

# Compulsory Part & Beautification

Xi Fu 517030910362 Rongfei Gu 517030910363Yaxuan Ma 517030910337Yuxia Zhan 51703091031

June 24, 2018

## 1 Compulsory part

The basic part is divided into two parts.

First, add the search boxes of paper and conference to the home page. And append the paper and conference page to result page

Second, set the page turning function.

### 1.1 Add paper and conference search

First, add the search boxes to the home page.

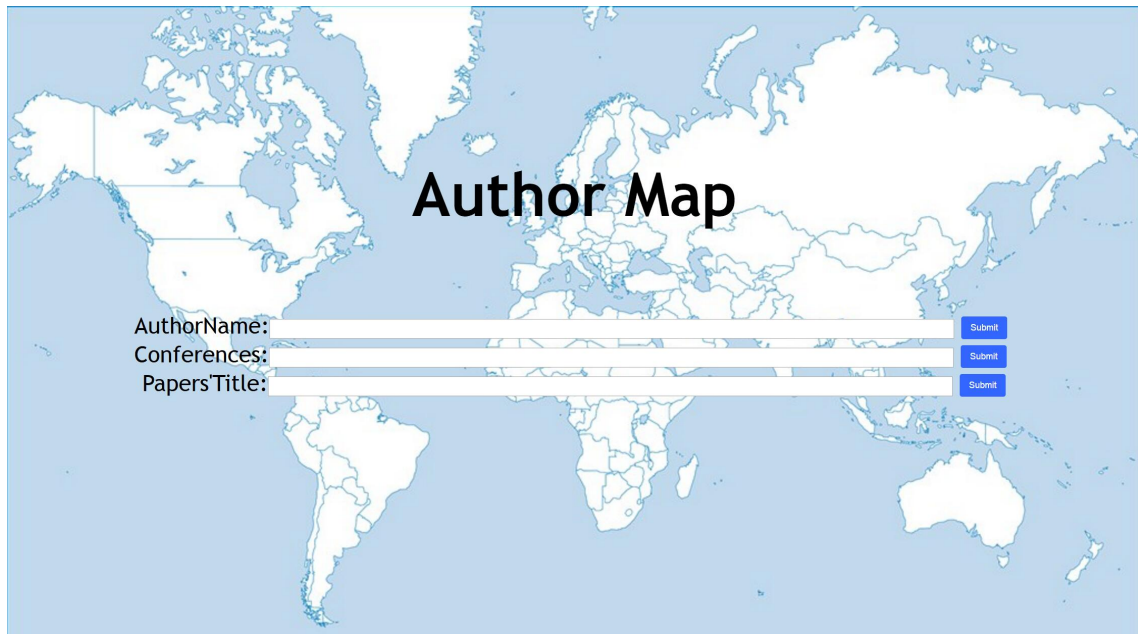


Figure 1: home.php page

```
1 |-----home4.php-----|
2 | <p align='center' style="font-size: 30px">
```

```

3 AuthorName:<input type="text" name="content1" id="content1"
  align='center' style="height:20px;width:900px;font-size: 20
  px">
4 <button type="submit" name="author" class="ui-button ui-corner
  -all ui-weight" style="background: #3366ff;color:white;
  border:none;height:30px;">Submit</button>
5 <br>
6 Conferences:<input type="text" name="content2" id="content2"
  align='center' style="height:20px;width:900px;font-size: 20
  px">
7 <button type="submit" name="conference" class="ui-button ui-
  corner-all ui-weight" style="background: #3366ff;color:white
  ;border:none;height:30px;">Submit</button>
8 <br>
9 Papers 'Title:<input type="text" name="content3" id="content3"
  align='center' style="height:20px;width:900px;font-size: 20
  px">
10 <button type="submit" name="paper" class="ui-button ui-corner-
  all ui-weight" style="background: #3366ff;color:white;border
  :none;height:30px;">Submit</button>
11 </form>

```

We can get from the codes that when we click the button named author, result page will deliver the content of 'content1' box. Similarly, when we click the buttons named conference and paper, it will deliver the contents of 'content2' and 'content3' boxes.

Therefore, the result page can show different reactions receiving different contents.

