

Our Own Web Page – Final Project of EE101

Mingjie Li F1703014, ID: 517030910344
 Mengtian Zhang F1703015, ID: 517030910387
 Pingyue Zhang F1703015, ID: 517030910388
 Zijie Zhu F1703015, ID: 517030910389

Abstract

This is the final project of EE101. It is a academic search website featuring different types of searching (author, paper, conference) and dynamic visualizations making it a user-friendly website.

The establishment of this website is done by 4 people – Zhang Mengtian whose work is mainly focused on basic part of this website, Li Mingjie whose work is the visualization part, paper recommendation and the debugging job, Zhang Pingyue whose work is paper recommendation based on word analysis, tree graphs and query optimization and Zhu Zijie whose work is to beautify the web page.

I. INTRODUCTION

- Section II – Basic Part
- Section III – Paper Recommendation according to Citation
- Section IV – Recommendation via Similar Titles
- Section V – Visualization
- Section VI – Query Optimization
- Section VII – Beautification
- **Note: At the end of this report, all pages will be shown there.**

II. BASIC PART

In basic part,there are 9 pages as the main part.
 And here are the relationship of the pages.

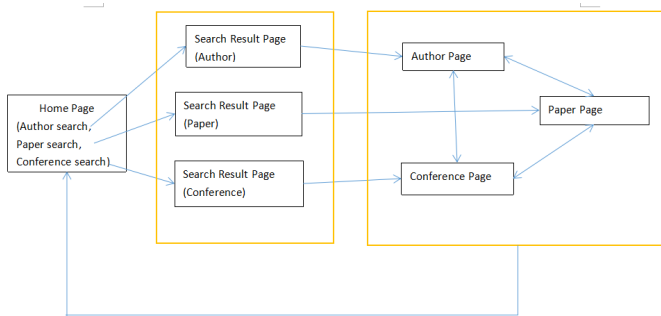


Fig. 1. Page Relationship.

A. Implementation

In our project,we have 9 pages.And I divide them into 3 parts.

- I:Home part(Author Home page,Paper Home page,Conference Home page).
- II:Search result part(Author Search Result page,Author Search Result page,Author Search Result page).
- III:main result part(Author page,Paper page,Conference page) .

That is,in every part there are 3 pages.And every page we use a php file.

- I:Home part(Author Home page,Paper Home page,Conference Home page).



Fig. 2. Author home.



Fig. 3. Paper home.



Fig. 4. Conference home.

In the formal lab, we have finished the author home page. Here is the source code about search box. And it submit the author name to the "author_result.php".

```

1  <form action="author_result.php"
2  method="get">
3  <input type="text"
4  name="AuthorName"
5  id = "AuthorName"
6  value=""
7  style="width:700px;
8  height:45px;
9  font-size:28px;
10 background:#ffffff;;
11 border:#3385ff solid;" >
12 <input type="submit"
13 value="SUBMIT"
14 style="width:90px;
15 height:45px;
16 font-size:20px;
17 background:#3385ff;
18 color:#ffffff;
19 border:0px" >
20 </form></h1>
21

```

And here is the code about Automatic completion.

```

1  <link rel="stylesheet" href=
2  "//code.jquery.com/ui/1.11.4/
3  themes/smoothness/jquery-ui.css">
4  <link rel="stylesheet"
5  href="css/home_style.css">
6  <script
7  src="//code.jquery.com/jquery-1.10.2.js"
8  >
9  </script>
10 <script
11 src="//code.jquery.com/ui/1.11.4/jquery-
12 ui.js">
13 </script>
14 <script>
15 $(document).ready(function() {
16   $( "#AuthorName" ).autocomplete({
17     source: 'author_hint.php',
18     minLength: 1
19   });
20 }
21 </script>
22

```

In this part, we connect it with 'author_hint.php'.

```

1  <?php
2  $dbHost = 'localhost';
3  $dbUsername = 'root';
4  $dbPassword = '';
5  $dbName = 'lab1';
6  $conn = new mysqli($dbHost,$dbUsername,
7  $dbPassword,
8  $dbName);
9  $searchTerm = $_GET['term'];
10 $sql = "SELECT tb_authors.AuthorName,
11         count(*)
12 AS PaperNumber ".
13 "FROM tb_authors INNER JOIN
14 tb_paper_author_affiliation ".
15 "ON tb_authors.AuthorID =
16 tb_paper_author_affiliation.AuthorID ".
17 "WHERE tb_authors.AuthorName LIKE
18 "'. $searchTerm . '%'. ' " ".
19 "GROUP BY tb_authors.AuthorID ".
20 "ORDER BY PaperNumber DESC,
21 tb_authors.AuthorName limit 10;";
22 $query = $conn->query($sql);
23 while ($row = $query->fetch_assoc()) {
24   $data[] = $row['AuthorName'];
25 }
26 echo json_encode($data);
27 ?>
28

```

And also we have a connection among 3 home pages. By clicking the picture above the search box, we can change the page so that we can search different content.

```

1  <div class="dropdown">
2  
6  <div class="dropdown-content">
7  <!--a href="#"></a-->
11 <a href="home_paper.php">
12 </a>
16 <a href="home_conference.php">
17 </a>
21
22 </div>
23 </div>
24

```

- II: Search result part (Author Search Result page, Author Search Result page, Author Search Result page).


```

67  },
68  function(data){
69      json = eval(data);
70      if (json.length !== 0){
71          display(json,"result");
72          pagenum++;
73      }
74      else{
75          alert("Already the last page!");
76      }
77  });
78  });
79  });
80  </script>
81

```

- III:main result part(Author page,Paper page,Conference page).

