

搜索引擎算法设计

现有引擎的问题

基于现有的数据库结构，目前的搜索引擎算法有两个局限：

- 仅支持对标题的搜索，无法支持对blog_question-content和blog_answer的匹配
- 搜索算法采用string match模式，复杂度较高

针对第一个问题，现在数据库里存放模式是：两张表 blog_question和blog_answer。

Blog questions:

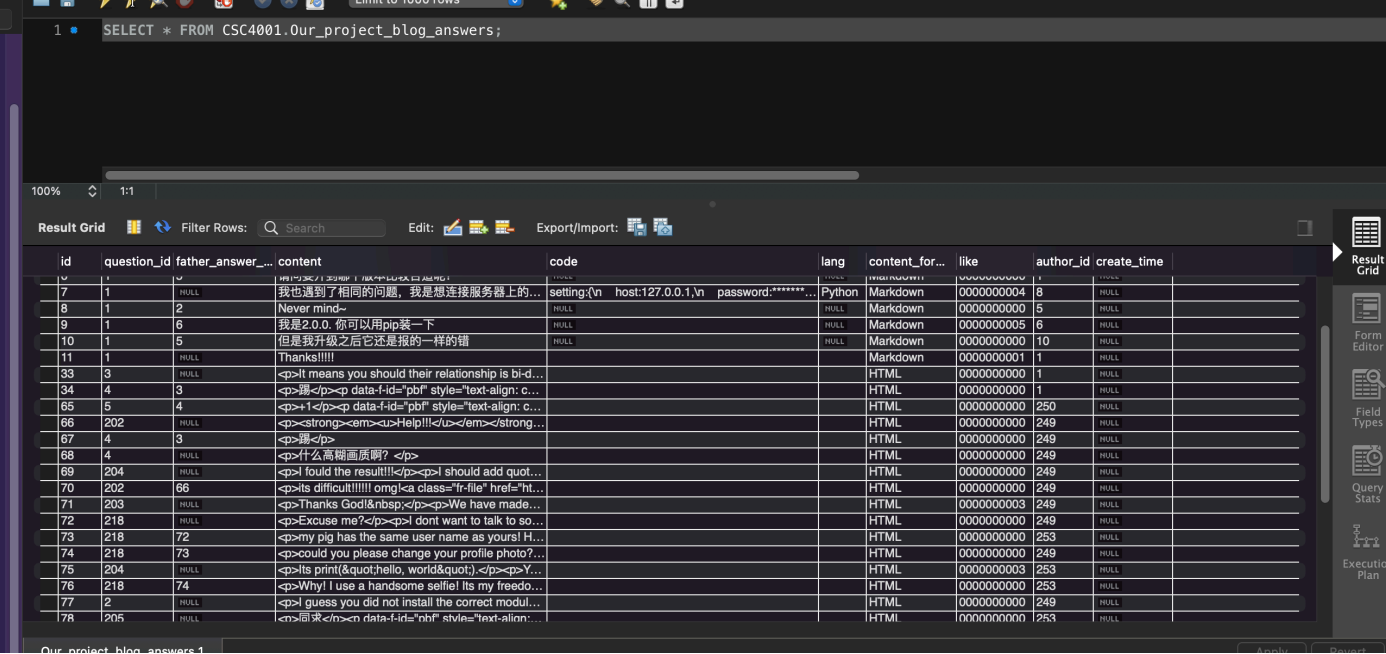
1 • SELECT * FROM CSC4001.Our_project_blog_questions;

100% 1:1

Result Grid Filter Rows: Search Edit: Export/Import:

