



Experiment Final Academic Galaxy

518030910279 Qu Bowen
518030910282 Tang Shuo

518030910280 Shi Runhan
518030910319 Zhang Jiale

July 23, 2019

Problem Description

Integrate the website system completed before, improve the function and optimize the interface.

1. Compulsory

1. Add paper and conference pages in the website.
2. When displaying too much information on each page, only display 10 items, and add page turning function.
3. Add echarts chart visualization, such as visualization of the year of the conference or scholar's publication on each conference home page and scholar's home page.
4. Make sure all the codes and special effects are compatible with each other.

2. Optional

For the sake of gaining more scores, we added the following kinds of functions.

1. Paper recommendation on the paper and author pages.
2. More visualization functions.
3. Some animations.
4. Database optimization.

3. Problem Analysis

To settle these problems, our group has been divided into 3 major parts: 1 for paper and conference pages, 1 for charts visualization, and 2 for beautification of the website. We cooperated, discussed, and presented all the work efficiently.

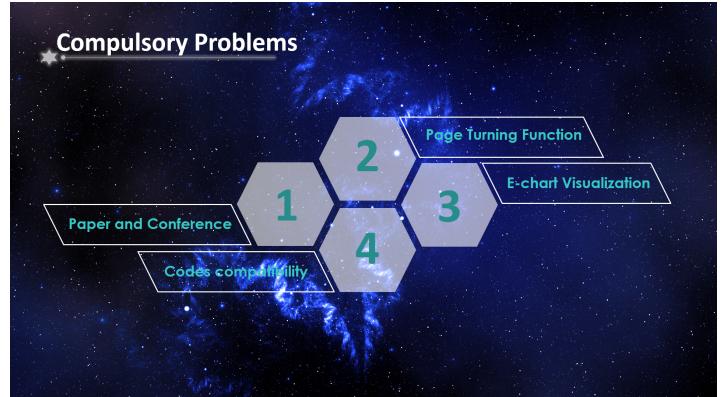


Figure 1: Compulsory Problems.

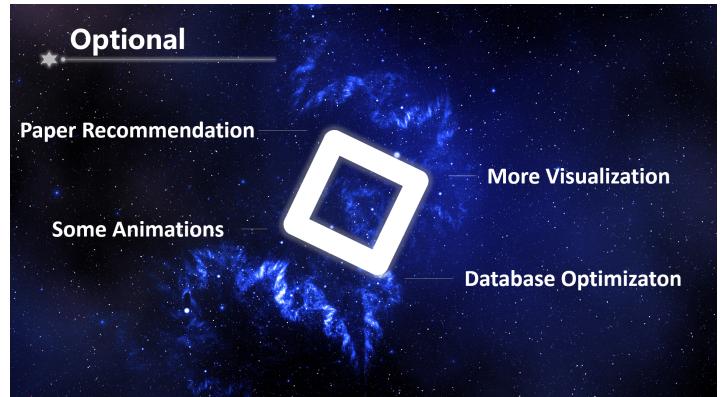


Figure 2: Optional.

Back-end

1. Queries

1.1 Basic statement

Select column1 from table1 where column2=value ;

If the amount of information in the database is pretty large, then the execution time is unimaginable. In MySQL Workbench, we can use a statement to see how many rows we have searched in our database. Just add “explain” before our query statement to see the result, if we run the query very slow , we might find in Figure 3. We can see that the value of ”rows” is 268730!

1.2 Add indexes

The most effient way to make it much faster is to add indexes in our database. It can greatly reduce the amount of the data items the computer needed to search. The example is in Figure 4.

Basic statement: CREATE INDEX index_name
ON table_name(column_name);

When you input a query statement, the computer quickly limited the rows to a very small value by using B+ tree.

Note: Using a function to change the value of the column , the system will not use the corresponding index. For example: “select column1 from table1 where column1-1=value”.

1.3 Data structure

The data structure of solr is just a big dictionary. In order to make the database more complete, we add fieldname "Reference ID", "Reference Name", "Reference Year". So we can just search the fieldname to find the result.

1.4 Code in php

In php, we need to connect to our solr and execute our query. First is to initialize the search engine, and use a variable \$url to represent a http request. In order to make the search more reliable, we should do trans-coding the data item. That's what the function “urlencode()” did. And the search result is put in the variable \$result.

Note: We can search the information accurately by adding double quotation mark.

Every data item is searched by solr, because the relationship of those data items is much more clear in solr than in MySQL. If we use MySQL, we will use many joint query statements to be slowed down.

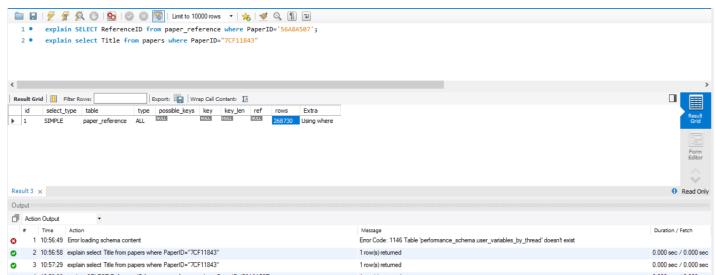


Figure 3: Basic statement.

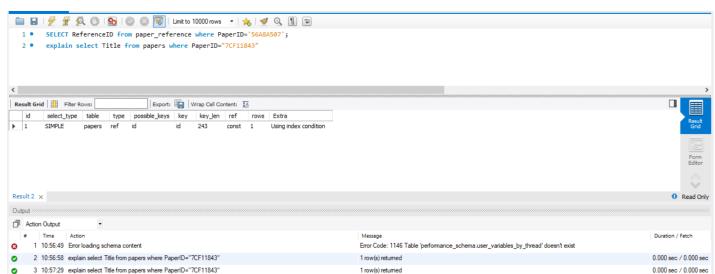


Figure 4: Add indexes.

```
for (int i = 0; i < data.length(); i++) {
    String line = data[i];
    String[] tokens = line.split(" ");
    List<String> list = new ArrayList<>();
    for (int j = 0; j < tokens.length; j++) {
        list.add(tokens[j]);
    }
    if (tokens[0].equals("PaperID")) {
        list.add("ReferenceID");
        list.add("ReferenceName");
        list.add("ReferenceYear");
        list.add("AuthorName");
        list.add("ReferenceName");
    } else if (tokens[0].equals("Title")) {
        list.add("ReferenceID");
        list.add("ReferenceName");
        list.add("ReferenceYear");
        list.add("AuthorName");
        list.add("ReferenceName");
    } else if (tokens[0].equals("Year")) {
        list.add("ReferenceID");
        list.add("ReferenceName");
        list.add("ReferenceYear");
        list.add("AuthorName");
        list.add("ReferenceName");
    } else if (tokens[0].equals("AuthorName")) {
        list.add("ReferenceID");
        list.add("ReferenceName");
        list.add("ReferenceYear");
        list.add("AuthorName");
        list.add("ReferenceName");
    } else if (tokens[0].equals("ReferenceName")) {
        list.add("ReferenceID");
        list.add("ReferenceName");
        list.add("ReferenceYear");
        list.add("AuthorName");
        list.add("ReferenceName");
    }
}
```

Figure 5: Data structure.

```
$ch = curl_init();
$timeout = 5;
$query = urlencode(str_replace(' ', '+', $paper_title));
$url = "http://localhost:8983/solr/lab2/select?indent=on&q=$Title";
$query .= "&wt=json" . "&rows=$rows" . "&start=$start";
curl_setopt ($ch, CURLOPT_URL, $url);
curl_setopt ($ch, CURLOPT_RETURNTRANSFER, 1);
curl_setopt ($ch, CURLOPT_CONNECTTIMEOUT, $timeout);
$result = json_decode(curl_exec($ch), true);
curl_close($ch);
```

Figure 6: Code in php.

