

DAIS Qualifying Examination  
*Spring 2006*

Department of Computer Science  
University of Illinois at Urbana-Champaign

February 13, 2006  
Time Limit: 180 minutes

- You are required to answer the **one** required question and **five** other questions.
- The required question will test the examinee's understanding of the basic concepts from all the three subareas of the exam. The required question has **10** subquestions.
- You are required to answer other **5** questions out of ten. If you answer more than five questions, the committee will randomly select five to grade. The distribution of the questions by areas is as follows.
  - **Database Systems:** 4 questions
  - **Data Mining:** 3 questions
  - **Information Retrieval:** 3 questions

For each such question, there are three subquestions. The first is to test the basic knowledge related to a topic, the second is to test current research related to the topic, and the third is to test research capability.

- Answer one question on a booklet. With one required and five other questions, you will use six booklets for your answers.
- To ensure the fairness of grading, all the exams will be kept anonymous. Your answer sheets should only bear the assigned ID but *no names*.
- Please make your answers clear and succinct; you will lose credit for verbose, convoluted, or confusing answers. *Simplicity does count!*

## Required Question (Basic Concepts): Problem 0

Short answers: The answer of each of the following questions is expected to be no more than two lines (i.e., no explanation nor reasoning is needed).

- (1) (*benchmarks*) In which benchmark would you expect to find the query below: TPC-C, TPC-H, or both? Why?

```
select l_orderkey, o_orderdate, o_shippriority, sum(l_extendedprice*(1-l_discount)) as revenue
from customer, orders, lineitem
where c_mktsegment= 'automobile'
      and c_custkey = o_custkey
      and l_orderkey = o_orderkey
      and o_orderdate < date '1995-03-15'
      and l_shipdate > date '1995-03-08'
group by l_orderkey, o_orderdate, o_shippriority
order by revenue desc, o_orderdate;
```

- (2) (*object-oriented and object-relational data models*) Compare and contrast the approaches taken to support *extensibility* in (a) object-oriented DBMSs and (b) object-relational DBMSs. Limit your answer to at most half a page of text.

- (3) (*two-phase commit*) What happens if a client fails during the two-phase commit algorithm?

- (4) (*B-trees*) (No one got this problem right on the Fall 2005 qual, so we will repeat it here so that you can show us how much smarter you are than the qual-takers last semester.)

Suppose that you want to index fixed-length records, each 256 bytes long. The key of each record is 4 bytes long. Records are stored on disk so that they do not cross page boundaries. If you use a three-level B+ tree to index this file, with the records themselves occupying the third level of the tree, then what is the maximum number of records that you can index if the B+ tree pointers are 4 bytes long and the B+ tree nodes are 4096 bytes long?

- (5) (*data cube and OLAP*) List (the names of) four methods, each best for the following task: (1) *compute dense cubes with 4-5 dimensions*, (2) *compute iceberg cubes of 7-10 dimensions of highly skewed data*, (3) *facilitate OLAP on high-dimensional data sets*, and (4) *facilitate OLAP on dynamic changing data streams*. (Note: write one-line description if you cannot remember the name of a method.)

- (6) (*frequent-pattern mining*) List four *frequent-pattern mining* methods, each having one of the following features: (1) candidate generation-and-test, (2) vertical data format, (3) depth-first search, and (4) efficient mining of closed frequent patterns.

- (7) (*classification*) Use one sentence for each method to characterize and distinguish the following four *classification* methods: (1) Rainforest, (2) SVM, (3) CBA, and (4) KNN.

- (8) (*TF-IDF weighting*) Let  $N$  be the total number of documents in a collection and  $n$  ( $n \leq N$ ) be the number of documents in which the word “computer” occurs. Suppose the word “computer” occurs  $k$  times in a document of length  $L$ . Write down one possible formula for computing the TF-IDF weight of “computer” in document  $D$ .