id	title	author_id	group_type	sub_group_ty...	content	code	lang	content_for...	like	
▶ 1	Question about the Django settings.py	1	CSC4001	1	#### This question annoys me a lot --- + I am...	DATABASES = { 'default':{ 'ENGINE':'djan...	Python	Markdown	0000	
2	Anybody know how to solve this problem of np...	2	CSC4001	1	When I using the npm run dev, it gives me an er...			Markdown	0000	
3	What's the meaning of two-direction association...	1	CSC4001	2	Actually, I don't know what's the meaning of two...			Markdown	0000	
4	Building Kernel fail.	3	CSC3150	3	When I run make, it gives me a error about 'per...			Markdown	0000	
5	How many storage should I assign to VM?	4	CSC3150	3	Is 10GB assigned to VM enough? cause my co...			Markdown	0000	
202	How to write a kernel using Rust??	249	CSC3150	6	<p>We are currently attending an OS competi...	fn main(){ println("Hello world"); }	Rust	HTML	0000	
203	About 4001 Project	9	CSC4001	1	<p>I find this project very hard, is there any brilli...			C++ HTML	0000	
204	Hello to run code on Python...	249	CSC1001	24	<p>我完全是一个代码新手, <p><p>请问为什...	print (hello, world)	Python	HTML	0000	
205	How to learn c++	249	CSC3150	8	<p>I dont know how to learn c++<p><p>can s...	#include <iostream> int main(){ std::cout << "...	C++	HTML	0000	
218	How to raise a pig?	253	CSC1001	24	<p>In this blog, I will show you how it is like to b...	# I use python. print("but i would love to learn c+...	Python	HTML	0000	
258	What is the ddl of 4001 project	250	CSC4001	1	<p>as the title?<p><p data-f-id="pbf" style="tex...			C++ HTML	0000	
262	求CSC3002的syllabus	3	CSC3002	10	请问有上过这门课的学长学姐能分享一下Syllabus...			C	HTML	0000
263	what is the segmentation fault	249	CSC3002	11	<p>Currently I am learning CSC3002, Introduci...	#include <stdio.h> int main(){ char *p; p = ...	C	HTML	0000	
284	how to write code in python?	261	CSC1001	24	<p>I am currently a fresh year student, and I wa...	print("Hello world")	Python	HTML	0000	
315	Can 4001 help all students find a good internship?	9	CSC4001	1	<p id="isPasted">Recently, I am teaching the co...	name = 'golbal' var obj = { name: 'local', p: fu...	Javas...	HTML	0000	
322	this is a test	269	CSC4001	1	<p><u>this is test</u></str...			C++ HTML	0000	

Blog answer:



The screenshot shows a database query result for 'SELECT * FROM CSC4001.Our_project_blog_answers;'. The result is displayed in a table with columns: id, question_id, father_answer..., content, code, lang, content_for..., like, author_id, and create_time. The table contains 18 rows of data, including answers to questions about Django settings, kernel development, and learning Python/C++.

id	question_id	father_answer...	content	code	lang	content_for...	like	author_id	create_time
7	1		我也遇到了相同的问题,我是想连接服务器上的...	setting:\n host:127.0.0.1\n password:*****	Python	Markdown	0000000004	8	
8	1		Never mind~			Markdown	0000000000	5	
9	1		我是2.0.0, 你可以用pip装一下			Markdown	0000000005	6	
10	1		但是我升级之后它还是报的一样的错			Markdown	0000000000	10	
11	1		Thanks!!!!			Markdown	0000000001	1	
33	3		<p>It means you should their relationship is bi-d...			HTML	0000000000	1	
34	4		<p>踢<p><p data-f-id="pbf" style="text-align: c...			HTML	0000000000	1	
65	5		<p>+1<p><p data-f-id="pbf" style="text-align: c...			HTML	0000000000	250	
66	202		<p><u>Help!!!</u>			HTML	0000000000	249	
67	4		<p>踢<p>			HTML	0000000000	249	
68	4		<p>什么高糊画质啊? <p>			HTML	0000000000	249	
69	204		<p>I fould the result!!!!<p><p>I should add quot...			HTML	0000000000	249	
70	202		<p>Its difficult!!!!!! omg<a class="fr-file" href="ht...			HTML	0000000000	249	
71	203		<p>Thanks God! <p><p>We have made...			HTML	0000000003	249	
72	218		<p>Excuse me?<p><p>I dont want to talk to so...			HTML	0000000000	249	
73	218		<p>my pig has the same user name as yours! H...			HTML	0000000000	253	
74	218		<p>could you please change your profile photo?...			HTML	0000000000	249	
75	204		<p>Its print("hello, world")<p><p>Y...			HTML	0000000003	253	
76	218		<p>Why! I use a handsome selfie! Its my freedo...			HTML	0000000000	253	
77	2		<p>I guess you did not install the correct modul...			HTML	0000000000	249	
78	205		<n>同求<n><p data-f-id="pbf" style="text-align: ...			HTML	0000000000	253	

如果存储的模式不发生改变的话，那么需要对两张表的内容同时进行搜索，I/O的读写次数很高，或者采用join的办法合并表格，但是合并表格的开销同样不小。

另外，content内容返回的是html格式，需要对html格式进行解码操作，去除html和css标识符，得到数据字符串。

针对第二个问题，目前的搜索算法如下：

```
433         DBlist.append([title, title].upper())
434
435         if (scope=='All'):
436             # start to search
437             for DBitem in DBlist:
438                 similarity = 0
439                 for title in questionElem:
440                     if title in DBitem:
441                         similarity += 1
442                 answerList.append(similarity)
443             #answerList contains all similarity in the order of id
```

