بسم الله الرحمن الرحیم

حل تمرینات سری دوم مبانی جستجوی اطلاعات

دکتر مشایخی

مصطفی فضلی شهری - ۹۸۲۲۸۰۳

دانشگاه صنعتی شاهرود

آبان ۱۴۰۰

حل تمرینات سری دوم

1. در نمایه Permuterm هر permuterm vocabulary term به عبارت (یا عبارتهای) واژگان اصلی که از آن مشتق شده است، اشاره میکند (در واقع لیست پست این کلمه شامل کلمه اصلی است که از آن مشتق شده است.)

حال چند عبارت واژگان اصلی میتواند در لیست پست یک permuterm vocabulary term وجود داشته باشد؟

حل تمرینات سری دوم

تنها یک عبارت اصلی کافی است اما اگر نماد $ در عبارت به کار رفته باشد می توان چندین حالت از عبارت اصلی نظیر معکوس آن و... را در نظر گرفت.

مثلا اگر کلمه ای داشته باشیم که به صورت persian نوشته شده باشد معکوس آن که naisrep می باشد را باید در نظر بگیریم.

حل تمرینات سری دوم

2. کلمات دیکشنری نمایه permuterm را برای کلمه sing بنویسید و بگویید برای پرس و جوی s*ng چه چیزی جستجو شود؟

برای کلمه sing با توجه به تعداد حروف داریم :

sing$, ings , ngsi , g$sin , $sing

برای یافتن عبارت s*ng هم می توان ng را در ابتدا و سپس بخش موهومی را آورد، بدین صورت :

ng$s*

حل تمرینات سری دوم

3. داده ساختارها اصلی برای جستجوی کلمات نمایه (دیکشنری) را نام ببرید و توضیح دهید چه ضوابطی را هنگام استفاده از آنها باید در نظر گرفت؟

دو داده ساختار اصلی هش و درخت داریم که هر دو به جهاتی استفاده می شوند:

حل تمرینات سری دوم

هش (Hash):

- هر کلمه به یک integer هش شده و از تداخل جلوگیری میکند.
- هنگام کوئری زدن انجام می‌شود، یعنی : هر کوئری ترم هش می شود، تداخل ها رفع می شوند و در آخر به آرایه اضافه می شود.
- از مزایای آن می توان سرعت بالاتر نسبت به درخت را نام برد.
- از معیاب آن سرعت پائین برای یافتن کلمات همریشه و پیش بینی نکردن کلمات و نیافتن کلمات مشابه و نیاز به هش دوباره هنگام گسترش کلمات اشاره کرد.

حل تمرینات سری دوم

درخت (Tree):

درخت ها راه حل مناسبی برای عبارات پیشوندی می باشند.

از ساده ترین آن ها می توان به درخت باینری اشاره کرد.

جستجوی در آن کندتر از هش است و از مرتبه زمانی O(logM) که M سایز کلمات است می باشد.

نیاز به بالانس دوباره درخت باینری هزینه زیادی (از نظر زمانی و حافظه) دارد که از B-Tree برای رفع این مشکل استفاده می کنند.

حل تمرینات سری دوم

اگر نیاز به پیش بینی کلمات یا جستجوی کلمات مشابه داشته باشیم و همه عبارات رو یکجا نداشته باشیم و هر بار نیاز به ایندکس کردن داشته باشیم، استفاده از درخت بهینه تر است اما اگر سرعت ملاک کار قرار بگیرد و عبارات را داشته باشیم و نیاز به هش دوباره برای کلمات نداشته باشیم و پیش بینی کلمات یا استفاده از پیشوند و پسوند کلمات اهمیت نداشته باشد، استفاده از هش کردن بهتر عمل می کند.

(به نظر استفاده از راه حلی که ترکیب این دو باشد، راه حل خوبی باشد، اما نیاز به محاسبه بیشتر دارد، اما عملکرد بهتری را نشان می دهد)

حل تمرینات سری دوم

4. فاصله کلمات roof و road را توسط الگوریتم levenshtein بیابید.

		r	o	a	d
	0	1	2	3	4
r	1	0	1	2	3
o	2	1	0	1	2
o	3	2	1	1	2
f	4	3	2	2	2

حل تمرینات سری دوم

فاصله دو عبارت roof و road با استفاده از این روش 2 به دست می آید که از تغییر ad از road می توان به این فاصله دست یافت.

حل تمرینات سری دوم

5. ضریب جاکارد را برای دو جمله ی زیر بدست آورید.

- Query: shahrood university of technology
- Document: the main field of our university is technology.

- A = shahrood university of technology
- B = the main field of our university is technology.

$$J = \frac{A \cap B}{A \cup B} = \frac{3}{9} = \frac{1}{3}$$

حل تمرینات سری دوم

6. Variable byte codeرا برای posting list زیر حساب کنید. در صورت امکان به جای DocIDاز gapها استفاده کنید.

- Posting list(777,17743,294068,3125136)

Gap method :

- **17,743 – 777 = 16,966**
- **294,068 – 17,743 =276,325**
- **3,125,136 – 294,068 =2,831,068**

به دلیل فاصله زیاد گپ ها از یکدیگر(بیش از 20) صرفا استفاده از این روش بهینه به نظر نمی رسد، اما ترکیب آن با روش VB code بهینه تر به نظر می رسد.

حل تمرینات سری دوم

Byte Code method:

777 =>001100001001 => 0001100001001

16,966 => 011010100011 => 0000011010100011

276,325 => 010000110111011100101 => 0010000110111011100101

2,831,068 => 001010110011001011011100 => 0000000100101100011001011011100

حل تمرینات سری دوم

7. میزان فضای مورد نیاز مجموعه داده Reuters با بلاک هایی با اندازه k=8 و k=16 با در نظر گرفتن دیکشنری به صورت رشته با بلوک را حساب کنید.

- K = 8

12 – 3 + 8 = 1 bytes per block

400,000 / 8 * 1 = 50,000 = 0.05 MB

7.6 – 0.05 = 7.55 MB

- K = 16

12 – 3 + 16 = 25 bytes per block

400,000 / 16 * 25 = 625,000 = 0.625 MB

7.6 – 0.625 = 6.975 MB

تمرین پیاده سازی

فایل های دیتاست همراه تمرین را به وسیله Luceneایندکس کنید. این دیتاست شامل چکیده و مقدمه تعدادی مقاله علمی در سایت arxiv می باشد. همه حروف را کوچک کنید و کلمات ایست را حذف کنید. ریشه یابی را به دلخواه خود می توانید انجام دهید. سپس حداقل پنج پرس و جو مطرح کنید که عملگرهای not، or، andو جستجوی عبارت و پرس و جوی wildcardرا شامل شوند و نتایج را بررسی کرده و گزارش کنید که آیا نتایج و رتبه بندی آنها منطقی هستند یا خیر. می توانید برای سادگی تنها بخش چکیده abstractفایل ها را ایندکس نمایید.

- تمرین پیاده سازی با همه شرایط ذکر شده با استفاده از زبان جاوا در پیوست تمرین آمده است.
- با توجه به پیاده سازی، پاسخ کوئری ها با دقت بسیار خوبی به جواب های مدنظر نزدیک است و گویا این کد منطقی است و به خوبی می توان عمل جستجو در بین متن ها را انجام دهد.

تمرین پیاده سازی

- assimilated and task => just doc1
- assimilated or task => more than 20 documents (ranking available)
- Not machine => more than 20 documents (ranking available)
- Mach* => more than 20 documents (ranking available)
- Conformal => just document 61