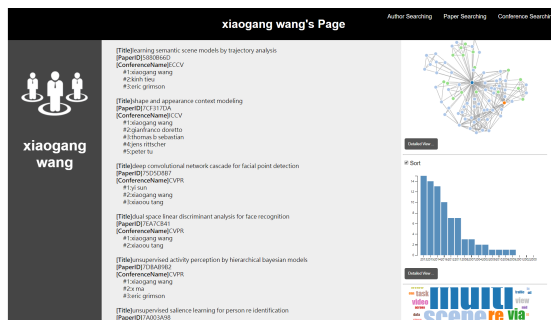


Fig. 8. Author.

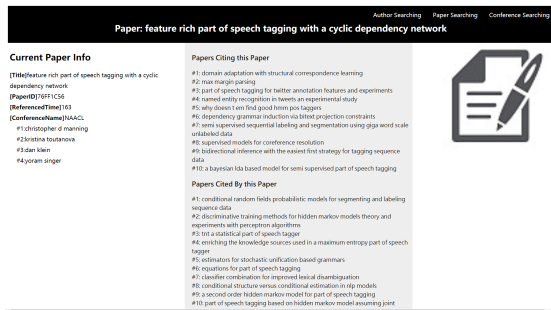


Fig. 9. Paper.

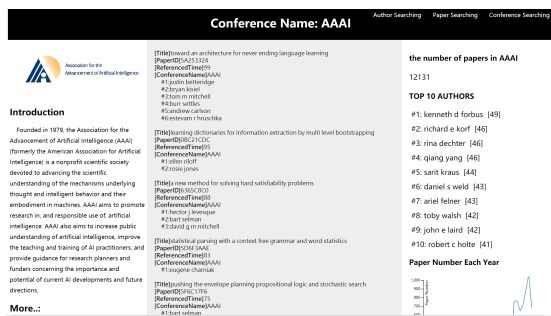


Fig. 10. Conference.

In the author page,the head part is author name;the main part is information about the author's papers;the right is visualization and it's my partner's work.

In the paper page,the head is paper name;the left is information about the paper;the center is recommendation and it's my partner's work.

In the conference page,the head is conference name;the left is introduction and link of the conference;the center is the papers of the conference;the right is top 10 authors of the conference.

Head part:

```

1  <div id="header">
2  <?php $authname = $_GET["AuthorName"];>
3  <h1>
4  <?php echo $authname."<s Page";?><br>
5  </h1>
6  <ul>
7  <!--li><a class="active" href="#home">
8
9  </a></li-->
10 <li><a> </a></li>
11 <li><a href="home_conference.php"
12 class="two">Conference Searching</a></li>
13 <li><a href="home_paper.php" class="
14 two">
15 Paper Searching</a></li>
16 <li><a href="home_author.php" class="
17 two">
18 Author Searching</a></li>
19 </ul>
</div>

```

Left part:

```

1  <div id="left">
2  <br><br>
3  <?php $authname = $_GET["AuthorName"];>
4  <!-- --><?php
5  // $affiliationname=$_GET["
6  AffiliationName"];
7  ?>
8  
9  <h1 style="font-size: 40px;">
10 <?php echo $authname;?><br><br>
11 </h1>
12 <!-- <h2>-->
13 <?php //echo $affiliationname;?>
14 <!--<br></h2>-->
15 </div>

```

Center part:

```

1  <div id="section">
2  <script src="js/author_main.js"></script>
3  <p id = "author"></p>
4  <button id = "previous">Previous</button>
5  <button id = "next">Next</button>
6  </div>
7

```

III. PAPER RECOMMENDATION ACCORDING TO CITATION

The most simple two criterion of paper recommendation is recommending according to citation papers citing the current paper and papers cited by the current paper. This part can be achieved using simple SQL query and the result is shown below. And the code in paper.php is also shown below.

Papers Citing this Paper
#1: knowledge based weak supervision for information extraction of overlapping relations
#2: improved part of speech tagging for online conversational text with word clusters
#3: semantic parsing on freebase from question answer pairs
#4: open language learning for information extraction
#5: large scale semantic parsing via schema matching and lexicon extension
#6: learning first order horn clauses from web text
#7: patty a taxonomy of relational patterns with semantic types
#8: random walk inference and learning in a large scale knowledge base
#9: open information extraction the second generation
#10: constrained semi supervised learning using attributes and comparative attributes
Papers Cited By this Paper
#1: unsupervised word sense disambiguation rivaling supervised methods
#2: learning dictionaries for information extraction by multi level bootstrapping
#3: unsupervised models for named entity classification
#4: guiding semi supervision with constraint driven learning
#5: a probabilistic model of redundancy in information extraction
#6: names and similarities on the web fact extraction in the fast lane
#7: counter training in discovery of semantic patterns
#8: methods for domain independent information extraction from the web an experimental comparison
#9: active learning by labeling features
#10: entity extraction via ensemble semantics

Fig. 11. Recommendation according to citation

```

1 <div id="section">
2
3   <script src="js/papers_citing_him.js">
4   </script>
5   <h3>
6     Papers Citing this Paper
7   </h3>
8   <p class="big" id="citehim"></p>
9
10  <script src="js/papers_cited_by_him.js">
11  </script>
12  <h3>
13    Papers Cited By this Paper
14  </h3>
15  <p class="big" id="citebyhim"></p>
16
17 </div>

```

IV. RECOMMENDATION VIA SIMILAR TITLES

There are several alternative methods to recommend papers, e.g., by author name or by subjects (title, key words, abstract, etc.). As I think the latter (by subjects) is more reasonable and effective, I decide to realize this feature in the project. Since the key words and abstract are not available in the database, only the paper title is used in this version. But the proposed method can also be extended to include key words, abstract and even the full text of a paper.

