Chuong Vu

Homework 7

Database I

5.9
Show how to express **group by cube**(*a, b, c, d*) using **rollup**; your answer should have only one **group by** clause.

```
groupby rollup(a), rollup(b), rollup(c ), rollup(d)
```

20.4
Consider the schemadepicted in Figure 20.2. Give an SQL query to summarize sales numbers and price by store and date, along with the hierarchies on store and date.

```
select store-id, city, state, country, date, month, quarter, year,
                              sum(number), sum(price)
from sales, store, date_info
where sales.store-id = store.store-id
and sales.date = date_info.date
groupby rollup(country, state, city, store-id),
     rollup(year, quarter, month, date)
```

20.13
Suggest how predictive mining techniques can be used by a sports team, using your favorite sport as an example.

Assume we have a data of all information about all game (score, player score) about a sport event. So, based on that information, we can predictive data mining.

- Can use to pick player based on the performance of the player.
- Predict the final score before the game.
- Give the strategies against each player in a game (pros and cons of each player)
-

21.2
Suppose you want to find documents that contain at least *k* of a given set of *n* keywords. Suppose also you have a keyword index that gives you a (sorted) list of identifiers of documents that contain a specified keyword. Give an efficient algorithm to find the desired set of documents.

Let S be an set on n keywords.
d = document identifer in D.
R = record of d in L

Psedocode:

```
Let L be an empty list;
for c = 1 to S do
begin
        for d = 1 to D do
                if R.document_id $\subseteq$L then
                        R.reference_count = R.reference_count + 1;
                else begin
                        make a new record R;
                        R.document_id = d;
                        R.reference_count = 1;
                        add R to L;
                end;
end;

for R=1 to L do
        if R.reference_count >= k then
                output R;
```

21.6
Using a simple definition of term frequency as the number of occurrences of the term in a document, give the TF–IDF scores of each term in the set of documents consisting of this and the next exercise.

$$TF(d.\,t) = \log\left(1 + \frac{n(d,t)}{n(d)}\right)$$

where *n(d)* denotes the number of terms in the document and *n(d, t)* denotes the number of occurrences of term *t* in the document *d*.

Let combine two paragraphs. We have total **75** words.

Using a simple definition of term frequency as the number of occurrences of the term in a document, give the TF–IDF scores of each term in the set of documents consisting of this and the next exercise. Create a small example of four small documents, each with a PageRank, and create inverted lists for the documents sorted by the PageRank. You do not need to compute PageRank, just assume some values for each page.

Word	TF-IDF	Word	TF-IDF	Word	TF-IDF
using	log(1 + 1/75)	scores	log(1 + 1/75)	inverted	log(1 + 1/75)
a	log(1 + 5/75)	each	log(1 + 3/75)	lists	log(1 + 1/75)
simple	log(1 + 1/75)	set	log(1 + 1/75)	sorted	log(1 + 1/75)
definition	log(1 + 1/75)	documents	log(1 + 3/75)	by	log(1 + 1/75)
of	log(1 + 6/75)	consisting	log(1 + 1/75)	you	log(1 + 1/75)
term	log(1 + 3/75)	this	log(1 + 1/75)	do	log(1 + 1/75)
frequency	log(1 + 1/75)	and	log(1 + 2/75)	not	log(1 + 1/75)
as	log(1 + 1/75)	next	log(1 + 1/75)	need	log(1 + 1/75)
the	log(1 + 1/75)	exercise	log(1 + 1/75)	to	log(1 + 1/75)
number	log(1 + 7/75)	create	log(1 + 1/75)	compute	log(1 + 1/75)
occurrences	log(1 + 1/75)	small	log(1 + 2/75)	just	log(1 + 1/75)
in	log(1 + 2/75)	example	log(1 + 1/75)	assume	log(1 + 1/75)
document	log(1 + 1/75)	four	log(1 + 1/75)	some	log(1 + 1/75)
give	log(1 + 1/75)	with	log(1 + 1/75)	values	log(1 + 1/75)
TF-IDF	log(1 + 1/75)	PageRank	log(1 + 3/75)	page	log(1 + 1/75)

✓

21.9

Web sites that want to get some publicity can join a Web ring, where they create links to other sites in the ring, in exchange for other sites in the ring creating links to their site. What is the effect of such rings on popularity ranking techniques such as PageRank?

Based on the PageRank defination (*PageRank works by counting the number and quality of links to a page to determine a rough estimate of how important the website is*). We know that when the website jon to a Web ring for the exchange, the number of pages in PageRank will increases. Also, the number of linkes referencing to each page increases. So, by all of this, the PageRank algorithm also increases (*running time*).

2/2