Client Perspective Based Documentation Related Over Query Outcomes from Numerous Web Databases

B. Santhosh Kumar *
b.santhoshkumar@gmail.com
Dept of Computer Science & Engg C S I College of Engineering, Ketti, The Nilgiris-643215, Tamilnadu, India.

L. Nanda Kumar
lnandud@gmail.com
Dept of Computer Science & Engg C S I College of Engineering, Ketti, The Nilgiris-643215, Tamilnadu, India.

Abstract

Documentation related which identifies the documentations that represent the same real-world entity, is an important step for data integration. A discovery shopping search engine tool is designed in order to remove the reproductions of documentation obtained from the query outcomes of numerous web databases and as well as to help shoppers make ideal buying decisions. To address the problem of documentation related in the web database scenario, we present an unsupervised, online documentation related method UDD which, for a given query can effectively identify reproductions from the query outcome documentations of numerous web databases. Most of the documentation related methods are supervised, which requires the client to provide training data. These methods are not applicable for the web database scenario. Hence an unsupervised and on-line approach with client perspective based search is offered for search within search.

Keywords: Documentation Similar reproduction detection, documentation linkage, Web database, query Outcome documentation, SVM.

1. Introduction

Although different book purchasing sites exists in the market. There exists an ambiguity among people to choose the best service and with lowest price. In order to eliminate this ambiguity a tool is developed that fetches the outcome for client given perspective and displays the most efficient outcomes to the client by laminating the reproduction of documentations. Web databases contain a much larger amount of high quality, usually structured information. Most web databases are only accessible via a query interface through which clients can submit queries. Once a query is received, the web server will retrieve the corresponding outcomes from the back end database and return them to the client.

2. Study on Online Shopping

A study is performed on the online existing system and the customer requirements are studied.

2.1. Customers

In recent years, online shopping has become popular, however it still caters to the middle and upper class. In order to shop online, one must be able to have access to a computer, a bank account and a debit card. Shopping has evolved with the growth of technology. Online shopping widened the target audience to men and women of the middle class. At first, the main clients of online shopping were young men with a high level of income and a university education. This profile is changing.

2.2. Customer Expectation

The main idea of online shopping is not just in having a good looking website that could be listed in a lot of search engines or the art behind the site. It also is not only just about disseminating

information, because it is also about building relationships and making money. Mostly, organizations try to adopt the techniques of online shopping without understanding these techniques and sound business model. Rather than supporting the organization's culture and brand name, the website should satisfy consumer's expectations. A majority of consumers choose online shopping for a faster and more efficient shopping experience. Many researchers notify that the uniqueness of the web has dissolved and the need for the design, which will be client centered, is very important.

2.3. Client Interface

It is important to take the country and customers into account. For example, in Japan privacy is very important and emotional involvement is more important on pension's site than on a shopping site. Nest to that, there is a difference in experience. Experienced clients focus more on the variables that directly influence the task, while clients are focusing more on understanding the information. There are several techniques for the inspection of the usability. Every technique has its own (dis-) advantages and it is therefore important to check per situation which technique is appropriate. When the customers went to the online shop, a couple of factors determine whether they will return to the site. The most important factors are the ease of use and the presence of client- friendly features.

2.4. Headings and Sections

One advantage of shopping online is being able to quickly seek out deals for items or services with many different vendors. Search engines online price comparison services and discovery shopping engines can be used to lookup sellers of a particular product or service. Shipping costs (if applicable) reduce the price advantage of online merchandise, though depending on the jurisdiction, a lack of sales tax may compensate for this.

Some retailers (Those who are selling small, high-value like electronics and books) offer free shipping on sufficiently large orders. Another major advantage for retailers is the ability to rapidly switch suppliers and vendors without disrupting client's shopping experience. The below figures 1 and 2 shows some of the query outcomes returned by two online bookstores, abebooks.com and amazon.com, in response to the query "Operating System Concepts" and " Wings of Fire" over the title field. It can be seen that the documentations obtained refer to the same book.