Two problems arise when I implement the feature: (1) how to measure the similarity of two paper? (2) how to improve the performance when there are a huge number of papers?

A. Implement

To measure the similarity, I use a method called **Simhash**, which was proposed by Google to detect the similar websites and eliminate them. Two important concepts are summarized as below.

- **Locality-Sensitive Hashing:** For two similar strings, this method will generate similar sequences, unlike the ordinary hash, which will generate a totally different sequence to avoid hash collision.
- **Hamming Distance:** To transform a string consisted of 01 to another string with the same length, how many characters should be changed? The number of characters needed to be changed is the Hamming Distance between two strings.

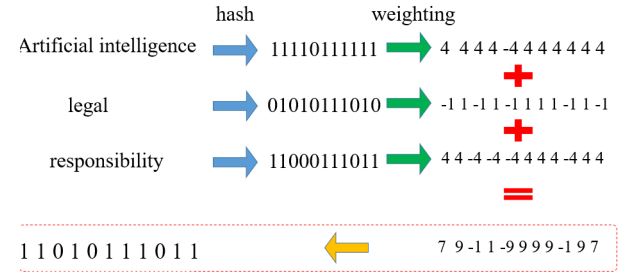


Fig. 12. Simhash implement.

Simhash can be separated into five steps, as illustrated in Fig.12.

- I: Splitting the words of a sentence. And words like a, an, the will be eliminated.
- II: Turning words into hash value, no matter how long a word is, it will be generated into a string which is 64 bits consisting only 0 and 1.
- III: Giving words weight, my method is to give more weight if the word is long, for generally speaking, the longer the word is, the more possible it could be a relatively important words. Zero will be -1 in this step.
- IV: Sum up all the words in a sentence bit by bit.
- V: Turing back to the 0 and 1 sequence.

B. Performance improvement

To deal with cases with a large number of papers, the following measures are taken to improve the performance:

- **Re-design the database:** I establish a new table named simhash in my database (The table Papers is altered by adding a new attribute called hash). A screenshot of the table is shown in Fig.13.
- For each paper, the Simhash value of the paper title is pre-generated and stored in the database. Thus, when finding similar papers, only the method of Hamming distance is called which is pretty fast. The typical searching time is in this project.

To limit the number of results, the threshold of the Hamming distance is set 10. In other words, given a paper, all the papers with a Hamming Distance (evaluated

by the paper title) less than 10 will be recommended as similar papers.

PaperID	Title	hash
00002E77	speech training systems using lateral shapes of ...	0001000001101111000000110111110000100...
00003F20	logical derivation of a prolog interpreter	11110011011011111100110111001010101...
000073E1	proceedings of the 6th annual international acm...	000100100100011100100001011001011110...
00009805	automated object identification and position esti...	01010101000100101010000110101010010...
0000E395	a general semantic analyzer for data base access	0000111010010001010101010110111011010...
0000EB9E	intelligent printing technique recognition and oh...	10000100000010110100001001000010101...
0000F1CF	creating image based virtual reality using a self...	10111110111111010011111000011000001110...
0000F908	computational challenges in propositional reaso...	01011010111101010101011110000011011101...
000104EF	strategic manipulation in iterative auctions prox...	1000011000001101111110000011101101001...
000127EC	learning of abstractions from structural descripti...	10111011101110111101110100100110011010...
000215D4	interactive web search by graphical query refin...	101001100100111111111010001010010101...
000238D0	diagrams as scaffolds for creativity	01011001100101011111111011100100000010...
000275F5	traffic observation and situation assessment	1110111000101100000000000110110111100...
00027620	using pattern action rules for the generation of ...	101100101110101110000111101001101010...
000279D0	von mises fisher clustering models	110010111110100110100000110010010101...
00027CFD	real time multi view face detection tracking pose...	00010100000101010000001001010010101...
000282C5	search lessons learned from crossword puzzles	110110110100110100100100100110110011...
000285C2	conceptual indexing practical large scale ai for e...	1011110010010011011011110111111011110...
0002C94D	encoding learning groups using social interacto...	10000100011111001011011110100010011001...
0002E110	automated identification of thoracolumbar verte...	1100010101001101111101111010001001000...
0002EFED	event based video analysis	01001111101110001100011011100101011001...
000370F3	adaptive localization in a dynamic wifi environm...	01000001111000110011001011100110011100...

Fig. 13. Simhash table.

C. Possible improvements

- The weighting algorithm can be improved, e.g., the keywords should be assigned a high weight.
- If we have the abstract of the paper, this proposed method might be much more effective.
- More advanced artificial intelligence technology may be employed to enhance the performance.