- (9) (*language model smoothing*) In Dirichlet Prior smoothing of a document language model, is it possible for a word unseen in the document to have a larger probability than a word seen in the document? Use no more than two sentences to explain why.
- (10) (*PageRank*) When PageRank is interpreted as computing the stationary distribution of a random walk, what is the meaning of the “damping factor”? Can we set it to zero? Why? Use one sentence to answer each question.

# Database Systems: Problem 1. Privacy Support

## Part 1

List the three pieces of auxiliary information that the Hippocratic Databases paper says that an on-line store like Mississippi needs to store for its customer records. (*1 points*)

## Part 2

What are the advantages and disadvantages of storing this information on a per record basis, versus a per field-in-a-record basis? (*2 points*)

## Part 3

The authors say that some of the record-level checks needed to implement their scheme can be converted into metadata checks. They go on to say that we need to understand under what conditions these checks can be compiled away or have their number reduced. Give your answer to this open problem, using a Mississippi-like database for sales over the internet to illustrate your answer: tables customer(customerID, name, shipping address, email, credit card info) and order(customerID, transactionID, bookInfo, status). *Hint*: The more checks you manage to compile away, the more we will like your answer. (*7 points*)

## Database Systems: Problem 2. Focused Crawling

### Part 1

For focused crawling (as described in the paper “The BINGO! System for Information Portal Generation and Expert Web Search”) to work successfully, the target domain must possess certain characteristics. Discuss these characteristics, and for each of them, give an example of a domain that does not have that characteristic.

### Part 2

Suppose you want to construct Citeseer, the online system that tracks paper citations. Explain how focused crawling can help in building such a system. In practice, do you think Citeseer’s builders did use focused crawling? Why or why not? If not, can you sketch a solution that you believe would work better than focused crawling?

### Part 3

1. The paper “The BINGO! System for Information Portal Generation and Expert Web Search” discusses focused crawling to retrieve relevant pages on the World Wide Web. Do you think we can have a similar focused crawling problem, but across heterogeneous relational databases? If yes, sketch a motivating example.
2. Now sketch a solution to the above problem. Discuss its challenges.

## Database Systems: Problem 3. Top-K Queries

### Part 1

Design a small example to illustrate how the TA algorithm works, given a scoring function  $F(u) = \text{average}(u.a, u.b)$  for a tuple  $u$  and its attributes  $a$  and  $b$ . (3 points)

### Part 2

The TA algorithm is said to be *instance optimal*. Suppose we would like to argue that instance optimality is a rather *strong* sense of optimality. Give your argument to support. (2 points)

### Part 3

The TA algorithm (and its family of algorithms) is designed for *middleware* environments. Suppose we want to support such “top-k” queries in a *relational* database, in SQL queries. For instance, consider the following query for retrieving top 5 answers from a Student table:

```
select u.name
from Student u
where u.dept = "cs" and u.year = 4 //i.e., senior year CS students.
order by average(u.a, u.b)
limit 5 //i.e., only top-5 answers.
```

Is the TA algorithm applicable to answer this query in a relational DBMS? Explain why or why not. (5 points)

## Database Systems: Problem 4. Wrapper Construction

### Part 1

1. Explain what is a *wrapper*. (1 points)
2. Wrapper construction is a major barrier for large scale integration– i.e., to integrate a large number of data sources. Explain why. (1 points)

### Part 2

Briefly explain: What is the main insight that makes RoadRunner work? (2 points)

### Part 3

RoadRunner gives a principled framework for wrapper generation, for a *single* source. However, in large scale integration, we often need to construct wrappers for *many* similar sources– e.g., Amazon.com and BN.com (Barnes&Noble) are both “book” sources. In such scenarios, we wonder if we can leverage a wrapped source to build a wrapper for a new but similar source.

