

## A Research on More Effective Model of Deep Web Integration

Hanmei Liu, Shitao Wan<sup>+</sup> and Gang Liu

Institute of Computer Science and Engineering Changchun University of Technology, Yan'an Street  
2055, Changchun 130012, China

**Abstract.** Deep web contains a great deal of high qualitative information, but it is a difficult job to use this information with a unified model. Nowadays, many people make a lot of research on key techniques of deep web integration, such as deep web source's finding[1][2] and classing[2][3] etc. This paper put forward a new model of deep web integration. It mainly puts tedious process—query processing module and query results processing module into background processing. According to the keywords which were submitted by users, it firstly looks up in local database, and then looks up in web databases. The important significance is that it improves the speed of response.

**Keywords:** model, complications of time, complications of space

### 1. Main text

When you use a search engine like Google or Yahoo!, the information you get back is sometimes referred to as the "Surface Web" or the "Visible Web." However, there's a lot more information out there - There are millions of web pages that Google and Yahoo can't find. That's the Deep Web. The Deep Web

- Consist of pages that are not part of the World Wide Web that you see .This information is invisible to the user.
- Is also known as the "invisible web".
- Is stored in special database.
- Often includes sites that require registration or have limited access. This is so that search engines can't browse them. For example, you have to log on to database like Academic Search Premier to search or browse.
- Contain information such as directories and breaking news [4].

In summary, internet can be divided into surface web and deep web. Surface web is pages that can be indexed by traditional search engine via hyperlink. Deep web includes a good many of useful information, and it covers all fields of the society. The problem is how to use this high quality information conveniently. Today, the study of deep web integration has made a lot of impressive achievements .But it still have a distance to actual application.

### 2. Theory basis and experiment data

Present stage many researches on deep web integration depend on the model (Fig.1).

---

<sup>+</sup> Corresponding author. Tel.: 15584457466.  
E-mail address: wst.521@163.com.

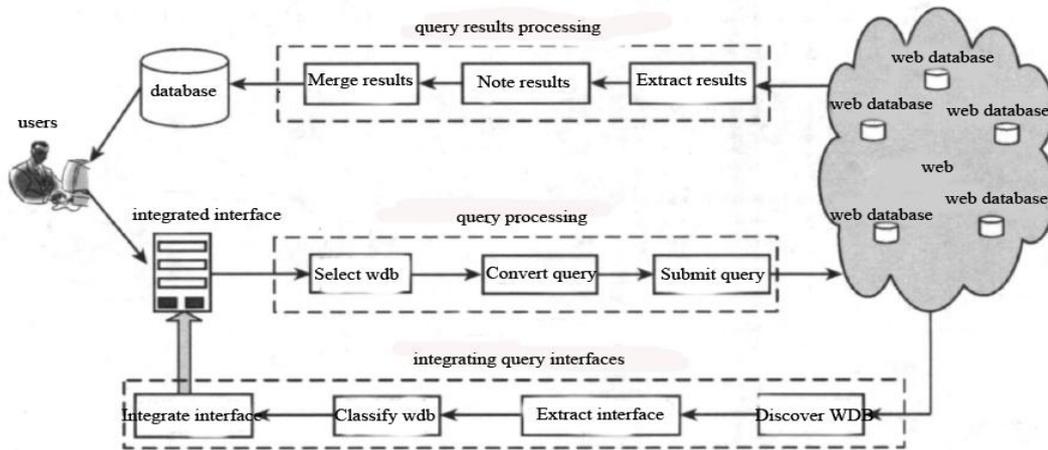


Fig.1: Frame of deep web integration.

This model consists of three modules: integrating query interfaces module, query processing module and query results processing module. The process—searching content user wanted and merging results that returned by each on-line database is considered the most time consuming part. It has a lot of reasons: the effect of network condition, the different design model among these on-line databases and the speed of response of each on-line database. Besides, the part of merging returned results should consider how to take on these web pages. Finally, system should delete repeat content, range content and generate web pages which user would see. We assume that this system composed by  $N$  web databases, and suppose one user, one submission. Then system would create  $N$  threads and they carry out query. Assuming that each on-line database's access time are  $T^i (i=1,2,\dots)$ , obviously, using linear dispatch, even it don't consider time of

$$\sum_{i=1}^N T_i$$

semantic label, synthesizing web pages and removing repeats, the total time is  $\sum_{i=1}^N T_i$ . We are easy to see that its efficiency is too low. Naturally intercurrent dispatch would be adopted. But, when  $M$  users access the system at the same time, dispatch algorithm would be more complicated. And system would take  $(M+1)*N$  thread's dispatch into account. Meantime, it needs to consider restriction in different mode of each web database [3].