D. Codes

```

1 dele=["a",'an','for','then',
2 'the','of','as','because','at',
3 'in','on','so','while','when',
4 'with','to','is','are','therefore',
5 'however','through','and']
6 def gethash(a):
7     final=[]
8     r=[]
9     for mn in range(64):
10         final.append(0)
11         r.append(0)
12     ll=a.split(" ")
13     for i in ll:
14         result=[]
15         weight=1
16         for mn in range(64):
17             result.append(0)
18         if (i) in dele: continue
19         if (len(i))>=8 : weight=3
20         if (len(i))>=10 : weight=4
21         if (len(i))>=15 : weight=5
22         al=string_hash(i)
23         for j in range(64):
24             if (str(al[j])=='0'):
25                 result[j]=(-1)*weight
26             else:
27                 result[j]=(1)*weight
28         for l in range(len(result)):
29             r[l]+=result[l]
30     for m in range(len(r)):
31         if (r[m]<0):
32             final[m]=0
33         else:
34             final[m]=1
35     a=''
```

```

36 for c in final:
37     a+=str(c)
38     return a
39 import pymysql
40
41 if __name__ == '__main__':
42
43     import pymysql
44     connection=pymysql.connect(host='localhost'
45     ,user='root',password='',db='test',
46     charset='utf8'
47     ,port=3306,cursorclass=pymysql.cursors.
48         Cursor)
49     cursor=connection.cursor()
50     cursor.execute("DROP TABLE IF EXISTS simhash
51         ")
52     j=0
53     sql_2='''CREATE TABLE simhash(
54     PaperID VARCHAR(20) NOT NULL,
55     Title VARCHAR(310) NOT NULL,
56     hash VARCHAR(12) NOT NULL,
57     PRIMARY KEY (PaperID)
58     )default charset=utf8;
59     '''
60     cursor.execute(sql_2)
61     connection.commit()
62     cursor.execute("select PaperID,
63     Title from paper")
64     results = cursor.fetchall()
65     for r in results:
66         id=r[0]
67         st=r[1]
68         sl=st.split()
69         s=[]
70         for i in sl:
71             if i in dele:
72                 continue
73             s.append(i)
74             hash = simhash(s)
75             ins_2 = '''INSERT INTO simhash(
76             PaperID,Title,hash)
77             VALUES (%s,%s,%s)'''
78             cursor.execute(ins_2,(id,st,str(hash)))
79     connection.commit()
```

V. VISUALIZATION

A. Improvements on teacher-student graph

Since every node in this graph represents an author, I considered adding onclick events to each node so that the website will jump to another authors page when clicking the node. In this way, it will be more convenient for users to visit the page of a certain author in the graph.

The code is shown below. Just use the on() function to monitor the event on each node.

```

1 node.on("click",function(d){
2     window.open("author.php?AuthorID="
3     +d.id.slice(0,8)+"&AuthorName="
4     +d.id.slice(9));
5 });
6
```



```

1 // js/word_cloud.js
2
3 var fill = d3.scaleOrdinal(d3.schemeCategory20
4 );
5
6 $(document).ready(function() {
7   $.post("word_search.php",
8     {
9       id: AuthorID
10    },
11    function() {
12      d3.json("word.json", function(list) {
13        d3.layout.cloud().size([360, 225])
14          .words(list)
15          .rotate(0)
16          .font("Impact")
17          .fontSize(function(d) { return d.size; })
18          .on("end", draw)
19          .start();
20
21      function draw(words) {
22        d3.select("#word_cloud").append("svg")
23          .attr("width", "100%")
24          .attr("height", "100%")
25          .append("g")
26          .attr("transform", "translate(187,75)")
27          .selectAll("text")
28          .data(words)
29          .enter().append("text")
30          .style("border", "1px solid blue")
31          .style("font-size", function(d) { return
32            d.size + "px"; })
33          .style("font-family", "Impact")
34          .style("fill", function(d, i) { return
35            fill(i); })
36          .attr("text-anchor", "middle")
37          .attr("transform", function(d) {
38            return "translate(" + [d.x, d.y] + ")";
39          })
40          .text(function(d) { return d.text; });
41      });
42    });
43

```

D. Line chart to show the trend of papers published on a conference

It's a good way to use a line chart to show the trend of papers published in a conference. In this way, the number of papers published on this conference can be shown to users clearly.

The code of the searching part is in the package `each_year_search.php`.

The code of the implementation of the chart is shown below.

```

1 // js/conference_line_chart.js
2
3 var svg = d3.select("#PaperNumEachYear"),
4   margin = {top: 10, right: 10, bottom: 20,
5     left: 40},
6   width = +svg.attr("width") - margin.left -
7     margin.right,
8   height = +svg.attr("height") - margin.top -
9     margin.bottom,

```

Paper Number Each Year

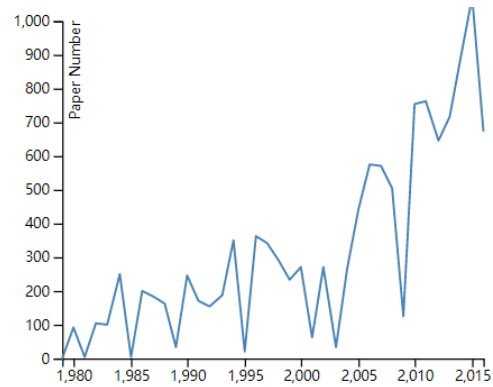


Fig. 18. Line Chart

```

7   g = svg.append("g").attr("transform", "
8     translate(" + margin.left + "," + margin.
9     top + ")");
10
11 var x = d3.scaleLinear()
12   .rangeRound([0, width]);
13
14 var y = d3.scaleLinear()
15   .rangeRound([height, 0]);
16
17 var line = d3.line()
18   .x(function(d) { return x(d.year); })
19   .y(function(d) { return y(d.n); });
20
21 $(document).ready(function() {
22   $.post("each_year_search.php",
23     {
24       name: ConferenceName
25     },
26     function() {
27       d3.json("data_linechart.json", function(
28         error, data) {
29         if (error) throw error;
30
31         x.domain(d3.extent(data, function(d) {
32           return d.year; }));
33         y.domain([0, 1000]);
34
35         g.append("g")
36           .attr("transform", "translate(0," +
37             height + ")")
38           .call(d3.axisBottom(x))
39           .append("text")
40           .attr("fill", "#000")
41           .select(".domain")
42           .remove();
43
44         g.append("g")
45           .call(d3.axisLeft(y))
46           .append("text")
47           .attr("fill", "#000")
48           .attr("transform", "rotate(-90)")
49           .attr("y", 6)
50           .attr("dy", "0.71em")
51           .attr("text-anchor", "end")
52           .text("Paper Number");