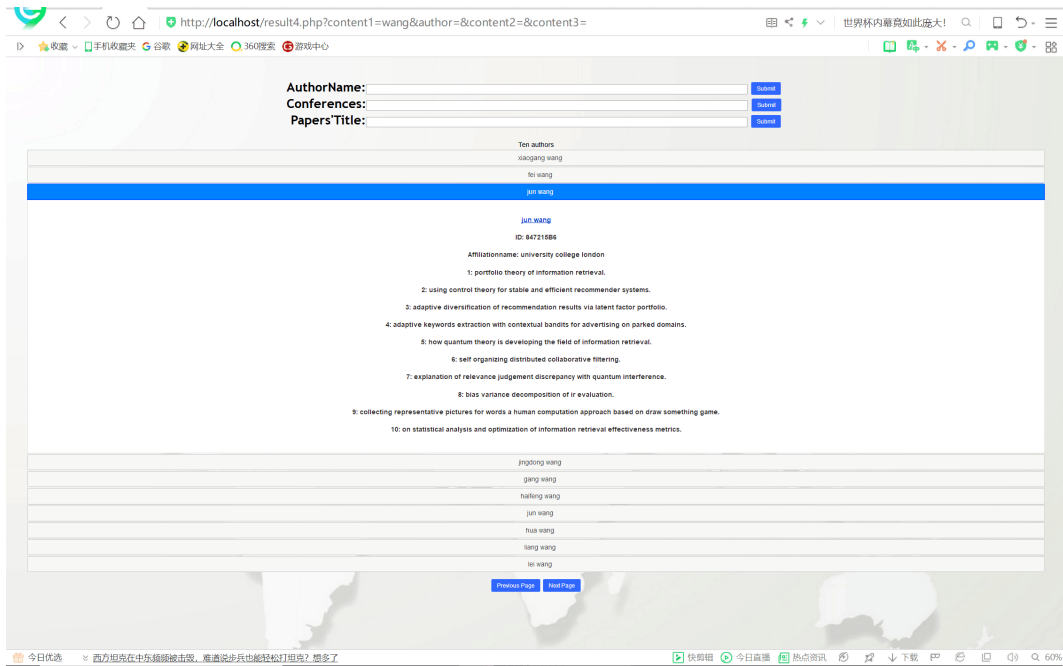


Figure 2: result.php page

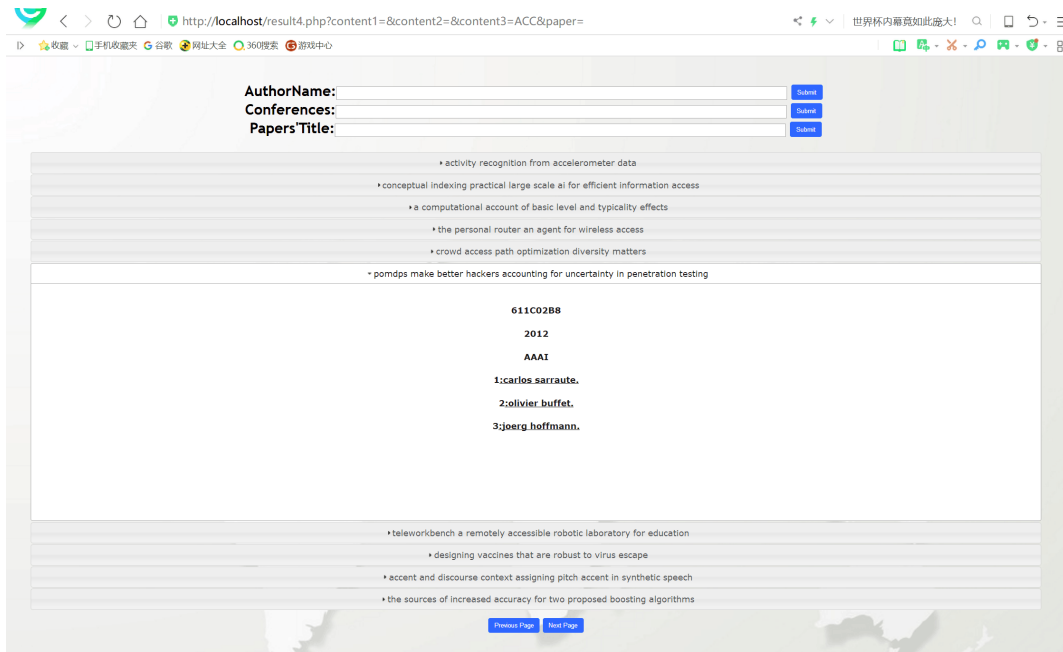


Figure 3: result.php page

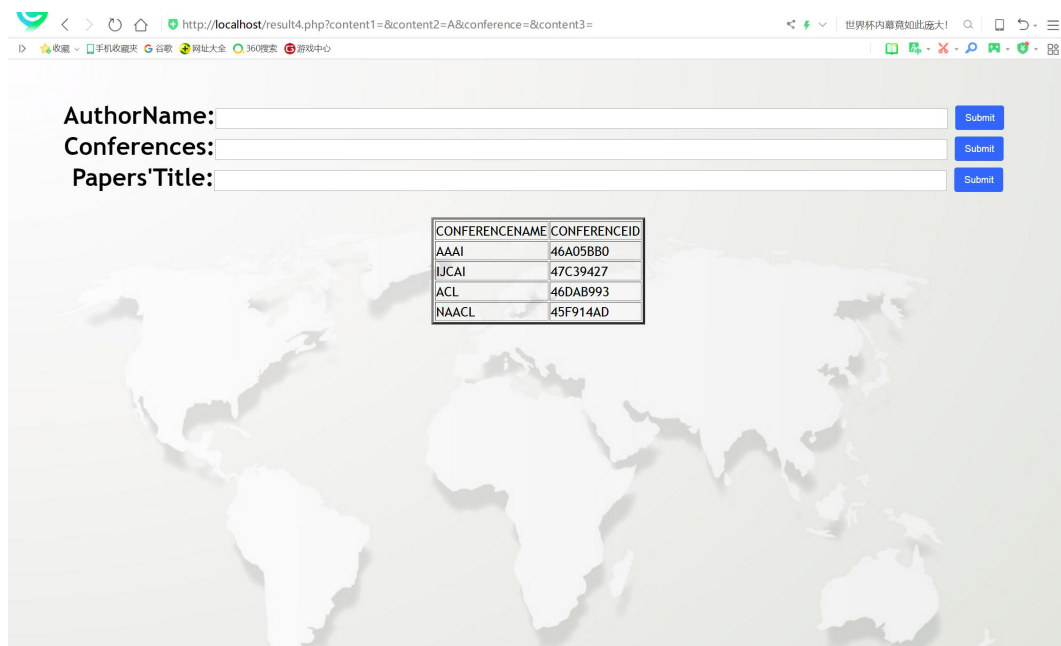


Figure 4: result.php page

```

1
2 session_start();
3 $_SESSION['aname'] = (isset($_GET['author']) ? $_GET['content1']
4   ] : null);
5 $_SESSION['PaperTitle'] = (isset($_GET['paper']) ? $_GET['
6   content3'] : null);
7 $_SESSION['ConferenceName'] = (isset($_GET['conference']) ?
8   $_GET['content2'] : null);
9
10 if (isset($_GET['author'])) {...}
11 if (isset($_GET['paper'])) {...}
12 if (isset($_GET['conference'])) {...}

```

And we also add the search of paper and conference to the result page.

```

1 -----result4.php-----
2 <p align='center' style="font-size: 30px" >
3   AuthorName:<input type="text" name="content1" id="
4     content1" align='center' style="height:20px; width
5     :900px;font-size: 20px">
6   <button type="submit" name="author" class="ui-button
7     ui-corner-all ui-weight" style="background: #3366ff;

```

```

        color : white ; border : none ; height : 30px ; ">Submit</button
    >
5   <br>
6   Conferences:<input type="text" name="content2" id="
        content2" align='center' style="height:20px; width
        :900px;font-size: 20px">
7   <button type="submit" name="conference" class="ui-
        button ui-corner-all ui-weight" style="background:
        #3366ff; color : white ; border : none ; height : 30px ; ">Submit
        </button>
8
9   <br>
10  Papers 'Title:<input type="text" name="content3" id="
        content3" align='center' style="height:20px; width
        :900px;font-size: 20px">
11 <button type="submit" name="paper" class="ui-button ui
        -corner-all ui-weight" style="background: #3366ff;
        color : white ; border : none ; height : 30px ; ">Submit</button
        >

```

## 1.2 Page turning function

At the author and paper part of result page, we added page turning button mainly supported by `getElementById()` function which returns a reference to the first object with the specified ID.

```

1  -----result4.php-----
2  $(document).ready(function(){
3      $("#previouspage").click(function(){
4          $.get("previous.php",function(data, status){
5              document.getElementById("accordion").
                innerHTML=data;
6          });
7      });
8  });
9
10 $(document).ready(function(){
11     $("#nextpage").click(function(){
12         $.get("next.php",function(data, status){
13             document.getElementById("accordion").
                innerHTML=data;
14         });

```

```

15         });
16     });
17
18     ...
19
20     <p align='center '>
21
22     <button id="previouspage" type="button" class="ui-button ui-
        corner-all ui-weight" style="background: #3366ff; color: white
        ; border: none; height: 30px; margin-left: 510px; ">Previous Page</
        button>
23 <button id="nextpage" type="button" class="ui-button ui-corner-
        all ui-weight" style="background: #3366ff; color: white; border
        : none; height: 30px; margin-right: 510px; ">Next Page</button>

```

We can see from the codes that, when we click the next or previous button, the accordion div will be replaced by the return value of next.php or previous.php.

## 2 Beautification

First I made the basic beautification including changing the background, buttons, searching box to make them more comfortable and adjusting the layout and so on.

```

1  -----home4.php-----
2  <meta charset="UTF-8">
3      <title>Author Map</title >
4      <link href="jquery-ui.css" rel="stylesheet ">
5      <style>
6      body{
7          font-family: "Trebuchet MS", sans-serif;
8          margin: 50px;
9          background-size:100%;
10     }
11     .demoHeaders {
12         margin-top: 3em;
13     }
14     #dialog-link {
15         padding: .4em 1em .4em 20px;
16         text-decoration: none;
17         position: relative;
18     }
19     #dialog-link span.ui-icon {
20         margin: 0 5px 0 0;
21         position: absolute;

```

```

22         left: .2em;
23         top: 50%;
24         margin-top: -8px;
25     }
26     #icons {
27         margin: 0;
28         padding: 0;
29     }
30     #icons li {
31         margin: 2px;
32         position: relative;
33         padding: 4px 0;
34         cursor: pointer;
35         float: left;
36         list-style: none;
37     }
38     #icons span.ui-icon {
39         float: left;
40         margin: 0 4px;
41     }
42     .fakewindowcontain .ui-widget-overlay {
43         position: absolute;
44     }
45     select {
46         width: 200px;
47     }
48 </style>
49 </head>
50 <body background="123.jpg"></body>
51 <br/><br/><br/><br/><br/>
52
53 -----error.html-----
54 <style>
55     body{
56         font-family: "Trebuchet MS", sans-serif;
57         margin: 50px;
58     }
59     .demoHeaders {
60         margin-top: 3em;
61     }
62     #dialog-link {
63         padding: .4em 1em .4em 20px;
64         text-decoration: none;

```

```

65         position: relative;
66     }
67     #dialog-link span.ui-icon {
68         margin: 0 5px 0 0;
69         position: absolute;
70         left: .2em;
71         top: 50%;
72         margin-top: -8px;
73     }
74     #icons {
75         margin: 0;
76         padding: 0;
77     }
78     #icons li {
79         margin: 2px;
80         position: relative;
81         padding: 4px 0;
82         cursor: pointer;
83         float: left;
84         list-style: none;
85     }
86     #icons span.ui-icon {
87         float: left;
88         margin: 0 4px;
89     }
90     .fakewindowcontain .ui-widget-overlay {
91         position: absolute;
92     }
93     select {
94         width: 200px;
95     }
96 </style>

```

## 2.1 Folder

When dealing with the selected ten papers and ten authors, at first I chose the table to present them, just like the following.



TITLE	ID	YEAR	CONFERENCE
a universal generalization for temporal difference learning using haar basis functions	0026CD26	2000	ICML
aaai 13 preface	0087F633	2013	AAAI
bayesian network based reparameterization of haar like feature	00913EBB	2006	AAAI
gaze estimation using regression analysis and aams parameters selected based on information criterion	0113E85E	2010	ICCV
the virtual aachen project	013E7F86	2008	CVPR
gsaam graph sets and associative memories	01BCB983	1999	ICML
physics problem solving isaac ii	01F9A19E	1981	IJCAI
geometrical pattern feature extraction by projection on haar orthonormal basis	022CF70B	1969	IJCAI
eaai 13 preface	050F7850	2013	AAAI
winning the aaai robot competition	06726082	1993	AAAI

Figure 5: previous

Authorname: aa

## Ten authors

NAME	ID	TIMES	AffiliationID
<a href="#">maarten de rijke</a>	17F37D50	108	university of amsterdam
<a href="#">tinne tuytelaars</a>	752CFFDC	57	
<a href="#">wolfgang maass</a>	7991E8E1	37	
<a href="#">anders sogaard</a>	7D34DA6A	33	
<a href="#">masaaki nagata</a>	7D510BC5	32	
<a href="#">kai salomaa</a>	8024331F	31	queen s university
<a href="#">jaap kamps</a>	7C82C519	30	university of amsterdam
<a href="#">tommi s jaakkola</a>	1447DD1D	28	
<a href="#">steffen staab</a>	73B7739A	26	
<a href="#">isaac cohen</a>	7F77328C	26	

Previous Page

Next Page

Figure 6: previous

But for the convenience and pleasing to the eye, I added the folding function. The user can scan the ten clauses swiftly. If he wants to know a specific paper or author more thoroughly, he can click and get more information. He can even click the authors'

name to know more about him or her.