## 2.1. Analysis for complication of time and space of Fig.1

Hypothesis:  $T^1$  denote integrating query interfaces module;  $T^2$  denote query processing module;  $T^3$  denote query results processing module;  $T^0$  denote the part of searching in web database. The time of each submodule:  $T^{00}$  denote a access to one web database, suppose it contains  $H$  web database, then the total time is  $H * T^{00}$ ;  $T^{11}$  denote web database's discover;  $T^{12}$  denote extract of query interface schema;  $T^{13}$  denote classing web databases;  $T^{14}$  denote integrating query interfaces;  $T^{21}$  denote selecting web databases;  $T^{22}$  denote conversion of query;  $T^{23}$  denote submitting query;  $T^{31}$  denote extract of query results;  $T^{32}$  denote results' note;  $T^{33}$  denote merging results.

In user's opinion,  $T^1$  is done in prophase. Time is spend on  $T^0$ ,  $T^2$ , and  $T^3$ . Then user submit one query and see results would spend total time  $T = T^0 + T^2 + T^3$ . And,  $T^0 = H * T^{00}$ ,  $T^2 = T^{21} + T^{22} + T^{23}$ ,  $T^3 = T^{31} + T^{32} + T^{33}$ . So,  $T = H * T^{00} + T^{21} + T^{22} + T^{23} + T^{31} + T^{32} + T^{33}$ . This model's complication of space is low, because it doesn't impropriate much more storage. It wills impropriate much more CPU.

## 2.2. Experiment data and its analysis

Many researches express that the most satisfied time of open web page is lower than 2s. And the time which users can stand is 8s. That is to say, 8s is a critical value. If a website be opened more than 8s, it is very possible for users to leave this website [6]. So it is a question that must be solved.

Before this, I do a survey. For some field; several databases can cover most content of this field. For example, it is obvious at railroad information service. It is provided by government or related authority, such as the railroad customer service center of China (<http://dynamic.12306.cn/TrainQuery/trainPassStationByTrainCode.jsp>), India railroad service net

(<http://www.indianrail.gov.in/>), British railroad information net (<http://www.trainenquiry.com/indexNS.aspx>) and European railway network (<http://www.raileurope.cn/>). So it is adequate to update data with several web databases. Besides, we observe some e-commerce website. We input “C++” in query interface of Dangdang web and Amazon web, and it return 1590 records, 992 records. These records cover both domestic and overseas relevant books. They cover from introduction to proficiency of C++ and its application in every field. But in fact user needn't to look over 2582 records. Perhaps user wants only one book. When the system is integrating records, what need may be the adoption of top N records. We once did a meticulous survey about each shopping website, and mainly look the access quantity of IP and PV (page view). Meantime, we range these websites. Results are follows (March.31th.2011) [7]:

Table 1. Shopping website taxis

Website name	Website address	mean of one day's IP access quantity[one week]	mean of one day's PV access quantity[one week]
Taobao	www.taobao.com	23,340,000	490,140,000
eBay	www.ebay.com	16,134,000	307,191,360
jindongshangcheng	www.360buy.com	3,282,000	35,511,240
fankechengpin	www.vancl.com	1,650,000	14,355,000
paipaiwang	www.paipai.com	2,004,000	12,945,840
dangdangwang	www.dangdang.com	1,302,000	12,759,600
amazon	www.amazon.com	1,920,000	12,230,400
walmart	www.walmart.com	2,112,000	12,080,640
bestbuy	www.bestbuy.com	1,632,000	10,559,040
target	www.target.com	1,710,000	8,652,600
360huicongwang	www.hc360.com	1,332,000	7,738,920
lashouwang	www.lashou.com	1,794,000	5,884,320
meituanwang	www.meituan.com	1,185,000	4,261,440
nuomiwang	www.nuomi.com	570,000	1,710,000
24tuan	www.24tuan.com	414,000	1,204,740
Ftuan	www.ftuan.com	294,000	1,146,600
maigouwang	www.maigoo.com	90,600	190,260
yiqu	www.eachnet.com	32,400	84,240
yibiargou	www.1b2g.com	4,800	9,120
dongdongpu	www.dongdongpu.com	720	1,080

