ONE WAY CORROBORATION: SERVICE SELECTION FOR WEB-BASED BUSINESS PROCESSES

K. Gopichand Reddy[1], J Armstrong Paulson [2]

[1] Pursuing M.Tech (CS), [2]Associate Professor Nalanda Institute of Engineering & Technology Siddharth Nagar, Kantepudi(V),Sattenapalli(M),Guntur (D)-522438, (India)

ABSTRACT

Internet has covered the way for the appearance of web databases. Data mining is procedure used by organizations to turn raw data into useful information. Data mining such databases for required information has become a common task. Data mining can be a reason for apprehension when only selected data, which is not delegate of the overall sample group, is used to establish a certain theory. There are many algorithms which are applied to get the exact data which is used by the user. The obtainable solutions such as user's profile, query logs and database principles execute ranking in user independent and/or query independent method. This can't give effective ranking strategy. This paper presents a new approach known as Query and User Dependent Ranking for giving ranking to query results of deep web databases. Search engine has got two types of searches i.e. User search and Query search. One of the problems in this context is ranking the results of a user query. We are going to discuss about these two techniques and also going to show an advanced search where we can tell that it is more efficient than the existing techniques. Data Mining is employed here and also the ranking algorithm is used for the query searches done by the user on a particular data from the raw database. To get the efficient output expected by user we have used the Ranking algorithm which is associated with the workload. The representation is based on the perception that comparable users show comparable ranking preferences over the results of similar queries. Then we describe these similarities properly in different ways and discuss their effectiveness analytically and experimentally over two distinct Web databases.

Index Terms: Raw Data, Web Databases, Data Mining, Query Search, Ranking Algorithm.

I INTRODUCTION

Data Mining is the very recognizable perception in Computer knowledge which helps users to get related data as of the raw data; it earnings that by the accomplishment of a data mining algorithm we can excavation the functional data from very huge data. As like Data Mining here is also one more term which is unfathomable Web. Profound Web is also called as Deepnet which is not noticeable to the user; it is the World Wide Web content that is not division of the exterior web. Profound web is the terms which tell that the standard user cannot reach the root of it i.e. not all the users can find the information from that detailed investigate steam engine only the sanctioned user's can admission all the information. For example, some university, administration agency and other organization preserve database of information that were not formed for broad

public access. Additional sites might limit database right to use to member or subscribers. Unfathomable web although it is not recognized to numerous of the user but they contain come across these types of websites, the websites which give access to their data only after a genuine confirmation of that user. For illustration, we have a website "Slideshare.net" which gives information only after you confirm yourself and only after this check that user can get data; point to be taken into deliberation is that we want not pay any quantity to that association for access the data. On the similar time we have Google explore engine which never asks a user to get register for the data access from their database.

Here we require recognizing a point here which is from Google we cannot get any data it's just an arrangement tool which helps user to find the way to his fortune from the Google search engine. For illustration, at any time we type for thorough content from Google it gives us entirety of links of other domain which keep up that data of our explore. But if we evaluate this with Slideshare.net, here user will search for his requirement and that domain will get the results closely matching to the need in this domain itself but it will not give the links of other domains from where a user can get his data.

To tell more about the work in data mining there are two possible scenarios i.e. user search and query search. Both these techniques are used for searching the data but the difference is that in user search the operation will be performed based on the user requirements and in the case of query search the operation will performed on the system based. Both the techniques use database query for the search and so the design used for the retrieval of the information will be purely on the different queries used by the developer of that domain. When user gives the requirements about the data the query will add the details using the AND/OR operators in the queries to search the data from the database. Each dataset that comes out from the DB when the search is performed is termed as tuples. Example: Suppose there are two users who want to know details about a bike which is available in the market and so we track the details provided by them for their search i.e. When first user searches the values provided by him are "made=Yamaha and year=2013" and as the input is not specific we get many tuples i.e. we get many options from which again a user to select the one he wants. Suppose the input was "made=Yamaha and price=50000 and year=2014" then also we get tuples but the no. of tuples compared to above result will be less. From this example, we can come to a conclusion that when we have more input parameters for the search the search will be more optimized and when the options are less the time taken to get the exact result will be more. Also from the queries we can see that operators like AND or OR can be used to fetch the results from the database.