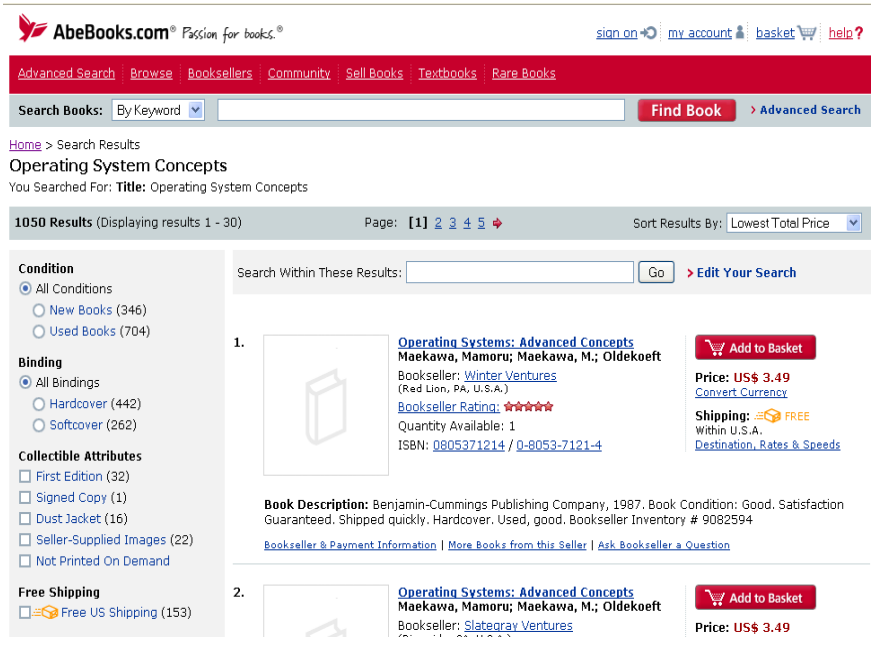

Figure 1: Query Outcomes from website-A

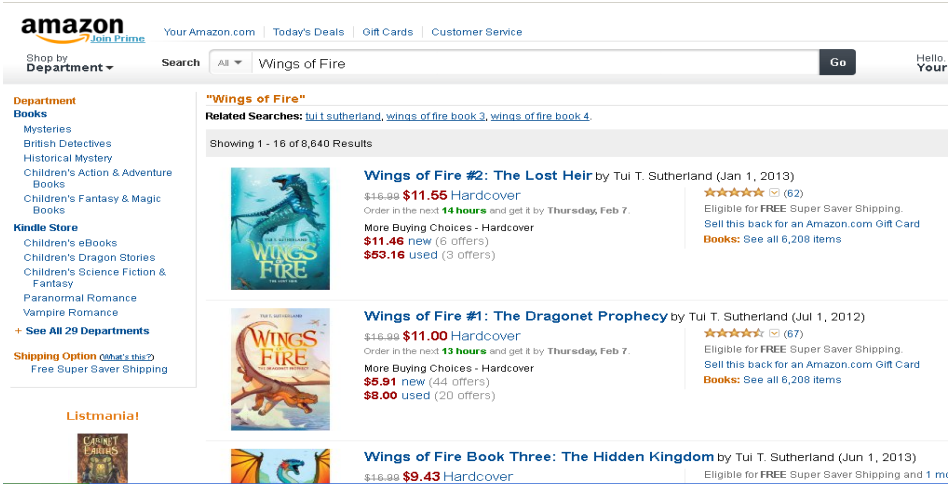

Figure 2: Query Outcomes from website-B

3. Related Works

In order to overcome the existing standard blocking and documentation clustering methods which requires to be supervised, unsupervised reproduction detection method is used in the proposed system in the removal of reproduction. The tool compares the query outcomes returned from numerous web databases and removes the reproductions.

3.1. Reproduction detection

Reproduction detection is done by means of weight age calculation and similarity calculation. The weightage calculation is carried by:

i. Comparing book's name, its author, price and its ISBN number of each documentation.
ii. Clients rating for documentation
iii. By information represent for each documentation.

Search within search, based on the client given perspective is performed and the outcome of detailed outcome which contains exact matched outcomes from various from the documentation, along with the weightage calculation as per the algorithm clients rating about the book is also considered.

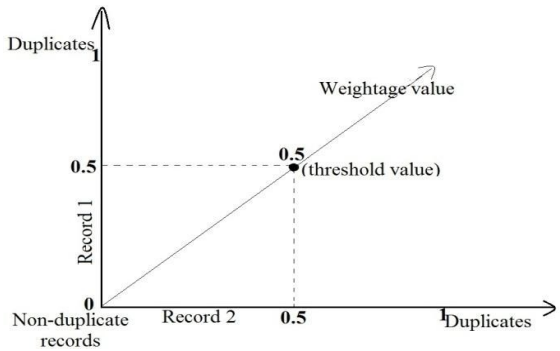

Figure 3: Documentation Comparison Graph

The above figure.3 represents the consideration of reproduction among the documentations by means of weightage value. Since the tool proposed is to be designed in java, it is easier to run in any operating system that supports java. This tool is a discovery shopping search engine with a mission to help shoppers make ideal buying decisions. Detailed search within search is performed, based on client given perspective i.e., another new search is performed for the client given perspective and the exact similar outcome is shown to the client obtained from various book purchasing websites. Some

more options such as adding of new book purchasing websites can be done for making more documentation. It can also be implemented for purchasing of other products through online.

4. Algorithms Used In Proposed System

The two algorithms, Unsupervised Duplicate Detection (UDD) and Support Vector Machine (SVM) are to be used in the proposed system.