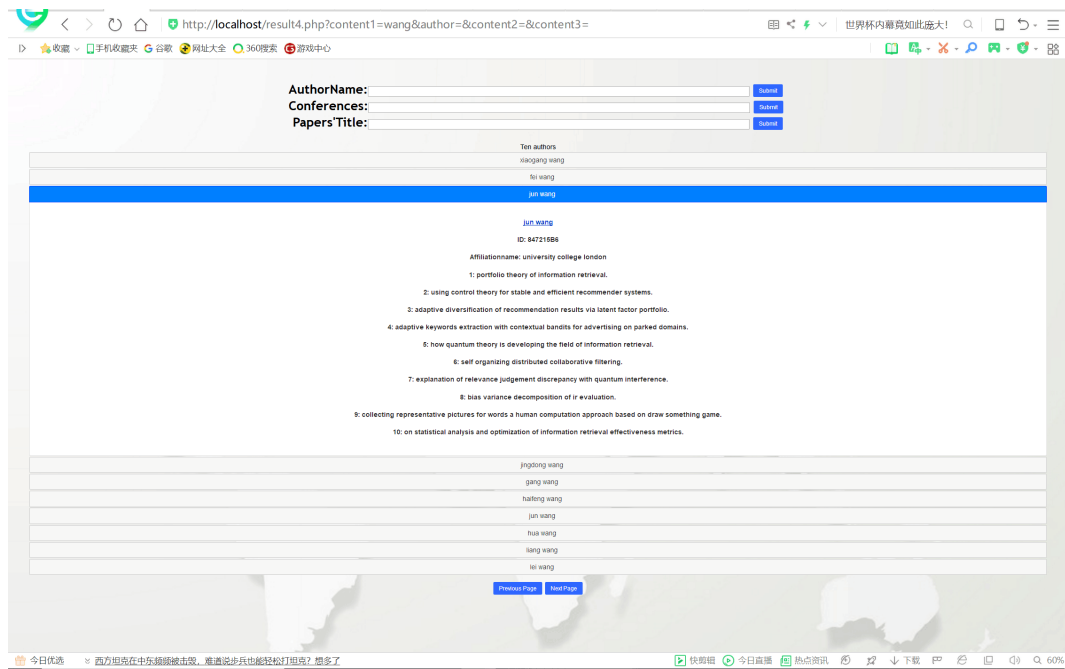


Figure 7: folder of author

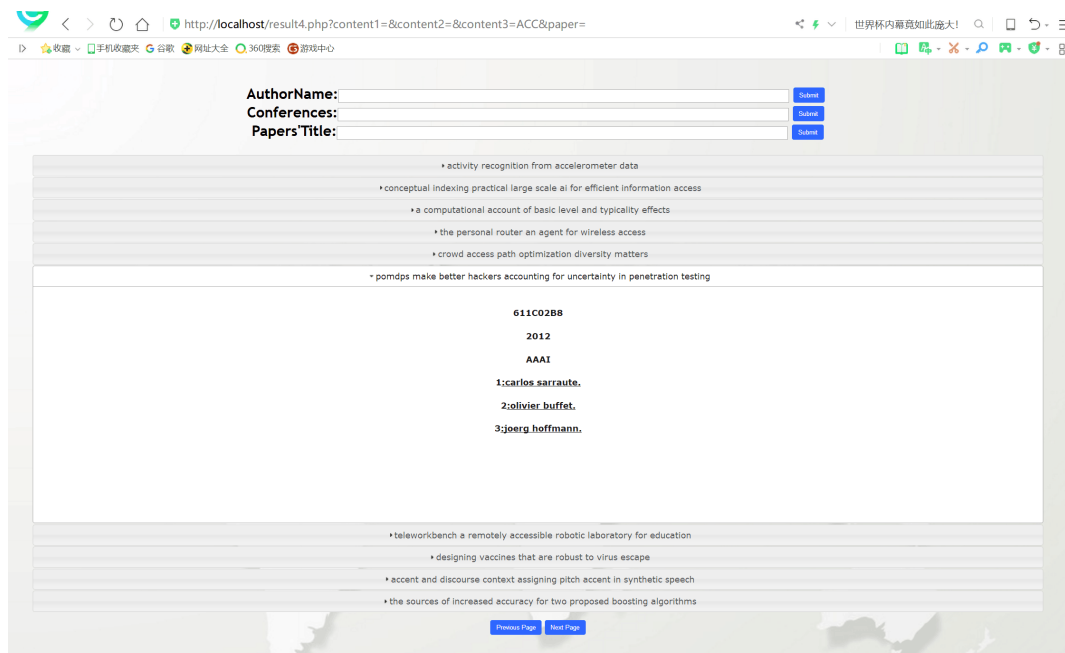


Figure 8: folder of paper

If the user click the author's name, it will turn to the author's own page. At page authors, on the basis of the ten papers and author id, we added the author's supposed teachers and students.

when we made the presentation, it took a long time for the folders to show. I must explain that it is because of the Internet. When we debugged ot, the speed is much faster. You can see it from the video in the package.

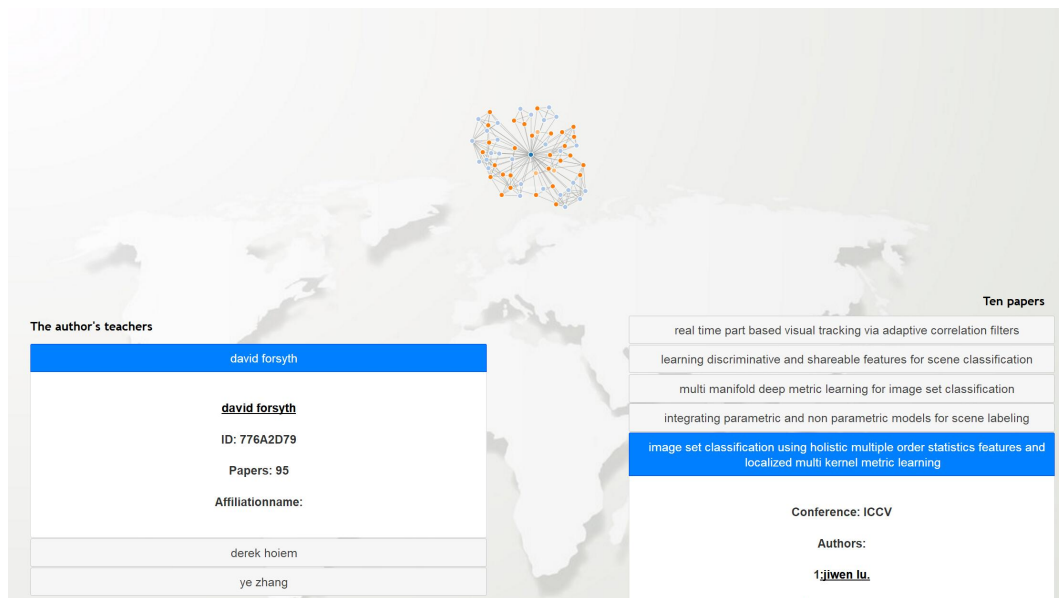


Figure 9: folder of author's teachers and students

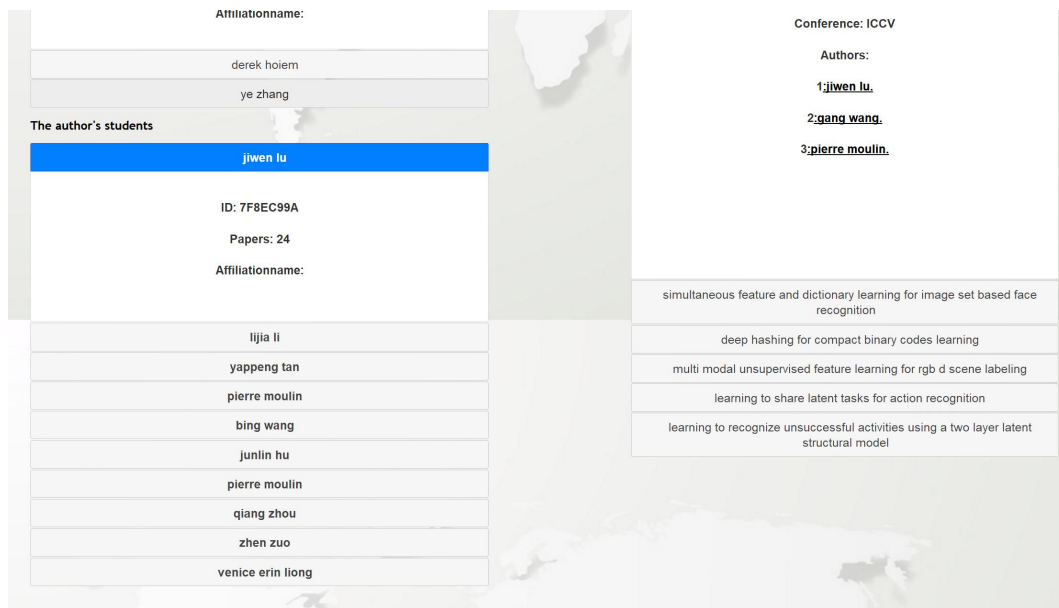


Figure 10: folder of author's teachers and students

The main idea of building the folder is about using 'accordion'.The following is the core codes.

```

1 $totala=0;
2 echo '<div id="accordion" style="width:100%" align="center">';
3 while($row=mysqli_fetch_array($retval0 , MYSQL_ASSOC))
4 {
5     $totala=$totala+1;
6 }
7 $_SESSION[ 'total ']=$totala ;
8
9 $sql = "select AuthorID,count(*) from paper_author_affiliation
10       where AuthorID in
11         (select AUTHORID from authors
12          where AUTHORNAME like '%$ANAME%')
13         group by AuthorID
14         order by count(*) desc
15         limit 10";
16 mysqli_select_db( $conn , 'main_db1' );
17 $retval = mysqli_query( $conn , $sql );
18
19
20 echo 'Ten authors';
21 $row = mysqli_fetch_array($retval , MYSQL_ASSOC);
22 if (empty($row))