Back-end

2. Page Framework

Our framework is simple, first is the homepage, the users input the data they want to search, then the data is transferred to a html form. Other pages just use GET() function to use the data in the html form. We have other 4 pages, they are the search page, papers page, conferences page and authors page. Every page used hyperlink to connect to other pages, and we added the information of “rows” and “start” to limit the query , in this way we can realize page turning.

```
262 $all_pages=$result["response"]["numFound"];
263 $first_page=0;$this_page=1;
264 if($all_pages>100 and $all_pages<10)
265 {
266     $total=(int)((($all_pages-1)/10));
267     if($this_page==0)
268     {
269         for($i=1;$i<$total;$i++)
270         {
271             if($i==1){echo "<div class=pagination><a href=search.php&paper_title=$paper_title&start=$page&author_name=null">1</a></div>"}
272             else{echo "<div class=pagination><a href=search.php&paper_title=$paper_title&start=$page&author_name=&conference_name=null">$i </a></div>"}
273         }
274         $next=($this_page+1)*10;
275         echo "<a href=search.php&paper_title=$paper_title&start=$next&author_name=&conference_name=null">next </a></div>";
276     }
277     else
278     {
279         $now=$this_page/10;
280         $prev_page=$now-1;
281         $next_page=$now*10;($next_page>$total){$next_page=$total};
282         $start=$prev_page*10;
283         $end=$next_page*10;
284         echo "<div class=pagination><a href=search.php&paper_title=$paper_title&start=$prev&author_name=&conference_name=null">prev </a></div>";
285         for($i=1;$i<$total;$i++)
286         {
287             if($i==1){echo "<div class=pagination><a href=search.php&paper_title=$paper_title&start=$page&author_name=&conference_name=null">1</a></div>"}
288             else{echo "<div class=pagination><a href=search.php&paper_title=$paper_title&start=$page&author_name=&conference_name=null">$i </a></div>"}
289         }
290         echo "<a href=search.php&paper_title=$paper_title&start=$next&author_name=&conference_name=null">next </a></div>";
291     }
292 }
```

Figure 7: Page turning-1.

Page turning

We just used the hyperlink to realize page turning. Basic idea is simple, in solr and MySQL, we can limit the result of our queries. So when the amount of result is very large, we can use the limited query to show the limited information. And we only need to transfer the value of “start”“rows” to let the query statement know the search starts at which position and ends where. Here is the related code:

Determine we will use how many pages, if pages is less than 10, then we will not make present page in the middle of the page list. And if the present page is the first page, we will not show the “last page” and the “front page”.

If total pages is more than 10, we need to put the present page in the middle, so that our users can both click the pages before this page and the pages after this page. Every page is packed by a hyperlink, and we added some determine statements to ensure the code be stable.

```
) elseif ($all_pages>100)
{
    $total=(int)((($all_pages-1)/10));
    if($this_page>0)
    {
        if($this_page==1){echo "<div class=pagination><a href=search.php&paper_title=$paper_title&start=$page&author_name=null">1</a></div>"}
        for($i=1;$i<10+$i);
        $page=($i-1)*10;
        if($i==($this_page+10)){echo "<div class=pagination><a href=search.php&paper_title=$paper_title&start=$page&author_name=null">...</a></div>"}
        else{echo "<div class=pagination><a href=search.php&paper_title=$paper_title&start=$page&author_name=&conference_name=null">$i </a></div>"}
    }
    $next=$this_page*10;
    $prev_page=$now-1;
    $now=$this_page/10;
    $out_page=$now*10;($out_page>$total){$out_page=$total};
    $next_page=$now*10;($next_page>$total){$next_page=$total};
    $prev=$prev_page*10;($prev>$out_page){$prev=$out_page};
    $in=$now*10;$out=$out_page*10;
    echo "<div class=pagination><a href=search.php&paper_title=$paper_title&start=$prev&author_name=&conference_name=null">prev </a></div>";
    for($i=$in;$i<$out_page+$i);
    $page=($i-1)*10;
    if($i==($this_page+10)){echo "<div class=pagination><a href=search.php&paper_title=$paper_title&start=$page&author_name=null">...</a></div>"}
    else{echo "<div class=pagination><a href=search.php&paper_title=$paper_title&start=$page&author_name=&conference_name=null">$i </a></div>"}
}
echo "<a href=search.php&paper_title=$paper_title&start=$next&author_name=&conference_name=null">next </a></div>";
```

Figure 8: Page turning-2.

Data Visualization

As the saying goes "A graph is worth a thousand words", to make data visualized is important in a searching website.

1. Interpretations of the Charts in the Website

There are more than ten charts of four types in the website: Line Charts, Bar Charts, Pie Charts and Radar Charts. We will introduce them one by one with examples.

1.1 Line Charts: Number of Papers in Different Years

Line chart is a good form to show the change of number in different period of time. In this broken line chart we can infer that the conference IJCAI is held every two years. The example is in Figure 9.

1.2 Pie Charts: Ratio of Papers in Different Conferences

Pie chart is suitable to show the ratio of number in different parts. So we can learn which conference makes up the majority of the papers. Here we search 'AI', and the result show that the two most famous artificial intelligence conferences 'IJCAI' and 'AAAI' have the most papers in this fields. The example is in Figure 10.

1.3 Bar Charts: The Top 10 Most Productive Authors

Bar chart can show the comparison of numbers between different items. Here we list the top 10 most productive authors and the number of their works. In this example, there is no one who has far more papers than others. The example is in Figure 11.

1.4 Radar Chart: The Comprehensive Assessment of A Conference

Radar chart is suitable for multidimensional data like the competence of a person or an affiliation or, as shown here, a conference. In Figure 12, a comprehensive assessment of the conference AAAI with 6 indexes revealed the average year of the papers in this conference is relatively new and the production of the 1st authors of this conference is very high.

To better visualize the data, the plotting scale of six axes are adjust several times. It may be better to squash the data with a function like sigmoid to make it convergent to this region.