在全局搜索范围下，将数据库里的每一个关键词（key）与标题分词后所出现的关键词进行匹配，如果数据库的关键词（key）包含了标题的关键词，其相似度加1，最终返回一个相似度的排序数组，选取相似度高所对应的blog

这种算法的复杂度是O(N * k), N是数据库里的所有关键词，k是标题的关键词。如果后期将blog和blog answer的内容考虑进来的话，代表每一次全局搜索的操作都需要将数据库里所有的字符全部扫一遍，这是一个相当大的搜索量。

解决方案

基于第一个问题，如果需要对blog content和blog answer的搜索，则需要对这两个分别进行扫描，或者在初始化搜索引擎的时候将其存入对应的数据结构。同时，由于这两个的存放形式是HTML格式，需要对其进行解码操作。

HTML格式的解码操作不复杂，python3里有自带的HTML.parser对其进行解码，

所以主要的问题还是数据结构的存放问题，和第二个问题本质都是构建一个搜索引擎。

搜索引擎的构建

解决这一问题的思路是构建反向索引(inverted index). 其具体流程如下：

首先如果采用正向搜索，举例

1	How to build kernel
2	How to learn C++
3	How about my code below

如果用户输入 how，那么这样的搜索引擎需要把数据库里(1,3)的数据全部扫一遍，工作量很大

所以我们构建一个反向索引(inverted index),形如下：

How	{1,2,3}
Build	{2}
Learn	{2}
code	{3}
..	..

那么用户在搜索How的时候，可以直接返回blog_id = {1,2,3},这样的搜索效率高，不需要对数据库进行扫描，缺点是

- 需要在每次创建一个blog的时候，需要对blog的所有内容进行扫描并且分词，然后将每个词插入到对应的索引表当中去，同时需要创建和维护索引表
- 搜索的复杂度虽然降低（因为一张索引表的词理论上常用的不会超过1w个，并且除去那些the, a, an等词),但是一张索引表的消耗的空间同样很大
- 创建blog的开销增大，但是此过程可以对用户不透明

这样一张索引表的创建和维护流程如下：

对数据库已有的数据建表：

创建一张表格存在数据库里，形如下：

index id	key word	key word in title	Key word in content	Update time
1	How	{1,2}, {tilte_id, number of occurance}	{12, 3}, {tilte_id, number of occurance}	0
2	Build	{1,2}, {tilte_id, number of occurance}	{12, 3}, {tilte_id, number of occurance}	1

搜索：

1. 创建一个空的字典数组，里面存放{文章id (key) ， 文章相似度 (value) }
2. 线性搜索关键词，如果搜索命中,则往字典里面添加{文章id， 文章相似度}，如果字典里存在文章id，那么修改文章相似度。如果字典里没有文章id，则为其创建一个。在title和在content里的命中具有不同的占比，比例可设置为3:1，如果在tilte里命中，可以相似度加3，在content里命中，则相似度+1。如果整个表里无一命中，则返回空，表示搜索不到结果。
3. 根据字典里的value值排序，选取前面几个相似度高的文章id，返回给前端

理论上每次的搜索复杂度是O(N)，N是索引表的长度，如果索引表长度较长，则可以考虑建立二级索引。比如一级索引根据A-Z的首字母排列，二级索引储存所有的关键词。

维护索引表：

1. 索引表的增加。当有新的blog输入的时候，需要对blog的内容进行分词切割和存储。分词的时候需要除去对应的a, an, she, he等词，保留key word。然后将所有的关键词存入索引表中

2. 索引表的修改，当有blog加入的时候，对应的关键词已经在索引表中出现，这个时候需要向索引表中添加对应的tuple进行修改
3. 索引表的删除，当有blog删除的时候，需要修改（暂时不支持删除，所以暂时不考虑）

问题：每次对关键词进行增加和修改的时候，每一个关键词都需要扫描一整张索引表来判断是否在里面，所以在考虑是否需要建立一个二级索引表来优化算法，或者对索引表采用其他的排序方式来适应一个二分查找的算法

分词

分词是比较复杂的一部分，一般采用**TF-IDF**的原则，选取其中重要的词汇，也可以实现建立一个表格储存所有的不重要词汇，例如[the, a, an, He, She, It]等等

Github也有一些分词器可以直接拿来使用：

<https://github.com/stanfordnlp/CoreNLP.git>