```

```

23     {
24         echo "<script language=\"javascript\">";
25         echo "document.location=\"error.html\"";
26         echo "</script>";
27     }
28 while($row)
29 {
30
31     $ID=$row[ 'AuthorID' ];
32     $sql3="select AUTHORNAME from authors
33           where AUTHORID='$ID'";
34     $retval3 = mysqli_query( $conn, $sql3 );
35
36     $row3=mysqli_fetch_array( $retval3 , MYSQL_ASSOC);
37     $NAME=$row3[ 'AUTHORNAME' ];
38
39     $sql2="select affiliationname from affiliations
40           where AFFILIATIONID =
41           (select AffiliationID from
42            paper_author_affiliation
43            where AuthorID='$ID'
44             group by AffiliationID
45             order by count(*) desc
46             limit 1)";
47
48     $retval2 = mysqli_query( $conn, $sql2 );
49
50     $row2=mysqli_fetch_array( $retval2 , MYSQL_ASSOC);
51     echo '<h3>';
52     echo "<a href='author4.php?Name=$NAME&Id=$ID' >". $NAME.
53         "</a>";
54     echo '</h3>';
55     echo "<div id='wrap'>";
56     echo "<h4><a href='author4.php?Name=$NAME&Id=$ID' >".
57         $NAME. "</a></h4>";
58     echo '<h4 class="thick" id="ID">ID: '. $ID. '</h4>';
59
60     echo '<h4 class="thick" id="Affiliationname">
61         Affiliationname: '. $row2[ 'affiliationname' ]. '</h4>';
62
63     $sql4="select title from papers
64           where paperid in
65           (select paperid from
66            (select paperid from

```

```

62         paper_author_affiliation
63         where authorid='$ID'
64         group by paperid
65         order by count(*) desc
66         limit 11)
67         as t)";
68 $retval4 = mysqli_query( $conn, $sql4 );
69 if(! $retval4 )
70 {
71     die( ' : ' . mysqli_error($conn));
72 }
73 $num=1;
74 $row4=mysqli_fetch_array( $retval4 , MYSQL_ASSOC);
75 while($row4 = mysqli_fetch_array($retval4 , MYSQL_ASSOC)
76 )
77 {
78     echo '<h4 class="thick">';
79     echo $num;
80     $num=$num+1;
81     $Atitle=$row4[ 'title '];
82     echo ": ".$Atitle." ";
83
84
85 }
86 echo '</h4>';
87 echo '</div>';
88 $row = mysqli_fetch_array( $retval , MYSQL_ASSOC);
89 }
90 echo '</div>';
91 mysqli_close( $conn);
92
93 ?>
94 <script >
95 $( "#accordion" ).accordion();
96 </script >

```

## 2.2 Error page

When the database cannot find any relative information about what the user input, it will turn to an error page. If the user doesn't click the 'return' button, it will return to the home page automatically.

Sorry! Can't find the corresponding information!

You will be transmitted to the HOME page in 3 seconds!

If not, please click [here](#).

Figure 11: error page

```
1 -----home4.php-----
2 if (empty($row))
3     {
4         echo "<script language=\"javascript\">";
5         echo "document.location=\"error.html\"";
6         echo "</script>";
7     }
8 -----error.html-----
9 <body>
10
11
12     <h1 align="center">Sorry! Can't find the corresponding
13         information!</h1>
14     <h2 align="center">You will be transmitted to the HOME
15         page in <span id="totalSecond">5</span> seconds!</h2
16         >
17     <h3 align='center '>If not, please click <a href="home4.
18         php">here</a>.</h3>
19     <script language="javascript" type="text/javascript">
20         var second = totalSecond.innerText;
21         setInterval("redirect()", 1000);
22         function redirect(){
23             if(second==1) location.href='home4.php'
24             ;
25     </script>
```

```

20         totalSecond.innerHTML+=second;
21     }
22 </script>
23
24 </body>

```

### 3 Teacher & Student Relationship

In page author, we add a new part to the original page, that is the author's teachers and the author's students. To realize this, we need a new table in the database, containing all pairs of authors who have teacher-student relationship. In the former experiment, we've already extracted a model, judging whether two authors are teacher and student by nine features. So to create the table, we should extract features of each two authors who have cooperated before, and judge their relationship through the model. Finally, we add two tables in page author, showing the author's teachers and students separately and make the corresponding nodes in the visualized graph into different colors.

#### 3.1 A subsection

First, we create a txt file to contain the teacher-student relationships through python. Import the third-party packages first.

```

1 from sklearn.externals import joblib
2 import numpy as np
3 import pymysql
4 import time

```

Connect the python program to the database.

```

1 clf = joblib.load("logistic_regression.model")
2
3 time_begin = time.time()
4 connection = pymysql.connect(host="localhost",
5                             user='root',
6                             password="",
7                             db='main_grf',
8                             port=3306,
9                             cursorclass=pymysql.cursors.Cursor)
10
11 cursor = connection.cursor()

```

Create the function.

The nine features are:

- 1.The number of papers A published before the first cooperation between A and B.
- 2.The number of papers A published before the first cooperation between A and B.



- 3.(Feature 1 - Feature 2)/ The number of papers A and B cooperated.
  - 4.The difference between the years that A first published a paper and A,B's first cooperation.
  - 5.The difference between the years that B first published a paper and A,B's first cooperation.
  - 6.(Feature 4 - Feature 5)/ The number of years A,B cooperated.
  - 7.The number of papers A published without B during their cooperation.
  - 8.The number of papers B published without A during their cooperation.
  - 9.(Feature 7 - Feature 8)/ The number of papers A and B cooperated.
- Actually there is no need to extract the nine features one by one, since some key values were reused in extracting these features.

The key values are:

- 1.The first year that AB cooperated.
- 2.The last year that AB cooperated.
- 3.The total cooperation year of AB.
- 4.The number of papers AB wrote together.
- 5.The number of papers A wrote before the first year that AB cooperated.
- 6.The number of papers A wrote while AB cooperated.
- 7.The number of papers B wrote before the first year that AB cooperated.
- 8.The number of papers B wrote while AB cooperated.
- 9.The year A published the first paper.
- 10.The year B published the first paper.

```

1 def feature(author_a, author_b):
2     # first_co_year: the first year that AB cooperated
3     # last_co_year: the last year that AB cooperated
4     # ab_co_year: the total cooperation year of AB
5     # ab_co_paper: the num of papers AB wrote together
6     sql_1_1 = "select PaperID from paauf where AuthorID=%s and PaperID in (
                                select PaperID from paauf where
                                AuthorID=%s)"
7
8     cursor.execute(sql_1_1, (author_a, author_b,))
9     results_1_1 = cursor.fetchall()
10    first_co_year = 3000
11    last_co_year = 1000
12    ab_co_paper = 0
13    for row_1_1 in results_1_1:
14        ab_co_paper += 1
15        paperid = row_1_1[0]
16        sql_1_2 = "select PAPERPUBLISHYEAR from papers where PAPERID=%s"
17        cursor.execute(sql_1_2, (paperid,))
18        results_1_2 = cursor.fetchone()
19        if results_1_2:
20            year = results_1_2[0]
21            if first_co_year > year:
22                first_co_year = year
23            if last_co_year < year:
                last_co_year = year

```

```
24 | ab_co_year = last_co_year - first_co_year + 1
```

In this part, key values 1 to 4 were fetched, where sql-1-1 picked out the papers cooperated by A and B, sql-1-2 returned the corresponding year when the papers were published. Through traversal, get the smallest year number and the biggest year number, that is the first year and the last year of cooperation, then calculate the total cooperation year.

```
1 | # a_bef_paper: the num of papers A wrote before first_co_year
2 | # a_du_paper: the num of papers A wrote while AB cooperated
3 | a_bef_paper = 0
4 | a_du_paper = 0
5 | sql_2_1 = "select PaperID from paauf where AuthorID=%s"
6 | cursor.execute(sql_2_1, (author_a,))
7 | results_2_1 = cursor.fetchall()
8 | for row_2_1 in results_2_1:
9 |     paperid = row_2_1[0]
10 |     sql_2_2 = "select PAPERPUBLISHYEAR from papers where PAPERID=%s"
11 |     cursor.execute(sql_2_2, (paperid,))
12 |     results_2_2 = cursor.fetchone()
13 |     if results_2_2:
14 |         if results_2_2[0] < first_co_year:
15 |             a_bef_paper += 1
16 |         elif first_co_year <= results_2_2[0] <= last_co_year:
17 |             a_du_paper += 1
```