Publishyear - number of papers diagram

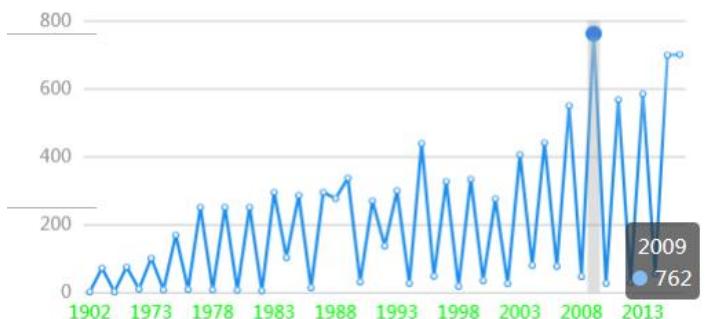


Figure 9: Line chart.

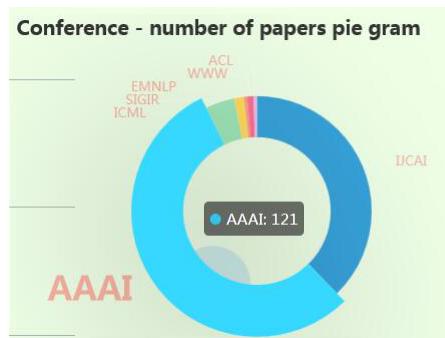


Figure 10: Pie chart.

Productive Authors Top 10



Figure 11: Bar chart.

Comprehensive Assessment

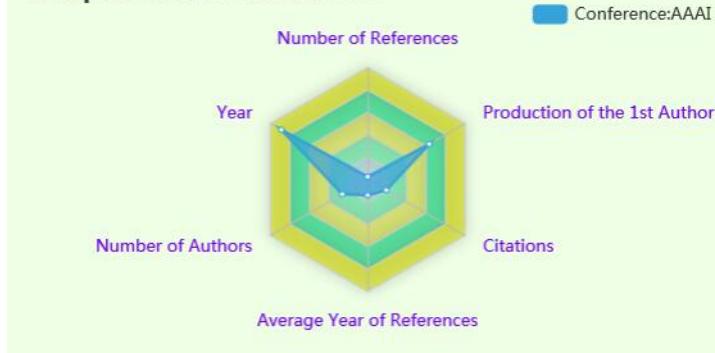


Figure 12: Radar chart.

Data Visualization

2.The Ways to Make Such Charts

We visualize the data with Echart, a powerful tool provided by baidu. To visualize the data, we need first move data from the back-end to the front-end with PHP, then set the option of the charts with javascript. The following is how we did so.

2.1 Get Data from Solr

In the block of code above, if we want to search for other information in solr, we just need to modify the url.

To take advantage of the efficiency of solr, we add more fields to the solr core and upload the data again.

2.2 Make Statistics in PHP

We need to process the data to get the statistics before we turn it into a chart.

2.3 Transfer PHP Variables to Javascript Variables

Transferring the variables into the forms that can be recognized by echarts is the critical process of data visualization. As can be found on the Internet, there are many ways to do it.

Here we adopt a easy way to conduct it without ajax or jquery.

The PHP code is executed by the server and echo the result before the js codes are parsed by the browser. So PHP can be inserted in the <script> label and produce js codes.

This is the easiest way we found to get the data from the back-end to the front-end.

2.4 Set the Option of Echarts

The options of echarts is very complicated. We only need to understand the most frequently used terms and revise these important parts.

Among these are title, legend, type, center, color and data. With these six attributes. We can control what information we express in the chart and the position, shape and color of charts.

We can also learn from others' example to make the chart more beautiful.

```
$ch = curl_init();
$timeout = 5;
$url = "http://localhost:8983/solr/lab2/select?indent=on&q=ConferenceID:".$conf_id."&wt=json". "&rows=99999";
curl_setopt ($ch, CURLOPT_URL, $url);
curl_setopt ($ch, CURLOPT_RETURNTRANSFER, 1);
curl_setopt ($ch, CURLOPT_CONNECTTIMEOUT, $timeout);
$result1 = json_decode(curl_exec($ch), true);
```

Figure 13: First we need to get data from solr or mysql.

Here we get the result from solr since it is faster than mysql

```
solr.add(
  [
    "PaperID": line[0], "Title": line[1],
    "Year": line[2], "ConferenceID": line[3],
    "ConferenceName": dic3[line[3]],
    "AuthorID": dic4[line[0]], "AuthorName": list1,
    "ReferenceID": [""], "ReferenceName": [""],
    "ReferenceYear": [""], "ReferenceConferenceName": [""]
  ],
  );
```

Figure 14: Part of the python code, showing the fields of the solr documents. With the expanded fields like 'ReferenceYear', 'ReferenceConferenceName'. We can make the statistics more conveniently.

```
foreach ($result1['response']['docs'] as $value) {
    $tot++;
    if($tot>20)
        break;
    if(@$value['ReferenceID'])
    {
        $data[0]+=$value['ReferenceID'];
        $data[3]+=$array_sum($value['ReferenceYear'])/count($value['ReferenceYear']);
    }
    $data[1]+=(int)$value['Year'];
    $data[2]+=count($value['AuthorID']);
```

Figure 15: Make statistics in PHP. In this example, we get the data of a conference and want to make comprehensive assessment of the conference with a radar graph. We need to calculate the value of all six dimension in PHP and store it with an array.)

```
<script>
<?php
    echo "var dataAxis =Array();var data=Array();";
    foreach ($data as $key => $value)
        echo "dataAxis[$key]=$key;data[$key]=$value;"; ?> </script>
```

Figure 16: Echo javascript code in <script> label to transfer data from php variables to js variables.

Homepage

For the homepage, we do several beatifications as we describe below.

1. We inherit the former lab to use three search boxes and turn them into three planets rotate round the central planet, T A C, and when your mouse hover on one of the planets, the three planets stop at once. The planet your mouse hover on will enlarged and will recover when your mouse out.

2. When you click one of the planets, it will increase to take up a position in the center of the pageclick the cross to returnchange the pattern and lengthen the input box.

3. Beautify the submit button.

4. The two image hyperlinks linked to acemap and homepage of our classyou're your mouse hover on it, it will shake.

5. Simulate a book to show our team.

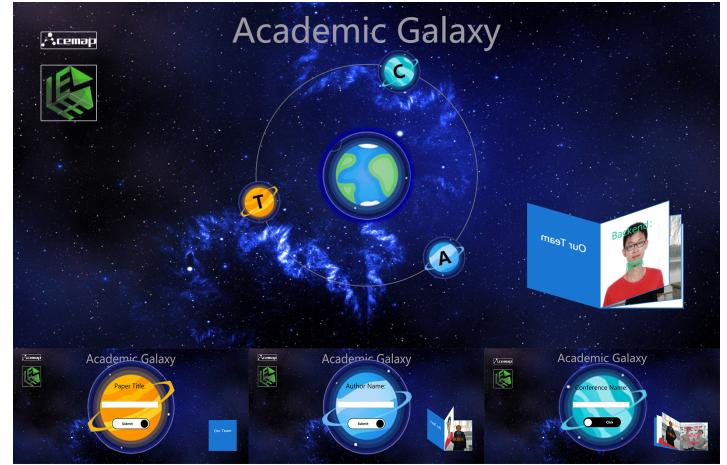


Figure 17: Our homepage.