Let’s consider a concrete scenario: Suppose you have built a perfect wrapper for Amazon.com– i.e., you have a wrapper for accessing Amazon.com’s book information, by executing a given query at Amazon.com and extracting its results. How will this existing wrapper help in building a new wrapper for BN.com? Describe one possibility and explain why. (6 points)

# Data Mining and Data Warehousing: Problem 1.

## Part 1

Assume a base cuboid of  $n$  dimensions contains  $n$  base cells, as shown below,

- (1)  $(b_1, a_2, a_3, \dots, a_n)$
- (2)  $(a_1, b_2, a_3, \dots, a_n)$
- (3)  $(a_1, a_2, b_3, \dots, a_n)$
- ...
- ( $n$ )  $(a_1, a_2, a_3, \dots, b_n)$

where  $a_i \neq b_i$  (for any  $i$ ). The measure of the cube is *count*. Let the condition of iceberg cube be “*count*  $\geq 2$ ”?

1. How many **nonempty** aggregated cells are there for all the iceberg cuboids of dimension  $(n - 2)$ ?
2. Answer the same question for the iceberg cuboids of dimension  $(n - 3)$ .

## Part 2

For network intrusion detection, one may like to detect substantial changes of certain measures of network data flow in multidimensional space. Design a data structure and computation method that may facilitate multi-dimensional analysis of measure changes in data streams.

- (a) Show your design requires limited space and can incrementally incorporate new incoming data streams, and
- (b) show how to perform efficient online detection of substantial changes in multi-dimensional space.

## Part 3

A chain store has collected data on the *effectiveness* of its promotion of certain new products in state  $A$  and found that the effectiveness is closely related to certain sensitive attributes in each region, such as (1) population density, (2) income level, (3) race distribution, (4) age distribution, (5) education level, (6) sales season, and (7) professional distribution. Discuss how the store can use such data to facilitate its effective promotion of the same new products in other states.

## Data Mining and Data Warehousing: Problem 2.

### Part 1

Someone says: “*The worst-case complexity of the existing frequent-pattern mining algorithms is exponential. Such algorithms only work at certain special conditions on special kinds of data.*” Give sufficient and clear arguments to support or rebut the statement.

### Part 2

Frequent-pattern mining algorithms can be extended to perform effective classification,.

- (a) Outline such an efficient frequent-pattern-based classification algorithm.
- (b) In comparison with a classical Naïve-Bayesian classification algorithm, what are the strength and weakness of the algorithm you outlined in Part 2(a)?

### Part 3

One often needs to mine frequent itemsets in data streams. However, due to the limited main memory space, it is often only realistic to mine *approximate* frequent itemsets in data streams.

- (a) Outline one efficient algorithm that used limited space to mine approximate frequent itemsets in long data streams, with a guaranteed error bound, and
- (b) Discuss how such an algorithm can be extended to find the *evolution* (with time) of approximate frequent itemsets in data streams.

## Data Mining and Data Warehousing: Problem 3.

### Part 1

- (a) Explain why a typical partition-based clustering algorithm, such as  $k$ -means algorithms, have difficulty to find arbitrary-shaped clusters.
- (a) Outline an algorithm that finds such arbitrary-shaped clusters efficiently.

### Part 2

In micro-array data analysis, each data object may contain tens of thousands of dimensions. Outline a method that can efficiently perform cluster analysis in such data sets.

### Part 3

A typical data relation, such as *Student*, may have dozens of attributes. Not all the attributes are relevant to a particular clustering task. A user may not know what are the relevant attributes but may provide one or two known relevant attributes, such as *Research\_group*, as a hint.

- (a) Outline an efficient clustering method that may take user's attribute hint to perform effective clustering of relational data.
- (b) Discuss how such a clustering method should be changed if a user gives a hint on a sample set of data (such as tuples  $A$  and  $B$  should be in the same cluster but not  $C$ ) instead of a sample set of attributes.

# Information Retrieval: Problem 1. Query Difficulty Prediction

## Part 1

- a (non-interpolated) Average Precision (AP) is commonly used to measure the ranking accuracy of an information retrieval system. Suppose a topic has 4 relevant documents and an IR system ranks these four relevant documents at rank 1, 2, 3, and 5, respectively. What is the Average Precision of this system for this topic?
- b State at least three reasons why predicting query difficulty is useful in information retrieval.

## Part 2

In [Yom-Tov et al. 2005], two approaches are proposed to predict query difficulty (i.e., the histogram method and the decision tree method).