```



```

49 g.append("path")
50 .datum(data)
51 .attr("fill", "none")
52 .attr("stroke", "steelblue")
53 .attr("stroke-linejoin", "round")
54 .attr("stroke-linecap", "round")
55 .attr("stroke-width", 1.5)
56 .attr("d", line);
57 });
58 });
59 });
60 });
61 });

```

E. Other visualizations in the conference page

- The number of all papers published on this conference
This part can be easily achieved using simple SQL query.
- Top 10 authors on this conference

The most active 10 authors on this conference and the number of their papers are shown on the right side of the conference page.

the number of papers in AAAI

12131

TOP 10 AUTHORS

- #1: kenneth d forbus [49]
- #2: richard e korf [46]
- #3: rina dechter [46]
- #4: qiang yang [46]
- #5: sarit kraus [44]
- #6: daniel s weld [43]
- #7: ariel felner [43]
- #8: toby walsh [42]
- #9: john e laird [42]
- #10: robert c holte [41]

Fig. 19. Other visualizations

The searching code of the two is `conference_paper_num_search.php` and `most_paper_author_search.php` respectively.

And their implementation code is `js/conference_paper_num.js` and `js/conference_top_authors.js` respectively.

The code of visualizations in the main page of conference is shown below.

```

1 <div id="rightSection">
2   <script src="js/conference_paper_num.js"></script>
3   <h4>

```

```

4   the number of papers in <?php echo
5   $confname ?>
6   </h4>
7   <p id="papernum"></p>
8   <script src="js/conference_top_authors.js"></script>
9   <h4>
10    TOP 10 AUTHORS
11  </h4>
12  <p id="mostpapers"></p>
13
14  <h4>
15    Paper Number Each Year
16  </h4>
17  <svg id="PaperNumEachYear" width="350"
18    height="270"></svg>
19  <script src="js/conference_line_chart.js"></script>
20 </div>

```

F. Teacher-student tree graph and knowledge graph

Teacher-student Tree Graph: The Graph is a two-way tree. It can be fold or unfold if you click the nodes.

- Red: The author himself / herself.
- Green: His / her possible students.
- Blue: Students' possible students.
- Purple: His / her possible teachers.

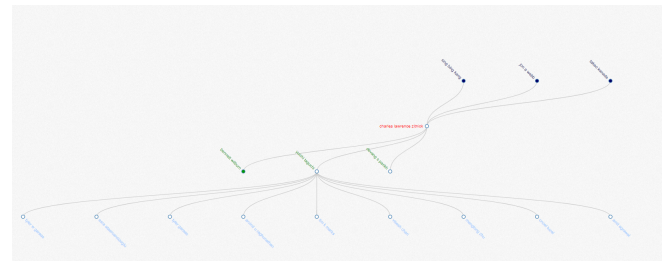


Fig. 20. Teacher_student graph.

```

1 <?php
2 $con = mysqli_connect("localhost","3306","");
3 if (!$con)
4 {
5     die('Could not connect: ' . mysqli_error());
6 }
7
8 mysqli_select_db($con,"test");
9
10 $q=$_GET['AuthorID'];
11 $q1=mysqli_query($con,"SELECT AuthorName
12 FROM coauthor WHERE AuthorID='$q' limit 1");
13 $query=mysqli_query($con,"SELECT Relation,
14 AuthorName,AuthorID,CoName,CoID FROM coauthor
15 WHERE AuthorID='$q' and Relation='1'
16 limit 30");
17 $query1=mysqli_query($con,
18 "SELECT Relation,AuthorName,AuthorID,CoName,
19 CoID FROM coauthor WHERE AuthorID='$q'
20 and Relation='-1' limit 30");
21 $row1 = mysqli_fetch_array($q1);
22 $data=array();

```

```

23 $n= $row1["AuthorName"];
24 $data['name']=$n;
25 $data['g']=1;
26 // $data['size']=100;
27 $data['children']=array();
28 $data['parent']=array();
29
30 while ($row = mysqli_fetch_array($query)) {
31     $children=array();
32     $children['name']=$row['CoName'];
33     $children['isparent']=false;
34     $children['children']=array();
35     $children['g']=3;
36     $children['size']=10;
37     $id1=$row['CoID'];
38     $q2 =mysqli_query($con,"SELECT Relation,
39     AuthorName,AuthorID,CoName,CoID FROM
40     coauthor WHERE AuthorID='$id1' and
41     Relation='1'limit 30");
42     while ($row1 = mysqli_fetch_array($q2)) {
43         $ch1=array();
44         $ch1['name']=$row1['CoName'];
45         $ch1['isparent']=false;
46
47         $ch1['g']=9;
48
49         $ch1['size']=1;
50         array_push($children['children'],$ch1);
51     }
52
53     array_push($data['children'],$children);}
54
55 while ($row = mysqli_fetch_array($query1)) {
56     $children=array();
57     $children['name']=$row['CoName'];
58     $children['isparent']=true;
59     $children['children']=array();
60     $children['g']=7;
61     $children['size']=10;
62     $id1=$row['CoID'];
63     $q2 =mysqli_query($con,"SELECT Relation,
64     AuthorName,AuthorID,CoName,CoID
65     FROM coauthor WHERE AuthorID='$id1' and
66     Relation='1'limit 30");
67     while ($row1 = mysqli_fetch_array($q2)) {
68         $ch1=array();
69         $ch1['name']=$row1['CoName'];
70         $ch1['isparent']=false;
71         if ($row1['CoID']===$q) $ch1['g']=1;
72         else $ch1['g']=5;
73         $ch1['children']=array();
74         $id2=$row1['CoID'];
75         $q3 =mysqli_query($con,"SELECT Relation,
76     AuthorName,AuthorID,CoName,CoID FROM
77     coauthor WHERE AuthorID='$id2' and
78     Relation='1'limit 30");
79         while ($row2 = mysqli_fetch_array($q3)) {
80             $ch2=array();
81             $ch2['name']=$row2['CoName'];
82             $ch2['isparent']=false;
83             $ch2['g']=3;
84             $ch2['size']=10;
85             $ch1['size']=1;
86             array_push($ch1['children'],$ch2);
87         }
88         array_push($children['children'],$ch1);
89     }
90 }
91 array_push($data['children'],$children);}