Figure 18: Homepage planet.

1. Code of planet

In Figure 3 from line 509 to line 513 we build central planet and the orbit, the form of the central planet are in line 92 to line 102 and the form of the orbit is in line 202 to line 214. Then from line 516 to line 522, we build the three planets, the form of the title search engine is in line 105 to line 124, and we use the css function animation and hover in line 126 to line 132 to let it grow bigger, the form of how the three plants rotate is inline 216 to line 219.

2. Code of search planet

From line 524 to line 521 in Figure 4 we build the title search planet. We control the appear and disappear of the planet by JavaScript function in line 20 to line 57 by control the width and height of the search planet.



Figure 19: Homepage search planet.

Homepage

3. Code of Submit

In Figure 6 we redesign the form of the submit button. It is a black circle on the right and a black "Submit" on the white background, and when your mouse hover on it, the circle turn into white and slide into the left side, at the same time, the "Submit" turn to a white "Click" and the background turn to black. This is realized by line 246 to line 324, we use css function hover and father and son elements to realize this.

4. Code of hyperlinks

In Figure 7 we build two image hyperlinks linked to acemap and homepage of our class, and if you put your mouse on it, it will shake. The body code is in Figure 8 line 469 to line 474, and this is also realized by the css function animation and hover and define the key frames shake to make it move left and right to simulate the movement shake in Figure 7 line329 to line 356.

5. Code of "Our Team"

In figure 8 we build a book named "Our Team" to introduce our team. The first page is our CBO(chief back-end officer) tangshuo, the second page is zhangjiale who do visualization, the last page is our front-end maker qubowen and shirunhan. And on each page we put our picture on it. We do this in body line 475 to line 507, and the pattern is made by css code in line 358 to line 461. The main idea of building this is first build several pages in the same scale. Then set a key frame of turn of the page, to make it lifelike, we use 3d turn, and set the same period for the pages and set specially for each page turning time. We take notes in the code to make it easy to understand, so you can get more details in the code.

```
246 /*提交按钮-1*/
247 .btn-slide {
248   position: relative;
249   display: inline-block;
250   height: 50px;
251   width: 200px;
252   line-height: 50px;
253   padding: 0;
254   border-radius: 50px;
255   background: #ffffdf;
256   border: 2px solid black;
257   margin: 10px;
258   transition: .5s;
259 }
260 /*提交按钮-2*/
261 .btn-slide:hover {
262   background-color: black;
263 }
264 /*提交按钮-3*/
265 .btn-slide:hover span.circle {
266   right: 100px;
267   margin-right: -45px;
268   background-color: #fdfdfd;
269   color: black;
270 }
271 }
272 /*提交按钮-4*/
273 .btn-slide:hover span.title {
274   right: 40px;
275   opacity: 0;
276 }
277 }
278 }
279 /*提交按钮-5*/
280 .btn-slide:hover span.title-hover {
281   opacity: 1;
282   right: 40px;
283 }
284 }
285 /*提交按钮-6*/
286 .btn-slide span.circle {
287   display: block;
288   background-color: black;
289   color: #fff;
290   position: absolute;
291   float: right;
292   margin: 5px;
293   line-height: 42px;
294   height: 40px;
295   width: 40px;
296   top: 0;
297   right: 0;
298   transition: .5s;
299   border-radius: 50%;
300 }
301 }
302 /*提交按钮-7*/
303 .btn-slide span.title, .btn-slide span.title-hover {
304   position: absolute;
305   right: 90px;
306   text-align: center;
307   margin: 0 auto;
308   font-size: 16px;
309   font-weight: bold;
310   color: black;
311   transition: .5s;
312 }
313 }
314 /*提交按钮-8*/
315 .btn-slide span.title-hover {
316   right: 80px;
317   opacity: 0;
318 }
319 }
320 /*提交按钮-9*/
321 .btn-slide span.title-hover {
322   color: #fff;
323 }
324 }
```

Figure 20: Code of submit.

```
329 /*按钮*/
330 .lanren {
331     width: 150px;
332     text-align: center;
333     line-height: 40px;
334     border: 1px solid #CCC;
335     border-radius: 2px;
336 }
337 .lanren:hover {
338     animation: shake 0.82s cubic-bezier(.36, .07, .19, .97) both;
339     transform: translate3d(0, 0, 0);
340     backface-visibility: hidden;
341     perspective: 1000px;
342 }
343 @keyframes shake {
344     10%, 90% {
345         transform: translate3d(-1px, 0, 0);
346     }
347     20%, 80% {
348         transform: translate3d(2px, 0, 0);
349     }
350     30%, 50%, 70% {
351         transform: translate3d(-4px, 0, 0);
352     }
353     40%, 60% {
354         transform: translate3d(4px, 0, 0);
355     }
356 }
```