- a Use no more than 5 sentences to explain what is the common basic idea behind both approaches. In particular, explain what kind of queries would be predicted as “difficult” by both approaches.
- b In the histogram method, the query difficulty prediction function can be written as a dot product  $Pred = c^T \cdot h$ . What is  $c$ ? What is  $h$ ? Suppose a query  $Q$  has 4 terms  $q_1$ ,  $q_2$ ,  $q_3$ , and  $q_4$ . The overlap between the top 5 results of these 4 query terms and the original query result is 2, 1, 1, 1 respectively. Assume that these four query terms have identical document frequency (DF). What is the highest element value in the vector  $h$ ? What is the meaning of this value? (Consider only those elements of  $h$  that correspond to the overlap of query-subquery results.)

## Part 3

In [Yom-Tov et al. 2005], the techniques for predicting query difficulty are applied to merge search results from multiple search engines. Can you suggest two different strategies for combining search engine results based on query difficulty weights? You may assume that each search engine would return a ranked list of results with scores and you are given a function  $w(q, E)$  to compute the difficulty of query  $q$  w.r.t. search engine  $E$ .

## Information Retrieval: Problem 2. Language Models for Retrieval

### Part 1

- a In the query likelihood retrieval method, we rank documents based on the log likelihood of query  $q$  given document  $d$ , i.e.,  $\log p(q|d)$ . We may view  $\log p(q|d)$  as a dot product of a query term weight vector and a document term weight vector. What are these two vectors? Assume that we smooth the document language model using Dirichlet prior smoothing method with parameter  $\mu$  and collection/background/reference language model  $p(w|C)$ . Also, use  $c(w, q)$  and  $c(w, d)$  to denote the count of word  $w$  in  $q$  and  $d$ , respectively.
- b TF-IDF weighting and document length normalization are in some sense *implicitly* implemented in the query likelihood retrieval method in the sense that similar scoring/weighting effects are also achieved when scoring a document based on the likelihood of the query given a document language model smoothed with a collection language model. Explain how exactly the query likelihood method can achieve each of the following retrieval heuristics (1) TF-weighting; (2)IDF-weighting; and (3) document length normalization.

### Part 2

- a The Linear Discriminant Model (LDM) proposed in [Gao et al. 2005] scores a document based on a linear combination of features. Give the general formula for using LDA to score a document  $d$  w.r.t. a query  $q$ , assuming that  $f_i(q, d)$  ( $i = 1, \dots, K$ ) defines a feature.
- b How can you define features so that the LDM scoring function would be a combination of unigram likelihood of the query and bigram likelihood of the query?

### Part 3

Can you suggest a way to use the Linear Discriminant Model (LDM) proposed in [Gao et al. 2005] to exploit the structure information to score a structured document?

# Information Retrieval: Problem 3. Implicit Feedback & Retrieval

## Part 1

- a What is implicit feedback? How is it different from relevance feedback and pseudo-relevance feedback?
- b According to the study of implicit feedback in [Shen et al. 2005], what implicit feedback information is most useful in improving retrieval accuracy? Can you give a negative example when using such information may hurt retrieval performance?

## Part 2

- a According to [Robertson et al. 2004], in which the authors proposed a simple extension to BM25 for scoring structured documents, what is the main argument for combining term frequencies, as opposed to combining scores of each field? In particular, why is combining scores of each field problematic?
- b According to the study of implicit feedback in [Shen et al. 2005], clickthroughs corresponding to *non-relevant* documents are still very useful for improving the retrieval accuracy of the current query. Is this what we should expect? Why?

## Part 3

- a Suggest a retrieval formula based on dot-product similarity and some kind of TF-IDF weighting, for which combining retrieval scores of different fields would be identical to combining the term frequencies in terms of ranking structured documents.
- b The proposed method in [Robertson et al. 2004] for combining term frequencies has a weight parameter for each field of a document. If we want to set these weight parameters, what factors should we consider? Can we simply give each field an equal weight? Why?