II RELATED WORK

This describes about the searches that will be performed and also the comparison between them. There are two types of searches that will be performed on the search engines i.e. user search and query search. In this paper, we are adding a new type of search i.e. advance search. Advance search is for fetching the data from the database in a smart manner i.e. the data will be retrieved even with the less details that user gives to the search engine. User search is a type of mechanism where the user gives his inputs to fetch the data from that organization but fails at the first try because the data which the user gives is very less as an input and expects the

exact result but the search engine does not have the capability to get the user expected result with the minimum values specified by the user. When the user can see different tuples that have been fetched by the application and seeing them the user changes the input given and again based on the second input the system gives some output to the user. Again the process is same i.e. seeing the tuples generated the user will select the data which he wants from the generated output and if still the requirement is not matching then again a new set of input is given till the expected output is not achieved.

Example: The inputs can be given in this manner

a. "made=Yamaha and year=2014"
b. "made=Yamaha and year=2014 and color=blue"
c. "made=Yamaha and year=2014 and model=abc and color=blue"

From the above four set of data points we can tell that the user started searching for the bike and could get the output what he was actually looking for so again the inputs were modified. In this manner the user was continuously changing the inputs at fourth try he got the exact output, the result can be expected appropriately when the requirement from the user is accurate and the search engine will not have any issue for searching the data from its database. We have a web portal called "olx.in" which uses the details provided by user to fetch the results from the database and again the user has a search bar where if the output what was seen dint match with the requirement then the user can change his requirement to get the data of his choice. Here the user is given chance to explore his thought for the type of requirement which he is actually looking for. This type of search actually refers to User search.

We have other type of search which is Query search. Here the user need to select the type of requirement and based on that the query will be generated by the application. With the query that is generated the result will be displayed to the user and if still the output shown to user is not matching the criteria then user has got the flexibility to modify the inputs and perform the search again with the new set of data. For every search that is being performed by the user a new query will be generated and will be tracked by the application and with this queries that are mined by the application later can be used to compute the results based on the query similarity. The term query similarity means that if a particular user is searching with his requirement then that requirement may match with some other user also for better results and feasibility the queries will be mined and used for the later purpose.

III PROPOSED SYSTEM AND ARCHITECTURE

Query Search is a search which can be found in many of the web portals, one which I would like to tell is "watchkart.com". This portal helps the user to give his requirement in the form of UI designed by the application and once the user has selected his requirements then the output will be generated for user and here also we get tuples and from which if the user is not satisfied then the user has got the flexibility to modify the search pattern and give different requirement. We will now see the difference between user search and query search, user search mainly depends on the user input i.e. when some input is received from the user these values

are substituted into the DB query and results will be fetched but where as in query search the user will select from the options given and a query will be generated with the set of data that user has selected. From the above said point it is very clear that in user search the query is predefined and only the inputs will be substituted but where as in query search every time a new set of input data is given a new query will be generated and this data will be mined for future purpose i.e. when a new user is giving same type of requirements then it will be selected from already created rather creating a new one. Also in this process when the same query is being generated number of times we are going to implement a ranking algorithm to know the rank of the queries which are being generated by different users for their requirements. Based on the queries a ranking function will be generated and when the queries are generated they also give rise to work load which will be defined in detail.