4.1 Unsupervised Duplicate Detection

Our focus is on web databases from the same domain, i.e., Web databases that provide the same type of documentation in response to data source A and there are t documentations client queries. Suppose there are s documentations in data source B with each documentation having a set of fields/attributes. The goal of reproduction detection is to determine the similar status, i.e., reproduction or non-reproduction of this s*t documentation pairs.

4.1.1 Assumptions and Observations

In this section, we present the assumptions and observations on which UDD is based. First, we make the following two assumptions:

1. A global schema for the specific type of outcome documentations is predefined and each databases individual query outcomes schema has been matched to the global schema.
2. Documentation extractors i.e., wrappers are available for each source to extract the outcome data from HTML pages and insert them into a relational database according to the global.

Beside these two assumptions, we also make use of the following two observations:

1. The documentations from the same data source usually have the same format.
2. Most reproductions from the same data source can be identifies and removed using an exact similar method.

4.1.2 Problem Formulation

We formulate the reproduction detection problem following the completion of the exact similar step. We represent a pair of documentations P12 = {r1, r2}, where r1 and r2 can come from the same or different data sources, as a similarity vector V12=<v1, v2,……., vn> in which I represents the ith field similarity between r1 and r2:0<vi<=1in which vi=1 means that the ith fields of r1 and r2 are equal and vi=0 means that the ith fields of r1 and r2 are totally different. Note that UDD can employ any similarity function (one or numerous) to calculate the field similarity. Initially, two sets of vectors can be built.

1. A non-reproduction vector set N that includes similarity vectors formed by any two different documentations from the same data source.

2. A potential reproduction vector set P that includes all similarity vectors formed by any two documentations from different data sources.

Given the non-reproduction vector set N, our goal is to try to identify the set of actual reproduction vectors D from the potential reproduction vector set P.

Input: Potential reproduction vector set P
Non-reproduction vector set N
Output: Reproduction vector set D
C1: a classification algorithm with adjustable parameters W that identifies reproduction vector pairs from p
C2: a supervised classifier, e.g., SVM

Algorithm:

1. D=ø
2. Set the parameters W of C1 according to N
3. Use C1 to get a set of reproductions vector pair's d1 from P
4. Use C1 to get a set of reproductions vector pair's f from N
5. P = P - d1

6. While |d1| ≠ 0
7. N' = N - f
8. D = D + d1 + f
9. Train C2 using D and N'
10. Classify P using C2 and get a set of newly identified reproduction vector pair d2
11. P = P – d2
12. Adjust the parameters W of C1 according to N and D
13. Use C1 to get a new set of reproduction vector pair's d1 from P
14. Use C1 to get a new set of reproduction vector pair's f from N
15. N = N'
16. Return D

4.1.3 Weighted Component Similarity Summing (WCSS) Classifier

This classifier is represented as C1. In our algorithm classifier C1 plays a vital role. At the beginning, it is used to identify some reproduction vectors when there are no positive examples available. Then, after iteration begins, it is used again to cooperate with C2 to identify new reproduction vectors. Because no reproduction vectors are available initially, classifiers that need class information to train, such as decision tree and Naïve Bayes, cannot be used. An intuitive method to identify reproduction vectors is to assume that two documentations are reproductions if most of their fields that are under consideration are similar. To evaluate the similarity between two documentations, we combine the values of each component in the similarity vector. As illustrated, different fields may have different importance when we decide whether two documentations are reproductions. The important is usually data-dependent, which in turn, depends on the query in the web database scenario. Hence, we define the similarity between documentations r1 and r2 as

$$Sim(r_1, r_2) = \sum w_i * v_i \qquad \sum w_i = 1$$

4.1.4 Component Weight Assignment

In the WCSS classifier, we assign a weight to a component to indicate the importance of its corresponding field under the condition that the sum of all component weights is equal to 1. The component weight assignment algorithm is shown below. The intuition for the weight assignment includes:

1. Reproduction intuition: The similarity between two reproduction documentations should be close to 1. For a reproduction vector V12 that is formed by a pair of reproduction documentations r1 and r2, we need to assign large weights to the components with large similarity values and small weights to the components with small similarity values (lines 4-8).