Figure 21: Code of hyperlinks.

```
66<div style="display: flex; justify-content: space-between; border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;">>
67    <a href="https://www.imooc.com">imooc</a>
68    <div style="text-align: center; position: relative; width: 100px; height: 100px; background-color: #ccc; border-radius: 50%;>
69        <div style="position: absolute; top: 0; left: 0; width: 100%; height: 100%; background-color: #fff; border-radius: 50%; z-index: 1;>
70            <div style="position: absolute; top: 50%; left: 50%; transform: translate(-50%, -50%); font-size: 2em; color: #ccc; opacity: 0.5; font-weight: bold; font-family: sans-serif; white-space: nowrap; text-decoration: none; text-align: center; line-height: 1.2; transition: all 0.3s ease-in-out; z-index: 2;>
71                起步进阶
72            </div>
73        </div>
74        <div style="position: absolute; top: 0; left: 0; width: 100%; height: 100%; background-color: #fff; border-radius: 50%; z-index: 1;>
75            <div style="position: absolute; top: 50%; left: 50%; transform: translate(-50%, -50%); font-size: 2em; color: #ccc; opacity: 0.5; font-weight: bold; font-family: sans-serif; white-space: nowrap; text-decoration: none; text-align: center; line-height: 1.2; transition: all 0.3s ease-in-out; z-index: 2;>
76                起步进阶
77            </div>
78        </div>
79    </div>
80</div>
81<div style="display: flex; justify-content: space-between; border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;">>
82    <a href="https://www.imooc.com/page/1/preserve-id">book page 1 preserve id</a>
83    <div style="text-align: center; position: relative; width: 100px; height: 100px; background-color: #ccc; border-radius: 50%;>
84        <div style="position: absolute; top: 0; left: 0; width: 100%; height: 100%; background-color: #fff; border-radius: 50%; z-index: 1;>
85            <div style="position: absolute; top: 50%; left: 50%; transform: translate(-50%, -50%); font-size: 2em; color: #ccc; opacity: 0.5; font-weight: bold; font-family: sans-serif; white-space: nowrap; text-decoration: none; text-align: center; line-height: 1.2; transition: all 0.3s ease-in-out; z-index: 2;>
86                book page 1
87            </div>
88        </div>
89        <div style="position: absolute; top: 0; left: 0; width: 100%; height: 100%; background-color: #fff; border-radius: 50%; z-index: 1;>
90            <div style="position: absolute; top: 50%; left: 50%; transform: translate(-50%, -50%); font-size: 2em; color: #ccc; opacity: 0.5; font-weight: bold; font-family: sans-serif; white-space: nowrap; text-decoration: none; text-align: center; line-height: 1.2; transition: all 0.3s ease-in-out; z-index: 2;>
91                book page 1
92            </div>
93        </div>
94    </div>
95</div>
96<div style="display: flex; justify-content: space-between; border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;">>
97    <a href="https://www.imooc.com/page/2/preserve-id">book page 2 preserve id</a>
98    <div style="text-align: center; position: relative; width: 100px; height: 100px; background-color: #ccc; border-radius: 50%;>
99        <div style="position: absolute; top: 0; left: 0; width: 100%; height: 100%; background-color: #fff; border-radius: 50%; z-index: 1;>
100            <div style="position: absolute; top: 50%; left: 50%; transform: translate(-50%, -50%); font-size: 2em; color: #ccc; opacity: 0.5; font-weight: bold; font-family: sans-serif; white-space: nowrap; text-decoration: none; text-align: center; line-height: 1.2; transition: all 0.3s ease-in-out; z-index: 2;>
101                book page 2
102            </div>
103        </div>
104        <div style="position: absolute; top: 0; left: 0; width: 100%; height: 100%; background-color: #fff; border-radius: 50%; z-index: 1;>
105            <div style="position: absolute; top: 50%; left: 50%; transform: translate(-50%, -50%); font-size: 2em; color: #ccc; opacity: 0.5; font-weight: bold; font-family: sans-serif; white-space: nowrap; text-decoration: none; text-align: center; line-height: 1.2; transition: all 0.3s ease-in-out; z-index: 2;>
106                book page 2
107            </div>
108        </div>
109    </div>
110</div>
111<div style="display: flex; justify-content: space-between; border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;">>
112    <a href="https://www.imooc.com/book/page/1/preserve-id">book page 1 preserve id</a>
113    <div style="text-align: center; position: relative; width: 100px; height: 100px; background-color: #ccc; border-radius: 50%;>
114        <div style="position: absolute; top: 0; left: 0; width: 100%; height: 100%; background-color: #fff; border-radius: 50%; z-index: 1;>
115            <div style="position: absolute; top: 50%; left: 50%; transform: translate(-50%, -50%); font-size: 2em; color: #ccc; opacity: 0.5; font-weight: bold; font-family: sans-serif; white-space: nowrap; text-decoration: none; text-align: center; line-height: 1.2; transition: all 0.3s ease-in-out; z-index: 2;>
116                book page 1
117            </div>
118        </div>
119        <div style="position: absolute; top: 0; left: 0; width: 100%; height: 100%; background-color: #fff; border-radius: 50%; z-index: 1;>
120            <div style="position: absolute; top: 50%; left: 50%; transform: translate(-50%, -50%); font-size: 2em; color: #ccc; opacity: 0.5; font-weight: bold; font-family: sans-serif; white-space: nowrap; text-decoration: none; text-align: center; line-height: 1.2; transition: all 0.3s ease-in-out; z-index: 2;>
121                book page 1
122            </div>
123        </div>
124    </div>
125</div>
126<div style="display: flex; justify-content: space-between; border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;">>
127    <a href="https://www.imooc.com/book/page/2/preserve-id">book page 2 preserve id</a>
128    <div style="text-align: center; position: relative; width: 100px; height: 100px; background-color: #ccc; border-radius: 50%;>
129        <div style="position: absolute; top: 0; left: 0; width: 100%; height: 100%; background-color: #fff; border-radius: 50%; z-index: 1;>
130            <div style="position: absolute; top: 50%; left: 50%; transform: translate(-50%, -50%); font-size: 2em; color: #ccc; opacity: 0.5; font-weight: bold; font-family: sans-serif; white-space: nowrap; text-decoration: none; text-align: center; line-height: 1.2; transition: all 0.3s ease-in-out; z-index: 2;>
131                book page 2
132            </div>
133        </div>
134        <div style="position: absolute; top: 0; left: 0; width: 100%; height: 100%; background-color: #fff; border-radius: 50%; z-index: 1;>
135            <div style="position: absolute; top: 50%; left: 50%; transform: translate(-50%, -50%); font-size: 2em; color: #ccc; opacity: 0.5; font-weight: bold; font-family: sans-serif; white-space: nowrap; text-decoration: none; text-align: center; line-height: 1.2; transition: all 0.3s ease-in-out; z-index: 2;>
136                book page 2
137            </div>
138        </div>
139    </div>
140</div>
```

Figure 22: Code of "Our Team".

Search

For the search page, we do several beautification as I describe below.

1. For the title, we use css animation to realize the gradient color function.

2. For the content, we use css to reach a relatively best visual effect.

3. For the background, we use js and canvas to let random bubbles appear at random position.

4. The visualization echarts are line chart, pie chart and bar chart on the right side of this page.

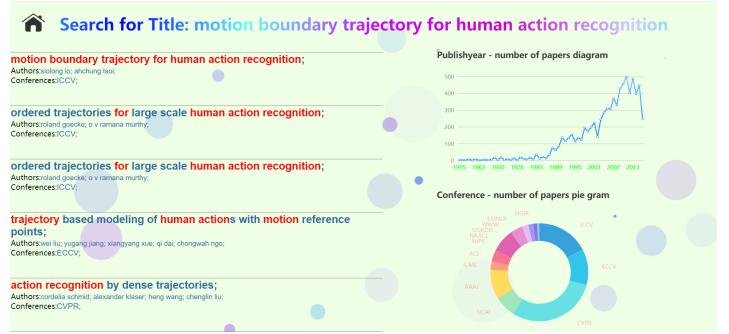


Figure 23: Our search page.

1.Code of title

We used CSS to change the search page title.

Key-codes: keyframes: this sets the title gradient color. 10 different kinds of color transforms from left to right to demonstrate its gradient effect per 4 seconds.

```
1 .test{  
2     position: relative;  
3     left: 80px; width: 1200px;  
4     font-weight: bold;  
5     background-image: -webkit-linear-gradient(left,blue,#66ffff 10%,#cc00ff 20%,  
6         #cc00cc 30%, #ccccff 40%, #00ffff 50%,#cccccc 60%,#cc00cc 70%,#cc00ff 80%,#66ffff 90%,blue 100%);  
7     -webkit-text-fill-color: transparent;  
8     -webkit-background-clip: text;  
9     -webkit-background-size: 200% 100%;  
10    -webkit-animation: masked-animation 4s linear infinite;  
11    font-size: 30px;  
12 }  
13 @keyframes masked-animation {  
14     0% {  
15         background-position: 0 0;  
16     }  
17     100% {  
18         background-position: -100% 0;  
19     }  
20 }
```

Figure 24: Code of title.