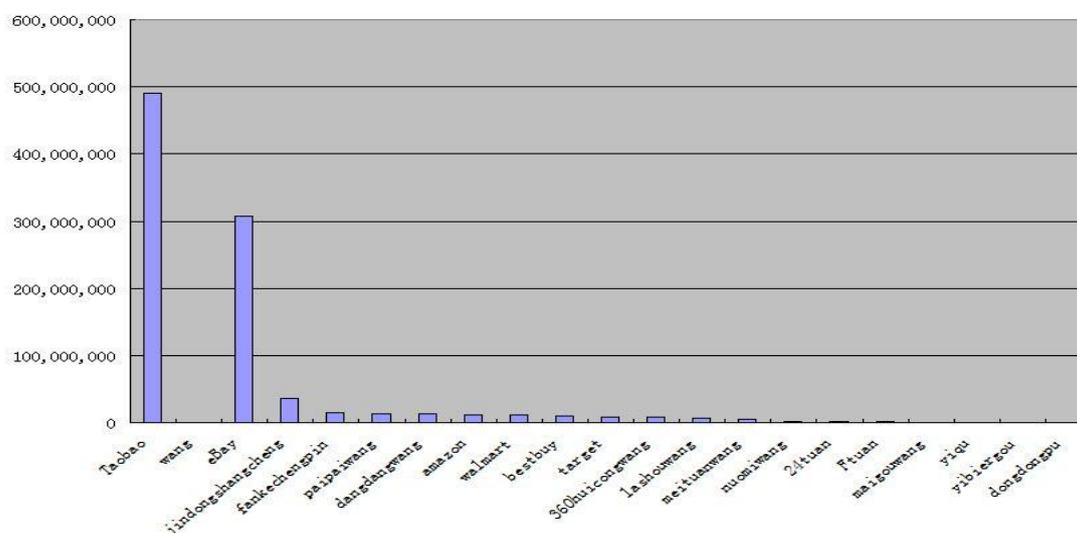


Fig. 2. Shopping website access quantity histogram.

From Fig.2 we can see that the first website's access quantity is 490,140,000 more than the sum of rest websites' access quantity 448,516,440. This full elucidation that we needn't integrate all interfaces of all websites, and only integrates several websites, such as fore ten websites of TABLE.I. Fore ten websites' access quantity is 916,425,720; all websites' quantity is 938,656,440; former accounted for 97.6% of latter. Indeed, access quantity of fore two websites accounted for 84.9% of all access quantity.

### 3. Diagram of structure, flow chart and algorithm and its analysis

When we want to integrate deep web, we have two different ways: one for one field and another for many fields. In the process of studying it we discover that if we do it for all fields, not only the workload is heavy but also it will consume more resource of hardware and software. What's more, user won't stand time of response. A compromised way is integrating deep web for one field. A new integrated model is proposed in this paper. Firstly, the system put contents returned by web databases into local database, and then allows users to query the local database. In order to satisfy users, the system should update local database according to certain strategy and return satisfactory results within the time which the users stand.

Now we put the query processing module and query results processing module in background. Users are allowed to use a complete system which linked to local database that stored latest and foremost information. When local database can't meet users' demand, the system would convert to web database to seek information. Meantime, it updates contents in local database according to certain strategy, check changes in web databases and map it to integrated query interface.

#### 3.1. Structure figure(Fig.3)

WDB denote web database.