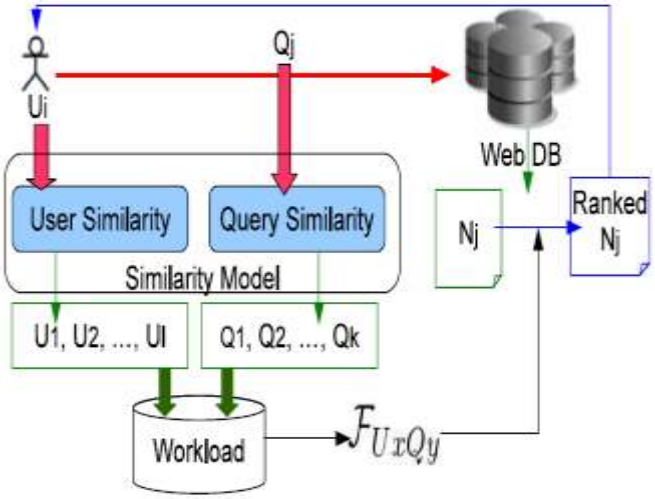

Fig 1: Similarity Model for Ranking.

Workload is nothing but the query generation and interaction with the database for the information retrieval.

Query	Make	Location	Price	Mileage	Color
Q1	Toyota	US	$1000	any	White
Q2	Lexus	India	any	any	Grey
Q3	Honda	Australia	$1500	any	Silver
Q4	any	China	any	any	White

Fig 2: Different queries from users

When a user gives his requirements for search then the application will give set of tuples i.e. matching the requirement different values will be shown and if the user is satisfied the result will be generated and if the requirement is not fulfilled then the input will vary for the second time and if still it does not match the requirement the process goes on till the requirement is fulfilled. For example assume the below outputs that we can get on different search mechanisms.

Number of users searching this type of item can be calculated and to know the query similarity the equation is,

$$\text{Similarity } (Q,Q') = \prod_{i=1}^{m} sim(Q[A_i = a_i], Q'[A_i = a_i'])$$

There is also a chance that the tuples may also be same and that similarity can be calculated as,

$$\text{Sim } (T,T') = \sum_{i=1}^{m} sim(t_i, t_i')$$

So far, we have discussed about the user search and query search and now the discussion is about the advance search that we are going to interject into this implementation for the best results. Advance search is going to fetch the data from the various databases and also it's not going to tell the user to give some more requirements for the best result, it is going to fetch the result from the database based on the data given by user for the first time itself and the main logic involved here is that it is going to display the content to the user which has the more rank and which was taken by more number of users i.e. it means that when a user has got some data and if that data is consumed by applying the ranking algorithm the count is incremented and in this manner where many users like this user consume this data then that data with user specific query will get good rank and so because of it that data will be presented to the user. Advance search is the best optimized one and the main logic part is similar to the user search and in the database query the input from user will be substituted with the like keyword. Below is the process for getting the ranking functions from the different user queries that are accepted by the application.

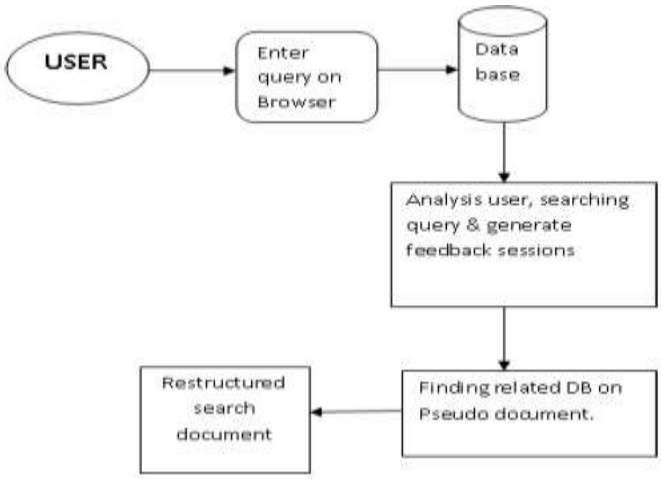

Fig 3: Architecture Diagram

Algorithm: Deriving Ranking Functions from Workload

Input : U_i, Q_i, Workload W (M queries, N users)

Output : Ranking Function F_{xy} to be used for U_i, Q_j

Step 1: for p = 1 to M do

Calculate Query Condition Similarity (Q_j, Q_p)

end for

Sort(Q_1, Q_2; . . .Q_M)

Select Q_{Kset} i.e., top-K queries from the above sorted set

Step 2: for r = 1 to N do

Calculate User Similarity (U_i, U_r) over Q_{Kset}

end for

Sort(U_1, U_2; . . .U_N) to yield U_{set}

Step 3: for Each $Q_s \in Q_{Kset}$ do

for Each $U_t \in U_{set}$ do

$Rank(U_t;Q_s) = Rank(U_t \in U_{set}) + Rank(Q_s \in Q_{Kset})$

end for

end for

Where in the above process,

U –user specific, Q – queries, F – function, W – workload.