```

```

92 echo json_encode($data);
93 ?>
94

```

The codes which generate the graph occupy tremendous space so I won't display it in my report.

Knowledge Graph: The Graph is to illustrate the knowledge flow. Papers (knowledge) flow into a paper, then generate new papers (knowledge). The nodes below the paper node are papers which referenced this essay, and the nodes above it are papers it referenced. Title will be shown if you move the mouse over the nodes.

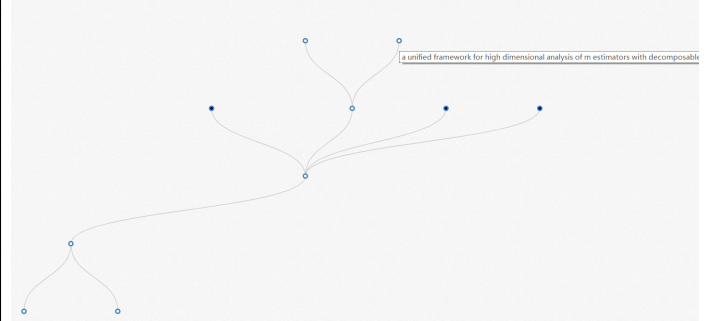


Fig. 21. Knowledge graph.

The codes are similar to the codes of the teacher-student graph. So I won't display them in my report.

VI. QUERY OPTIMIZATION

I found that query optimization plays a very important role in the performance improvements of the whole system. For example, after adding proper indices to the database, the query speed could be 20 times faster than before, as shown in Fig. 22.

0.312 sec
0.016 sec

Fig. 22. index 3 (before / after optimization).

Several rules of thumb are summarized according to many tests in the project, which are described one by one as below.

RULE 1 - single-column index:

Index should be created on the column which appears in the where sentence. But when "like %..." appears in the where sentence, then index is unnecessary.

Select * from paper; — Just take Title from database directly.
Select * from paper where **paperid**= '0029E503'; — Takes more time.

Index

Select * from paper where **paperid** like '%123';

Index is unnecessary

Fig. 23. sample index 1.

RULE 2 - multiple-column index:

For index including multiple columns, the column order is important. For example, suppose there is a multiple-column index named `Paper_index` which consists “paperid” and “title”, and “paperid” appears in front of “title” in the index. Then “paperid” should also appear in front of “title” in the “where” conditions.

Furthermore, if the “where” sentence contains only “paperid”, an separate single-column index on “paperid” is unnecessary, i.e., a query on “paperid” can share the multiple-column index. However, for a query on “title” only, a separate single-column index is needed to speed the query.

```
Select * from paper where paperid= '0029E503' and title='.....';
```

`Paper_index(#1)` `Paper_index(#2)`

```
Select * from paper where paperid= '0029E503';
```

Extra index unnecessary

```
Select * from paper where title='.....';
```

A new isolated-column index is needed

Fig. 24. sample index 2.

RULE 3 - use index only when necessary:

While the index will significantly improve the query performance, it will also lower the writing speed of the database notably. So the index shouldn't appear unless its necessary. As a rule of thumb, the index should be created in the optimization phase, not in the very beginning of the database design.

VII. BEAUTIFY THE PAGES

A. Job as a beautifier

A qualified beautifier should not only provide a good-looking page design, but also try to make his website used comfortable.

To accomplish this we designed Icon switching, navigation bar, a lot of hyperlink etc. to provide our users the best experience.

Whats more, insisting not to use a model, our teams page is beautification is purely handmade.

Some attribute frequently-used to set the style:

height, width to set size.

font-size, font-family, font to set fonts size and color etc.

position, float, padding, margin, border to set position of elements.

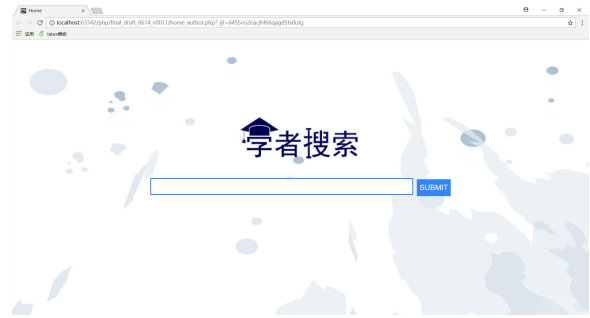
background, bgcolor to set background.

B. The home page

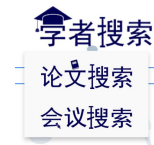
First, let me show you the final design sketch.

The logo

Here we designed a logo switch to change the searching type. Each logo is a .png file. Use `img` label and set the `src` (which means source) attribute to include a image. And we attached each image to a hyperlink to jump to another page. The method to put the images in a spinner is using `div` label to distribute an area and using css syntax to set its style. One thing deserves mentioned is the logos are designed and



made by ourselves using some software such as Photoshop and Illustration.