In this part, key values 5,6 were fetched, where sql-2-1 selected the papers that A wrote, sql-2-2 selected the corresponding years, then pick out the papers written before their first cooperation and during their cooperation separately. Key values 7,8 were realized through similar program where the only difference is to change A to B.

```
1 | # a_al_year: the year A published the first paper
2 | # b_al_year: the year B published the first paper
3 | sql_4_1 = "select PAPERPUBLISHYEAR from papers where PAPERID in (select
4 |                                     PaperID from paauf where AuthorID
5 |                                     =%s) order by PAPERPUBLISHYEAR
6 |                                     limit 1"
7 |
8 | cursor.execute(sql_4_1, (author_a,))
9 | results_4_1 = cursor.fetchone()
10 | a_al_year = results_4_1[0]
11 |
12 | sql_4_2 = "select PAPERPUBLISHYEAR from papers where PAPERID in (select
13 |                                     PaperID from paauf where AuthorID
14 |                                     =%s) order by PAPERPUBLISHYEAR
15 |                                     limit 1"
16 |
17 | cursor.execute(sql_4_2, (author_b,))
18 | results_4_2 = cursor.fetchone()
19 | b_al_year = results_4_2[0]
```

In this part, key values 9,10 were fetched, where sql-4-1 gave the first year A published a paper and sql-4-2 gave B's.

```
1 fea_one = a_bef_paper
2 fea_two = b_bef_paper
3 fea_three = (fea_one - fea_two) / ab_co_paper
4 fea_four = first_co_year - a_al_year
5 fea_five = first_co_year - b_al_year
6 fea_six = (b_al_year - a_al_year) / ab_co_year
7 fea_seven = a_du_paper - ab_co_paper
8 fea_eight = b_du_paper - ab_co_paper
9 fea_nine = (fea_seven - fea_eight) / ab_co_paper
10 return [fea_one, fea_two, fea_three, fea_four, fea_five, fea_six,
          fea_seven, fea_eight, fea_nine]
```

At last, calculate the nine features from the ten key values and return them in a list.

After the function is created, we select the pairs of authors who have cooperated together from the database:

```
1 sql_0 = "select AUTHORID from authors"
2 cursor.execute(sql_0)
3 results_0 = cursor.fetchall()
4 text = open("test_rela.txt", "w")
5 #open the target file
```

Lastly, judge whether each pair of authors are student and teacher through the model we've got in experiment three(here we use model LogisticRegression), if they are, add the author's id to the txt file, and load the txt file into the database so as to create the table we need.

```
1 for row_0 in results_0:
2     sql_0_1 = "select AuthorID from paauf where PaperID in (select PaperID
3                                                         from paauf where AuthorID=%s)
4                                                         group by AuthorID"
5
6     cursor.execute(sql_0_1,(row_0[0],))
7     results_0_1 = cursor.fetchall()
8     for row_0_1 in results_0_1:
9         if row_0[0] != row_0_1[0]:
10            Author_A = row_0[0]
11            Author_B = row_0_1[0]
12            feat = feature(Author_A, Author_B)
13            feat_list = []
14            for i in range(9):
15                feat_list.append(float(feat[i]))
16            feat_array = np.array(feat_list)
17            a = clf.predict([feat_array])
18            if a == 1:
19                text.write(Author_A + "\t" + Author_B + "\n")
20
21 text.close()
22 connection.close()
```

After we already have the teacher-student relationship table, we can apply it to our website.

First create the two tables, containing the author's students and teachers.

```
1 $sql_5_1="select AuthorID, count(*) from
   paper_author_affiliation where AuthorID in
2         (select TeacherID from testre where StudentID
           = '$AUTHORID' )
3         group by AuthorID
4         order by count(*) desc
5         limit 10";
6 mysqli_select_db( $conn, 'main_db1' );
7 $retval = mysqli_query( $conn, $sql_5_1 );
```

Select id of the author's teachers.

Then, find their authorid, number of papers and their affiliation. Exhibit all these information.

```
1 echo "<h2 class='thick'>The author's teachers</h2>";
2 echo '<h2 class="ten_au">Ten papers</h2>';
3 echo '<div id="accordion2" class="divcss5-left" align="center">
   ' ;
4 while($row = mysqli_fetch_array($retval, MYSQL_ASSOC))
5 {
6     $ID_1=$row[ 'AuthorID' ];
7     $sql3_1="select AUTHORNAME from authors
8             where AUTHORID='$ID_1'";
9     $retval3_1 = mysqli_query( $conn, $sql3_1 );
10
11     $row3=mysqli_fetch_array($retval3_1, MYSQL_ASSOC);
12     $NAME=$row3[ 'AUTHORNAME' ];
13
14     $sql2="select AFFILIATIONNAME from affiliations
15           where AFFILIATIONID =
16           (select AffiliationID from
17             paper_author_affiliation
18             where AuthorID='$ID_1'
19             group by AffiliationID
20             order by count(*) desc
21             limit 1)";
22     $retval2 = mysqli_query( $conn, $sql2 );
23     $row2=mysqli_fetch_array($retval2, MYSQL_ASSOC);
```

```

24 echo '<h3 class="thick">'. $NAME. '</h3>';
25 echo "<div>";
26 echo '<h4 class="thick">';
27 echo "<a href='author4.php?Name=$NAME&Id=$ID_1' >".
    $NAME. "</a>";
28 echo '</h4>';
29
30 echo '<h4 class="thick" id="ID">ID: '. $ID_1. '</h4>';
31 echo '<h4 class="thick" id="ID1">Papers: '. $row[ 'count
    (*) ']. '</h4>';
32 echo '<h4 class="thick" id="Affiliationname">
    Affiliationname: '. $row2[ 'AFFILIATIONNAME' ]. '</h4>';
33 echo '</div>';
34 }
35 echo '</div>';

```

Similarly, create the table of the author's students.

```

1 $sql_5_1="select AuthorID, count(*) from
    paper_author_affiliation where AuthorID in
2         (select StudentID from testre where TeacherID
           = '$AUTHORID ')
3         group by AuthorID
4         order by count(*) desc
5         limit 10";
6 mysqli_select_db( $conn, 'main_db1' );
7 $retval = mysqli_query( $conn, $sql_5_1 );
8
9 echo "<h2>The author's students</h2>";
10 echo '<div id="accordion1" class="divcss5-left" align="center">
    ';
11 while($row = mysqli_fetch_array($retval, MYSQL_ASSOC))
12 {
13     $ID_1=$row[ 'AuthorID' ];
14     $sql3_1="select AUTHORNAME from authors
15             where AUTHORID='$ID_1'";
16     $retval3_1 = mysqli_query( $conn, $sql3_1 );
17
18     $row3=mysqli_fetch_array($retval3_1, MYSQL_ASSOC);
19     $NAME=$row3[ 'AUTHORNAME' ];
20
21     $sql2="select AFFILIATIONNAME from affiliations
22           where AFFILIATIONID =

```

```

23         (select AffiliationID from
24             paper_author_affiliation
25             where AuthorID='$ID_1'
26             group by AffiliationID
27             order by count(*) desc
28             limit 1)";
29     $retval2 = mysqli_query( $conn , $sql2 );
30     $row2=mysqli_fetch_array( $retval2 , MYSQL_ASSOC );
31     echo '<h3 class="thick">'. $NAME. '</h3>';
32     echo '<h4 class="thick">';
33     echo "<a href='author4.php?Name=$NAME&Id=$ID_1' >".
34         $NAME. "</a>";
35     echo '</h4>';
36     echo "<div>";
37     echo '<h4 class="thick" id="ID">ID: '. $ID_1. '</h4>';
38     echo '<h4 class="thick" id="ID1">Papers: '. $row[ 'count
39         (*) ']. '</h4>';
40     echo '<h4 class="thick" id="Affiliationname">
41         Affiliationname: '. $row2[ 'AFFILIATIONNAME' ]. '</h4>';
42     echo '</div>';
43 }
44 echo '</div>';

```

Then, create the visualized graph of the author's cooperation relationship:

This part of exercise can be divided into the following steps:

1. Write the JavaScript program that can transfer a json file into a visible graph.
2. Extract the authors' information and their cooperative relationship from the database.
3. Write the information of authors and relationship into a .json file in the form of json.

Here is the JavaScript program, from the given website, transforming a json file into a visible graph:

```

1 <style>
2
3 .links line {
4     stroke: #999;
5     stroke-opacity: 0.6;
6 }
7
8 .nodes circle {

```

```

9     stroke: #fff;
10    stroke-width: 1.5px;
11  }
12
13 </style>
14 <svg width="960" height="600"></svg>
15 <script src="https://d3js.org/d3.v4.min.js"></script>
16 <script>
17
18 var svg = d3.select("svg"),
19     width = +svg.attr("width"),
20     height = +svg.attr("height");
21
22 var color = d3.scaleOrdinal(d3.schemeCategory20);
23
24 var simulation = d3.forceSimulation()
25     .force("link", d3.forceLink().id(function(d) { return d.id;
26         }))
27     .force("charge", d3.forceManyBody())
28     .force("center", d3.forceCenter(width / 2, height / 2));
29
30 d3.json("test.json", function(error, graph) {
31     if (error) throw error;
32
33     var link = svg.append("g")
34         .attr("class", "links")
35         .selectAll("line")
36         .data(graph.links)
37         .enter().append("line")
38         .attr("stroke-width", function(d) { return Math.sqrt(d.
39             value); });
40
41     var node = svg.append("g")
42         .attr("class", "nodes")
43         .selectAll("circle")
44         .data(graph.nodes)
45         .enter().append("circle")
46         .attr("r", 5)
47         .attr("fill", function(d) { return color(d.group); })
48         .call(d3.drag()
49             .on("start", dragstarted)
50             .on("drag", dragged)
51             .on("end", dragended));