2.Code of css style

We used CSS to change the color and the typeface of the information displayed. And we set the “link” “visited” and “hover” properties. So when your mouse is above the words, it becomes brighter. And the searched words in the result will be red.

My page turning style is the same. Just add buttons on every page. We make the color of my present page’s background be blue, so the user can know what the present page is according to the color. It is shown in Figure 23.

```
7     div.div1 a:link{color:#336699;text-decoration:none;}  
8     div.div1 a:visited{color:#336699;text-decoration: underline;}  
9     div.div1 a:hover{color: #0066CC;text-decoration:underline;}  
10    div.div1 {font-size: 23px;font-family: Arial;display: inline;}  
11    div.div2 a:link{color:#336699;text-decoration:none; }  
12    div.div2 a:visited{color:#336699;text-decoration: underline;}  
13    div.div2 a:hover{color: #0066CC;text-decoration:underline;}  
14    div.div2 {font-size: 15px;font-family: Arial;display: inline;}  
15    div.div3 a:link{color:#336699;text-decoration:none; }  
16    div.div3 a:visited{color:#336699;text-decoration: underline;}  
17    div.div3 a:hover{color: #0066CC;text-decoration:none;}  
18    div.div3 {font-size: 17px;font-family: Arial;display: inline;}
```

Figure 25: Code of css style.

Search

3.Code of background

In addition, we create a class MyBubble and use canvas to draw bubbling bubbles background.

1. constructor(opts): This the constructor function, setting the default attribute of each bubble. All the annotation are revealed in the picture such as numbers, radius, opacity, color, and color-light.

2. random(a, b): This will return a random number between a and b for the animation.

3. initBubble(color, isSameColor): This sets a single bubble's attribute. The position, radius, radiusChange, opacity, opacityChange, color-light and color are all random. If the color is the same with the original one, remain its color to reach the relatively best effect.

4. bubbling(ctx, color, isSameColor): This implements the bubbling function, draws a arc, changes its radius and color-light. If the opacity is less than 0, stop it. And ctx is a 2d canvas.

5. createCanvas(): This sets the overall canvas. The style of display is ‘block’ to prevent a full-screen scroll bar appearing. And window.onresize() will be exerted when the window or frame is changed(when you slide the scroll bar) to regain its window size. As a result, the bubbling background will not be influenced by the length of the page contents.

6. start(): This starts the whole function. If the number is less than the default one and the probability is enough, push a new MyBubble. In addition, we use requestAnimationFrame() to use the browsing set to control the best animation API. This will best be compatible with your computer and browser.

```
background.js
1 class MyBubble {
2   constructor(opts) {
3     this.userOpts = {
4       id: opts.num, num: 50, start_probability: 0.05, radius_min: 2, radius_max: 3, radius_add_min: .2,
5       radius_add_max: .4, opacity_min: 0.3, opacity_max: 0.5, opacity_prev_min: .005, opacity_prev_max: .005,
6       light_min: 40, light_max: 70, is_same_color: false
7     }
8   this.color = 220 this.bubbleNum = []
9   this.requestAnimationFrame = window.requestAnimationFrame || window.mozRequestAnimationFrame ||
10  window.webkitRequestAnimationFrame || window.msRequestAnimationFrame
11 }
12
13 initBubble(color, isSameColor) {
14   const width = window.innerWidth const height = window.innerHeight
15   const userOpts = this.userOpts const light = this.random(userOpts.light_min, userOpts.light_max)
16   this.bubble = {
17     x: this.random(0, width), y: this.random(0, height),
18     radius: this.random(userOpts.radius_min, userOpts.radius_max),
19     radiusChange: this.random(userOpts.radius_min, userOpts.radius_max),
20     opacity: this.random(userOpts.opacity_min, userOpts.opacity_max),
21     opacityChange: this.random(userOpts.opacity_min, userOpts.opacity_max),
22     color: `hsl(${isSameColor ? color : this.random(200, 280)},100%,${[light]})`,
23   }
24 }
```

Figure 26: Code of background-1.

```
background.js
1 < bubbling(ctx, color, isSameColor) {
2   if(this.bubble && this.initBubble(color, isSameColor))
3     const bubble = this.bubble ctx.fillStyle = bubble.color; ctx.globalAlpha = bubble.opacity;
4     ctx.beginPath(); ctx.arc(bubble.x, bubble.y, bubble.radius, 0, 2 * Math.PI, true);
5     ctx.closePath(); ctx.fill();
6     ctx.globalAlpha = 1; bubble.opacity -= bubble.opacityChange; bubble.radius += bubble.radiusChange;
7   if (bubble.opacity <= 0) {
8     this.initBubble(color, isSameColor) return
9   }
10 }
11 < createCanvas() {
12   this.canvas = document.createElement('canvas') this.ctx = this.canvas.getContext('2d')
13   this.canvas.style.display = 'block';
14   this.canvas.width = window.innerWidth this.canvas.height = window.innerHeight
15   this.canvas.style.position = 'fixed' this.canvas.style.top = '0'
16   this.canvas.style.left = '0' this.canvas.style.zIndex = '-1'
17   document.getElementById(this.userOpts.id).appendChild(this.canvas)
18 < window.onresize = () => {
19   this.canvas.width = window.innerWidth this.canvas.height = window.innerHeight
20 }
21 }
```

Figure 27: Code of background-2.

```
background.js
1 random(a, b) {return Math.random() * (b - a) + a}
2
3 < start() {
4   const width = window.innerWidth const height = window.innerHeight
5   this.ctx.fillStyle = `hsl(${this.color},2%,30%)` this.ctx.fillRect(0, 0, width, height);
6   if(this.bubbleNum.length < this.userOpts.num && Math.random() < this.userOpts.start_probability) {
7     this.bubbleNum.push(new MyBubble())
8   }
9   this.bubbleNum.forEach(bubble => bubble.bubbling(this.ctx, this.color, this.userOpts.is_same_color))
10 const requestAnimationFrame = this.requestAnimationFrame
11 this.myReq = requestAnimationFrame(() => this.start())
12 }
13
16 const bubbleDemo = new MyBubble('bubble')
17 bubbleDemo.createCanvas() bubbleDemo.start()
18
```

Figure 28: Code of background-3.

Papers

For the papers page, we do several beautifications as I describe below.

1. We use css to realize the hover function of the title, including mouse on and mouse off.
 2. We set a menu bar which has three plates. We also build the home button on each page, if your mouse hover on it, it will rotate one round, and click it will return to our homepage.