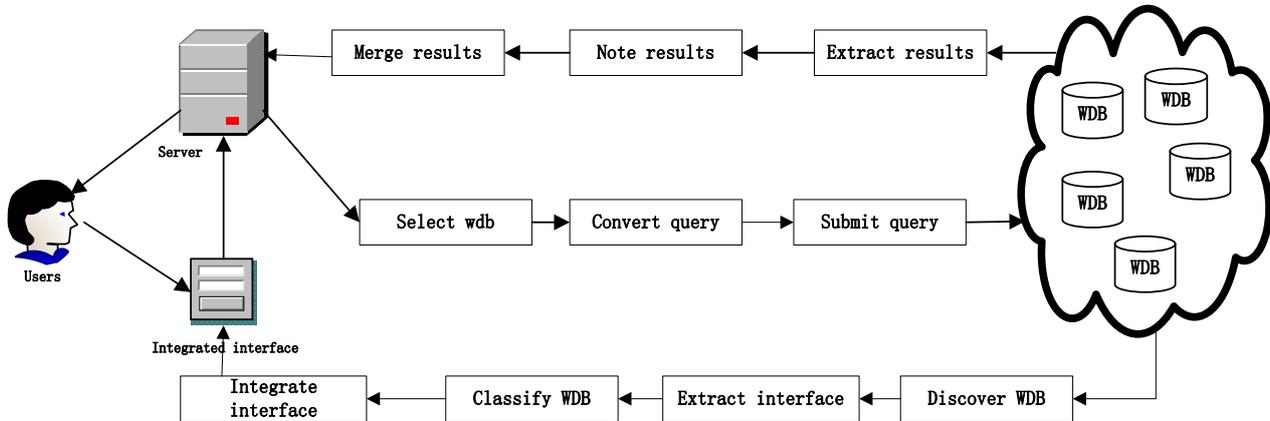


Fig. 3.The new frame of deep web integration.

#### 3.2. Flow chart(Fig.4)

Top n WDB denote this field web database that is new and authoritative; REST denotes this field's WDB except top n WDB.

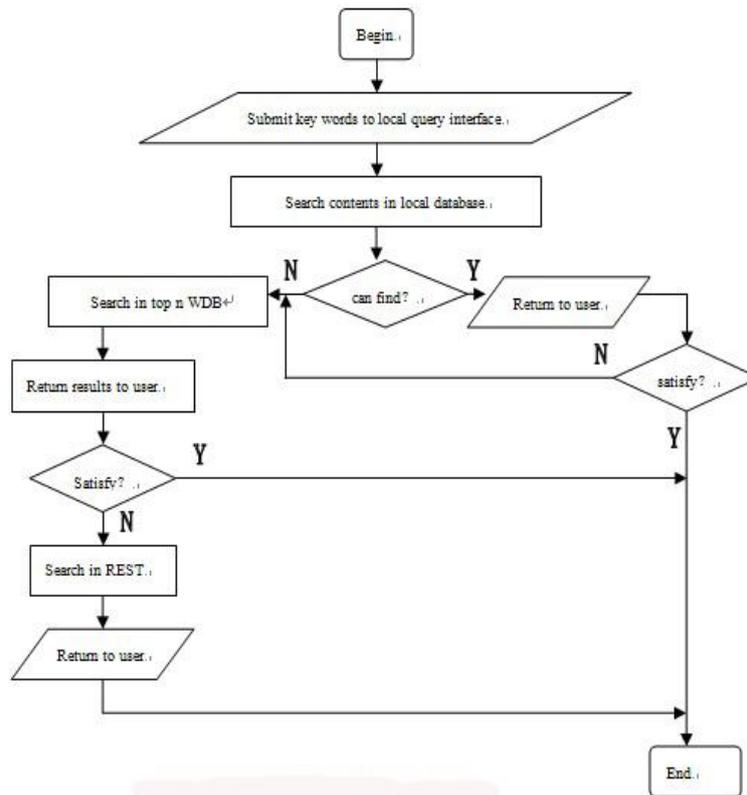


Fig. 4.Flow chart of new frame

### 3.3. Algorithm descriptions

#### Algorithm 1

Input: key words

Output: inquiry result

BEGIN

L=Keywords;

Parse (L); //parse key words

Match (L); // let L match with records in tables

if (success == 1) // "success" denote whether success or not; 1 denote successful ,else unsuccessful.