```

```

50
51 node.append("title")
52   .text(function(d) { return d.id; });
53
54 simulation
55   .nodes(graph.nodes)
56   .on("tick", ticked);
57
58 simulation.force("link")
59   .links(graph.links);
60
61 function ticked() {
62   link
63     .attr("x1",function(d){return d.source.x;})
64     .attr("y1",function(d){return d.source.y;})
65     .attr("x2",function(d){return d.target.x;})
66     .attr("y2",function(d){return d.target.y;});
67
68   node
69     .attr("cx", function(d) { return d.x; })
70     .attr("cy", function(d) { return d.y; });
71 }
72 });
73
74 function dragstarted(d) {
75   if (!d3.event.active) simulation.alphaTarget(0.3).restart();
76   d.fx = d.x;
77   d.fy = d.y;
78 }
79
80 function dragged(d) {
81   d.fx = d3.event.x;
82   d.fy = d3.event.y;
83 }
84
85 function dragended(d) {
86   if (!d3.event.active) simulation.alphaTarget(0);
87   d.fx = null;
88   d.fy = null;
89 }
90
91 </script>
92 <style>

```



```

93 a:link {color: rgb(200,200,200)}
94 a:visited {color: rgb(180,180,180)}
95 a:hover {color: rgb(160,160,160)}
96 a:active {color: rgb(140,140,140)}
97 </style>

```

Then, extract the informations needed in author.php.

```

1 $arr_all=array();
2 $arr_node=array();
3 $arr_link=array();
4 $arr_id=array();

```

First, create empty arrays for storing the informations to get. In the later steps, the exact function of the arrays will be explained.

```

1 $dbhost = 'localhost:3306';
2 $dbuser = 'root';
3 $dbpass = '';
4 $conn = mysqli_connect($dbhost, $dbuser, $dbpass);
5
6 mysqli_query($conn, "set names utf8");

```

Build link to MySQL.

Create two functions, judging whether two authors have teacher-student relationship.

```

1 function if_t($AUI,$AUI_1){
2     $dbhost = 'localhost:3306';
3     $dbuser = 'root';
4     $dbpass = '';
5     $conn = mysqli_connect($dbhost, $dbuser, $dbpass);
6     $sql_t="select TeacherID from testre where StudentID='
7         $AUI' and TeacherID='$AUI_1'";
8
9     mysqli_select_db($conn, 'main_db1');
10    $retval_t=mysqli_query($conn, $sql_t);
11    $row_t=mysqli_fetch_array($retval_t, MYSQL_ASSOC);
12    if($row_t) return 1;
13    else return 0;
14 }
15 function if_s($AUI,$AUI_1){
16     $dbhost = 'localhost:3306';

```

```

17     $dbuser = 'root';
18     $dbpass = '';
19     $conn = mysqli_connect($dbhost, $dbuser, $dbpass);
20     $sql_s="select StudentID from testre where TeacherID='
           $AUI' and StudentID='$AUI_1'";
21
22     mysqli_select_db($conn, 'main_db1');
23     $retval_s=mysqli_query($conn, $sql_s);
24     $row_s=mysqli_fetch_array($retval_s, MYSQL_ASSOC);
25     if($row_s) return 1;
26     else return 0;
27 }

```

```

1 $AUTHORID=$_GET["Id"];
2 $sql_0="select AUTHORID from authors where AUTHORID in
3       (select AuthorID from paauf
4       where PaperID in(select PaperID from paauf
5       where AuthorID='$AUTHORID'))";
6 mysqli_select_db($conn, 'main_grf');
7 $retval_0 = mysqli_query($conn, $sql_0);
8
9 while($row_0 = mysqli_fetch_array($retval_0, MYSQL_ASSOC))
10 {
11     $AID=$row_0['AUTHORID'];
12     $sql_01="select AUTHORNAME from authors where AUTHORID
13            ='$AID'";
14     $retval_01=mysqli_query($conn, $sql_01);
15     $row_01=mysqli_fetch_array($retval_01, MYSQL_ASSOC);
16     $AN=$row_01['AUTHORNAME'];
17     if($AID==$AUTHORID){
18         $arr_tmp=array("id"=>$AN, "id"=>$AI, "group"=>1);
19     }
20     else if (if_t($AID,$AUTHORID)==1){
21         $arr_tmp=array("id"=>$AN, "id"=>$AI, "group"=>2);
22     }
23     else if (if_s($AID,$AUTHORID)==1){
24         $arr_tmp=array("id"=>$AN, "id"=>$AI, "group"=>3);
25     }
26     else {
27         $arr_tmp=array("id"=>$AN, "id"=>$AI, "group"=>4);
28     }
29     array_push($arr_node, $arr_tmp);
30     array_push($arr_id, $AID);

```

30 }

Select all the authors who have cooperated with the target author, including the target author. As we've got the ids of them, we can easily find the corresponding names. Create an array for each author, the array should include label 'id' pointing to the authors' name and label 'group' to classifier the authors so as to give different colors to the authors. In this exercise, the target author is divided into group one, two, three and four, symbolizing the author, the author's teachers, the author's students and other authors. Then, push the array of the author into the array of nodes, and the ids into array id.

```
1 foreach($arr_id as $ids){
2     array_shift($arr_id);
3     foreach($arr_id as $ids_2){
4         $dbhost = 'localhost:3306';
5         $dbuser = 'root';
6         $dbpass = '';
7         $conn = mysqli_connect($dbhost, $dbuser,
8                                 $dbpass);
9
10        mysqli_query($conn, "set names utf8");
11
12        $sql_11="select PAPERID from paauaf
13                where AuthorID='$ids'
14                and PAPERID in
15                (select PAPERID from
16                 paauaf where AuthorID='$ids_2
17                 ')" ;
18
19        mysqli_select_db($conn, 'main_grf');
20        $retval_11 = mysqli_query($conn, $sql_11);
21
22        $row_11=mysqli_fetch_array($retval_11,
23                                   MYSQL_ASSOC);
24        if($row_11)
25        {
26            $sql_12="select AUTHORNAME from authors
27                    where AUTHORID='$ids'"
28                    ;
29            mysqli_select_db($conn, 'main_grf');
30            $retval_12 = mysqli_query($conn,
31                                    $sql_12);
32            $row_12=mysqli_fetch_array($retval_12,
33                                       MYSQL_ASSOC);
34            $NAME1=$row_12['AUTHORNAME'];
```

```

28
29         $sql_13="select AUTHORNAME from authors
30                 where AUTHORID='$ids_2
31                 '";
32         mysqli_select_db($conn, 'main_grf');
33         $retval_13 = mysqli_query( $conn ,
34                 $sql_13);
35         $row_13=mysqli_fetch_array($retval_13 ,
36                 MYSQL_ASSOC);
37         $NAME2=$row_13[ 'AUTHORNAME' ];
38         $arr_tmp1=
39         array( "source"=>$NAME1, "target"=>$NAME2
40               , "value"=>1);
41         array_push($arr_link , $arr_tmp1);
42     }
43 }

```

Now, the ids of the authors are in the array: `arr_id`. To extract the cooperative relationship between each two of them, we need to pick out each two of them. This is accomplished through 'foreach' loop of array. For each item in the array, this item is removed from the array once it is pointed to, and for all the remaining items, compare one by one with the pointed item. If there exist a cooperative relationship between the two authors, create an array of label 'source' pointing to one author, label 'target' pointing to the other and label 'value' to the kind of line.

Then, collect the array of nodes and the array of links into one array, transform it into json code and write it into a .json file.

```

1 $arr_all=array( "nodes"=>$arr_node , "links"=>$arr_link );
2 $jfile=fopen( "test.json" ,"w") or die( "unable to open file." );
3 $json_all=json_encode( $arr_all );
4 fwrite( $jfile , $json_all );
5 fclose( $jfile );

```

## 4 Solr

### 4.1 Problem Description

Use solr or elastic search instead of mysql to search for things and transfer the database in mysql to solr or elastic search to boost efficiency.

## 4.2 Analysis and Solution

First, we should which one to use, solr or elasticsearch, which both have their merits and demerits.

- Solr supports more types of data, like csv,php,etc; while elasticsearch only supports json.
- Solr has more inner functions; while elasticsearch prefers plug-ins.
- Solr performs better in traditional searching; while elasticsearch is a better choice for real time search.

Since in this program, there is not too much nessacity for real time search in that all the information are from the database given, we can choose solr. And it is proved that solr is quicker in this type of searching and provides various daata types to operate on.

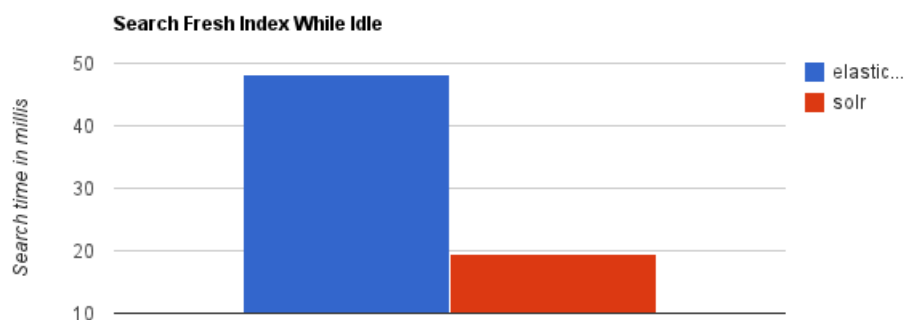


Figure 12: comparison between solr and elasticsearch in traditional searching

Second, we should download solr in our own computer and change the configuration sets inside. Then we should import the data from mysql to solr using either JDBC or mysql-river, or we can simply export the mysql database into a csv or json file and set the file format so solr can recognize the different fields inside. And since JDBC does not support java version higher than 9, I chose the second method.