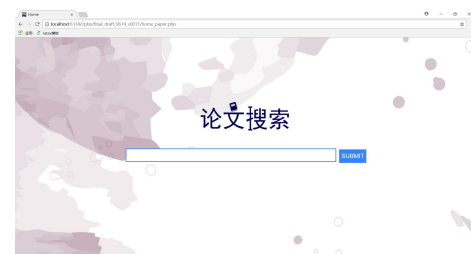
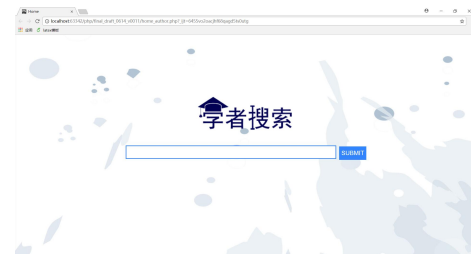


The text area

Here we imitate Baidus page a little. Just add some style attribute to set its format. Such as height, width, font-size, background and border.

```
style="width:700px;height:45px; font-size:28px;
background: #ffffff;;border: #3385ff
solid;"
```

Last, add background attribute in the body label. And we have three different bg in these three different home page.

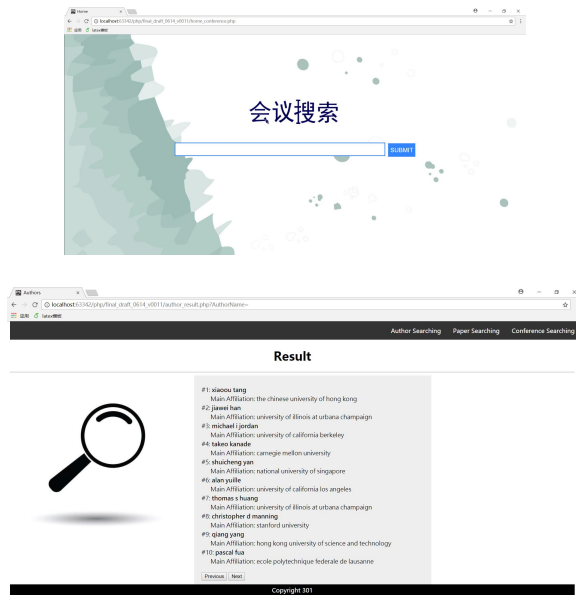


C. The result page

The result page of author is shown below.

the navigation bar

With this design, users can conveniently return the home page and do another new search. Its realized by using `ul` label to create a transverse list and set each element a hyperlink. We



can change the color of words with hyperlink to make it more pretty. Use float attribute to set the position of the bar.

Load animation



Before the searching result being displayed. Users always need to wait one or more seconds. During this time, a load animation can make things interesting.

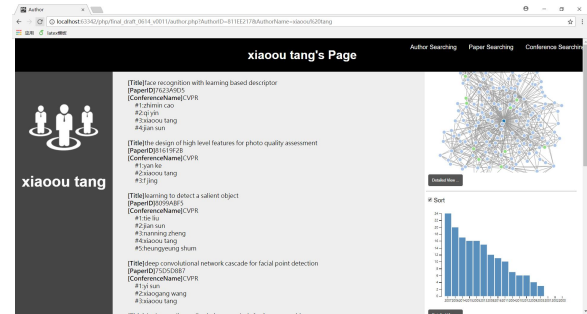
```

1 $(document).ready(function() {
2   document.getElementById("result").innerHTML='<
3     img height="20%" width="20%" src="img/
4     loading.gif">';
5   $.post("result_search.php",
6     {
7       name: AuthorName,
8       page: pagenum
9     },
10    function(data) {
11      json = eval(data);
12      display(json, "result");
13    });
14    $("#previous").click(function() {
15      if (pagenum === 1) {
16        alert("Already the first page!");
17      }
18      else {
19        $.post("result_search.php",
20          {
21            name: AuthorName,
22            page: pagenum-1
23          },
24          function(data) {
25            json = eval(data);
26            display(json, "result");
27            pagenum--;
28          });
29    });
30  });

```

Put the image address in the innerHTML attribute and its a .gif animation.

D. The author page and conference page



Things difficult here is format dividing. We need to put each part into its suitable position. We use `div` label to divide areas and set their style in a css file. Indicate the class attribute to distinguish each `div`.

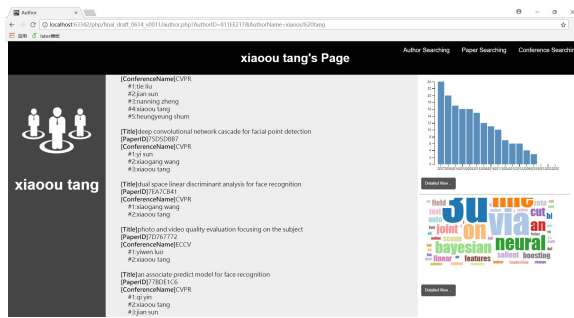
For example:

```

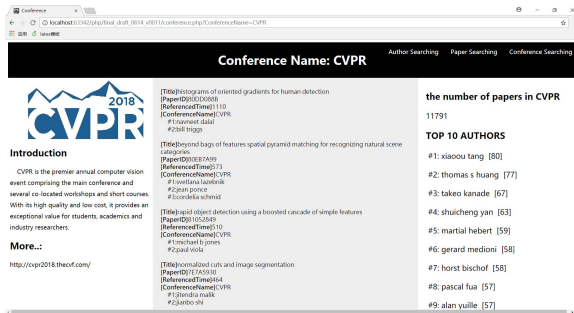
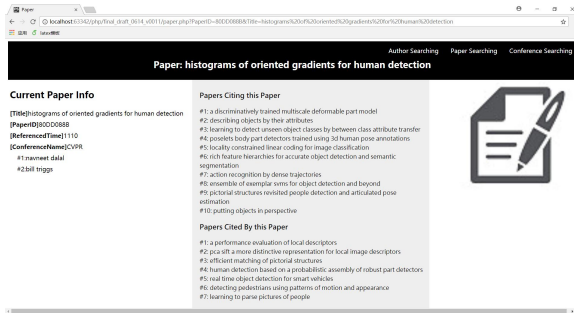
1 <div id="footer" >
2   Copyright 301
3 </div>
4
5 #footer {
6   background-color:black;
7   color:white;
8   clear:both;
9   text-align:center;
10  padding:5px;
11  /*bottom: 0;
12  position:fixed;*/
13  left:250px;
14  right:400px;
15  float:bottom;
16 }

