of the 2nd Text REtrieval Conference,* 1994, pp. 243-252.

[6] HTTP – Hypertext Transfer Protocol Overview, http://www.w3.org/Protocols.

[7] S. Kapidakis, S. Terzis, and J. Sairamesh, A Management Architecture for Measuring and Monitoring the Behavior of Digital Libraries, in *Proceedings of the 2nd European Conference on esearch and Advanced Technology for Digital Libraries*, 1998, *R*pp. 95-114.

[8] M. Montague, and J. A. Aslam, Condorcet Fusion for Improved Retrieval, in *Proceedings of the 11th International Conference on Information and knowledge management,* 2002, pp. 538-548.

[9] S. P. Ran, A Model for Web Services Discovery with QoS, *ACM SIGecom Exchanges*, 4(1): 1-10, 2003.

[10] Nils Gruschka,Meiko Jensen, Luigi Lo lacono and Norbert Luttenberger, "Server-Side Streaming Processing of WS-Security", IEEE Transactions on Services Computing", Vol 4, No 4,pp. 272-285, 2011.

[11] Freddy Lecue and Nikolay Mehandjiev, "Seeking Quality of Web Service Composition in a Semantic Dimension", IEEE Transactions on

ISSN: 2278 – 7798

Knoweledge and Data Engineering, Vol 23, No 6, pp. 942-958, 2011.

[12] Hui Ma,Favyen Bastani,I-Ling Yen, and Hong Mei, "QoS-Driven Service Composition with Reconfigurable Services", IEEE Transactions on Services Computing", Vol 6, No 1,pp. 20-34,2013.

[13] Jian Wu, Liang Chen, Yipeng Feng, Zibin Zheng,Meng Chu Zhou,Zhaohui Wu, "Predicting Quality of Service for Selection by Neighborhood-Based Collaborative Filtering", IEEE Transactions on Systems, Man, And Cybernetics: Systems, Vol 43, No 2,pp. 428-439, 2013.

[14] Wattana Viriyasitavat and Andrew Martin," A Survey of Trust in Workflows and Relevant Contexts", IEEE Communications Surveys & Tutorials, Vol. 14, No. 3, pp 911-940, 2012.

Author Profile

F.Ezhil Mary Arasi Asst.Professor in SRM University has been serving as a the Education Profession for the past 10+ years. Earlier she was working as lecturer for St. Joseph's College, Trichy for about 3 years. Currently, she carries out Research in Service Oriented Architecture. She has guided more than 50 students in completing their MCA Graduation Projects. She has published papers in 3 International Journals.

Subodh Kumar, Pursuing Master of Computer Application(MCA) from SRM University, Chennai. Passed Bachelor of Computer Application(BCA) from Magadh University.

Aditya Anand, Pursuing Master of Computer Application(MCA) from SRM University, Chennai. Passed Bachelor of Computer Application(BCA) from SMU.

ISSN: 2278 – 7798