3. As for Details plate you can see the title, publish year, conference name and affiliations of this papers, and the details of this form include top and bottom border, half-transparent background and hover color change.

4. As for author plate, we set some blocks to show the author's name, and when your mouse on it, the recommendation will come out. The recommendation is the author's other papers.

5. As for conference plate, the plate shows the reference name of this paper, and when your mouse on it, the recommendation will come out. The recommendation is those papers also quoted the same paper. If the paper has no reference, we add a picture to point out it.

6. The visualization echarts are pie chart, bar chart and radar chart on the right side of this page.

1.Code of title

1. `hover::before`: This sets the hover-title gradient color, when you move your mouse on it, the text transform from left to right while off it the reversal. Each lasts for 0.4 seconds. `Attr()` function insert the content to cover the original title.

2. `hover::after`: This sets the gradient white line, similar to the text gradient color. And all the animations just come from x-axis 3d translated.

2.Code of guide

In Figure 31, we set the menu bar of the papers page. From line 175 to line 211 we create the menu bar and the color change when click or mouse over it. Then from line 6 to line 44, we use JavaScript to set three functions, to control the appear and disappear of the information and the color of the menu bar.

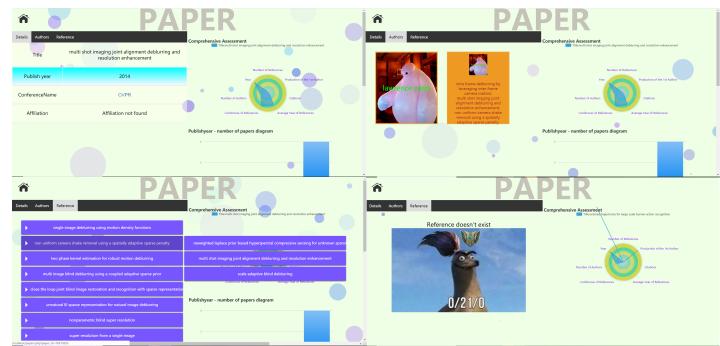


Figure 29: Our papers page.

```
for loading.js          for page_title.css      for se          for loading.js          for page_title.css      for se
1 .for_title {           for page_title.css      25 .for_title:after {           for page_title.css      26 .for_title:after {           for page_title.css      27 content: ''; position: absolute;
2 outline: none; text-decoration: none;           28 height: 16px; width: 100%;           28 top: 50%; margin-top: -8px;
3 position: relative; font-size: 8em;           29 right: 0; background: #F9F9F9;           29 -webkit-transform: translate3d(-100%, 0, 0);
4 line-height: 1; color: #9E9EAD;           30 -webkit-transition: translate3d(0, 100%, 0);           30 transform: translate3d(-100%, 0, 0);
5 display: inline-block; text-transform: uppercase;           31 -webkit-transition: translate3d(0, 100%, 0.4s);           31 transform: translate3d(0, 100%, 0.4s);
6 font-weight: bold; overflow: hidden;           32 -webkit-transition: -webkit-transform 0.4s;           32 transition: transform 0.4s;
7 line-height: 0.75; color: #5C5C8B;           33 transition-timing-function: cubic-bezier(0.7, 0, 0.3, 1);           33 transition-timing-function: cubic-bezier(0.7, 0, 0.3, 1);
8 }           34 }
9 .for_title:hover {           35 }
10 color: #C5C2B8;           36 }
11 }           37 }
12 }           38 }
13 .for_title:before {           39 .for_title:hover:after {
14 content: attr(data-letters); position: absolute;           40 -webkit-transform: translate3d(100%, 0, 0);           40 transform: translate3d(100%, 0, 0);
15 z-index: 2; overflow: hidden;           41 -webkit-transition: -webkit-transform 0.4s;           41 transform: translate3d(100%, 0, 0);
16 color: #A2A2A2; white-space: nowrap;           42 }
17 width: 0%;           43 }
18 -webkit-transition: width 0.4s 0.3s; transition: width 0.4s 0.3s;           44 }
19 }           45 }
20 .for_title:hover:before {           46 }
21 width: 100%;           47 }
22 }           48 }
```

Figure 30: Code of title.

```
6 <script>
7 function details() {
8     document.getElementById("det").style.display="inline";
9     document.getElementById("aut").style.display="none";
10    document.getElementById("ref").style.display="none";
11    document.getElementById("de").style.color="#ddd";
12    document.getElementById("de").style.backgroundColor="#fff";
13    document.getElementById("au").style.color="black";
14    document.getElementById("au").style.backgroundColor="#333";
15    document.getElementById("au").style.color="white";
16    document.getElementById("re").style.backgroundColor="#333";
17    document.getElementById("re").style.color="white";
18 }
19
20 function authors() {
21     document.getElementById("det").style.display="none";
22     document.getElementById("aut").style.display="inline";
23     document.getElementById("ref").style.display="none";
24     document.getElementById("de").style.backgroundColor="#333";
25     document.getElementById("de").style.color="white";
26     document.getElementById("au").style.backgroundColor="#ddd";
27     document.getElementById("au").style.color="black";
28     document.getElementById("re").style.backgroundColor="#333";
29     document.getElementById("re").style.color="white";
30 }
31
32 function reference() {
33     document.getElementById("det").style.display="none";
34     document.getElementById("aut").style.display="none";
35     document.getElementById("ref").style.display="inline";
36     document.getElementById("de").style.backgroundColor="#333";
37     document.getElementById("de").style.color="white";
38     document.getElementById("au").style.backgroundColor="#333";
39     document.getElementById("re").style.backgroundColor="#ddd";
40     document.getElementById("re").style.color="black";
41 }
42
43 window.onload=details();
44
45 </script>
```

Figure 31: Code of guide.

Papers

3.Code of details

table, tr, td: These set the table formatting as the half-transparent background and the color change when you move your mouse on it. I just set the top and bottom border to beautify them.

```
1  table {
2    font-size: 20px;
3    background-color: rgba(255,255,240,0.5);
4  }
5  tr,td {
6    padding: 1.25em;
7    border-bottom: 1px solid #ddd;
8    border-top: 1px solid #ddd;
9    text-align: center;
10 }
11 tr:hover {
12   background: linear-gradient(#00FFFF, #F8F8FF);
13 }
```

Figure 32: Code of details.