{

    output (result); //output all results matched

    if(satisfy == 1) //user is satisfied

        return 1;

    else

replay: {

    output research(Top N); //search in most authoritative web database in this field, and //return results to user.

    if (satisfy == 1)

        return 1;

    else

{

    output(Rest); //search in rest web databases in this field, and return results to user.

        if(success == 1)

            return 1;

        else

{

        Output fail;

        return 0;

    }}}

```

else
  goto replay;
END
Algorithm 2
parse (L)
{
  //split and convert key words
  while(the rest of L remain to split )
  {

    Mi = malloc sizeof(words);
    Mi = separate L's rest part;
    translate Mi into styles; //styles is that can be receive

  }
  Return L =  $\sum M_i$ ; //i=1,2,.....; that was logistic link
}

```

```

Algorithm 3
match (L)
{// let L match with records in tables
  while (Table. record != NULL) //it not check to //the last
  {
    i = strcmp(L, Table.record);
    if(i == 0)
      output Table.record;
      success = 1;
      Table.record++;
  }
  return all matched Table. record;
}

```

```

Algorithm 4
research (Top N)
{// search in most authoritative web database in this field, and return results to user.
  for (int i=0; i<N; i++)
  {
    submit L to DB[i];
    crawl DB[i]'s result to table[i]; //this need web crawler.
    combine tables into a webpage;
  }
  /*replace tables in local database with Lately- got records according to strategy */
  replace LocalDB.Tables.record with this tables.record; //LocalDB is local database
}
return the webpage;
}

```

This model's analysis for complication of time and space:

We assume that the time of searching in local database is  $TT$  and mean rate of success is  $p$ . So the mean time is follow:

$$T = TT * P + (H * T00 + T21 + T22 + T23 + T31 + T32 + T33) * (1 - P)$$

Measures that be taken is to improve  $p$  and to reduce  $H$ . Following discussion, we would analyze time function of this model.

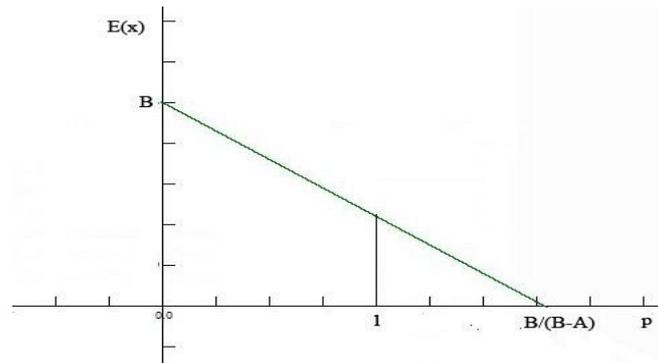


Fig. 5. The relationship curve between  $E(x)$  and  $p$ .

Postulate:  $x$  denotes access time;  $y=T$ ;  $A=TT$ ;  $B=H \cdot T_{00}+T_{21}+T_{22}+T_{23}+T_{31}+T_{32}+T_{33}$  [8].

Expectation of  $x$   $E(x) = A \cdot p + B \cdot (1-p)$ . Relationship between  $E(x)$  and  $p$  as Fig.5 (assume  $A$  and  $B$  have no changes).

Analyse this curve: We only choose part of  $0 < p < 1$ . If we want to lower  $E(x)$ , and then we should improve  $p$ . This means that the system should strengthen performance of local database. But it will result in increased  $A$ . It is thus obvious that to improve  $p$  and to reduce  $A$  is paradoxical. This model would improve capacity of local database. And in order to reduce  $A$ , we take measures of intercurrent access and disposal.

#### 4. Conclusion

Paper put forward a new model of deep web integration by sacrifice rate of precision and raises the actual maneuverability of system. That is to say, system put consuming time part on background, and returns user satisfied results within time user can stand.

#### 5. References

- [1] Ying Wang, "Research on Key Technologies of Deep Web Data Integtation," unpublished.
- [2] Pengpeng Zhao, "Research on Key Technologies of Deep Web information Integtation," unpublished.
- [3] Hexiang Xu, "Research on Deep Web Integration and Its Related Several Technologies," unpublished.
- [4] <http://www.lib.lsu.edu/instruction/deepweb/deepweb00.html>.
- [5] Wei Liu, Xiao-Feng Meng and Wei-Yi Meng, "A Survey of Deep Web Data Integration," *Chinese Journal of Computers*, Vol.30, Sept.2007, pp.1475-1489.
- [6] <http://www.chinaz.com/Webbiz/Exp/0325F5232009.html>.
- [7] <http://alexa.chinaz.com/>.
- [8] Hong-fei Liu, Improvement of query efficiency on the Deep Web data integration system, *Journal of the Hebei Academy of Sciences*, Vol.26, Jun.2009, pp.11-13.