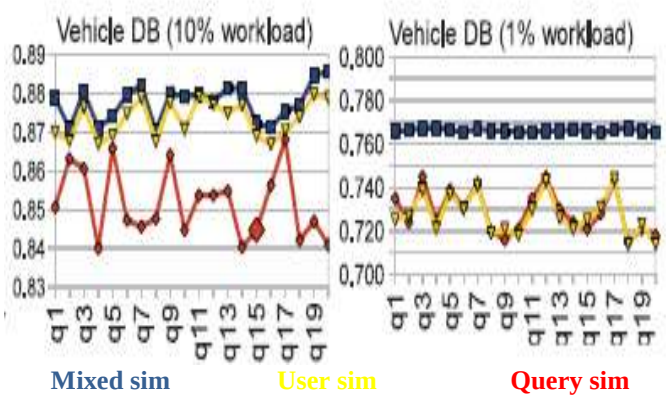

Fig 4: Ranking quality of a combined similarity model.

IV CONCLUSION

In this paper, we projected a user and query reliant solution for ranking query results for Web databases. We appropriately distinct the likeness models such as user, query and combined and accessible untried consequences over two Web databases to confirm our analysis. We confirmed the expediency of our conclusion for real-life databases. Further, we discuss the problem of establish a workload, and presented a learning method for inferring individual ranking functions. And we have proposed an advance search option for the various users using this application to get the result in very less time. Apart from this we have also shown the implementation of user search and query search also shown the feasibility report between them. Moreover we have used the multiple web databases in this implementation to show the results gathered from different databases depending on the user requirement. We demonstrated the practicality of our implementation for real-life databases. Further, we discussed the problem of establishing a workload, and presented a learning method for inferring individual ranking functions. In this paper apart from the user and query search we have also imparted the ranking functionality which is useful in the advance search as well as the query search.

REFERENCES

[1] S. Agrawal, S. Chaudhuri, G. Das, and A. Gionis. Automated ranking of database query results. In *CIDR*, 2003.

[2] C. Basu, H. Hirsh, and W.W. Cohen, "Recommendation as Classification: Using Social and Content-Based Information in Recommendation".

[3] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press, 1999.

[4] M.K. Bergman, "The Deep Web: Surfacing Hidden Value," J. Electronic Publishing.

[5] S. Chaudhuri, G. Das, V. Hristidis, and G. Weikum, "Probabilistic Ranking of Database Query Results,"

[6] Z. Xu, X. Luo, and W. Lu, "Association Link Network: An Incremental Semantic Data Model on Organizing Web Resources,"

[7] L. Wu et al., "Falcon: Feedback Adaptive Loop for Content-Based Retrieval,"

[8] Telang, S. Chakravarthy, and C. Li, "Establishing a Workload for Ranking in Web Databases," technical report, UT Arlington, http://cse.uta.edu/research/Publications/Downloads/CSE- 2010-3.pdf, 2010

AUTHOR PROFILE

K. Gopichand reddy is currently pursuing M.Tech in the Department of Computer Science, from Nalanda Institute of Engineering & Technology (NIET), siddharth Nagar, Kantepudi(V), Sattenapalli (M), Guntur (D), Andhra Pradesh , Affiliated to JNTU-KAKINADA.

J Armstrong Paulson working as Associate Professor at Nalanda Institute of Engineering & Technology (NIET), siddharth Nagar, Kantepudi(V), Sattenapalli (M), Guntur (D), Andhra Pradesh , Affiliated to JNTU-KAKINADA.