4.Code of author

In Figure 33, we build the author and the recommendation. Both the author's name and the recommendation are hyperlinks link to author page or paper page. From line 309 to line 380, we realize this by the css hover function and the parent and son elements. When you hover on it, we change the scale and the location of Y axis, then control the recommendation to move up from down.

```
309 /* effect-au.css */
310 nav {
311   margin-top: 40px;
312   margin-left: 40px;
313 }
314 .loadq {
315   margin: 0; padding: 0; box-sizing: border-box;
316   position: absolute;
317   width: 100px;
318   height: 30px;
319   font-size: 30px;
320   position: relative;
321   display: flex;
322   align-items: center;
323   justify-content: center;
324   background-image: url('au.gif');
325   background-size: 100% 100%;
326   color: green;
327 }
328 .loadq :link{color:#00FFFF;text-decoration:none}
329 .loadq :visited{color:#000;text-decoration:none}
330 .text-desc {
331   float: left;
332   width: 270px;
333   margin-top: 40px;
334   margin-left: 40px;
335 }
336 .text-desc{
337   position: absolute;
338   left: 0;
339   top: 0;
340   width: 16px;
341   background-color: #2e6b2e;
342   height: 100px;
343   opacity: 0;
344   width: 100px;
345   padding: 200px;
346 }
347 .text-desc a:link{color:#A9A9A9;text-decoration:none}
348 .text-desc a:visited{color:#A52A2A;text-decoration: none}
349 .text-desc .port{
350   float: left;
351   width: 100px;
352   position: relative;
353   overflow: hidden;
354   text-align: center;
355   border: 4px solid rgba(238,153,34, 0.9);
356   overflow: visible;
357 }
358 .port.effect .loadq{
359   transition: 0.5s;
360 }
361 .port.effect:hover .loadq{
362   transform: scale(0.3) translateY(-110%);
363   position: relative;
364   z-index: 9;
365 }
366 .port.effect .text-desc{
367   transform: translate(100%);
368   opacity: 0;
369   padding: 125px 20px 10px;
370   transition: 0.5s;
371 }
372 .port.effect:hover .text-desc{
373   transform: translate(0px);
374   opacity: 1;
375 }
376 /* effect-au.css end */
```

Figure 33: Code of author.

5.Code of reference

In Figure 34, we build the reference and the recommendation. Both the reference name and the recommendation are hyperlinks link to the paper page. From line 220 to line 307, we realize this by the css hover function and the parent and son elements. When you hover on it, we change the color of the block to turn dark, then control the recommendation to move left from right.

```
220 /* 书签显示 */
221 nav menutitle {
222   position: relative;
223   display: block;
224   opacity: 0.5;
225   cursor: pointer;
226 }
227 /* 并列菜单定位 */
228 nav menutitle > menu {
229   position: absolute;
230   pointer-events: none;
231   top: 0;
232   left: 0;
233   width: 100px;
234 }
235 /* 菜单显示 */
236 nav > menu { display: -webkit-box; display: -ms-flexbox; display: flex; }
237 /* 菜单初始显示 */
238 nav > menu > menutitle { pointer-events: all; opacity: 1; }
239 /* 菜单向上下跳动 */
240 nav > menu > menutitle a { white-space: nowrap; display: block; }
241 /* 菜单点击 */
242 menutitle:hover > menu {
243   pointer-events: initial;
244 }
245 /* 菜单悬停 */
246 menutitle:hover > menuitem,
247 menutitle:hover > menuitem {
248   opacity: 0.7;
249 }
250 }
251 }
252 /* 菜单名字 */
253 nav > menu > menutitle menu {
254   transform: translateX(100%);
255   top: 0; right: 0;
256 }
257 /* 菜单定位 */
258 nav {
259   margin-top: 40px;
260   margin-left: 40px;
261 }
262 /* 菜单框属性 */
263 nav a {
264   background-color: transparent;
265   color: #fff;
266   width: 100px;
267   height: 30px;
268   box-sizing: border-box;
269   border-radius: 3px;
270   box-shadow: 0px 2px 4px rgba(0, 0, 0, 0.5);
271   position: relative;
272   margin: 0px 6px 0px 0px;
273   padding: 2px 40px;
274   text-align: center;
275   box-sizing: border-box;
276   border-radius: 3px;
277   border: 1px solid transparent;
278   border-left: 10px solid white;
279   top: 22px;
280   left: 22px;
281   width: 100px;
282   height: 100px;
283   transition: opacity 0.6, transform 0.6s;
284   position: absolute;
285   width: 100px;
286   height: 100px;
287 }
288 /* 右侧显示 */
289 nav > menu > menutitle > a + menu:after {
290   content: '';
291   position: absolute;
292   border: 0px solid transparent;
293   border-left: 10px solid white;
294   top: 22px;
295   left: 22px;
296   width: 100px;
297   height: 100px;
298   transition: opacity 0.6, transform 0.6s;
299 }
300 /* 左侧打开 */
301 nav > menu > menutitle > menu {
302   transform: translate(0px) translateY(0px);
303   opacity: 0;
304 }
305 nav > menu > menutitle: hover > menu {
306   transform: translateX(0px) translateY(0px);
307   opacity: 1;
308 }
```

Figure 34: Code of reference.

Author

This is the author page, and all the codes and effects have been discussed in detail before, namely, the hover-animation title, the menu bar, the style of the major content, the pie and bar charts, and the bubbling background.

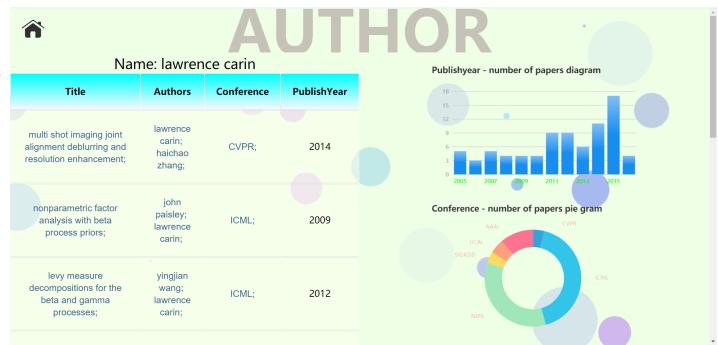


Figure 35: Our author page.

Conference

This is the conference page, and all the codes and effects have been discussed in detail before, namely, the hover-animation title, the style of the major content, the bar and radar charts, and the bubbling background.

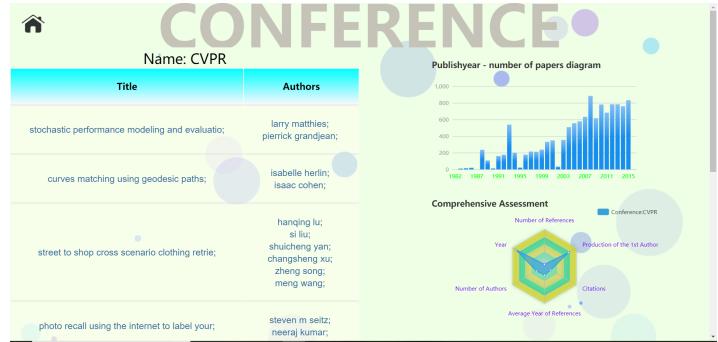


Figure 36: Our conference page.

Conclusion of our report

In a nutshell, we benefit a lot from this team work. In some ways, it is hard to differentiate the respective work of each member since we work together, debug together and remain depressed together. But we finally made it with tremendous entertainment, satisfaction and skills. Sometimes it is not the single work but the code splicing that trapped us. We do hope we can make a big difference through this team work.



Figure 37: Our Team.