Here is the configuration.

```
1 <field name="PAPER_ID" column="PAPER_ID"/>
2 <field name="TITLE" column="TITLE"/>
3 <field name="PAPER_PUBLISH_YEAR" column="PAPER_PUBLISH_YEAR"/>
4 <field name="CONFERENCE_ID" column="CONFERENCE_ID"/>
5 <field name="AUTHOR_ID" column="AUTHOR_ID"/>
6 <field name="AUTHOR_NAME" column="AUTHOR_NAME"/>
7 <field name="AFFILIATION_ID" column="AFFILIATION_ID"/>
8 <field name="AFFILIATION_NAME" column="AFFILIATION_NAME"/>
9 <field name="CONFERENCE_NAME" column="CONFERENCE_NAME"/>
10 <field name="AUTHOR_SEQUENCE" column="AUTHOR_SEQUENCE"/>
11 <field name="REFERENCE_ID" column="REFERENCE_ID"/>
```

Third, we should search by solr and export the data to php file. For this, we can either use the method given by solr in its search engine extensions and import the data based on some php files given, or we can simply get the file by tracing the url and loading down all the contents on it. For convenience, I chose the latter. After this, we can decode the string into an array and use the array to print the table.

Here is the first method.

```

1 <?php
2 include "bootstrap.php";
3 $options = array
4 (
5     'hostname' => "localhost",
6     'path'      => "solr/papers",
7     'port'      => "8983",
8 );
9 $client = new SolrClient($options);
10 $query = new SolrQuery();
11 $query->setQuery(' {!join from=AUTHOR_ID to=AUTHOR_ID}AUTHOR_NAME:
12     marian* ');
13 $query->setStart(0);
14 $query->setRows(0);
15 $query->addFacetField('AUTHOR_ID');
16 $query_response = $client->query($query);
17 $response = $query_response->getResponse();
18 print_r($response);
19 ?>

```

Here is the second method, which will return array-like data.

```

1 $page=$_GET['page'];
2 $name=$_GET['name'];
3 $pageSize=10;
4 $start = 10*($page-1);
5 $full_request_URL1 = "http://localhost:8983/solr/papers/select?
6     facet.field=AUTHOR_ID&facet=on&facet.limit=10
7     &facet.offset=$start&fl=PAPER_ID,AUTHOR_ID
8     &q={!join+from=AUTHOR_ID+to=AUTHOR_ID}
9     AUTHOR_NAME:*$name*&rows=0&omitHeader=true
10     &wt=php&facet.mincount=1";
11 $solr_return1 = file_get_contents($full_request_URL1);
12 eval(' $solr_array1 = '. $solr_return1. ');
13 $usefel_array1 = $solr_array1['facet_counts']['facet_fields']['
14     AUTHOR_ID'];

```

And then it is easy to substitute mysql sentences.

Solr sentence in hint.php

```

1 http://localhost:8983/solr/papers/select?
2 fl=AUTHOR_NAME&q=AUTHOR_ID:(AUTHOR_NAME:*$ANAME*)
3 &rows=10&omitHead=true&wt=php

```

Solr sentence in result.php

```

1 http://localhost:8983/solr/papers/select?
2 facet.field=AUTHOR_ID&facet=on
3 &facet.limit=10&facet.offset=$start
4 &fl=PAPER_ID,AUTHOR_ID
5 &q={!join+from=AUTHOR_ID+to=AUTHOR_ID}
6 AUTHOR_NAME:*$name*&rows=0
7 &omitHeader=true&wt=php&facet.mincount=1

```

Solr sentence in author.php

```

1 http://localhost:8983/solr/papers/select?
2 facet.field=REFERENCE_ID&facet=on
3 &fl=AUTHOR_ID,REFERENCE_ID
4 &q={!join+from=PAPER_ID+to=REFERENCE_ID}AUTHOR_ID:$id
5 &omitHead=true&rows=0&facet.mincount=1&wt=php

```

### 4.3 Discussion

Solr has significantly boosted the speed. And its user interface makes it easier to manipulate. It also has many fascinating functions like facet and dismax.

## 5 Pagination

### 5.1 Problem Description

Create a page-bar to enable users to jump to other pages and also the previous and next page. Use ajax in jquery to reach this target.

### 5.2 Analysis and Solution

First, we should get the current page, and decide from which to which pages are needed. Then we can operate in the javascript to create a page-bar with all the numbers and "previous" "next" pages involved. Each of them are linked to a hyperlink. Here is how to get the page. And we will use the getList function to substitute the current page.

```

1 $("#page").on( 'click', 'a', function () {
2     var page=$(this).attr( "data" );
3     getList( page );
4 });

```

Here is the getList function.

```

1 function getList( page ) {
2     $.ajax( {
3         type: "get",

```

```

4      data: {
5          page: page,
6          name: name
7      },
8      url: "result_back.php",
9      success: function (response) {
10         var json = $.parseJSON(response);
11         var s="";
12         for (var i = 0; i < json.length - 1; i
13             ++ ) {
14             s += '<h3>' + '<a href="
15                 author_front.php?id='+json[i]
16                 ].AUTHOR_ID+'>' + json[i].
17                 AUTHOR_NAME+'</a>' + '</h3>'
18                 +
19                 '<div>'+
20                 '<h4>' + '<a href="
21                     author_front.php?id=
22                     ' + "NAME:" + json[i]
23                     ].AUTHOR_ID+'>' +
24                     json[i].AUTHOR_NAME+
25                     '</a>' + '</h4>' +
26                     '<h4 class="thick">' +
27                     "ID:" + json[i].
28                     AUTHOR_ID + '</h4>'
29                     +
30                     '<h4>' + "COUNT:" +
31                     json[i].COUNTS + '</
32                     h4>' +
33                     '<h4 class="thick">' +
34                     "AFFILIATION:" +
35                     json[i].
36                     AFFILIATION_NAME + '
37                     </h4>' +
38                     '</div>';
39         }
40         $("#content").html(s);
41         var s2 = '';
42         var previous = page-1;
43         var next = parseInt(page)+1;
44         if (page!=1)s2 += '<a href="javascript:
45             void(0)" data="" + previous + ">' +
46             'previous' + '</a>';

```



```

26         for (var j = 1; j <= json[i].total && j
27             <= parseInt(page)+5; j++) {
28             if (page == j) s += j;
29             else s2 += '<a href="javascript:void(0)" data
30                 =' + j + '>' + j + '</a>';
31             }
32             if (page!=json[i].total && json[i].total
33                 !=1)s2 += '<a href="javascript:void
34                 (0)" data=" + next + '>' + 'next '
35                 + '</a>';
36             $(" #page").html(s2);
37         }
38     });
39 }

```

### 5.3 Discussion

This kind of page-bar allows jumping instead of clicking on "next" button several times to reach a page. And we can also decide how many pages to reveal for beautification.

## 6 Must Section

### 6.1 Introduction

The final project requires us to form a team including 4 team members. We have to integrate the web system we complete before and perfect its functions as well as do some improvements. Concretely, there are three must-finished tasks and some optional add tasks.

### 6.2 Must Task

#### 6.2.1 Task 1

**Problem Description** In this task, we have to add searching function for paper title and conference. **Analysis and Solution** To make the function more logical, we should add this function in home page, which means adding a triple option searching bar in home page. To implement this function, the home page should include one input box and three buttons with three different names. The code are as below

```

1 <body>
2     <form action="result.php" method="get">
3     <div class="form">
4         <input type="text" id="content" name="content">

```

```

5         <button type="submit" name="author">Author</
           button>
6         <button type="submit" name="paper">Paper</
           button>
7         <button type="submit" name="conference">
           Conference</button>
8     </div>
9 </form>
10 </body>

```

Since there are three different buttons in home page, once the user click one of them, the result.php should get the input content and decide which button was clicked. This can be done by function isset and if statement. The code are as below

```

1 session_start();
2 ini_set('max_execution_time', '0');
3 $con = mysql_connect("localhost", "root", "");
4 if (!$con) {die('could not connect: '.mysql_error());}
5 mysql_select_db("main_db", $con);
6
7 $_SESSION['count']=0;
8
9 $_SESSION['AuthorName'] = (isset($_GET['author']) ? $_GET['
   content'] : null);
10 $_SESSION['PaperTitle'] = (isset($_GET['paper']) ? $_GET['
   content'] : null);
11 $_SESSION['ConferenceName'] = (isset($_GET['conference']) ?
   $_GET['content'] : null);
12 if (isset($_GET['author'])) {...}
13 if (isset($_GET['paper'])) {...}
14 if (isset($_GET['conference'])) {...}

```

Discussion Actually, our first idea is to create a drop-down box with three options, but after overall considering, we decided to adopt the solution above.