2. Non reproduction intuition: The similarity for two non-reproduction documentations should be close to 0. Hence, for a non-reproduction vector V12 that is formed by a pair of non-reproduction documentations r1 and r2, we need to assign small weights to the components with large values and large weights to the components with small similarity values (lines 9-14)

Input: Reproduction vector set D
Non-reproduction vector set N
Weighting scheme co-efficient a
Output: Component Weight W

Algorithm:
1. For i=1 to n
2. pi=0
3. qi=0
4. For each vector Vk={vk1,…,vkn} in D
5. Pi=pi+vki
6. S=
7. For i=1 to n
8. Wdi=pi/S
9. For each vector Vk={vk1,…,vkn} in N

10. qi=qi+1-vki
11. S=
12. For i=1 to n
13. Wni=pi/S
14. Wi = a.wdi +(1-a)wni
15. Return W={w1,…,wn}

4.2 Support Vector Machine Classifier

Support Vector Machine Classifier is represented as C2. After detecting a few reproduction vectors whose similarity scores are bigger than the threshold using the WCSS classifier, we have positive examples, the identified reproduction vectors in D, and negative examples namely the remaining non reproduction vectors in N0. Hence we can train another classifier C2 and use this trained classifier to identify new reproduction vectors from the remaining potential reproduction vectors in P and the non-reproduction vectors in N0. First, it should not be sensitive to the relative size of the positive and negative examples because the size of the negative examples is usually much bigger than the size of positive examples. This is especially the case at the beginning of the reproduction vector detection iterations when a limited number of reproductions are detected. Another requirement is that the classifier should work well given limited examples. Because our algorithm identifies reproduction vectors in an iterative way, any incorrect identification due to noise during the first several iterations, when the number of positive examples is limited will greatly affect the final outcome. According to [6], Support Vector Machine (SVM), which is known to be insensitive to the number of training examples satisfies all the desired requirements and is selected for use in UDD. Because our algorithm will be used for online reproduction detection, we use a linear kernel which is the fastest as the kernel function in our experiments.

5. Conclusion

This proposed approach works well with greater efficiency for searching and buying of books and it is an unsupervised on-line approach client perspective based search for detecting reproductions over the query outcomes from numerous web databases. The tool is proposed with a mission to help shoppers make ideal buying decisions with the help of client perspective based search which gives clients a various options to choose from the list of outcomes obtained from numerous web databases of different book purchasing websites.

6. Acknowledgements

This research is supported by the management of C.S.I College of Engineering, Ketti, The Nilgiris, whose support we are pleased to acknowledge. We are also grateful to our colleagues in CSICE for useful discussions, and thank to our beloved CSICE students.

7. References

1. Weifeng Su, Jiying Wang, Lochovsky F.H, Record Matching Over Query results from Multiple Web databases", IEEE Transactions on Knowledge and Data Engineering, Vol:22, Iss:4, pp. 578-589, 2010
2. O. Bennjelloun, H. Garcia-Molina, D. Menestrina, Q. Su, S.E. Whang, and J. Widom, "Swoosh: A Generic Approach to Entity Resolution," The VLDB J., vol. 18, no. 1, pp. 255-276, 2009.
3.
4. P. Christen, "Automatic Record Linkage Using Seeded Nearest Neighbour and Support Vector machine Classification," Proc. ACM SIGKDD, pp. 151-159, 2008.
5.
6. R. Ananthakrishna, S. Chaudhuri, and V. Ganti, "Eliminating Fuzzy Reproductions in Data Warehouses," Proc. 28th Int'l Conf. Very Large Data Bases, pp. 586-597, 2002.
7. R. Baeza- Yates and B. Ribeiro-Neto, Modern Information Retrieval. ACM Press, 1999.

8. K. Simon and G. Lausen, "ViP ER: Augmenting Automatic Information Extraction with Visual Perceptions," Proc. 14th ACM Int'l Conf. Information and Knowledge Management, pp. 381-388, 2005.

9. Y. Zhai and B. Liu, "Structured Data Extraction from the Web Based on Partial Tree Alignment," IEEE Transactions Knowledge and Data Eng., vol. 18, no.12, pp. 1614-1628, Dec. 2006.

10. H. Zhao,W. Meng, A. Wu,V. Raghavan, and C.Yu," Fully Automatic Wrapper Automatic Wrapper Generation for search Engines," Proc. 14th World Wide Web Conf., pp. 66-75, 2005.

11. V. Vapnik, The Nature of Statistical Learning Theory, second ed. Springer, 2000.

12. B.He and K.C.-C. Chang, "Automatic Complex Schema Similar Across Web Query Interfaces: A Correlation Mining Approach," ACM Trans. Database Systems, vol.31, no.1, pp. 346-396, 2006.

13. W. Su, J. Wang, and F.H. Lochovsky, "Holistic Schema Similar for Web Query Interfaces," P roc. 10th Int'l. Conf. Extending Database Technology, pp. 77-94, 2006.

14. P. Christen, T. Churches, and M. Hegland, "Febrl— A Parallel Open Source Data Linkage System," Advances in Knowledge Discovery and Data Mining, pp. 638-647, Springer, 2004.

15. W.W. Cohen, H. Kautz, and D. McAllester, "Hardening Soft information Sources," Proc. ACM SIGKDD, pp. 255-259, 2000.