```

Here we made the navigation bars position as fixed, then it can keep still at the top of the page not moving with the page scrolling.



Similarly here is the paper page and conference page.



We searched the logos of the conferences and put them in our page.






学者搜索

Author Searching Paper Searching Conference Searching

Result




- #1: xiaogang wang
Main Affiliation: the chinese university of hong kong
- #2: fei wang
Main Affiliation: ibm
- #3: jun wang
Main Affiliation: university college london
- #4: jingdong wang
Main Affiliation: microsoft
- #5: gang wang
Main Affiliation: university of illinois at urbana champaign
- #6: haifeng wang
Main Affiliation: toshiba
- #7: jun wang
Main Affiliation: ibm
- #8: hua wang
Main Affiliation: university of texas at arlington
- #9: jianrong wang
Main Affiliation: tsinghua university
- #10: lei wang
Main Affiliation: australian national university

Previous Next

Copyright 301

Author Searching Paper Searching Conference Searching

xiaogang wang's Page



xiaogang wang

[Title]learning semantic scene models by trajectory analysis
[PaperID]5800660
[ConferenceName]ECCV
#1:xiaogang wang
#2:kinh tieu
#3:eric grimson

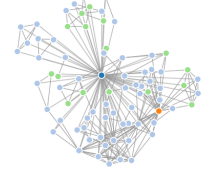
[Title]shape and appearance context modeling
[PaperID]7CF317DA
[ConferenceName]CCV
#1:xiaogang wang
#2:gianfranco doretto
#3:thomas b sebastian
#4:jens rittscher
#5:peter tu

[Title]deep convolutional network cascade for facial point detection
[PaperID]75D5DB87
[ConferenceName]CVPR
#1:yi sun
#2:xiaogang wang
#3:xiaoou tang

[Title]dual space linear discriminant analysis for face recognition
[PaperID]7EA7CB41
[ConferenceName]CVPR
#1:xiaogang wang
#2:xiaoou tang

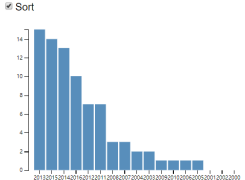
[Title]unsupervised activity perception by hierarchical bayesian models
[PaperID]7DBA89B2
[ConferenceName]CVPR
#1:xiaogang wang
#2:x ma
#3:eric grimson

[Title]unsupervised salience learning for person re-identification
[PaperID]7A003A98



Detailed View

Sort



Detailed View


task video scene data

re via



[Author Searching](#) [Paper Searching](#) [Conference Searching](#)

Result



#1: conditional random fields probabilistic models for segmenting and labeling sequence data
PaperID:80060D7C

#2: optimizing search engines using clickthrough data
PaperID:7A061E69

#3: imagenet a large scale hierarchical image database
PaperID:7DAEA7E3

#4: a database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics
PaperID:7E7ADAC6

#5: the weka data mining software an update
PaperID:7F321C50

#6: uci repository of machine learning databases
PaperID:7C43B6E0

#7: wordnet a lexical database for english
PaperID:7C126405

#8: accurately interpreting clickthrough data as implicit feedback
PaperID:8049B52A

#9: a density based algorithm for discovering clusters in large spatial databases with noise
PaperID:63BCB1FC

[Author Searching](#) [Paper Searching](#) [Conference Searching](#)

Paper: feature rich part of speech tagging with a cyclic dependency network

Current Paper Info

[Title]feature rich part of speech tagging with a cyclic dependency network
[PaperID]76FF1C56
[ReferencedTime]163
[ConferenceName]NAACL
#1:christopher d manning
#2:kristina toutanova
#3:dan klein
#4:yoram singer

Papers Citing this Paper

#1: domain adaptation with structural correspondence learning
#2: max margin parsing
#3: part of speech tagging for twitter annotation features and experiments
#4: named entity recognition in tweets an experimental study
#5: why doesn't em find good hmm pos taggers
#6: dependency grammar induction via bitext projection constraints
#7: semi supervised sequential labeling and segmentation using giga word scale unlabeled data
#8: supervised models for coreference resolution
#9: bidirectional inference with the easiest first strategy for tagging sequence data
#10: a bayesian lda based model for semi supervised part of speech tagging

Papers Cited By this Paper

#1: conditional random fields probabilistic models for segmenting and labeling sequence data
#2: discriminative training methods for hidden markov models theory and experiments with perceptron algorithms
#3: trn1 a statistical part of speech tagger
#4: enriching the knowledge sources used in a maximum entropy part of speech tagger
#5: estimators for stochastic unification based grammars
#6: equations for part of speech tagging
#7: classifier combination for improved lexical disambiguation
#8: conditional structure versus conditional estimation in nlp models
#9: a second order hidden markov model for part of speech tagging
#10: part of speech tagging based on hidden markov model assuming joint