## 6.2.2 Task 2

**Problem Description** In this task, we should add paper page and conference page. And according to the PaperID or ConferenceID it get, it will display its detailed information. **Solution for Paper Page** First, get the PaperID and build connection, which is the most important part, for the following the searching statements are based on it directly or indirectly. In Task 1, in second if statement, we also build a href from result.php to paper.php to pass the PaperID the user choose. So we can get the PaperID in paper.php.

```

1 <body>

```

```

2 <form action="paper.php" method="get">
3 <?php
4 ini_set( 'max_execution_time' , '0' );
5 $PaperID=$_GET[ "PaperID" ];
6 $con = mysql_connect( "localhost" , "root" , "" );
7 if (!$con) {die( 'could not connect: '.mysql_error() );}
8 mysql_query( "set names utf8" , $con );
9 mysql_select_db( "main_db" , $con );

```

Then find all the authors of this paper in order.

```

1 $sql="select AuthorID
2         from paper_author_affiliation
3         where PaperID='$PaperID'
4         order by AuthorSequence";
5 $res1=mysql_query( $sql , $con );

```

Then we can show them with hyperlink to author.php.

```

1 echo "<table align='center' class='gridtable'>
2 <tr>
3 <th> Author Name (By Author Sequence)</th>
4 </tr>";
5
6 while ( $row=mysql_fetch_array( $res1 ) )
7 {
8     $aid=$row[ 'AuthorID' ];
9     $sql2="select AuthorName from authors where AuthorID='
10         $aid'";
11     $res2=mysql_query( $sql2 , $con );
12     $row2=mysql_fetch_array( $res2 );
13     echo "<tr><td><a href='author.php?AuthorID=$aid'>".
14         $row2[ 'AuthorName' ]. "</a></td>";
15 }
16 echo '</tr>';
17 echo '</table>';

```

Solution for Conference Page Similarly, get the ConferenceID and find the most cited papers of this conference.

```

1 <?php
2
3 $con = mysql_connect( "localhost" , "root" , "" );
4 if (!$con) {die( 'could not connect: '.mysql_error() );}
5 mysql_query( "set names utf8" , $con );

```

```

6 mysql_select_db("main_db", $con);
7
8 $ConferenceID = $_GET['ConferenceID'];
9 $sql = "select ReferenceID, count(*) from paper_reference where
      ReferenceID in
10 (select PaperID from papers where ConferenceID='$ConferenceID')
11 group by ReferenceID order by count(*) desc limit 20;";
12 $res = mysql_query($sql, $con);
13 $all_row = array();
14 while ($row = mysql_fetch_array($res))
15     array_push($all_row, $row['ReferenceID']);
16 $all_row = array_unique($all_row);

```

Then display them each with a hyperlink to paper.php.

```

1 echo "<table align='center' class='gridtable'>
2 <tr>
3 <th>Paper Title</th>
4 </tr>";
5 foreach($all_row as $pid) {
6     echo "<br>";
7     $sql2="select Title from papers where PaperID='$pid'";
8     $res2=mysql_query($sql2, $con);
9     $row2=mysql_fetch_array($res2);
10    $pname=$row2['Title'];
11    echo "<tr><td><a href='paper.php?PaperID=$pid'>". $pname
12        . "</a></td>";
13 }
14 echo "</table>";
15 mysql_close($con);
16 ?>

```

### 6.2.3 Task 3

**Problem Description** In this task, we have to add paging function whenever needed.  
**Analysis and Solution** To make the web system more compact, we incomplete all the paging function in one previous and next pages. And, considering the number of conferences is so small, we don't add paging function for it. This is done also by using if statement and most importantly, the global variable. Take the code of next.php as an example.

```

1 session_start();
2
3 $con = mysql_connect("localhost", "root", "");

```

```

4 if (!$con) {die('could not connect: '.mysql_error());}
5 mysql_select_db("main_db", $con);
6
7 $_SESSION['count'] = $_SESSION['count']+1;
8 $begin = $_SESSION['count']*10;
9
10 if ($_SESSION['AuthorName']==null) {...}
11 if ($_SESSION['PaperTitle']==null) {...}

```

Discussion To avoid unnecessary duplicate searching and to save time, we play a little trick. Take author searching as an example. In result.php, first we search for all the authors and get the total number.

```

1 $AuthorName=$_SESSION['AuthorName'];
2
3     $sql = "select AuthorID, count(*)
4     from paper_author_affiliation
5     where AuthorID in
6     (select AuthorID from authors where AuthorName like '%
7         $AuthorName%')
8     group by AuthorID order by count(*) desc;
9     ";
10    $result = mysql_query($sql, $con);
11    $_SESSION['total'] = 1+round(mysql_num_rows($result) /
12        10);

```

Then, the information of top 10 that has to be shown in result.php could be get from the first 10 elements in \$result directly.

```

1 echo "<table align='center' class='gridtable' id='newpage'>
2     <tr>
3     <th>Author Name</th>
4     <th>Author ID</th>
5     <th>Num of Paper</th>
6     <th>Main Affiliation</th>
7     </tr>";
8     $i = 0;
9     while($row = mysql_fetch_array($result) and $i<10) {
10        echo "<tr>";
11        $ID=$row['AuthorID'];
12        $sql2 = "select AuthorName from authors where
13            AuthorID='$ID'";
14        $result2 = mysql_query($sql2);
15        echo "<td> <a href='author.php?AuthorID=$ID'>".
16            mysql_fetch_array($result2)['AuthorName'] .

```

```

15         "</a> </td>";
16     echo '<td>'. $row[ 'AuthorID' ]. '</td>';
17     echo "<td>" . $row[ 'count(*)' ] . "</td>";
18     $sql3 = "select AffiliationID , count(*) from
19             paper_author_affiliation where AuthorID='$ID
20             ' and AffiliationID <>'None'
21             group by AffiliationID order by count(*) desc
22             limit 1";
23     $result3 = mysql_query($sql3);
24     $afID = mysql_fetch_array($result3)[ '
25             AffiliationID '];
26     $sql4 = "select AffiliationName from
27             affiliations where AffiliationID='$afID '";
28     $result4 = mysql_query($sql4);
29     echo "<td>" . mysql_fetch_array($result4)[ '
30             AffiliationName ' ] . "</td>";
31     echo "</tr>";
32     $i = $i + 1;
33 }
34 echo "</table>";

```

## 7 Guess you like

### 7.1 Recommendation

#### 7.1.1 Problem Description

In this task, we have to recommend some papers to users. We call it 'Gauss you like' part.

#### 7.1.2 Analysis and Solution

The principles of recommending is as below 1. Same author. 2. Same reference. 3. Most cited For a certain paper, in task 2, we have already find all the authors. Then, we can find all the papers the author have and find the most cited ones.

```

1 $gyl_paper=array();
2
3 while ($row=mysql_fetch_array($res1))
4 {
5     $aid=$row[ 'AuthorID' ];
6     $sql2="select AuthorName from authors where AuthorID='
7         $aid '";
8     $res2=mysql_query($sql2 , $con);

```

```

8     $row2=mysql_fetch_array($res2);
9     echo "<tr><td><a href='author.php?AuthorID=$aid'>".
        $row2[ 'AuthorName' ]. "</a></td>";
10
11     $sql3="select ReferenceID, count(*) from
        paper_reference where ReferenceID in (
12     select PaperID from paper_author_affiliation where
        AuthorID='$aid' )
13     group by ReferenceID order by count(*) desc limit 2";
14     $res3=mysql_query($sql3, $con);
15
16     while($row3=mysql_fetch_array($res3)) {array_push(
        $gyl_paper, $row3[ 'ReferenceID' ]);}
17 }

```

Next, we can find the same reference and so get the most cited ones.

```

1 $sql4="select ReferenceID from paper_reference where PaperID='
    $PaperID' ";
2 $res4=mysql_query($sql4, $con);
3 while($row4=mysql_fetch_array($res4)) {
4     $rid=$row4[ 'ReferenceID' ];
5     $sql5="select ReferenceID, count(*) from
        paper_reference where ReferenceID in (
6     select PaperID from paper_reference where ReferenceID='
        $rid' )
7     group by ReferenceID order by count(*) desc limit 2;";
8     $res5=mysql_query($sql5, $con);
9     while($row5=mysql_fetch_array($res5)) {array_push(
        $gyl_paper, $row5[ 'ReferenceID' ]);}
10 }

```

Now, we can display the recommend list each with a hyperlink to paper.php.

```

1 echo "<table align='center' class='gridtable'>
2 <tr>
3 <th> Guess you like </th>
4 </tr>";
5 $gyl_paper=array_unique($gyl_paper);
6 foreach($gyl_paper as $pid) {
7     echo "<br>";
8     $sql6="select Title from papers where PaperID='$pid' ";
9     $res6=mysql_query($sql6, $con);
10    $row6=mysql_fetch_array($res6);
11    $pname=$row6[ 'Title' ];

```

```
12     echo "<tr><td><a href='paper.php?PaperID=$pid'>". $pname  
13         . "</a></td>";  
14 }  
15 echo '</tr>';  
16 echo '</table>';  
17 mysql_close($con);
```

### 7.1.3 Discussion

The recommend principles can be various. For example, we can take key words into consideration, or we can use a score system to weight the nominated papers. Scoring system is easy to complete, while recommending by key words is sort of challenging, which can be put in our improvement part.